

Apostolos Antonacopoulos ·
Subhasis Chaudhuri · Rama Chellappa ·
Cheng-Lin Liu · Saumik Bhattacharya ·
Umapada Pal (Eds.)

LNCS 15330

Pattern Recognition

27th International Conference, ICPR 2024
Kolkata, India, December 1–5, 2024
Proceedings, Part XXX

30 Part XXX

ICPR
2024 INDIA



 Springer

MOREMEDIA 

Lecture Notes in Computer Science

15330

Founding Editors


Gerhard Goos
Juris Hartmanis

Editorial Board Members

Elisa Bertino, *Purdue University, West Lafayette, IN, USA*

Wen Gao, *Peking University, Beijing, China*

Bernhard Steffen , *TU Dortmund University, Dortmund, Germany*

Moti Yung , *Columbia University, New York, NY, USA*

The series Lecture Notes in Computer Science (LNCS), including its subseries Lecture Notes in Artificial Intelligence (LNAI) and Lecture Notes in Bioinformatics (LNBI), has established itself as a medium for the publication of new developments in computer science and information technology research, teaching, and education.


LNCS enjoys close cooperation with the computer science R & D community, the series counts many renowned academics among its volume editors and paper authors, and collaborates with prestigious societies. Its mission is to serve this international community by providing an invaluable service, mainly focused on the publication of conference and workshop proceedings and postproceedings. LNCS commenced publication in 1973.


Apostolos Antonacopoulos ·
Subhasis Chaudhuri · Rama Chellappa ·
Cheng-Lin Liu · Saumik Bhattacharya ·
Umapada Pal
Editors


Pattern Recognition

27th International Conference, ICPR 2024
Kolkata, India, December 1–5, 2024
Proceedings, Part XXX

Editors

Apostolos Antonacopoulos 
University of Salford
Salford, UK

Rama Chellappa 
Johns Hopkins University
Baltimore, MD, USA

Saumik Bhattacharya 
IIT Kharagpur
Kharagpur, India

Subhasis Chaudhuri 
Indian Institute of Technology Bombay
Mumbai, India

Cheng-Lin Liu 
Chinese Academy of Sciences
Beijing, China

Umapada Pal 
Indian Statistical Institute Kolkata
Kolkata, India

ISSN 0302-9743

ISSN 1611-3349 (electronic)

Lecture Notes in Computer Science

ISBN 978-3-031-78112-4

ISBN 978-3-031-78113-1 (eBook)

<https://doi.org/10.1007/978-3-031-78113-1>

© The Editor(s) (if applicable) and The Author(s), under exclusive license
to Springer Nature Switzerland AG 2025

This work is subject to copyright. All rights are solely and exclusively licensed by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

If disposing of this product, please recycle the paper.

President's Address

On behalf of the Executive Committee of the International Association for Pattern Recognition (IAPR), I am pleased to welcome you to the 27th International Conference on Pattern Recognition (ICPR 2024), the main scientific event of the IAPR.

After a completely digital ICPR in the middle of the COVID pandemic and the first hybrid version in 2022, we can now enjoy a fully back-to-normal ICPR this year. I look forward to hearing inspirational talks and keynotes, catching up with colleagues during the breaks and making new contacts in an informal way. At the same time, the conference landscape has changed. Hybrid meetings have made their entrance and will continue. It is exciting to experience how this will influence the conference. Planning for a major event like ICPR must take place over a period of several years. This means many decisions had to be made under a cloud of uncertainty, adding to the already large effort needed to produce a successful conference. It is with enormous gratitude, then, that we must thank the team of organizers for their hard work, flexibility, and creativity in organizing this ICPR. ICPR always provides a wonderful opportunity for the community to gather together. I can think of no better location than Kolkata to renew the bonds of our international research community.

Each ICPR is a bit different owing to the vision of its organizing committee. For 2024, the conference has six different tracks reflecting major themes in pattern recognition: Artificial Intelligence, Pattern Recognition and Machine Learning; Computer and Robot Vision; Image, Speech, Signal and Video Processing; Biometrics and Human Computer Interaction; Document Analysis and Recognition; and Biomedical Imaging and Bioinformatics. This reflects the richness of our field. ICPR 2024 also features two dozen workshops, seven tutorials, and 15 competitions; there is something for everyone. Many thanks to those who are leading these activities, which together add significant value to attending ICPR, whether in person or virtually. Because it is important for ICPR to be as accessible as possible to colleagues from all around the world, we are pleased that the IAPR, working with the ICPR organizers, is continuing our practice of awarding travel stipends to a number of early-career authors who demonstrate financial need. Last but not least, we are thankful to the Springer LNCS team for their effort to publish these proceedings.

Among the presentations from distinguished keynote speakers, we are looking forward to the three IAPR Prize Lectures at ICPR 2024. This year we honor the achievements of Tin Kam Ho (IBM Research) with the IAPR's most prestigious King-Sun Fu Prize "for pioneering contributions to multi-classifier systems, random decision forests, and data complexity analysis". The King-Sun Fu Prize is given in recognition of an outstanding technical contribution to the field of pattern recognition. It honors the memory of Professor King-Sun Fu who was instrumental in the founding of IAPR, served as its first president, and is widely recognized for his extensive contributions to the field of pattern recognition.

The Maria Petrou Prize is given to a living female scientist/engineer who has made substantial contributions to the field of Pattern Recognition and whose past contributions, current research activity and future potential may be regarded as a model to both aspiring and established researchers. It honours the memory of Professor Maria Petrou as a scientist of the first rank, and particularly her role as a pioneer for women researchers. This year, the Maria Petrou Prize is given to Guoying Zhao (University of Oulu), “for contributions to video analysis for facial micro-behavior recognition and remote bio-signal reading (RPPG) for heart rate analysis and face anti-spoofing”.

The J.K. Aggarwal Prize is given to a young scientist who has brought a substantial contribution to a field that is relevant to the IAPR community and whose research work has had a major impact on the field. Professor Aggarwal is widely recognized for his extensive contributions to the field of pattern recognition and for his participation in IAPR's activities. This year, the J.K. Aggarwal Prize goes to Xiaolong Wang (UC San Diego) “for groundbreaking contributions to advancing visual representation learning, utilizing self-supervised and attention-based models to establish fundamental frameworks for creating versatile, general-purpose pattern recognition systems”.

During the conference we will also recognize 21 new IAPR Fellows selected from a field of very strong candidates. In addition, a number of Best Scientific Paper and Best Student Paper awards will be presented, along with the Best Industry Related Paper Award and the Piero Zamperoni Best Student Paper Award. Congratulations to the recipients of these very well-deserved awards!

I would like to close by again thanking everyone involved in making ICPR 2024 a tremendous success; your hard work is deeply appreciated. These thanks extend to all who chaired the various aspects of the conference and the associated workshops, my ExCo colleagues, and the IAPR Standing and Technical Committees. Linda O’Gorman, the IAPR Secretariat, deserves special recognition for her experience, historical perspective, and attention to detail when it comes to supporting many of the IAPR’s most important activities. Her tasks became so numerous that she recently got support from Carolyn Buckley (layout, newsletter), Ugur Halici (ICPR matters), and Rosemary Stramka (secretariat). The IAPR website got a completely new design. Ed Sobczak has taken care of our web presence for so many years already. A big thank you to all of you!

This is, of course, the 27th ICPR conference. Knowing that ICPR is organized every two years, and that the first conference in the series (1973!) pre-dated the formal founding of the IAPR by a few years, it is also exciting to consider that we are celebrating over 50 years of ICPR and at the same time approaching the official IAPR 50th anniversary in 2028: you’ll get all information you need at ICPR 2024. In the meantime, I offer my thanks and my best wishes to all who are involved in supporting the IAPR throughout the world.

September 2024

Arjan Kuijper
President of the IAPR

Preface

It is our great pleasure to welcome you to the proceedings of the 27th International Conference on Pattern Recognition (ICPR 2024), held in Kolkata, India. The city, formerly known as ‘Calcutta’, is the home of the fabled Indian Statistical Institute (ISI), which has been at the forefront of statistical pattern recognition for almost a century. Concepts like the Mahalanobis distance, Bhattacharyya bound, Cramer–Rao bound, and Fisher–Rao metric were invented by pioneers associated with ISI. The first ICPR (called IJCPD then) was held in 1973, and the second in 1974. Subsequently, ICPR has been held every other year. The International Association for Pattern Recognition (IAPR) was founded in 1978 and became the sponsor of the ICPR series. Over the past 50 years, ICPR has attracted huge numbers of scientists, engineers and students from all over the world and contributed to advancing research, development and applications in pattern recognition technology.

ICPR 2024 was held at the Biswa Bangla Convention Centre, one of the largest such facilities in South Asia, situated just 7 kilometers from Kolkata Airport (CCU). According to ChatGPT “Kolkata is often called the ‘Cultural Capital of India’. The city has a deep connection to literature, music, theater, and art. It was home to Nobel laureate Rabindranath Tagore, and the Bengali film industry has produced globally renowned filmmakers like Satyajit Ray. The city boasts remarkable colonial architecture, with landmarks like Victoria Memorial, Howrah Bridge, and the Indian Museum (the oldest and largest museum in India). Kolkata’s streets are dotted with old mansions and buildings that tell stories of its colonial past. Walking through the city can feel like stepping back into a different era. Finally, Kolkata is also known for its street food.”

ICPR 2024 followed a two-round paper submission format. We received a total of 2135 papers (1501 papers in round-1 submissions, and 634 papers in round-2 submissions). Each paper, on average, received 2.84 reviews, in single-blind mode. For the first-round papers we had a rebuttal option available to authors.

In total, 945 papers (669 from round-1 and 276 from round-2) were accepted for presentation, resulting in an acceptance rate of 44.26%, which is consistent with previous ICPR events. At ICPR 2024 the papers were categorized into six tracks: Artificial Intelligence, Machine Learning for Pattern Analysis; Computer Vision and Robotic Perception; Image, Video, Speech, and Signal Analysis; Biometrics and Human-Machine Interaction; Document and Media Analysis; and Biomedical Image Analysis and Informatics.

The main conference ran over December 2–5, 2024. The main program included the presentation of 188 oral papers (19.89% of the accepted papers), 757 poster papers and 12 competition papers (out of 15 submitted). A total 10 oral sessions were held concurrently in four meeting rooms with a total of 40 oral sessions. In total 24 workshops and 7 tutorials were held on December 1, 2024.

The plenary sessions included three prize lectures and three invited presentations. The prize lectures were delivered by Tin Kam Ho (IBM Research, USA; King Sun

Fu Prize winner), Xiaolong Wang (University of California, San Diego, USA; J.K. Aggarwal Prize winner), and Guoying Zhao (University of Oulu, Finland; Maria Petrou Prize winner). The invited speakers were Timothy Hospedales (University of Edinburgh, UK), Venu Govindaraju (University at Buffalo, USA), and Shuicheng Yan (Skywork AI, Singapore).

Several best paper awards were presented in ICPR: the Piero Zamperoni Award for the best paper authored by a student, the BIRPA Best Industry Related Paper Award, and the Best Paper Awards and Best Student Paper Awards for each of the six tracks of ICPR 2024.

The organization of such a large conference would not be possible without the help of many volunteers. Our special gratitude goes to the Program Chairs (Apostolos Antonacopoulos, Subhasis Chaudhuri, Rama Chellappa and Cheng-Lin Liu), for their leadership in organizing the program. Thanks to our Publication Chairs (Ananda S. Chowdhury and Wataru Ohyama) for handling the overwhelming workload of publishing the conference proceedings. We also thank our Competition Chairs (Richard Zanibbi, Lianwen Jin and Laurence Likforman-Sulem) for arranging 12 important competitions as part of ICPR 2024. We are thankful to our Workshop Chairs (P. Shivakumara, Stephanie Schuckers, Jean-Marc Ogier and Prabir Bhattacharya) and Tutorial Chairs (B.B. Chaudhuri, Michael R. Jenkin and Guoying Zhao) for arranging the workshops and tutorials on emerging topics. ICPR 2024, for the first time, held a Doctoral Consortium. We would like to thank our Doctoral Consortium Chairs (Véronique Eglin, Dan Lopresti and Mayank Vatsa) for organizing it.

Thanks go to the Track Chairs and the meta reviewers who devoted significant time to the review process and preparation of the program. We also sincerely thank the reviewers who provided valuable feedback to the authors.

Finally, we acknowledge the work of other conference committee members, like the Organizing Chairs and Organizing Committee Members, Finance Chairs, Award Chair, Sponsorship Chairs, and Exhibition and Demonstration Chairs, Visa Chair, Publicity Chairs, and Women in ICPR Chairs, whose efforts made this event successful. We also thank our event manager Alpcord Network for their help.

We hope that all the participants found the technical program informative and enjoyed the sights, culture and cuisine of Kolkata.

October 2024

Umapada Pal
Josef Kittler
Anil Jain

Organization

General Chairs

Umapada Pal
Josef Kittler
Anil Jain

Indian Statistical Institute, Kolkata, India
University of Surrey, UK
Michigan State University, USA

Program Chairs

Apostolos Antonacopoulos
Subhasis Chaudhuri
Rama Chellappa
Cheng-Lin Liu

University of Salford, UK
Indian Institute of Technology, Bombay, India
Johns Hopkins University, USA
Institute of Automation, Chinese Academy of
Sciences, China

Publication Chairs

Ananda S. Chowdhury
Wataru Ohyama

Jadavpur University, India
Tokyo Denki University, Japan

Competition Chairs

Richard Zanibbi
Lianwen Jin
Laurence Likforman-Sulem

Rochester Institute of Technology, USA
South China University of Technology, China
Télécom Paris, France

Workshop Chairs

P. Shivakumara
Stephanie Schuckers
Jean-Marc Ogier
Prabir Bhattacharya

University of Salford, UK
Clarkson University, USA
Université de la Rochelle, France
Concordia University, Canada

Tutorial Chairs

B. B. Chaudhuri	Indian Statistical Institute, Kolkata, India
Michael R. Jenkin	York University, Canada
Guoying Zhao	University of Oulu, Finland

Doctoral Consortium Chairs

Véronique Eglin	CNRS, France
Daniel P. Lopresti	Lehigh University, USA
Mayank Vatsa	Indian Institute of Technology, Jodhpur, India

Organizing Chairs

Saumik Bhattacharya	Indian Institute of Technology, Kharagpur, India
Palash Ghosal	Sikkim Manipal University, India

Organizing Committee

Santanu Phadikar	West Bengal University of Technology, India
SK Md Obaidullah	Aliah University, India
Sayantari Ghosh	National Institute of Technology Durgapur, India
Himadri Mukherjee	West Bengal State University, India
Nilamadhaba Tripathy	Clarivate Analytics, USA
Chayan Halder	West Bengal State University, India
Shibaprasad Sen	Techno Main Salt Lake, India

Finance Chairs

Kaushik Roy	West Bengal State University, India
Michael Blumenstein	University of Technology Sydney, Australia

Awards Committee Chair

Arpan Pal	Tata Consultancy Services, India
-----------	----------------------------------

Sponsorship Chairs

P. J. Narayanan	Indian Institute of Technology, Hyderabad, India
Yasushi Yagi	Osaka University, Japan
Venu Govindaraju	University at Buffalo, USA
Alberto Bel Bimbo	Università di Firenze, Italy

Exhibition and Demonstration Chairs

Arjun Jain	FastCode AI, India
Agnimitra Biswas	National Institute of Technology, Silchar, India

International Liaison, Visa Chair

Balasubramanian Raman	Indian Institute of Technology, Roorkee, India
-----------------------	--

Publicity Chairs

Dipti Prasad Mukherjee	Indian Statistical Institute, Kolkata, India
Bob Fisher	University of Edinburgh, UK
Xiaojun Wu	Jiangnan University, China

Women in ICPR Chairs

Ingela Nystrom	Uppsala University, Sweden
Alexandra B. Albu	University of Victoria, Canada
Jing Dong	Institute of Automation, Chinese Academy of Sciences, China
Sarbani Palit	Indian Statistical Institute, Kolkata, India

Event Manager

Alpcord Network

Track Chairs – Artificial Intelligence, Machine Learning for Pattern Analysis

Larry O’Gorman	Nokia Bell Labs, USA
Dacheng Tao	University of Sydney, Australia
Petia Radeva	University of Barcelona, Spain
Susmita Mitra	Indian Statistical Institute, Kolkata, India
Jiliang Tang	Michigan State University, USA

Track Chairs – Computer and Robot Vision

C. V. Jawahar	International Institute of Information Technology (IIIT), Hyderabad, India
João Paulo Papa	São Paulo State University, Brazil
Maja Pantic	Imperial College London, UK
Gang Hua	Dolby Laboratories, USA
Junwei Han	Northwestern Polytechnical University, China

Track Chairs – Image, Speech, Signal and Video Processing

P. K. Biswas	Indian Institute of Technology, Kharagpur, India
Shang-Hong Lai	National Tsing Hua University, Taiwan
Hugo Jair Escalante	INAOE, CINVESTAV, Mexico
Sergio Escalera	Universitat de Barcelona, Spain
Prem Natarajan	University of Southern California, USA

Track Chairs – Biometrics and Human Computer Interaction

Richa Singh	Indian Institute of Technology, Jodhpur, India
Massimo Tistarelli	University of Sassari, Italy
Vishal Patel	Johns Hopkins University, USA
Wei-Shi Zheng	Sun Yat-sen University, China
Jian Wang	Snap, USA

Track Chairs – Document Analysis and Recognition

Xiang Bai	Huazhong University of Science and Technology, China
David Doermann	University at Buffalo, USA
Josep Lladós	Universitat Autònoma de Barcelona, Spain
Mita Nasipuri	Jadavpur University, India

Track Chairs – Biomedical Imaging and Bioinformatics

Jayanta Mukhopadhyay	Indian Institute of Technology, Kharagpur, India
Xiaoyi Jiang	Universität Münster, Germany
Seong-Whan Lee	Korea University, Korea

Metareviewers (Conference Papers and Competition Papers)

Wael Abd-Almageed	University of Southern California, USA
Maya Aghaei	NHL Stenden University, Netherlands
Alireza Alaei	Southern Cross University, Australia
Rajagopalan N. Ambasmudram	Indian Institute of Technology, Madras, India
Suyash P. Awate	Indian Institute of Technology, Bombay, India
Inci M. Baytas	Bogazici University, Turkey
Aparna Bharati	Lehigh University, USA
Brojeshwar Bhowmick	Tata Consultancy Services, India
Jean-Christophe Burie	University of La Rochelle, France
Gustavo Carneiro	University of Surrey, UK
Chee Seng Chan	Universiti Malaya, Malaysia
Sumohana S. Channappayya	Indian Institute of Technology, Hyderabad, India
Dongdong Chen	Microsoft, USA
Shengyong Chen	Tianjin University of Technology, China
Jun Cheng	Institute for Infocomm Research, A*STAR, Singapore
Albert Clapés	University of Barcelona, Spain
Oscar Dalmau	Center for Research in Mathematics, Mexico

Tyler Derr	Vanderbilt University, USA
Abhinav Dhall	Indian Institute of Technology, Ropar, India
Bo Du	Wuhan University, China
Yuxuan Du	University of Sydney, Australia
Ayman S. El-Baz	University of Louisville, USA
Francisco Escolano	University of Alicante, Spain
Siamac Fazli	Nazarbayev University, Kazakhstan
Jianjiang Feng	Tsinghua University, China
Gernot A. Fink	TU Dortmund University, Germany
Alicia Fornes	CVC, Spain
Junbin Gao	University of Sydney, Australia
Yan Gao	Amazon, USA
Yongsheng Gao	Griffith University, Australia
Caren Han	University of Melbourne, Australia
Ran He	Institute of Automation, Chinese Academy of Sciences, China
Tin Kam Ho	IBM, USA
Di Huang	Beihang University, China
Kaizhu Huang	Duke Kunshan University, China
Donato Impedovo	University of Bari, Italy
Julio Jacques	University of Barcelona and Computer Vision Center, Spain
Lianwen Jin	South China University of Technology, China
Wei Jin	Emory University, USA
Danilo Samuel Jodas	São Paulo State University, Brazil
Manjunath V. Joshi	DA-IICT, India
Jayashree Kalpathy-Cramer	Massachusetts General Hospital, USA
Dimosthenis Karatzas	Computer Vision Centre, Spain
Hamid Karimi	Utah State University, USA
Baiying Lei	Shenzhen University, China
Guoqi Li	Chinese Academy of Sciences, and Peng Cheng Lab, China
Laurence Likforman-Sulem	Institut Polytechnique de Paris/Télécom Paris, France
Aishan Liu	Beihang University, China
Bo Liu	Bytedance, USA
Chen Liu	Clarkson University, USA
Cheng-Lin Liu	Institute of Automation, Chinese Academy of Sciences, China
Hongmin Liu	University of Science and Technology Beijing, China
Hui Liu	Michigan State University, USA

Jing Liu	Institute of Automation, Chinese Academy of Sciences, China
Li Liu	University of Oulu, Finland
Qingshan Liu	Nanjing University of Posts and Telecommunications, China
Adrian P. Lopez-Monroy	Centro de Investigacion en Matematicas AC, Mexico
Daniel P. Lopresti	Lehigh University, USA
Shijian Lu	Nanyang Technological University, Singapore
Yong Luo	Wuhan University, China
Andreas K. Maier	FAU Erlangen-Nuremberg, Germany
Davide Maltoni	University of Bologna, Italy
Hong Man	Stevens Institute of Technology, USA
Lingtong Min	Northwestern Polytechnical University, China
Paolo Napoletano	University of Milano-Bicocca, Italy
Kamal Nasrollahi	Milestone Systems, Aalborg University, Denmark
Marcos Ortega	University of A Coruña, Spain
Shivakumara Palaiahnakote	University of Salford, UK
P. Jonathon Phillips	NIST, USA
Filiberto Pla	University Jaume I, Spain
Ajit Rajwade	Indian Institute of Technology, Bombay, India
Shanmuganathan Raman	Indian Institute of Technology, Gandhinagar, India
Imran Razzak	UNSW, Australia
Beatriz Remeseiro	University of Oviedo, Spain
Gustavo Rohde	University of Virginia, USA
Partha Pratim Roy	Indian Institute of Technology, Roorkee, India
Sanjoy K. Saha	Jadavpur University, India
Joan Andreu Sánchez	Universitat Politècnica de València, Spain
Claudio F. Santos	UFSCar, Brazil
Shin'ichi Satoh	National Institute of Informatics, Japan
Stephanie Schuckers	Clarkson University, USA
Srirangaraj Setlur	University at Buffalo, SUNY, USA
Debdoot Sheet	Indian Institute of Technology, Kharagpur, India
Jun Shen	University of Wollongong, Australia
Li Shen	JD Explore Academy, China
Chen Shengyong	Zhejiang University of Technology and Tianjin University of Technology, China
Andy Song	RMIT University, Australia
Akihiro Sugimoto	National Institute of Informatics, Japan
Qianru Sun	Singapore Management University, Singapore
Arijit Sur	Indian Institute of Technology, Guwahati, India
Estefania Talavera	University of Twente, Netherlands

Wei Tang	University of Illinois at Chicago, USA
Joao M. Tavares	Universidade do Porto, Portugal
Jun Wan	NLPR, CASIA, China
Le Wang	Xi'an Jiaotong University, China
Lei Wang	Australian National University, Australia
Xiaoyang Wang	Tencent AI Lab, USA
Xinggang Wang	Huazhong University of Science and Technology, China
Xiao-Jun Wu	Jiangnan University, China
Yiding Yang	Bytedance, China
Xiwen Yao	Northwestern Polytechnical University, China
Xu-Cheng Yin	University of Science and Technology Beijing, China
Baosheng Yu	University of Sydney, Australia
Shiqi Yu	Southern University of Science and Technology, China
Xin Yuan	Westlake University, China
Yibing Zhan	JD Explore Academy, China
Jing Zhang	University of Sydney, Australia
Lefei Zhang	Wuhan University, China
Min-Ling Zhang	Southeast University, China
Wenbin Zhang	Florida International University, USA
Jiahuan Zhou	Peking University, China
Sanping Zhou	Xi'an Jiaotong University, China
Tianyi Zhou	University of Maryland, USA
Lei Zhu	Shandong Normal University, China
Pengfei Zhu	Tianjin University, China
Wangmeng Zuo	Harbin Institute of Technology, China

Reviewers (Competition Papers)

Liangcai Gao	Da-Han Wang
Mingxin Huang	Yang Xue
Lei Kang	Wentao Yang
Wenhui Liao	Jiaxin Zhang
Yuliang Liu	Yiwu Zhong
Yongxin Shi	

Reviewers (Conference Papers)

Aakanksha Aakanksha
 Aayush Singla
 Abdul Muqet
 Abhay Yadav
 Abhijeet Vijay Nandedkar
 Abhimanyu Sahu
 Abhinav Rajvanshi
 Abhisek Ray
 Abhishek Shrivastava
 Abhra Chaudhuri
 Aditi Roy
 Adriano Simonetto
 Adrien Maglo
 Ahmed Abdulkadir
 Ahmed Boudissa
 Ahmed Hamdi
 Ahmed Rida Sekkat
 Ahmed Sharafeldeen
 Aiman Farooq
 Aishwarya Venkataramanan
 Ajay Kumar
 Ajay Kumar Reddy Poreddy
 Ajita Rattani
 Ajoy Mondal
 Akbar K.
 Akbar Telikani
 Akshay Agarwal
 Akshit Jindal
 Al Zadid Sultan Bin Habib
 Albert Clapés
 Alceu Britto
 Alejandro Peña
 Alessandro Ortis
 Alessia Auriemma Citarella
 Alexandre Stenger
 Alexandros Sopasakis
 Alexia Toumpa
 Ali Khan
 Alik Pramanick
 Alireza Alaei
 Alper Yilmaz
 Aman Verma
 Amit Bhardwaj

Amit More
 Amit Nandedkar
 Amitava Chatterjee
 Amos L. Abbott
 Amrita Mohan
 Anand Mishra
 Ananda S. Chowdhury
 Anastasia Zakharova
 Anastasios L. Kesidis
 Andras Horvath
 Andre Gustavo Hochuli
 André P. Kelm
 Andre Wyzykowski
 Andrea Bottino
 Andrea Lagorio
 Andrea Torsello
 Andreas Fischer
 Andreas K. Maier
 Andreu Girbau Xalabarder
 Andrew Beng Jin Teoh
 Andrew Shin
 Andy J. Ma
 Aneesh S. Chivukula
 Ángela Casado-García
 Anh Quoc Nguyen
 Anindya Sen
 Anirban Saha
 Anjali Gautam
 Ankan Bhattacharyya
 Ankit Jha
 Anna Scius-Bertrand
 Annalisa Franco
 Antoine Doucet
 Antonino Staiano
 Antonio Fernández
 Antonio Parziale
 Anu Singha
 Anustup Choudhury
 Anwesan Pal
 Anwesha Sengupta
 Archisman Adhikary
 Arjan Kuijper
 Arnab Kumar Das

Arnav Bhavsar
Arnav Varma
Arpita Dutta
Arshad Jamal
Artur Jordao
Arunkumar Chinnaswamy
Aryan Jadon
Aryaz Baradarani
Ashima Anand
Ashis Dhara
Ashish Phophalia
Ashok K. Bhateja
Ashutosh Vaish
Ashwani Kumar
Asifuzzaman Lasker
Atefeh Khoshkhahtinat
Athira Nambiar
Attilio Fiandrotti
Avandra S. Hemachandra
Avik Hati
Avinash Sharma
B. H. Shekar
B. Uma Shankar
Bala Krishna Thunakala
Balaji Tk
Balázs Pálffy
Banafsheh Adami
Bang-Dang Pham
Baochang Zhang
Baodi Liu
Bashirul Azam Biswas
Beiduo Chen
Benedikt Kottler
Beomseok Oh
Berkay Aydin
Berlin S. Shaheema
Bertrand Kerautret
Bettina Finzel
Bhavana Singh
Bibhas C. Dhara
Bilge Günsel
Bin Chen
Bin Li
Bin Liu
Bin Yao
Bin-Bin Jia
Binbin Yong
Bindita Chaudhuri
Bindu Madhavi Tummala
Binh M. Le
Bi-Ru Dai
Bo Huang
Bo Jiang
Bob Zhang
Bowen Liu
Bowen Zhang
Boyang Zhang
Boyu Diao
Boyun Li
Brian M. Sadler
Bruce A. Maxwell
Bryan Bo Cao
Buddhika L. Semage
Bushra Jalil
Byeong-Seok Shin
Byung-Gyu Kim
Caihua Liu
Cairong Zhao
Camille Kurtz
Carlos A. Caetano
Carlos D. Martá-Nez-Hinarejos
Ce Wang
Cevahir Cigla
Chakravarthy Bhagvati
Chandrakanth Vipparla
Changchun Zhang
Changde Du
Changkun Ye
Changxu Cheng
Chao Fan
Chao Guo
Chao Qu
Chao Wen
Chayan Halder
Che-Jui Chang
Chen Feng
Chenan Wang
Cheng Yu
Chenghao Qian
Cheng-Lin Liu

Chengxu Liu
Chenru Jiang
Chensheng Peng
Chetan Ralekar
Chih-Wei Lin
Chih-Yi Chiu
Chinmay Sahu
Chintan Patel
Chintan Shah
Chiranjoy Chattopadhyay
Chong Wang
Choudhary Shyam Prakash
Christophe Charrier
Christos Smailis
Chuanwei Zhou
Chun-Ming Tsai
Chunpeng Wang
Ciro Russo
Claudio De Stefano
Claudio F. Santos
Claudio Marrocco
Connor Levenson
Constantine Dovrolis
Constantine Kotropoulos
Dai Shi
Dakshina Ranjan Kisku
Dan Anitei
Dandan Zhu
Daniela Pamplona
Danli Wang
Danqing Huang
Daoan Zhang
Daqing Hou
David A. Clausi
David Freire Obregon
David Münch
David Pujol Perich
Davide Marelli
De Zhang
Debalina Barik
Debapriya Roy (Kundu)
Debashis Das
Debashis Das Chakladar
Debi Prosad Dogra
Debraj D. Basu
Decheng Liu
Deen Dayal Mohan
Deep A. Patel
Deepak Kumar
Dengpan Liu
Denis Coquenat
Désiré Sidibé
Devesh Walawalkar
Dewan Md. Farid
Di Ming
Di Qiu
Di Yuan
Dian Jia
Dianmo Sheng
Diego Thomas
Diganta Saha
Dimitri Bulatov
Dimpy Varshni
Dingcheng Yang
Dipanjan Das
Dipanjoyoti Paul
Divya Biligere Shivanna
Divya Saxena
Divya Sharma
Dmitrii Matveichev
Dmitry Minskiy
Dmitry V. Sorokin
Dong Zhang
Donghua Wang
Donglin Zhang
Dongming Wu
Dongqiangzi Ye
Dongqing Zou
Dongrui Liu
Dongyang Zhang
Dongzhan Zhou
Douglas Rodrigues
Duarte Folgado
Duc Minh Vo
Duoxuan Pei
Durai Arun Pannir Selvam
Durga Bhavani S.
Eckart Michaelsen
Elena Goyanes
Élodie Puybareau

Emanuele Vivoli
Emna Ghorbel
Enrique Naredo
Enyu Cai
Eric Patterson
Ernest Valveny
Eva Blanco-Mallo
Eva Breznik
Evangelos Sartinas
Fabio Solari
Fabiola De Marco
Fan Wang
Fangda Li
Fangyuan Lei
Fangzhou Lin
Fangzhou Luo
Fares Bougourzi
Farman Ali
Fatiha Mokdad
Fei Shen
Fei Teng
Fei Zhu
Feiyan Hu
Felipe Gomes Oliveira
Feng Li
Fengbei Liu
Fenghua Zhu
Fillipe D. M. De Souza
Flavio Piccoli
Flavio Prieto
Florian Kleber
Francesc Serratosa
Francesco Bianconi
Francesco Castro
Francesco Ponzio
Francisco Javier Hernández López
Frédéric Rayar
Furkan Osman Kar
Fushuo Huo
Fuxiao Liu
Fu-Zhao Ou
Gabriel Turinici
Gabrielle Flood
Gajjala Viswanatha Reddy
Gaku Nakano
Galal Binamakhshen
Ganesh Krishnasamy
Gang Pan
Gangyan Zeng
Gani Rahmon
Gaurav Harit
Gennaro Vessio
Genoveffa Tortora
George Azzopardi
Gerard Ortega
Gerardo E. Altamirano-Gomez
Gernot A. Fink
Gibran Benitez-Garcia
Gil Ben-Artzi
Gilbert Lim
Giorgia Minello
Giorgio Fumera
Giovanna Castellano
Giovanni Puglisi
Giulia Orrù
Giuliana Ramella
Gökçe Uludoğan
Gopi Ramena
Gorthi Rama Krishna Sai Subrahmanyam
Gourav Datta
Gowri Srinivasa
Gozde Sahin
Gregory Randall
Guanjie Huang
Guanjun Li
Guanwen Zhang
Guanyu Xu
Guanyu Yang
Guanzhou Ke
Guhnoo Yun
Guido Borghi
Guilherme Brandão Martins
Guillaume Caron
Guillaume Tochon
Guocai Du
Guohao Li
Guoqiang Zhong
Guorong Li
Guotao Li
Gurman Gill

Haechang Lee
Haichao Zhang
Haidong Xie
Haifeng Zhao
Haimei Zhao
Hainan Cui
Haixia Wang
Haiyan Guo
Hakime Ozturk
Hamid Kazemi
Han Gao
Hang Zou
Hanjia Lyu
Hanjoo Cho
Hanqing Zhao
Hanyuan Liu
Hanzhou Wu
Hao Li
Hao Meng
Hao Sun
Hao Wang
Hao Xing
Hao Zhao
Haoan Feng
Haodi Feng
Haofeng Li
Haoji Hu
Haojie Hao
Haojun Ai
Haopeng Zhang
Haoran Li
Haoran Wang
Haorui Ji
Haoxiang Ma
Haoyu Chen
Haoyue Shi
Harald Koestler
Harbinder Singh
Harris V. Georgiou
Hasan F. Ates
Hasan S. M. Al-Khaffaf
Hatef Otroschi Shahreza
Hebeizi Li
Heng Zhang
Hengli Wang
Hengyue Liu
Hertog Nugroho
Hieyong Jeong
Himadri Mukherjee
Hoai Ngo
Hoda Mohaghegh
Hong Liu
Hong Man
Hongcheng Wang
Hongjian Zhan
Hongxi Wei
Hongyu Hu
Hoseong Kim
Hossein Ebrahimnezhad
Hossein Malekmohamadi
Hrishav Bakul Barua
Hsueh-Yi Sean Lin
Hua Wei
Huafeng Li
Huali Xu
Huaming Chen
Huan Wang
Huang Chen
Huanran Chen
Hua-Wen Chang
Huawen Liu
Huayi Zhan
Hugo Jair Escalante
Hui Chen
Hui Li
Huichen Yang
Huiqiang Jiang
Huiyuan Yang
Huizi Yu
Hung T. Nguyen
Hyeongyu Kim
Hyeonjeong Park
Hyeonjun Lee
Hymalai Bello
Hyung-Gun Chi
Hyunsoo Kim
I-Chen Lin
Ik Hyun Lee
Ilan Shimshoni
Imad Eddine Toubal

Imran Sarker
Inderjot Singh Saggu
Indrani Mukherjee
Indranil Sur
Ines Rieger
Ioannis Pierros
Irina Rabaev
Ivan V. Medri
J. Rafid Siddiqui
Jacek Komorowski
Jacopo Bonato
Jacson Rodrigues Correia-Silva
Jaekoo Lee
Jaime Cardoso
Jakob Gawlikowski
Jakub Nalepa
James L. Wayman
Jan Čech
Jangho Lee
Jani Boutellier
Javier Gurrola-Ramos
Javier Lorenzo-Navarro
Jayasree Saha
Jean Lee
Jean Paul Barddal
Jean-Bernard Hayet
Jean-Philippe G. Tarel
Jean-Yves Ramel
Jenny Benois-Pineau
Jens Bayer
Jerin Geo James
Jesús Miguel García-Gorrostieta
Jia Qu
Jiahong Chen
Jiaji Wang
Jian Hou
Jian Liang
Jian Xu
Jian Zhu
Jianfeng Lu
Jianfeng Ren
Jiangfan Liu
Jianguo Wang
Jiangyan Yi
Jiangyong Duan
Jianhua Yang
Jianhua Zhang
Jianhui Chen
Jianjia Wang
Jianli Xiao
Jianqiang Xiao
Jianwu Wang
Jianxin Zhang
Jianxiong Gao
Jianxiong Zhou
Jianyu Wang
Jianzhong Wang
Jiaru Zhang
Jiashu Liao
Jiaxin Chen
Jiaxin Lu
Jiaxing Ye
Jiaxuan Chen
Jiaxuan Li
Jiayi He
Jiayin Lin
Jie Ou
Jiehua Zhang
Jiejie Zhao
Jignesh S. Bhatt
Jin Gao
Jin Hou
Jin Hu
Jin Shang
Jing Tian
Jing Yu Chen
Jingfeng Yao
Jinglun Feng
Jingtong Yue
Jingwei Guo
Jingwen Xu
Jingyuan Xia
Jingzhe Ma
Jinhong Wang
Jinjia Wang
Jinlai Zhang
Jinlong Fan
Jinming Su
Jinrong He
Jintao Huang

Jinwoo Ahn
Jinwoo Choi
Jinyang Liu
Jinyu Tian
Jionghao Lin
Jiuding Duan
Jiwei Shen
Jiyang Pan
Jiyoun Kim
João Papa
Johan Debayle
John Atanbori
John Wilson
John Zhang
Jónathan Heras
Joohi Chauhan
Jorge Calvo-Zaragoza
Jorge Figueroa
Jorma Laaksonen
José Joaquim De Moura Ramos
Jose Vicent
Joseph Damilola Akinyemi
Josiane Zerubia
Juan Wen
Judit Szücs
Juepeng Zheng
Juha Roning
Jumana H. Alsubhi
Jun Cheng
Jun Ni
Jun Wan
Junghyun Cho
Junjie Liang
Junjie Ye
Junlin Hu
Juntong Ni
Junxin Lu
Junxuan Li
Junyaup Kim
Junyeong Kim
Jürgen Seiler
Jushang Qiu
Juyang Weng
Jyostna Devi Bodapati
Jyoti Singh Kirar
Kai Jiang
Kaiqiang Song
Kalidas Yeturu
Kalle Åström
Kamalakar Vijay Thakare
Kang Gu
Kang Ma
Kanji Tanaka
Karthik Seemakurthy
Kaushik Roy
Kavisha Jayathunge
Kazuki Uehara
Ke Shi
Keigo Kimura
Keiji Yanai
Kelton A. P. Costa
Kenneth Camilleri
Kenny Davila
Ketan Atul Bapat
Ketan Kotwal
Kevin Desai
Keyu Long
Khadiga Mohamed Ali
Khakon Das
Khan Muhammad
Kilho Son
Kim-Ngan Nguyen
Kishan Kc
Kishor P. Upla
Klaas Dijkstra
Komal Bharti
Konstantinos Triaridis
Kostas Ioannidis
Koyel Ghosh
Kripabandhu Ghosh
Krishnendu Ghosh
Kshitij S. Jadhav
Kuan Yan
Kun Ding
Kun Xia
Kun Zeng
Kunal Banerjee
Kunal Biswas
Kunchi Li
Kurban Ubul

Lahiru N. Wijayasingha
Laines Schmalwasser
Lakshman Mahto
Lala Shakti Swarup Ray
Lale Akarun
Lan Yan
Lawrence Amadi
Lee Kang Il
Lei Fan
Lei Shi
Lei Wang
Leonardo Rossi
Lequan Lin
Levente Tamas
Li Bing
Li Li
Li Ma
Li Song
Lia Morra
Liang Xie
Liang Zhao
Lianwen Jin
Libing Zeng
Lidia Sánchez-González
Lidong Zeng
Lijun Li
Likang Wang
Lili Zhao
Lin Chen
Lin Huang
Linfei Wang
Ling Lo
Lingchen Meng
Lingheng Meng
Lingxiao Li
Lingzhong Fan
Liqi Yan
Liqiang Jing
Lisa Gutzeit
Liu Ziyi
Liushuai Shi
Liviú-Daniel Stefan
Liyuan Ma
Liyun Zhu
Lizuo Jin

Longteng Guo
Lorena Álvarez Rodríguez
Lorenzo Putzu
Lu Leng
Lu Pang
Lu Wang
Luan Pham
Luc Brun
Luca Guarnera
Luca Piano
Lucas Alexandre Ramos
Lucas Goncalves
Lucas M. Gago
Luigi Celona
Luis C. S. Afonso
Luis Gerardo De La Fraga
Luis S. Luevano
Luis Teixeira
Lunke Fei
M. Hassaballah
Maddimsetti Srinivas
Mahendran N.
Mahesh Mohan M. R.
Maiko Lie
Mainak Singha
Makoto Hirose
Malay Bhattacharyya
Mamadou Dian Bah
Man Yao
Manali J. Patel
Manav Prabhakar
Manikandan V. M.
Manish Bhatt
Manjunath Shantharamu
Manuel Curado
Manuel Günther
Manuel Marques
Marc A. Kastner
Marc Chaumont
Marc Cheong
Marc Lalonde
Marco Cotogni
Marcos C. Santana
Mario Molinara
Mariofanna Milanova

Markus Bauer
Marlon Becker
Mårten Wadenbäck
Martin G. Ljungqvist
Martin Kämpel
Martina Pastorino
Marwan Turki
Masashi Nishiyama
Masayuki Tanaka
Massimo O. Spata
Matteo Ferrara
Matthew D. Dawkins
Matthew Gadd
Matthew S. Watson
Maura Pintor
Max Ehrlich
Maxim Popov
Mayukh Das
Md Baharul Islam
Md Sajid
Meghna Kapoor
Meghna P. Ayyar
Mei Wang
Meiqi Wu
Melissa L. Tijink
Meng Li
Meng Liu
Meng-Luen Wu
Mengnan Liu
Mengxi China Guo
Mengya Han
Michaël Clément
Michal Kawulok
Mickael Coustaty
Miguel Domingo
Milind G. Padalkar
Ming Liu
Ming Ma
Mingchen Feng
Mingde Yao
Minghao Li
Mingjie Sun
Ming-Kuang Daniel Wu
Mingle Xu
Mingyong Li
Mingyuan Jiu
Minh P. Nguyen
Minh Q. Tran
Minheng Ni
Minsu Kim
Minyi Zhao
Mirko Paolo Barbato
Mo Zhou
Modesto Castrillón-Santana
Mohamed Amine Mezghich
Mohamed Dahmane
Mohamed Elsharkawy
Mohamed Yousuf
Mohammad Hashemi
Mohammad Khalooei
Mohammad Khateri
Mohammad Mahdi Dehshibi
Mohammad Sadil Khan
Mohammed Mahmoud
Moises Diaz
Monalisha Mahapatra
Monidipa Das
Mostafa Kamali Tabrizi
Mridul Ghosh
Mrinal Kanti Bhowmik
Muchao Ye
Mugalodi Ramesha Rakesh
Muhammad Rameez Ur Rahman
Muhammad Suhaib Kanroo
Muming Zhao
Munender Varshney
Munsif Ali
Na Lv
Nader Karimi
Nagabhushan Somraj
Nakkwan Choi
Nakul Agarwal
Nan Pu
Nan Zhou
Nancy Mehta
Nand Kumar Yadav
Nandakishor Nandakishor
Nandyala Hemachandra
Nanfeng Jiang
Narayan Hegde

Narayan Ji Mishra	Palash Ghosal
Narayan Vetrekar	Pallav Dutta
Narendra D. Londhe	Paolo Rota
Nathalie Girard	Paramanand Chandramouli
Nati Ofir	Paria Mehrani
Naval Kishore Mehta	Parth Agrawal
Nazmul Shahadat	Partha Basuchowdhuri
Neeti Narayan	Patrick Horain
Neha Bhargava	Pavan Kumar
Nemanja Djuric	Pavan Kumar Anasosalu Vasu
Newlin Shebiah R.	Pedro Castro
Ngo Ba Hung	Peipei Li
Nhat-Tan Bui	Peipei Yang
Niaz Ahmad	Peisong Shen
Nick Theisen	Peiyu Li
Nicolas Passat	Peng Li
Nicolas Ragot	Pengfei He
Nicolas Sidere	Pengrui Quan
Nikolaos Mitianoudis	Pengxin Zeng
Nikolas Ebert	Pengyu Yan
Nilah Ravi Nair	Peter Eisert
Nilesh A. Ahuja	Petra Gomez-Krämer
Nilkanta Sahu	Pierrick Bruneau
Nils Murrugarra-Llerena	Ping Cao
Nina S. T. Hirata	Pingping Zhang
Ninad Aithal	Pintu Kumar
Ning Xu	Pooja Kumari
Ningzhi Wang	Pooja Sahani
Niraj Kumar	Prabhu Prasad Dev
Nirmal S. Punjabi	Pradeep Kumar
Nisha Varghese	Pradeep Singh
Norio Tagawa	Pranjal Sahu
Obaidullah Md Sk	Prasun Roy
Oguzhan Ulucan	Prateek Keserwani
Olfa Mechi	Prateek Mittal
Oliver Tüselmann	Praveen Kumar Chandaliya
Orazio Pontorno	Praveen Tirupattur
Oriol Ramos Terrades	Pravin Nair
Osman Akin	Preeti Gopal
Ouadi Beya	Preety Singh
Ozge Mercanoglu Sincan	Prem Shanker Yadav
Pabitra Mitra	Prerana Mukherjee
Padmanabha Reddy Y. C. A.	Prerna A. Mishra
Palaash Agrawal	Prianka Dey
Palaiahnakote Shivakumara	Priyanka Mudgal

Qc Kha Ng
Qi Li
Qi Ming
Qi Wang
Qi Zuo
Qian Li
Qiang Gan
Qiang He
Qiang Wu
Qiangqiang Zhou
Qianli Zhao
Qiansen Hong
Qiao Wang
Qidong Huang
Qihua Dong
Qin Yuke
Qing Guo
Qingbei Guo
Qingchao Zhang
Qingjie Liu
Qinhong Yang
Qiushi Shi
Qixiang Chen
Quan Gan
Quanlong Guan
Rachit Chhaya
Radu Tudor Ionescu
Rafal Zdunek
Raghavendra Ramachandra
Rahimul I. Mazumdar
Rahul Kumar Ray
Rajib Dutta
Rajib Ghosh
Rakesh Kumar
Rakesh Paul
Rama Chellappa
Rami O. Skaik
Ramon Aranda
Ran Wei
Ranga Raju Vatsavai
Ranganath Krishnan
Rasha Friji
Rashmi S.
Razaib Tariq
Rémi Giraud
René Schuster
Renlong Hang
Renrong Shao
Renu Sharma
Reza Sadeghian
Richard Zanibbi
Rimon Elias
Rishabh Shukla
Rita Delussu
Riya Verma
Robert J. Ravier
Robert Sablatnig
Robin Strand
Rocco Pietrini
Rocio Diaz Martin
Rocio Gonzalez-Diaz
Rohit Venkata Sai Dulam
Romain Giot
Romi Banerjee
Ru Wang
Ruben Machucho
Ruddy Théodose
Ruggero Pintus
Rui Deng
Rui P. Paiva
Rui Zhao
Ruifan Li
Ruigang Fu
Ruikun Li
Ruirui Li
Ruixiang Jiang
Ruwei Jiang
Rushi Lan
Rustam Zhumagambetov
S. Amutha
S. Divakar Bhat
Sagar Goyal
Sahar Siddiqui
Sahbi Bahroun
Sai Karthikeya Vemuri
Saibal Dutta
Saihui Hou
Sajad Ahmad Rather
Saksham Aggarwal
Sakthi U.

Salimeh Sekeh
Samar Bouazizi
Samia Boukir
Samir F. Harb
Samit Biswas
Samrat Mukhopadhyay
Samriddha Sanyal
Sandika Biswas
Sandip Purnapatra
Sanghyun Jo
Sangwoo Cho
Sanjay Kumar
Sankaran Iyer
Sanket Biswas
Santanu Roy
Santosh D. Pandure
Santosh Ku Behera
Santosh Nanabhau Palaskar
Santosh Prakash Chouhan
Sarah S. Alotaibi
Sasanka Katreddi
Sathyanarayanan N. Aakur
Saurabh Yadav
Sayan Rakshit
Scott McCloskey
Sebastian Bunda
Sejuti Rahman
Selim Aksoy
Sen Wang
Seraj A. Mostafa
Shanmuganathan Raman
Shao-Yuan Lo
Shaoyuan Xu
Sharia Arfin Tanim
Shehreen Azad
Sheng Wan
Shengdong Zhang
Shengwei Qin
Shenyuan Gao
Sherry X. Chen
Shibaprasad Sen
Shigeaki Namiki
Shiguang Liu
Shijie Ma
Shikun Li
Shinichiro Omachi
Shirley David
Shishir Shah
Shiv Ram Dubey
Shiva Baghel
Shivanand S. Gornale
Shogo Sato
Shotaro Miwa
Shreya Ghosh
Shreya Goyal
Shuai Su
Shuai Wang
Shuai Zheng
Shuaifeng Zhi
Shuang Qiu
Shuhei Tarashima
Shujing Lyu
Shuliang Wang
Shun Zhang
Shunming Li
Shunxin Wang
Shuping Zhao
Shuquan Ye
Shuwei Huo
Shuyue Lan
Shyi-Chyi Cheng
Si Chen
Siddarth Ravichandran
Sihan Chen
Siladitya Manna
Silambarasan Elkana Ebinazer
Simon Benaïchouche
Simon S. Woo
Simone Caldarella
Simone Milani
Simone Zini
Sina Lotfian
Sitao Luan
Sivaselvan B.
Siwei Li
Siwei Wang
Siwen Luo
Siyu Chen
Sk Aziz Ali
Sk Md Obaidullah

Sneha Shukla
 Snehasis Banerjee
 Snehasis Mukherjee
 Snigdha Sen
 Sofia Casarin
 Soheila Farokhi
 Soma Bandyopadhyay
 Son Minh Nguyen
 Son Xuan Ha
 Sonal Kumar
 Sonam Gupta
 Sonam Nahar
 Song Ouyang
 Sotiris Kotsiantis
 Souhaila Djaffal
 Soumen Biswas
 Soumen Sinha
 Soumitri Chattopadhyay
 Souvik Sengupta
 Spiros Kostopoulos
 Sreeraj Ramachandran
 Sreya Banerjee
 Srikanta Pal
 Srinivas Arukonda
 Stephane A. Guinard
 Su O. Ruan
 Subhadip Basu
 Subhajit Paul
 Subhankar Ghosh
 Subhankar Mishra
 Subhankar Roy
 Subhash Chandra Pal
 Subhayu Ghosh
 Sudip Das
 Sudipta Banerjee
 Suhas Pillai
 Sujit Das
 Sukalpa Chanda
 Sukhendu Das
 Suklav Ghosh
 Suman K. Ghosh
 Suman Samui
 Sumit Mishra
 Sungho Suh
 Sunny Gupta

Suraj Kumar Pandey
 Surendrabikram Thapa
 Suresh Sundaram
 Sushil Bhattacharjee
 Susmita Ghosh
 Swakkhar Shatabda
 Syed Ms Islam
 Syed Tousiful Haque
 Taegyeong Lee
 Taihui Li
 Takashi Shibata
 Takeshi Oishi
 Talha Ahmad Siddiqui
 Tanguy Gernot
 Tangwen Qian
 Tanima Bhowmik
 Tanpia Tasnim
 Tao Dai
 Tao Hu
 Tao Sun
 Taoran Yi
 Tapan Shah
 Taveena Lotey
 Teng Huang
 Tengqi Ye
 Teresa Alarcon
 Tetsuji Ogawa
 Thanh Phuong Nguyen
 Thanh Tuan Nguyen
 Thattapon Surasak
 Thibault Napoléon
 Thierry Bouwmans
 Thinh Truong Huynh Nguyen
 Thomas De Min
 Thomas E. K. Zielke
 Thomas Swearingen
 Tianatahina Jimmy Francky Randrianasoa
 Tianheng Cheng
 Tianjiao He
 Tianyi Wei
 Tianyuan Zhang
 Tianyue Zheng
 Tiecheng Song
 Tilottama Goswami
 Tim Büchner

Tim H. Langer	Wataru Ohyama
Tim Raven	Wee Kheng Leow
Ting kai Liu	Wei Chen
Tingting Yao	Wei Cheng
Tobias Meisen	Wei Hua
Toby P. Breckon	Wei Lu
Tong Chen	Wei Pan
Tonghua Su	Wei Tian
Tran Tuan Anh	Wei Wang
Tri-Cong Pham	Wei Wei
Trishna Saikia	Wei Zhou
Trung Quang Truong	Weidi Liu
Tuan T. Nguyen	Weidong Yang
Tuan Vo Van	Weijun Tan
Tushar Shinde	Weimin Lyu
Ujjwal Karn	Weinan Guan
Ukrit Watchareeruetai	Weining Wang
Uma Mudenagudi	Weiqiang Wang
Umarani Jayaraman	Weiwei Guo
V. S. Malemath	Weixia Zhang
Vallidevi Krishnamurthy	Wei-Xuan Bao
Ved Prakash	Weizhong Jiang
Venkata Krishna Kishore Kolli	Wen Xie
Venkata R. Vavilthota	Wenbin Qian
Venkatesh Thirugnana Sambandham	Wenbin Tian
Verónica Maria Vasconcelos	Wenbin Wang
Véronique Ve Eglin	Wenbo Zheng
Víctor E. Alonso-Pérez	Wenhan Luo
Vinay Palakkode	Wenhao Wang
Vinayak S. Nageli	Wen-Hung Liao
Vincent J. Whannou De Dravo	Wenjie Li
Vincenzo Conti	Wenkui Yang
Vincenzo Gattulli	Wenwen Si
Vineet Padmanabhan	Wenwen Yu
Vishakha Pareek	Wenwen Zhang
Viswanath Gopalakrishnan	Wenwu Yang
Vivek Singh Baghel	Wenxi Li
Vivekraj K.	Wenxi Yue
Vladimir V. Arlazarov	Wenxue Cui
Vu-Hoang Tran	Wenzhuo Liu
W. Sylvia Lilly Jebarani	Widhiyo Sudiyono
Wachirawit Ponghiran	Willem Dijkstra
Wafa Khlif	Wolfgang Fuhl
Wang An-Zhi	Xi Zhang
Wanli Xue	Xia Yuan

Xianda Zhang
Xiang Zhang
Xiangdong Su
Xiang-Ru Yu
Xiangtai Li
Xiangyu Xu
Xiao Guo
Xiao Hu
Xiao Wu
Xiao Yang
Xiaofeng Zhang
Xiaogang Du
Xiaoguang Zhao
Xiaoheng Jiang
Xiaohong Zhang
Xiaohua Huang
Xiaohua Li
Xiao-Hui Li
Xiaolong Sun
Xiaosong Li
Xiaotian Li
Xiaoting Wu
Xiaotong Luo
Xiaoyan Li
Xiaoyang Kang
Xiaoyi Dong
Xin Guo
Xin Lin
Xin Ma
Xinchi Zhou
Xingguang Zhang
Xingjian Leng
Xingpeng Zhang
Xingzheng Lyu
Xinjian Huang
Xinqi Fan
Xinqi Liu
Xinqiao Zhang
Xinrui Cui
Xizhan Gao
Xu Cao
Xu Ouyang
Xu Zhao
Xuan Shen
Xuan Zhou

Xuchen Li
Xuejing Lei
Xuelu Feng
Xueting Liu
Xuewei Li
Xueyi X. Wang
Xugong Qin
Xu-Qian Fan
Xuxu Liu
Xu-Yao Zhang
Yan Huang
Yan Li
Yan Wang
Yan Xia
Yan Zhuang
Yanan Li
Yanan Zhang
Yang Hou
Yang Jiao
Yang Liping
Yang Liu
Yang Qian
Yang Yang
Yang Zhao
Yangbin Chen
Yangfan Zhou
Yanhui Guo
Yanjia Huang
Yanjun Zhu
Yanming Zhang
Yanqing Shen
Yaoming Cai
Yaoxin Zhuo
Yaoyan Zheng
Yaping Zhang
Yaqian Liang
Yarong Feng
Yasmina Benmabrouk
Yasufumi Sakai
Yasutomo Kawanishi
Yazeed Alzahrani
Ye Du
Ye Duan
Yechao Zhang
Yeong-Jun Cho

Yi Huo
Yi Shi
Yi Yu
Yi Zhang
Yibo Liu
Yibo Wang
Yi-Chieh Wu
Yifan Chen
Yifei Huang
Yihao Ding
Yijie Tang
Yikun Bai
Yimin Wen
Yinan Yang
Yin-Dong Zheng
Yinfeng Yu
Ying Dai
Yingbo Li
Yiqiao Li
Yiqing Huang
Yisheng Lv
Yisong Xiao
Yite Wang
Yizhe Li
Yong Wang
Yonghao Dong
Yong-Hyuk Moon
Yongjie Li
Yongqian Li
Yongqiang Mao
Yongxu Liu
Yongyu Wang
Yongzhi Li
Youngha Hwang
Yousri Kessentini
Yu Wang
Yu Zhou
Yuan Tian
Yuan Zhang
Yuanbo Wen
Yuanxin Wang
Yubin Hu
Yubo Huang
Yuchen Ren
Yucheng Xing
Yuchong Yao
Yuecong Min
Yuewei Yang
Yufei Zhang
Yufeng Yin
Yugen Yi
Yuhang Ming
Yujia Zhang
Yujun Ma
Yukiko Kenmochi
Yun Hoyeoung
Yun Liu
Yunhe Feng
Yunxiao Shi
Yuru Wang
Yushun Tang
Yusuf Osmanlioglu
Yusuke Fujita
Yuta Nakashima
Yuwei Yang
Yuwu Lu
Yuxi Liu
Yuya Obinata
Yuyao Yan
Yuzhi Guo
Zaipeng Xie
Zander W. Blasingame
Zedong Wang
Zeliang Zhang
Zexin Ji
Zhanxiang Feng
Zhaofei Yu
Zhe Chen
Zhe Cui
Zhe Liu
Zhe Wang
Zhekun Luo
Zhen Yang
Zhenbo Li
Zhenchun Lei
Zhenfei Zhang
Zheng Liu
Zheng Wang
Zhengming Yu
Zhengyin Du

Zhengyun Cheng
Zhenshen Qu
Zhenwei Shi
Zhenzhong Kuang
Zhi Cai
Zhi Chen
Zhibo Chu
Zhicun Yin
Zhida Huang
Zhida Zhang
Zhifan Gao
Zhihang Ren
Zhihang Yuan
Zhihao Wang
Zhihua Xie
Zhihui Wang
Zhikang Zhang
Zhiming Zou
Zhiqi Shao
Zhiwei Dong
Zhiwei Qi
Zhixiang Wang
Zhixuan Li
Zhiyu Jiang
Zhiyuan Yan
Zhiyuan Yu
Zhiyuan Zhang
Zhong Chen
Zhongwei Teng
Zhongzhan Huang
Zhongzhi Yu
Zhuan Han
Zhuangzhuang Chen
Zhuo Liu
Zhuo Su
Zhuojun Zou
Zhuoyue Wang
Ziang Song
Zicheng Zhang
Zied Mnasri
Zifan Chen
Žiga Babnik
Zijing Chen
Zikai Zhang
Ziling Huang
Zilong Du
Ziqi Cai
Ziqi Zhou
Zi-Rui Wang
Zirui Zhou
Ziwen He
Ziyao Zeng
Ziyi Zhang
Ziyue Xiang
Zonglei Jing
Zongyi Xu

Contents – Part XXX

YOLO-RSOD: Improved YOLO Remote Sensing Object Detection	1
<i>Yang Xu and Jun Lu</i>	
Cross-Modal Ship Grounding: Towards Large Model for Enhanced Few-Shot Learning	16
<i>Quan Hu, Li Chen, Zhida Feng, and Yaojie Chen</i>	
STNet: Small Target Detection Network for IR Imagery	29
<i>Nikhil Kumar, Pranav Singh Chib, and Pravendra Singh</i>	
FF-Yolo: A Feature-Fusion Yolo Model for Small Scale FODs Detection in Airport Runways	45
<i>Soumen Biswas and Ananth Ganesh</i>	
Weakly Aligned Multi-spectral Pedestrian Detection via Cross-Modality Differential Enhancement and Multi-scale Spatial Alignment	61
<i>Zhenzhou Shao, Yongxin Chen, Yibo Zou, Jie Zhang, and Yong Guan</i>	
CrackUDA: Incremental Unsupervised Domain Adaptation for Improved Crack Segmentation in Civil Structures	74
<i>Kushagra Srivastava, Damodar Datta Kancharla, Rizvi Tahereen, Pradeep Kumar Ramancharla, Ravi Kiran Sarvadevabhatla, and Harikumar Kandath</i>	
DS MYOLO: A Reliable Object Detector Based on SSMs for Driving Scenarios	90
<i>Yang Li and Jianli Xiao</i>	
Robust Single-Cam Surround View Object Detection and Localization Using Memory Maps	105
<i>Yitong Quan, Benjamin Kiefer, Martin Messmer, Charan Ram Akupati, Rainer Graser, and Andreas Zell</i>	
Exploring the Reliability of Foundation Model-Based Frontier Selection in Zero-Shot Object Goal Navigation	119
<i>Shuaihang Yuan, Halil Utku Unlu, Hao Huang, Congcong Wen, Anthony Tzes, and Yi Fang</i>	

Reliable Semantic Understanding for Real World Zero-Shot Object Goal Navigation	135
<i>Halil Utku Unlu, Shuaihang Yuan, Congcong Wen, Hao Huang, Anthony Tzes, and Yi Fang</i>	
AllWeather-Net: Unified Image Enhancement for Autonomous Driving Under Adverse Weather and Low-Light Conditions	151
<i>Chenghao Qian, Mahdi Rezaei, Saeed Anwar, Wenjing Li, Tanveer Hussain, Mohsen Azarmi, and Wei Wang</i>	
Uni4DAL: A Unified Baseline for Multi-dataset 4D Auto-Labeling	167
<i>Zhiyuan Yang, Xuekuan Wang, Wei Zhang, Xiao Tan, Jinchen Lu, Jingdong Wang, Errui Ding, Zhihui Lai, and Cairong Zhao</i>	
Dual-Attention Fusion Network with Edge and Content Guidance for Remote Sensing Images Segmentation	183
<i>Shuaipeg Ding, Jianan Shui, Xin Li, and Mingyong Li</i>	
Distortion Correction Sub-network for Semantic Segmentation Based on Deep Hough Transform	198
<i>Wanpeng Geng, Jing Liu, Dexin Zhang, and Hui Zhang</i>	
MemoFlow: Modifying Explicit Motion of Inconsistency in Optical Flow	219
<i>Mengfei Wang, Wenjun Shi, Dongchen Zhu, Lei Wang, and Jiamao Li</i>	
Enhanced Brain Tumor Segmentation Using Preprocessing Techniques and 3D U-Net	235
<i>Abdelrahman Telib and Mohamed Gabr</i>	
Joint Top-Down and Bottom-Up Frameworks for 3D Visual Grounding	249
<i>Yang Liu, Daizong Liu, and Wei Hu</i>	
Anticipating Future Object Compositions Without Forgetting	265
<i>Youssef Zahran, Gertjan Burghouts, and Yke B. Eisma</i>	
SPK: Semantic and Positional Knowledge for Zero-Shot Referring Expression Comprehension	280
<i>Zetao Du, Jianhua Yang, Junbo Wang, Yan Huang, and Liang Wang</i>	
Can Language Improve Visual Features For Distinguishing Unseen Plant Diseases?	296
<i>Jerad Zherui Liaw, Abel Yu Hao Chai, Sue Han Lee, Pierre Bonnet, and Alexis Joly</i>	

Show Me the World in My Language: Establishing the First Baseline
for Scene-Text to Scene-Text Translation 312
Shreyas Vaidya, Arvind Kumar Sharma, Prajwal Gatti, and Anand Mishra

iGrasp: An Interactive 2D-3D Framework for 6-DoF Grasp Detection 329
*Jian-Jian Jiang, Xiao-Ming Wu, Zibo Chen, Yi-Lin Wei,
and Wei-Shi Zheng*

Goal-Driven Transformer for Robot Behavior Learning from Play Data 346
*Congcong Wen, Jiazhao Liang, Shuaihang Yuan, Hao Huang, Yu Hao,
Hui Lin, Yu-Shen Liu, and Yi Fang*

Adaptive Dynamic VSLAM: Refining Semantic-Geometric Fusion
and Static Background inpainting 360
Qi Mu, Baizhang Guo, Shuai Guo, and Zhanli Li

Hierarchical Visual Place Recognition with Semantic-Guided Attention 377
Wenwen Ming, Xucan Chen, Zhe Liu, Ruihao Li, and Wei Yi

Dense Reconstruction and Localization in Scenes with Glass Surfaces
Based on ORB-SLAM2 393
*Zeyuan Chen, Ziquan Wang, Qiang Gao, Masahiko Mikawa,
and Makoto Fujisawa*

Content-Aware Feature Upsampling for Voxel-Based 3D Semantic
Segmentation 411
Yu Song, Ruigang Fu, Qingyong Hu, Biao Li, and Ping Zhong

Enhancing 3D Referential Grounding by Learning Coarse Spatial
Relationships 427
*Soham Joshi, Aditay Tripathi, Viswanath Gopalakrishnan,
and Anirban Chakraborty*

PointGADM: Geometry Acquainted Deep Model for 3D Point Cloud
Analysis 443
*Seema Kumari, Samay Kalpesh Patel, Raja Muthalagu,
and Shanmuganathan Raman*

CroMA: Cross-Modal Attention for Visual Question Answering in Robotic
Surgery 459
Gretta Antonio, Jobin Jose, Sudhish N George, and Kiran Raja

Author Index 473



DCI-Net: Remote Sensing Image-Based Object Detector

Quanyue Cui^{1,2,3} and Jun Lu^{1,2,3(✉)}

¹ College of Computer Science and Technology, Heilongjiang University,
Harbin 150080, China

22319810s.hlju.edu.cn, lujun111_lily@sina.com

² Jiaxiang Industrial Technology Research Institute of Heilongjiang University,
Jining 272400, Shandong, China

³ Key Laboratory of Database and Parallel Computing of Heilongjiang Province,
Heilongjiang University, Harbin 150080, China

Abstract. Recently the analysis of remotely sensed images has played a vital role in various aspects of research. The current researches ignore the unique prior knowledge in remote sensing images and do not consider exploring the contextual information of the object, while the existence of multi-scale and high image resolution of objects in remote sensing images also affects the accuracy of the object detection task. Based on the above problems, this paper proposes a object detector DCI-Net (Dynamic Context-Aware IoU Network) based on remote sensing images, in which the proposed CASK (Context-Aware Selective Kernel) module can explicitly model the interdependence between the convolutional feature channels. A loss function P_i IoU is proposed, which adaptively adjusts the penalty factor in combination with the size of the detected object. A DySample module is introduced, which is able to effectively extract and utilize the spatial structure features. The model in this paper improves the detection accuracy of complex objects in remote sensing images. On the DIOR dataset, compared with the baseline model YOLOV9, the accuracy is improved by 0.6%, the number of parameters is decreased by 4%, and the floating point operation speed is improved by 27.8%.

Keywords: Remote sensing images · Object detection · Sample less learning · Context-aware

1 Introduction

Remote sensing images analysis is pivotal in current research, providing key information support for environmental monitoring, resource management, disaster warning and other fields. Among them, object detection is a crucial task in the analysis of remote sensing images [12]. In recent years, Convolutional Neural Networks (CNNs) have driven significant advances in remote sensing images (RSI) object detection tasks [2, 15, 21, 22]. However, due to the complexity of

remote sensing data acquisition, RSI often shows the problem of sparse category labeling. Traditional CNN-based object detectors are prone to severe overfitting with limited training data [6]. Therefore, enabling the model to achieve accurate detection and recognition of objects from limited labeled data with efficient algorithms is known as Few Sample Object Detection (FSOD).

Currently, the mainstream methods used to solve FSOD are categorized into two main groups: based on Few-Shot Learning and based on Migration Learning. Among them, Few-Shot Learning enables the model to learn effectively with a small amount of labeled data. This type of learning is especially suitable for real-world scenarios where data is scarce, and is therefore very applicable to the task of object detection in remote sensing images [5]. For example, MetaYOLO [9] and FRW [13] introduce reweighting vectors to recalibrate query features at various scales. Building on the two-stage object detector Meta R-CNN [34], Zhang et al. [36] extended the method to train data to handle objects in arbitrary directions in RSI. However, there is a significant limitation of few-sample learning in that it may be difficult to generalize effectively to unseen classes when faced with diverse or highly complex object domains.

The multi-scale presence of objects in remote sensing images and the large size of the images data are also two great challenges [25]. First of all, the multi-scale problem that exists in objects in images is mainly caused by the large difference in the scale size of the objects. Small-scale objects are easily ignored, and large-scale objects may appear partially obscured because of the boundary [6], which is not well solved by many current two-stage object detectors. Secondly, remote sensing images have the problem of too large data size. This type of images is generally taken by taking a bird's eye view from the air downward, and the images contain a relatively large range and has a high resolution. Therefore, the characteristics of this type of images not only require a model that can solve the problem of small sample size and multi-scale objects, but also have the ability to handle large-scale images.

Based on the above analysis, this paper proposes a new object detection model DCI-Net, which is mainly used to realize the object detection task in remote sensing images. In order to better cope with the multi-scale and high-resolution problems existing in remote sensing images, this paper adopts YOLOV9 [32] as a benchmark to construct the feature extraction module and the detection module in the DCI-Net model, respectively, to make full use of its lightweight features as well as its extensive global receptive field. The contributions of this paper are as follows:

- 1) Propose the DCI-Net model, which outperforms the baseline model in all aspects of object detection performance on the DIOR dataset.
- 2) Propose the CASK module, which detects more contextual information of the object while reducing the number of parameters, thus improving the accuracy of detection.
- 3) Propose the loss function Pi_IoU to adaptively adjust the penalty factor according to the size of the object and combine with the loss function of

the anchor frame quality for gradient adjustment, so that the object can be detected more accurately.

- 4) Introducing the DySample module, which not only reduces the GPU memory, but also has a great advantage over other up-sampling modules in the detection task.

2 Related Work

2.1 Object Detection

Currently, popular object detectors are trained on datasets of predefined categories (e.g., the COCO dataset [18] and the Objects365 dataset [30]), and the rapid development of object detectors has benefited from these rich datasets. Modern object detection models are usually categorized into two types: two-stage object detectors [4,16], such as Faster R-CNN [29] and R-FCN [4], and single-stage object detectors [10,17,26], such as YOLO [28] and FCOS [31]. Two-stage detectors first generate candidate frames through a candidate frame generation network, and then extract features and feed these features into a object classifier and bounding box regressor. The single-stage object detectors, on the other hand, accomplishes object detection by generating and classifying candidate frames directly on the input images, omitting the step of explicitly generated candidate frames. This paper is mainly based on the more popular single-stage YOLOV9 to improve. Compared with the baseline model, the DCI-Net in this paper has more powerful object detection ability, and achieves better detection effect in the detection of remote sensing images.

2.2 Object Detection in Remote Sensing Images

Object detection in remotely sensed images is an important problem in the field of aerial images analysis and plays an important role in many applications with the wide availability of satellite images [1]. However, remote sensing images usually exhibit unique characteristics in terms of multi-scale objects. The complexity and diversity of the surface landscapes they capture usually require in-depth exploration of objects at different scales for a comprehensive analysis [6]. The dense distribution of detection objects in remote sensing images and the complexity of the background, as well as the relatively large number of objects in the images, generally make it difficult to achieve good detection results using current object detectors. These limitations have hindered the progress of traditional deep learning object detectors because they require large amounts of well-labeled, carefully curated data. The inherent properties of remote sensing images themselves present challenges and opportunities for the task of object detection and analysis, and existing research has explored this area. Deng et al. [13] introduced a reweighting module on the YOLO architecture for recalibrating the feature maps from a set of annotated support images. Wolf et al. [33] designed a two-header architecture to prevent the loss of base class knowledge and to work with sampling and preprocessing strategies to better utilize base class annotations.

2.3 Few Sample Object Detection in Remote Sensing Images

Few sample object detection aims to cope with the situation of insufficient samples of object objects in large datasets [37,38]. As an emerging field in the field of remote sensing images detection, the core concept of FSOD is to accurately detect objects in images by training the model using a small number of labeled samples. Compared to natural images, remote sensing images show greater diversity in terms of object size and orientation [22]. To cope with these challenges, some studies have introduced more advanced feature extraction modules for few sample object detection in remote sensing images [1,7,13,33]. For example, Cheng et al. [3] proposed a prototype-guided region proposal network (RPN) that integrates support feature information into candidate box scoring for better region proposal generation. And Zhang et al. [8] used directional enhancement of support features to mitigate diversity in object orientation. Compared to these existing methods, the method in this paper aims to achieve sample balance and to make the model more focused on the features in the images to better improve the object detection accuracy of the network.

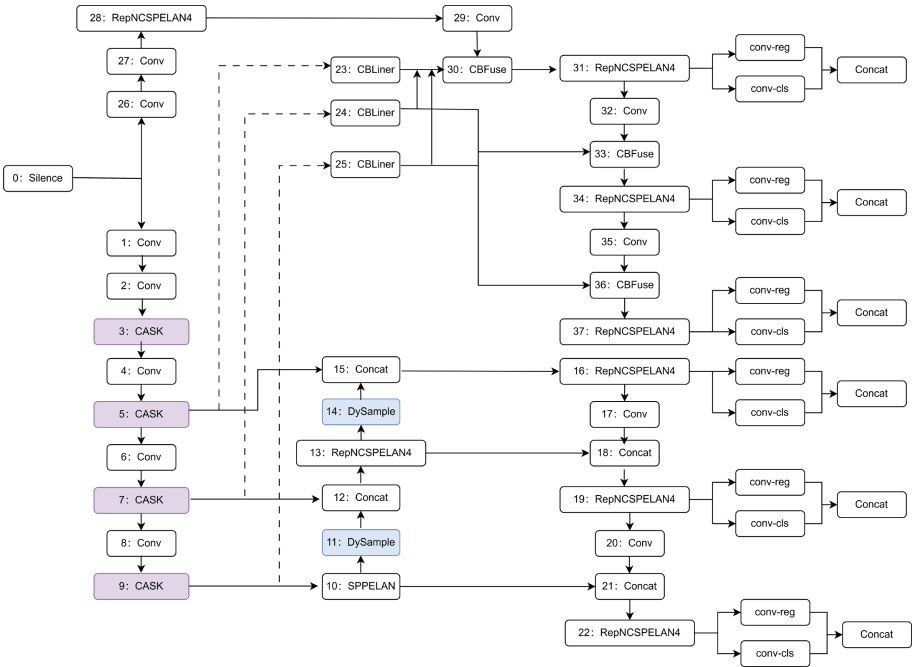


Fig. 1. DCI-Net model.

3 Methodology

In this section, the DCI-Net model is designed and implemented, which includes CASK module, DySample module, and Pi_IoU loss.

3.1 DCI-Net Model

Compared to natural images, remote sensing images usually exhibit unique characteristics in terms of multi-scale objects. The complexity and diversity of the surface landscapes they capture usually require in-depth exploration of objects at different scales for a comprehensive analysis [6]. At the same time, object detection tasks in remote sensing images face dense object distribution, complex backgrounds, and a large number of objects, which are difficult to be recognized by only one appearance factor. In order to solve the above problems, this paper proposes the DCI-Net model, as shown in Fig. 1.

As can be seen in Fig. 1, this object detection model modifies the Upsample layer in YOLOV9 to DySample, and at the same time modifies the loss function to Pi_IoU. Finally, DCI-Net replaces the RepNCSPeLan4 module inside Backbone with the CASK module proposed in this paper, so as to realize the sample balancing and make the model pay more attention to the features in the images, which in turn better improves object detection capability.

3.2 CASK Module

Recently, the improvement of the directed bounding box is more popular in the research of object detection tasks in remote sensing images, but it ignores the unique prior knowledge in remote sensing images. Because aerial images are mainly captured from high altitude at a high resolution [14], in order to successfully and correctly recognize a object in an image, it is often necessary to rely on the content of its broad context. The CASK module proposed in this paper is a good solution to the above problem. The module uses an innovative hybrid convolutional kernels strategy to capture richer contextual information and extract features with more details and different levels, which can improve the accuracy especially for the detection of complex objects in remote sensing images, as shown in Fig. 2.

This module divides the data after convolutional processing according to channels and processes only part of it at a time, which reduces the amount of computation. The CASK module consists of two main branches: one branch goes into the Bottleneck layer, which extracts and fuses more useful features while reducing the amount of computation by using a low-latitude feature space, since this module retains the results of each process through the Bottleneck module. Another branch carries out the attention residual join operation, in order to pay more attention to the object information, this branch adopts the strategy of using hybrid convolutional kernels, which greatly reduces the number of parameters under the premise of guaranteeing the sensing field. At the same time extracts richer features and improves the module's ability to detect the

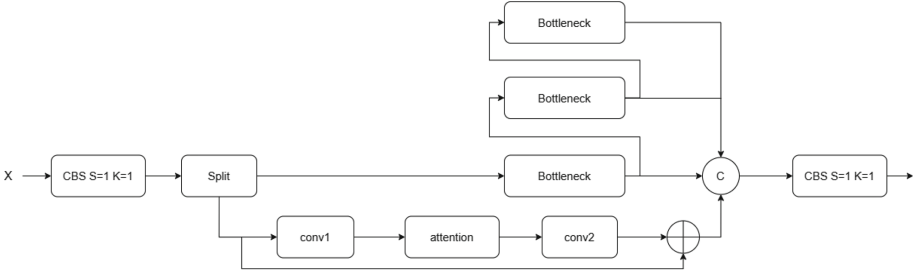


Fig. 2. CASK module.

objects of different scales. Combined with the spatial selection mechanism, the sensing field is dynamically adapted to capture the multiscale features. Thus, flexible adaptation and accurate recognition of different object contexts can be realized in remote sensing images object detection.

3.3 Pi_IoU Loss Function

The computation of existing bounding box regression loss functions is constantly updated and optimized, which has a significant impact on the performance of the object detection task. However, the existing IoU loss function generally improves the speed of convergence by adding a loss function, ignoring the limitations of the IoU loss value itself. Although the IoU loss can describe the regression state of the bounding box, it can be seen after a large number of experiments that it is unable to realize its own adjustment according to different detectors and detection tasks [35]. Meanwhile the existing IoU loss function has the problem of unreasonable penalty factor [20]. In order to solve the above problems, a new loss function Pi_IoU is proposed in this paper. This function combines the size of detection object with self-adaptive penalty factor and gradient adjustment is done based on the loss function of anchor frame quality. Meanwhile, in order to make up for the shortcomings of the existing IoU loss function, which has poor generalization and slow convergence speed in different detection tasks, this paper introduces a scaling factor for different datasets and detectors, so as to control the scale size of the auxiliary bounding box when calculating the loss.

In Eq. 1, this paper introduces the scaling factor ratio to control the size of the auxiliary bounding box and perform coordinate transformation. b and b^{gt} are denoted as the predicted bounding box and the real bounding box, respectively. x_i and y_i are the coordinates of the upper-left and lower-right corners, and are the width and height of the bounding box.

$$\begin{aligned}
 b_{x_1}^{gt} &= x_c^{gt} - w^{gt} * \text{ratio}, & b_{x_2}^{gt} &= x_c^{gt} + w^{gt} * \text{ratio} \\
 b_{y_1}^{gt} &= y_c^{gt} - h^{gt} * \text{ratio}, & b_{y_2}^{gt} &= y_c^{gt} + h^{gt} * \text{ratio} \\
 b_{x_1} &= x_c - w * \text{ratio}, & b_{x_2} &= x_c^{gt} + w * \text{ratio} \\
 b_{y_1} &= y_c - h * \text{ratio}, & b_{y_2} &= y_c^{gt} + h * \text{ratio}
 \end{aligned} \tag{1}$$

The coordinates obtained from Eq. 1 are then used to calculate the intersection and concurrency ratio of the bounding box, as shown in Eq. 2.

$$\begin{aligned} \text{inter} &= (\min(b_{x_2}^{gt}, b_{x_2}) - \max(b_{x_1}^{gt}, b_{x_1})) * \\ &\quad (\min(b_{y_2}^{gt}, b_{y_2}) - \max(b_{y_1}^{gt}, b_{y_1})) \\ \text{union} &= (w^{gt} * h^{gt}) * (\text{ratio})^2 + (w * h) * (\text{ratio})^2 - \text{inter} + \text{eps} \\ \text{iou}_1 &= \frac{\text{inter}}{\text{union}} \end{aligned} \quad (2)$$

In order to evaluate and optimize the positioning accuracy and spatial coherence of the bounding box using a more accurate method, thus enhancing the performance and robustness of the object detection and tracking algorithm. In Eq. 3, this paper calculates the bounding differences in the directions of x and y respectively. dw_i and dh_i calculate the minimum and maximum boundary differences between the two bounding boxes in the direction of the coordinate axes, respectively.

$$\begin{aligned} dw_1 &= |\min(b_{x_1}, b_{x_2}) - \min(b_{x_1}^{gt}, b_{x_2}^{gt})| \\ dw_2 &= |\max(b_{x_1}, b_{x_2}) - \max(b_{x_1}^{gt}, b_{x_2}^{gt})| \\ dh_1 &= |\min(b_{y_1}, b_{y_2}) - \min(b_{y_1}^{gt}, b_{y_2}^{gt})| \\ dh_2 &= |\max(b_{y_1}, b_{y_2}) - \max(b_{y_1}^{gt}, b_{y_2}^{gt})| \end{aligned} \quad (3)$$

In this paper, the position and size differences of the bounding boxes are considered together to provide a more comprehensive and fine-grained similarity evaluation, using the boundary difference values calculated in Eq. 3 for the boundary difference metric operation, as shown in Eq. 4.

$$P = \left(\frac{(dw_1 + dw_2)}{w^{gt}} + \frac{(dh_1 + dh_2)}{h^{gt}} \right) / 4 \quad (4)$$

Combining the overlap metric and the positional difference metric, this in turn provides a more fine-grained similarity evaluation of the bounding boxes, as shown in Eq. 5. This evaluation takes into account not only the overlap of the bounding boxes, but also the exact location of the bounding boxes, enabling the model to focus more on accurate bounding box prediction, thus improving the accuracy of object detection.

$$\text{iou} = 1 - \text{iou}_1 - e^{-P^2} + 1 \quad (5)$$

3.4 DySample Module

Feature up-sampling is a key component of dense predictive modeling to progressively restore feature resolution. Since backbone networks typically output multi-scale features and low-resolution features need to be upsampled to high resolution, a lightweight and efficient upsampler is beneficial for dense prediction models. While the performance gains of the more recent popular kernel-based dynamic upsamplers such as FADE [27] and SAPA [26] are impressive, they impose a significant workload on the detectors because of the time-consuming dynamic convolution and the additional sub-networks used to generate the dynamic kernel. Thus, the DySample module was proposed in the paper [24],

which no longer required a customized CUDA package compared to previous dynamic upsamplers. At the same time, there are extensive tuning of parameters such as GFLOPs, GPU memory and latency.

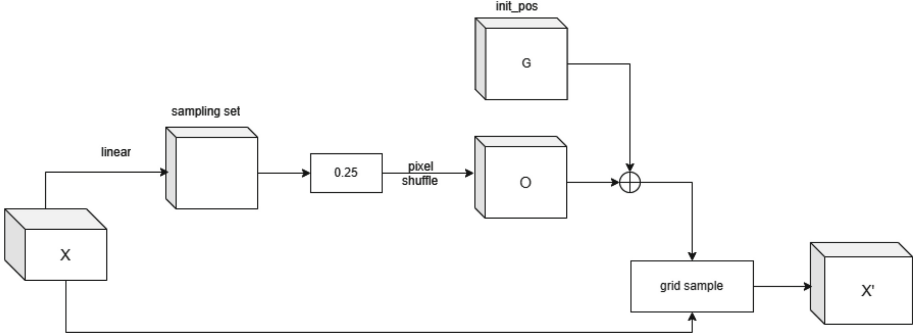


Fig. 3. DySample module. X , X' , O and G denote the input features, the up-sampled features, the generated offsets and the original mesh, respectively. The sample set is generated by the sample point generator and the input features are resampled using the mesh sampling function. In the generator, the sampling set is the sum of the generated offsets and the original mesh positions, where the offsets are generated through a linear layer.

The DySample module is shown in Fig. 3. In this paper, the static range factor in the DySample module is used, and the offsets are generated directly from the input data through a linear layer without further dynamic adjustments. This version can be used as the base implementation of DySample, which is simpler in implementation and has fewer parameters. And the choice is a Linear+Pixel Shuffle style of processing, i.e., a linear layer is first used to generate the offsets, and then these offsets are rearranged to fit the spatial dimensions, which is done by a Pixel Shuffle operation. This approach is more advantageous in terms of memory footprint, inference speed, etc.

4 Experiments

This section begins with a description of the dataset DIOR to be used, followed by a comparison of the performance of the model investigated in this paper with that of current state-of-the-art models, and finally ablation experiments are also performed on the proposed method.

4.1 Dataset

The DIOR (Dense Image Overlapping Regions) dataset is a large and comprehensive object detection dataset for remote sensing images from Google Earth

released by Northwestern Polytechnical University (NWPU) in 2018, aiming to promote object detection research in the field of remote sensing images [11]. The DIOR dataset contains 20 different classes of objects from a wide range of sensors, covering multiple geographic regions and time periods, providing a rich diversity of scenarios and diversity of object classes. The dataset consists of a total of 23,463 high-resolution images of 800*800 pixels labeled with 192,472 instances, which are mainly used for the task of object detection in remote sensing images.

4.2 Implementation Details

For the DIOR dataset, the model is trained and validated in this paper using an NVIDIA GeForce RTX 3090Ti GPU with 24 GB of graphics memory. In addition, the model uses a Stochastic Gradient Descent (SGD) optimizer with a batch size of 8, a learning rate of 0.01, a weight decay coefficient of 0.0005, and an epoch of 500.

4.3 Comparison with State-of-the-Art Models

In order to demonstrate the effectiveness of the DCI-Net model proposed in this paper for object detection in remote sensing images, its performance is analyzed in this paper in comparison with the performance of different existing models on the DIOR dataset. This work uses several metrics to evaluate the performance and effectiveness of DCI-Net. These metrics include precision (P) and mean average precision (mAP). Table 1 shows the accuracy of detecting each class compared to existing models.

As can be seen from Table 1, the proposed model DCI-Net achieves the best accuracy in most of the categories, especially in categories C9 and C20, where DCI-Net outperforms the baseline model by nearly 2.1% and 0.6%.

Table 2 compares the detection results of other models as well as the DCI-Net model in this paper on the DIOR dataset. From the table, the model proposed in this paper improves 0.6% in accuracy and 0.1% in mAP@50:95 relative to the baseline model. Although mAP@50 is slightly inferior to the model YOLOV9-C, which exhibits that the performance may not be optimal under certain categories or thresholds, the subsequent ablation experiments show that the number of parameters of the DCI-Net model is relatively lower and the computing speed is faster as well. The substantial improvement in accuracy indicates that the model proposed in this paper performs excellently in reducing false positives, which is crucial for remote sensing images analysis tasks that require high confidence. The model in this paper is able to identify and localize objects more accurately, which is particularly suitable for application scenarios that require high accuracy and allow for some inference delay, and can provide better performance than the lightweight model when the computational resources are sufficient.

Table 1. Mean accuracy values of detection in 20 classes. The 20 classes are aircraft (C1), airports (C2), baseball stadiums (C3), basketball courts (C4), bridges (C5), chimneys (C6), dams (C7), highway service areas (C8), highway toll booths (C9), golf courses (C10), surface runways (C11), harbors (C12), overpasses (C13), ships (C14), stadiums (C15), storage tanks (C16), tennis courts (C17), train stations (C18), vehicles (C19) and windmills (C20).

Model	PANet [23]	MRCNN [7]	RetinaNet [29]	FPN [16]	Carafe	CSFF [4]	GCF [2]	FFPF [19]	YOLOV8	YOLOV9	OURS
C1	61.9	53.8	53.3	60.2	58.9	57.2	62.8	65.5	94.7	<u>97.6</u>	97.7
C2	70.4	72.3	77.0	83.4	83.7	79.6	86.5	86.7	95.0	98.3	<u>97.2</u>
C3	71.0	63.2	69.3	73.8	77.8	70.1	74.8	79.4	95.8	<u>98.0</u>	98.1
C4	80.4	81.0	85.0	88.7	88.9	87.4	89.2	89.0	93.9	<u>97.1</u>	97.2
C5	38.9	38.7	44.1	49.0	50.6	46.1	49.2	50.3	57.1	<u>67.7</u>	68.1
C6	72.5	72.6	73.2	78.9	79.1	76.6	76.6	79.2	86.9	94.1	<u>93.2</u>
C7	56.6	55.9	62.4	66.7	72.7	62.7	72.5	73.3	79.3	<u>89.0</u>	89.3
C8	68.4	71.6	78.6	85.4	82.8	82.6	85.7	87.6	96.6	<u>98.3</u>	98.4
C9	60.0	67.0	62.8	71.3	72.7	73.2	75.1	73.6	81.3	<u>90.9</u>	93.0
C10	69.0	73.0	78.6	81.5	82.8	78.2	81.3	83.5	86.6	93.8	<u>92.0</u>
C11	74.6	75.8	76.6	82.8	84.3	81.6	83.3	85.1	90.4	<u>93.6</u>	93.8
C12	41.6	44.2	49.9	54.7	55.8	50.7	60.2	57.3	74.7	77.8	<u>76.3</u>
C13	55.8	56.5	59.6	62.4	62.4	59.5	62.7	63.5	71.0	<u>77.1</u>	77.8
C14	71.7	71.9	71.1	73.3	74.3	73.3	72.7	74.1	94.3	96.1	<u>95.7</u>
C15	72.9	58.6	68.4	77.3	75.2	63.4	77.3	78.4	96.8	<u>97.7</u>	98.3
C16	62.3	53.6	45.8	59.4	59.0	58.5	61.9	59.3	88.1	91.7	<u>89.0</u>
C17	81.2	81.1	81.3	87.5	88.7	85.9	88.0	88.6	95.5	97.2	<u>96.8</u>
C18	54.6	54.0	55.2	65.0	70.4	61.9	69.9	71.0	71.6	<u>82.0</u>	82.3
C19	48.2	43.1	44.4	42.2	43.6	42.9	47.0	43.3	64.3	75.0	<u>72.6</u>
C20	86.7	81.1	85.5	85.1	86.8	86.9	89.7	87.4	91.4	<u>95.6</u>	96.2

Table 2. Comparison results between DCI-Net and other models.

Model	P	mAP@50	mAP@50:95
CSFF [4]	—	68.0	—
FPN [16]	—	71.4	—
Carafe	—	72.8	—
GCF [2]	—	73.3	—
FFPF [19]	—	73.8	—
YOLOV8	88.9	85.3	62.7
YOLOV9-S	<u>90.3</u>	89.2	68.6
YOLOV9-M	90.6	89.9	69.5
YOLOV9-C	90.0	90.5	<u>70.5</u>
OURS	90.6	<u>90.3</u>	70.6

4.4 Ablation Experiments

To further demonstrate the effectiveness of the CASK module, Pi_IoU loss function proposed in this paper, and the introduced DySample on the remote sensing images object detection task. Four sets of ablation experiments are set up on the DIOR dataset to evaluate the performance impact of incorporating each component into the proposed DCI-Net model, and the results are shown in Table 3.

Table 3. Ablation experiments. P is the precision, mAP is the mean average precision, Parameters is the number of parameters, and GFLOPs is the value of floating point operations.

Baseline	DySample	CASK	Pi_IoU	P	mAP@50	mAP@50:95	Parameters	GFLOPs
✓				90.0	90.5	70.5	51.18M	239.9
✓	✓			90.2	90.4	70.6	51.08M	239.2
✓	✓	✓		90.4	90.4	70.5	49.15M	306.5
✓	✓	✓	✓	90.6	90.3	70.6	49.15M	306.5

- 1) DySample: compared to the baseline model, the introduced DySample module achieves a significant improvement of 0.2% in precision and a performance gain of 0.1% in the mAP@50:95 metric. This result is attributed to the optimized design of the DySample module in capturing image geometric information, which enables more efficient extraction and utilization of spatial structural features. In addition, the introduction of the DySample module is accompanied by a reduction in the number of parameters, which reduces the overall complexity of the model, which is a significant advantage in model lightweight design.
- 2) CASK module: the CASK module proposed in this paper similarly achieves 0.2% improvement in accuracy, 4% reduction in the number of parameters, and 27.8% improvement in computing speed. The CASK module enhances the model’s ability to learn the object features through deep learning of the object’s contextual information, which further improves the recognition accuracy. In addition, the CASK module exhibits a higher concentration on bounding box prediction, which demonstrates its potential for fine-grained object localization.
- 3) Pi_IoU loss function: the proposed Pi_IoU loss function effectively improves the accuracy of the model prediction by 0.2% by enhancing the focus on the internal matching and size ratio of the bounding box. This loss function is designed to take into account the geometric consistency of the bounding box, thus achieving better localization accuracy in object detection tasks.

4.5 Visualization

This section demonstrates comparing the DCI-Net model with other state-of-the-art models for detection visualization, as shown in Fig. 4.

From the visualization results in Fig. 4, it can be seen that the model in this paper shows excellent performance in the task of object detection in remote sensing images, especially in terms of object accuracy and the accuracy of bounding box localization, which is significantly better than the other comparative models.

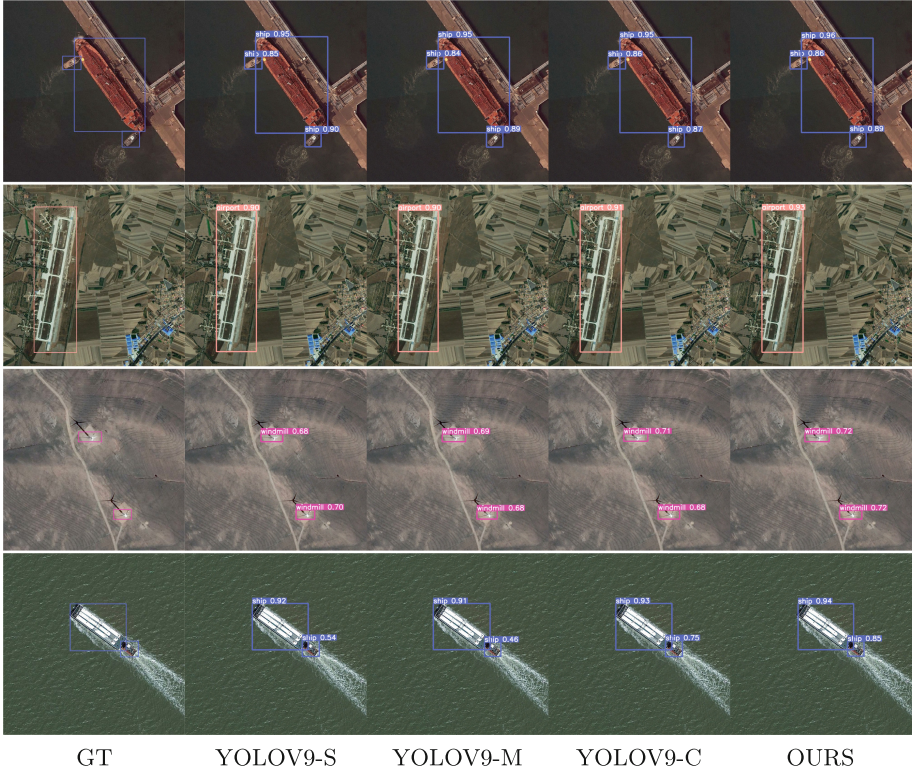


Fig. 4. Visualization results.

In the evaluation of the first set of images, the model proposed in this paper demonstrates significant advantages in object detection accuracy. Especially in recognizing the ship object in the images, the model achieves a high confidence level of 0.96, which reflects the high accuracy of the model detection. In the second set of images, the DCI-Net model outperforms the other models in detecting the category of “airports” with a confidence level of 0.93, which is 2% higher than the secondary high value. In the third set of images, the model in this paper achieves the best confidence level for the category of “windmill” compared with other models by effectively combining the contextual information. In the fourth set of images, the DCI-Net achieves a significant increase in the detection accuracy of small objects by utilizing its highly refined feature extraction capability, and its confidence level is 10% higher than the secondary high value.

5 Conclusion

In this paper, a object detector DCI-Net based on remote sensing images is proposed. This model addresses the challenges posed by multi-scale and high

resolution in remote sensing images. This object detector is based on the current widely recognized YOLOV9 architecture and improves the accuracy of object detection. The CASK module is designed in this paper to extract the object features in the images more accurately, and the representation of the features is significantly improved. The Pi_IoU loss function is proposed and replaces the original loss function to further improve the detection accuracy. The DySample module is introduced to replace the Unsample module, which achieves significant optimization in the number of parameters and other indicators. The experimental results show that the DCI-Net model proposed in this paper can effectively improve the object detection accuracy, and can improve the operation speed and reduce the number of parameters. However, DCI-Net still has subtle deficiencies in mAP@50, and there is room for further improvement in its overall performance. In addition, the model needs more in-depth evaluation and careful optimization in areas such as the accuracy of small object detection. Future research can focus on enhancing the generalizability of the model and optimizing its architecture to meet the application requirements of real-time processing, in the expectation of achieving better performance and a wider range of applications.

References

1. Cheng, G., Han, J.: A survey on object detection in optical remote sensing images. *ISPRS J. Photogramm. Remote. Sens.* **117**, 11–28 (2016)
2. Cheng, G., Si, Y., Hong, H., Yao, X., Guo, L.: Cross-scale feature fusion for object detection in optical remote sensing images. *IEEE Geosci. Remote Sens. Lett.* **18**(3), 431–435 (2020)
3. Cheng, G., et al.: Prototype-CNN for few-shot object detection in remote sensing images. *IEEE Trans. Geosci. Remote Sens.* **60**, 1–10 (2021)
4. Dai, J., Li, Y., He, K., Sun, J.: R-FCN: object detection via region-based fully convolutional networks. In: *Advances in Neural Information Processing Systems*, vol. 29 (2016)
5. Finn, C., Abbeel, P., Levine, S.: Model-agnostic meta-learning for fast adaptation of deep networks. In: *International Conference on Machine Learning*, pp. 1126–1135. PMLR (2017)
6. Guan, W., et al.: Efficient meta-learning enabled lightweight multiscale few-shot object detection in remote sensing images. arXiv preprint [arXiv:2404.18426](https://arxiv.org/abs/2404.18426) (2024)
7. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN. In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2961–2969 (2017)
8. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778 (2016)
9. Kang, B., Liu, Z., Wang, X., Yu, F., Feng, J., Darrell, T.: Few-shot object detection via feature reweighting. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8420–8429 (2019)
10. Law, H., Deng, J.: CornerNet: detecting objects as paired keypoints. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) *Computer Vision – ECCV 2018*. LNCS, vol. 11218, pp. 734–750. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01264-9_45

11. Li, K., Wan, G., Cheng, G., Meng, L., Han, J.: Object detection in optical remote sensing images: a survey and a new benchmark. *ISPRS J. Photogramm. Remote Sens.* **159**, 296–307 (2020)
12. Li, W., Zhou, J., Li, X., Cao, Y., Jin, G., Zhang, X.: InfRS: incremental few-shot object detection in remote sensing images. arXiv preprint [arXiv:2405.11293](https://arxiv.org/abs/2405.11293) (2024)
13. Li, X., Deng, J., Fang, Y.: Few-shot object detection on remote sensing images. *IEEE Trans. Geosci. Remote Sens.* **60**, 1–14 (2021)
14. Li, Y., Hou, Q., Zheng, Z., Cheng, M.M., Yang, J., Li, X.: Large selective kernel network for remote sensing object detection. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16794–16805 (2023)
15. Li, Z., et al.: Deep learning-based object detection techniques for remote sensing images: a survey. *Remote Sens.* **14**(10), 2385 (2022)
16. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2117–2125 (2017)
17. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2980–2988 (2017)
18. Lin, T.-Y., et al.: Microsoft COCO: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) *ECCV 2014*. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_48
19. Lingyun, G., Popov, E., Ge, D.: Fast Fourier convolution based remote sensor image object detection for earth observation. arXiv preprint [arXiv:2209.00551](https://arxiv.org/abs/2209.00551) (2022)
20. Liu, C., Wang, K., Li, Q., Zhao, F., Zhao, K., Ma, H.: Powerful-IoU: more straight-forward and faster bounding box regression loss with a nonmonotonic focusing mechanism. *Neural Netw.* **170**, 276–284 (2024)
21. Liu, N., Celik, T., Li, H.C.: Gated ladder-shaped feature pyramid network for object detection in optical remote sensing images. *IEEE Geosci. Remote Sens. Lett.* **19**, 1–5 (2021)
22. Liu, N., Xu, X., Celik, T., Gan, Z., Li, H.C.: Transformation-invariant network for few-shot object detection in remote-sensing images. *IEEE Trans. Geosci. Remote Sens.* **61**, 1–14 (2023)
23. Liu, S., Qi, L., Qin, H., Shi, J., Jia, J.: Path aggregation network for instance segmentation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8759–8768 (2018)
24. Liu, W., Lu, H., Fu, H., Cao, Z.: Learning to upsample by learning to sample. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6027–6037 (2023)
25. Liu, Y., Li, Q., Yuan, Y., Du, Q., Wang, Q.: ABNet: adaptive balanced network for multiscale object detection in remote sensing imagery. *IEEE Trans. Geosci. Remote Sens.* **60**, 1–14 (2021)
26. Lu, H., Liu, W., Ye, Z., Fu, H., Liu, Y., Cao, Z.: SAPA: similarity-aware point affiliation for feature upsampling. In: *Advances in Neural Information Processing Systems*. NeurIPS (2022)
27. Lu, H., Liu, W., Fu, H., Cao, Z.: FADE: Fusing the assets of decoder and encoder for task-agnostic upsampling. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) *Computer Vision, ECCV 2022*. LNCS, vol. 13687, pp. 231–247. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-19812-0_14
28. Redmon, J.: YOLOv3: an incremental improvement. arXiv preprint [arXiv:1804.02767](https://arxiv.org/abs/1804.02767) (2018)

29. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. In: *Advances in Neural Information Processing Systems*, vol. 28 (2015)
30. Shao, S., et al.: Objects365: a large-scale, high-quality dataset for object detection. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8430–8439 (2019)
31. Tian, Z., Shen, C., Chen, H., He, T.: FCOS: fully convolutional one-stage object detection. arXiv preprint [arXiv:1904.01355](https://arxiv.org/abs/1904.01355) (2019)
32. Wang, C.Y., Yeh, I.H., Liao, H.Y.M.: YOLOv9: learning what you want to learn using programmable gradient information. arXiv preprint [arXiv:2402.13616](https://arxiv.org/abs/2402.13616) (2024)
33. Wolf, S., Meier, J., Sommer, L., Beyerer, J.: Double head predictor based few-shot object detection for aerial imagery. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 721–731 (2021)
34. Yan, X., Chen, Z., Xu, A., Wang, X., Liang, X., Lin, L.: Meta R-CNN: towards general solver for instance-level low-shot learning. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9577–9586 (2019)
35. Zhang, H., Xu, C., Zhang, S.: Inner-IoU: more effective intersection over union loss with auxiliary bounding box. arXiv preprint [arXiv:2311.02877](https://arxiv.org/abs/2311.02877) (2023)
36. Zhang, Q., Liu, Y., Blum, R.S., Han, J., Tao, D.: Sparse representation based multi-sensor image fusion for multi-focus and multi-modality images: a review. *Inf. Fus.* **40**, 57–75 (2018)
37. Zhang, Y., Zhang, B., Wang, B.: Few-shot object detection with self-adaptive global similarity and two-way foreground stimulator in remote sensing images. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **15**, 7263–7276 (2022)
38. Zhou, Y., Hu, H., Zhao, J., Zhu, H., Yao, R., Du, W.L.: Few-shot object detection via context-aware aggregation for remote sensing images. *IEEE Geosci. Remote Sens. Lett.* **19**, 1–5 (2022)



Cross-Modal Ship Grounding: Towards Large Model for Enhanced Few-Shot Learning

Quan Hu^{1,2}, Li Chen^{1,2(✉)}, Zhida Feng^{1,2}, and Yaojie Chen^{1,2}

¹ School of Computer Science and Technology, Wuhan University of Science and Technology, Wuhan, China

chenli@wust.edu.cn

² Hubei Province Key Laboratory of Intelligent Information Processing and Real-Time Industrial System, Wuhan University of Science and Technology, Wuhan, China

Abstract. A growing body of research indicates that employing large models for adaptation to downstream tasks often yields remarkable performance. However, in the domain of ship detection, the potential of these large models is frequently underutilized due to domain shift issues. This paper introduces the Cross-Modal Ship Grounding (CSG) model, which leverages an efficient Cross-Modal Adapter (CMA) technology to transfer the general detection capabilities of large models to ship images, addressing domain shift with minimal training costs. To mitigate the challenges posed by complex and variable background interference, the Water-Land Separation (WLS) module is proposed to focus specifically on the water area. This module effectively addresses the issue of background target interference, thereby enhancing the model's accuracy in complex scenes. Empirical evaluations on both private and public datasets demonstrate that the CSG model surpasses all state-of-the-art models in performance.

Keywords: Ship detection · Large model · Cross-modal · Few-shot

1 Introduction

The application scenarios of ship detection are diverse. With the rapid advancement of digital cameras, intelligent video surveillance systems are increasingly being deployed in ports and coastal areas, facilitating visible ship detection. Through video surveillance, port management systems can automatically assign suitable berthing positions based on the ship detection results, thereby reducing ship waiting times and enhancing the throughput of berthing areas. Additionally, unmanned ships utilize camera-based sensing to perform water operations autonomously according to the detection outcomes.

In recent years, ship detection has garnered increasing attention. Kim et al. [7] combined the Bayesian method with the Faster R-CNN network to achieve high average accuracy in ship detection tasks using a self-constructed dataset

through deep learning techniques. Lee et al. [10] adapted the passthrough approach and improved YOLOv2 to realize real-time detection of 10 types of targets, including speedboats and sailing ships, on the Singapore maritime dataset. Shao et al. [20] released the SeaShips dataset, a public dataset for visual image ship detection, encompassing 31,455 images of six common ship types (ore ships, bulk carriers, general cargo ships, container ships, fishing boats, and passenger ships) and 7,000 pieces of public data, providing a robust database for visual image ship detection. Based on the SeaShips dataset, Shao et al. [19] proposed the Saliency-Aware CNN, which uses coastline segmentation to reduce background interference, narrow the detection area, and enhance the accuracy of ship target positioning by integrating significance maps. However, the introduction of significance maps resulted in a decline in detection efficiency. Liu et al. [13] improved the loss function based on the YOLOv3 model and added uncertain border regression to enhance ship object location capability. Their proposed eYOLOv3 improved the detection of small targets, increasing the average detection accuracy on the SeaShips dataset. Huang et al. [4] enhanced the YOLOv4 algorithm from the theoretical perspectives of feature extraction, feature fusion, and loss function design, applying these improvements to ship detection tasks.

Currently, image-based object detection algorithms, predominantly from the YOLO series and its variants, exhibit limitations in terms of accuracy. Recent advancements in cross-modal object detection, integrating visual and textual information, have demonstrated superior performance. Kamath et al. [5] project visual language features into a multimodal space and introduce contrastive alignment loss to maintain alignment between textual and visual features within this mapped space. Gu et al. [2] employ CLIP’s model for knowledge distillation, transforming the detection task into a proposal classification challenge. Zareian et al. [27] leverage image-text description pairs for learning, utilizing the richer semantic information present in descriptions to enhance model understanding. Liu et al. [14] adopt a dual-encoder single-decoder architecture and integrate semantic information across the network’s neck, query initialization, and head components, achieving state-of-the-art results in zero-shot learning on COCO dataset.

Inspired by the impressive generalization and zero-shot capabilities of large-scale object detection models, this study extends the adaptive approach proposed by Hu et al. [3] to cross-modal applications. We incorporate a learnable adapter module into the existing structure, harnessing the robust generalization and semantic expressiveness of large models while leveraging cross-modal information guidance. Through minimal data training, our method aims to achieve high-precision ship detection.

Furthermore, the presence of ship-like objects on land can sometimes interfere with surface ship detection. The Segment Anything Model (SAM) [8] has demonstrated impressive zero-shot segmentation performance, often rivaling or surpassing previous fully supervised results. The CLIP model [16], trained on 400 million image-text pairs, exhibits strong zero-shot image classification capabilities. By providing SAM with grid point prompts and incorporating text prompts

from CLIP, our approach requires no additional training data. Instead, it leverages the zero-shot capabilities of these large models to distinguish between water and land and filter out interfering objects, thereby further improving ship detection accuracy.

2 Related Work

2.1 Large Models

Large models refer to models trained on extensive datasets that can be adapted to a wide range of downstream tasks, often employing techniques like self-supervised learning, transfer learning, and prompt learning. The Segment Anything Model (SAM) [8] has been introduced for various Computer Vision tasks. SAM utilizes prompt learning with a foundational model to perform multiple tasks on unseen images. However, SAM’s performance on medical images is limited. To address this, Ma et al. [15] constructed a large-scale medical dataset and refined SAM through fine-tuning processes. Zhang et al. [28] applied knowledge distillation to compress SAM into a smaller model suitable for mobile devices, reducing its size by 60 times compared to the original. Ke et al. [6] enhanced segmentation precision by adding parameters and conducting training without altering SAM’s weight significantly.

CLIP [16] utilizes a large-scale dataset of paired images and textual descriptions. It effectively retrieves images based on given text prompts, with applications spanning image classification and generation.

2.2 Visual Grounding

Current methods typically extend object detection frameworks [24,30] to address visual grounding tasks. Two-stage approaches [12,22] initially employ a detector to generate region proposals from the image and subsequently match these proposals with textual inputs to select the most suitable ones. However, this method heavily relies on the accuracy of the detector in the initial stage; if the detector fails to produce correct region proposals, the matching process in the second stage may yield inaccurate results.

In response to these challenges, recent developments have introduced single-stage methods [11,25] aimed at directly predicting target locations without pre-generated region proposals. For instance, FAOA [26] encodes textual inputs into embeddings and integrates them into YOLOv3. The model conducts intensive object detection to identify objects with confidence scores, selecting the highest-ranking object as the reference for prediction. Grounding DINO [14] achieves SOTA performance by integrating text features into three parts of the detector for closed-set, open-set object detection, and visual grounding tasks.

2.3 Few-Shot Learning

After extensive training on large datasets, deep learning models can rapidly adapt to new data with only a small number of samples, a capability known as few-shot learning. Koch et al. [9] trained a dual twin network in a supervised manner and utilized the extracted features for subsequent few-shot learning tasks. Ravi et al. [17] investigated the limitations of gradient-based optimization algorithms when applied to few-shot learning scenarios due to insufficient data. They proposed methods to enhance generalization by iteratively optimizing the few-shot learner to converge effectively on new tasks. Yang et al. [23] advanced few-shot learning research in histopathological images by introducing three cross-domain tasks to simulate real-world clinical challenges. Su et al. [21] introduced a technique for automatically selecting self-supervised learning images tailored to specific datasets from a large pool of unlabeled images.

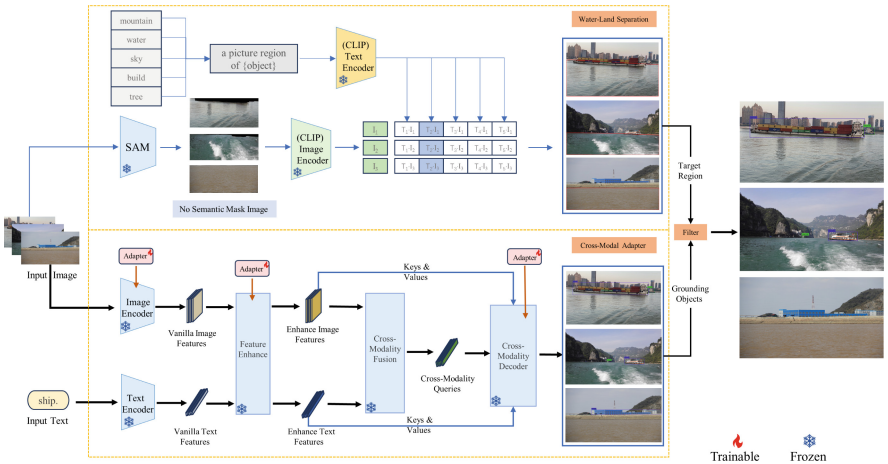


Fig. 1. The CSG architecture is designed such that the upper half of the WLS can be precisely positioned within the target area range, while the lower half of the CMA seamlessly adapts to the unique characteristics of the ship. This approach fully harnesses the pre-training potential of large-scale models, introducing ship detection into the realm of cross-modal applications.

3 Method

The execution process of CSG begins with the CMA extracting multimodal features using two encoders. Subsequently, feature enhancement and fusion processes are employed to decode the output positioning box. To address domain shift challenges, adapters are strategically placed at various locations within the architecture. Additionally, integrating results from the WLS module ensures focused attention on the water surface area, thereby enhancing the accuracy of the final detection results.

3.1 Cross-Modal Adapter

To leverage the extensive knowledge from general object detection for the specialized field of ship detection, we adopt a strategy where we do not completely fine-tune all parameters. Throughout the training process, the text input remains consistent, allowing us to freeze the weights of pre-trained text encoders. Instead, we introduce Adapter modules at specific locations within the architecture. This approach enables us to utilize few-shot learning, requiring only a small amount of data to adapt the representation of the general large model to ship detection tasks.

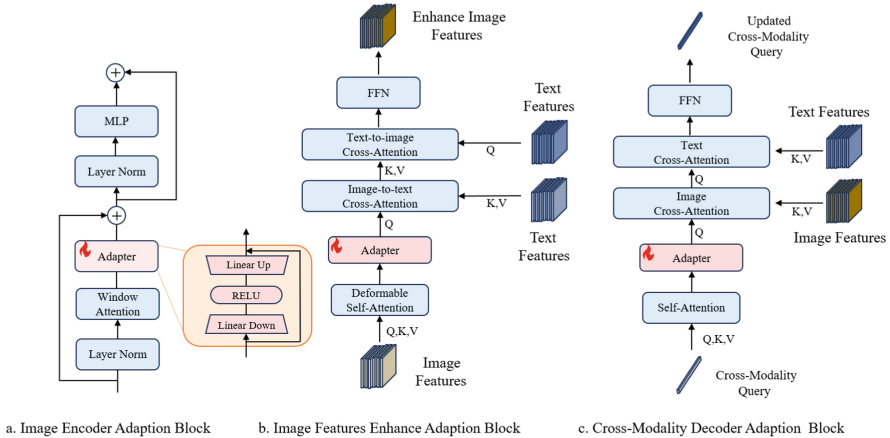


Fig. 2. Cross-Modal Adapter structure. This work does not use the Adapter to learn individual text features, but primarily uses it to learn multimodal features of the larger model.

Each Adapter module consists primarily of three components: down-projection, ReLU activation, and up-projection. The down-projection reduces the dimensionality of the input embedding using a simple MLP layer. Subsequently, the up-projection expands the compressed embedding back to its original dimensionality using another MLP layer. Additionally, a skip connection is employed to add the Adapter’s input directly to the final output. This design ensures that even if the initial parameters of the Adapter are initialized close to zero, the skip connection allows the Adapter to function effectively as an identity map during training. This approach guarantees the Adapter’s effectiveness in learning specific task adaptations.

$$h \leftarrow h + f(hW_{down})W^{up} \quad (1)$$

where h represents Adapter input, W_{down} represents downward projection, W^{up} represents upward projection, and the function f represents non-linear activation.

In the Image Encoder, an Adapter block is incorporated for each Swin Transformer block, positioned within the residual pathway before the MLP layer following window attention (as depicted in Fig. 2(a)). Given the diverse sizes of ship objects, the Adapter module is introduced to fine-tune the Swin Transformer, originally pretrained for general object detection in everyday scenes. Leveraging the Swin Transformer’s capability to hierarchically extract multi-scale image features, this adaptation enables the model to specifically learn and adjust multi-scale features tailored for ship detection tasks.

In the Image Features Enhancer, we integrate an Adapter module following the Deformable Self-Attention of each image feature enhancement layer and preceding the Cross-Attention. Similarly, within the cross-modal decoder, an Adapter block is introduced after the Self-Attention and before the cross-modal Cross-Attention in each layer (as illustrated in Fig. 2(b, c)). The fused multi-modal features exhibit a strong correlation between visual and textual inputs, and the global attention of the large model ensures comprehensive consideration of both large and small object features. By incorporating Adapters, we aim to enhance the alignment between ship object representations and textual prompts, thereby improving the model’s attention to ship features across different scales.

3.2 Water Land Separation

In specific scenarios, we have observed instances where the model erroneously detects objects on land. To address this issue, we propose the Water-Land Separation algorithm, inspired by traditional ship detection methods that detect the sea-sky-line. The aim of this algorithm is to mitigate background interference and concentrate on identifying the relevant areas. This paper adopts a multimodal approach using a large model to implement the WLS algorithm effectively.

SAM achieves segmentation primarily through box and point prompts. Due to the typically large and irregular shape of water areas in images, accurately segmenting water with box prompts can be challenging. Therefore, point prompts are utilized where a point is placed randomly on the water surface, effectively segmenting the entire water area. While achieving a more detailed mask requires additional point prompts, the computational cost increases exponentially. Utilizing an 8×8 grid of points proves adequate, focusing specifically on segmenting water areas efficiently.

Masks without class are generated using point prompts, extracting the corresponding areas which are then passed to the CLIP image encoder. Since these areas do not encompass the entire image, background information is excluded. To preserve more edge details, the mask is expanded.

$$I_E\left(\sum_{i=1}^N I_i * k\right) \quad (2)$$

where I_E represents CLIP’s image encoder. N represents the number of masks generated by SAM. I_i represents the region image. k represents the expansion factor, and k is set to 1.3 in this work.

Use predefined classes such as “mountain”, “water”, “sky”, “building”, “tree”, combined with text to enhance the template. These feed into CLIP’s text encoder, giving the model more precise text instruction.

$$T_E\left(\sum_{i=1}^M \text{Template}(C_i)\right) \quad (3)$$

where T_E represents CLIP’s text encoder. M represents the number of classes. C_i represents the input class. *Template* represents a text enhancement template. Use “a picture region of object” to enhance class input.

CLIP computes similarity scores between masks and different classes, selecting the mask with the highest similarity to water as the desired result. Some model detections include correctly identified ships in water and false detections of objects on land. WLS partitions a water isolation zone.

It is important to note that WLS extends the functionality of the large model to detect water surfaces, achieving the separation of water and land. However, it is not suitable for detecting small targets due to the randomness of point prompts, which may cause target misses. Moreover, generating a point prompt for every pixel in the entire image requires an excessively large amount of computation, making it an impractical solution. Therefore, WLS focuses on selecting water surfaces with strong connectivity and high coverage ratios as the target for segmentation.

4 Experiments

4.1 Dataset

To evaluate the ship grounding performance of the CSG model, this experiment was trained and tested on the SeaShips dataset and RealShips dataset.

SeaShips Dataset. The SeaShips dataset comprises 7,000 images, each with a resolution of 1920×1080 pixels, annotated with precise ship labels and bounding boxes. These images were captured by an on-site video surveillance system deployed around Hengqin Island, Zhuhai, China. The dataset includes a diverse range of ship types, hull sections, scales, viewing angles, lighting conditions, and varying levels of occlusion within complex environments.

RealShips Dataset. The RealShips dataset was captured in real-world conditions and comprises 715 images featuring 1,048 annotated ships with labels and bounding boxes. The dataset includes diverse shooting angles, showcasing ships with varying scales and complex, cluttered backgrounds. It also features ships partially visible and ships overlapping with each other.

4.2 Performance Metric

Precision (P), recall (R), and F1-score ($F1$) are employed to evaluate the detection performance of the model.

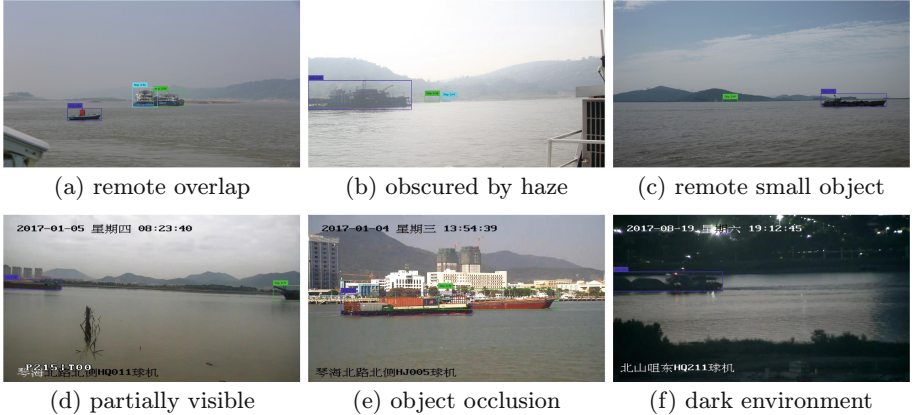


Fig. 3. The detection results (a, b, c from RealShips dataset; d, e, f from SeaShips dataset) demonstrate the robust performance of the CSG model in various complex scenarios. Figure 3(a) and 3(b) showcase the algorithm’s ability to accurately distinguish between closely positioned ships, correctly identifying them as separate entities. Furthermore, CSG effectively locates ships with low visibility, even when obscured by haze. Figure 3(c) illustrates the model’s capability to detect both a distant small ship and a nearby large ship simultaneously, adapting well to scale differences. Real-world scenarios often feature partially captured ships or ships obscuring one another (as shown in Fig. 3(d) and 3(e)). Our algorithm excels in these situations, accurately locating ships even in challenging lighting conditions (Fig. 3(f)). These results underscore the effectiveness and robustness of the proposed CSG method.

IoU is the ratio of the overlap area between the prediction box and the label box of the object detection to the union of their areas. An IoU threshold is usually set, which we set as 0.5 in the experiment.

$$IoU = \frac{B_{pred} \cap B_{gt}}{B_{pred} \cup B_{gt}} \quad (4)$$

where B_{gt} is the area of annotated ground truth box and B_{pred} is the area of predicted bounding box.

It needs to calculate the IoU to determine whether a test result is correct or wrong. So, if the IoU is > 0.5 , we consider the test result to be True Positives. If $IoU < 0.5$ or the same GT is detected with redundant detection boxes, we consider the detection result to be a False Positive. FN is originally a real box, but not detected. Refers to the number of GT boxes that are not detected.

4.3 Experiment Results

We conducted a comparative evaluation of the CSG model against other SOTA methods for ship detection, including Grounding DINO [14], YOLOv8 [18], DETR [1], and RT-DETR [29]. Experiments were performed on two datasets: the

Table 1. The experimental results are compared with other methods in SeaShips. Best results are denoted as **bold**.

Models		F1↑	P↑	R↑
Zero-shot				
Grounding DINO(tiny)		0.710	0.627	0.816
Grounding DINO(base)		0.739	0.675	0.816
YOLOv8-L		0.631	0.694	0.579
YOLOv8-X		0.643	0.655	0.631
DETR(r50)		0.414	0.346	0.516
DETR(r101)		0.481	0.403	0.595
RT-DETR(r50)		0.649	0.716	0.593
RT-DETR(r101)		0.713	0.725	0.701
Fine-tuning	Turnable Param(M)↓			
Grounding DINO(tiny)	65M	0.831	0.790	0.876
Grounding DINO(base)	127M	0.853	0.828	0.881
YOLOv8-L	43M	0.865	0.872	0.859
YOLOv8-X	68M	0.869	0.865	0.873
DETR(r50)	41M	0.821	0.800	0.843
DETR(r101)	60M	0.879	0.859	0.899
RT-DETR(r50)	42M	0.867	0.845	0.890
RT-DETR(r101)	76M	0.892	0.887	0.897
CSG(ours)	4M	0.901	0.899	0.902

public SeaShips dataset, with results summarized in Table 1, and the proprietary RealShips dataset we curated, presented in Table 2. Our evaluation encompassed two methodologies: Zero-shot experiments and Fine-tuning experiments.

In Table 1, the Zero-shot performance of the four methods exhibits significant variation. Grounding DINO (base) achieved the highest $F1$ score, surpassing DETR (r50) by 32.3%, highlighting the pivotal role of encoding strategies in algorithm accuracy. Notably, using the same algorithm with different backbone scales yields notably different results. Specifically, DETR (r101) outperforms DETR (r50) by 6.7%, indicating that deeper network layers with more parameters can learn richer representations.

During the fine-tuning of Grounding DINO, we followed a typical approach for large model fine-tuning: freezing the text encoder and training the remaining parameters. This strategy aligns with common transfer learning practices where models are pre-trained on large, general-purpose datasets before fine-tuning on task-specific data. As shown in Table 1, fine-tuning improves the $F1$ score by an average of approximately 20%. Effective training strategies can achieve or even surpass the performance of full-parameter fine-tuning while using fewer parameters. CSG achieved SOTA performance with only 4 M trained parameters.

Table 2. The experimental results are compared with other methods in RealShips. Best results are denoted as **bold**.

Models		F1↑	P↑	R↑
Zero-shot				
Grounding DINO(tiny)		0.704	0.758	0.657
Grounding DINO(base)		0.780	0.729	0.838
YOLOv8-L		0.697	0.669	0.728
YOLOv8-X		0.714	0.610	0.862
DETR(r50)		0.470	0.366	0.656
DETR(r101)		0.535	0.429	0.711
RT-DETR(r50)		0.779	0.771	0.786
RT-DETR(r101)		0.790	0.764	0.817
Fine-tuning	Turnable Param(M)↓			
Grounding DINO(tiny)	65M	0.811	0.851	0.774
Grounding DINO(base)	127M	0.892	0.927	0.859
YOLOv8-L	43M	0.805	0.796	0.814
YOLOv8-X	68M	0.837	0.812	0.863
DETR(r50)	41M	0.887	0.939	0.841
DETR(r101)	60M	0.889	0.934	0.848
RT-DETR(r50)	42M	0.891	0.912	0.870
RT-DETR(r101)	76M	0.909	0.941	0.879
CSG(ours)	4M	0.926	0.974	0.882

In comparison, Grounding DINO (base) utilizes 30 times more parameters than CSG, yet achieves a slightly lower $F1$ score by 4.2%.

In Table 2, it’s observed that larger backbone architectures generally exhibit better robustness for the same algorithm. The Zero-shot performance shows considerable variation between the SeaShips and RealShips datasets. Specifically, the difference in $F1$ score between Grounding DINO (base) and Grounding DINO (tiny) is 2.9% in Table 1, while it increases to 7.6% in Table 2. This discrepancy can be attributed to the SeaShips dataset images being captured from adjacent frames in a video, resulting in high similarities between images. Conversely, the RealShips dataset contains images with diverse backgrounds and angles, providing a more challenging test of the model’s generalization capability.

4.4 Ablation Study

We conducted a comprehensive ablation study to evaluate the effectiveness of WLS and CMA as proposed in Table 3. The baseline (first row) involved fine-tuning pre-training weights using Grounding DINO, with the text encoder frozen and the remaining parameters trained. Introducing WLS on top of this baseline significantly reduced false positives, demonstrating the algorithm’s ability to

Table 3. Ablation Study in RealShips. Best results are denoted as **bold**.

WLS	CMA	F1↑	P↑	R↑
		0.892	0.927	0.859
✓		0.898	0.941	0.859
	✓	0.918	0.959	0.882
✓	✓	0.926	0.974	0.882

accurately focus on the water surface area and filter out background interference. Incorporating the CMA module into the baseline model further reduced false positives and missed detections, indicating improved transfer of large model representation capabilities from general images to ship images with minimal training resources required.

4.5 Limitation

The proposed CSG method can achieve high precision ship grounding with minimal training cost. But it has some limitations. Due to the large number of parameters, large models may not be as fast as small models in terms of reasoning speed. In real-time inspection applications, the computation time and processing delay of CSG may not meet the requirements of some specific situations. Therefore, we will work hard on the lightweight of large models in the future, so that they can be applied to scenarios that require high-speed inference.

5 Conclusion

This paper introduces CSG, an extension of large object detection models specifically tailored for ship detection, evaluated using the RealShips dataset under complex backgrounds. By integrating Cross-modal Adapter modules strategically within the architecture, CSG achieves effective few-shot learning capabilities, leveraging the large model’s ship detection prowess with minimal training data. Leveraging the zero-shot ability of the large model, CSG accurately separates water and land areas, effectively eliminating land-based object interference. Experimental results underscore the effectiveness of our approach in ship detection and highlight the potential for large models to excel in diverse downstream tasks.

Acknowledgement. This work was supported by National Natural Science Foundation of China (62271359).

References

1. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: European Conference on Computer Vision, pp. 213–229. Springer, Cham (2020)
2. Gu, X., Lin, T.Y., Kuo, W., Cui, Y.: Open-vocabulary object detection via vision and language knowledge distillation. In: International Conference on Learning Representations (2021). <https://api.semanticscholar.org/CorpusID:238744187>
3. Hu, J.E., et al.: Lora: low-rank adaptation of large language models. arXiv abs/2106.09685 (2021). <https://api.semanticscholar.org/CorpusID:235458009>
4. Huang, Q., Sun, H., Wang, Y., Yuan, Y., Guo, X., Gao, Q.: Ship detection based on yolo algorithm for visible images. IET Image Process. (2023)
5. Kamath, A., Singh, M., LeCun, Y., Synnaeve, G., Misra, I., Carion, N.: MDETR-modulated detection for end-to-end multi-modal understanding. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 1780–1790 (2021)
6. Ke, L., et al.: Segment anything in high quality. arXiv preprint [arXiv:2306.01567](https://arxiv.org/abs/2306.01567) (2023)
7. Kim, K., Hong, S., Choi, B., Kim, E.: Probabilistic ship detection and classification using deep learning. Appl. Sci. **8**(6), 936 (2018)
8. Kirillov, A., et al.: Segment anything. In: 2023 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 3992–4003 (2023). <https://api.semanticscholar.org/CorpusID:257952310>
9. Koch, G., Zemel, R., Salakhutdinov, R., et al.: Siamese neural networks for one-shot image recognition. In: ICML Deep Learning Workshop, vol. 2. Lille (2015)
10. Leela, S., Roh, M.I., Ohb, M.: Image-based ship detection using deep learning. Ocean Syst. Eng. **10** (2020)
11. Liao, Y., et al.: A real-time cross-modality correlation filtering method for referring expression comprehension. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10880–10889 (2020)
12. Liu, D., Zhang, H., Wu, F., Zha, Z.J.: Learning to assemble neural module tree networks for visual grounding. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4673–4682 (2019)
13. Liu, R.W., Yuan, W., Chen, X., Lu, Y.: An enhanced CNN-enabled learning method for promoting ship detection in maritime surveillance system. Ocean Eng. **235**, 109435 (2021)
14. Liu, S., et al.: Grounding dino: marrying dino with grounded pre-training for open-set object detection. arXiv abs/2303.05499 (2023). <https://api.semanticscholar.org/CorpusID:257427307>
15. Ma, J., He, Y., Li, F., Han, L., You, C., Wang, B.: Segment anything in medical images. Nat. Commun. **15**(1), 654 (2024)
16. Radford, A., et al.: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning, pp. 8748–8763. PMLR (2021)
17. Ravi, S., Larochelle, H.: Optimization as a model for few-shot learning. In: International Conference on Learning Representations (2016). <https://api.semanticscholar.org/CorpusID:67413369>
18. Reis, D., Kupec, J., Hong, J., Daoudi, A.: Real-time flying object detection with yolov8. arXiv preprint [arXiv:2305.09972](https://arxiv.org/abs/2305.09972) (2023)

19. Shao, Z., Wang, L., Wang, Z., Du, W., Wu, W.: Saliency-aware convolution neural network for ship detection in surveillance video. *IEEE Trans. Circuits Syst. Video Technol.* **30**(3), 781–794 (2019)
20. Shao, Z., Wu, W., Wang, Z., Du, W., Li, C.: Seaships: a large-scale precisely annotated dataset for ship detection. *IEEE Trans. Multimedia* **20**, 2593–2604 (2018). <https://api.semanticscholar.org/CorpusID:52285314>
21. Su, J.C., Maji, S., Hariharan, B.: When does self-supervision improve few-shot learning? In: *European Conference on Computer Vision*, pp. 645–666. Springer, Cham (2020)
22. Wang, P., Wu, Q., Cao, J., Shen, C., Gao, L., Hengel, A.V.D.: Neighbourhood watch: referring expression comprehension via language-guided graph attention networks. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1960–1968 (2019)
23. Yang, J., Chen, H., Yan, J., Chen, X., Yao, J.: Towards better understanding and better generalization of few-shot classification in histology images with contrastive learning. *arXiv preprint arXiv:2202.09059* (2022)
24. Yang, L., et al.: Pdnet: toward better one-stage object detection with prediction decoupling. *IEEE Trans. Image Process.* **31**, 5121–5133 (2022)
25. Yang, Z., Chen, T., Wang, L., Luo, J.: Improving one-stage visual grounding by recursive sub-query construction. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) *ECCV 2020. LNCS*, vol. 12359, pp. 387–404. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58568-6_23
26. Yang, Z., Gong, B., Wang, L., Huang, W., Yu, D., Luo, J.: A fast and accurate one-stage approach to visual grounding. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4683–4693 (2019)
27. Zareian, A., Rosa, K.D., Hu, D.H., Chang, S.F.: Open-vocabulary object detection using captions. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14393–14402 (2021)
28. Zhang, C., et al.: Faster segment anything: towards lightweight SAM for mobile applications. *arXiv preprint arXiv:2306.14289* (2023)
29. Zhao, Y., et al.: DETRs beat YOLOs on real-time object detection. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16965–16974 (2024)
30. Zhu, X., Ma, Y., Wang, T., Xu, Y., Shi, J., Lin, D.: SSN: shape signature networks for multi-class object detection from point clouds. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) *ECCV 2020. LNCS*, vol. 12370, pp. 581–597. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58595-2_35



STNet: Small Target Detection Network for IR Imagery

Nikhil Kumar^{1,2}, Pranav Singh Chib¹, and Pravendra Singh¹

¹ Indian Institute of Technology, Roorkee, Roorkee, India
{pranavs_chib, pravendra.singh}@cs.iitr.ac.in

² Instruments Research and Development Establishment, Dehradun, India
nikhil_k1@cs.iitr.ac.in

Abstract. Despite advancements in technology, including deep learning techniques, Single-frame InfraRed Small Target (SIRST) detection in InfraRed (IR) imagery remains challenging, requiring further research and innovation. The lack of high-level semantic information causes small IR target features to diminish in the deeper layers of convolutional neural networks, reducing the network's ability to accurately represent and identify these targets. This paper proposes a novel SIRST detection approach, STNet (Small Target detection Network), built on the U2-Net (Nested U-shape Network) architecture. STNet incorporates two key components: the MultiLayer Feature Fusion (MLFF) module and the Fast Fourier Block (FFB). The MLFF module enhances the model's ability to integrate and leverage features from multiple layers, combining low-level details with high-level semantic information for more accurate SIRST detection. The FFB further improves the model's performance by enabling feature extraction in the frequency domain, preserving small target features in the deep layers of the network, which enhances the performance of the detection process. Experimental results on the NUDT-SIRST and IRSTD-1K datasets show that STNet consistently outperforms other state-of-the-art methods. On the NUDT-SIRST dataset, STNet achieves the highest performance with an IoU of 87.25%, nIoU of 87.23%, and Pd of 98.51%, coupled with a low FA of 5.92×10^{-6} . Similarly, on the IRSTD-1K dataset, STNet achieves the best performance with 72.04% IoU, 68.95% nIoU, 95.29% Pd, and 1.92×10^{-6} FA. These results underscore STNet's effectiveness in detecting and segmenting small IR targets in cluttered backgrounds.

Keywords: Infrared Small Target Detection · SIRST Detection · Deep Learning · Multi Layer Feature Fusion Module · Fast Fourier Block

1 Introduction

Tasks like military surveillance, search and rescue missions, and environmental monitoring often demand precise identification of small targets [4–6, 16] within cluttered and complex backgrounds [18], where the ability to identify subtle

target signatures can be the difference between success and failure. For instance, in military operations, accurately detecting aerial targets or ground targets from long ranges can provide a strategic advantage.

A small target is defined as one that occupies less than 0.12% of the total pixels in an image, meaning that in a 320×256 image, targets smaller than 98 pixels fall into this category. Detecting such small targets at considerable distances, sometimes extending to several hundred kilometers, is crucial in many military scenarios. At these distances, IR sensors can only perceive distant targets with small angular sizes and limited pixel-based target signatures. This makes the detection process highly complex, as the targets appear dim and blend into the background clutter. Successfully identifying these small targets is essential for effective surveillance and defense operations, as it can significantly impact strategic decision-making and response times. SIRST methods are particularly valuable in scenarios where quick and accurate detection of small targets is crucial. The ability to detect targets in a single frame allows for real-time processing, which is essential for applications requiring immediate response and decision-making. Consequently, a wide range of SIRST detection methods have been developed, each aiming to improve detection accuracy, reduce false alarms, and enhance performance in cluttered and dynamic environments. Furthermore, the complexity of detecting these targets is increased by factors such as noise, clutter, and varying environmental conditions. These factors can obscure the faint thermal signatures of small targets, posing a significant challenge for traditional detection methods. To address these challenges, innovative approaches are required to enhance the representation of small target features and improve overall detection accuracy in complex IR imagery.

Small targets in IR images are typically areas of high intensity within cluttered backgrounds, making them visually salient but lacking specific semantic information [1, 6]. To effectively detect and segment these targets at the pixel level, it is essential to focus on the most salient regions. Recognizing these targets as both salient and small in IR images necessitates multi-level deep feature integration. The methods must integrate low-level details with high-level semantic features to ensure precise localization and segmentation of small IR targets.

In this work, we propose a novel approach STNet (**S**mall **T**arget detection **N**etwork), an encoder-decoder paradigm that employs feature fusion to combine low-level detail features with high-level semantic features. To achieve this Multi Layer Feature Fusion (MLFF) module has been proposed, which significantly improves performance through hierarchical feature fusion. Furthermore, STNet performs feature extraction in both the spatial and frequency domains using Fast Fourier Block (FFB), which enriches the extracted features. Apart from this, the fast Fourier block also plays a crucial role in preventing the vanishing of small target features in deep layers. By performing feature extraction in the frequency domain, the FFB ensures that small but significant features are preserved throughout the network. This is particularly important for maintaining the integrity of small IR targets, which can be easily lost in deeper layers [17, 31] of conventional networks. Experimental results on the NUDT-SIRST

and IRSTD-1K datasets show that STNet consistently outperforms other state-of-the-art methods. Empirical evaluations underscore STNet’s effectiveness in detecting and segmenting small IR targets in cluttered backgrounds.

2 Related Work

2.1 Traditional Paradigm for SIRST

There exist several algorithms for detecting small and dim targets in the IR domain. Initially, researchers relied on conventional image processing techniques. These approaches include human vision system (HVS) inspired methods for small target detection in IR imagery, utilizing various local contrast measures to enhance target signals and suppress background clutter. LCM [5], ILCM [14], RLCM [13], HBMLCM [28], TLLCM [15], MPCM [33], DLCM [24] WLDM [9] are major algorithms under this category. Entropy-based small and dim target detection techniques [8, 27, 38] are also rooted in conventional image processing approaches. These methods employ various techniques to enhance signal-to-noise ratios and improve target differentiation. MGDWE [8], LEF [38], LR [27] are examples under this category. Gradient-based methods [35, 41] represent another significant category within conventional image processing for small and dim target detection. These methods leverage gradient properties to enhance target visibility and suppress background clutter. Representative algorithms include LIG [41] and DGRAD [35]. Background reconstruction [1–3, 10, 22] a key approach of image processing plays a crucial role in small target detection algorithms, significantly enhancing their ability to identify targets by effectively differentiating them from complex backgrounds. Prominent examples of this approach include AAGD [1], ADMD [22], THM [12], MTHM [2, 3], MAXMEAN [10] and MAXMED [10]. These approaches, while effective to some extent, often struggled with varying background conditions and noise.

2.2 Deep Learning Paradigm for SIRST

The emergence of deep learning, especially CNNs, has significantly advanced image detection tasks. The field of fully supervised detection features a variety of innovative approaches. YOLO-FR [23] is known for its fully supervised detection type. ACM [6] employs customized down-sampling with attention modules, while ALCnet [7] integrates feature learning with a model-driven approach. DNAnet [17] uses a nested U-Net with attention mechanisms. MDvsFA [32] leverages a GAN-based method, and ISNet [43] incorporates Taylor finite difference-based attention. DBR [25] stands out with its vision transformer-based strategy. MPAnet [30] utilizes axial attention modules, and Ganet [40] focuses on global attention. RDIAN [29] is a receptive-field and direction-induced attention network designed to address the interclass imbalance between targets and backgrounds by leveraging the characteristics of target size and grayscale. AGPCNet [44] includes an Attention-Guided Context Block (AGCB) for computing local

and global associations, a context pyramid module for integrating features from multi-scale AGCBs, and an asymmetric fusion module for enhancing feature utilization by integrating low-level and deep-level semantics. RepISD-Net [36] introduced an edge compensation block to improve local salient features and capture finer contour details of small targets. UIU-Net [37] integrates a small U-Net within a larger U-Net framework, facilitating multi-level and multi-scale representation learning of objects. SRNet [20] provides a unified framework for learning shape-based representations, enhancing SIRST detection by explicitly integrating shape information into the model’s learning process. SSPS [16] introduced single-point supervision combined with Monte Carlo linear clustering. CSENet [19] integrated a contrast-shape encoder and a shape-reconstructable decoder in a cascading manner to learn discriminative representations for effectively identifying target objects. DCFR [11] is a diffusion-based continuous feature representation network equipped with a dedicated block to accurately capture the contours of extremely small targets. RPCANet [34] addresses the detection task by performing sparse target extraction, low-rank background estimation, and image reconstruction within a relaxed robust principal component analysis model. IRPruneDet [42] represents the weight matrix in the wavelet domain and formulates a wavelet channel pruning strategy. However, applying CNNs to SIRST detection has highlighted certain limitations. The lack of high-level semantic information causes small target features to weaken in the deeper layers of the CNN, reducing the network’s ability to effectively represent and detect these small targets. Our STNet addresses this shortcoming by using multi-layer feature fusion and feature extraction in the frequency domain.

3 Methodology

In this section, we first introduce the underlying architecture of the STNet, which is based on the Residual U-Net [26]. Then, we will delve into the Multi-Layer Feature Fusion (MLFF) module that fuses the extracted multi-layer features, which are then used by the subsequent decoder. Furthermore, due to the deep layers, the target may suffer from the vanishing problem. To overcome this, we leverage the Fast Fourier Block (FFB) to capture local and global context and avoid the vanishing problem of small targets.

3.1 STNet

The architecture underlying the proposed STNet is a stacked U-Net with a dual-level nested U-Structure [26] as shown in Fig. 1. This configuration facilitates the extraction of diverse multi-scale and multi-level features. STNet consists of encoders and decoders, each utilizing a Residual U-block. Within the Residual U-block, the input transforms into intermediate feature maps that capture local features. These feature maps undergo progressive down sampling to extract multi-scale features at varying levels of detail. Subsequently, these multi-scale features are up sampled through convolution to produce high-resolution feature

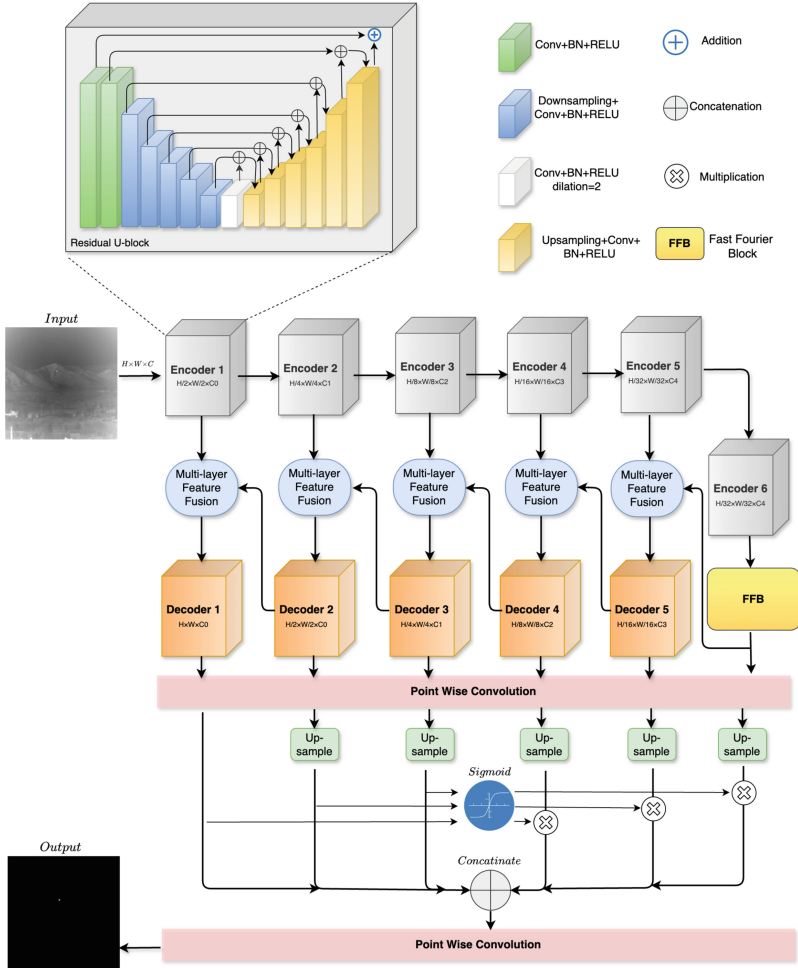


Fig. 1. The overview of our proposed STNet. The architecture is based on U-Net structures, where each encoder-decoder pair follows a residual U-block structure. Feature maps are gradually downsampled by encoders and subsequently upsampled by decoders. We introduce a Fast Fourier Block (FFB) at the bottleneck encoder to capture global features. Furthermore, the MLFF module combines subsequent encoder and decoder features, which are then passed to the next decoder. We concatenate deep and shallow features from the decoder to obtain the final output.

maps. Residual connections are employed to merge local features and multi-scale features, preserving detailed local and multi-scale information.

The input IR image is passed on to six encoders that integrate a Residual U-block with feature map down sampling. Furthermore, as shown in Fig. 1 STNet incorporates five decoder blocks, where each decoder upsamples the feature map

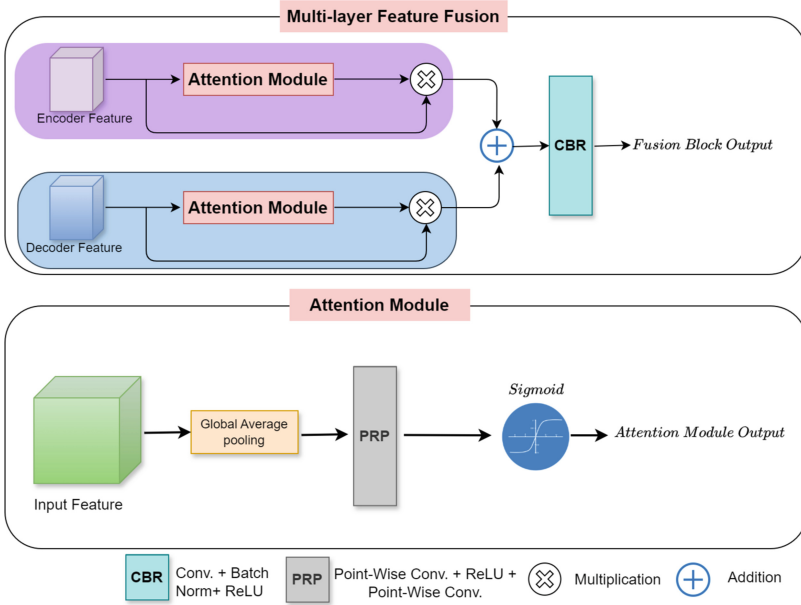


Fig. 2. The illustration of the Multi-Layer Feature Fusion (MLFF module) and Attention Module.

from the previous stage and fuses it with the corresponding encoder output in the MLFF module, serving as input for the next subsequent decoder.

To integrate the outputs globally and dynamically from each decoder, we merge the shallow outputs (from decoders D1, D2, and D3) and the deep outputs (from decoders D4, D5, and encoder E6) using a scaling mechanism. As illustrated in Fig. 1, shallow outputs undergo point-wise convolution and upsampling to generate feature maps. Similarly, deep outputs undergo point-wise convolution and upsampling to obtain their respective feature maps. Then, we derive scaling weights using the following steps: performing point-wise convolution, upsampling (for D2 and D3), concatenating the outputs, and finally applying a sigmoid function. Each feature map from the deep outputs is multiplied by these scaling weights. Finally, scaled deep output and shallow outputs are concatenated and undergo point-wise convolution to produce the final output of STNet.

3.2 Multi-layer Feature Fusion Module

Figure 2 illustrates the proposed MLFF module, which also includes an integrated attention module. The encoder increases the receptive field and extracts high-level information, while the decoder restores the size of feature maps to match that of the input images. To achieve gradual fusion, we introduced an MLFF module. The features from the subsequent encoder and decoder are passed

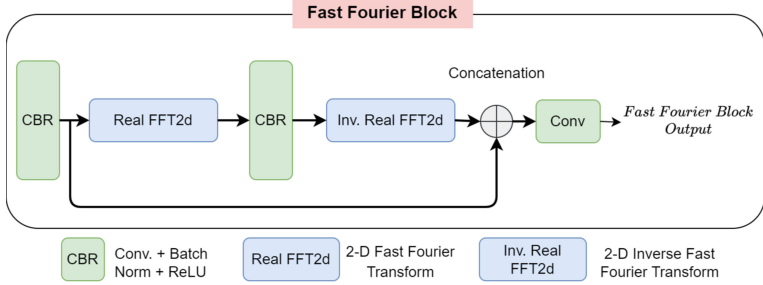


Fig. 3. The illustration of the Fast Fourier Block (FFB). FFB converts features from the spatial domain into the frequency domain and applies the CBR (convolution, batch normalization, and ReLU) layer in the frequency domain. We then use the inverse Fourier transform, a residual connection, and convolution to capture local and global information.

to the attention module, and a residual connection is used to retain the input features. Both features are fused by element-wise addition and passed to the convolution, batch norm, and ReLU layers.

The attention module is used for feature enhancement and to improve representation. The feature map for the respective encoder/decoder is passed to the attention module. The attention process can be summarized in Eq. 1 and Fig. 2 (Lower). The input features (L) are passed through global average pooling and then processed by the PRP (Point-wise convolution, ReLU, and Point-wise convolution) layer. The sigmoid activation function is then applied to get the final output from the attention module.

$$M_a(L) = \sigma((\text{PRP}(\mathcal{P}_{\text{avg}}(L)))) \quad (1)$$

Here, L represents the input features, and \mathcal{P}_{avg} denotes the global average pooling operation. PRP represents the point-wise convolution, ReLU, and point-wise convolution operations. σ is a sigmoid function.

3.3 Fast Fourier Block

As depicted in Fig. 3, our approach utilizes FFB block with a Fast Fourier Transform (FFT) operation to extract global information using convolution in the frequency domain. This technique is crucial for preserving small target features in deep network layers. The FFT block analyzes the frequency components of the image, allowing for the separation of high-frequency target signals from the lower-frequency background components. By amplifying these high-frequency components associated with the targets, the FFT block improves the signal-to-noise ratio, making the targets more distinguishable.

The FFB operates as a 2D block based on Real FFT, transforming spatial features into the frequency domain to capture broader context. The process involves several steps: initial convolution, followed by batch normalization and

ReLU activation; then, applying Real FFT2d to convert spatial features into complex frequency representations, extracting both real and imaginary parts as shown in Eq. 2:

$$\text{Real FFT2d} : \mathbb{R}^{H \times W \times C} \rightarrow \mathbb{C}^{H \times \frac{W}{2} \times C} \quad (2)$$

After concatenating real and imaginary parts, another round of convolution, batch normalization, and ReLU activation is applied within the frequency domain. Notably, Real FFT operates on real-valued signals, ensuring the output remains in the real domain. To revert to the spatial domain while preserving features, we employ the inverse Real FFT operation as described in Eq. 3:

$$\text{Inverse Real FFT2d} : \mathbb{C}^{H \times \frac{W}{2} \times C} \rightarrow \mathbb{R}^{H \times W \times C}. \quad (3)$$

Here, H , W , and C denote the height, width, and number of channels of the feature map, respectively, with \mathbb{R} and \mathbb{C} representing the real and complex domains.

4 Experiment

4.1 Datasets

IRSTD-1K [43] dataset is a real-world image collection featuring 1000 IR images, each with dimensions of 512×512 pixels, showcasing diverse backgrounds such as sea, river, field, mountain, city, and cloud scenes. Each image is annotated with ground truth data that includes targets classified into three types: point, spot, and extended. To ensure precise annotations, the targets within these images have been meticulously labeled at the pixel level. This extensive and varied dataset is ideal for advancing and benchmarking small target detection algorithms. The diversity of backgrounds and target types makes IRSTD-1K a valuable resource for research in small target detection in IR imagery.

NUDT-SIRST [17] dataset consists of a total of 1,327 synthetically generated images, encompassing a diverse array of background scenarios such as clouds, urban environments, and maritime scenes. Accompanying the dataset is ground truth data, which provides valuable information for accurate evaluation and benchmarking. The dataset includes both point and extended targets, offering a comprehensive resource for developing and testing small target detection methods in the IR domain.

4.2 Performance Metric

In this work, performance evaluation has been conducted at both the pixel level and the object level. Following previous works, we use the following metrics:

Metrics Defined at Pixel Level: Intersection over Union (IoU) is a fundamental evaluation metric used in computer vision, particularly for object detection and image segmentation tasks. It measures the accuracy of a predicted bounding box or segmentation mask by comparing it with the ground truth. It is expressed as given in the following Eq. 4:

$$IoU = \frac{A_i}{A_u} = \frac{\sum_{i=1}^n TP_i}{\sum_{i=1}^n T_i + P_i - TP_i}, \quad (4)$$

where A_i and A_u are the intersection and union, respectively. T denotes the pixels predicted as the targets. P denotes the pixels of the ground truth targets. TP is the true positive pixels. n represents the number of IR images in the test set.

Normalized Intersection over Union (nIoU) is a normalized version of the traditional IoU metric, which is particularly important in datasets where objects vary significantly in size [6]. It is expressed as the following Eq. 5:

$$nIoU = \frac{1}{n} \sum_{i=1}^n \frac{TP_i}{T_i + P_i - TP_i}. \quad (5)$$

Metrics Defined at Object Level: The object level metrics for Probability of Detection P_d and False Alarm rate F_a are defined in the following Eqs. 6 and 7.

$$P_d = \frac{1}{n} \sum_{i=0}^n \frac{N_i^{\text{pred}}}{N_i^{\text{all}}} \quad (6)$$

$$F_a = \frac{1}{n} \sum_{i=0}^n \frac{P_i^{\text{false}}}{P_i^{\text{all}}} \quad (7)$$

Here, N^{pred} represents the number of correctly detected objects, while N^{all} signifies the total number of objects. Similarly, P^{false} indicates the pixels of falsely detected objects and P^{all} denotes the total pixels of objects. A detection is considered accurate when the distance between the centroid of the predicted result and the ground truth is less than 3 pixels [17].

4.3 Implementation Details

Experiments were conducted using Python 3.8.13 and PyTorch Version 1.13.1 + cu117. The training phase utilized an NVIDIA RTX A5000 GPU and an AMD EPYC 7543 CPU, with a batch size of 8 over 600 epochs. The Adan optimizer [39] was used, featuring 10 warm-up epochs and a learning rate of $1e^{-3}$. Input images were resized to 512×512 for both training and testing. The binary cross-entropy loss was applied with a weight decay of $1e^{-4}$. The STNet architecture

consists of 6 encoder blocks and 5 decoder blocks. In the PRP block (Sect. 3.2), a point-wise convolution with a kernel size of 1 is used, and a reduction factor of 16 is used to reduce the output channels by a factor of 16. After ReLU activation, this is followed by another point-wise convolution with a kernel size of 1, and the reduced channels are increased by a factor of 16 in output to match the original number of channels. The training time of STNet on IRSTD-1k is 43s per epoch, and the test inference time is 45 milliseconds per infrared image sample.

Table 1. Performance Comparison of various SOTA Deep Learning based methods for SIRST detection on IRSTD-1K Dataset. \uparrow arrow signifies that a higher value is better, and \downarrow arrow signifies that a lower value is best.

Method	Venue	Pixel Level		Object Level	
		IoU(%) \uparrow	nIoU(%) \uparrow	Pd(%) \uparrow	FA(10^{-6}) \downarrow
MDvsFA [32]	ICCV-2019	49.03	46.94	82.49	51.93
ACM [6]	WACV 2021	60.28	57.00	89.90	18.11
ALCNet [7]	TGRS-2021	62.63	60.70	89.23	19.28
DNA-Net [17]	TIP-2022	62.66	60.86	90.24	9.565
ISNet [43]	CVPR-2022	61.80	62.27	89.56	2625
RDIAN [29]	TGRS-2023	58.33	60.73	91.92	26.53
APGCNet [44]	TAES-2023	62.82	63.01	90.57	29.72
RepsISD [36]	TGRS-2023	65.45	–	91.59	7.62
UIU-Net [37]	TIP-2023	62.49	61.91	91.88	21.65
SRNet [20]	TMM-2023	69.45	65.51	96.77	13.05
SSPS [16]	ICCV-2023	64.13	–	90.74	14.93
CSENet [19]	TIP-2024	66.70	65.87	98.16	12.08
DCFR [11]	TGRS-2024	65.41	65.45	96.30	7.345
RPCANet [34]	WACV-2024	–	63.21	88.31	4.39
IRPruneDet [42]	AAAI-2024	64.54	62.71	91.74	16.04
MSHNe [21]	CVPR-2024	67.16	–	93.88	15.03
STNet(Ours)	-	72.04	68.95	95.29	1.92

5 Comparison to State-of-the-Art Methods

The performance comparison of SOTA deep learning-based methods for SIRST reported on the IRSTD-1K dataset is detailed in Table 1, focusing on metrics such as IoU, nIoU, Pd, and FA. Our proposed STNet stands out with the highest IoU of 72.04% and nIoU of 68.95%, along with a Pd of 95.29% and the lowest FA of 1.92×10^{-6} . Other noteworthy methods include SRNet, which achieved an IoU

Table 2. Performance Comparison of various Deep Learning based SIRST methods for NUDT-SIRST dataset. \uparrow arrow signifies that a higher value is better, and \downarrow arrow signifies that a lower value is best.

Method	Venue	Pixel Level		Object Level	
		IoU(%) \uparrow	nIoU(%) \uparrow	Pd(%) \uparrow	FA(10^{-6}) \downarrow
MDvsFA [32]	ICCV-2019	67.01	60.78	90.56	27.25
ACM [6]	WACV 2021	63.33	65.86	93.26	52.88
ALCNet [7]	TGRS-2021	76.35	77.53	96.96	16.78
DNA-Net [17]	TIP-2022	84.00	84.23	97.17	4.090
ISNet [43]	CVPR-2022	74.72	74.69	93	21
RDIAN [29]	TGRS-2023	73.04	74.59	95.65	22.11
APGCNet [44]	TAES-2023	85.48	86.60	98.04	7.124
UIU-Net [37]	TIP-2023	72.44	69.47	97.83	21.51
MSHNe [21]	CVPR-2024	80.55	-	97.99	11.77
STNet(Ours)	-	87.25	87.23	98.51	5.92

of 69.45%, nIoU of 65.51%, Pd of 96.77%, and FA of 13.05×10^{-6} , and CSENet with an IoU of 66.70%, nIoU of 65.87%, Pd of 98.16%, and FA of 12.08×10^{-6} .

Table 2 presents the performance comparison of various state-of-the-art deep learning-based methods for SIRST detection on the NUDT-SIRST dataset, evaluated using pixel-level and object-level metrics including IoU, nIoU, Pd, and FA. Our proposed method STNet achieves the highest performance with an IoU of 87.25%, nIoU of 87.23%, and Pd of 98.51% coupled with a low FA of 5.92×10^{-6} . In comparison, other notable methods such as APGCNet reported an IoU of 85.48%, nIoU of 86.60%, Pd of 98.04%, and FA of 7.12×10^{-6} , and DNA-Net achieved an IoU of 84%, nIoU of 84.23%, Pd of 97.17%, and FA of 4.09×10^{-6} .

The proposed STNet method outperforms other state-of-the-art deep learning-based methods for SIRST detection on both the IRSTD-1K and NUDT-SIRST datasets. It achieves the highest scores in key metrics such as IoU, nIoU, and Pd and the lowest false alarm rates. This performance highlights STNet’s effectiveness in detecting and segmenting small targets in cluttered backgrounds.

Table 3. A comparison of different components of STNet on the IRSTD-1K dataset is presented. The results are evaluated in terms of IoU, nIoU, Pd, and FA, with the best results highlighted in bold.

Component	IOU(%)	nIOU(%)	Pd(%)	Fa(10^{-6})
STNet w/o MLFF	68.23	65.55	94.27	5.58
STNet w/o FFB	71.06	67.19	94.61	3.23
STNet	72.04	68.95	95.29	1.92



Fig. 4. Illustration of qualitative results obtained using our proposed STNet.

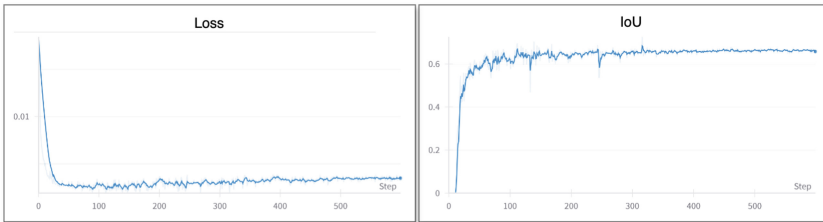


Fig. 5. STNet training loss and IoU with respect to the training epochs.

5.1 Abalation Study

Qualitative Results. Qualitative results of STNet on the IRSTD-1K dataset are shown in Fig. 4. The visualization demonstrates that STNet is able to accurately detect small targets in infrared images, with the better fusion of low-level detail features with high-level semantic features, along with small target preservation.

Effectiveness of Components. Experiments were conducted to analyze the effectiveness of the major components of STNet, specifically the MLFF and FFB modules. The results, as reported in Table 3, demonstrate that the fusion of multi-layer features (MLFF) significantly improved the IoU, as without the

MLFF block, the IoU was reduced from 72.04% to 68.23%. Furthermore, the removal of the FFB from STNet resulted in an IoU being reduced from 72.04% to 71.06%, underscoring its role in enhancing the model's ability to learn both local and global contexts more effectively.

Training and Inference Time. The time required for training and evaluation of our STNet model was analyzed. Training the STNet model on theIRSTD-1K dataset took 43 s per epoch, while test time was 0.045 s per sample. Additionally, we visualized the convergence of the training loss and the improvement in IoU throughout the training process, as depicted in Fig. 5. This highlights the efficiency of our model in both the training and evaluation phases, demonstrating its rapid processing capabilities and effective learning progression.

6 Conclusion and Future Scope

In this work, we proposed STNet, a novel SIRST detection approach designed to address the inherent challenges of detecting small targets in IR images. Our approach leverages U2-Net architecture and integrates the MLFF module and the FFB block in this architecture. The MLFF module enhances feature integration across multiple layers, effectively combining detailed low-level features with high-level semantic information. Meanwhile, the FFB extends the model's capability by extracting features in the frequency domain, thereby preserving crucial small target details even in deep network layers. Extensive experiments on theIRSTD-1K and NUDT-SIRST datasets demonstrate STNet's superiority over other state-of-the-art methods on both public datasets. Notably, STNet is able to accurately identify and segment small targets amidst cluttered backgrounds. Moving forward, the effectiveness of STNet highlights the potential of integrating frequency-domain feature extraction and multi-layer feature fusion in deep learning models for infrared small target detection. Future research should explore other approaches to accurately fuse multi-layer features for better detection. Furthermore, more work is needed to preserve small targets in the deeper layers of CNN-based methods to overcome the vanishing small target problem. These efforts could expand the applicability of SIRST methods across diverse infrared imaging tasks.

References

1. Aghaziyarati, S., Moradi, S., Talebi, H.: Small infrared target detection using absolute average difference weighted by cumulative directional derivatives. *Infrared Phys. Technol.* **101**, 78–87 (2019)
2. Bai, X., Zhou, F., Jin, T.: Enhancement of dim small target through modified top-hat transformation under the condition of heavy clutter. *Signal Process.* **90**(5), 1643–1654 (2010)
3. Bai, X., Zhou, F., Xie, Y.: New class of top-hat transformation to enhance infrared small targets. *J. Electron. Imaging* **17**(3), 030501–030501 (2008)

4. Bao, C., et al.: Improved dense nested attention network based on transformer for infrared small target detection. arXiv preprint [arXiv:2311.08747](https://arxiv.org/abs/2311.08747) (2023)
5. Chen, C.P., Li, H., Wei, Y., Xia, T., Tang, Y.Y.: A local contrast method for small infrared target detection. *IEEE Trans. Geosci. Remote Sens.* **52**(1), 574–581 (2013)
6. Dai, Y., Wu, Y., Zhou, F., Barnard, K.: Asymmetric contextual modulation for infrared small target detection. In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 950–959 (2021)
7. Dai, Y., Wu, Y., Zhou, F., Barnard, K.: Attentional local contrast networks for infrared small target detection. *IEEE Trans. Geosci. Remote Sens.* **59**(11), 9813–9824 (2021)
8. Deng, H., Sun, X., Liu, M., Ye, C., Zhou, X.: Infrared small-target detection using multiscale gray difference weighted image entropy. *IEEE Trans. Aerosp. Electron. Syst.* **52**(1), 60–72 (2016)
9. Deng, H., Sun, X., Liu, M., Ye, C., Zhou, X.: Small infrared target detection based on weighted local difference measure. *IEEE Trans. Geosci. Remote Sens.* **54**(7), 4204–4214 (2016)
10. Deshpande, S.D., Er, M.H., Venkateswarlu, R., Chan, P.: Max-mean and max-median filters for detection of small targets. In: *Signal and Data Processing of Small Targets 1999*, vol. 3809, pp. 74–83. SPIE (1999)
11. Fan, L., et al.: Diffusion-based continuous feature representation for infrared small-dim target detection. *IEEE Trans. Geosci. Remote Sens.* (2024)
12. Gonzalez, R.C., Woods, R.E., Eddins, S.L.: *Digital Image Processing Using MATLAB*. Pearson Education India (2004)
13. Han, J., Liang, K., Zhou, B., Zhu, X., Zhao, J., Zhao, L.: Infrared small target detection utilizing the multiscale relative local contrast measure. *IEEE Geosci. Remote Sens. Lett.* **15**(4), 612–616 (2018)
14. Han, J., Ma, Y., Zhou, B., Fan, F., Liang, K., Fang, Y.: A robust infrared small target detection algorithm based on human visual system. *IEEE Geosci. Remote Sens. Lett.* **11**(12), 2168–2172 (2014)
15. Han, J., Moradi, S., Faramarzi, I., Liu, C., Zhang, H., Zhao, Q.: A local contrast method for infrared small-target detection utilizing a tri-layer window. *IEEE Geosci. Remote Sens. Lett.* **17**(10), 1822–1826 (2019)
16. Li, B., et al.: Monte Carlo linear clustering with single-point supervision is enough for infrared small target detection. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1009–1019 (2023)
17. Li, B., et al.: Dense nested attention network for infrared small target detection. *IEEE Trans. Image Process.* **32**, 1745–1758 (2022)
18. Li, H., Yang, J., Wang, R., Xu, Y.: Ilnet: low-level matters for salient infrared small target detection. arXiv preprint [arXiv:2309.13646](https://arxiv.org/abs/2309.13646) (2023)
19. Lin, F., Bao, K., Li, Y., Zeng, D., Ge, S.: Learning contrast-enhanced shape-biased representations for infrared small target detection. *IEEE Trans. Image Process.* (2024)
20. Lin, F., Ge, S., Bao, K., Yan, C., Zeng, D.: Learning shape-biased representations for infrared small target detection. *IEEE Trans. Multimedia* (2023)
21. Liu, Q., Liu, R., Zheng, B., Wang, H., Fu, Y.: Infrared small target detection with scale and location sensitivity. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17490–17499 (2024)
22. Moradi, S., Moallem, P., Sabahi, M.F.: Fast and robust small infrared target detection using absolute directional mean difference algorithm. *Signal Process.* **177**, 107727 (2020)

23. Mou, X., Lei, S., Zhou, X.: YOLO-FR: a YOLOV5 infrared small target detection algorithm based on feature reassembly sampling method. *Sensors* **23**, 2710 (2023)
24. Pan, S., Zhang, S., Zhao, M., An, B.: Infrared small target detection based on double-layer local contrast measure. *Acta Photonica Sin.* **49**, 0110003 (2020)
25. Peng, J., Zhao, H., Hu, Z., Zhao, K., Wang, Z.: Dynamic background reconstruction via transformer for infrared small target detection. arXiv preprint [arXiv:2301.04497](https://arxiv.org/abs/2301.04497) (2023)
26. Qin, X., Zhang, Z., Huang, C., Dehghan, M., Zaiane, O.R., Jagersand, M.: U2-net: going deeper with nested u-structure for salient object detection. *Pattern Recogn.* **106**, 107404 (2020)
27. Shang, K., Sun, X., Tian, J., Li, Y., Ma, J.: Infrared small target detection via line-based reconstruction and entropy-induced suppression. *Infrared Phys. Technol.* **76**, 75–81 (2016)
28. Shi, Y., Wei, Y., Yao, H., Pan, D., Xiao, G.: High-boost-based multiscale local contrast measure for infrared small target detection. *IEEE Geosci. Remote Sens. Lett.* **15**(1), 33–37 (2017)
29. Sun, H., Bai, J., Yang, F., Bai, X.: Receptive-field and direction induced attention network for infrared dim small target detection with a large-scale dataset irdst. *IEEE Trans. Geosci. Remote Sens.* **61**, 1–13 (2023)
30. Wang, A., Li, W., Wu, X., Huang, Z., Tao, R.: Mpanet: multi-patch attention for infrared small target object detection. In: *IGARSS 2022-2022 IEEE International Geoscience and Remote Sensing Symposium*, pp. 3095–3098. IEEE (2022)
31. Wang, C., Wang, H., Pan, P.: Local contrast and global contextual information make infrared small object salient again. arXiv preprint [arXiv:2301.12093](https://arxiv.org/abs/2301.12093) (2023)
32. Wang, H., Zhou, L., Wang, L.: Miss detection vs. false alarm: adversarial learning for small object segmentation in infrared images. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8509–8518 (2019)
33. Wei, Y., You, X., Li, H.: Multiscale patch-based contrast measure for small infrared target detection. *Pattern Recogn.* **58**, 216–226 (2016)
34. Wu, F., Zhang, T., Li, L., Huang, Y., Peng, Z.: Rpanet: deep unfolding RPCA based infrared small target detection. In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 4809–4818 (2024)
35. Wu, L., Ma, Y., Fan, F., Wu, M., Huang, J.: A double-neighborhood gradient method for infrared small target detection. *IEEE Geosci. Remote Sens. Lett.* **18**(8), 1476–1480 (2020)
36. Wu, S., Xiao, C., Wang, L., Wang, Y., Yang, J., An, W.: Repisd-net: learning efficient infrared small-target detection network via structural re-parameterization. *IEEE Trans. Geosci. Remote Sens.* (2023)
37. Wu, X., Hong, D., Chanussot, J.: UIU-net: U-net in U-net for infrared small object detection. *IEEE Trans. Image Process.* **32**, 364–376 (2022)
38. Xia, C., Li, X., Zhao, L., Shu, R.: Infrared small target detection based on multiscale local contrast measure using local energy factor. *IEEE Geosci. Remote Sens. Lett.* **17**(1), 157–161 (2019)
39. Xie, X., Zhou, P., Li, H., Lin, Z., Yan, S.: Adan: adaptive nesterov momentum algorithm for faster optimizing deep models. arXiv preprint [arXiv:2208.06677](https://arxiv.org/abs/2208.06677) (2022)
40. Zhang, F., Lin, S., Xiao, X., Wang, Y., Zhao, Y.: Global attention network with multiscale feature fusion for infrared small target detection. *Opt. Laser Technol.* **168**, 110012 (2024)
41. Zhang, H., Zhang, L., Yuan, D., Chen, H.: Infrared small target detection based on local intensity and gradient properties. *Infrared Phys. Technol.* **89**, 88–96 (2018)

42. Zhang, M., Yang, H., Guo, J., Li, Y., Gao, X., Zhang, J.: Irprunedet: efficient infrared small target detection via wavelet structure-regularized soft channel pruning. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 38, pp. 7224–7232 (2024)
43. Zhang, M., Zhang, R., Yang, Y., Bai, H., Zhang, J., Guo, J.: Isnet: shape matters for infrared small target detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 877–886 (2022)
44. Zhang, T., Li, L., Cao, S., Pu, T., Peng, Z.: Attention-guided pyramid context networks for detecting infrared small target under complex background. *IEEE Trans. Aerosp. Electron. Syst.* (2023)



FF-Yolo: A Feature-Fusion Yolo Model for Small Scale FODs Detection in Airport Runways

Soumen Biswas^(✉) and Ananth Ganesh

R&D Center, Hitachi India Pvt. Ltd., Bangalore 560055, Karnataka, India
soumenbiswas@outlook.com

Abstract. The foreign object debris (FODs) on airport runways often disrupt operations while taking off and landing flights, leading to accidents. Small-scale FODs cannot manually rule out on time, which threatens aviation safety. This paper presents an intelligent computer vision system for small-scale FOD detection. This work proposes a Feature-Fusion Yolo (FF-Yolo) to accelerate the Yolov5 model to detect FODs in airports. A lightweight convolution-based attention module (CBAM) is considered in the backbone of the proposed architecture to improve the model efficiency by focusing on the target features. In addition, to reduce the overfitting problem, a C3TR module is included in the FF-Yolo model's backbone and neck, which captures both spatial and temporal features. Further, GhostConv is used in neck network, which helps in increasing the overall accuracy. Finally, an improved detection head is introduced in FF-Yolo to find out the size of the small-scale FODs along with their pixel location, which helps the aviation personnel to measure the severity and take prompt action. The experiments are performed on a FOD-A dataset with a runway and taxiway background, including different light and weather conditions. The proposed model achieved 98.61% mAP@0.5 and 83.21% mAP@0.95, which are higher than other state-of-the-art (SOTA) models. An overall improvement of 7.33%, 6.43%, 5.79%, and 4.93% of mAP@0.5 and 6.26%, 16.51%, and 5.1% of mAP@0.95 are noted compared to the Yolov5 baseline models. The FF-Yolo model achieves a detection speed of 33.31 frame/s, much higher than the other SOTA models. Consequently, the ablation study verifies the robustness of the FF-Yolo model for small-scale FOD detection.

Keywords: Attention Module · Foreign Object Debris · FODs Detection · Improved Yolov5 model · FF-Yolo model · FODs in airport runways

1 Introduction

1.1 Literature Study and Research Gaps

To ensure runway security and safety, the detection of FOD is an important task in aviation. Constant supervision of airport runways is essential for smooth operations in the airport. Missed identification of FOD may cause missed identification of FOD and lead to accidents. In recent

times, many airports have performed the inspection manually with the help of human labor which is crucial and may lead to missed identification of FOD [1]. Generally, FOD refers to various artifacts such as parts of equipment, luggage straps, small tire parts, aircraft components, repairing components, etc. are present in the airport runways. The presence of FOD sometimes delays the operations of aircraft and increases repair costs [2]. Figure 1 shows an example of FOD in airport runways.

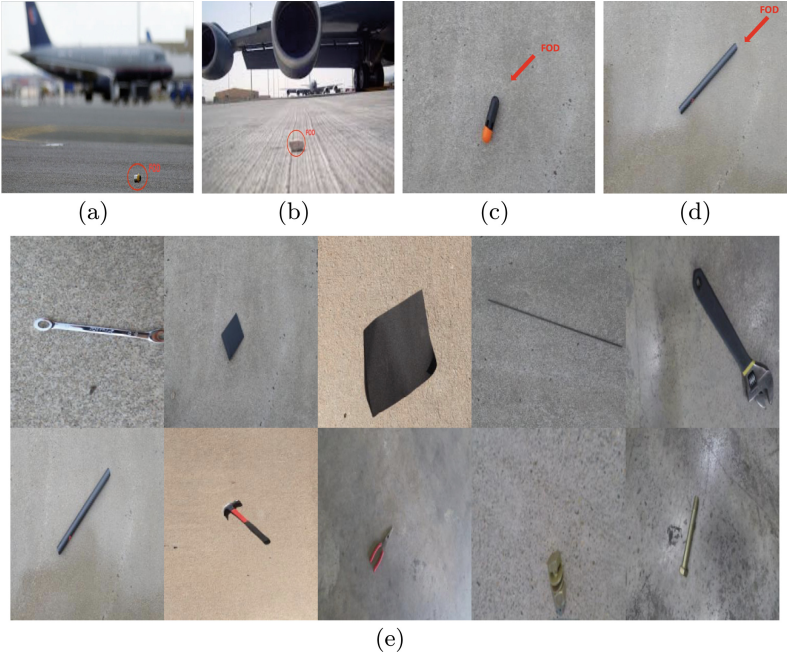


Fig. 1. Images of small scale FODs in airport runways (a)–(d) Sample FOD from airport runways [1], (e) Sample FOD from FOD-A [21] Datasets.

The airport authorities are trying to mitigate the problem of FOD identification in many ways as it leads to high financial losses [3]. Hence, it is essential to find an innovative approach to detect FOD in runways. Recent advancements in technology have come up with various solutions from time to time to mitigate different problems in various industries. The computer vision technique leads among all because of its robustness and efficiency in the detection of objects. In the literature, researchers introduced different object detection models in various fields other than aviation [4]. Few are findings in the literature where researchers have developed FOD detection systems but the use of hardwires leads those systems to high cost [5]. Also, few of them are based on cameras to capture the runway areas and later human efforts are required to identify those FODs [6]. In the last decade, few researchers have reported some effective methods to detect FOD [7–10]. The methods or systems are based on LiDAR

[7], mm-wave FMCW radar technology [8], which produces good results in different environmental setups. Authors in [11] introduced a region-based CNN for FOD detection. In this research, authors introduced a region proposal network and spatial transformation network to detect FOD using optical camera sensors. There are also few studies in literature based on Faster RCNN [12] and Mask RCNN [13] to extract the FOD region and classify the types of FOD. Authors in [14] demonstrate the FOD detection using region-based CNN. The authors described a transfer learning and deep neural network-based methodology to improve the overall detection results. The key point detection using CNN is one of the challenges in FOD detection which is improved by authors in [15]. In the literature, many methodologies exist to detect airport FODs using sensor data, unmanned aerial vehicle (UAV) images, LiDAR data, radar-based data, etc. However, all this equipment needs higher computational costs and a large-scale setup to implement in real time [7–10]. The data collection itself is challenging for FODs. The intelligent computer vision system is an emerging technique that mitigates various problems and computational costs. Collecting data using UAVs, sensors, and mm-wave radar technology involves high computational costs and requires continuous attention to handle the failures of detection systems. A research gap exists in the artificial intelligence (AI) domain for FOD detection and identification. The major limitation of this research may include data collection from the airport authorities, which leads to the problem needing to be solved. Timely detection of FODs is essential to handle unavoidable situations. Intelligent computer vision systems come up with various advantages that may resolve the difficulties of identifying the FODs in airport runways. Object detection models can identify and detect various objects in real-time. Also, it helps in reducing the computational complexity by retaining higher accuracy results. In literature, object detector in YOLO network improves the speed by lowering the regression loss and producing higher accuracy [16,17]. There are also few studies on FOD detection other than airport runways using YOLO architecture where authors improved the architecture to achieve good results [18,20]. The research is still open to detect FOD in airport runways using computer vision and artificial intelligence (AI) due to the limitation of methodology in this area.

1.2 Contributions

Motivated by the advantages of YOLO architecture, the present paper introduced a FF-Yolo to detect the FOD in airport runways. It is essential to mitigate the missed detection problem in aviation by detecting the FOD. Also, the size of the FOD is often small and difficult to detect. Thus, we introduced an improved Yolov5 model where we changed the backbone and neck of the architecture. Also, the detection head added the novelty of calculating the approximate size of FOD. In backbone, fusion of C3TR and lightweight CBAM layers are used to get the refined features from the backbone. Subsequently, the Ghost Convolution is used in the neck replacing the traditional convolution layer that helps to capture the spatial and temporal features to improve the efficiency of the model. This strategy not only improves the computational time but also improves the accuracy of

the model. The FF-Yolo achieved $\text{map}@0.5$ of 98.61% and $\text{mAP}@0.95$ of 83.21% on the FOD-A [21] dataset.

The main contributions are summarized as follows:

1. The C3TR layer is used as a bridge between the convolution layer and the CBAM layer that helps in enhancing the feature on combined image blocks. The size of the input images is initially divided into different image blocks and later combined results for feature extraction. Further, C3TR is also added in the neck after concat layer to fusion the features.
2. The lightweight CBAM layer is used before SPPF layer to extract more essential feature information from channel and Spatial attention. The fusion of these features helps in enhancing the feature quality.
3. In neck of FF-Yolo, Convolution layer is replaced by Ghost Convolution layer after upsampling feature layers which help in regularizing features to improve the detection accuracy.
4. The novelty of FF-Yolo is the arrangement in uses of CBAM, C3TR and ghost convolution in the architecture along with the feature enhancement methodology.
5. Finally, we introduced a detection head, where we added a size estimation function based on the object bounding box to get an approximate size of FOD.

The rest of the paper is arranged as follows: Sect. 2 describes the architecture of FF-Yolo. The comparative analysis, ablation study and results are discussed in Sect. 3. Finally, Sect. 4 concludes the research work and future scope followed by references.

2 Improvement Method: Proposed FF-Yolo Model

2.1 Overall Framework

The architecture of the FF-Yolo is shown in Fig. 2. The traditional Yolov5 consists of backbone, neck, and prediction heads. The backbone of the network extracts the features whereas the neck performs the feature fusion followed by three prediction heads. In the proposed architecture, we consider three modifications in backbone and head and they are: 1) we added C3TR and CBAM layer in the backbone. The C3TR layer is a bridge between the Convolution layer and CBAM and the extracted features are fed into SPPF layer which is the last layer in backbone, 2) In addition, C3TR layer is also added in neck of FF-Yolo after concat layer to fusion the features that helps in reducing overfitting problem, 3) After upsampling layers, convolution layers are replaced by ghost convolution layers to regularize the feature fusion to improve the detection accuracy, 4) Introduced a detection head, where we include size estimation of the object from the bounding box information of the detected objects.

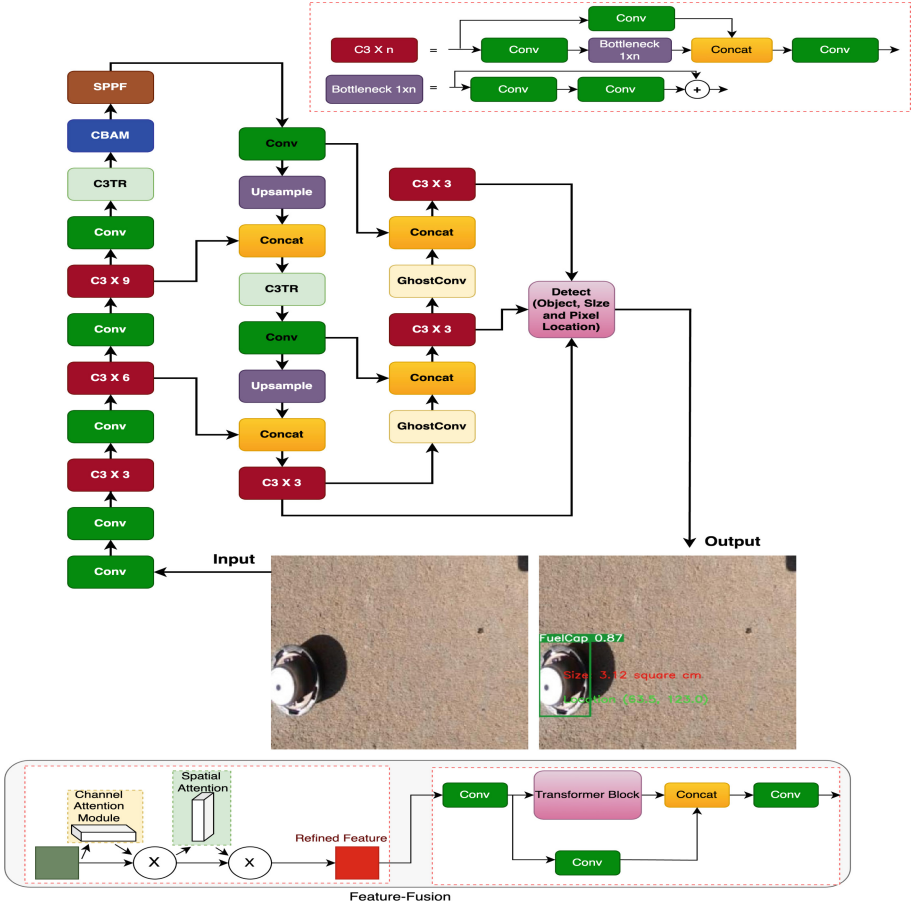


Fig. 2. Architecture of FF-Yolo for FOD Detection

2.2 Convolution Block Attention Module (CBAM) in FF-Yolo

The CBAM module is based on an attention mechanism that can be integrated into CNN architecture. The CBAM consists of two Attention modules, namely Channel Attention and Spatial Attention. The purpose of using CBAM is to get the features in backbone instead of using them in neck to generate the feature fusion of pyramids. Also, it is very lightweight which makes the model smaller in terms of size as the number of parameters required is very less. In general, during training existence of a large number of parameters will affect the model performance as it is difficult to train a large number of parameters and also require high computational time. So using the CBAM in the backbone reduces the size of feature maps. Further, Fig. 2 includes the structure of CBAM where Global

Max-pooling and Average pooling are performed in channel attention module to extract the feature maps on different channels. The elementwise summation and sigmoid activation execute to redesign the feature maps. Similarly, on separate feature maps global max-pooling and average pooling operations are performed on pixel values and then concatenated with these features followed by a two-dimensional convolution layer and sigmoid activation function. The following formula can express the whole process.

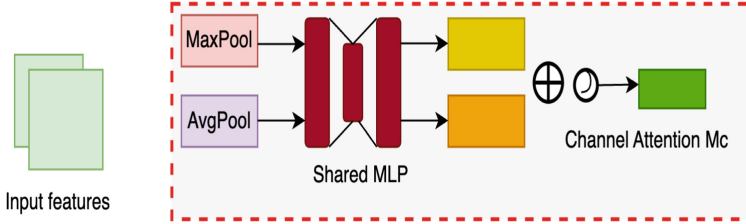


Fig. 3. Channel Attention Module

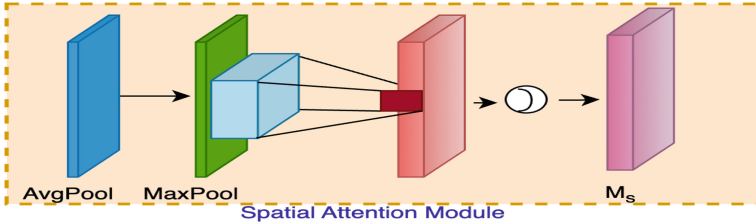


Fig. 4. Spatial Attention Module

$$I'_F = M_c(I_{F_i}) \times I_F \quad (1)$$

$$Y_F = M_s(I'_{F_i}) \times I_F \quad (2)$$

where \times denote element-wise multiplication. I'_F is channel module output whereas Y_F is the spatial attention module output.

Figure 3 depicts that the channel attention module is used to find more intersection areas [22]. In the channel attention module, two feature descriptor is used to find the features by performing global average pooling (I_c^{avg}) and max-pooling operations (I_c^{max}). These feature maps are fed to multi-layer perceptron (MLP) to produce channel attention maps (M_c).

$$\begin{aligned} M_c &= \sigma(MLP(AvgPool(I_F)) + MLP(MaxPool(I_F))) \\ &= \sigma(W''(W'(I_c^{avg})) + W''(W'(I_c^{max}))) \end{aligned} \quad (3)$$

where, σ denotes a sigmoid function. W' and W'' are the weights shared for both the inputs.

In Fig. 4, the Spatial attention layer [22] is used to find the spatial information of defects by performing average pooling or max-pooling operations to generate the feature maps for average pool and max-pool layers. The convolution operation of both features produces spatial attention feature maps.

2.3 C3TR and Ghost Convolution in FF-Yolo

The C3TR is an improved version of C3 layer which is used in the backbone of the FF-Yolo for feature extraction. The advantage of this layer is to improve the efficiency of the network by reducing computational cost as it combines the benefits of CSPNet and C3 layer [23]. Hence, the motivation of using this layer is to facilitate the low-level features to high-level representations of the feature information. Similarly, Ghost convolution is used in neck of the proposed architecture to refine the features and extract the feature maps. It ensures the lightweight convolution operation which makes the model low in size while improving the speed. The working strategy of the Ghsot convolution is to spit the input tensors into multiple ghost tensors which effectively helps in reducing the computational cost. It captures the contextual information and same time enhances the extracted features from the backbone. Also, uses of this layer in the neck maintains the performance of FF-Yolo in terms of accuracy. Figure 5 illustrate the structure of C3TR and Ghost Convolution layer.

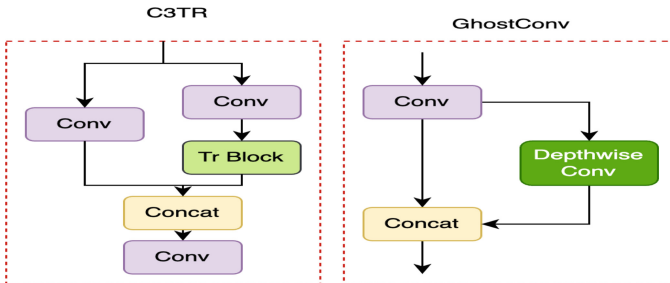


Fig. 5. Illustration of C3TR and GhostConv layers

2.4 Modified Prediction Heads of FF-Yolo

The proposed model consists of three detection heads of small, mid, and large feature maps. In each prediction head the output vectors are fused maps from backbone and neck. These output vectors represent the regression bounding box that illustrates the coordinate and size. From the regression bounding box, we extract the variable of size into a new function to get the approximate object

size. Along with this, it consists of the confidence score of each class detected by the prediction head. The final bounding box is generated by anchors to visually represent the objects in a frame or image.

2.5 Loss Function in FF-Yolo

The loss function of the FF-Yolo is expressed as follows:

$$Loss = xLoss_{objectness} + yLoss_{Bounding_box} + zLoss_{Classification_prob} \quad (4)$$

where, x, y and z are denoted as objectness loss weight, bounding box loss weight and classification probability loss weight. In our experiments, we set the values as 1, 0.05 and 0.5.

In proposed model binary cross entropy loss is used for both classification probability and objectness. Similarly, for bounding box regression we used CIOU loss [24].

3 Experimental Results

We perform the experiments on FOD-A dataset using FF-Yolo model. The experimental results demonstrate the robustness of the proposed model for FOD detection in terms of accuracy and computational time.

3.1 Experiment Settings

The experiments are performed on 13th Gen Intel(R) Core(TM) i9-13900K CPU with 24 GB GPU Memory NVIDIA GeForce RTX 4090. The proposed model is implemented on pytorch using CUDA version 12.2.

3.2 Dataset

For the experimental purpose, we consider FOD-A dataset [21] to have 31 different classes of FOD objects with 30,000 instances. The instances include different lightening and weather category condition images. A total of 15,000 images are considered from the dataset with 13950 instances for this research experiment. We split the dataset into 70:20:10 for train, validation, and test sets respectively. Figure 6 shows the sample images from the dataset and details of annotation instances.

3.3 Hyperparameter Settings

The experiments in this paper considered the values of hyperparameter as mentioned below. The training epoch is 200 epochs with a batch size of 16. In these experiments using FF-Yolo, we set early stop criteria during training and it set to 10 epochs. The initial learning rate is considered as 0.001 and the considerable input size is 640×640 .



Annotation Instances

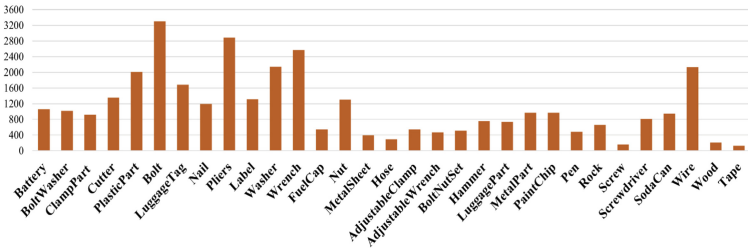


Fig. 6. Sample images from FOD-A dataset, types of FODs and Number of annotation Instances [21]

3.4 Evaluation Criteria

The performance evaluation of the FF-Yolo is measured using different parameters, namely mean average precision (mAP), precision, and recall. The mAP can be calculated by calculating the average precision (AP) value for each class and then taking an average over the number of classes [19]. The formula can be defined as:

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \quad (5)$$

Further, precision (P_R) is a calculation of finding all true positives (TP) out of all TP and false positive (FP) which can be expressed as below [25]:

$$P_R = \frac{TP}{TP + FP} \quad (6)$$

Similarly, recall (R_E) is calculated for all TP out of all TP and false negatives (FN). The R_e can be expressed as below [25]:

$$R_E = \frac{TP}{TP + FN} \quad (7)$$

The significance of calculating mAP as it incorporates the trade-off between P_R and R_E . Also mAP metric considered both TP and FN .

Further, mAP@0.5 and mAP@0.95 are calculated when the IoU is 0.5 and 0.95 for all classes.

3.5 Experimental Results

The experiments on the FOD dataset using FF-Yolo produce better results in terms of accuracy while comparing the results with other existing state-of-the-art (SOTA) models, namely Yolovs [25], Yolov5m [25], Yolov5x [25], Yolov3 [26], Yolov3-tiny [26], Yolov3-SPP [26] and Li et al. [27]. It is noted that all the experiments performed using SOTA models are on FOD dataset. The experimental results also show the robustness of the proposed model while comparing it to the baseline Yolov5s model. Table 1 illustrates a comparison of the FF-Yolo model with other SOTA models. From Table 1 it can be depicted that FF-Yolo model outperforms the baseline Yolov5s, Yolov5x, and Yolov5m by 7.33%, 6.43% and 5.79% in terms of mAP@0.5, respectively. Similarly, proposed model shows an improvement of 6.26%, 16.51%, and 5.1% in terms of mAP@0.95 while compared to Yolov5s, Yolov5x and Yolov5m, respectively. Also, there is a significant improvement of FF-Yolo in terms of mAP@0.5 and mAP@0.95 while comparing it with the Yolov3 [26] and Li et al. [27] models. Further, FF-Yolo outperforms the other existing models in terms of Precision (P_R) and Recall (R_E). Figure 7 shows the output detection results using proposed model. It can be observed that the proposed model is also efficient in calculating the size of FOD and the location of those objects in the specific frame. It is noted that for calculating the size of objects we consider the camera calibration parameter of pixel size of 0.01 cm per pixel.

Table 1. A comparative analysis of Improved YOLOv5 model compared to other existing models

Models	P_R (%)	R_E (%)	mAP@0.5(%)	mAP@0.95 (%)
Yolov5s [25]	93.57	89.56	91.28	76.95
Yolov5x [25]	94.26	90.10	92.18	66.70
Yolov5m [25]	95.04	91.31	92.28	78.12
Yolov3 [26]	90.80	89.87	90.75	62.83
Yolov3-tiny [26]	92.80	92.40	96.40	70.00
Yolov3-SPP [26]	92.0	86.91	94.00	71.40
Li et al. [27]	93.00	90.00	93.68	79.68
yolov8 [28]	94.90	94.20	97.30	79.20
FF-Yolo (Ours)	97.08	95.97	98.61	83.21

Further, Fig. 8 below shows the comparison of the mAP@0.5-0.95 values curve of FF-Yolo with other SOTA models.



Fig. 7. Detection results of FODs using FF-Yolo model.

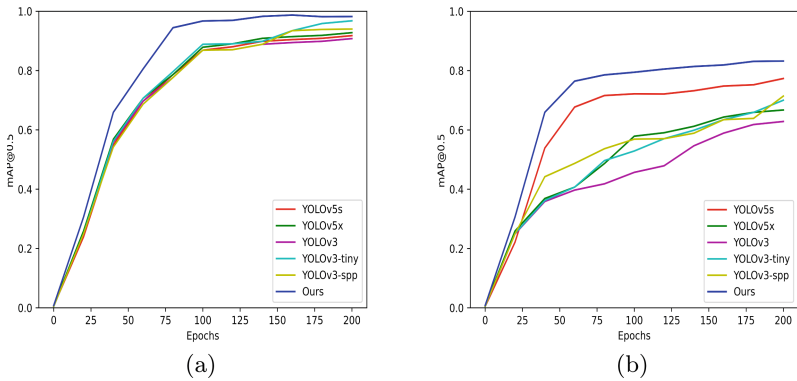


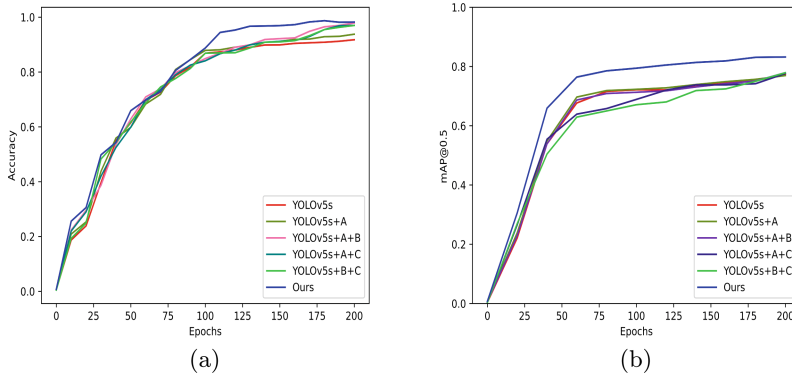
Fig. 8. Comparison Curve of mAP@0.5-0.95 values

3.6 Ablation Study

We perform the ablation study to evaluate the performance of proposed FF-Yolo model. Different experimental setups is formed to train FOD dataset. Table 2 explains the ablation study performed in this research. In first setup, we considered only Ghost Convolution layer in neck and removed C3TR and CBAM

Table 2. The Ablation study on FOD dataset

Model	Ablation Setting			mAP@0.5 (%)	mAP@0.95 (%)
	GhostConv (A)	C3TR (B)	CBAM (C)		
Yolov5s	✗	✗	✗	91.28	76.95
Yolov5s + A	✓	✗	✗	93.67	77.32
Yolov5s+A+B	✓	✓	✗	97.45	77.92
Yolov5s+A+C	✓	✗	✓	97.27	77.65
Yolov5s+B+C	✗	✓	✓	97.56	77.90
FF-Yolo (Ours)	✓	✓	✓	98.61	83.21

**Fig. 9.** Ablation study analysis for (a) mAP@0.5 and (b) mAP@0.95 [A= GhostConv, B= C3TR, and C=CBAM]

layers from backbone of the architecture shown in Fig. 2. Instead of C3TR, C3 layer was added, and performed the experiment. It is depicted from the table that it achieve 93.67% of mAP@0.5 and 77.32% of mAP@0.95 which are lower than the proposed model results. In second setup, Ghost Convolution layers and C3TR layer are considered in neck and backbone, respectively while we removed CBAM from backbone. This arrangement in the architecture produces 97.45% and 77.92% of mAP@0.5 and mAP@0.95 values, respectively. In third setup, we considered Ghost Convolution and CBAM in the architecture and replaced C3TR with C3 layer in backbone. It produces 97.27% of mAP@0.5 and 77.65% of mAP@0.95 which are lower than proposed FF-Yolo model. In final setup of ablation study, instead of Ghost Convolution layer traditional convolution layer was used in neck, and rest no changes were made in the backbone. This arrange-

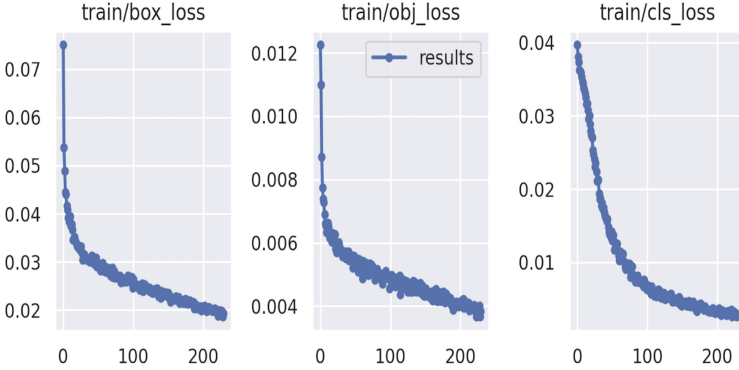


Fig. 10. Training Loss using FF-Yolo

Table 3. A comparative analysis of FPS and Runtime

Models	FPS	Runtime(in Sec.)
Yolov5s [25]	31.24	28.08
Yolov5x [25]	25.21	34.71
Yolov5m [25]	29.93	29.23
Yolov3 [26]	26.01	33.63
Yolov3-tiny [26]	30.92	28.30
Yolov3-SPP [26]	25.77	33.95
Li et al. [27]	24.69	35.44
FF-Yolo	33.31	26.27

ment in architecture produces 97.56% of mAP@0.5 and 77.90% of mAP@0.95 which are also less compared to FF-Yolo model. Hence, it is depicted that FF-Yolo is superior in detecting FOD as it produces higher mAP@0.5-0.95 values. Further, Fig. 9 and Fig. 10 illustrate mAP@0.5 values curve of ablation study and loss curve of FF-Yolo, respectively. It is depicted that the training loss is minimal and it convergence using the FF-Yolo model.

3.7 Inference Time Analysis

Table 3 depicts the comparative study of runtime and FPS using FF-Yolo model with other SOTA models. The experiments are performed on a test video of 875 frames of FOD. It depicts that the proposed model meets the criteria of detecting FOD in a real-world scenario by providing FPS of 33.31 and a total execution time or runtime of 26.27 s. The FPS value using the FF-Yolo is quite high compared to the other SOTA models which is evidence of fast detection of FOD.

4 Conclusion and Future Scope

The present paper introduces an improved Yolov5, namely the FF-Yolo model for FOD detection in airport runways. Detecting FOD is essential to ensure aviation safety, thereby reducing unavoidable circumstances. The improved FF-Yolo model not only detected FODs with higher accuracy but was also capable of identifying the approximate size of FODs and their pixel locations. The FF-Yolo is considered a lightweight attention module, i.e., CBAM, that helps decrease the computational time and enhances the feature fusion quality by integrating channel and spatial information. It also reduces the number of parameters, which helps in reducing the training cost. Additionally, the use of the C3TR layer enhances the feature maps, and the adaptation of ghost convolution layer regularizes the fusion features to improve detection accuracy. The proposed FF-Yolo model achieved higher mAP@0.5 of 98.61% and mAP@0.95 of 83.21%, which are higher than the other SOTA models. Also, results using the proposed model show the effectiveness while detecting the FODs in less time compared to SOTA models, which is 33.31 frame/s. Further, the ablation study also proves the robustness of the FF-Yolo model. In the future, we intend to collect FOD data from a certain distance of airport runways and evaluate the performance using the proposed model. Further, including other types of FOD in the dataset is also possible to make the model more robust.

References

1. European Union Aviation Safety Agency: Certification Specifications and Guidance Material for Aerodrome Design (CS-ADR-DSN) (2022). <https://www.easa.europa.eu/en/downloads/136283/en>. Accessed 8 Feb 2024
2. AC 150/5210-24 - Airport Foreign Object Debris (FOD) Management. Federal Aviation Administration (2023). https://www.faa.gov/documentLibrary/media/Advisory_Circular/AC_150_5210-24.pdf
3. Suder, J., Maciejewski, P., Podbucki, K., Marciniak, T., Dabrowski, A.: Measuring platform for quality testing of airport lamps. *Pomiary Autom. Robot.* **23**(2), 5–13 (2019). https://doi.org/10.14313/PAR_232/5
4. Elrayes, A., Ali, M.H., Zakaria, A., Ismail, M.H.: Smart airport foreign object debris detection rover using LiDAR technology. *Internet Things* **5**, 1–11 (2019). <https://doi.org/10.1016/j.iot.2018.11.001>
5. Papadopoulos, E., Gonzalez, F.: UAV and AI application for runway foreign object debris (FOD) detection. In: 2021 IEEE Aerospace Conference (50100), Big Sky, MT, USA, pp. 1–8 (2021). <https://doi.org/10.1109/AERO50100.2021.9438489>
6. Jing, Y., Zheng, H., Lin, C., Zheng, W., Dong, K., Li, X.: Foreign object debris detection for optical imaging sensors based on random forest. *Sensors* **22**(7), 2463 (2022). <https://doi.org/10.3390/s22072463>
7. Li, Y., Xiao, G.: A new FOD recognition algorithm based on multi-source information fusion and experiment analysis. *Proc. SPIE* (2011). <https://doi.org/10.1117/12.900576>

8. Futatsumori, S., Morioka, K., Kohmura, A., Okada, K., Yonemoto, N.: Detection characteristic evaluations of optically-connected wideband 96 GHz millimeter-wave radar for airport surface foreign object debris detection. In: Proceedings of the 41st International Conference on Infrared, Millimeter, and Terahertz waves, Copenhagen, Denmark, 25–30 September 2016, pp. 1–2 (2016)
9. Zeitler, A., Lanteri, J., Pichot, C., Migliaccio, C., Feil, P., Menzel, W.: Folded reflectarrays with shaped beam pattern for foreign object debris detection on runways. *IEEE Trans. Antennas Propag.* **58**, 3065–3068 (2010)
10. Mund, J., Zouhar, A., Meyer, L., Fricke, H., Rother, C.: Performance evaluation of LiDAR point clouds towards automated FOD detection on airport aprons. In: Proceedings of the 5th International Conference on Application and Theory of Automation in Command and Control Systems, Toulouse, France, 30 September–2 October 2015, pp. 85–94 (2015)
11. Cao, X., et al.: Region based CNN for foreign object debris detection on airfield pavement. *Sensors* **18**(3), 737 (2018)
12. Girshick, R.B.: Fast R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision, Los Alamitos, CA, USA, 7–13 December 2015, pp. 1440–1448 (2015)
13. He, K., Gkioxari, G., Dollár, P., Girshick, R.B.: Mask R-CNN. *arXiv* (2017). [arXiv:1703.06870](https://arxiv.org/abs/1703.06870)
14. Xu, H., Han, Z., Feng, S., Zhou, H., Fang, Y.: Foreign object debris material recognition based on convolutional neural networks. *EURASIP J. Image Video Process.* **2018**(1), 1–10 (2018). <https://doi.org/10.1186/s13640-018-0261-2>
15. He, K., Gkioxari, G., Dollár, P., Girshick, R.B.: Mask R-CNN. *arXiv* (2017). [arXiv:1703.06870](https://arxiv.org/abs/1703.06870)
16. Redmon, J., Farhadi, A.: YOLO9000: Better, Faster, Stronger. *arXiv* (2016). [arXiv:1612.08242](https://arxiv.org/abs/1612.08242)
17. Redmon, J., Divvala, S.K., Girshick, R.B., Farhadi, A.: You only look once: unified, real-time object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 27–30 June 2016, pp. 779–788 (2016)
18. Dai, Y., Liu, W., Wang, H., Xie, W., Long, K.: Yolo-former: marrying yolo and transformer for foreign object detection. *IEEE Trans. Instrum. Meas.* **71**, 1–14 (2022)
19. Ren, M., Wan, W., Yu, Z., Zhao, Y.: Bidirectional YOLO: improved YOLO for foreign object debris detection on airport runways. *J. Electron. Imaging* **31**(6), 063047–063047 (2022)
20. Li, M., Ding, L.: DF-YOLO: highly accurate transmission line foreign object detection algorithm. *IEEE Access* (2023)
21. Munyer, T., Huang, P.-C., Huang, C., Zhong, X.: FOD-A: A Dataset for Foreign Object Debris in Airports. [arXiv:2110.03072](https://arxiv.org/abs/2110.03072) (2021)
22. Cheng, X., Yu, J.: RetinaNet with difference channel attention and adaptively spatial feature fusion for steel surface defect detection. *IEEE Trans. Instrum. Meas.* **70**, 1–11 (2020)
23. Tan, X., He, X.: Improved Asian food object detection algorithm based on YOLOv5. In: E3S Web of Conferences, vol. 360. EDP Sciences (2022)
24. Zheng, Z., et al.: Enhancing geometric factors in model learning and inference for object detection and instance segmentation (2021)

25. Chutia, G., Biswas, S., Palanivel, D.A., Gopalakrishnan, S.: LW-DCNN: a lightweight CNN model for human activity classification using radar micro-doppler signatures. In: IEEE International Symposium on Smart Electronic Systems (iSES), pp. 73–77. IEEE (2022)
26. Horvat, M., Gledec, G.: A comparative study of YOLOv5 models performance for image localization and classification. In: Central European Conference on Information and Intelligent Systems, pp. 349–356. Faculty of Organization and Informatics Varazdin (2022)
27. Li, P., Li, H.: Research on FOD detection for airport runway based on YOLOv3. In: 2020 39th Chinese Control Conference (CCC), Shenyang, China, pp. 7096–7099 (2020). <https://doi.org/10.23919/CCC50068.2020.9188724>
28. Jocher, G., Chaurasia, A., Qiu, J.: Ultralytics YOLOv8 (2023). <https://github.com/ultralytics/ultralytics>



Weakly Aligned Multi-spectral Pedestrian Detection via Cross-Modality Differential Enhancement and Multi-scale Spatial Alignment

Zhenzhou Shao¹, Yongxin Chen¹, Yibo Zou³, Jie Zhang²,
and Yong Guan¹

¹ The College of Information Engineering, Beijing Key Laboratory of Light Industrial Robot and Safety Verification, Capital Normal University, Beijing 100048, China

{zshao, 2201002016, guanyong}@cnu.edu.cn

² The School of Optoelectronic Science and Engineering, Soochow University, Suzhou, China

jzhang@mail.buct.edu.cn

³ The College of Information Science and Technology, University of Chemical Technology, Beijing, China

yibo.zou@suda.edu.cn

Abstract. Multi-spectral pedestrian detection has attracted extensive attention in recent years. In particular, the combination of RGB and thermal infrared images allows the around-the-clock applications, even in the poor illumination conditions. Considering the fact that RGB and thermal infrared (RGB-T) image pairs are not well aligned, it leads to the inaccuracy of pedestrian detection. To this end, this paper proposes a Multi-scale Alignment and Differential Enhancement Network (MADENet) for multi-spectral pedestrian detection, consisting of Cross-Modality Differential Enhancement Module (CDEM) and Multi-scale Spatial Alignment Module (MSAM). CDEM module is embedded in the backbone to suppress the redundant features and extract complementary information between modalities, and MSAM module is designed to align the RGB-T features by the transformation of thermal features using features of RGB image as the reference. The proposed network is evaluated on the public KAIST dataset across different scenarios. Experimental results demonstrate that the proposed method outperforms the state-of-the-art methods. Miss rate using all test set can reach 8.01.

Keywords: Pedestrian Detection · Multi-spectral · Image Processing

1 Introduction

Pedestrian detection is an active research area in the field of computer vision and widely used in several applications, including autonomous driving [1], video surveillance [2] and so forth. Although a lot of progress has been made in deep



Fig. 1. Weak alignment of RGB-T image pairs.

learning based pedestrian detection using RGB images, the performance usually suffers from the poor illumination, resulting in false or missing detection. Considering the thermal infrared imaging principle that captures the thermal radiation on the surface of objects [3], pedestrian detection based on RGB and thermal infrared (RGB-T) images pairs becomes draws more and more attentions in recent years. How to extract and fuse the multi-spectral complementary features from both modalities is the key for pedestrian detection.

Since the KAIST multi-spectral pedestrian detection dataset [4] was proposed in 2015, several RGB-T multi-spectral pedestrian detection methods have been presented [5–7]. These methods are proposed under the assumption that RGB-T image pairs are geometrically aligned. However, there exists a weak position shift problem, i.e., positions of the same pedestrian are different in both modalities, as shown in Fig. 1. This problem can be caused by the physical properties of different sensors, imperfection of alignment algorithm or external disturbance. It becomes a concern to improve the performance of multi-spectral pedestrian detection.

According to the plausible solutions provided by the traditional methods, the aforementioned problem is narrowed down to the alignment of features in the candidate region proposals, which are generated using the scheme of anchor-based object detection method. AR-CNN [8] is proposed using a random RoI jittering strategy to align the features of region proposals (bounding box), followed by the regression of bounding box for pedestrian detection. Similarly, Zhou *et al.* predicts the proposal offset of both modalities, and proposes the deformable anchors for regression [9]. However, the alignment procedure is implemented

using the limited amount of features in each region proposal in existing methods, which might mismatch the features of both modalities, leading to the failure of pedestrian detection. In addition, the redundancy of multi-spectral fusion is required to be considered to extract the complementary features.

To address above problems, this paper proposes a novel network for pedestrian detection with the weakly aligned RGB-T image pairs as inputs. Two effective modules, Cross-modality Differential Enhancement Module (CDEM) and Multi-scale Spatial Alignment Module (MSAM), are proposed. Inspired by the principle of differential circuit, CDEM is embedded in the backbone to suppress the redundant information and extract the complementary information. MSAM allows the multi-scale feature alignment before generating the region proposal, avoiding the mismatch caused by the limited features in the traditional methods. In this paper, the features of thermal infrared image is transformed with the RGB image as the reference. The main contributions of this paper are three-fold:

- A novel network for multi-spectral pedestrian detection called MADENet (Multi-scale Alignment and Differential Enhancement Network) is proposed. In particular, the features of RGB-T images are globally aligned in multiple scales, compared with local feature alignment of region proposals in traditional methods.
- CDEM is presented to extract the complementary features from RGB-T image pairs, providing more discriminative representations for MSAM.
- The proposed method is evaluated on the challenging KAIST multispectral pedestrian dataset. Experimental results demonstrate that the proposed method outperforms the state-of-the-art approaches.

2 Related Work

Hwang *et al.* proposes an extended all-weather pedestrian recognition method ACF based on aligned color and thermal image pairs [4]. Inspired by the Faster R-CNN [10], Liu *et al.* presents four fusion methods for RGB-T pedestrian detection task based on Faster R-CNN [11]. Song *et al.* proposed a multi-spectral pedestrian detection network with simultaneous detection and segmentation, MSDS-RCNN [12], and introduced an auxiliary segmentation task to further improve the performance of this network. CIAN [13] designed a cross-modal interaction attention network in which the cross-modal interaction attention mechanism encodes interactions between modalities and is able to adaptively fuse features.

FusionRPN was first proposed to generate proposals on color and thermal infrared images using independent RPNs, and then FusionRPN was evaluated using support vector regression [5]. Cao *et al.* [7] fed annotation information into the Two-Stream Region Proposal Network (TS-RPN) to learn visible and thermal features. Guan *et al.* introduced a new light-aware weighting mechanism to RGB-T pedestrian detection [14], which can learn multi-modal features under different lighting conditions. In addition to these huge models, Cheng *et al.* proposed a lightweight unified network to balance multi-level information features

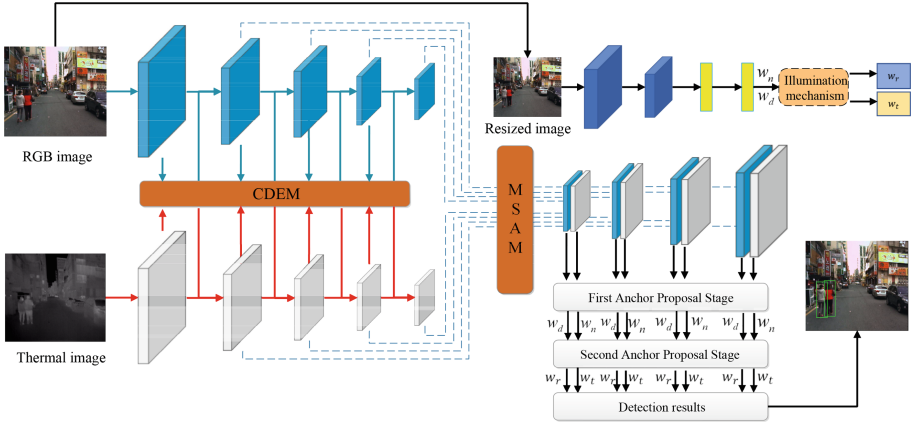


Fig. 2. Overall architecture of MADENet.

from different channels [15]. IAF R-CNN model integrates RGB sub-network [16], thermal IR sub-network and weighting layer into a unified framework, which can effectively improve the multi-spectral pedestrian detection accuracy.

In the earlier literature with respect to multi-spectral pedestrian detection, the weak alignment between RGB and thermal infrared image pairs is not considered. Recently, AR-CNN [8] first try to deal with the weak alignment problem in multi-modality and proposes a new alignment region CNN for end-to-end processing of weakly aligned multi-spectral data. Zhou *et al.* proposes a modal balance network MBNet based on the KAIST-Paired dataset to perform the alignment of the two modalities [9]. In MBNet, the illumination conditions are modeled to calculate the weights of the complementary features of RGB and thermal images and predict the offset values. To solve the problem that the features of different modalities are independent of each other, Hua *et al.* [17] proposed a multi-spectral feature cross-guided learning mechanism to enhance the interaction between multi-spectral feature generation modules and reduce the multi-modal differences. In our view, the intrinsic differences between two modalities can be eliminated by an explicit and simple mechanism, and further improve the performance of multi-spectral pedestrian detection.

3 MADENet: Multi-scale Alignment and Differential Enhancement Network for Pedestrian Detection

3.1 Overall Architecture

Figure 2 presents the overall architecture of proposed network. It is built based on the SSD detection framework [18] and uses a two-branch structure to extract the

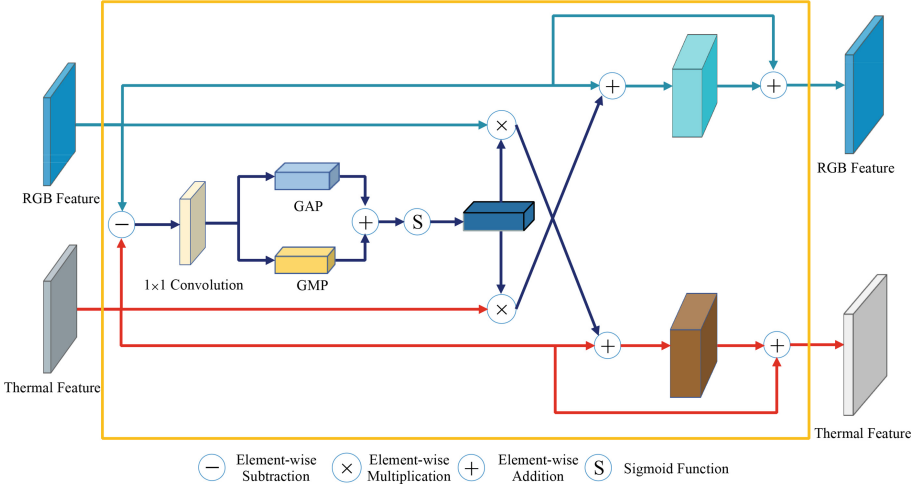


Fig. 3. Cross-modality differential enhancement module. This module uses both global average pooling and maximum pooling features to obtain global receptive fields, which can improve the spatial information aggregation capability of the network.

features of RGB-T modalities, respectively. ResNet-50 [19] is chosen as the backbone. Cross-modality Differential Enhancement Modules (CDEMs) are embedded between both branches to exchange the features and obtain the complementary information for pedestrian detection later. Following the backbone, Multi-scale Spatial Alignment Module (MSAM) is introduced to register the features of weakly aligned modalities. CDEM and MSAM will be presented in detail in Subsects. 3.2 and 3.3.

Inspired by MBNet [9], an illumination network is adopted to estimate the weights to balance the influence of illumination in the stage of detection, as shown in the upper right corner of Fig. 2. To reduce the computational complexity, the resolution of RGB image is firstly resized to 56×56 . For the detection header, a cascaded prediction strategy is used to improve the detection performance, following the concept of ALFNet [20]. w_d and w_n are two illumination parameters used in the first anchor proposal stage. To balance the effect of different illumination environments and improve the detection accuracy, a illumination mechanism is introduced, consisting of the ReLU activation function and the maxpooling layer. w_n and w_d are adjusted to obtain the illumination weights w_r and w_t , which are used in the second anchor proposal stage.

3.2 CDEM: Cross-Modality Differential Enhancement Module

CDEM module is presented to effectively suppress redundant information and extract complementary information, inspired by the ability of differential amplification circuits to suppress common mode signals to amplify differential signals.

CDME is embedded in the backbone, as shown in Fig. 2. Figure 3 illustrates the detailed structure of CDEM.

By denoting the RGB features as F_R and thermal infrared features as F_T , the difference F_D is denoted by

$$F_D = F_R - F_T. \quad (1)$$

Then F_D is processed by a 1×1 convolution to obtain \bar{F}_D , followed by global average pooling and global maximum pooling operations to obtain s_1 and s_2 , respectively.

$$s_1 = F_{GAP}(\bar{F}_D) = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W \bar{F}_D(i, j), \quad (2)$$

$$s_2 = F_{GMP}(\bar{F}_D) = \max(\bar{F}_D(i, j)), \quad (3)$$

where F_{GAP} denotes global average pooling, F_{GMP} indicates global maximum pooling, H and W represent the height and width of the feature map, respectively, followed by the summation of s_1 and s_2 . Sigmoid activation function σ is chosen to yield M_D , as shown in Eq. (4).

$$M_D = \sigma(s_1 + s_2). \quad (4)$$

Subsequently, to capture the complementary features of both modalities, F_T and F_R are augmented by M_D encoded in the residual structure.

$$F'_T = F_T + \text{Res}(F_T + (M_D \cdot F_R)), \quad (5)$$

$$F'_R = F_R + \text{Res}(F_R + (M_D \cdot F_T)), \quad (6)$$

where $\text{Res}()$ denotes the residual function, F'_T and F'_R are the augmented thermal infrared and RGB features generated by CDEM. CDEM is able to enhance the features of one modality by fusing the features of the other, e.g., the information lost at night in RGB image can be partially recovered by the complementary information provided by thermal infrared image. It allows the around-the-clock pedestrian detection.

3.3 MSAM: Multi-scale Spatial Alignment Module

Once the enhanced RGB-T features are obtained based on CDEM, Multi-scale Spatial Alignment Module is implemented to align both features at multiple scales, as shown in Fig. 4. Four scales of features are chosen in this paper, which are generated by the different layers in the backbone. In this paper, features of thermal infrared image are transformed to align ones of RGB image, which is used as the reference.

The multi-scale RGB and thermal features are first spliced, and then the spliced features are input to the localisation network to obtain the affine transformation parameters M_j at scale j , which is formulated by

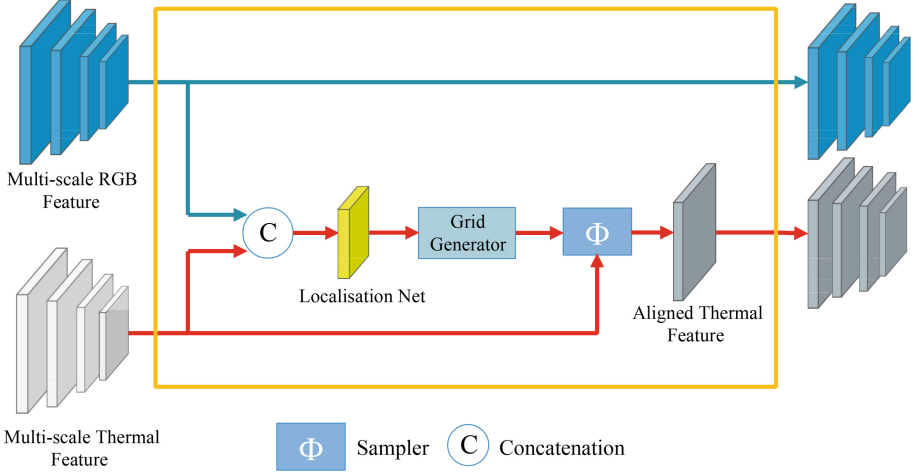


Fig. 4. Multi-scale Spatial Alignment Module.

$$M_j = L_{net}([F_{r,j}, F_{t,j}]), \quad (7)$$

where L_{net} denotes the localisation network, RGB-T features at scale j are denoted as $F_{r,j}$ and $F_{t,j}$, respectively. M_j is fed into the grid generator to calculate $grid_j$. Then combined with thermal features, $grid_j$ goes forward into the sampler Φ to generate aligned features with RGB modality.

$$\tilde{F}_{t,j} = \Phi(F_{t,j}, grid_j). \quad (8)$$

3.4 Loss Function

Motivated by focal loss [21], focal weights are used to alleviate the positive and negative imbalance problem, while the classification loss can be expressed as:

$$L_{cls} = -\alpha \sum_{i \in S_+} (1 - s_i)^\gamma \log(s_i) - (1 - \alpha) \sum_{i \in S_-} s_i^\gamma \log(1 - s_i) \quad (9)$$

where S_+ and S_- denote positive sample anchor frames and negative sample anchor frames, respectively, s_i is the confidence score of sample i . α and β are the focal parameters, which are empirically set to 0.25 and 2, respectively.

The proposed network is optimized by minimizing the cross-entropy loss between the illumination parameters w_n and w_d and the ground true \hat{w}_d and \hat{w}_n during the day and night, which is the illumination loss L_1 .

$$L_1 = -\hat{w}_d \cdot \log(w_d) - \hat{w}_n \cdot \log(w_n) \quad (10)$$

The total loss L consists of two stages of classification loss L_{cls1} and L_{cls2} , regression loss L_{reg1} and L_{reg2} , and illumination loss L_1 , with smoothed L1 loss

used for regression loss [10], and the total loss function L equation is shown below:

$$L = L_1 + L_{cls1} + L_{cls1} + [y = 1] L_{reg1} + [y = 1] L_{reg2} \quad (11)$$

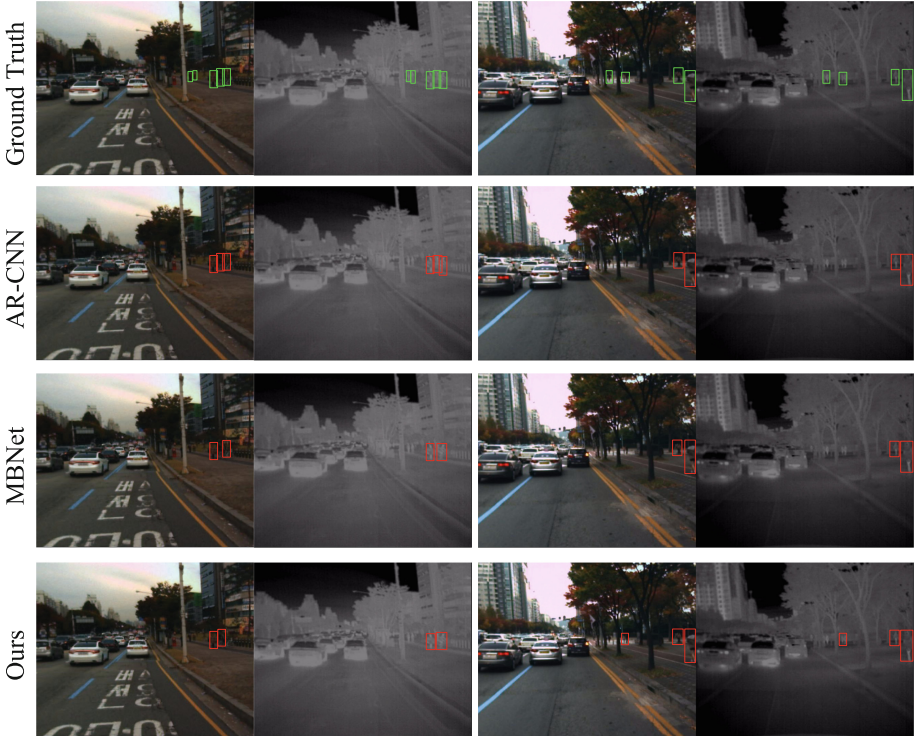


Fig. 5. Comparison results in daytime scenes. From top to bottom: Ground truth, and detection results based on AR-CNN, MBNet and our method. The green box indicates the ground truth, and the red box shows the result of detected pedestrian. (Color figure online)

4 Experimental Results and Discussion

To evaluate our proposed method, two sets of experiments are conducted using the KAIST dataset [4]. In the first set of experiments, the proposed method is compared with the state-of-the-art approaches. The ablation study is implemented in the second set of experiments to verify the effects of CDEM and MSAM modules for pedestrian detection, respectively. Miss rate (MR) is used as the metric for evaluation in experiments.

KAIST pedestrian dataset consists of images of various traffic scenes under different lighting conditions, which contains about 95,000 RGB-thermal image pairs and 1,182 different pedestrian annotations with 640×512 image resolution. The scenes set00-set05 containing 8,892 image pairs are classified as the training set, and scenes set06-set11 containing 2,252 image pairs are the test set. To verify the performance of proposed method, the test set is further divided into nine subsets based on lighting conditions (all, day and night subsets), pedestrian scales (near, medium and far subsets), and occlusion levels (including none, partial, heavy subsets). The MADENet is implemented on a computer with 12GB RAM Nvidia GeForce GTX 2080Ti GPU.

4.1 Comparison with the State-of-the-Art Methods

In this paper, the proposed method is compared with ACF [4], Halfway Fusion [11], Fusion RPN+BF [5], IAF R-CNN [16], IATDNN + IASS [14], CIAN [13], MSDS-RCNN [12], AR-CNN [8], and MBNet [9] as the state-of-the-art methods. As shown in Table 1, our method outperforms the state-of-the-art methods using the entire test set (all subset). MR can reach 8.01, lower than MBNet by 0.12. It indicates that the complementary information from thermal infrared images are better fused, and extract more prominent features for pedestrian detection using RGB-T image pairs. Additionally, the proposed method performs the best in night, near, medium, far, none, and partial subsets among the nine subsets. In particular, in the night subset, MR is decreased by 1.04, compared with MBNet that is the best in the state-of-the-art methods. For the remaining two subsets (day and heavy), our method ranks second, lower than MBNet and AR-CNN, respectively.

Table 1. Miss rate comparison with the state-of-the-art methods using nine subsets of test set.

Methods	All	Day	Night	Near	Medium	Far	None	Partial	Heavy
ACF [4]	47.32	42.57	56.17	28.74	53.67	88.20	62.94	81.40	88.08
Halfway Fusion [11]	25.75	24.88	26.59	8.13	30.34	75.70	43.13	65.21	74.36
Fusion RPN+BF [5]	18.29	19.57	16.27	0.04	30.87	88.86	47.45	56.10	72.20
IAF R-CNN [16]	15.73	14.55	18.26	0.96	25.54	77.84	40.17	48.40	69.76
IATDNN + IASS [14]	14.95	14.67	15.72	0.04	28.55	83.42	45.43	46.25	64.57
CIAN [13]	14.12	14.77	11.13	3.71	19.04	55.82	30.31	41.57	62.48
MSDS-RCNN [12]	11.63	10.60	13.73	1.29	16.19	63.73	29.86	38.71	63.37
AR-CNN [8]	9.34	9.94	8.38	0.00	16.08	69.00	31.40	38.63	55.73
MBNet [9]	8.13	8.28	7.86	0.00	16.07	55.99	27.74	35.43	59.14
Ours	8.01	8.89	6.82	0.00	14.28	54.52	26.91	34.98	58.10

To better demonstrate the effectiveness of proposed method, the results of pedestrian detection under different illumination conditions are visualized,

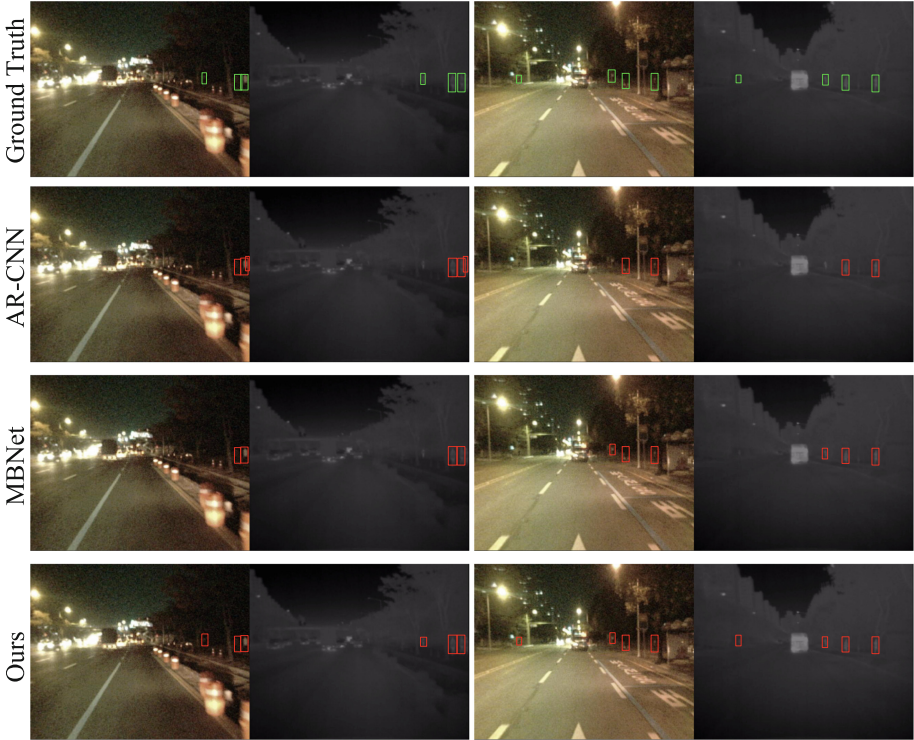


Fig. 6. Comparison results in night scenes. From top to bottom: Ground truth, and detection results of AR-CNN, MBNet and our method. The green box indicates the ground truth, and the red box shows the result of detected pedestrian. (Color figure online)

compared with AR-CNN [8] and MBNet [9], which perform better in Table 1. Figure 5 shows the results in the two classical scenes in day time. It can be observed that both AR-CNN and MBNet have missed pedestrians in the daytime scenes in Fig. 5, especially the pedestrians at a longer distance in the scene are not detected. According to Table 1, although the performance of our proposed method is not improved compared with MBNet in the day subset, our method performs better when the pedestrian is at distance, as shown in the second daytime scene in Fig. 5. The corresponding results in the night scenes are shown in Fig. 6, our proposed method has the superior performance, while AR-CNN has missed and false detection and MBNet has missed detection.

4.2 Ablation Study

To assess how CDEM and MSAM modules affect the performance of pedestrian detection, the ablation study using the KAIST dataset is carried out. The baseline is the model without CDEM and MSAM modules in the proposed method,

Table 2. Miss rate comparison of ablation study using nine subsets of test set.

CDEM	MSAM	All	Day	Night	Near	Medium	Far	None	Partial	Heavy
		11.45	12.54	9.53	0.00	18.74	57.90	29.65	39.95	62.20
	✓	9.56	10.13	8.72	0.00	16.49	56.79	28.32	37.19	60.87
✓		9.44	10.49	7.45	0.00	16.08	57.42	28.05	37.86	62.14
✓	✓	8.01	8.89	6.82	0.00	14.28	54.52	26.91	34.98	58.10

using a simple feature summation instead as the fusion strategy. Table 2 shows the comparison results with different configurations of modules. By any one of CDEM and MSAM modules, both MRs are reduced greatly, which means the proposed modules can effectively improve the performance of pedestrian detection. In particular, for the night subset, although the thermal infrared image provides the complementary information in CDEM module, the performance of pedestrian detection is improved less than the model with MSAM only. The main cause is the weak alignment of image pairs. Considering the results of other subsets, CDEM module performs better in day, far, partial and heavy subsets. The reasons are analyzed as follows.

- Only RGB image can provide enough features for detection in the day time;
- The position shift of both modalities becomes tiny in the far subset with small pedestrian scale;
- Thermal infrared images can preserve less representative features in subsets with partial and heavy occlusion.

By combining CDEM and MSAM modules in the proposed method, all MRs of nine subsets is reduced further. Especially, the performance is improved by 1.9, compared with the result using the model with only CDEM module. For the worst case using the heavy subset, MR can reach 58.10. It is decreased by 4.1, compared with the baseline. Therefore, the proposed CDEM and MSAM modules can effectively improve the performance of pedestrian detection.

5 Conclusion

This paper presents a multi-spectral pedestrian detection, named Multi-scale Alignment and Differential Enhancement Network (MADENet). Two key components are Cross-Modality Differential Enhancement Module (CDEM) and Multi-scale Spatial Alignment Module (MSAM). The former one is used to suppress the redundant features and extract complementary information between modalities, and the latter is designed to transform thermal features to align the features of RGB image. The public KAIST dataset is used to verify the proposed method. Experimental results show that the proposed method has the superior performance, compared with the existing approaches. The ablation study also proves that the proposed CDEM and MSAM modules can effectively improve the performance of pedestrian detection.

Acknowledgment. This work was sponsored by Beijing Nova Program (20230-484409), National Natural Science Foundation of China (62272322, 62272323), applied basic research project of Liaoning province (2022JH2/101300279).

References

1. Chen, Z., Huang, X.: Pedestrian detection for autonomous vehicle using Multispectral cameras. *IEEE Trans. Intell. Veh.* **4**(2), 211–219 (2019)
2. Selvi, C., Amudha, J.: Automatic video surveillance system for pedestrian crossing using digital image processing. *Indian J. Sci. Technol.* **12**, 1–6 (2019)
3. Buddhharaju, P., Pavlidis, I.T., Tsiamyrtzis, P., Bazakos, M.: Physiology-based face recognition in the thermal infrared spectrum. *IEEE Trans. Pattern Anal. Mach. Intell.* **29**(4), 613–626 (2007)
4. Hwang, S., Park, J., Kim, N., Choi, Y., Kweon, I.S.: Multispectral pedestrian detection: benchmark dataset and baseline. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1037–1045 (2015)
5. Konig, D., Adam, M., Jarvers, C., Layher, G., Neumann, H., Teutsch, M.: Fully convolutional region proposal networks for multispectral person detection. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 243–250 (2017)
6. Ding, L., Wang, Y., Laganière, R., Huang, D., Luo, X., Zhang, H.: A robust and fast multispectral pedestrian detection deep network. *Knowl.-Based Syst.* **227**, 106990 (2021)
7. Cao, Y., Guan, D., Huang, W., Yang, J., Cao, Y., Qiao, Y.: Pedestrian detection with unsupervised multispectral feature learning using deep neural networks. *Inf. Fusion* **46**, 206–217 (2019)
8. Zhang, L., Zhu, X., Chen, X., Yang, X., Lei, Z., Liu, Z.: Weakly aligned cross-modal learning for multispectral pedestrian detection. In: IEEE/CVF International Conference on Computer Vision (ICCV), pp. 5126–5136 (2019)
9. Zhou, K., Chen, L., Cao, X.: Improving multispectral pedestrian detection by addressing modality imbalance problems. In: *Computer Vision – ECCV 2020*, pp. 787–803. Springer, Cham (2020)
10. Ren, J.S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **39**(6), 1137–1149 (2016)
11. Liu, J., Zhang, S., Wang, S., Metaxas, D.: Multispectral deep neural networks for pedestrian detection. In: Richard, E.R.H., Wilson, C., Smith, W.A.P. (eds.) *Proceedings of the British Machine Vision Conference (BMVC)*, pp. 73.1–73.13. BMVA Press (2016)
12. Li, C., Song, D., Tong, R., Tang, M.: Multispectral pedestrian detection via simultaneous detection and segmentation (2018)
13. Zhang, L., et al.: Cross-modality interactive attention network for multispectral pedestrian detection. *Inf. Fusion* **50**, 20–29 (2019)
14. Guan, D., Cao, Y., Yang, J., Cao, Y., Yang, M.Y.: Fusion of multispectral data through illumination-aware deep neural networks for pedestrian detection. *Inf. Fusion* **50**, 148–157 (2019)
15. Cheng, C., Wu, X.-J., Xu, T., Chen, G.: Unifusion: a lightweight unified image fusion network. *IEEE Trans. Instrum. Meas.* **70**, 1–14 (2021)
16. Li, M., Tang, R.: Illumination-aware faster R-CNN for robust multispectral pedestrian detection. *Pattern Recognit. J. Pattern Recognit. Soc.* **85** (2019)

17. Hua, C., Sun, M., Zhu, Y., Jiang, Y., Yu, J., Chen, Y.: Pedestrian detection network with multi-modal cross-guided learning. *Digit. Signal Process.* 103370 (2022)
18. Liu, W., et al.: SSD: single shot multibox detector. In: *European Conference on Computer Vision*, pp. 21–37. Springer, Cham (2016)
19. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778 (2016)
20. Liu, W., Liao, S., Hu, W., Liang, X., Chen, X.: Learning efficient single-stage pedestrian detectors by asymptotic localization fitting. In: *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 618–634 (2018)
21. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. *IEEE Trans. Pattern Anal. Mach. Intell.* 2999–3007 (2017)



CrackUDA: Incremental Unsupervised Domain Adaptation for Improved Crack Segmentation in Civil Structures

Kushagra Srivastava^(✉), Damodar Datta Kancharla, Rizvi Tahereen,
Pradeep Kumar Ramancharla, Ravi Kiran Sarvadevabhatla,
and Harikumar Kandath

International Institute of Information Technology, Hyderabad, India
kushagra2000@gmail.com

Abstract. Crack segmentation plays a crucial role in ensuring the structural integrity and seismic safety of civil structures. However, existing crack segmentation algorithms encounter challenges in maintaining accuracy with domain shifts across datasets. To address this issue, we propose a novel deep network that employs incremental training with unsupervised domain adaptation (UDA) using adversarial learning, without a significant drop in accuracy in the source domain. Our approach leverages an encoder-decoder architecture, consisting of both domain-invariant and domain-specific parameters. The encoder learns shared crack features across all domains, ensuring robustness to domain variations. Simultaneously, the decoder's domain-specific parameters capture domain-specific features unique to each domain. By combining these components, our model achieves improved crack segmentation performance. Furthermore, we introduce BuildCrack, a new crack dataset comparable to sub-datasets of the well-established CrackSeg9K dataset in terms of image count and crack percentage. We evaluate our proposed approach against state-of-the-art UDA methods using different sub-datasets of CrackSeg9K and our custom dataset. Our experimental results demonstrate a significant improvement in crack segmentation accuracy and generalization across target domains compared to other UDA methods - specifically, an improvement of 0.65 and 2.7 mIoU on source and target domains respectively. Additional details and code can be accessed from <https://crackuda.github.io>.

Keywords: Crack Segmentation · Civil Inspection · Domain Adaptation · Dataset · Incremental Learning

1 Introduction

Identifying cracks in structures such as roads, pavements, and buildings is an important civil engineering task. This is especially crucial in determining a building's structural health and risk of failure during seismic activity [37]. This task is

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-78113-1_6.

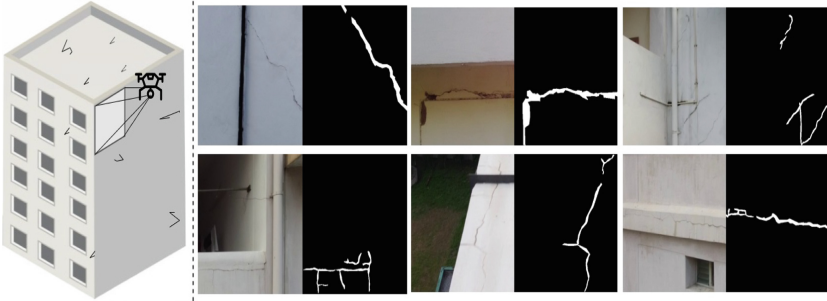


Fig. 1. BuildCrack dataset was captured by imaging building facades using a drone-mounted camera from different angles and distances. BuildCrack has images with low contrast, occlusions, and shadows, which challenge the model’s robustness. Sample images from our building crack dataset are shown. This dataset will be made public.

being increasingly performed using visual imagery. However, the small footprint of cracks relative to building size and lack of regular structure make crack localization a challenging image segmentation problem. Different approaches have been explored over the years, ranging from rule-based to data-driven methods, for crack segmentation. Data-driven methods [24] have gained prominence with the rise in the availability of crack datasets [1, 23, 30, 45, 50]. These methods have shown remarkable results in segmentation tasks.

However, a key limitation of these approaches is their poor generalization across different domains, as datasets from various sources often have different distributions. This lack of generalization is evident when a model trained on one domain (source domain) is applied to a dataset from a different domain (target domain). Several factors contribute to the domain shift observed in crack datasets. These include differences in image features, such as the contrast between cracks and their background, variations in crack shapes due to surface textures and lighting conditions, and the overall appearance of cracks [25, 34].

To address this challenge, domain adaptation techniques can be employed to reduce the domain shift. It is a viable solution since it alleviates the need for costly and labor-intensive annotation of crack segmentation data. Unsupervised Domain Adaptation (UDA) is a specific approach that adapts a network trained on a labeled source dataset to an unlabeled target dataset, effectively mitigating the problems associated with domain shift across datasets and high annotation costs [3, 17, 35, 41, 42, 46, 47, 57]. While these UDA approaches have been extensively tested in domain adaptation tasks using real and synthetic autonomous driving datasets, our work demonstrates that these methods do not yield satisfactory results for the challenging crack segmentation setting

Our approach is designed to address the challenges of crack segmentation through an incremental learning setting. We employ a two-step process to adapt our network, trained on a labeled source dataset, to an unlabeled target dataset. To overcome catastrophic forgetting often observed in incremental learning approaches [32], our network architecture learns both domain-invariant and domain-specific feature representations. Our paper makes the following key contributions:

- We propose CrackUDA, a novel incremental UDA approach that ensures robust adaptation and effective crack segmentation (Sect. 4).
- We demonstrate the effectiveness of CrackUDA by achieving higher accuracy in the task of building crack segmentation, surpassing the state-of-the-art UDA methods. Specifically, CrackUDA yields an improvement of 0.65 and 2.7 mIoU on the source and target domains, respectively (Sect. 6).
- We introduce BuildCrack, a new building crack dataset collected via a drone (Sect. 6).

2 Related Works

Crack segmentation approaches can be broadly categorized into two types: (i) rule-based and (ii) data-driven methods. Rule-based methods use human-defined rules to make decisions. Most of the rule-based methods have low accuracy because of non-uniform backgrounds, varying light conditions, and the brittle nature of the parameters [20]. Data-driven methods leverage data samples to learn patterns and adjust the parameters of a model for a specific task. In particular, deep learning-based methods have demonstrated significant potential in crack segmentation and can be divided into supervised, weakly supervised, and semi-supervised based on the extent of supervision.

Crack segmentation is dominated by supervised learning approaches. Encoder-decoder architecture [2, 30, 55] has been popular for excellent performance in pixel-wise segmentation, provided accurately labeled segmentation maps are available. The encoder downsamples the input images to form a high-dimensional feature vector while the decoder reconstructs unique segmentation maps using this feature vector. CrackNet [51] modifies the encoder-decoder architecture by using same-size convolution filters across layers to maintain explicit pixel-pixel representation. DeepCrack [30] uses a fully convolution network (FCN) architecture with additional convolution layers at the end of a traditional CNN which upsamples feature maps of different scales to the original size and recovers fine-grained structures. CrackSeg9K [23] compared different state-of-the-art segmentation models. It was concluded that DeepLab v3+ [2] with ResNet and XceptionNet as the backbones worked best on linear cracks but the accuracy drops on webbed and branched cracks. With the introduction of vision transformer [10], self-attention has become an efficient tool for learning non-local features. Crackformer [29] uses sequential self-attention networks for crack segmentation. The performance of supervised segmentation approaches relies on accurate semantic labels. Such approaches seldom generalize to datasets of different domains. [6] proposed a curvilinear structure segmentation approach for crack segmentation on diverse datasets such as Crack500 [50] and CrackTree200 [55].

[22] propose a weakly supervised approach that uses inferior quality labels for crack segmentation. They demonstrated their network’s capability to perform in out-of-domain (OOD) cases, but accuracy suffers when there are thin cracks in the target dataset. Semi-supervised approaches have used generative adversarial networks [27] and super-resolution [21] to generate pseudo-labels for

training their network. However, the performance of these methods depends on the quality of the pseudo-labels. Though semi-supervised approaches perform well in the case of OOD, they require some labeled data of the target domain.

Since the conspicuous hurdle is reliable labeled data and poor generalization, our work uses UDA. UDA has demonstrated its potential for various vision tasks such as object detection [4, 5, 28, 54], classification [12, 31, 36, 44, 49], and more relevantly, semantic segmentation [3, 17, 35, 41, 42, 47, 56, 57] as well as crack segmentation [48].

3 Preliminaries

In this section, we formally define our problem statement and provide an overview of UDA and incremental learning.

3.1 Problem Statement

Consider a source distribution S and target distribution T , both defined on the input-label space $X \times Y$. In this setting, $X \in \mathbb{R}^{H \times W \times 3}$ represents RGB images, while $Y \in \mathbb{R}^{H \times W}$ corresponds to semantic labels. Both the source and target distributions share the same K semantic class labels, $1, \dots, K$. Specifically, X represents building patches, and Y contains label maps with two class labels, namely *background* and *crack* ($K = 2$). We have access to a set of labeled source samples $S = (x_j^s, y_j^s), j = 1, 2, \dots, n_s$ and unlabeled target samples $T = x_j^t, j = 1, 2, \dots, n_t$, where n_s and n_t denote the total number of source and target samples, respectively. Our objective is to train a network using the labeled source domain data S and the unlabeled target domain data T to generate accurate predictions $\hat{y}_j^t, j = 1, 2, \dots, n_t$. In the context of crack segmentation, this problem is reduced to a binary segmentation task. However, due to the relatively small number of crack pixels present in each patch, a significant class imbalance exists.

3.2 Incremental Learning

Incremental learning involves training an existing model on a sequence of τ tasks, where each task τ_i corresponds to a distinct dataset of domains D_i . In our specific setting, D_i represents an image dataset consisting of pairs of input images and their corresponding semantic labels, denoted as $D_i = \{X_j, Y_j\}$. We will use τ and D interchangeably. Each task τ_i is focused on semantic segmentation. Typically, a domain shift exists between consecutive tasks (i.e. D_t exhibits non-trivial differences compared to D_{t-1}). The objective is to train a single semantic segmentation model M that can effectively segment image data across each domain D_t in a sequential manner. Thus, for a given task τ , at each step t , our aim is to learn a mapping $M_t(X_t, t) = Y_t$ for the t^{th} domain $D_t = (X_t, Y_t)$. Importantly, the learned model should maintain satisfactory performance on previous domains D_{t-i} , where $0 < i < t$, ensuring minimal degradation in performance.

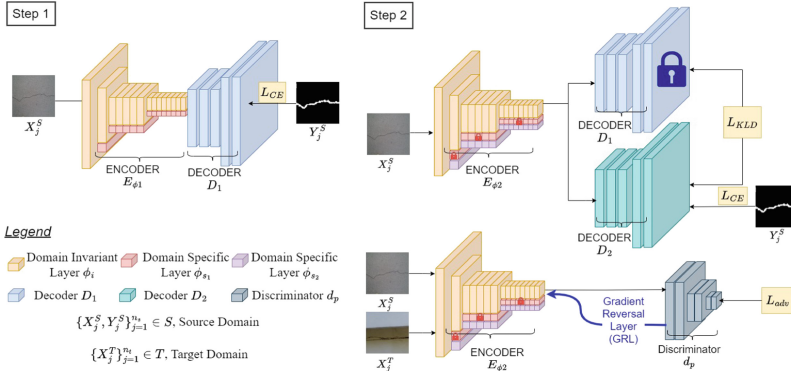


Fig. 2. Overview of our proposed architecture (Sect. 5). In step 1 we train our network, M_1 , using the labeled source dataset S for binary segmentation. In step 2, decoder D_1 and ϕ_{s_1} are frozen, and a new set of domain-specific parameters ϕ_{s_2} are added and we call this model M_2 . An alternating training strategy is followed in which we first train for binary segmentation on the source domain followed by adversarial training on both source and target domains.

3.3 Unsupervised Domain Adaptation (UDA)

UDA methods for semantic segmentation can be broadly classified into three groups: Self-training, Feature Alignment, and Adversarial Training approaches. Self-training approaches involve training a segmentation model on the labeled source domain to compute pseudo-labels [26] for the target domain. These pseudo-labels can be pre-computed offline [57] or online during training. To avoid training instabilities, several methods such as consistency regularization [40] based on data augmentation [7], domain mix-up [41], and pseudo-label prototypes [52] have been used. Several methods also combine [47] self-training and adversarial training to perform UDA. Feature alignment [16] approaches aim to align the feature representations of the source and target domains. This technique involves training a segmentation model with a domain adaptation loss, which encourages the feature representations of the source and target domains to be similar. For further details about UDA, we recommend reading [39]. In the context of this work, we mainly focus on adversarial training.

Adversarial training aims to minimize the domain discrepancy between the source and target distributions in a GAN framework [13]. The distributions can be aligned at the input [15], output [46], or patch [43] levels. For a GAN framework, the architecture (see Step 2 in Fig. 2) is composed of a feature extractor (E_{ϕ_2}), a label predictor (D_1 and D_2), a domain classifier (d_p), and a gradient reversal layer (GRL) in between E_{ϕ_2} and d_p . d_p is trained to classify the source and target domains, while the segmentation model is trained to generate

segmentation maps that are domain indistinguishable. A high-dimensional feature vector x corresponding to input X can be obtained below.

$$E_\phi(X) = x \quad (1)$$

In forward propagation, the GRL is implemented as an identity-mapping function while in back-propagation the GRL multiplies the gradient calculated from the domain-classification error by a negative scalar. This negative gradient is propagated to the feature extractor. It can be formulated as below.

$$R_\lambda(x) = x \quad (2)$$

$$\frac{dR_\lambda}{dx} = -\lambda I \quad (3)$$

x is the corresponding feature vector for input X obtained from Eq. 1, I is an identity matrix and R_λ is the GRL. To mitigate the impact of large domain classification errors at the early stages of training, the value of λ is regulated adaptively as given below where p stands for the number of elapsed epochs.

$$\lambda = \frac{2}{1 + e^{-\lambda p}} - 1 \quad (4)$$

4 Methodology

4.1 Proposed Framework (CrackUDA)

We design CrackUDA, a two-step unsupervised domain adaptation approach for binary segmentation of cracks (see Fig. 2). Our model M comprises an encoder E_{ϕ_k} , two domain-specific decoders D_1 and D_2 for predicting domain-specific labels, and a discriminator network d_ρ which acts as a domain classifier. The encoder E_{ϕ_k} consists of a set of shared domain-invariant parameters ϕ_i which are universal to all domains and a set of domain-specific parameters ϕ_{s_k} which are exclusive to respective domains. Domain-invariant parameters learn common features across all domains and domain-specific parameters learn domain-specific features for the respective domains.

As shown in Fig. 2, the first step involves learning a binary segmentation M_1 on the source dataset S . M_1 is composed of decoder D_1 and encoder E_{ϕ_1} in which both ϕ_i and ϕ_{s_1} (domain-specific parameters for source dataset) are trainable. In step 2, we add new domain-specific parameters, ϕ_{s_2} , to the new encoder E_{ϕ_2} and a domain-specific decoder D_2 and call this model M_2 . We follow an alternating training strategy in which M_2 is trained for binary segmentation followed by adversarial training through the discriminator d_ρ . This training strategy enables our model to adapt to T while retaining its performance on S .

4.2 Optimization Strategy

For any given step k , the domain-specific layers ϕ_{s_k} are trained only on the softmax cross-entropy loss function over the label space of the source domain S . The

Table 1. Quantitative comparison of existing datasets and our dataset. The datasets mentioned above (except our new dataset) have been aggregated as CrackSeg9K [23].

Dataset	Size	% of Crack	Description
Crack500 [50]	3126	6.03	Collected using a smartphone
Rissbilder [1]	2736	2.70	Architectural Cracks
SDNET2018 [9]	1411	0	Non-crack images
Volker [45]	427	4.05	Cracks collected from pavements and buildings.
DeepCrack [30]	443	3.58	Cracks collected from pavements and buildings.
GAPS384 [11]	383	1.21	Cracks collected from pavements
BuildCrack (ours)	358	4.30	Building Cracks collected using a drone.
Masonry [8]	240	4.21	Contains crack in masonry walls
CrackTree200 [55]	175	0.31	Cracks collected from pavements and buildings.
CFD [19]	118	1.61	Urban road surface cracks
Ceramic [18]	100	2.05	Cracks on different colors and textures of ceramics.

forward pass and the softmax cross-entropy loss function, ζ , can be formulated as given below.

$$D_k(E_{\phi_k}(X_j, \phi_i, \phi_{s_k})) = \hat{Y}_j \quad (5)$$

$$L_{CE} = \frac{1}{N} \sum_{X_j, Y_j \in S} \zeta(Y_j, \hat{Y}_j) \quad (6)$$

In step 2, in addition to the cross-entropy loss, we use a regularization loss L_{KLD} to optimize the shared weights ϕ_i during the segmentation phase as given in the equations below.

$$\hat{y}_j^1 = M_1(X_j, \phi_i, \phi_{s_1}) \quad (7)$$

$$\hat{y}_j^2 = M_2(X_j, \phi_i, \phi_{s_1}) \quad (8)$$

$$L_{KLD} = \sum_{X_j \in S} \psi(\hat{y}_j^2, \hat{y}_j^1) \quad (9)$$

where \hat{y}_j^1 and \hat{y}_j^2 are the softmax probability distributions maps of M_1 and M_2 on samples from the source domain respectively and ψ is the KL-divergence loss between the two probability distributions. The total loss for the segmentation phase is given as below.

$$L_{Total} = \lambda_{CE} \cdot L_{CE} + \lambda_{KLD} \cdot L_{KLD} \quad (10)$$

For the adversarial training phase, we use a binary cross-entropy loss, L_{adv} to classify whether the feature vector obtained from E_{ϕ_2} corresponds to an image sample from the source or target domain. This loss function can be formulated as given in the equations below.

$$d_\rho(E_{\phi_2}(X_j, \phi_i, \phi_{s_2})) = \hat{d}_j \quad (11)$$

$$L_{BCE} = \frac{1}{N} \sum_{X_j \in S, T} \omega(d_j, \hat{d}_j) \quad (12)$$

where ω is the binary cross-entropy loss, d_j and \hat{d}_j are the true and predicted domain labels and d_ρ is the discriminator network. d_j is a binary variable that indicates whether the sample is from the source or the target domain.

5 Implementation Details

5.1 Network Architecture

We use ERFNet [38] as the backbone for our network with the discriminator as an FCN. The value of λ is updated as per Eq. 4. The encoder comprises residual-adaptor blocks [14]. Each residual-adaptor block has a set of domain-invariant parameters ($\phi_i = \{\phi_{w1}, \phi_{w2}\}$) and a set of domain-specific parameters ($\phi_{s_k} = \{\alpha^w, \alpha^s, \alpha^b\}$). ϕ_{w1} and ϕ_{w2} are 3×3 convolutional layers of a residual unit shared across all the domains. Domain-specific layers in the residual adaptor unit are of two kinds: Domain-specific parallel residual adapter layers (DS-RAP) and domain-specific batch normalization layers (DS-BN). DS-RAP (α_w) are 1×1 convolutional layers added to the shared convolutional layers in parallel. DS-BN shifts and scales the normalized input as $s \cdot x + b$ where α_s and α_b represent the scaling and shifting parameters respectively.

5.2 Training

In step 1 (see Fig. 2), we train M_1 on the source domain S in a binary segmentation setting. In step 2, we follow an alternating training strategy. We first train M_2 for binary segmentation for 10 epochs on S followed by adversarial training on a mini-batch of an equal number of samples from S and T for 5 epochs. Overall, M_2 is trained for 150 epochs. λ_{CE} and λ_{KLD} are set to 1 and 0.1 respectively. In Step 1, segmentation is performed on S for a total of 150 epochs. The Adam optimizer is utilized with a learning rate (LR) of $5e^{-4}$, and a batch size of 8. In Step 2, segmentation is again performed on S for 10 epochs, employing the same optimizer, learning rate, and batch size as in Step 1. Additionally, an adversarial training step is introduced, involving both the source (S) and target (T) datasets. This adversarial training step is conducted for 5 epochs. Training protocols have been summarized in Algorithm 1 and 2 in the supplementary material. For both steps, the model checkpoints were saved during training. For Step 2, The model checkpoints are saved only if there is an increase in mIoU scores for both the source and target domains.

Table 2. mIoU scores of CrackUDA (our approach) for steps 1 and 2 for sub-datasets of CrackSeg9K and BuildCrack (our custom dataset). Here, *Dataset Excluded* is the sub-dataset left out of training and validation sets of the source domain. This *Dataset Excluded* is aggregated with our dataset to form the overall dataset. Source mIoU is the performance of the network on the CrackSeg9K validation set excluding the mentioned dataset. The results show that using an incremental learning strategy for UDA leads to better performance in the target domain (see columns Dataset Excluded, Our Dataset, and Overall for Step 2) without a severe drop in performance in the source domain.

Dataset Excluded	Step 1				Step 2			
	Source mIoU	Target mIoU			Source mIoU	Target mIoU		
		Dataset Excluded	Build Crack	Overall (Excluded + Our)		Dataset Excluded	Build Crack	Overall (Excluded + Our)
Mason	82.72	53.03	54.69	54.12	79.94	61.94	55.35	57.62
Ceramic	82.67	49.55	62.55	59.98	78.86	50.55	63.73	62.16
CFD	82.87	78.83	62.57	67.92	79.91	79.08	55.91	63.80
Crack500	83.30	56.84	62.58	57.27	78.33	79.24	54.16	78.10
CrackTree200	82.52	77.64	57.69	66.61	79.26	81.48	52.05	65.28
DeepCrack	82.24	78.92	59.61	72.78	78.61	82.55	59.02	74.71
GAPS	82.77	65.03	60.37	62.71	78.47	70.62	59.57	65.26
Rissbilder	82.90	71.92	57.02	70.19	79.97	78.33	55.36	75.40
Volker	82.64	75.20	57.80	69.98	79.77	76.80	57.60	70.40

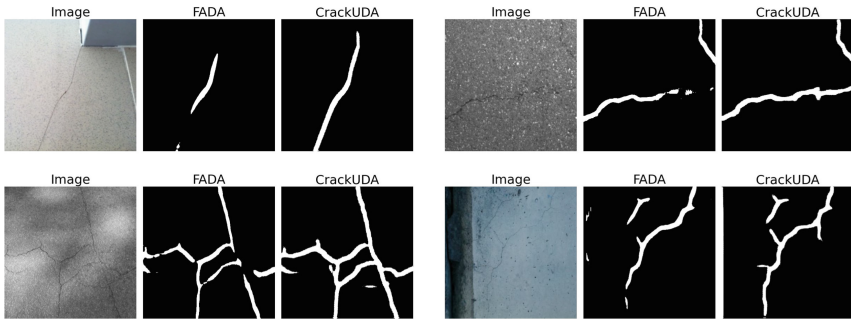


Fig. 3. Qualitative results for CrackSeg9K validation set for CrackUDA and FADA [47].

6 Experiments and Results

6.1 Datasets and Evaluation Metrics

We validate the performance of our approach using two datasets: CrackSeg9K [23] and BuildCrack, the custom dataset that we introduce (see Fig. 1). BuildCrack has images with low contrast, occlusions, and shadows, which challenge the model’s robustness. CrackSeg9K is a culmination of smaller open-source crack datasets (CFD [19], Masonry [8], Ceramic [18], Rissbilder [1], Volker [45], SDNET2018 [9], DeepCrack [30], GAPS384 [11], Crack500 [50], and CrackTree200 [55]) with more consistent labeling. Details regarding these 10 sub-datasets can be found in Table 1. After removing duplicate images in the original dataset of 9255 images, we divided the remaining 8513 images into 6794 training images, and 1719 validation images. This ratio of 4:1 was maintained across all

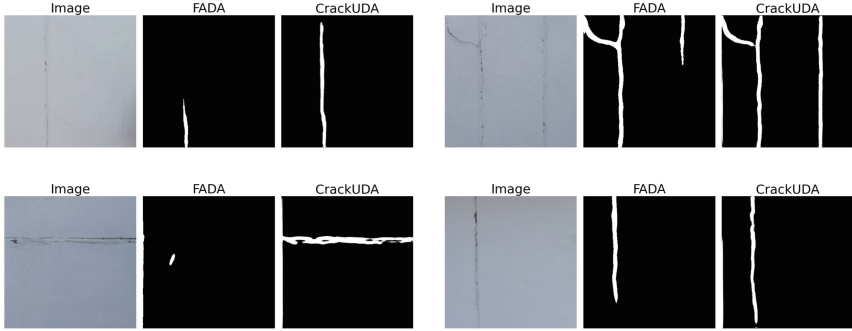


Fig. 4. Qualitative results for BuildCrack for our network and FADA [47].

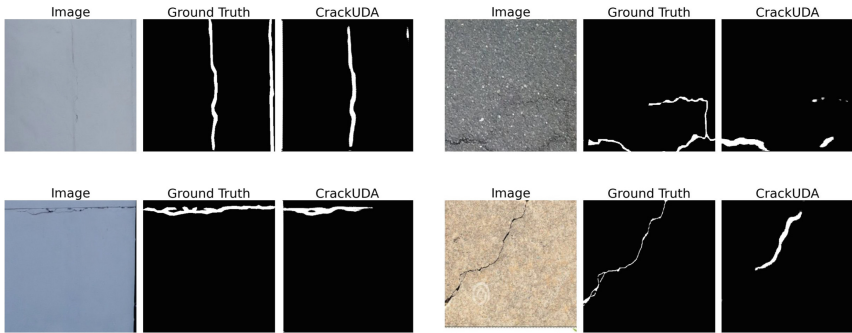


Fig. 5. Some cases in which our approach does not perform well in CrackSeg9K and BuildCrack.

sub-datasets to ensure an equal proportion of each sub-dataset in both the training and validation sets. Our dataset BuildCrack comprises 358 binary labeled crack images collected using DJI Mavic Mini¹. All the ground-truth labels in both CrackSeg9K and BuildCrack have two class labels: *background* and *crack*. We use all 358 BuildCrack images for training and validation. We use mean Intersection-over-Union (mIoU) to evaluate the performance of our approach.

6.2 UDA Baselines

We evaluate the performance of our network (CrackUDA) against 8 state-of-the-art UDA baselines and a state-of-the-art self-supervised UDA baseline in which CrackSeg9K and our dataset (BuildCrack) are the source and target datasets respectively. The performance of our approach and the baselines are reported on the validation set of CrackSeg9K and all 358 images of the BuildCrack (see Table 3). [17, 33, 46, 56] did not converge for this setting. Out of the baselines,

¹ UAV specification details can be found at the official DJI website: <https://www.dji.com/mavic-mini>.

Table 3. Comparison of mIoU scores on the validation set of CrackSeg9K and BuildCrack (target dataset) with state-of-the-art UDA methods. * approaches did not converge for our setting. Our approach achieves the best generalization performance.

DA Method	Source (CrackSeg9k)	Target (BuildCrack)
AdaptSegnet* [42]	47.53	48.47
MaxSquare (ICCV '19) [3]	57.60	50.50
ADVENT* (CVPR '19) [46]	47.51	48.47
IAST* (ECCV '20) [33]	46.79	46.78
DAFormer* (CVPR '22) [17]	47.54	48.47
DACS (WACV '21) [41]	58.46	58.11
CBST * (ECCV '18) [56]	47.53	48.47
ProDA (CVPR '21) [53]	50.32	47.94
FADA (ECCV '20) [47]	79.18	60.73
CrackUDA	79.83	63.43

FADA [47] obtains the best mIoU score of 79.18 on the validation set of CrackSeg9K and 60.73 on our dataset. CrackUDA outperformed FADA by 0.65 and 2.7 mIoU on the validation set of CrackSeg9K and the entire BuildCrack dataset respectively.

6.3 Experiments on Sub-datasets of CrackSeg9K

We conduct experiments on sub-datasets of CrackSeg9K, systematically excluding one sub-dataset at a time from both the training and validation sets of the source domain S during the two-step process. This exclusion preserves the 4:1 ratio between training and validation sets, maintaining the proportion of samples within each subset. The excluded dataset, combined with BuildCrack, forms the target dataset T . Table 2 presents the mIoU scores obtained on the source dataset, excluded dataset, BuildCrack, and the new target dataset (excluded dataset + BuildCrack). The results show a significant increase in mIoU scores for the excluded datasets in Step 2. We observe an mIoU increase of 8.91 for Masonry, 0.99 for Ceramic, 0.25 for CFD, 22.4 for Crack500, 3.84 for Crack-Tree200, 3.63 for DeepCrack, 5.59 for GAPS, 6.41 for Rissbilder, and 1.60 for Volker. This demonstrates the generalization capabilities of our approach across target domains without notable decline in performance on the source domain. A comparison of our approach against a state-of-the-art supervised approach and performance impact due to switching source and target domains can be found in the supplementary material.

6.4 Ablation Studies

In step 2 of our approach, we use L_{KLD} to optimize the shared parameters ϕ_i through the softmax probability maps obtained from the domain-specific

Table 4. Ablation study on the contribution of each component of CrackUDA for the validation set of CrackSeg9K (source domain) and BuildCrack (target domain) setting.

Method	L_{KLD}	GRL	CrackSeg9K	BuildCrack
1 Step	×	×	82.17	60.44
2 Step w/o GRL	✓	×	78.99	61.82
2 Step w/o KLD	×	✓	78.99	53.5
2 Step	✓	✓	79.83	63.43

decoders. Our experiments show that removing this loss from step 2 leads to a 9.93 mIoU drop for the target dataset and a 0.84 mIoU drop for the source dataset ('2 Step w/o KLD' in Table 4). This shows that optimizing for the shared parameters ϕ_i in step 2 helps the network learn common features of the source and target domain leading to better generalization across both domains. Next, we show that disabling adversarial training in step 2 leads to a 0.84 mIoU drop in the source dataset and a 1.61 mIoU drop in the target dataset (referred to as 2 Step w/o GRL in Table 4). Intuitively, GRL plays a significant role in adapting the network to unlabeled target data. Overall, these ablation studies indicate that our proposed network with GRL and L_{KLD} leads to the best overall performance on both the source and target domains. Analysis of the impact of λ_{CE} and λ_{KLD} can be found in the supplementary material.

7 Conclusion

We propose CrackUDA, a novel two-step incremental Unsupervised Domain Adaptation (UDA) approach to address the challenging task of crack segmentation in civil structures. Our approach stands out from existing UDA methods by effectively addressing the issue of catastrophic forgetting through simultaneous learning of domain-invariant and domain-specific representations. Our experimental results demonstrate notable improvements, with 0.65 mIoU and 2.7 mIoU improvement on the source and target domains. Furthermore, we showcase the generalization capabilities of our approach across various sub-datasets of CrackSeg9K, and BuildCrack, our custom-created dataset. By providing an effective solution through incremental UDA, our work makes significant contributions to crack localization and structural health assessment in civil engineering. Additionally, our approach could serve as a benchmark to the research community focusing on unsupervised domain adaptation for semantic segmentation.

Acknowledgement. The authors acknowledge the financial support provided by IHUB, IIIT Hyderabad to carry out this research work under the project: IIIT-H/IHUB/Project/Mobility/2021-22/M2-003.

References

1. Bianchi, E., Hebdon, M.: Concrete Crack Conglomerate Dataset (2021). <https://doi.org/10.7294/16625056.v1>
2. Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: *Computer Vision – ECCV 2018*, pp. 833–851. Springer, Cham (2018)
3. Chen, M., Xue, H., Cai, D.: Domain adaptation for semantic segmentation with maximum squares loss. In: *ICCV* (2019)
4. Chen, X., Mottaghi, R., Liu, X., Fuchs, T., Yuille, A.: Unsupervised domain adaptation for object detection via back-propagation. In: *Proceedings of the European Conference on Computer Vision*, pp. 784–800 (2018)
5. Chen, Y., Li, W., Sakaridis, C., Dai, D., Van Gool, L.: Domain adaptive faster r-CNN for object detection in the wild. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3339–3348 (2018)
6. Cheng, M., Zhao, K., Guo, X., Xu, Y., Guo, J.: Joint topology-preserving and feature-refinement network for curvilinear structure segmentation. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7147–7156 (2021)
7. Choi, J., Kim, T., Kim, C.: Self-ensembling with gan-based data augmentation for domain adaptation in semantic segmentation. In: *ICCV 2019*, pp. 6829–6839 (2019). <https://doi.org/10.1109/ICCV.2019.00693>
8. Dais, D., Engin Bal, İ., Smyrou, E., Sarhosis, V.: Automatic crack classification and segmentation on masonry surfaces using convolutional neural networks and transfer learning. *Automat. Construct.* **125**, 103606 (2021)
9. Dorafshan, S., Thomas, R.J., Maguire, M.: SDNET 2018: an annotated image dataset for non-contact concrete crack detection using deep convolutional neural networks. *Data Brief* **21**, 1664–1668 (2018)
10. Dosovitskiy, A., et al.: An image is worth 16x16 words: transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929* (2020)
11. Eisenbach, M., et al.: How to get pavement distress detection ready for deep learning? a systematic approach. In: *IJCNN 2017*, pp. 2039–2047 (2017). <https://doi.org/10.1109/IJCNN.2017.7966101>
12. Ganin, Y., Lempitsky, V.: Unsupervised domain adaptation by backpropagation. In: *International Conference on Machine Learning*, pp. 1180–1189 (2015)
13. Ganin, Y., et al.: Domain-adversarial training of neural networks (2016)
14. Garg, P., Saluja, R., Balasubramanian, V.N., Arora, C., Subramanian, A., Jawahar, C.: Multi-domain incremental learning for semantic segmentation. In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 761–771 (2022)
15. Gong, R., Li, W., Chen, Y., Van Gool, L.: Dlow: domain flow for adaptation and generalization. In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2472–2481 (2019)
16. Hoffman, J., Wang, D., Yu, F., Darrell, T.: FCNs in the wild: pixel-level adversarial and constraint-based adaptation. *arXiv preprint arXiv:1612.02649* (2016)
17. Hoyer, L., Dai, D., Van Gool, L.: DAFormer: improving network architectures and training strategies for domain-adaptive semantic segmentation. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9924–9935 (2022)

18. Junior, G.S., Ferreira, J., Millán-Arias, C., Daniel, R., Junior, A.C., Fernandes, B.J.T.: Ceramic cracks segmentation with deep learning. *Appl. Sci.* **11**(13), 6017 (2021). <https://doi.org/10.3390/app11136017>
19. Khaledi, S., Ahmadi, A.: Automatic road crack detection and classification using image processing techniques, machine learning and integrated models in urban areas: a novel image binarization technique (2020)
20. Koch, C., Georgieva, K., Kasireddy, V., Akinci, B., Fieguth, P.: A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure. *Adv. Eng. Inform.* **29**(2), 196–210 (2015). <https://doi.org/10.1016/j.aei.2015.01.008>. *Infrastructure Computer Vision*
21. Kondo, Y., Ukita, N.: Crack segmentation for low-resolution images using joint learning with super-resolution. In: 2021 17th International Conference on Machine Vision and Applications (MVA), pp. 1–6. IEEE (2021)
22. König, J., Jenkins, M.D., Mannion, M., Barrie, P., Morison, G.: Weakly-supervised surface crack segmentation by generating pseudo-labels using localization with a classifier and thresholding. *IEEE Trans. Intell. Transp. Syst.* **23**(12), 24083–24094 (2022)
23. Kulkarni, S., Singh, S., Balakrishnan, D., Sharma, S., Devunuri, S., Korlapati, S.C.R.: Crackseg9k: a collection and benchmark for crack segmentation datasets and frameworks. In: *Computer Vision—ECCV 2022 Workshops: Tel Aviv, 23–27 October 2022, Proceedings, Part VII*, pp. 179–195. Springer (2023)
24. Lateef, F., Ruichek, Y.: Survey on semantic segmentation using deep learning techniques. *Neurocomputing* **338**, 321–348 (2019)
25. Lau, S.L., Chong, E.K., Yang, X., Wang, X.: Automated pavement crack segmentation using u-net-based convolutional neural network. *IEEE Access* **8**, 114892–114899 (2020)
26. Lee, D.H.: Pseudo-label : the simple and efficient semi-supervised learning method for deep neural networks. *ICML 2013 Workshop: Challenges in Representation Learning (WREPL)* (2013)
27. Li, G., Wan, J., He, S., Liu, Q., Ma, B.: Semi-supervised semantic segmentation using adversarial learning for pavement crack detection. *IEEE Access* **8**, 51446–51459 (2020). <https://doi.org/10.1109/ACCESS.2020.2980086>
28. Liu, F., Li, X., Wang, H., Cheng, J.: Domain adaptive faster r-CNN via cross-domain marginal alignment. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 12016–12025 (2020)
29. Liu, H., Miao, X., Mertz, C., Xu, C., Kong, H.: Crackformer: transformer network for fine-grained crack detection. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3783–3792 (2021)
30. Liu, Y., Yao, J., Lu, X., Xie, R., Li, L.: Deepcrack: a deep hierarchical feature learning architecture for crack segmentation. *Neurocomputing* **338**, 139–153 (2019)
31. Long, M., Zhu, H., Wang, J., Jordan, M.I.: Deep adaptation networks: a more general robustification scheme for deep learning. *IEEE Trans. Pattern Anal. Mach. Intell.* **41**(9), 1956–1970 (2018)
32. McCloskey, M., Cohen, N.J.: Catastrophic interference in connectionist networks: the sequential learning problem. *Psychology of Learning and Motivation*, vol. 24, pp. 109–165. Academic Press (1989)
33. Mei, K., Zhu, C., Zou, J., Zhang, S.: Instance adaptive self-training for unsupervised domain adaptation (2020)
34. Oliveira, H., Correia, P.L.: Road surface crack detection: improved segmentation with pixel-based refinement. In: *2017 25th European Signal Processing Conference (EUSIPCO)*, pp. 2026–2030. IEEE (2017)

35. Pan, F., Shin, I., Rameau, F., Lee, S., Kweon, I.S.: Unsupervised intra-domain adaptation for semantic segmentation through self-supervision. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2020)
36. Pei, W., Wang, Y., Vigneron, V., Wu, T.: Adversarial discriminative domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7167–7176 (2018)
37. Ramancharla, P., et al.: A Primer on Rapid Visual Screening (RVS) Consolidating Earthquake Safety Assessment Efforts in India (2020)
38. Romera, E., Alvarez, J.M., Bergasa, L., Arroyo, R.: Erfnet: efficient residual factorized convnet for real-time semantic segmentation. *IEEE Trans. Intell. Transp. Syst.* **PP**, 1–10 (2017)
39. Schwonberg, M., et al.: Survey on unsupervised domain adaptation for semantic segmentation for visual perception in automated driving. *IEEE Access* **11**, 54296–54336 (2023). <https://doi.org/10.1109/ACCESS.2023.3277785>
40. Tarvainen, A., Valpola, H.: Weight-averaged consistency targets improve semi-supervised deep learning results. arXiv preprint [arXiv:1703.01780](https://arxiv.org/abs/1703.01780) (2017)
41. Tranheden, W., Olsson, V., Pinto, J., Svensson, L.: DACS: domain adaptation via cross-domain mixed sampling. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 1379–1389 (2021)
42. Tsai, Y.H., Hung, W.C., Schuster, S., Sohn, K., Yang, M.H., Chandraker, M.: Learning to adapt structured output space for semantic segmentation. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7472–7481 (2018). <https://doi.org/10.1109/CVPR.2018.00780>
43. Tsai, Y.H., Sohn, K., Schuster, S., Chandraker, M.: Domain adaptation for structured output via discriminative patch representations. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 1456–1465 (2019)
44. Tzeng, E., Hoffman, J., Zhang, N., Saenko, K., Darrell, T.: Learning transferable features with deep adaptation networks. In: International Conference on Machine Learning, pp. 2208–2217 (2017)
45. Volker, A., Pahlavan, L., Blacquiere, G.: Crack depth profiling using guided wave angle dependent reflectivity. In: AIP Conference Proceedings, vol. 1650, pp. 785–791. American Institute of Physics (2015)
46. Vu, T.H., Jain, H., Bucher, M., Cord, M., Pérez, P.: Advent: adversarial entropy minimization for domain adaptation in semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2517–2526 (2019)
47. Wang, H., Shen, T., Zhang, W., Duan, L., Mei, T.: Classes matter: a fine-grained adversarial approach to cross-domain semantic segmentation. In: The European Conference on Computer Vision (ECCV) (2020)
48. Weng, X., Huang, Y., Li, Y., Yang, H., Yu, S.: Unsupervised domain adaptation for crack detection. *Autom. Constr.* **153**, 104939 (2023)
49. Xu, J., et al.: Unsupervised domain adaptation with adversarial residual transform networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1365–1374 (2019)
50. Yang, F., Zhang, L., Yu, S., Prokhorov, D., Mei, X., Ling, H.: Feature pyramid and hierarchical boosting network for pavement crack detection. *IEEE Trans. Intell. Transp. Syst.* **21**(4), 1525–1535 (2019)
51. Zhang, A., et al.: Automated pixel-level pavement crack detection on 3d asphalt surfaces using a deep-learning network. *Comput.-Aided Civil Infrast. Eng.* **32**(10), 805–819 (2017)

52. Zhang, P., Zhang, B., Zhang, T., Chen, D., Wang, Y., Wen, F.: Prototypical pseudo label denoising and target structure learning for domain adaptive semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12414–12424 (2021)
53. Zhang, P., Zhang, B., Zhang, T., Chen, D., Wang, Y., Wen, F.: Prototypical pseudo label denoising and target structure learning for domain adaptive semantic segmentation. arXiv preprint [arXiv:2101.10979](https://arxiv.org/abs/2101.10979) (2021)
54. Zhang, Y., Qiao, Y., Liu, C., Shen, W., Wang, X.: Domain adaptive faster r-CNN with co-attention networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 3695–3704 (2019)
55. Zou, Q., Cao, Y., Li, Q., Mao, Q., Wang, S.: Cracktree: automatic crack detection from pavement images. *Pattern Recogn. Lett.* **33**(3), 227–238 (2012)
56. Zou, Y., Yu, Z., Kumar, B.V., Wang, J.: Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 289–305 (2018)
57. Zou, Y., Yu, Z., Liu, X., Kumar, B., Wang, J.: Confidence regularized self-training. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 5982–5991 (2019)



DS MYOLO: A Reliable Object Detector Based on SSMs for Driving Scenarios

Yang Li and Jianli Xiao(✉)

University of Shanghai for Science and Technology, Shanghai 200093, China
223330831@st.usst.edu.cn, audyxiao@sjtu.edu.cn

Abstract. Accurate real-time object detection enhances the safety of advanced driver-assistance systems, making it an essential component in driving scenarios. With the rapid development of deep learning technology, CNN-based YOLO real-time object detectors have gained significant attention. However, the local focus of CNNs results in performance bottlenecks. To further enhance detector performance, researchers have introduced Transformer-based self-attention mechanisms to leverage global receptive fields, but their quadratic complexity incurs substantial computational costs. Recently, Mamba, with its linear complexity, has made significant progress through global selective scanning. Inspired by Mamba's outstanding performance, we propose a novel object detector: DS MYOLO. This detector captures global feature information through a simplified selective scanning fusion block (SimVSS Block) and effectively integrates the network's deep features. Additionally, we introduce an efficient channel attention convolution (ECACConv) that enhances cross-channel feature interaction while maintaining low computational complexity. Extensive experiments on the CCTSDB 2021 and VLD-45 driving scenarios datasets demonstrate that DS MYOLO exhibits significant potential and competitive advantage among similarly scaled YOLO series real-time object detectors.

Keywords: Driving Scenarios · Object Detection · SSM · YOLO

1 Introduction

In recent years, the rapid development of deep learning has continuously injected new energy into the field of object detection. In autonomous driving scenarios, real-time detection and accurate identification of traffic signs and vehicle identities are crucial for enhancing the safety of driving systems [1]. However, in driving scenarios, targets often vary significantly in scale and size, leading to poor visual features and susceptibility to noise interference. This makes object detection one of the most challenging tasks in autonomous driving. CNNs, with their parameter sharing and optimized hardware acceleration, have made significant progress in real-time object detectors. However, their local focus makes it difficult to effectively capture targets of different scales in driving scenarios,

limiting their performance. Therefore, developing a high-performance real-time object detector is an important and meaningful endeavor.

In the past, general object detection paradigms primarily focused on CNN-based two-stage detection networks, such as Faster R-CNN [2], Mask R-CNN [3], and Cascade R-CNN [4]. However, the pre-generation of candidate region proposals in two-stage detectors often results in inadequate real-time performance. Recently, research in object detection has increasingly shifted towards end-to-end single-stage detection algorithms, such as YOLO [5], SSD [6], CornerNet [7], and FCOS [8]. Single-stage detection models feature simpler architectures, with the YOLO series models, in particular, achieving a commendable balance between speed and accuracy. This has garnered significant attention from both the academic and industrial communities.

The YOLO networks, especially from YOLOv3 [9] onwards, typically consist of three main structures: backbone, neck, and head. The backbone extracts deep features from input images. For instance, YOLOv3, YOLOX [10], YOLOv7 [11], and YOLOv8 [12] use Darknet-53 [9], while YOLOv4 [13] and YOLOv5 [14] use CSPDarknet-53 [13]. YOLOv6 [15] employs EfficientRep [15], and YOLOv9 [16] uses the lightweight GELAN. The neck structure fuses multi-scale features to enhance multi-scale representation capabilities. SPPELAN [16] optimizes multi-scale feature extraction efficiency, and PAN [17] enhances feature fusion based on FPN [18]. The head structure decodes the features from the neck to generate final detection results, evolving from anchor-based (e.g., YOLOv5 [14], YOLOv7 [11]) to more efficient anchor-free (e.g., YOLOv6 [15], YOLOv8 [12], YOLOv9 [16]) and NMS-free (YOLOv10 [19]) designs.

Object detectors based on the Transformer encoder-decoder architecture, such as the DETR [20] series, leverage the global feature modeling capabilities of the self-attention mechanism to achieve performance comparable to state-of-the-art detectors. However, the quadratic computational complexity poses challenges in balancing speed and accuracy. Inspired by the effectiveness of attention mechanisms, channel attention mechanisms based on CNNs, such as SE [21], ECA [22], and their variants [23, 24], have also demonstrated significant gains. Recent research has shown that methods based on State Space Models (SSMs), such as Mamba [25, 26], have achieved remarkable success in visual tasks due to their powerful global modeling capabilities and linear complexity advantages [27–29].

Inspired by previous works, we propose a novel object detector named DS MYOLO. This detector integrates a Simplified Volitional Scan Fusion Block (SimVSS Block) to achieve deep global feature fusion, and introduces an Efficient Convolutional Operator (ECAConv) to address the shortcomings of the Standard Convolution(SC) in cross-channel interactions. We validate the superiority of DS MYOLO on the publicly available CCTSDB 2021 [30] traffic sign dataset and the VLD-45 [31] vehicle logo dataset. Experimental results demonstrate that DS MYOLO exhibits strong competitiveness among state-of-the-art detectors of similar scale. In summary, our contributions can be outlined as follows:

- 1) To further enhance detection performance through feature fusion, we design a Simplified Volitional Scan Fusion Block (SimVSS Block) to achieve deep

global feature fusion. This block consists of a State Space Model (SSM) in series with a feedforward network, enhanced by residual connections, effectively integrating global and local features.

- 2) We propose an Efficient Channel Attention Convolutional Operator (ECAConv). By decoupling the channels post-convolution and performing cross-channel attention interactions, ECAConv significantly establishes dependencies between channels and enhances representation, while maintaining computational complexity similar to SC.
- 3) We further design different scales of DS MYOLO (-N/-S/-M) real-time object detectors based on the proposed SimVSS Block and ECAConv. On the CCTSDB 2021 [30] and VLD-45 [31] traffic scene datasets, DS MYOLO demonstrates robust competitiveness compared to existing state-of-the-art real-time object detectors.

2 Related Works

2.1 Real-Time Object Detectors

With the rapid development of autonomous driving, developing real-time and efficient object detectors is crucial for real-world applications. To balance speed and accuracy, researchers have dedicated significant time and effort to developing efficient object detectors. Among these, the YOLO series models have garnered widespread attention due to their simple structure and end-to-end detection characteristics. Starting from the initial YOLOv3 [9], the architectural design of backbone-neck-head networks has been a key factor in enhancing model performance. YOLOv4 [13], based on CSPNet [32], optimized the previously used DarkNet backbone structure [9] and introduced a series of data augmentation methods [13, 33]. YOLOv5 [14] incorporated strategies such as adaptive anchor box computation and automated learning rate adjustment. YOLO-X [10] employed a label assignment strategy (SimOTA) and introduced a decoupled head to further improve training efficiency and detection performance. YOLOv6 [15] integrated re-parameterization methods into the YOLO architecture to balance accuracy and speed. YOLOv7 [11] introduced the Extended Efficient Layer Aggregation Network (E-ELAN) as the backbone to further enhance performance. YOLOv8 [12] focused on analyzing the shortcomings of previous YOLO models and achieved higher performance by integrating their strengths. Gold-YOLO [34] proposed the GD mechanism to improve multi-scale object fusion performance. YOLOv9 [16] introduced the GELAN backbone and enhanced the model’s expressive capabilities through PGI. YOLOv10 [19] proposed a dual-label assignment strategy without NMS, improving the overall efficiency of the model.

2.2 Transformer-Base Object Detection

Transformers [35], with their self-attention mechanism, excel in addressing long-range dependency issues. DETR [20] was the first to apply the Transformer

architecture to object detection, simplifying the pipeline by eliminating manually designed anchor boxes and NMS components, garnering significant attention. However, DETR’s training convergence remains inefficient. Subsequently, Deformable-DETR [36] improved upon DETR by combining deformable convolutions with self-attention calculations, effectively accelerating convergence. Conditional DETR [37] introduced the Conditional Cross-Attention mechanism to expedite DETR’s training. DAB-DETR [38] utilized dynamic anchor boxes directly as queries in the Transformer decoder, enhancing training speed and inference performance. Anchor DETR [39] incorporated anchor-based query design and Row-Column Decoupled Attention (RCDA), achieving comparable performance to DETR while improving efficiency. DN-DETR [40] introduced a query-denoising training method to accelerate DETR’s training process and further enhance performance. Group DETR [41] employed a group-based training strategy with one-to-many assignments to increase training efficiency. RT-DETR [42] proposed an efficient hybrid encoder architecture by separating intra-scale interactions and cross-scale fusion, further improving model efficiency and accuracy. Rank-DETR [43] introduced a rank-oriented architecture design, significantly boosting inference precision.

2.3 SSMS-Based Vision State Space Model

Recently, Mamba [25, 26] has garnered significant attention for its linear complexity in addressing long-range dependency problems. Subsequently, Vision Mamba [27] was the first to apply the SSM to visual backbone networks, achieving performance comparable to, or even surpassing, Vision Transformers (ViT). VMamba [44] introduced the Cross-Scan Module (CSM) to capture the global receptive field, enhancing visual representation with linear computational complexity. LocalMamba [45] proposed a local scanning strategy to strengthen feature dependencies within local windows while maintaining a global perspective. EfficientVMamba [29] combined efficient selective scanning with convolution in the backbone, achieving a balance between accuracy and efficiency. MambaOut [46] explored the necessity of SSM in visual tasks, experimentally validating SSM’s higher value for tasks with long sequences and autoregressive characteristics, and providing foundational support for downstream tasks like segmentation. MSV-Mamba [47] introduced a multi-scale scanning mechanism, enhancing the ability to learn dependencies across different resolutions. Inspired by Mamba’s outstanding contributions to various visual tasks, we integrated the SSM module into our network’s feature fusion, achieving significant performance enhancement.

3 Method

3.1 Overall Architecture of DS MYOLO

The overall architecture of DS MYOLO is illustrated in Fig. 1. In the backbone network, the Stem is composed of SC, batch normalization, and a SiLU acti-

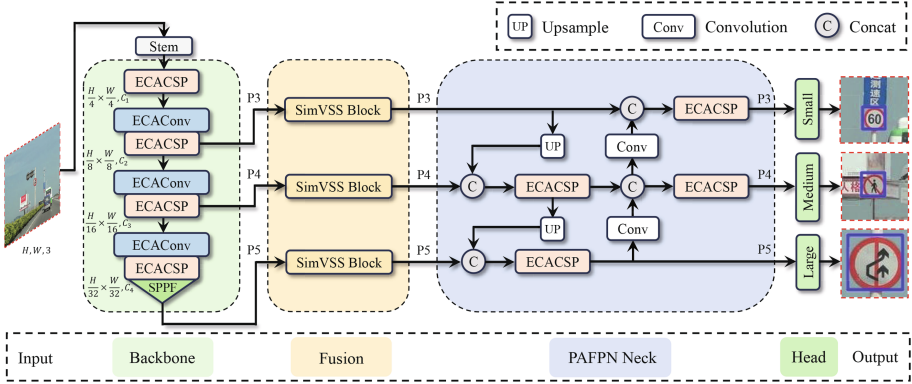


Fig. 1. Overall architecture of DS MYOLO.

vation function, stacked sequentially and downsampled twice, resulting in a 2D feature map with dimensions $(\frac{H}{4}, \frac{W}{4})$, and C_i channels. To effectively extract rich features in the backbone network, ECACConv is used for downsampling with a stride of 2, and ECACSP is employed to further extract abundant local features. Our object detection model introduces a fusion layer before the neck network. This fusion layer uses three SimVSS Blocks to achieve deep integration of feature layers $\{P_3, P_4, P_5\}$ while maintaining low computational complexity. In the neck, we follow the PAFPN [12] approach, using 3×3 SC for downsampling with a stride of 2 and further integrating local features through ECACSP. We adopt a practical decoupled head and NMS-free design [19], which effectively decodes small, medium, and large targets in the input, enabling efficient detection across different scales.

3.2 Fusion Layer Based on SimVSS Block

The traditional YOLO model transmits features extracted by the backbone network directly to the neck network for feature communication. While this method effectively enhances the salience of local features, it overlooks the feature dependencies within the global receptive field. Previous research has demonstrated that increasing the receptive field can beneficially enhance model performance. Given the larger feature map size of shallow networks, we employ a simplified SimVSS Block based on SSM to process the output features of the backbone network. The fused global features are then subjected to nonlinear transformations through a forward network to improve the model’s fitting capacity.

The structure of SimVSS Block is illustrated in Fig. 2. The primary design is based on the SSM and a feedforward network, with residual connections and normalization layers included to stabilize gradient training and accelerate model convergence. A traditional SSM can be viewed as a linear time-invariant system function that maps a univariate sequence $x(t) \in \mathbb{R}$ to an output sequence $y(t) \in \mathbb{R}$ via an intermediate hidden state $h(t) \in \mathbb{R}^N$. Given the state transition matrix

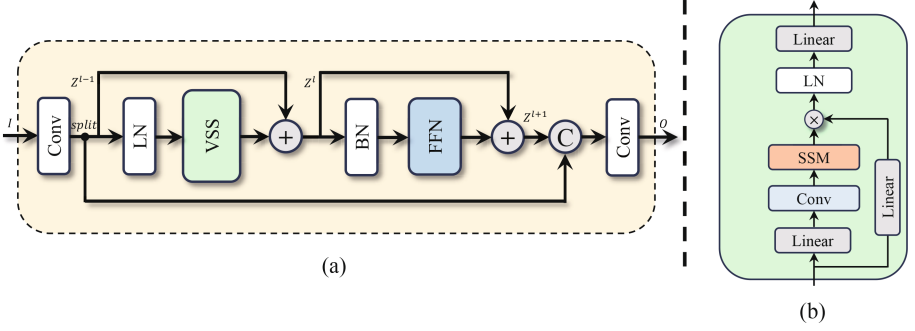


Fig. 2. Detailed structure of SimVSS Block ((a) represents component modules of SimVSS Block, (b) represents key internal architecture of VSS module).

$A \in \mathbb{R}^{N \times N}$ as the evolution factor, the weight matrix and the observation matrix $B, P \in \mathbb{C}^N$ as projection factors respectively, and the skip connection defined as $Q \in \mathbb{C}^1$, the mathematical formulation is as follows:

$$h'(t) = Ah(t) + Bx(t) \quad (1)$$

$$y(t) = Ph(t) + Qx(t) \quad (2)$$

Moreover, the system function can be discretized for handling discrete-time sequence data by incorporating a time scale parameter $\Delta \in \mathbb{R}^Q$. This transformation can be defined as follows:

$$\begin{cases} h_t = \bar{A}h_{k-2} + \bar{B}x_k \\ y_t = Ph_k + Qx_k \\ \bar{A} = e^{\Delta A} \\ \bar{B} = (e^{\Delta A} - \mu)A^{-1}B \\ \bar{P} = P \end{cases} \quad (3)$$

where $B, P \in \mathbb{R}^{D \times N}$, To refine the approximation of B using a first-order Taylor series expansion:

$$\bar{B} = (e^{\Delta A} - \mu)A^{-1}B \approx (\Delta A)(\Delta A)^{-1}\Delta B = \Delta B \quad (4)$$

For the input $I \in \mathbb{R}^{H \times W \times C}$, the processing steps within the SimVSS Block can be described as follows:

$$Z^{l-1} = \text{split}\{\text{SiLU}(\text{BN}(\text{Conv}_{1 \times 1}(I)))\} \quad (5)$$

$$Z^l = \text{VSS}(\text{LN}(Z^{l-1})) + Z^{l-1} \quad (6)$$

$$Z^{l+1} = Z^l + \text{FFN}(\text{BN}(Z^l)) \quad (7)$$

where Z^{l-1} , Z^l and Z^{l+1} represent the output states of the input I at different layers l of the SimVSS Block. The Feedforward Network (FFN) consists of two 1×1 SC and a SiLU non-linear activation function.

3.3 ECACConv and ECACSP Module

Previous studies [21,22] have shown that standard convolutions lack attention to channel salience. Inspired by ECA [22], we propose a novel Efficient Channel Attention Convolution (ECACConv), as illustrated in Fig. 3. Specifically, we perform adaptive channel peeling after standard convolution and aggregate salient features through global pooling. Then, we use a one-dimensional convolution with adaptive kernels to quickly map salient features and generate weights. These weights are applied to the corresponding channels and enhance salient feature expression via element-wise multiplication. Finally, the weighted channels are merged with the unweighted channels, and a Shuffle operation is employed to reorganize the channels, facilitating inter-channel information exchange and enhancing feature representation diversity.

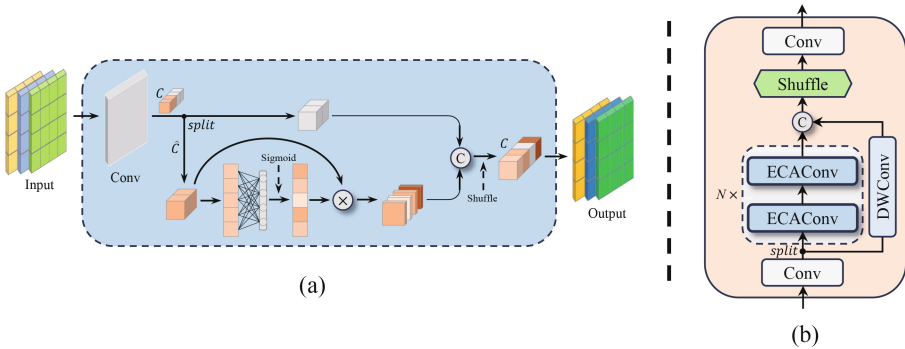


Fig. 3. Key architectures and components of ECACConv and ECACSP ((a) Basic architecture of ECACConv, (b) Detailed structure of ECACSP).

It is noteworthy that as the number of channels C increases, capturing more effective features and establishing channel correlations become critically important. Therefore, we have designed an adaptive channel allocation strategy to ensure the effective interaction range of features. Specifically, for a given extended linear function $\phi(k) = \gamma \times k - b$, when the number of channels $C \in 2^n$ (where n is a positive integer), the mapping relationship between the adaptive convolution kernel and the channels can be defined as:

$$C = \phi(k) = 2^{(\gamma \times k - b)} \tag{8}$$

Furthermore, the channel stripping ratio $\sigma \in (0, 1]$ can be expressed as follows:

$$\sigma = \min(1, \max(0.1, \frac{\log_2(C)}{10})) \tag{9}$$

In practice, by focusing on channel $\hat{C} = \sigma \times C$ as the object of channel attention, and setting the parameters γ and b to 2 and 1 respectively, the mapping relationship between the adaptive convolution kernel of the one-dimensional convolution

and the target channel can be defined as:

$$k = \left\lceil \frac{\log_2(\hat{C})}{\gamma} + \frac{b}{\gamma} \right\rceil = \left\lceil \frac{\log_2(\hat{C}) + 1}{2} \right\rceil \quad (10)$$

Clearly, as the channel stripping ratio expands, the higher-dimensional channels possess a larger receptive field, while the lower-dimensional channels establish a non-linear mapping to capture the local channel correlations. In this work, we set σ to 0.5 and k to 3.

Furthermore, we designed a lightweight feature extraction module named ECACSP, whose architecture is illustrated in Fig. 3(b). Specifically, ECACSP adjusts the dimensions through a 1×1 SC and applies two 3×3 ECACConv layers for deep feature extraction. These deep features are then merged with the input features processed by depthwise separable convolution, followed by a Shuffle operation to achieve inter-channel feature interaction. In the backbone network, we use ECACConv for downsampling and employ ECACSP to extract rich information from the feature maps.

4 Experiments

4.1 Setups

Dataset: We conducted extensive experiments on the publicly available traffic sign detection dataset CCTSDB 2021 [30] and the vehicle logo detection dataset VLD-45 [31] to validate the effectiveness of the proposed object detector. Notably, the CCTSDB 2021 dataset includes three categories, each consisting of multi-scale targets from real traffic scenes under different lighting conditions. The VLD-45 dataset comprises 45 categories of large vehicle logos collected from the internet using web crawlers. To ensure a fair comparison, we followed the dataset division methods provided in CCTSDB 2021 and VLD-45.

Implementation Details: We conducted experiments using a single NVIDIA 4090 GPU within the PyTorch framework. All experiments were trained from scratch for 200 epochs without using pre-trained weights, with a 3-epoch warm-up period. We used the SGD optimizer, setting the initial learning rate to decrease from 0.01 to 0.0001 and the momentum to 0.937. The input size was fixed at 640×640 , and the batch size was set to 16. Our data augmentation strategies included random scaling, translation, and Mosaic [13], with Mosaic data augmentation being disabled during the last 10 epochs.

4.2 Comparison with State-of-the-Arts

In this section, we compare the proposed DS MYOLO with other latest state-of-the-art real-time detectors in the YOLO series, including YOLOv5 [14], YOLOv6 [15], YOLOv7 [11], YOLOv8 [12], Gold YOLO [34], YOLOv9 [16], and YOLOv10 [19]. We primarily measure model parameters(M), FLOPs(G), mAP(%), detection box precision, and recall rate.

Table 1. Comparison with state-of-the-art real-time object detectors from the YOLO series on the CCTSDB 2021 [30] test set.

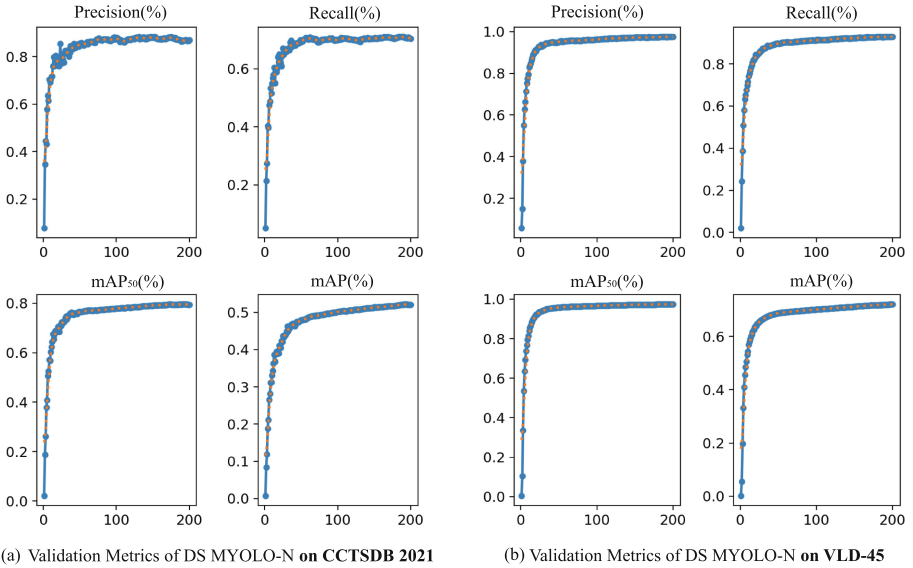
Method	#Params. (M)	FLOPs (G)	mAP _{50:95} (%)	mAP ₅₀ (%)	mAP ₇₅ (%)	P (%)	R (%)
YOLOv5-N [14]	2.5	7.2	47.31	75.39	54.4	86.1	67.6
YOLOv6-N [15]	4.2	11.8	47.05	74.91	54.17	85.8	67.9
YOLOv7-Tiny [11]	6	13.2	48.61	76.43	55.85	86.5	68.4
YOLOv8-N [12]	3	8.1	49.72	78.66	57	88.1	71
Gold YOLO-N [34]	5.6	12.1	49.98	79.05	57.1	87.5	71.3
YOLOv10-N [19]	2.3	6.5	51.37	79.36	60.81	87.9	72
DS MYOLO-N (Ours)	4	9	52.22	79.63	62.02	88.1	71.1
YOLOv5-S [14]	9.1	23.8	53.21	82.15	61.87	88.6	72.7
YOLOv6-S [15]	16.3	44	51.9	80.44	59.87	86.9	73.2
YOLOv8-S [12]	11.1	28.5	54.35	82.52	64.73	89.6	75
Gold YOLO-S [34]	21.5	46	54.17	82.33	64.29	89.1	75.1
YOLOv10-S [19]	7.2	21.4	55.2	82.55	65.34	89.1	75.6
DS MYOLO-S (Ours)	14.8	31.4	55.78	80.98	66.13	89.7	73.5
YOLOv5-M [14]	25	64.1	55.63	83.56	65.57	88	76.4
YOLOv6-M [15]	32.8	81.4	53.36	81.89	62.44	88.5	74.7
YOLOv7 [11]	36.5	104.3	56.12	83.77	66.48	88.1	75.3
YOLOv8-M [12]	25.9	78.7	56.97	84.85	67.11	87.7	78.6
Gold YOLO-M [34]	41.3	87.3	56.22	83.81	67.16	89.2	76.2
YOLOv9-C [16]	25.3	102.3	57.85	84.72	68.87	89.3	77
YOLOv10-M [19]	15.3	58.9	56.36	83.35	67.22	89.3	76.7
DS MYOLO-M (Ours)	30.7	82.7	58.35	85.11	69.83	91	75.4

As shown in Table 1, we compared different versions of DS MYOLO (-N/-S/-M) with the latest YOLO series real-time detectors on CCTSDB 2021. Overall, DS MYOLO models excelled in multiple metrics. In the lightweight models, DS MYOLO-N achieved a 52.22% mAP with 4M parameters and 9G FLOPs, outperforming similar models like YOLOv5-N [14], YOLOv6-N [15], YOLOv7-Tiny [11], and surpassing the latest Gold YOLO-N [34] (49.98%) and YOLOv10-N [19] (51.37%). With the increase of the channel scaling factor, DS MYOLO showed further performance improvement, with DS MYOLO-S and DS MYOLO-M increasing mAP by 0.58% and 0.5%, respectively. Notably, the introduced SimVSS Block significantly improved the precision of detection boxes, achieving 88.1%, 89.7%, and 91%, respectively, surpassing all versions of state-of-the-art real-time detectors.

On the VLD-45 dataset, we performed a similar comparison of DS MYOLO with lightweight models of different YOLO variants. As shown in Table 2, several lightweight models achieved over 95% detection accuracy. In terms of mAP, our DS MYOLO achieved the highest mAP, mAP₅₀, and mAP₇₅. Regarding detec-

Table 2. Comparison with state-of-the-art real-time object detectors from the YOLO series on the VLD-45 [31] test set.

Method	#Params. (M)	FLOPs (G)	mAP _{50:75} (%)	mAP ₅₀ (%)	mAP ₇₅ (%)	P (%)	R (%)
YOLOv5 [14]	2.5	7.2	69.08	94.86	85.2	95.4	90.5
YOLOv6 [15]	4.2	11.8	68.15	94.3	84.75	95.1	89.6
YOLOv7 [11]	6	13.2	69.66	95.77	85.81	96.5	91.4
YOLOv8 [12]	3	8.1	70.71	96.25	87.59	96.8	91.8
Gold YOLO [34]	5.6	12.1	70.83	96.6	87.19	96.6	92.2
YOLOv10 [19]	2.3	6.5	71.4	96.52	88.31	97.1	92.7
DS MYOLO (Ours)	4	9	72.3	97.59	89.51	97.7	93.2

**Fig. 4.** Trends in validation metrics for DS MYOLO-N across epochs ((a) results on CCTSDB 2021 [30], (b) results on VLD-45 [31]).

tion accuracy and recall rate, DS MYOLO demonstrated the best performance, reaching 97.7% and 93.2%, respectively. Overall, DS MYOLO significantly outperformed others in terms of overall performance. Our model excelled in key metrics such as mAP and mAP₇₅ and also surpassed the current state-of-the-art YOLO models in both detection accuracy and recall rate.

Figure 4 shows the trends in validation metrics for our DS MYOLO on the CCTSDB 2021 [30] and VLD-45 [31] datasets as epochs progress. It can be observed that the DS MYOLO models exhibit high accuracy and stable detection capabilities across different datasets and model scales. Specifically, on CCTSDB

2021, the detection accuracy and recall rate of DS MYOLO rapidly increase within the first 50 epochs and then continue to improve steadily, with mAP consistently trending upwards. On VLD-45, DS MYOLO maintained considerable stability and significant performance, converging as epochs increased to their maximum.

4.3 Ablation Studies

In this section, we perform a series of ablation studies on the proposed DS MYOLO using the CCTSDB 2021 dataset. To further validate the effectiveness of DS MYOLO, we take DS MYOLO-N as an example and independently examine each of its major modules, focusing on Params (M), FLOPs (G), and mAP (%). To facilitate observation of the impact of each module on the overall model performance, all models are trained for 80 epochs to amplify the differences.

Table 3. Ablation study results of DS MYOLO on CCTSDB 2021 [30].

#	ECACConv	SimVSS Block	ECACSP	#Params.(M)	FLOPs(G)	mAP(%)
1				2.7	6.8	46.53
2	✓			2.7	6.8	47.67
3		✓		4	8.9	48.7
4			✓	2.7	6.9	48.21
5	✓	✓		4	8.9	49.08
6	✓	✓	✓	4	9	49.35

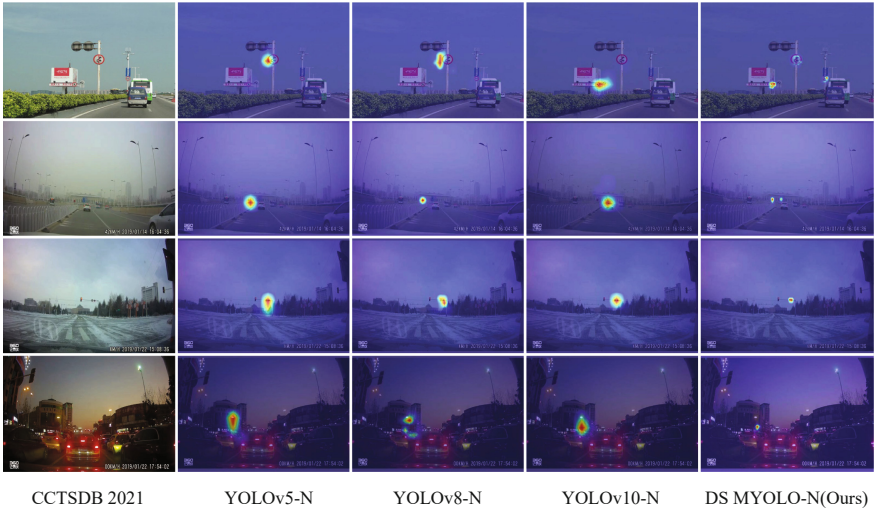
As shown in Table 3, ECACConv significantly improved the mAP by 1.14% with similar parameter and computational costs, demonstrating the enhancement of model performance through the incorporation of local inter-channel dependencies. The addition of the SSM-based fusion layer in the SimVSS Block further boosted model performance by 2.17%, albeit with an increase of 1.3M parameters and 2.1G FLOPs, highlighting its effectiveness. The introduced ECACSP improved model performance by 1.68% while maintaining nearly the same level of model complexity. When both ECACConv and SimVSS Block were incorporated, there was a slight increase in parameters and computational cost, but the mAP reached 49.08%. The subsequent inclusion of ECACSP resulted in an additional mAP improvement of 0.27%. Overall, the integration of these modules into DS MYOLO significantly enhanced object detection performance with relatively low computational cost. Additionally, we conducted an ablation study on the performance of ECACConv compared to other downsampling operators on YOLOv8 [12], as shown in Table 4.

Table 4. Ablation study results of ECACConv and other downsampling operators on CCTSDB 2021 [30].

Downsampling	#Params.(M)	FLOPs(G)	mAP _{50:75} (%)	mAP ₅₀ (%)	mAP ₇₅ (%)
Conv [12]	3.0	8.1	45.31	73.29	51.17
GhostConv [48]	2.8	7.8	45.07	74.55	50.31
GSCConv [49]	2.8	7.8	45.22	73.43	51.5
Waveletpool [50]	2.7	7.5	45.74	73.92	51.82
SPDCConv [51]	4.2	10.2	46.25	74.66	52.49
ADown [16]	2.7	7.6	44.8	73.17	48.73
SCDown [19]	2.7	7.7	45.92	74.17	51.65
ECACConv	3.1	8.2	46.33	75.05	52.87

5 CAM Visualization

Figure 5 shows the CAM visualization results for YOLOv5 [14], YOLOv8 [12], YOLOv10 [19], and our DS MYOLO on CCTSDB 2021 [30]. It can be observed that our model accurately detects target locations and assigns higher weights to the detection areas. Additionally, our DS MYOLO is capable of focusing on targets at different scales, thereby reducing the false detection rate.

**Fig. 5.** CAM visualization results for YOLOv5 [14], YOLOv8 [12], YOLOv10 [19], and our DS MYOLO-N on CCTSDB 2021 [30].

6 Conclusions

In this paper, we propose a novel high-performance object detector for driving scenarios, named DS MYOLO. The designed SimVSS Block effectively enhances feature fusion in deep networks. Additionally, the proposed Efficient Channel Attention Convolution (ECAConv) significantly boosts cross-channel feature interactions. Extensive experiments conducted on the CCTSDB 2021 traffic sign dataset and the VLD-45 vehicle logo dataset demonstrate that our DS MYOLO achieves the highest performance among YOLO series real-time object detectors of comparable scale and exhibits strong competitiveness.

Acknowledgements. This work is supported by China NSFC Program under Grant NO. 61603257.

References

1. Cheng, G., et al.: Towards large-scale small object detection: survey and benchmarks. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2023)
2. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* **28** (2015)
3. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2961–2969 (2017)
4. Cai, Z., Vasconcelos, N.: Cascade r-cnn: delving into high quality object detection. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6154–6162 (2018)
5. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: unified, real-time object detection. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779–788 (2016)
6. Liu, W., et al.: SSD: single shot multibox detector. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) *ECCV 2016*. LNCS, vol. 9905, pp. 21–37. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_2
7. Law, H., Deng, J.: Cornernet: detecting objects as paired keypoints. In: *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 734–750 (2018)
8. Tian, Z., Shen, C., Chen, H., He, T.: FCOS: a simple and strong anchor-free object detector. *IEEE Trans. Pattern Anal. Mach. Intell.* **44**(4), 1922–1933 (2020)
9. Redmon, J., Farhadi, A.: Yolov3: an incremental improvement. *arXiv preprint arXiv:1804.02767* (2018)
10. Ge, Z., Liu, S., Wang, F., Li, Z., Sun, J.: Yolox: exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430* (2021)
11. Wang, C.Y., Bochkovskiy, A., Liao, H.Y.M.: Yolov7: trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7464–7475 (2023)
12. Jocher, G.: Ultralytics yolov8 (2023). <https://github.com/ultralytics/ultralytics>
13. Bochkovskiy, A., Wang, C.Y., Liao, H.Y.M.: Yolov4: optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934* (2020)
14. Jocher, G.: Yolov5 release v6.1 (2022). <https://github.com/ultralytics/yolov5/releases/tag/v6.1>

15. Li, C., et al.: Yolov6: a single-stage object detection framework for industrial applications. arXiv preprint [arXiv:2209.02976](https://arxiv.org/abs/2209.02976) (2022)
16. Wang, C.Y., Yeh, I.H., Liao, H.Y.M.: Yolov9: learning what you want to learn using programmable gradient information. arXiv preprint [arXiv:2402.13616](https://arxiv.org/abs/2402.13616) (2024)
17. He, K., Zhang, X., Ren, S., Sun, J.: Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **37**(9), 1904–1916 (2015)
18. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2117–2125 (2017)
19. Wang, A., et al.: Yolov10: real-time end-to-end object detection. arXiv preprint [arXiv:2405.14458](https://arxiv.org/abs/2405.14458) (2024)
20. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: *European Conference on Computer Vision*, pp. 213–229. Springer (2020)
21. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7132–7141 (2018)
22. Wang, Q., Wu, B., Zhu, P., Li, P., Zuo, W., Hu, Q.: Eca-net: efficient channel attention for deep convolutional neural networks. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11534–11542 (2020)
23. Fu, J., Liu, J., Tian, H., Li, Y., Bao, Y., Fang, Z., Lu, H.: Dual attention network for scene segmentation. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3146–3154 (2019)
24. Hou, Q., Zhou, D., Feng, J.: Coordinate attention for efficient mobile network design. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13713–13722 (2021)
25. Gu, A., Dao, T.: Mamba: linear-time sequence modeling with selective state spaces. arXiv preprint [arXiv:2312.00752](https://arxiv.org/abs/2312.00752) (2023)
26. Dao, T., Gu, A.: Transformers are SSMS: generalized models and efficient algorithms through structured state space duality. arXiv preprint [arXiv:2405.21060](https://arxiv.org/abs/2405.21060) (2024)
27. Zhu, L., Liao, B., Zhang, Q., Wang, X., Liu, W., Wang, X.: Vision mamba: efficient visual representation learning with bidirectional state space model. arXiv preprint [arXiv:2401.09417](https://arxiv.org/abs/2401.09417) (2024)
28. Wang, Z., Zheng, J.Q., Zhang, Y., Cui, G., Li, L.: Mamba-unet: unet-like pure visual mamba for medical image segmentation. arXiv preprint [arXiv:2402.05079](https://arxiv.org/abs/2402.05079) (2024)
29. Pei, X., Huang, T., Xu, C.: Efficientvmamba: atrous selective scan for light weight visual mamba. arXiv preprint [arXiv:2403.09977](https://arxiv.org/abs/2403.09977) (2024)
30. Zhang, J., Zou, X., Kuang, L.D., Wang, J., Sherratt, R.S., Yu, X.: Ctsdb 2021: a more comprehensive traffic sign detection benchmark. *Human-centric Comput. Inf. Sci.* **12** (2022)
31. Yang, S., Bo, C., Zhang, J., Gao, P., Li, Y., Serikawa, S.: Vld-45: a big dataset for vehicle logo recognition and detection. *IEEE Trans. Intell. Transp. Syst.* **23**(12), 25567–25573 (2021)
32. Wang, C.Y., Liao, H.Y.M., Wu, Y.H., Chen, P.Y., Hsieh, J.W., Yeh, I.H.: Cspnet: a new backbone that can enhance learning capability of CNN. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 390–391 (2020)

33. Yun, S., Han, D., Oh, S.J., Chun, S., Choe, J., Yoo, Y.: Cutmix: regularization strategy to train strong classifiers with localizable features. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 6023–6032 (2019)
34. Wang, C., et al.: Gold-yolo: efficient object detector via gather-and-distribute mechanism. *Adv. Neural Inf. Process. Syst.* **36** (2024)
35. Vaswani, A., et al.: Attention is all you need. *Adv. Neural Inf. Process. Syst.* **30** (2017)
36. Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J.: Deformable detr: deformable transformers for end-to-end object detection. arXiv preprint [arXiv:2010.04159](https://arxiv.org/abs/2010.04159) (2020)
37. Meng, D., et al.: Conditional detr for fast training convergence. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 3651–3660 (2021)
38. Liu, S., et al.: Dab-detr: dynamic anchor boxes are better queries for detr. arXiv preprint [arXiv:2201.12329](https://arxiv.org/abs/2201.12329) (2022)
39. Wang, Y., Zhang, X., Yang, T., Sun, J.: Anchor detr: query design for transformer-based detector. *Proc. AAAI Conf. Artif. Intell.* **36**, 2567–2575 (2022)
40. Li, F., Zhang, H., Liu, S., Guo, J., Ni, L.M., Zhang, L.: Dn-detr: accelerate detr training by introducing query denoising. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 13619–13627 (2022)
41. Chen, Q., et al.: Group detr: fast detr training with group-wise one-to-many assignment. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 6633–6642 (2023)
42. Zhao, Y., et al.: Dets beat yolos on real-time object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 16965–16974 (2024)
43. Pu, Y., et al.: Rank-detr for high quality object detection. *Adv. Neural Inf. Process. Syst.* **36** (2024)
44. Liu, Y., et al.: Vmamba: visual state space model. arXiv preprint [arXiv:2401.10166](https://arxiv.org/abs/2401.10166) (2024)
45. Huang, T., Pei, X., You, S., Wang, F., Qian, C., Xu, C.: Localmamba: visual state space model with windowed selective scan. arXiv preprint [arXiv:2403.09338](https://arxiv.org/abs/2403.09338) (2024)
46. Yu, W., Wang, X.: Mambaout: do we really need mamba for vision? arXiv preprint [arXiv:2405.07992](https://arxiv.org/abs/2405.07992) (2024)
47. Shi, Y., Dong, M., Xu, C.: Multi-scale vmamba: hierarchy in hierarchy visual state space model. arXiv preprint [arXiv:2405.14174](https://arxiv.org/abs/2405.14174) (2024)
48. Han, K., Wang, Y., Tian, Q., Guo, J., Xu, C., Xu, C.: Ghostnet: more features from cheap operations. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1580–1589 (2020)
49. Li, H., Li, J., Wei, H., Liu, Z., Zhan, Z., Ren, Q.: Slim-neck by gconv: a better design paradigm of detector architectures for autonomous vehicles. arXiv preprint [arXiv:2206.02424](https://arxiv.org/abs/2206.02424) (2022)
50. Williams, T., Li, R.: Wavelet pooling for convolutional neural networks. In: International Conference on Learning Representations (2018)
51. Sunkara, R., Luo, T.: No more strided convolutions or pooling: a new CNN building block for low-resolution images and small objects. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pp. 443–459. Springer (2022)



Robust Single-Cam Surround View Object Detection and Localization Using Memory Maps

Yitong Quan¹(✉), Benjamin Kiefer¹, Martin Messmer¹, Charan Ram Akupati²,
Rainer Graser², and Andreas Zell¹

¹ Faculty of Computer Science, University of Tuebingen, Tuebingen, Germany
yitong.quan@uni-tuebingen.de

² InMach Intelligente Maschinen GmbH, Ulm, Germany

Abstract. This paper presents a cost-effective approach for geo-localization and location-aware object detection using a single 360° fish-eye camera lens on mobile platforms such as street cleaning vehicles. We propose a system that captures a comprehensive view of the surroundings and accurately detects people and objects. Using the camera's geometry, the system infers distances to objects on the ground and projects them into global coordinates, creating a temporal spatial map. This 'memory map' is continuously updated, allowing for the accumulation of detection predictions over time. This approach significantly enhances the robustness and accuracy of object detection in dynamic environments. Our experiments demonstrate the system's efficacy, making it a strong candidate for implementation in various real-world applications requiring enhanced situational awareness and autonomous decision-making capabilities.

Keywords: Object Detection · Localization · 360 footage · Robotics

1 Introduction

Accurately perceiving and interpreting the surrounding environment is crucial, especially for autonomous vehicles operating in dynamic settings. Traditional object detection systems frequently encounter difficulties due to their restricted field of view, limiting their effectiveness in monitoring dynamic changes across entire environments. This paper introduces a new approach that addresses these challenges by integrating a single 360° camera with advanced object detectors on mobile platforms, such as street cleaning vehicles.

Our motivation stems from the critical need for enhanced situational awareness in these vehicles. In agricultural and urban maintenance applications, the ability to detect people and objects reliably in real-time is not just a matter

This work has been supported by the German Ministry for Economic Affairs and Climate Action, Project SafeAI, FKZ: 19A21009C.

of efficiency, but of safety and operational effectiveness. The dynamic nature of these environments, coupled with the diverse range of dynamic obstacles encountered, poses a unique set of challenges. Current methods often fall short in providing a comprehensive and accurate detection necessary for these applications [1, 2].

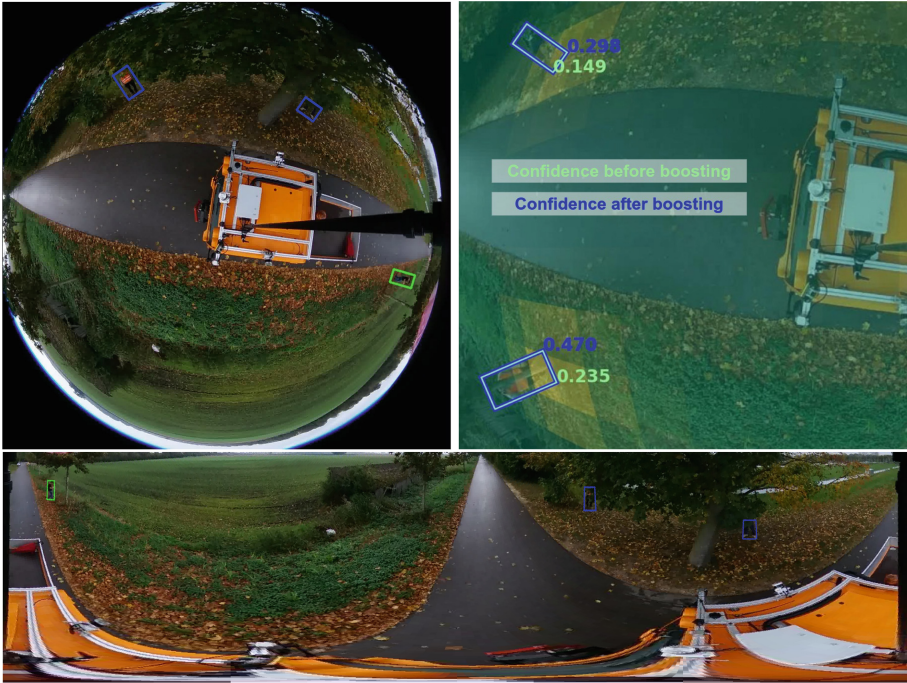


Fig. 1. With just a single camera, we obtain a full surround view: top left fisheye view, bottom equirectangular view. We propose a triangulation-based distance estimation and downstream ‘memory map’ in world space to improve object detection and alleviate missing detections caused by (partial) occlusions. With the ‘memory map’, we boost the confidence of our object detector in regions where we’ve seen more detections before (top right). A standard object detector only detects the green box, whereas the memory map is able to detect the blue boxes as well (top left).

While LiDAR has been a solution for many autonomous systems, offering precise distance measurements and object detection [3, 4], it comes with notable downsides: LiDAR systems are often expensive, adding significant cost to the deployment of autonomous technologies. Additionally, LiDAR data requires intensive computational resources to process, which can introduce latency in real-time decision-making scenarios. Furthermore, LiDAR sensors often only provide sparse point clouds that may miss objects and are insufficient for obtaining a comprehensive semantic understanding of the scene [5].

While multi-camera systems enhance situational awareness by offering a panoramic view, they also present challenges. Calibration and stitching complexity, increased data processing demands, and susceptibility to environmental conditions such as varying lighting and weather disturbances are notable disadvantages [6]. These systems require sophisticated setup and high-capacity computing resources, leading to higher costs and potential limitations in scalability and adaptability [7].

To address this, we propose a simple and inexpensive system that leverages the expansive field of vision offered by a 360° camera, combined with an object detection system running in real-time (≈ 20 FPS). This setup not only captures a complete view of the surrounding environment but also uses the camera’s geometric properties to infer the distance to detected objects. These objects are then projected into global coordinates, forming a temporal spatial map. A key innovation of our approach is the development of a “memory map” – an averaged aggregation of detection predictions over time, enhancing the robustness and accuracy of the object detection process.

Our main contributions are:

- **Dataset:** We collected a dataset comprising omnidirectional images with accompanying ground truth object detection labels and point cloud data on a mobile platform and make it publicly available¹. This dataset features a novel setup that requires only a single camera without the need for stitching.
- **3D Localization:** We explore an application by merging omnidirectional images with point cloud data for 3D localization, where the point cloud serves as a ground truth for evaluating our image-based localization method. This approach seeks to refine the precision and extend the application of localization techniques in varied outdoor settings.
- **Memory Map:** Our method enhances the object detection process by integrating geometric projections and 3D spatial information into a memory map. This enhancement aims to improve both the accuracy and robustness of detecting objects in complex outdoor environments, using spatial data to navigate the intricacies of such scenes effectively.

2 Related Work

Traditional object detection systems often are constrained by the narrow field of view (FOV) of perspective cameras, facing challenges in dynamic environments. Most of the works on surround view systems have been done on multi-camera systems [8, 9], requiring extensive installation and calibration procedures alongside heavy compute demands.

The integration of a single 360° camera offers a broader FOV, mitigating data synchronization and stitching issues and enhancing environmental perception. [10] represents a seminal approach in fast and accurate environment modeling using omnidirectional vision. Their methodology emphasizes the importance of

¹ <https://cloud.cs.uni-tuebingen.de/index.php/s/SPcSPDsigYboy7S>.

a panoramic view for complete environmental analysis. However, while their approach contributes significantly to indoor environment modeling, our method extends these concepts to address the dynamic nature of outdoor settings.

The LOAF dataset [11] employs a fisheye camera for large-scale person detection and localization, focusing on fisheye-specific distortion through novel network training techniques. However, the method employed for evaluating localization accuracy, measuring the pixel-distance relationship with a static camera placement and a single instance measurement using a measuring tape on the floor, poses limitations in terms of dynamic accuracy and real-world applicability. In contrast, our work advances this approach by leveraging LiDAR-recorded point cloud data as a ground truth. This method not only provides a more robust and accurate basis for evaluating localization but also enables the exploration of camera performance on dynamic platforms, offering a significant enhancement over the static and singular measurement technique previously.

Complementing this, [12] leverages UAV metadata to create robust dynamic maps for improved object tracking and localization, showcasing the potential of integrating metadata for enhanced UAV surveillance capabilities. DAB-DETR [13] integrates dynamic anchor boxes with the DETR model, significantly accelerating training convergence and enhances detection precision, showcasing the power of combining explicit positional priors with transformer architectures. Our work benefits from this advancement, applying DAB-DETR within context of 360° environmental understanding, pushing the boundaries of detection accuracy in dynamic settings.

The class of (occupancy) grid maps [14] is related to the simple memory map we discuss in our work. We are not concerned with creating static maps and occupancies, but rather with high-likelihood areas to make object detection temporally more robust. There are related concepts of dynamic maps, e.g. [15], but our focus to leverage these is on the improvement of object detection in surround view object detection via employing a simple memory map.

3 Methodology

3.1 Dataset Generation

Our dataset comprises of omnidirectional images and point cloud data, collected from a mobile platform equipped with a single-lens Ricoh Theta Z1 camera and three Ouster OS-1 sensors. The downward-facing camera was mounted atop the platform via a camera pole mounted at varying lengths above the vehicle, aligning the lens' central axis perpendicular to the platform. This setup gives us a *full 360° horizontal field of view* and a vertical view up to the horizon line. It mitigates the need for image stitching, simplifying the data processing pipeline considerably.

We focus on person detection and manually annotated each visible individual with radius-aligned rotated bounding boxes on the collected omnidirectional images, using our own newly designed annotation tool. This annotation approach significantly enhances the dataset's accuracy by facilitating exact person



Fig. 2. Sensor setup on the data collection vehicle. Ouster lidar sensors are positioned at the front, right, and left (B, C, D). The 360° camera is mounted at A.

positioning on the ground and enabling precise 3D geolocation from 2D pixel coordinates. In contrast, classical axis-aligned bounding boxes do not allow for precisely geolocating objects as their ground surface point is not as exact, as shown in Fig. 3. Moreover, it establishes a ground truth for evaluating our 3D geo-location-aware detection model against traditional detectors that do not use 3D geo-location information. Figure 2 shows the setup of our sensor platform.

To capture accurate 3D positions of objects surrounding the platform, we used three Ouster OS-1 LiDAR sensors, oriented towards the front, left, and right. The integration of these sensors with the camera enabled the synchronization of omnidirectional imagery with dense point cloud data for precise 3D localization of detected objects. The transformations among the camera and lidar sensors are pre-defined. The camera is located inside the field of view of at least one of the ouster sensors. See an overview of the dataset in Table 1.

Table 1. Overview of Recorded Dataset. The arrow ↓ indicates down-sampling.

Data Type	#Frames	FPS	Resolution	#Instances
Total Image Frames	5770	3 (↓ 30)	1920 × 1920	17275
Total Point Cloud Frames	1921	1 (↓ 10)	1024 × 108	5536
Image Frames - Occlusion	859	3 (↓ 30)	1920 × 1920	1875
Point Cloud Frames - Occlusion	286	1 (↓ 10)	1024 × 108	725

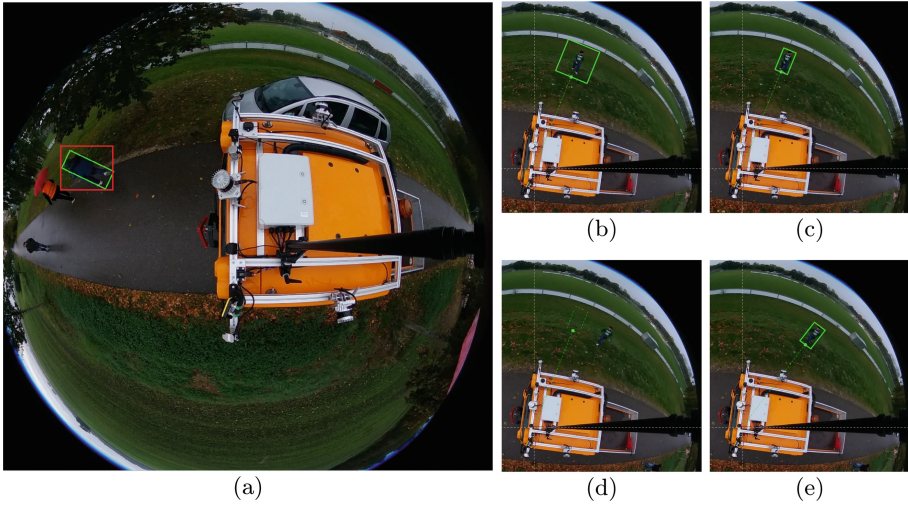


Fig. 3. (a) The axis-aligned box (red) exhibits worse properties for downstream geo-localization as the surface contact point is not as precise as for the radially corrected bounding box (green). (b)-(e) illustrate the labeling process: A rotated bounding box of default size is initially placed using mouse cursor input (b), the box size is adjusted via mouse wheel (c). We copy the annotation to the next frame (d), from where its position needs to be adjusted (e). (Color figure online)

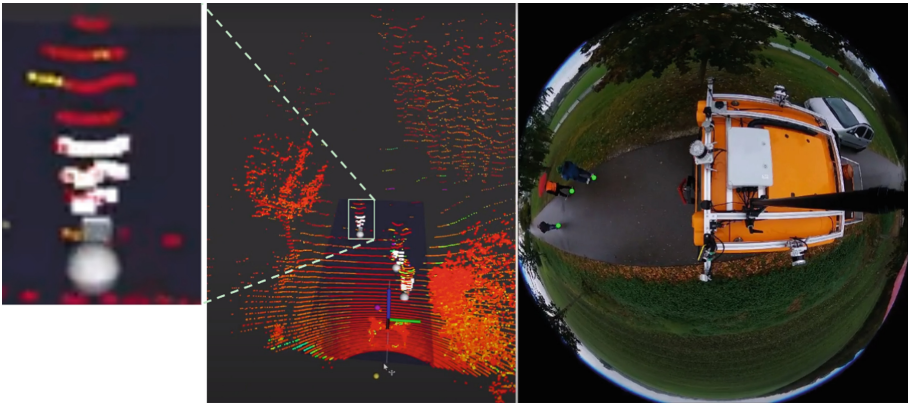


Fig. 4. Visual representation of geo-localization results using our lens model. The right panel shows an omnidirectional image captured by the Ricoh, with the green dots representing the person detections. In the middle, the point cloud data from the LiDAR sensor is displayed, where white points indicate the LiDAR’s person-classified points. The centroids of these white point clusters are projected onto the ground, depicted as gray cubes, representing the actual standing locations of persons in 3D space. The gray spheres correspond to the projected locations of the green dots from the omnidirectional image, transformed into the 3D coordinate system using the fisheye lens model. (Color figure online)

3.2 Lens Model for Passive Geo-Localization

Figure 5 illustrates the fisheye lens model. Through radial distortion, a point P on the ground plane is mapped to a point p on the image plane. The figure delineates how the distortion caused by the fisheye lens alters the projection compared to that of a traditional perspective lens, represented by point p' .

As stated in [16], a general mapping between the radial distance on the image plane r and the incidence angle θ measured with respect to the optical axis can be described as following, where k_i are the distortion coefficients of the lens model.

$$r(\theta) = k_1\theta + k_2\theta^3 + k_3\theta^5 + k_4\theta^7 + k_5\theta^9 + \dots \quad (1)$$

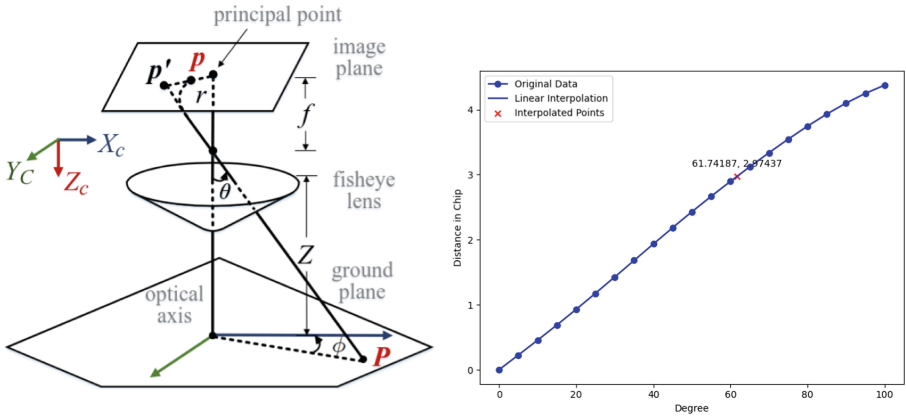


Fig. 5. Fisheye lens model for 3d localization, illustrated from [11] (left). Interpolation between θ in degrees measured with respect to the optical axis and r in millimeters on the chip. (right).

Rather than calculating the coefficients k_i for the general lens model as expressed in Eq. 1, we establish the correlation between θ and r through linear interpolation between the calibration data supplied by Ricoh, as shown in Fig. 5. This approach not only bypasses the need for higher-order polynomial computations and their inverses but also enhances the inference speed.

Once the mapping between θ and r has been set, the location of P on the ground plane can be determined from geometric calculations, knowing the projected pixel location p' on image, and the camera focus length f , the camera height Z . In the following formulae, (u, v) represent the pixel coordinates of point p , (u_c, v_c) denote the central pixel of the image. The function $\theta(r)$ relates the radial distance r to the angle θ , where θ is the angle of incidence. The angle ϕ is calculated between the vector pointing to the pixel and the image's horizontal

axis. I_{res} represents the image resolution in pixels, and S_{size} is the sensor size in millimeters.

$$x = Z \cdot \tan(\theta(r)) \cdot \cos(\phi) \quad (2)$$

$$y = Z \cdot \tan(\theta(r)) \cdot \sin(\phi) \quad (3)$$

$$z = 0 \quad (4)$$

$$\phi = \arctan 2(v - v_c, u - u_c) \quad (5)$$

$$r = \frac{\sqrt{(u - u_c)^2 + (v - v_c)^2}}{I_{res}} \cdot S_{size} \quad (6)$$

Figure 4 presents a visual exemplification of our lens model’s capabilities in geolocating objects from image data.

3.3 Confidence Calibration via Memory Mapping

The dynamic nature of the observed scene, both from the sensor platform’s mobility and the subjects’ movements, prompts a shift in pixel coordinates within the image space. However, the physical constraints on a person’s movement speed in the real world imply a relative constancy in 3D space. Leveraging this disparity, we propose a method that employs a temporal-spatial memory map to calibrate detection confidence levels, particularly in the presence of partial occlusions.

Further developing the work from [12], we construct a 3D memory map that evolves over time, informed by the projection of detected object locations and their confidence scores. This probabilistic-like approach extrapolates the change of object presence in future frames based on accumulated historical data. For the next frame, we project the newly detected bounding boxes into this 3D memory map to retrieve the corresponding memory values, which then inform the calibration of the confidence scores of the newly detected bounding boxes. This method not only fortifies detection reliability in areas with a history of presence but also reduces false positives in areas without prior detections.

The memory map M spans a predefined area centered at the sensor platform, encompassing a 100 by 100m region, discretized with grid size $s \times s$. In the following, $M^t(x, y)$ denotes the grid cell containing x and y in the discretization M for arbitrary x, y at time $t \in \mathbb{N}_0$. For $t = 0$, each grid cell the memory map is initialized to a value of 0.5 to represent an equi-probable state of object presence:

$$M^0(x, y) := 0.5, \quad \forall x, y. \quad (7)$$

For each of the raw detections on image at timestamp $t > 0$, c_i^t denotes its confidence, p_i^t its bottom center pixel, and $l_i^t = (x_i^t, y_i^t)$ its coordinate in the

world frame. This is obtained from the pixel to 3D projection as described in Eqs. 2,3,4,5 and 6. These contribute to the memory map update in the close vicinity of the respective point, in detail:

$$M^t(x, y) = \text{normMean}(M^{t-1}(l_i^t) + c_i^t), \forall i \quad (8)$$

$$l_i^t = \text{proj}_{3D}(p_i^t), \quad (9)$$

where $\text{normMean}()$ is an operation that shifts the values within the memory map M^t to ensure an average value of 0.5, thus standardizing the probable state distribution. Above, the update step in Eq. 8 is applied to all indices in a 3×3 grid around l_i^t ,

$$(x, y) \in \{x_i^t, x_i^t \pm s\} \times \{y_i^t, y_i^t \pm s\}. \quad (10)$$

In the following, $\beta > 1$ is a boosting factor for updating the confidence value. Then the calibrated confidence value \tilde{c}_i^t of the detected object can be formulated as follows:

$$\tilde{c}_i^t = \begin{cases} c_i^t \cdot \beta, & \text{if } M^{t-1}(l_i^t) \geq 0.5, \\ c_i^t / \beta, & \text{if } M^{t-1}(l_i^t) < 0.5. \end{cases} \quad (11)$$

Equation 11 increases the confidence of a detection if the memory map M^{t-1} from last time step suggests there should be an object and vice versa.

Afterwards, the detected object with the calibrated confidence will be filtered by the confidence threshold or by the non maximum suppression following the standard processes of the detection model, leading to the final detection result.

Figure 6 shows a visual representation of the 3D memory map, which is projected back onto the image plane, offering a color-coded likelihood of person presence within each grid cell.

4 Experimental Results

This section provides a detailed overview of our hardware used, experimental setup and the results obtained.

4.1 Hardware and Tools

- **Ricoh Camera:** the camera Ricoh Theta Z1 was configured to capture image data at a resolution of 1920×1920 pixels at a frequency of 30 Hz. The camera orientation was horizontal, with one lens directed towards the ground and the other skyward. This setup was deliberately chosen to circumvent the complexities and potential inaccuracies associated with image stitching. While this configuration results in the camera pole being visible within the image frame, we contend that the obstruction is minimal and does not significantly impact the overall effectiveness of our method’s evaluation.

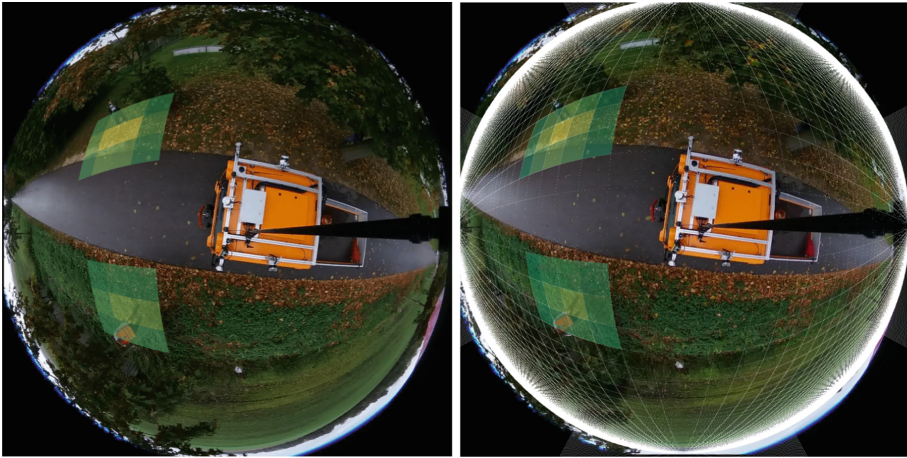


Fig. 6. Illustration of the reprojected 3D memory map onto the image space, where bright yellow indicates high likelihood areas for person presence. The grid size is set at 1 m. For optimal clarity, only regions with map values exceeding a selected threshold are used for overlay. Please see how the memory map evolves overtime in the supplementary video. The right image visualizes the discretization of the geo-memory map. (Color figure online)

- **Ouster Lidar:** The three Ouster Lidar OS1 sensors, each with 64 channels, are operating at 10 Hz. They were mounted on the front, left, and right sides of our mobile platform. The front sensor was angled 20° downward from the horizontal plane, while the side sensors were tilted 60° downward, optimizing the field of view for comprehensive spatial data capture. This configuration ensured a rich point cloud dataset, enhancing the evaluation of our localization model.
- **Custom Annotation Tool for Object Detection:** We developed an annotation tool² tailored to produce radius-aligned rotated bounding boxes for person in 360° footage. Our annotation framework characterizes each bounding box by a set of parameters that denote its position, dimensions, and orientation in the image space. Specifically, each bounding box is defined by the coordinates of its center (x_{center}, y_{center}) , its dimensions given by the width and height, and the angle of rotation α relative to the image axes. This precise parameterization allows us to accurately pinpoint the location of a person’s stance in the image space. This annotation approach, as illustrated in the right panel of Fig. 4, is crucial for the subsequent accurate projection into the person’s 3D location.

² <https://cloud.cs.uni-tuebingen.de/index.php/s/SPcSPDsigYboy7S>.

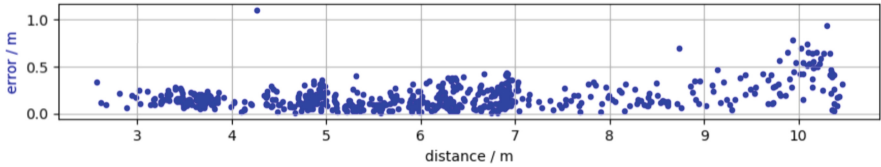


Fig. 7. The 3D localization error relative to object distance, where the clustering of points below the 0.5-meter error margin up to 10 m highlights the method’s precision.

4.2 Experimental Setup and Results

To evaluate our lens model and geo-localization approach, we synchronized our point cloud and image data. The bottom center pixel of each annotated bounding box in the images is projected into 3D space to obtain a geolocation estimate. For the point cloud data, given the tilted positioning of the lidar sensors, we used RANSAC [17] to define the ground plane’s coefficients. Following this, DBSCAN [18] was applied to identify individual clusters, with their centroids projected onto the ground plane to serve as ground truth for our 3D location estimates from the lens model. Figure 7 showcases the 3D localization error of our method with respect to the distance from the mobile platform. The error values generally remain below 0.5 m for distances up to 10 m, indicating a high degree of accuracy in the localization process. This consistent error margin underscores the model’s reliability, particularly notable in the mid-range distances where the density of data points suggests robust performance despite increasing distance. The plot emphasizes the method’s accuracy in estimating positions in 3D space.

Table 2. Experimental results for temporal-spatial memory mapping approach with model DAB-DETR. Here, s is the map grid size, and β is the boosting factor. Columns with - for s and β denote the baseline without memory maps.

s	Non-occlusion			Partial occlusion				
	-	1.5	1	-	0.75	1.5	1	1.5
β	-	1.3	2	-	2	1.3	2	2
AP@50	0.9702	0.9702	0.9703	0.410	0.425	0.426	0.430	0.473
AR10	0.5778	0.5778	0.5778	0.1695	0.176	0.178	0.176	0.198

To assess the effectiveness of our confidence calibration approach through temporal-spatial memory mapping, we conducted the following experiments: detection performance with vs. without 3D location information, impact of spatial memory map resolution, evaluation of memory map and confident update functions.

We adopted the DAB-DETR [13] as our primary model for image-based object detection, in alignment with the methodologies proposed in [11]. We

Table 3. Experimental results with model YoloV7-tiny.

s	Non-occlusion			Partial occlusion				
	–	1.5	1	–	0.75	1.5	1	1.5
β	–	1.3	2	–	2	1.3	2	2
AP@50	0.6773	0.7087	0.6989	0.1412	0.1458	0.1543	0.1420	0.1409
AR10	0.4072	0.4093	0.4065	0.0915	0.1127	0.1127	0.1127	0.1127

Table 4. Experimental results with YoloV9-tiny.

s	Non-occlusion			Partial occlusion				
	–	1.5	1	–	0.75	1.5	1	1.5
β	–	1.3	2	–	2	1.3	2	2
AP@50	0.5425	0.5698	0.5841	0.0829	0.0869	0.0926	0.0869	0.0925
AR10	0.2972	0.3065	0.3102	0.0559	0.0636	0.0627	0.0636	0.0627

trained the model on the LOAF dataset for 50 epochs and then fine-tuned further on our custom annotated dataset for an additional 20 epochs with the default settings.

To assess the efficacy of our temporal-spatial memory mapping approach for confidence calibration, we selected $AP@50$ and $AR10$ as our metrics due to their established reliability in object detection evaluation [19–22]. $AP@50$ provides a balanced measure of precision at a 50% IoU threshold, emphasizing the accuracy of detections, while $AR10$ evaluates the model’s recall, considering the top 10 detection results, to gauge its ability to detect relevant objects without being overwhelmed by false positives. Our experiments were stratified into two distinct scenarios. For the first, absent of occlusions, we explored a variety of memory map grid sizes (0.5, 0.75, 1, and 1.5 m) and confidence boost factors (1.3, 1.5, and 2). The $AP@50$ observed across these configurations ranged from 0.9700 to 0.9703, closely aligning with the baseline of 0.9702, achieved without incorporating 3D geo-information. This outcome shows the robustness of our detection method even in configurations not optimized for the highest performance, without significantly compromising detection accuracy in scenarios free from occlusions.

In the second scenario, with partial occlusions by tree trunks or leaves, the application of 3D geolocation data significantly elevated the $AP@50$ and $AR10$, highlighting the method’s value in visually complex situations. The $AP@50$ ranges from 0.428 to 0.473, while the baseline is 0.410. Among the 12 settings we explored, ten showed significant improvement over the baseline, underscoring the effectiveness of our approach. The best four performances are quantitatively presented in Table 2. To test the generalization of our approach, we also conducted the same experiments on other models, e.g. YOLOv7-tiny [23] and YOLOv9-tiny [24], obtaining similar results. Specifically, on an NVIDIA Orin AGX, our app-

roach with the YOLOv7-tiny model achieved real-time performance of 25 FPS. The results are presented in Table 3 and Table 4.

5 Conclusion, Limitations and Future Work

We presented a new approach for surround view object localization and enhancing object detection in dynamic environments through the integration of a single 360° camera. Our method significantly improves the detection and mapping of partially occluded objects by using the comprehensive view provided by the camera and the aggregating object detection proposals to create a temporal spatial map. This map is continuously updated, enhancing the detection accuracy in complex settings.

Despite its promising results, our approach has limitations, such as the assumption of an underlying 2D surface and a perpendicular 360° camera facing downwards. Furthermore, the camera pole is currently blocking the view slightly. Future work will aim to refine these aspects and explore further applications. In particular, we aim to design a construction where the lens is placed directly above the pole, and having a transparent mounting around it, so that there is no pole blocking the view.

References

1. Miller, D., Goode, G., Bennie, C., Moghadam, P., Jurdak, R.: Why object detectors fail: Investigating the influence of the dataset. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4823–4830 (2022)
2. Wang, J., et al.: When, where and how does it fail? A spatial-temporal visual analytics approach for interpretable object detection in autonomous driving. *IEEE Trans. Vis. Comput. Graph.* **29**(12), 5033–5049 (2022)
3. Venugopala, S.H.: Comparative study of 3D object detection frameworks based on lidar data and sensor fusion techniques. In: *Journal of Physics: Conference Series*, vol. 2232, no. 1, pp. 012015 (2022). IOP Publishing
4. Li, Y., Ibanez-Guzman, J.: Lidar for autonomous driving: the principles, challenges, and trends for automotive lidar and perception systems. *IEEE Signal Process. Mag.* **37**(4), 50–61 (2020)
5. Wang, W., Sakurada, K., Kawaguchi, N.: Incremental and enhanced scanline-based segmentation method for surface reconstruction of sparse lidar data. *Remote Sens.* **8**(11), 967 (2016)
6. Rampinelli, M., et al.: An intelligent space for mobile robot localization using a multi-camera system. *Sensors* **14**(8), 15:039–15:064 (2014)
7. Losada, C., Mazo, M., Palazuelos, S., Pizarro, D., Marrón, M.: Multi-camera sensor system for 3d segmentation and localization of multiple mobile robots. *Sensors* **10**(4), 3261–3279 (2010)
8. Varlashin, V., Semakova, A., Shmakov, O.: Real-time surround view system for mobile robots. In: Ronzhin, A., Shishlakov, V. (eds.) Proceedings of 14th International Conference on Electromechanics and Robotics Zavalishin’s Readings. SIST, vol. 154, pp. 465–476. Springer, Singapore (2020). https://doi.org/10.1007/978-981-13-9267-2_38

9. Kumar, V.R., et al.: OmniDet: surround view cameras based multi-task visual perception network for autonomous driving. *IEEE Robot. Autom. Lett.* **6**(2), 2830–2837 (2021)
10. Heinemann, P., Rückstieß, T., Zell, A.: Fast and accurate environment modelling using omnidirectional vision. *Dynamic Perception*, pp. 9–14 (2004)
11. Yang, L., Li, L., Xin, X., Sun, Y., Song, Q., Wang, W.: Large-scale person detection and localization using overhead fisheye cameras. In: *ICCV* (2023)
12. Kiefer, B., Quan, Y., Zell, A.: Memory maps for video object detection and tracking on UAVs. 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 3040–3047 (2023). <https://api.semanticscholar.org/CorpusID:257378530>
13. Liu, S., et al.: DAB-DETR: dynamic anchor boxes are better queries for DETR. In: *International Conference on Learning Representations* (2022). <https://openreview.net/forum?id=oMI9PjOb9Jl>
14. Nuss, D., et al.: A random finite set approach for dynamic occupancy grid maps with real-time application. *Int. J. Robot. Res.* **37**(8), 841–866 (2018)
15. Chung, S.-Y., Huang, H.-P.: SLAMMOT-SP: simultaneous SLAMMOT and scene prediction. *Adv. Robot.* **24**(7), 979–1002 (2010)
16. Kannala, J., Brandt, S.: A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses. *IEEE Trans. Pattern Anal. Mach. Intell.* **28**, 1335–40 (2006)
17. Fischler, M., Bolles, R.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM* **24**(6), 381–395 (1981). <http://publication.wilsonwong.me/load.php?id=233282275>
18. Ram, A., Sunita, J., Jalal, A., Manoj, K.: A density based algorithm for discovering density varied clusters in large spatial databases. *Int. J. Comput. Appl.* **3**, 06 (2010)
19. Lin, T.-Y., et al.: Microsoft COCO: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) *ECCV 2014*. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_48
20. Hosang, J., Benenson, R., Dollár, P., Schiele, B.: What makes for effective detection proposals? *IEEE Trans. Pattern Anal. Mach. Intell.* **38**(4), 814–830 (2016)
21. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. *IEEE Trans. Pattern Anal. Mach. Intell.* **32**(9), 1627–1645 (2010)
22. Everingham, M., Eslami, S.M.A., Gool, L.V., Williams, C.K.I., Winn, J.M., Zisserman, A.: The pascal visual object classes challenge: a retrospective. *Int. J. Comput. Vis.* **111**, 98 – 136 (2014). <https://api.semanticscholar.org/CorpusID:207252270>
23. Wang, C.-Y., Bochkovskiy, A., Liao, H.-Y.M.: YOLOv7: trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7464–7475 (2023)
24. Wang, C.-Y., Yeh, I.-H., Liao, H.-Y.M.: YOLOv9: Learning what you want to learn using programmable gradient information. *arXiv preprint arXiv:2402.13616* (2024)



Exploring the Reliability of Foundation Model-Based Frontier Selection in Zero-Shot Object Goal Navigation

Shuaihang Yuan^{1,2,4}, Halil Utku Unlu³, Hao Huang^{2,4}, Congcong Wen^{2,4},
Anthony Tzes^{1,2}, and Yi Fang^{1,2,4} (✉)

¹ NYUAD Center for Artificial Intelligence and Robotics (CAIR), Abu Dhabi, UAE

² Electrical Engineering, New York University Abu Dhabi, Abu Dhabi 129188, UAE

³ Electrical and Computer Engineering Department, New York University, Brooklyn, NY 11201, USA

⁴ Embodied AI and Robotics (AIR) Lab, NYU, Abu Dhabi, UAE
yfang@nyu.edu

Abstract. In this paper, we present a novel method for reliable frontier selection in Zero-Shot Object Goal Navigation (ZS-OGN), enhancing robotic navigation systems with foundation models to improve commonsense reasoning in indoor environments. Our approach introduces a multi-expert decision framework to address the nonsensical or irrelevant reasoning often seen in foundation model-based systems. The method comprises two key components: Diversified Expert Frontier Analysis (DEFA) and Consensus Decision Making (CDM). DEFA utilizes three expert models—furniture arrangement, room type analysis, and visual scene reasoning—while CDM aggregates their outputs, prioritizing unanimous or majority consensus for more reliable decisions. Demonstrating state-of-the-art performance on the RoboTHOR and HM3D datasets, our method excels at navigating towards untrained objects or goals and outperforms various baselines, showcasing its adaptability to dynamic real-world conditions and superior generalization capabilities.

Keywords: Zero-shot Object Goal Navigation · Foundation Model Reasoning

1 Introduction

Leveraging foundation models has greatly advanced robotic navigation systems, particularly for frontier-based object goal navigation in indoor environments [58, 60, 64, 66, 67]. These models enable robots to apply commonsense reasoning during exploration and object search [17, 26, 53, 54, 59]. For instance, when the target is a desk, the robot understands that desks are often paired with chairs. This enhanced perception and reasoning allow navigation systems to more effectively identify promising frontiers for exploration, resulting in higher success rates compared to traditional methods like distance-based and Gaussian-process-based frontier selection [1, 2, 21, 38, 46, 65].

Recently, researchers have integrated chain-of-thought (COT) prompting [49,57,61] into foundation models to enhance commonsense reasoning in navigation systems [27,42,56,67]. COT generates short, human-like reasoning steps during navigation tasks. For example, when searching for a desk, the system might reason, “A desk is often found in a study room, which typically contains books, laptops, and chairs. If I encounter these objects near an unexplored frontier, I should explore it first.” COT enhances navigation performance by offering a more transparent and interpretable decision-making process. However, it often relies on greedy decoding, which can lead to suboptimal reasoning [8,48,49], resulting in nonsensical or irrelevant conclusions and decreasing system reliability [47]. This limitation highlights the need for more advanced methods to enhance the robustness of foundation model-driven navigation systems (Fig. 1).

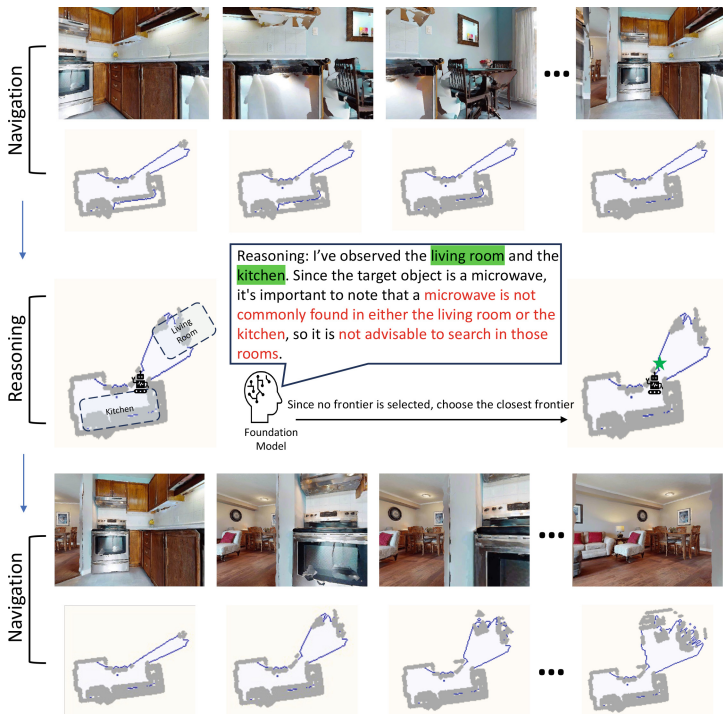


Fig. 1. Instances of nonsensical or irrelevant reasoning, during the frontier selection in Zero-Shot Object Goal Navigation. The green text indicates a correct understanding of the scene, while the red text refers to the reasoning that contradicts human intuition. (Color figure online)

In real-world dynamic environments, the reliability of foundation model-driven navigation systems is vital, especially in adapting to changing scenarios where consistent performance is difficult to maintain. Our research addresses this challenge by developing robust commonsense reasoning for zero-shot object goal navigation. This

approach is critical for navigating unpredictable conditions where robots encounter previously unseen objects or situations. Unlike traditional tasks, zero-shot navigation [16, 67] requires the system to orient toward goals without prior explicit training, demanding advanced generalization and commonsense knowledge to reliably adapt across unfamiliar objects and situations in dynamic environments.

In this paper, we introduce a novel frontier selection method for zero-shot object goal navigation (ZS-OGN). The method features two key components, with the first being Diversified Expert Frontier Analysis (DEFA), inspired by Portfolio Theory [9, 32–34]. DEFA leverages the expertise of three foundation models, each serving as an expert in a specific aspect of frontier selection. The first expert focuses on selecting frontiers based on furniture arrangements, such as identifying desk-like setups with chairs. The second expert prioritizes frontiers leading to rooms where the target object, like a desk, is more likely to be found, such as study rooms. The third expert uses visual observation to apply commonsense reasoning dynamically based on the scene.

We introduce Consensus Decision Making (CDM) as the second component for frontier selection, inspired by self-consistency. CDM first seeks unanimous expert approval, and if not achieved, selects a frontier endorsed by at least two experts. This approach balances the diversity of DEFA experts while enhancing reliability in zero-shot object goal navigation. Our system’s state-of-the-art performance on the RoboTHOR [10] and HM3D [39] datasets validates its effectiveness, with detailed analysis further demonstrating the reliability of our method compared to baseline approaches.

2 Related Work

2.1 Language-Driven Zero-Shot Object Navigation

In Object Goal Navigation (OGN), the goal is to efficiently explore a new environment while searching for a non-visible target object. Previous research often relies on visual context through imitation [24, 44] or reinforcement learning [23, 51], which require extensive data collection and annotations, limiting their practicality in real-world environments. The focus has shifted towards zero-shot object navigation, enabling robots to adapt to new objects and environments without specific training [12, 30, 62, 63]. Clip-Nav [13] and CoW [15] use CLIP [37] for zero-shot navigation, while L-ZSON [16] employs Frontier-Based Exploration (FBE) [55] to navigate between known and unknown spaces, outperforming learning-based methods [40, 50]. Unlike recent works [3, 52] that train policy networks for frontier exploration, we leverage Large Language Models (LLMs) like GPT-3.5 [36] and GPT-4 [35] to make navigational decisions directly, bypassing the need for any training process.

2.2 Commonsense Reasoning in Navigation

Commonsense reasoning [25, 28] is critical for achieving human-like intelligence in robotics [19, 45]. Large pre-trained LLMs with reasoning capabilities are becoming

increasingly vital for navigation. For instance, BERT [11] enhances navigation by linking language instructions to navigational paths [31], while GPT-4 further improves commonsense reasoning in navigation [6, 12, 43, 66]. NavGPT [66] integrates prompt-based methods, like ReAct [58], with discrete action spaces for better navigation. Other work leverages commonsense knowledge and semantic mapping to improve goal identification and navigation [4, 5, 7]. Recent research also integrates predictions from language models with planning or probabilistic inference [20, 43], while some focus on grounding language models in image observations [14, 18, 22]. JARVIS [64] offers a neuro-symbolic framework for generalizable conversational embodied agents, while ESC [67] pre-computes object-room relationships for zero-shot navigation, though it struggles with evolving environments. We propose a novel approach that dynamically infers commonsense knowledge from observed scenes, overcoming this limitation.

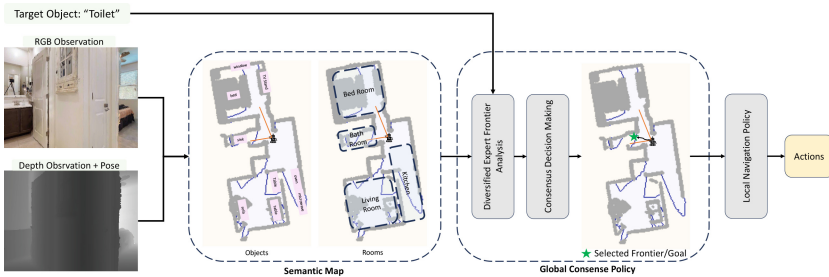


Fig. 2. Workflow of the proposed ZS-OGN system, RF-NAV, for Zero-Shot Object Goal Navigation (ZS-OGN). The process begins with RGB and depth observations leading to the creation of a semantic map, which includes identified objects and room labels. This map informs the Diversified Expert Frontier Analysis (DEFA) and subsequent Consensus Decision Making (CDM) to select the most viable frontier or goal, here exemplified by the search for a ‘Toilet.’ The chosen goal is then fed into the Local Navigation Policy, which determines the actions necessary for the robot to explore the unknown environment.

3 Problem Formulation

In ZS-OGN, the robot must navigate to a target object g_i in an unfamiliar environment s_i , without prior training on navigation data. Each episode is defined as $\mathcal{E}_i = \{g_i, s_i, p_0\}$, where p_0 is the robot’s starting position. At each step t , the robot receives an observation $\mathcal{O}_t = \{I_t, d_t, x_t, y_t, \theta_t\}$, which includes a color image I_t , depth image d_t , and its pose (position (x_t, y_t) and orientation θ_t). Over time, the robot accumulates pose readings to track its relative position p_t . Based on these observations, the robot selects an action a from the action space $\mathcal{A} = \{\text{“move forward”}, \text{“turn left”}, \text{“turn right”}, \text{“stop”}\}$ via a policy function $\pi(\cdot)$. Success is achieved if the robot executes the “stop” action within a predefined distance of the target object. In this study, we frame the navigation task as a sequence of decisions starting at time step $t = 0$ and ending at the final step T , either when the target is found

or the maximum steps are reached. The challenge lies in developing a zero-shot policy π , designed to select the optimal action a_t at each step t based on the observation \mathcal{O}_t .

4 Method

To tackle the problem outlined above, we employ Frontier-based Exploration [55]. Our method is organized into three modules: Mapping, Global Commonsense Policy, and Local Navigation Policy, as illustrated in Fig. 2. First, RF-NAV constructs a semantic and frontier map based on the observation \mathcal{O}_t (see Sect. 4.1). Next, in the Global Commonsense Policy, we introduce Diversified Expert Frontier Analysis (DEFA) and Consensus Decision Making (CDM) to select the most promising frontier for further exploration (see Sect. 4.2). Finally, the Local Navigation Policy plans the path to the frontier or target object and generates the necessary actions to reach it, as detailed in Sect. 4.3.

4.1 Mapping

Constructing semantic and frontier maps are fundamental modules in various frontier-based navigation systems [38]. Following the approach outlined in [5], we construct the semantic map using RGB-D images and the agent’s pose. The RGB-D input is transformed into 3D voxels and then projected onto a top-down 2D navigation map. We utilize the ESC [67] framework to extract semantic information, including common objects and room types in \mathcal{E}_i , using the Grounded Language-Image Pre-training (GLIP) model [29]. This model enables zero-shot detection capabilities through natural language prompting, allowing us to detect both objects and room types. The zero-shot detection process using object prompting P_o and room prompting P_r is formally described as: $\{o_{t,n}, b_{o_{t,n}}\} = GLIP(I_t, P_o)$, $\{r_{t,n}, b_{r_{t,n}}\} = GLIP(I_t, P_r)$. Here, $o_{t,i}$ and $r_{t,i}$ represent the predicted labels of objects and rooms, respectively, while $b_{o_{t,i}}$ and $b_{r_{t,i}}$ denote their bounding boxes. The index i indicates the detected room or object at step t . The locations of detected rooms and objects are then projected to form the semantic map.

To generate the frontier map from the navigation map, we adhere to the methodology outlined in [38]. Initially, we identify the edges of the free area, defined as the space visible to the agent and not obstructed by obstacles. Subsequently, the boundary points between the free area and the unexplored space are identified as candidate frontiers. These candidates are then sent to the Global Commonsense Policy to determine the most promising frontier for the next phase of exploration.

Algorithm 1. Consensus Decision Making

```

1: Input: Sets  $F_{O2F}$ ,  $F_{R2F}$ ,  $F_{SLE}$  of frontiers recommended by three experts; frontier distance
   matrix  $d$  ▷ Define inputs
2: Output: A frontier ▷ Define output
3: procedure FINDCONSENSUS( $F_{O2F}$ ,  $F_{R2F}$ ,  $F_{SLE}$ ) ▷ Begin consensus finding
4:    $F_{\text{unanimous}} \leftarrow F_{O2F} \cap F_{R2F} \cap F_{SLE}$  ▷ Check for unanimous consensus
5:   if  $F_{\text{unanimous}} \neq \emptyset$  then
6:     return  $F_{\text{unanimous}}$  ▷ Return if unanimous consensus exists
7:    $F_{\text{partial}} \leftarrow (F_{O2F} \cap F_{R2F}) \cup (F_{O2F} \cap F_{SLE}) \cup (F_{R2F} \cap F_{SLE})$  ▷ Check for partial consensus
8:   if  $F_{\text{partial}} \neq \emptyset$  then
9:     return  $F_{\text{partial}}$  ▷ Return if partial consensus exists
10:  else
11:    return CLOSESTFRONTIER( $d$ ) ▷ Return the closest frontier
12: procedure SELECTFRONTIER ▷ Begin frontier selection
13:    $F_{\text{consensus}} \leftarrow \text{FINDCONSENSUS}(F_{\text{object}}, F_{\text{room}}, F_{\text{proximity}})$  ▷ Get consensus set
14:    $F_{\text{selected}} \leftarrow \arg \max_{f \in F_{\text{consensus}}} d[f]$  ▷ Select max confidence frontier
15:   return  $f_{\text{selected}}$  ▷ Return selected frontier
16:  $f_{\text{final}} \leftarrow \text{SELECTFRONTIER}$  ▷ Determine final selection
17: Output  $f_{\text{final}}$  ▷ Output the final selected frontier

```

4.2 Global Commonsense Policy

Global Commonsense Policy, π_{global} , is responsible for selecting the best frontier by leveraging the commonsense reasoning ability from the foundation models. The selection of a frontier is based on the nearby objects, the room type, and the room configuration. The output is a chosen frontier f_t , which is a point in the environment the robot aims to reach. Global Commonsense Policy consists of two components: (1) Diversified Expert Frontier Analysis (DEFA) to analyze the frontiers from diverse perspectives and (2) Consensus Decision Making (CDM) to produce the final decision by considering all the options produced by the DEFA.

Diversified Expert Frontier Analysis. In the DEFA module, we employ three distinct expert models to realize the decision-making process in ZS-OGN, each bringing a unique perspective to frontier selection. The Object2Frontier Expert (O2F) specializes in analyzing the objects near potential frontiers, identifying frontiers that are indicative of the target object’s likely presence. We leverage the reasoning ability from an LLM to realize this, as $F_{O2F}(\{o\}) \rightarrow \{S_{O2F}\}$, where $\{o\}$ is the set of observed objects near each frontier, and S_{O2F} is the selected frontiers. We use ChatGPT-3.5 as the O2F in all of our experiments.

In addition to the O2F, we further employ an LLM as the Room2Frontier Expert (R2F). R2F assesses the room type associated with each frontier, prioritizing those that align with the expected location of the target, such as study rooms for a desk, denoted as $F_{R2F}(\{r\}) \rightarrow S_{R2F}$, with r denoting room types and S_{R2F} the selected frontier by this expert. We also adopt the ChatGPT-3.5 as the R2F throughout our experiments.

Lastly, to complement the analysis, we adopt a Scene Layout Expert (SLE) specifically to compensate for the loss of visual information not addressed by the previous

experts. This expert leverages visual data to dynamically reason commonsense knowledge based on the observed scene. The SLE is implemented using a Multimodal Large Language Model (MLLM), GPT-4V, which processes RGB observations $\{I\}$, alongside detected objects and room types. This is formulated as $F_{SLE}(\{I\}, \{o\}, \{r\}) \rightarrow S_{SLE}$. Each expert operates independently, diversifying the frontier evaluation criteria and determining a reliable frontier to navigate. To integrate the decisions from various experts, we utilize the Consensus Decision Making component.

Consensus Decision Making. We introduce Consensus Decision Making (CDM) for selecting the frontier from all recommendations from the experts. This straightforward yet effective approach relies on majority voting and reduces the occurrence of instances of nonsensical or irrelevant reasoning. Ideally, all experts agree upon a single frontier. When a unanimous selection is not achieved, the strategy chooses the frontier endorsed by the majority. After the selection is made, we rank the frontiers according to their distances to the robot’s current location to determine the final selection.

While we have demonstrated that the lower bound of our method to produce an irrational result is lower than that of relying on a single expert to determine the frontier, it is important to note that our algorithm might still encounter situations where the three experts do not reach any consensus. To mitigate this issue, we have incorporated a fallback strategy, in which the robot will select the closest frontier if no consensus is achieved. We detail this entire Consensus Decision-Making (CDM) process in Algorithm 1. After the goal (either the frontier or the location of the target object) has been selected, the Local Navigation Policy generates the path planning and sequences the actions needed to reach the goal.

Algorithm 2. Frontier-based Exploration Method

Require: Observation \mathcal{O}_t , Semantic Map $\mathcal{M}_{\text{semantic}}$, Frontier Map $\mathcal{M}_{\text{frontier}}$

- 1: **Mapping:**
 - 2: Detect objects and room types using GLIP model
 - 3: Construct a semantic map using RGB-D images and the agent’s position
 - 4: Construct a frontier map and identify candidate frontiers
 - 5: **End Mapping**
 - 6: **Global Commonsense Policy** π_{global} :
 - 7: *Diversified Expert Frontier Analysis (DEFA):*
 - 8: $S_{O2F} \leftarrow F_{O2F}(\{o\})$ ▷ Object2Frontier Expert
 - 9: $S_{R2F} \leftarrow F_{R2F}(\{r\})$ ▷ Room2Frontier Expert
 - 10: $S_{SLE} \leftarrow F_{SLE}(\{I\}, \{o\}, \{r\})$ ▷ Scene Layout Expert
 - 11: *Consensus Decision Making (CDM):*
 - 12: $f_{\text{final}} \leftarrow \text{CDM}(S_{O2F}, S_{R2F}, S_{SLE})$
 - 13: **End Global Commonsense Policy**
 - 14: **Local Navigation Policy** π_{local} :
 - 15: Plan path to f_{final} using Fast Marching Method (FMM)
 - 16: $a_t \leftarrow \pi_{\text{local}}(\mathcal{O}_t, f_{\text{final}})$
 - 17: **End Local Navigation Policy**
-

4.3 Local Navigation Policy

To navigate from the agent’s current location to a goal produced from the CDM, we employ the Fast Marching Method (FMM) [41], a numerical technique that efficiently solves the Eikonal equation, providing a way to estimate the minimal time necessary for the agent to reach the selected frontier from its starting point in the environment. Once a frontier is selected, the local navigation policy π_{local} is responsible for planning the path to this frontier and generating the appropriate actions to navigate along this path. This network takes as input the current observation \mathcal{O}_t and the selected frontier f_{final} , and outputs the action a_t to be taken at time step t . $a_t = \pi_{\text{local}}(\mathcal{O}_t, f_{\text{final}})$. The combined policy π operates by first using π_{global} to select a frontier and then using π_{local} to navigate towards this frontier. This process is repeated at each time step t until the robot either reaches the target object or the episode ends.

In this formulation, π_{global} provides a strategic decision-making capability, selecting waypoints or goals that guide the overall navigation task. In contrast, π_{local} is focused on the immediate, tactical decisions required to navigate safely and efficiently to the chosen frontier. This division allows the policy to effectively manage both the high-level navigation objectives and the detailed, moment-to-moment challenges of robot movement in an unknown environment. The formulation of our complete navigation system flow can be found in Algorithm 2

5 Simulation Studies

5.1 Datasets and Metrics

HM3D [39], a foundational dataset for the Habitat 2022 ObjectNav challenge, includes 142,646 object instances across 40 classes and 216 3D environments, covering 3,100 rooms. We follow prior validation settings [16, 67] to evaluate our method. **RoboTHOR** [10] serves as a real-world benchmark with 89 apartment scenes and 731 unique objects. We assess our method on 1,800 validation episodes across 15 environments, focusing on 12 target object categories for zero-shot object goal navigation.

We use Success Rate (SR) and Success Weighted by Path Length (SPL) to evaluate the effectiveness of our proposed method. **SR** metric focuses on the agent’s accuracy in reaching the designated target, expressed as a percentage, where a higher value indicates better performance. SR is a binary indicator of whether the robot successfully stops within 0.1 m of the target object g_i within the episode. In addition, we also measure SPL.

SPL is a metric that evaluates success relative to the shortest possible path, normalized by the actual path taken by the agent. It effectively measures the efficiency of the agent’s success in reaching its goal.

5.2 Baselines

We compare our method with two state-of-the-art (SOTA) approaches in zero-shot object goal navigation (ZS-OGN) and our own baseline methodologies. CoW (CLIP on Wheels) [16] tackles language-driven ZS-OGN without fine-tuning, using CLIP to

Table 1. Comparison of Zero-shot OGN methods on the HM3D and RoboTHOR benchmarks using SPL and SR metrics, showing our models’ superior performance, especially with the Consensus Commonsense strategy.

Model	Frontier Selection	HM3D		RoboTHOR	
		SPL↑	SR↑	SPL↑	SR↑
CLIP-Ref	Closest	–	–	2.1	2.7
MDETR	Closest	–	–	8.4	9.9
CLIP-Grad	Closest	–	–	9.7	13.8
CLIP-Patch	Closest	–	–	10.6	20.3
CoW	Closest	–	–	16.9	26.7
ESC	Commonsense	17.8	35.4	18.2	34.5
Ours (k = 3)	Majority Commonsense	18.9	36.3	20.6	35.2
Ours (k = 5)	Majority Commonsense	19.1	36.6	20.8	35.6
Ours	Consensus Commonsense	21.7	37.4	22.3	36.8

identify target objects and select frontiers. We also evaluate CoW variants: *CLIP-Ref*, *CLIP-Patch*, *CLIP-Grad*, *MDETR*. ESC [67] applies commonsense knowledge from a pre-trained LLM to navigate unseen environments, combining vision and language models for object identification and reasoning. Additionally, we developed a baseline where a single expert repeatedly determines the next frontier, selecting the most frequent outcome for exploration, unlike the more sophisticated reasoning process in ESC.

5.3 Results on HM3D

In this dataset, our ZS-OGN system outperforms the CoW and ESC models in both SPL and SR metrics, as shown in Table 1. The SR improvement from 35.4 to 37.4 highlights our model’s enhanced understanding of environmental semantics, aided by the Multimodal Large Language Model expert. The SPL increase from 17.8 to 21.7 demonstrates the effectiveness of our multi-expert approach in exploring unknown environments. Additionally, our model surpasses the Ours (M.V) approach, which relies on a single expert’s majority consensus. The collaborative decision-making process in our model results in a more refined navigation strategy, leading to higher SPL in complex real-world HM3D environments.

5.4 Results on RoboTHOR

In this dataset, we test our ZS-OGN system in an unknown environment, where it outperforms the CoW and ESC models in both SPL and SR metrics, as shown in Table 1. The increase in SR from 35.4 to 37.4 suggests that our model’s understanding of environmental semantics, enhanced by the Multimodal Large Language Model expert, significantly improves navigation capabilities. Similarly, the SPL improvement from 17.8 to 21.7 indicates that our model’s multi-expert approach is particularly effective in

exploring unknown environments. Moreover, the superiority of our proposed frontier selection method is further confirmed by our model’s improved performance over the single expert’s majority consensus baseline, indicating that the improvements in our navigation strategy generalize to RoboTHOR’s environments as well (Fig. 3).

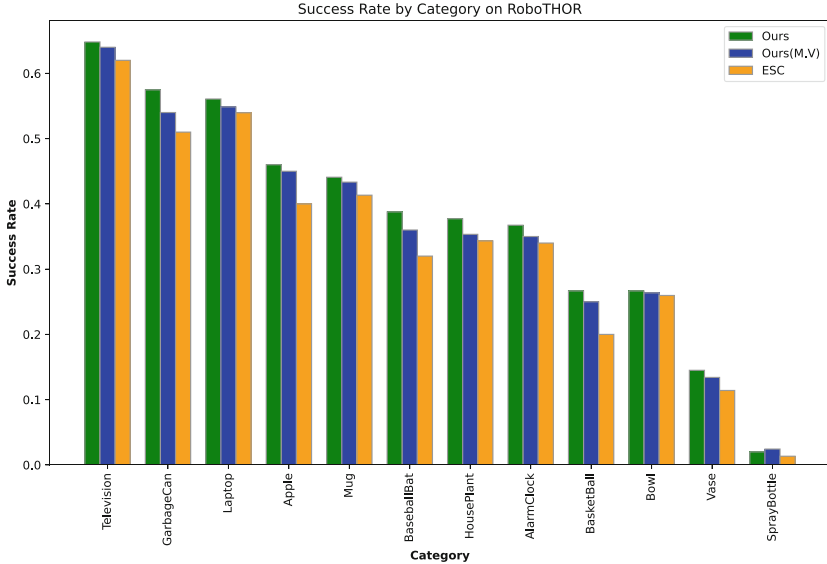


Fig. 3. Success rates for ZS-OGN in twelve target goal categories. The comparison is among our proposed method, our baseline method, and ESC [67]

6 Analysis

6.1 Effect of Reliable Frontier Selection

In this section, we analyze the effectiveness of our proposed frontier selection method within the RoboTHOR environment. For a fair comparison, we evaluate our method alongside ESC and GoW (GLIP on Wheel) [16,67]. GoW is a variant of CoW, utilizing GLIP instead of CLIP. The key difference among these models lies in their frontier selection mechanisms. Our method introduces a novel multi-expert frontier reasoning process combined with an innovative, condensed decision-making approach. In contrast, ESC employs a single expert (GPT-3) for frontier reasoning, while GoW uses a closest frontier strategy.

We visualize the navigation paths of ESC and our method across unknown environments with three room layouts and four object placements each. Figure 4 shows our approach is more reliable and efficient than ESC. In Floor Plan 1, ESC’s zigzagging, highlighted by red ellipses, indicates indecision and poor frontier selection, while

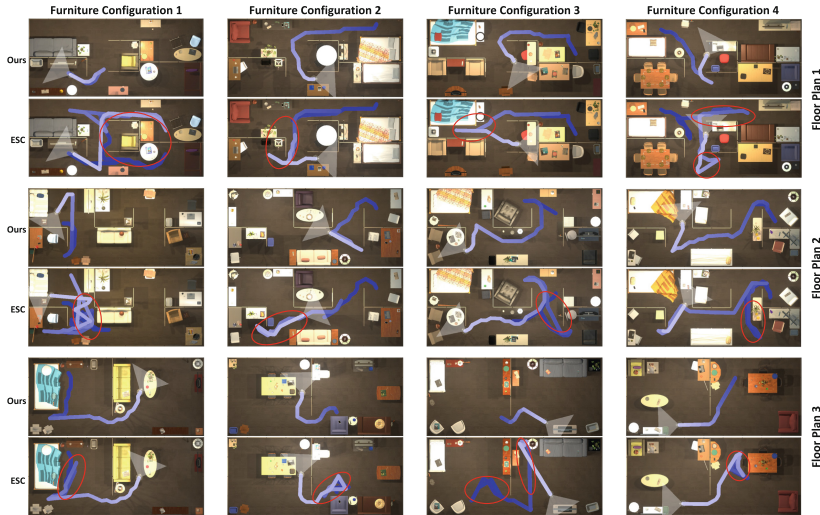


Fig. 4. A comparison of the generated paths to target objects between our proposed method and ESC [67]. Paths generated by our proposed method are more direct and efficient. Instances of zigzagging motion are marked in red ellipses.

Table 2. Comparison of models across different metrics.

Metric	GoW	ESC	Ours
FrontierDist (m)	8.2	7.6	6.8
Exploration (%)	14.3	10.6	9.2
Detection (%)	40.6	40.8	40.6
Planning (%)	12.1	9.5	9.6

our method takes a more direct path, demonstrating better reasoning and efficiency. These results emphasize a more effective frontier selection and a navigation strategy that enhances reliability, reduces detours, and shortens goal-reaching time.

We evaluate the number of actions required by robots to locate target objects across categories in unknown environments using RoboTHOR, comparing our method with ESC and GoW. The box plot in Fig. 5 shows the median actions, 25th and 75th percentiles, and overall range. Our method consistently achieved lower median values across most categories compared to ESC, indicating higher efficiency, despite greater variability. ESC showed more predictable performance but required more actions, while CoW had fewer outliers due to its nearest-frontier strategy but resulted in higher median values. Overall, our method demonstrated more efficient navigation with fewer actions on average.

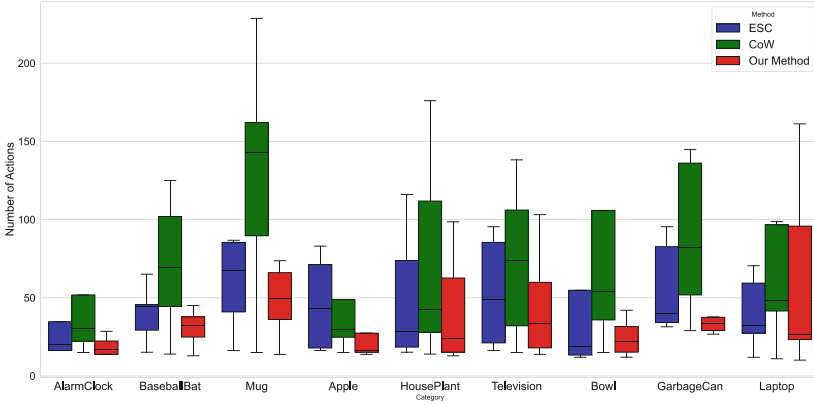


Fig. 5. The comparison of the distribution of the average number of actions to complete the zero-shot OGN across different target objects between our method and ESC. [67]

6.2 Error Analysis

In this section, we conduct an error analysis of our proposed method, ESC and GoW. We follow the standard protocols in [16, 67] to analyze three types of error: 1) *Detection error* happens when the agent either misses the goal or incorrectly believes it has detected the goal. 2) *Planning error* arises when the agent either recognizes the target but cannot reach it or gets stuck without spotting the goal, reflecting the path-planning ability of the system. 3) *Exploration error* occurs when the agent fails to see the goal object due to issues other than planning or detection, assessing its ability to approach the goal.

In Table 2, we note that the detection errors for our method, ESC, and GoW are nearly identical, which is expected given that all three methods employ the same detection head. The similarity of detection errors across all methods suggests that enhancing zero-shot object detection models could be a valuable direction for future ZS-OGN research. Regarding planning errors, our method and ESC exhibit similar rates, as both use FMM for path planning, whereas GoW, which employs A*, shows a higher error rate in this specific dataset. Concerning the Exploration Error, our method outperforms ESC, indicating that it more effectively aids the agent in exploring the environment and approaching the object.

6.3 Effect of Different Experts

We conducted an experiment to validate the effectiveness of various experts. The results are presented in Table 3, where, for the multi-expert method, we adjusted the consensus decision-making process by only proceeding once both experts concurred.

The results indicate that visual cues improve the outcome significantly. Notably, SLE+R2F and SLE+O2F exhibit similar performances, which is expected since visual information can provide insights into both room type and object co-occurrence.

Table 3. Comparison of models across different metrics.

Metric	O2F	R2F	SLE	SLE+R2F	SLE+O2F	O2F+R2F
SPL	17.8	18.2	19.6	21.7	21.7	20.9

7 Conclusion and Future Work

Our study introduces an innovative approach to enhance the reliability of foundation model-driven frontier selection for navigation systems, particularly in zero-shot object goal navigation scenarios. By integrating the Diversified Expert Frontier Analysis (DEFA) and Consensus Decision Making (CDM), our method improves common-sense reasoning for frontier selection by diversifying the reasoning and decision-making process. The CDM component, inspired by the concept of self-consistency, further ensures reliability by requiring majority expert agreement for frontier selection. The promising performance on the RoboTHOR and HM3D datasets, along with a comprehensive analysis against various baselines, demonstrate its effectiveness and reliability in zero-shot navigation tasks. While our approach shows great promise, there are opportunities for future improvement. The system’s complexity introduces computational demands that can be optimized to enhance real-time performance. Additionally, although decision-making consistency has improved, occasional instances of nonsensical or irrelevant reasoning highlight areas where further refinement can increase reasoning accuracy. These enhancements will be key to advancing the system’s efficiency and reliability, particularly in complex and dynamic environments.

References

1. Ali, M., Jardali, H., Roy, N., Liu, L.: Autonomous navigation, mapping and exploration with gaussian processes. In: Robotics: Science and Systems XIX (2023). <https://api.semanticscholar.org/CorpusID:259343521>
2. Ali, M., Liu, L.: Gp-frontier for local mapless navigation. In: 2023 IEEE International Conference on Robotics and Automation (ICRA), pp. 10047–10053. IEEE (2023)
3. Cai, W., et al.: Bridging zero-shot object navigation and foundation models through pixel-guided navigation skill. arXiv preprint [arXiv:2309.10309](https://arxiv.org/abs/2309.10309) (2023)
4. Chaplot, D.S., Gandhi, D., Gupta, S., Gupta, A., Salakhutdinov, R.: Learning to explore using active neural slam. arXiv preprint [arXiv:2004.05155](https://arxiv.org/abs/2004.05155) (2020)
5. Chaplot, D.S., Gandhi, D.P., Gupta, A., Salakhutdinov, R.R.: Object goal navigation using goal-oriented semantic exploration. Adv. Neural. Inf. Process. Syst. **33**, 4247–4258 (2020)
6. Chen, J., Li, G., Kumar, S., Ghanem, B., Yu, F.: How to not train your dragon: training-free embodied object goal navigation with semantic frontiers. arXiv preprint [arXiv:2305.16925](https://arxiv.org/abs/2305.16925) (2023)
7. Chen, P., et al.: Weakly-supervised multi-granularity map learning for vision-and-language navigation. Adv. Neural. Inf. Process. Syst. **35**, 38149–38161 (2022)
8. Chowdhery, A., et al.: Palm: scaling language modeling with pathways. J. Mach. Learn. Res. **24**(240), 1–113 (2023)
9. Constantinides, G.M., Malliaris, A.G.: Portfolio theory. Handbooks Oper. Res. Manag. Sci. **9**, 1–30 (1995)

10. Deitke, M., et al.: Robothor: an open simulation-to-real embodied ai platform. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3164–3174 (2020)
11. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: pre-training of deep bidirectional transformers for language understanding. arXiv preprint [arXiv:1810.04805](https://arxiv.org/abs/1810.04805) (2018)
12. Dorbala, V.S., Mullen Jr, J.F., Manocha, D.: Can an embodied agent find your “cat-shaped mug”? IIm-based zero-shot object navigation. arXiv preprint [arXiv:2303.03480](https://arxiv.org/abs/2303.03480) (2023)
13. Dorbala, V.S., Sigurdsson, G.A., Thomason, J., Piramuthu, R., Sukhatme, G.S.: Clip-nav: using clip for zero-shot vision-and-language navigation. In: Workshop on Language and Robotics at CoRL 2022 (2022)
14. Driess, D., et al.: Palm-e: an embodied multimodal language model. arXiv preprint [arXiv:2303.03378](https://arxiv.org/abs/2303.03378) (2023)
15. Gadre, S.Y., Wortsman, M., Ilharco, G., Schmidt, L., Song, S.: Clip on wheels: zero-shot object navigation as object localization and exploration, **3**(4), 7 (2022). arXiv preprint [arXiv:2203.10421](https://arxiv.org/abs/2203.10421)
16. Gadre, S.Y., Wortsman, M., Ilharco, G., Schmidt, L., Song, S.: Cows on pasture: base-lines and benchmarks for language-driven zero-shot object navigation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 23171–23181 (2023)
17. Huang, H., Yuan, S., Wen, C., Hao, Y., Fang, Y.: Noisy few-shot 3d point cloud scene segmentation. In: 2024 IEEE International Conference on Robotics and Automation (ICRA), pp. 11070–11077. IEEE (2024)
18. Huang, S., et al.: Language is not all you need: aligning perception with language models. arXiv preprint [arXiv:2302.14045](https://arxiv.org/abs/2302.14045) (2023)
19. Huang, W., Abbeel, P., Pathak, D., Mordatch, I.: Language models as zero-shot planners: extracting actionable knowledge for embodied agents. In: International Conference on Machine Learning, pp. 9118–9147. PMLR (2022)
20. Huang, W., et al.: Grounded decoding: guiding text generation with grounded models for robot control. arXiv preprint [arXiv:2303.00855](https://arxiv.org/abs/2303.00855) (2023)
21. Jadidi, M.G., Miró, J.V., Valencia, R., Andrade-Cetto, J.: Exploration on continuous gaussian process frontier maps. In: 2014 IEEE International Conference on Robotics and Automation (ICRA), pp. 6077–6082. IEEE (2014)
22. Jiang, Y., et al.: Vima: general robot manipulation with multimodal prompts. In: NeurIPS 2022 Foundation Models for Decision Making Workshop (2022)
23. Kahn, G., Villafior, A., Ding, B., Abbeel, P., Levine, S.: Self-supervised deep reinforcement learning with generalized computation graphs for robot navigation. In: 2018 IEEE International Conference on Robotics and Automation (ICRA), pp. 5129–5136. IEEE (2018)
24. Karnan, H., Warnell, G., Xiao, X., Stone, P.: Voila: visual-observation-only imitation learning for autonomous navigation. In: 2022 International Conference on Robotics and Automation (ICRA), pp. 2497–2503. IEEE (2022)
25. Kojima, T., Gu, S.S., Reid, M., Matsuo, Y., Iwasawa, Y.: Large language models are zero-shot reasoners. *Adv. Neural. Inf. Process. Syst.* **35**, 22199–22213 (2022)
26. Komorowski, J.: Minkloc3d: point cloud based large-scale place recognition. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 1790–1799 (2021)
27. Koubaa, A.: Rosgpt: Next-generation human-robot interaction with chatgpt and ros. Preprints (2023). <https://doi.org/10.20944/preprints202304.0827.v3>
28. Krause, S., Stolzenburg, F.: Commonsense reasoning and explainable artificial intelligence using large language models. In: European Conference on Artificial Intelligence, pp. 302–319. Springer, Heidelberg (2023)

29. Li, L.H., et al.: Grounded language-image pre-training. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10965–10975 (2022)
30. Majumdar, A., Aggarwal, G., Devnani, B., Hoffman, J., Batra, D.: Zson: zero-shot object-goal navigation using multimodal goal embeddings. *Adv. Neural. Inf. Process. Syst.* **35**, 32340–32352 (2022)
31. Majumdar, A., Shrivastava, A., Lee, S., Anderson, P., Parikh, D., Batra, D.: Improving vision-and-language navigation with image-text pairs from the web. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) *ECCV 2020. LNCS*, vol. 12351, pp. 259–274. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58539-6_16
32. Mangram, M.E.: A simplified perspective of the markowitz portfolio theory. *Glob. J. Bus. Res.* **7**(1), 59–70 (2013)
33. Markowitz, H.M.: Foundations of portfolio theory. *J. Financ.* **46**(2), 469–477 (1991)
34. Markowitz, H.M.: Portfolio theory: as i still see it. *Annu. Rev. Financ. Econ.* **2**(1), 1–23 (2010)
35. OpenAI: Gpt-4 technical report. arXiv preprint [arXiv:2303.08774](https://arxiv.org/abs/2303.08774) (2023)
36. OpenAI: Introducing chatgpt (2023). <https://openai.com/blog/chatgpt>. Accessed 2 Aug 2023
37. Radford, A., et al.: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning, pp. 8748–8763. PMLR (2021)
38. Ramakrishnan, S.K., Chaplot, D.S., Al-Halah, Z., Malik, J., Grauman, K.: Pon: potential functions for objectgoal navigation with interaction-free learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 18890–18900 (2022)
39. Ramakrishnan, S.K., et al.: Habitat-matterport 3d dataset (HM3d): 1000 large-scale 3d environments for embodied AI. In: Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (2021). <https://arxiv.org/abs/2109.08238>
40. Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.: Proximal policy optimization algorithms. arXiv preprint [arXiv:1707.06347](https://arxiv.org/abs/1707.06347) (2017)
41. Sethian, J.A.: A fast marching level set method for monotonically advancing fronts. *Proc. Natl. Acad. Sci.* **93**(4), 1591–1595 (1996)
42. Shah, D., Equi, M.R., Osiński, B., Xia, F., Ichter, B., Levine, S.: Navigation with large language models: semantic guesswork as a heuristic for planning. In: Conference on Robot Learning, pp. 2683–2699. PMLR (2023)
43. Shah, D., Osiński, B., Levine, S., et al.: Lm-nav: robotic navigation with large pre-trained models of language, vision, and action. In: Conference on Robot Learning, pp. 492–504. PMLR (2023)
44. Silver, D., Bagnell, J., Stentz, A.: High performance outdoor navigation from overhead data using imitation learning. In: Robotics: Science and Systems IV, Zurich, Switzerland, vol. 1 (2008)
45. Song, C.H., Wu, J., Washington, C., Sadler, B.M., Chao, W.L., Su, Y.: Llm-planner: few-shot grounded planning for embodied agents with large language models. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 2998–3009 (2023)
46. Suzuki, S., Takeno, S., Tamura, T., Shitara, K., Karasuyama, M.: Multi-objective bayesian optimization using pareto-frontier entropy. In: International Conference on Machine Learning, pp. 9279–9288. PMLR (2020)
47. Wang, W., Haddow, B., Birch, A., Peng, W.: Assessing the reliability of large language model knowledge. arXiv preprint [arXiv:2310.09820](https://arxiv.org/abs/2310.09820) (2023)
48. Wang, X., et al.: Self-consistency improves chain of thought reasoning in language models. arXiv preprint [arXiv:2203.11171](https://arxiv.org/abs/2203.11171) (2022)
49. Wei, J., et al.: Chain-of-thought prompting elicits reasoning in large language models. *Adv. Neural. Inf. Process. Syst.* **35**, 24824–24837 (2022)

50. Wijmans, E., et al.: Dd-ppo: Learning near-perfect pointgoal navigators from 2.5 billion frames. In: International Conference on Learning Representations (2019)
51. Wöhlke, J., Schmitt, F., van Hoof, H.: Hierarchies of planning and reinforcement learning for robot navigation. In: 2021 IEEE International Conference on Robotics and Automation (ICRA), pp. 10682–10688. IEEE (2021)
52. Wu, P., et al.: Voronav: voronoi-based zero-shot object navigation with large language model. arXiv preprint [arXiv:2401.02695](https://arxiv.org/abs/2401.02695) (2024)
53. Xia, Y., et al.: Casspr: cross attention single scan place recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 8461–8472 (2023)
54. Xia, Y., et al.: Soe-net: a self-attention and orientation encoding network for point cloud based place recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11348–11357 (2021)
55. Yamauchi, B.: A frontier-based approach for autonomous exploration. In: Proceedings 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation CIRA'97.'Towards New Computational Principles for Robotics and Automation, pp. 146–151. IEEE (1997)
56. Yang, M.S., Schuurmans, D., Abbeel, P., Nachum, O.: Chain of thought imitation with procedure cloning. *Adv. Neural. Inf. Process. Syst.* **35**, 36366–36381 (2022)
57. Yao, S., et al.: Tree of thoughts: deliberate problem solving with large language models. arXiv preprint [arXiv:2305.10601](https://arxiv.org/abs/2305.10601) (2023)
58. Yao, S., et al.: React: synergizing reasoning and acting in language models. arXiv preprint [arXiv:2210.03629](https://arxiv.org/abs/2210.03629) (2022)
59. Yuan, S., Fang, Y.: Ross: Robust learning of one-shot 3d shape segmentation. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 1961–1969 (2020)
60. Yuan, S., Shafique, M., Baghdadi, M.R., Khorrami, F., Tzes, A., Fang, Y.: Zero-shot object navigation with vision-language foundation models reasoning. In: 2024 10th International Conference on Automation, Robotics and Applications (ICARA), pp. 501–505. IEEE (2024)
61. Zhang, Z., Zhang, A., Li, M., Smola, A.: Automatic chain of thought prompting in large language models. arXiv preprint [arXiv:2210.03493](https://arxiv.org/abs/2210.03493) (2022)
62. Zhao, Q., Zhang, L., He, B., Liu, Z.: Semantic policy network for zero-shot object goal visual navigation. *IEEE Rob. Autom. Lett.* (2023)
63. Zhao, Q., Zhang, L., He, B., Qiao, H., Liu, Z.: Zero-shot object goal visual navigation. In: 2023 IEEE International Conference on Robotics and Automation (ICRA), pp. 2025–2031. IEEE (2023)
64. Zheng, K., et al.: Jarvis: a neuro-symbolic commonsense reasoning framework for conversational embodied agents. arXiv preprint [arXiv:2208.13266](https://arxiv.org/abs/2208.13266) (2022)
65. Zhou, B., Zhang, Y., Chen, X., Shen, S.: Fuel: fast uav exploration using incremental frontier structure and hierarchical planning. *IEEE Rob. Autom. Lett.* **6**(2), 779–786 (2021)
66. Zhou, G., Hong, Y., Wu, Q.: Navgpt: explicit reasoning in vision-and-language navigation with large language models. arXiv preprint [arXiv:2305.16986](https://arxiv.org/abs/2305.16986) (2023)
67. Zhou, K., et al.: Esc: exploration with soft commonsense constraints for zero-shot object navigation. arXiv preprint [arXiv:2301.13166](https://arxiv.org/abs/2301.13166) (2023)



Reliable Semantic Understanding for Real World Zero-Shot Object Goal Navigation

Halil Utku Unlu¹(✉) , Shuaihang Yuan^{2,3,4} , Congcong Wen^{2,4} ,
Hao Huang^{2,4} , Anthony Tzes^{2,3} , and Yi Fang^{2,3,4} 

¹ Electrical and Computer Engineering Department, New York University (NYU),
Brooklyn, NY 11201, USA

utku@nyu.edu

² NYU Abu Dhabi, Electrical Engineering, Abu Dhabi, UAE

³ Center for Artificial Intelligence and Robotics, NYU Abu Dhabi, UAE

⁴ Embodied AI and Robotics (AIR) Lab, NYU, Abu Dhabi, UAE

Abstract. We introduce an innovative approach to advancing semantic understanding in zero-shot object goal navigation (ZS-OGN), enhancing the autonomy of robots in unfamiliar environments. Traditional reliance on labeled data has been a limitation for robotic adaptability, which we address by employing a dual-component framework that integrates a GLIP Vision Language Model for initial detection and an Instruction-BLIP model for validation. This combination not only refines object and environmental recognition but also fortifies the semantic interpretation, pivotal for navigational decision-making. Our method, rigorously tested in both simulated and real-world settings, exhibits marked improvements in navigation precision and reliability.

Keywords: Zero-shot Navigation · Object Goal Navigation · Semantic Scene Understanding · Vision-Language Models · Safe Navigation

1 Introduction

Object navigation is crucial for the autonomous operation of robots, which has traditionally depended on extensive labeled visual data. In Object Goal Navigation (OGN), the objective is to navigate uncharted environments in search of a specified, yet initially unseen, target object. Traditional methodologies in this domain have predominantly hinged on visual cues through either imitation [11, 20] or reinforcement learning [10, 22] techniques, necessitating substantial data and annotations for effective training, thereby constraining their utility in diverse, real-world settings.

This necessity has catalyzed a paradigm shift towards ZS-OGN strategies, designed to imbue robots with the capacity for immediate adaptation to novel objects and contexts [6, 17, 27, 28]. Zero-shot object goal navigation (ZS-OGN) [2–4, 25, 29–31] equips robots with the ability to identify and interact with objects they have not previously encountered, leveraging sensory inputs and exploratory behaviors [17]. This process allows robots to transcend the limitations of their

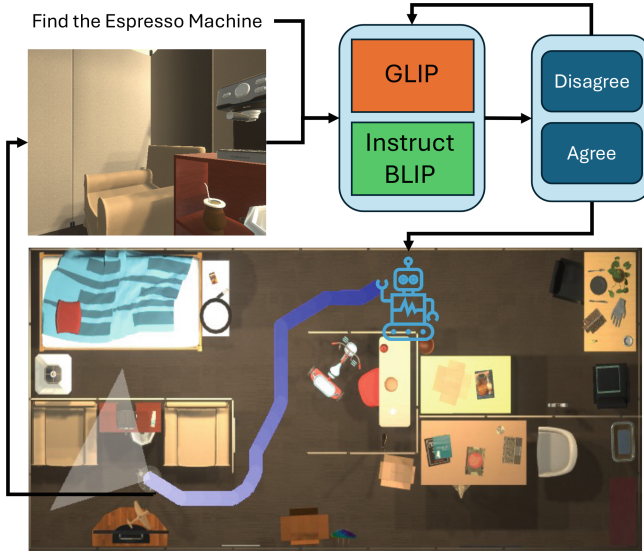


Fig. 1. Illustration of the key component of our method ZS-OGN. The process begins with the GLIP Vision Language Model detecting the target object, in this case, an espresso machine. Subsequently, the InstructBLIP model evaluates the detection, either confirming the GLIP’s proposal (‘Agree’) or not (‘Disagree’), which influences the continuation or adjustment of the navigational plan.

training data, enhancing their versatility and expanding their potential for operation in dynamic and novel settings [8, 31]. Advancements in ZS-OGN bring us closer to developing robots that can comprehend and engage with a more diverse array of environments, representing a notable evolution in the field of robotics.

The existing ZS-OGN framework can be broken down into distinct components: semantic understanding, high-level exploration, and low-level navigation [8, 19, 24, 31]. Semantic understanding is crucial as it provides the robot with the ability to discern unseen objects through environmental observations, and it lays the groundwork for subsequent exploration and navigation strategies. This underscores its critical role and highlights the significance of advancing semantic understanding within ZS-OGN.

Addressing the inherent limitations of direct visual embedding reliance, as observed in the ZER framework’s [1] two-stage process which begins with an ImageNav agent’s foundational training, subsequent innovations have pivoted towards the exploitation of multimodal, semantic embedding spaces. This shift not only facilitates the comprehension of objects articulated in natural language but also circumvents the semantic void typical of image-goal embeddings bereft of semantic annotations. Alternative sensor modalities, such as LiDAR point cloud data, can enhance the semantic scene understanding via geometry-based reasoning [13, 23].

To enhance the semantic knowledge of the robot within its environment, the integration of vision-language models into robotic systems has proved highly effective. These models merge the perceptual power of visual data with the deep contextual insights offered by linguistic information, creating a robust framework for interpreting complex environments. A notable innovation in this domain is VLMaps [9], which integrates pre-trained visual-language features with 3D environmental reconstructions to improve spatial and semantic understanding and facilitate more natural interaction with the environment. The versatility of vision-language models in ZS-OGN [8, 17] is further demonstrated by the use of CLIP [7], a pre-trained model that excels in diverse navigation scenarios. Moreover, ESC [31] proposes a commonsense reasoning for an efficient frontier selection during robot exploration and enhances the CLIP-based model using a GLIP [15], which represents a significant leap in scene comprehension. These advancements highlight the essential role of vision-language models in advancing the semantic understanding aspect of ZS-OGN and pushing robotic navigation toward higher levels of intelligence and adaptability. Despite such progress, the success rates for ZS-OGN approaches have not been fully satisfactory, often due to errors in object detection—either false identifications of visible goal objects or mistaken detections of nonexistent ones. This highlights the need for a more reliable semantic scene understanding framework that does not rely on further training. Addressing this gap, our research aims to develop a robust semantic understanding framework to reduce detection errors and enhance the precision and effectiveness of robot navigation.

Further advancements, as demonstrated by EmbCLIP and CoW [8, 12], incorporate CLIP for enhanced vision-and-language navigation and object goal navigation, respectively, leveraging Frontier-Based Exploration (FBE) [24] to effectively demarcate and traverse the boundary between explored and unexplored territories. Such methodologies underscore a significant leap forward in navigating autonomously through open-world settings with unprecedented efficiency and adaptability.

Many of the aforementioned papers conduct studies in simulated or pre-recorded data. The common benchmarks and datasets provide photorealistic data, and the rate of improvement in the field is astounding. However, the real-life realizations of such systems are lacking. Attempts to run the proposed algorithms require many additional considerations for safety, one of which is the lack of accurate positioning information in indoor environments. To that end, we adapted parts of the ZS-OGN framework to be resilient to real-world uncertainty, noise, and latency, by integrating a safe global path planner and a local planner with dynamic obstacle avoidance. To the best of our knowledge, this work is the first in a ZS-OGN framework deployed in a real-life task.

In this paper, we present a novel approach to improve the semantic understanding of ZS-OGN through a two-part semantic pipeline, as shown in Fig. 1, and demonstrate its efficacy in both simulated and real-world scenarios. The proposed framework consists of the GLIP Vision Language Model (VLM), responsible for the initial object and context detection, and InstructionBLIP, a validating

VLM, to verify GLIP’s detections. These components collaboratively enhance the accuracy of identifying objects and their surroundings.

The contributions of this work are:

- A “Doubly Right” semantic understanding framework for Zero-Shot Object Goal Navigation (ZS-OGN), which employs a dual verification system to enhance the reliability and accuracy of object detection in unfamiliar environments,
- State-of-the-art performance on common simulation platforms, indicating that the proposed framework effectively improves the semantic understanding necessary for ZS-OGN,
- Implementation of a safety-optimal path planning algorithm to minimize chances of collision in real-world conditions, and
- Experimental validation of the ZS-OGN pipeline in a real-world apartment

The rest of the paper is structured as follows: Problem details are provided in Sect. 2. Implementation details of the proposed system are given in Sect. 3, and simulation and real-world study results are shown in Sects. 4 and 5, respectively.

2 Problem Statement

In ZS-OGN tasks, a robot is tasked with locating the target object, denoted as g_i , which it has not previously encountered within an unexplored environment s_i , and this must be accomplished without prior navigational data training. At each timestep t , the robot captures a color image I_t , depth data d_t , and its own pose—comprising its coordinates (x_t, y_t) and heading θ_t . The robot integrates its pose data over time to compute its current location. Utilizing the data from each timestep, the robot selects an action a from a set of possible actions \mathcal{A} which includes actions such as advancing, rotating left, rotating right, and halting. The navigation process is deemed successful when the robot elects to halt within a specified proximity to the target object.

3 Approach

In this section, we delineate our framework for ZS-OGN. As depicted in Fig. 2, our approach incorporates a frontier-based exploration method, which is widely recognized in the field of ZS-OGN. The process initiates with the transformation of the input image into semantic data, subsequently integrating this information into a semantic map. The framework then harnesses the commonsense reasoning capabilities of large language models to determine the subsequent frontier for exploration. Next, the Fast Marching Method is employed to compute the shortest path from the agent’s current location to the designated target. Lastly, our innovative ‘Doubly Right’ semantic understanding framework is applied to verify the accurate detection of the target object.

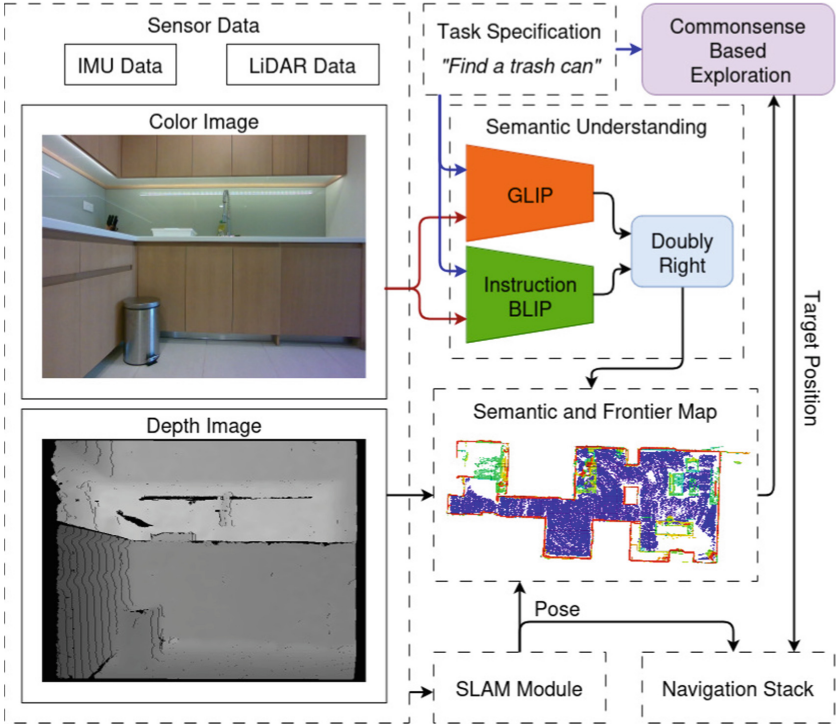


Fig. 2. The flow of our proposed method in a real-world scenario.

3.1 Doubly Right Semantic Understanding

In developing a robust framework for ZS-OGN, we introduce a novel method called “Doubly Right,” which incorporates a dual verification system using Vision-Language Models (VLMs) to mitigate common detection errors. This approach, provided in Algorithm 1, seeks to increase the reliability of object detection and room recognition within navigational tasks.

The process begins with an Initiator VLM. Following the ESC framework, we employ the Grounded Language-Image Pre-training (GLIP) model for the preliminary identification of objects and rooms in environmental representations. The GLIP model processes visual inputs using zero-shot learning capabilities to generalize its detection beyond the training data through natural language prompts. When the GLIP model detects an object or room type, it assigns a provisional label and triggers the Validator VLM, InstructionBLIP, to assess the detection for accuracy. For each visual input I_t at time t , the framework executes an initial detection with the GLIP model:

$$O_{init} = \text{GLIP}(I_t, P_o), \quad (1)$$

where O_{init} denotes the set of detected target objects by GLIP, and P_o represent the respective object prompting phrases applied to GLIP. Regardless of whether

O_{init} is an empty set or contains detections, the Validator VLM, Instruction-BLIP, then reviews these initial findings. It cross-references the output from the GLIP with the task instructions to confirm their validity. If Instruction-BLIP identifies a need for reassessment, it advises that the environment be re-examined, indicating potential discrepancies in the initial detection. The algorithm will then return to the Initiator VLM to conduct a reevaluation. In cases where InstructionBLIP does not advise further inspection, a flag *Goal* is set to **True**, indicating that the target object has been reliably detected and the navigational task is complete. This flag’s status is critical as it confirms the end of the navigational sequence, ensuring that erroneous detections are addressed and the accuracy of the navigational decision-making is improved.

The validation process serves as a double-check mechanism, confirming the detected objects and their spatial contexts are indeed relevant and accurate for the navigational task at hand. By implementing this two-fold verification strategy, our framework aims to reduce detection errors that impede the success of autonomous navigation systems. The approach ensures that navigation decisions are based on a reliable semantic understanding of the environment, enhancing the system’s performance in novel or previously unseen settings.

Algorithm 1. Doubly Right: Zero-Shot Object Goal Navigation Framework

- 1: **Input:** Visual input I_t at time t , object prompting phrases P_o , validation prompting P_v
 - 2: **Output:** a flag *Goal* indicates whether the target object is found.
 - 3:
 - 4: **Initiator VLM:** Employ GLIP model following the ESC framework
 - 5: $O_{init} \leftarrow \text{GLIP}(I_t, P_o)$ ▷ Detect objects and rooms with GLIP
 - 6: **Validator VLM:** Invoke InstructionBLIP model
 - 7: **if** INSTRUCTIONBLIP(I_t, O_{init}, P_v) advises reassessment **then**
 - 8: **Goto** Initiator VLM ▷ Reassess the detections
 - 9: **else**
 - 10: $Goal = \text{True}$
-

3.2 Semantic and Frontier Map

Our approach for ZS-OGN follows the ESC framework to construct a semantic navigation map critical for autonomous navigation. Utilizing depth input d_t and the agent’s 2D pose $[x_t \ y_t \ \theta_t]^\top \in \mathbb{R}^2 \times \mathbb{S}$, we generate a foundational 2D navigation map. The GLIP model is then applied to enrich this map semantically by detecting objects and room types via zero-shot learning:

$$S_{map} = f(d_t, x_t, y_t, \theta_t, \mathcal{R}_i), \quad (2)$$

where S_{map} represents the semantic map, d_t is the depth input at time t , and \mathcal{R}_i symbolizes the environmental representation obtained from the GLIP model including room types and commonly occurring objects.

Then, the frontier map is constructed to delineate the boundaries for exploration. The process begins by identifying the periphery of the unoccupied area within the navigation map, forming candidate frontiers between free and unknown spaces [18]. These candidates are scrutinized by the Commonsense Policy π_{cs} to prioritize the next exploration target:

$$F_t = \pi_{cs}(\widehat{F}), \quad (3)$$

where \widehat{F} denotes the set of candidate frontiers, and F_t is the selected target frontier.

The semantic navigation map S_{map} is critical for the understanding of environmental semantics, enhancing the robot’s interaction with the environment. Concurrently, the frontier map guides the exploratory progression, enabling systematic and informed exploration of new areas. Together, these maps form a comprehensive navigation system that supports autonomous agents in complex and dynamic settings.

3.3 Commonsense Policy for Exploration

Our navigation framework assimilates a commonsense reasoning module, adapted from the established ESC framework, to complement the autonomous navigation process. This module leverages the semantic understanding derived from our semantic navigation map to make contextual inferences, enhancing navigational decision-making.

The adopted ESC commonsense module ESC_{cs} analyzes spatial relationships and object functionalities, drawing on a knowledge base of object-room correlations and navigational heuristics. Such inferences enable the anticipation of environmental elements indirectly indicated by the current sensory data:

$$C_i = ESC_{cs}(S_{map}), \quad (4)$$

where C_i denotes the reasoned inferences based on the semantic map S_{map} . These inferences feed into the Global Commonsense Policy, informing frontier selection for targeted exploration.

By incorporating the ESC commonsense reasoning, the framework attains a sophisticated level of environmental interpretation, pivotal for navigating through dynamic spaces. This integration, although not our core innovation, significantly enhances the overall efficacy of the navigation system.

4 Simulation Studies

4.1 Dataset

RoboTHOR. The RoboTHOR dataset has been developed to validate navigation systems within authentic real-world scenarios. This benchmark features 89

meticulously designed apartment scenes, augmented by a comprehensive collection of 731 unique objects. Adhering to protocols from previous studies [8, 31], our proposed method is subjected to an assessment comprising more than 1,800 validation episodes across 15 different environments. The evaluation is focused on 12 principal object categories crucial for navigation.

PASTURE. The Pasture dataset augments RoboTHOR’s validation scenarios with additional object variations and complexity across 2,520 navigation tasks, enhancing the robustness of navigational model testing. It introduces an intricate mix of object sizes, colors, and materials, alongside detailed spatial relationships, to create a more challenging benchmark that closely mimics real-world conditions.

4.2 Metrics

Consistent with established benchmarks in the field, we utilize Success Rate (SR) and Success Weighted by Path Length (SPL) to evaluate agent performance. SR measures the agent’s accuracy in reaching the target within a meter’s distance, presented as a percentage, where a higher rate indicates better performance. Conversely, SPL gauges the efficiency of the navigation, comparing the agent’s actual traveled path with the ideal shortest path, thus reflecting the agent’s navigational efficacy and the optimization of its chosen route.

4.3 Baselines

In our experiment, we measure the performance of our method against baseline models such as CoW and ESC. CoW, designed for Zero-Shot Object Navigation (ZS-OGN), leverages CLIP for dynamic object detection, enabling localization without prior navigational training. We further evaluate CoW’s efficacy by comparing it with its variants that utilize different CLIP-based localization strategies, namely CLIP-Ref, CLIP-Patch, CLIP-Grad, MDETR, and OWL. Additionally, we assess ESC, which incorporates commonsense knowledge into navigation actions through a pre-trained vision and language model, enhancing the agent’s ability to navigate and reason about objects and rooms in unseen environments.

4.4 Results

The performance data in Table 1 reveals that our method leads with an average Success Rate (SR) of 23.0% and Success Weighted by Path Length (SPL) of 13.7, indicating effective and efficient navigation in diverse scenarios. It surpasses others, particularly in challenging categories involving uncommon objects and hidden distractions, underscoring its robustness and sophisticated semantic understanding. Notably, the OWL model also demonstrates commendable SRs, especially in environments with spatial distractions. In contrast, the CoW model and other CLIP-based methods display more modest performance, highlighting the complexities these models face in the richly varied navigation tasks

Table 1. Zero-shot object goal navigation results on PASTURE [8] benchmarks.

Method	Uncom.	Appear.	Space	Appear. distract	Space distract	Hid.	Hid. distract	Average	
	SR	SR	SR	SR	SR	SR	SR	SPL	SR
CoW CLIP-Ref.	3.6	2.8	2.8	3.1	3.3	4.7	5.0	1.7	2.5
CLIP-Patch	18.1	13.3	13.3	10.8	10.8	17.5	17.8	9.0	14.2
CLIP-Grad.	16.1	11.9	11.7	9.7	10.3	14.4	16.1	9.2	12.9
MDETR	3.1	7.2	5.0	7.2	4.7	8.1	8.9	5.4	6.3
OWL	32.8	26.9	19.4	19.4	16.1	19.2	15.8	12.6	21.1
Ours	33.0	29.0	22.4	19.7	18.2	20.8	17.6	13.7	23.0

of the Pasture dataset, which is designed to evaluate navigation models with its intricate array of objects. In addition to the evaluation on the Pasture Dataset, We conduct further experiments on the RoboTHOR dataset. The results on the RoboTHOR dataset, as presented in Table 2 reveal that our method stands out with superior performance, achieving the highest Success Weighted by Path Length (SPL) at 18.3 and Success Rate (SR) at 35.2, demonstrating exceptional navigation efficacy among the 12 crucial object categories across 15 environments.

Table 2. Zero-shot object navigation results on RoboTHOR [5] benchmarks. * denotes the reproduced result using the official implementation.

Method	Performance	
	SPL	SR
CLIP-Ref. [8]	1.0	1.8
CLIP-Patch. [8]	7.7	15.3
CLIP-Grad. [8]	7.4	12.1
MDETR. [8]	8.4	9.9
CLIP-OWL. [8]	13.4	21.9
ESC* [31]	18.2	34.5
Ours	18.3	35.2

5 Experimental Studies

The proposed ZS-OGN pipeline was validated in real-world scenarios on real robot hardware. Following the procedures from the simulation environments and benchmarks, the robotic agent was tasked with navigation near various household objects with no prior information about the environment or the actual object.

Additional modifications were necessary to allow the system to operate in a real-world environment to improve overall system robustness and safety in the

face of noise and uncertainties. All such modifications are mentioned in their respective sections below.

5.1 Environment

The layout of the apartment in which the tests are performed is provided in Fig. 3, along with the locations of sample items to be detected and the first-person view from the robot when they are detected.



Fig. 3. Layout of the apartment used in the experiment, along with the first-person views of the detected mug (red), remote (green), and trash can (blue). The starting location is marked with a star. (Color figure online)

The placement of the objects was guided by common sense: a TV remote is expected to be found near the TV in a living room, and a garbage can can be found in a bathroom or a kitchen.

5.2 Robotic Platform

For the study, a Unitree B1 quadruped robot, equipped with a LiDAR (Pandar XT16) and an RGBD camera with IMU (Realsense D455) was used. Simultaneous localization and mapping (SLAM), path planning, navigation, and low-level control were executed entirely on the robot's internal CPU, whereas the common-sense module for identifying goal location was run on the additional computer, attached to the robot. A photo of the vehicle is provided in Fig. 4.

The platform additionally streams depth images from 5 extra depth cameras, inertial measurements from an internal IMU, and a proprioceptive odometry estimate. However, none of the aforementioned data is used for this study.



Fig. 4. A photo of the Unitree B1 robotic platform.

5.3 Odometry and Mapping

The related literature for ZS-OGN utilizes RGBD-based mapping schemes that assume the agent pose is always available. However, without external infrastructure (e.g. GNSS for outdoors, motion capture for indoors, or dead-reckoning in both) the agent pose is neither readily available nor always accurate.

RTAB-Map [14] was selected as the main driver for pose estimation, mapping, and localization. The robotic platform uses the onboard LiDAR and IMU sensors to estimate its odometry using RTAB-Map’s ICP odometry module. With the addition of color images from the RGBD sensor, RTAB-Map’s SLAM module is used for localization and mapping with global loop closure identification. Pose estimation and mapping are restricted to 3DoF (xy -coordinates and yaw angle θ) since the environment is a single-story indoor location.

5.4 Safe Path Planning

Many of the ZS-OGN frameworks output an action from the action space, such as “move forward” or “turn left” to carry out the navigation task. While such schemes function well in simulated environments, in which the actions can be carried out instantaneously and precisely, the real-world interaction needs to perform a trade-off between speed and accuracy. discrete action spaces prevent the agents from executing complex maneuvers (e.g. go through a narrow opening for a robot with non-circular footprint). Finally, inherent noise in sensors and estimation for robotic platforms necessitate enhanced safety precautions to prevent inadvertent collisions with the environment. While a distance-optimal or effort-optimal path could take the robot to its desired location with high efficiency, the robot usually needs to navigate close to the obstacles in the environment. Latency in state estimation, control, and/or the actual movement can lead to collisions.

To that end, a medial-axis-based path planning algorithm, as proposed in [21], was adopted. In this scheme, the agent is directed to follow a path that is maximally distant from any obstacles in the environment. Contrary to the original implementation, the system was adapted to use a 2D cost map for planning, to allow the paths to extend the unknown space, albeit at a high cost.

Let $\mathcal{M} \subseteq \mathbb{Z}_+^2$ denote the coordinates of the 2D occupancy grid map, with the occupancy encoded as an integer through a mapping $C : \mathcal{M} \rightarrow [0, 255]$. Since the cost map obtained from RTAB-Map is metric, there exists a surjective mapping between the real, metric coordinates $p \in \mathbb{R}^2$ and the map coordinates $\bar{p} \in \mathbb{Z}_+^2$. Let p_r (\bar{p}_r) and p_g (\bar{p}_g) denote the metric (map) coordinate of the robot's current position and target location, respectively.

The traversable set is a subset of the 2D cost map that does not map to a lethal cost (i.e. fully occupied or dangerous to approach) in the original cost map. The exact encoding may differ between different implementations, but defining the lethal cost set as $C_l \subset [0, 255]$, the traversable region is defined formally as

$$\mathcal{M}_t = \{p \in \mathbb{Z}_+^2, C(p) \notin C_l\}, \quad (5)$$

with a cost to traverse the coordinate determined via the mapping C . The traversable set need not be connected. However, only the portion of the traversable set that contains the current coordinate of the robot is of interest, which will be denoted as \mathcal{M}_t for simplicity.

The medial axis for this context is defined as the set of coordinates $\mathcal{S} \subseteq \mathcal{M}_t$ that are equidistant to multiple closest points in the boundary of the traversable region. Many implementations exist, but the thinning algorithm implementation in OpenCV [26] is used over a binary image representation in this work.

Finally, let $\bar{p}_{\text{entry}} \in \mathcal{S}$ be the closest coordinate to \bar{p}_r with a linear path completely contained in \mathcal{M}_t . Similarly, let $\bar{p}_{\text{exit}} \in \mathcal{S}$ be the closest coordinate to \bar{p}_g in the same fashion.

Given the above definitions, the path planning algorithm generates a path for navigation from the current position to the target position in 3 different segments:

- A linear path $\mathcal{P}_{\text{entry}}$ from \bar{p}_r to \bar{p}_{entry} ,
- An ordered set $\mathcal{P}_{\text{axis}} \subseteq \mathcal{S}$, comprised of a series of adjacent coordinates that connect \bar{p}_{entry} to \bar{p}_{exit} , and
- A linear $\mathcal{P}_{\text{exit}}$ from \bar{p}_{exit} to \bar{p}_g .

A sample demonstration of the algorithm on a cost map computed during the experiment is provided in Fig. 5

Under the appropriate handling of the cost map to account for the robot dimensions (e.g. circular footprint with correct inflation parameters), the above algorithm generates a path that minimizes the chance of collisions with the environment and provides a margin of error for uncertainty and latency.



Fig. 5. Sample path generated from the medial axis algorithm. The path (light gray) connects the current (magenta) and target (dark blue) coordinates through the medial axis (dark gray), entirely within the traversable region (green) and maximizing distance from obstacles (red). Black denotes lethal cost. (Color figure online)

5.5 Implementation Details

The path planning algorithm as proposed was implemented as a planner plugin within Nav2 [16] framework, an open-source navigation stack with production deployments.

Since the robot doesn't have a circular footprint, the cost map calculation, and in return the safe path planning, becomes challenging, as the orientation of the robot has an impact on its footprint. To avoid collisions and to follow the generated path, the DWB controller, a critic-based local planner with a dynamic window approach, is used. With basic tuning of DWB parameters and critics, the local and global planner combination was found to provide sufficient agility while preventing collisions.

5.6 Results

The path taken by the robot to find a remote controller overlaid on the architectural plan of the apartment, is given in Fig. 6, and the path for finding a trashcan is provided in Fig. 7.

In finding the TV remote, the system appears to recognize the living room area, prioritizing exploration of the right-hand side of the apartment. Upon getting closer to the television, the remote controller was successfully recognized, and the mission was over.

For the setup to find a trash can, the robot initially attempts to navigate towards the bedroom on the left. At the time of exploration, the algorithm confused the featureless environment to be a bathroom and focused on its exploration. Eventually, the robot sees the bed and proceeds to focus on another frontier point. The robot observed the trashcan located in the kitchen initially,



Fig. 6. Robot path in searching for a TV remote, overlaid on the architectural plan of the apartment.

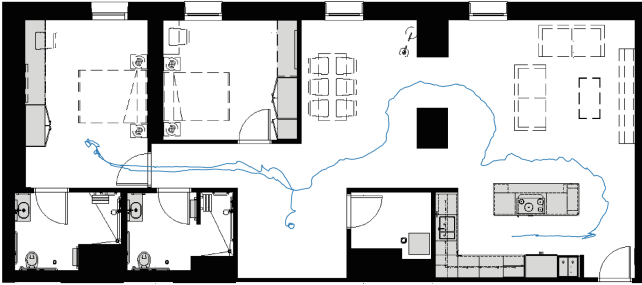


Fig. 7. Robot path in searching for a trashcan, overlaid on the architectural plan of the apartment.

but the path planning algorithm proposed a more indirect approaching angle for the trashcan, in an attempt to minimize the cost of traversal. Eventually, the robot reached the trashcan, completing the task.

6 Conclusion

Our work presents the ‘Doubly Right’ framework, a step forward in Zero-Shot Object Goal Navigation (ZS-OGN), enabling reliable semantic understanding in robotics. Our approach stands out as the first to implement ZS-OGN in real-world settings, demonstrating strong potential through simulation and practical application. Real-world tests demonstrated that lack of discerning features in the environment can result in the system making a poor initial choice in hindsight, due to the lack of knowledge about the observed environment, but the system is nevertheless able to correct course and complete the task. This breakthrough lays the groundwork for future autonomous systems to navigate novel environments without prior training, offering a glimpse into the next frontier of robotic adaptability and intelligence.

References

1. Al-Halah, Z., Ramakrishnan, S.K., Grauman, K.: Zero experience required: plug & play modular transfer learning for semantic visual navigation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 17031–17041 (2022)
2. Chaplot, D.S., Gandhi, D., Gupta, S., Gupta, A., Salakhutdinov, R.: Learning to explore using active neural SLAM. arXiv preprint [arXiv:2004.05155](https://arxiv.org/abs/2004.05155) (2020)
3. Chaplot, D.S., Gandhi, D.P., Gupta, A., Salakhutdinov, R.R.: Object goal navigation using goal-oriented semantic exploration. *Adv. Neural. Inf. Process. Syst.* **33**, 4247–4258 (2020)
4. Chen, P., et al.: Weakly-supervised multi-granularity map learning for vision-and-language navigation. *Adv. Neural. Inf. Process. Syst.* **35**, 38149–38161 (2022)
5. Deitke, M., et al.: RoboTHOR: an open simulation-to-real embodied AI platform. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3164–3174 (2020)
6. Dorbala, V.S., Mullen Jr, J.F., Manocha, D.: Can an embodied agent find your “Cat-shaped Mug”? LLM-guided exploration for zero-shot object navigation. arXiv preprint [arXiv:2303.03480](https://arxiv.org/abs/2303.03480) (2023)
7. Gadre, S.Y., Wortsman, M., Ilharco, G., Schmidt, L., Song, S.: CLIP on wheels: zero-shot object navigation as object localization and exploration, **3**(4), 7 (2022). arXiv preprint [arXiv:2203.10421](https://arxiv.org/abs/2203.10421)
8. Gadre, S.Y., Wortsman, M., Ilharco, G., Schmidt, L., Song, S.: CoWs on pasture: baselines and benchmarks for language-driven zero-shot object navigation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 23171–23181 (2023)
9. Huang, C., Mees, O., Zeng, A., Burgard, W.: Visual language maps for robot navigation. In: 2023 IEEE International Conference on Robotics and Automation (ICRA), pp. 10608–10615. IEEE (2023)
10. Kahn, G., Villaflor, A., Ding, B., Abbeel, P., Levine, S.: Self-supervised deep reinforcement learning with generalized computation graphs for robot navigation. In: 2018 IEEE International Conference on Robotics and Automation (ICRA), pp. 5129–5136. IEEE (2018)
11. Karnan, H., Warnell, G., Xiao, X., Stone, P.: VOILA: visual-observation-only imitation learning for autonomous navigation. In: 2022 International Conference on Robotics and Automation (ICRA), pp. 2497–2503. IEEE (2022)
12. Khandelwal, A., Weihs, L., Mottaghi, R., Kembhavi, A.: Simple but effective: CLIP embeddings for embodied AI. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 14829–14838 (2022)
13. Komorowski, J.: MinkLoc3D: point cloud based large-scale place recognition. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 1790–1799 (2021)
14. Labbé, M., Michaud, F.: RTAB-map as an open-source lidar and visual SLAM library for large-scale and long-term online operation. *J. Field Rob.* **36**(2), 416–446 (2019)
15. Li*, L.H., et al.: Grounded language-image pre-training. In: CVPR (2022)
16. Macenski, S., Martin, F., White, R., Ginés Clavero, J.: The marathon 2: a navigation system. In: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (2020)

17. Majumdar, A., Aggarwal, G., Devnani, B., Hoffman, J., Batra, D.: ZSON: zero-shot object-goal navigation using multimodal goal embeddings. *Adv. Neural. Inf. Process. Syst.* **35**, 32340–32352 (2022)
18. Ramakrishnan, S.K., Chaplot, D.S., Al-Halah, Z., Malik, J., Grauman, K.: PONI: potential functions for objectgoal navigation with interaction-free learning. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18890–18900 (2022)
19. Shah, D., Equi, M.R., Osiński, B., Xia, F., Ichter, B., Levine, S.: Navigation with large language models: semantic guesswork as a heuristic for planning. In: *Conference on Robot Learning*, pp. 2683–2699. PMLR (2023)
20. Silver, D., Bagnell, J., Stentz, A.: High performance outdoor navigation from overhead data using imitation learning. In: *Robotics: Science and Systems IV*, Zurich, Switzerland, vol. 1 (2008)
21. Unlu, H.U., Chaikalis, D., Tsoukalas, A., Tzes, A.: UAV indoor exploration for fire-target detection and extinguishing. *J. Intell. Rob. Syst.* **108**(3), 54 (2023)
22. Wöhlke, J., Schmitt, F., van Hoof, H.: Hierarchies of planning and reinforcement learning for robot navigation. In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 10682–10688. IEEE (2021)
23. Xia, Y., Shi, L., Ding, Z., Henriques, J.F., Cremers, D.: Text2Loc: 3D point cloud localization from natural language. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14958–14967 (2024)
24. Yamauchi, B.: A frontier-based approach for autonomous exploration. In: *Proceedings 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation CIRA'97. Towards New Computational Principles for Robotics and Automation*, pp. 146–151. IEEE (1997)
25. Yao, S., et al.: ReAct: synergizing reasoning and acting in language models. *arXiv preprint [arXiv:2210.03629](https://arxiv.org/abs/2210.03629)* (2022)
26. Zhang, T.Y., Suen, C.Y.: A fast parallel algorithm for thinning digital patterns. *Commun. ACM* **27**(3), 236–239 (1984)
27. Zhao, Q., Zhang, L., He, B., Liu, Z.: Semantic policy network for zero-shot object goal visual navigation. *IEEE Rob. Autom. Lett.* (2023)
28. Zhao, Q., Zhang, L., He, B., Qiao, H., Liu, Z.: Zero-shot object goal visual navigation. In: *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2025–2031. IEEE (2023)
29. Zheng, K., et al.: JARVIS: a neuro-symbolic commonsense reasoning framework for conversational embodied agents. *arXiv preprint [arXiv:2208.13266](https://arxiv.org/abs/2208.13266)* (2022)
30. Zhou, G., Hong, Y., Wu, Q.: NavGPT: explicit reasoning in vision-and-language navigation with large language models. *arXiv preprint [arXiv:2305.16986](https://arxiv.org/abs/2305.16986)* (2023)
31. Zhou, K., et al.: ESC: exploration with soft commonsense constraints for zero-shot object navigation. In: *International Conference on Machine Learning*, pp. 42829–42842. PMLR (2023)



AllWeather-Net: Unified Image Enhancement for Autonomous Driving Under Adverse Weather and Low-Light Conditions

Chenghao Qian¹(✉) , Mahdi Rezaei¹ , Saeed Anwar² , Wenjing Li¹ ,
Tanveer Hussain³ , Mohsen Azarmi¹ , and Wei Wang⁴ 

¹ University of Leeds, Leeds, UK

² Australian National University, Canberra, Australia

³ Edge Hill University, Ormskirk, UK

⁴ Shenzhen Campus of Sun Yat-sen University, ShenZhen, China

Abstract. Adverse conditions like snow, rain, nighttime, and fog, pose challenges for autonomous driving perception systems. Existing methods have limited effectiveness in improving essential computer vision tasks, such as semantic segmentation, and often focus on only one specific condition, such as removing rain or translating nighttime images into daytime ones. To address these limitations, we propose a method to improve the visual quality and clarity degraded by such adverse conditions. Our method, AllWeather-Net, utilizes a novel hierarchical architecture to enhance images across all adverse conditions. This architecture incorporates information at three semantic levels: scene, object, and texture, by discriminating patches at each level. Furthermore, we introduce a Scaled Illumination-aware Attention Mechanism (SIAM) that guides the learning towards road elements critical for autonomous driving perception. SIAM exhibits robustness, remaining unaffected by changes in weather conditions or environmental scenes. AllWeather-Net effectively transforms images into normal weather and daytime scenes, demonstrating superior image enhancement results and subsequently enhancing the performance of semantic segmentation, with up to a 5.3% improvement in mIoU in the trained domain. We also show our model's generalization ability by applying it to unseen domains without re-training, achieving up to 3.9 % mIoU improvement. Code can be accessed at: <https://github.com/Jumpthemoon/AllWeatherNet>.

Keywords: Image enhancement · Semantic segmentation · Hierarchical discrimination · Illumination-aware attention

1 Introduction

Autonomous driving systems heavily rely on clear and optimal environmental images; however, these are not guaranteed in real life due to natural conditions,

like snow, rain, fog, low light at night, etc. This can significantly reduce visibility and distort the captured information within an image, which impacts the performance of autonomous driving perception systems, including but not limited to object detection and semantic segmentation (Fig. 1).

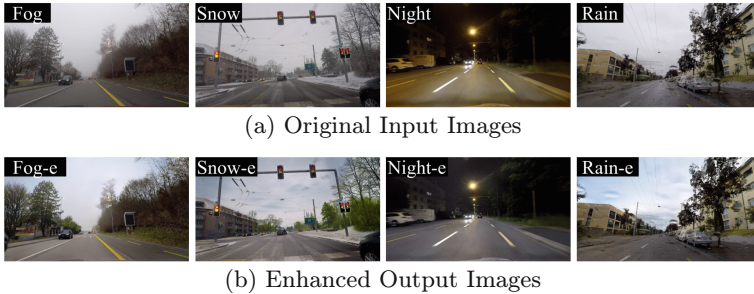


Fig. 1. Given images captured under adverse conditions in (a), we propose a method that can effectively adjust color and texture, modify lighting and shadows, and remove weather effects within a unified model. This results in a visually appealing appearance that resembles normal, day-like weather conditions (b), thereby enhancing the robust performance of autonomous driving perception systems.

To counter the mentioned problem, some methods remove weather artifacts via deraining [22, 24], dehazing [3, 25], and desnowing [15, 21, 27]. Moreover, some unified frameworks [4, 12, 14] handle three types of weather while mainly focusing on removing hydrometer particles, neglecting alterations in color and texture details; hence, restricting their effectiveness under adverse weather conditions for autonomous driving computer vision systems.

In contrast to weather artifacts removal, pixel-level image translation approaches transform challenging weather situations into clear, sunny-day image styles. Regardless, these methodologies mainly focus only on specific individual conditions, such as rain [13] or nighttime scenarios [2]. In addition, the model may alter irrelevant pixels or areas and introduce unwanted changes, leading to visual discrepancies and negatively impacting the performance of downstream tasks. Likewise, low-light enhancement aims to improve the visibility and quality of images captured in low-light conditions. This involves enhancing the brightness, contrast, and details of dark images due to insufficient lighting; however, this technique can mistakenly brighten already well-lit areas, leading to overexposure in weather conditions like snow, as shown in Fig. 2.

We aim to improve image quality and clarity by adjusting image attributes and enhancing texture under four distinct adverse conditions, all within a unified framework. Subsequently, we seek to improve semantic segmentation performance. To achieve this goal, we need to consider several critical factors:

Firstly, while a unified network is cost-effective, weather variability introduces instability in the learning process. Therefore, it is crucial to identify a

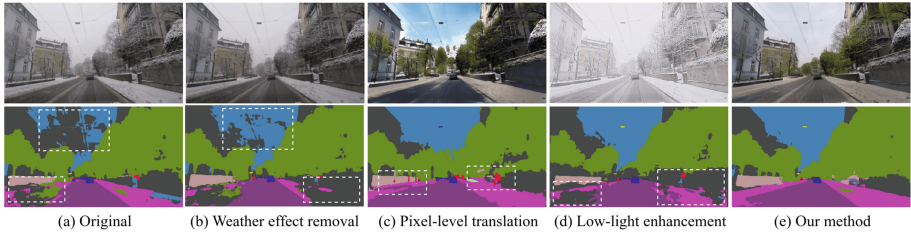


Fig. 2. (a) Original Image. The evaluation of image processing techniques for semantic segmentation under adverse conditions reveals the deficiencies of (b) weather effect removal [4], (c) pixel-level translation [29], and (d) low-light enhancement [16]. Images processed by these methods either fail to sufficiently enhance image quality or introduce artifacts, affecting semantic prediction accuracy. (e) Our method, AllWeather-Net, effectively enhances color and texture detail while preserving most of the original image information, achieving the best performance.

stable and invariant signal that can guide the network’s learning, ensuring consistent performance across all conditions. *Secondly*, unfavorable conditions differently impact various regions within a captured image. For example, in foggy scenes, distant objects are more blurred than nearby ones due to light scattering and attenuation. In addition, adverse weather conditions tend to preserve larger patterns in images while diminishing the clarity of finer details. So, it is essential to focus on both the overall enhancement and the intricate recovery of texture details. This motivates us to design a network architecture that is contextually aware and sensitive to variations in texture. *Lastly*, employing a pair-to-pair training strategy can improve performance, yet finding perfectly matched pairs in autonomous driving scenes is challenging due to inaccurate GPS pairing and environmental variations. Alternatively, we consider adopting a strategy that utilizes roughly aligned images for more robust discrimination during training when exactly matched pairs are unavailable.

To address these challenges, we propose a novel architecture, namely AllWeather-Net, and our contributions can be summarized as follows:

- We are the first to introduce a unified image enhancement method to address image quality degradation under adverse weather and low-light conditions, including snow, rain, fog, and nighttime.
- To achieve robust image enhancement across various adverse conditions, we introduce a Scaled Illumination-aware Attention Mechanism (SIAM) that directs a balanced learning process towards different road elements irrespective of changes in weather and scenes.
- To achieve both overall image consistency and detailed enhancement, we design a novel architecture that enhances input images by conducting discrimination tasks at three hierarchical levels of semantic patches: scene, objects, and texture.

2 Related Work

In this section, we review the image processing techniques for adverse weather conditions and low-light environments.

Weather Effect Removal. Current methods for removing visual artifacts, including raindrops, fog particles, and snowflakes, utilize processes such as deraining [22, 24], dehazing [3, 25, 26] and desnowing [15, 21, 27]. Recently, a unified bad weather removal network was proposed in [14]. In [12], the researcher simplifies this architecture with a single encoder-single decoder network. To reduce the computational cost, [4] proposed a knowledge transfer mechanism via teacher-student architecture.

Pixel-Level Translation transforms the visual representation and convert adverse weather conditions into scenes resembling sunny, daytime environments. It involves a direct modification of the image pixels, altering the fundamental appearance and context of the scene. CycleGAN [29] introduced a cycle-consistency loss for unsupervised translation between the source and target domains. CUT [18] uses contrastive learning to ensure content preservation and style transfer. Santa [23] proposes an approach to find the shortest path between source and target images without paired information.

Low-Light Enhancement aims to adjust attributes of an image, such as lighting and color balance, to enhance the visual appearance in low-light conditions. Traditional methods utilize histogram equalization [1] and Retinex [10] to perform low-light image enhancement. Recent deep learning approaches proposed end-to-end framework [7, 8, 16]. Compared to traditional methods, these frameworks demonstrate the capability of enhancing the quality of images captured in low-light conditions.

Limitation of Existing Works. Removing weather-related unfavourable effects typically targets minor disturbances such as snowflakes or raindrops in the image. However, merely eliminating these atmospheric particles is insufficient, as the primary cause of image quality degradation often stems from alterations in colors and texture details, which significantly contribute to domain shifts. This limitation also applies to pixel-level translation, which frequently introduces unwanted artifacts, thereby reducing the overall image quality and constraining their applicability in safety-critical scenarios. Similarly, low-light enhancement techniques, while focusing on improving visibility under low-light conditions, do not adequately address the challenges posed by adverse weather conditions.

3 Proposed Method

Our proposed method uses a generative model for generating image enhancement masks based on the original input image. We introduce a scaled illumination-aware attention mechanism (SIAM) within a unified framework to focus learning

on road elements regardless of weather condition. Additionally, our hierarchical framework performs discrimination at multiple semantic levels, ensuring consistent and detailed enhancement. To further ensure precise enhancement, we utilize a ranked adaptive window pairing strategy for accurate discrimination.

3.1 Enhancement Pipeline

Our image enhancement method involves two networks: a generator and a discriminator, which are trained simultaneously through adversarial training to enhance image quality. Unlike pixel-level translation (Fig. 3a), where the generator takes the source image I_S and directly outputs translation results to mimic the style of the target image, our image enhancement generates intermediate results that are then combined with the original image I_S to produce final enhancement results. As illustrated in Fig. 3b, the process of generating enhanced image I' can be formulated as:

$$I' = G(I_S) + I_S. \quad (1)$$

Pixel-level translation often suffers from generating unwanted artifacts, which can be attributed to the large search space during the training process. By conditioning the output on the input image, the image enhancement model can effectively reduce the search space for the generated result through residual learning. This method ensures that the model’s outputs are contextually relevant to the original image, thereby significantly reducing the likelihood of producing unwanted artifacts and improving the quality of the generated images.

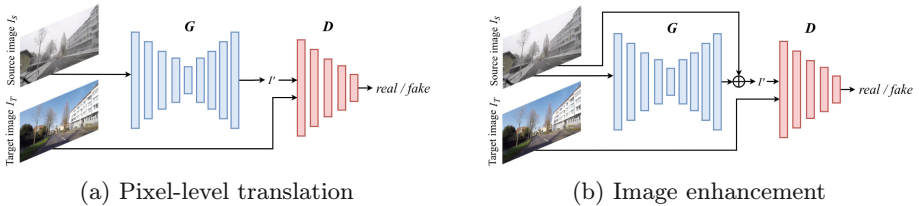


Fig. 3. Comparison of pixel-level translation and image enhancement process.

In our model (Fig. 4), we initially cropped the same area from the paired source images I_S and target image I_T as input. The cropped source scene patch P_S^s is processed through a scaled illumination-aware attention mechanism and the generator to produce an enhancement mask M^s . This mask is then added to generate the final enhanced results P_f^s and evaluated by different discriminators according to its scale.

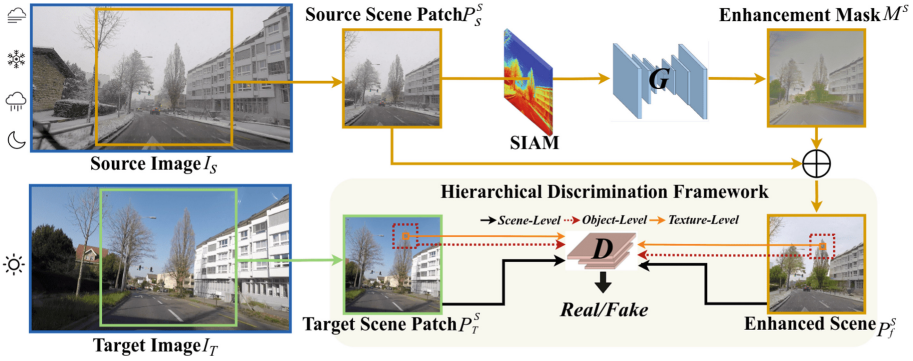


Fig. 4. Overview of AllWeather-Net architecture. SIAM: Scaled Illumination-aware Attention Mechanism. AllWeather-Net can enhance images across all adverse conditions (e.g., fog, snow, rain, nighttime) with the help of the proposed SIAM and Hierarchical Discrimination Framework.

3.2 Scaled Illumination-Aware Attention Mechanism (SIAM)

Training a unified network for image enhancement across different adverse conditions is cost-effective yet challenging. Each condition uniquely alters the scene’s visibility, color, and texture, complicating the learning of consistent information across different scenarios. This variability can significantly degrade the model’s capacity to effectively enhance images, especially when adverse conditions heavily obscure scene details. Given these complexities, the importance of a condition-invariant signal in guiding the learning process is paramount. Such a signal should guide the model to learn critical aspects of the scene regardless of weather or lighting conditions.

Drawing inspiration from previous work [9] targeting low-light conditions, we consider using illumination as a guiding cue. However, the naive approach of employing illumination intensity as attention tends to overemphasize areas of low illumination while neglecting well-lit regions. This can result in inadequate focus on pixel regions obscured by snow or fog, which often appear brighter due to higher illumination levels. This discrepancy can lead to suboptimal learning outcomes, as crucial details in these areas may not receive sufficient attention, resulting in inconsistent enhancements in the generated images.

To direct a balanced learning for different road elements, rather than merely focusing on dark regions, we propose the scaled illumination-aware attention mechanism (SIAM) to allocate reasonable attention based on illumination intensity. Let I_{ij} and Att_{ij} denote the illumination intensity and the naive illumination attention value at the given pixel location (i, j) , the scaled illumination attention S_Att_{ij} can be formulated as follows:

$$Att_{ij} = 1 - I_{ij}, \tag{2}$$

$$S_Att_{ij} = -Att_{ij} \cdot (Att_{ij} - 2). \tag{3}$$

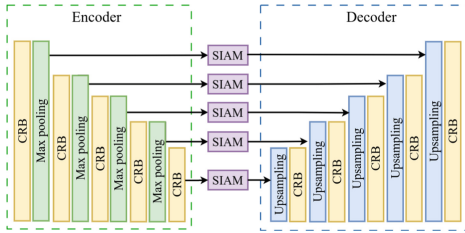


Fig. 5. Scaled illumination-aware attention mechanism in the generator.

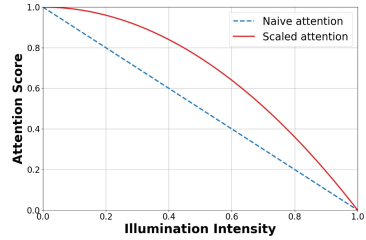


Fig. 6. Attention scores by illumination: naive vs. scaled attention.

In the network, the SIAM will guide the learning throughout the generator shown in Fig. 5. As shown in Fig. 6, the scaled illumination-aware attention exhibits higher scores for low illumination and maintains consistently high attention levels across the input range for low illumination. This design demonstrates heightened sensitivity towards regions with low illumination while ensuring that high-illumination areas are allocated reasonable focus. With the implementation of the scaled attention, our model prioritizes objects in the distance obscured by fog particles with high illumination (Fig. 7).

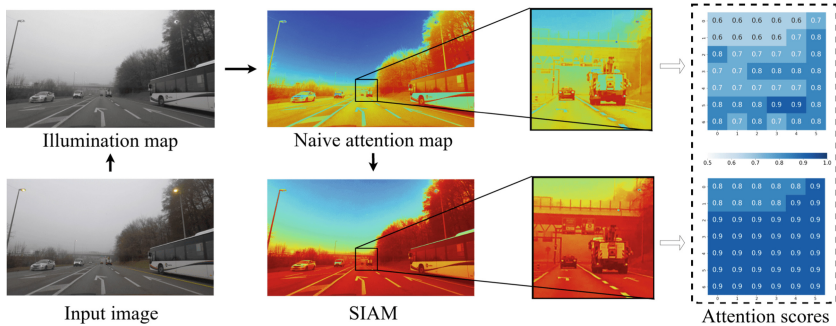


Fig. 7. The flow of generating attention map with SIAM and the comparison between naive attention and SIAM at the image and patch levels. Note that a higher attention score indicates that the model is paying more attention to such an area. This observation suggests that the presented SIAM, compared to naive attention mechanisms, is more adept at focusing on areas containing road elements.

3.3 Hierarchical Discrimination Framework

Hierarchical Discrimination. From the perspective of discrimination, employing a single discriminator generates unrealistic colors, while the global-local structure [9] has limited performance in providing fine-grained textural details.

To address this issue, we propose our hierarchical discrimination architecture with scene-, object-, and texture-level patches/discriminators.

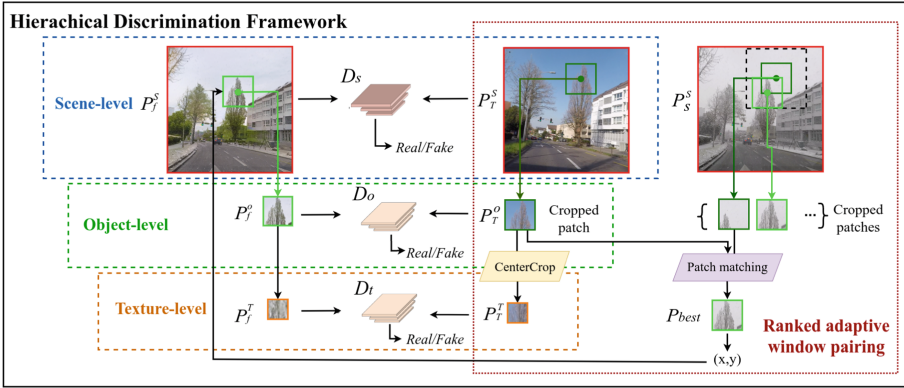


Fig. 8. The details of hierarchical discrimination framework.

The scene-level discriminator D_s assesses a randomly cropped scene patch P^s from the original image I_S , as shown in Fig. 8. Its primary purpose is to evaluate the overall coherence and realism of the generated scene. Next, we crop object patches P^s with 1/4 of the size of the scene-level patch for the object-level discriminator D_o . This selective cropping allows for the exploration of intricate image details that the scene-level discriminator might overlook due to its broader perspective. Lastly, we derive the texture-level patch P^t by 1/4 center cropping from each object-level patch P^o . The texture-level discriminator D_t represents the highest level of inspection in our discriminator hierarchy, focusing on examining fine details and texture quality of the generated image.

By incorporating the mentioned three different levels of patches and discriminators, our model differentiates between the broad scales and fine levels of detail. This enables generated images that are not only realistic in overall appearance but also exhibit improved textural fidelity.

Ranked Adaptive Window Pairing. Scene-level discrimination is accurate due to similar information in paired images. However, accuracy drops at the object level, where patch pairs often show significant information shifts due to more apparent changes in perspective, leading to suboptimal outcomes.

To address the above mentioned issue, we utilize an adaptive window with a ranked score to identify the object-level patches that are most closely aligned. We begin by cropping object-level patches P^o at the same location from scene-level patches P^s . Moreover, we then define a fixed search area A with a width of w and a height of h . Within this area, we deploy a dynamic window W , of size $z \times z$, to traverse the defined area with a stride of s , thereby generating object-level patch candidates. These candidate patches are subsequently compared to

the corresponding target patch using a ranked pairing score to determine the best match of P_S^o and P_T^o . Finally, we center crop the top-matched object pairs to obtain patches P_S^t and P_T^t for texture-level discrimination. Let (x, y) represent the top-left coordinate of the matched source object patch. $N = \frac{x+w-z}{s}$ represents the horizontal steps required, while $M = \frac{x+h-z}{s}$ denotes the vertical steps needed to traverse the search area with the dynamic window. The formulation of the pairing score F is defined as:

$$F(P_{S_c}^o, P_T^o) = \sum_{i=0}^N \sum_{j=0}^M |I_{P_T^o}(x, y) - I_{P_{S_c}^o}(x + i \cdot s, y + j \cdot s)|, \quad (4)$$

where $P_{S_c}^o$ denotes the c -th candidate patch within the search window area and I represents the RGB values of the patch. The variables i and j signify the horizontal and vertical offsets within the search window, respectively. The best-matched patch, denoted as P_{best} , is then determined:

$$P_{\text{best}} = \arg \min_{P \in W} S(P_{S_c}^o, P_T^o). \quad (5)$$

By identifying the location of P_{best} in the source scene, we can locate its counterpart in the corresponding generated scene patch.

3.4 Loss Function

We utilize a relativistic approach [11] that compares the realism between real and generated images. We employ LSGAN [17] loss for direct assessment of the realism of the object- and texture-level discrimination. The scene-level losses for the discriminator and generator are given below:

$$L_D^s = \mathbb{E}_{P_r^s \sim P_{\text{real}}} [(D_s(P_r^s, P_f^s) - 1)^2] + \mathbb{E}_{P_f^s \sim P_{\text{fake}}} [D_R(P_f^s, P_r^s)^2], \quad (6)$$

$$L_G^s = \mathbb{E}_{P_f^s \sim P_{\text{fake}}} [(D_s(P_f^s, P_r^s) - 1)^2] + \mathbb{E}_{P_r^s \sim P_{\text{real}}} [D_R(P_r^s, P_f^s)^2], \quad (7)$$

where D_s represents the relativistic discriminator, P_r^s and P_f^s denote the real and generated fake scene patch. $\mathbb{E}_{P_r^s \sim P_{\text{real}}}$ and $\mathbb{E}_{P_f^s \sim P_{\text{fake}}}$ represent expectations over the real and fake data distributions. For object- and texture-level loss, The discriminator and generator for losses $P^x \in \{P^o, P^t\}$ are given by:

$$L_D^x = \mathbb{E}_{P_r^x \sim P_{\text{real}}} [(D_x(P_r^x) - 1)^2] + \mathbb{E}_{P_f^x \sim P_{\text{fake}}} [(D_x(P_f^x) - 0)^2], \quad (8)$$

$$L_G^x = \mathbb{E}_{P_f^x \sim P_{\text{fake}}} [(D_x(P_f^x) - 1)^2]. \quad (9)$$

Here, P_r^x and P_f^x denote real and fake patches of type x , respectively. D_x represents the discriminator for the patch type x , and $\mathbb{E}_{P_f^x \sim P_{\text{fake}}}$ and $\mathbb{E}_{P_r^x \sim P_{\text{real}}}$ are the same meaning as for scene patch. Consider the set $\mathcal{L} = \{s, o, t\}$ corresponding to the types of losses and use λ_1 , λ_2 , and λ_3 to control each loss contribution to balance loss; the total training loss can be written as:

$$\text{Total Loss} = \sum_{\ell \in \mathcal{L}} \lambda_\ell \cdot (L_G^\ell + L_D^\ell),$$

where $A_s = \lambda_1$, $A_o = \lambda_2$, and $A_t = \lambda_3$.

4 Experiments

We conduct our experiments on image enhancement and evaluate the outcomes from two perspectives: image quality and semantic segmentation. Both aspects are assessed qualitatively and quantitatively.

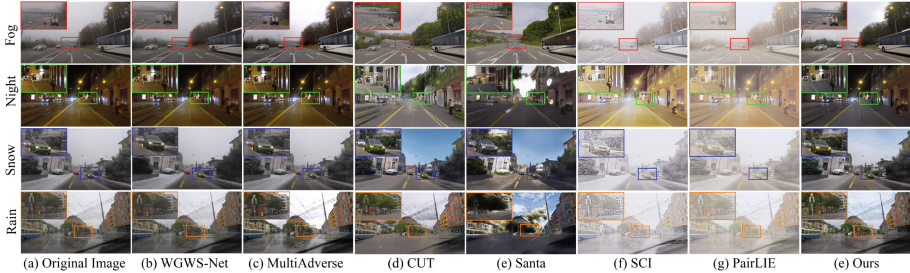


Fig. 9. Comparison with other image processing methods regarding weather effect removal, pixel-level translation, and low-light enhancement, using zoomed-in red regions to highlight visual distinctions. (Color figure online)

4.1 Dataset

For image enhancement model training, we use 1,600 images from the ACDC [20], evenly distributed among snow, rain, night, and fog conditions, and 2,416 nighttime images from the Dark Zurich [5]. For the evaluation of semantic segmentation, our model enhances images from both the ACDC and the Dark Zurich validation set, which are subsequently tested using a pre-trained PSPNet [28] model. To demonstrate the generalization capabilities of our model, we apply it to the test datasets of Foggy Zurich [19] and Nighttime driving [6].

4.2 Comparisons

We evaluate AllWeather-Net against three distinct approaches: (a) weather effect removal, (b) pixel-level translation, and (c) low-light enhancement. We employ the Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM) to assess enhancement quality. We utilize the Natural Image Quality Evaluator (NIQE) to evaluate the image naturalness. Additionally, we consider the similarity between the improved and reference daytime images employing the Learned Perceptual Image Patch Similarity (LPIPS). We adopt the Mean Intersection over Union (mIoU) metric to evaluate semantic segmentation performance.

Image Quality. In Fig. 9, the first column represents the input images captured under different adverse conditions while the subsequent columns are the enhanced results by different models. We observe that weather effect removal methods such as WGWS-Net [30] and MultiAdverse [4] excel in eliminating weather-related artifacts but fall short in achieving a sunny daytime appearance. Pixel-level translation methods, including CUT [18] and Santa [23] can effectively transform images from adverse conditions to daytime-like settings, yet these models struggle with visual consistency in nighttime scenes and may inadvertently remove less prominent objects, such as a car in the distance, in foggy conditions. Low-light enhancement-based method SCI [16] and PairLIE [7] can enhance lighting and brightness in night scenes but over-expose in well-lit conditions, *e.g.* fog and snow. In contrast, our method demonstrates the most significant improvements in brightness and contrast across various scenes. It delivers the most realistic and clear daytime representation under diverse adverse conditions, significantly enhancing visibility without introducing artifacts or excessive noise. Additionally, it outperforms other models in color correction and detail enhancement, offering a more comprehensive solution.

As shown in Table 1 and Table 2, our method achieves the most natural image enhancement outcomes with the lowest NIQE score and excels in converting images to the daytime domain, indicated by the lowest LPIPS scores for nighttime scenes.

Table 1. Qualitative evaluation of image quality and semantic segmentation performance on ACDC dataset.

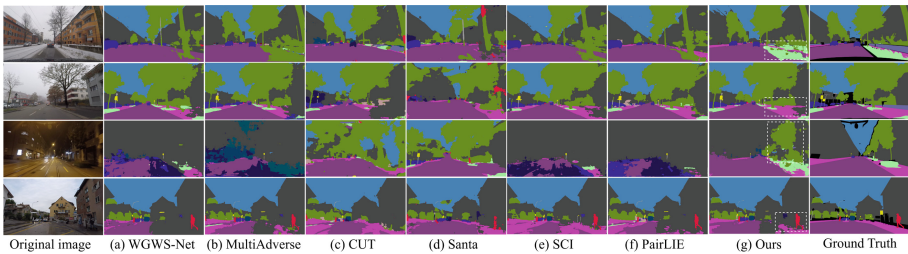
Method Type	Models	Metrics				
		SSIM \uparrow	PSNR \uparrow	NIQE \downarrow	LPIPS \downarrow	mIoU \uparrow
Weather effect removal	WGWS-Net [30]	0.3916	11.4812	0.1480	0.4740	36.4
	MultiAdverse [4]	0.3822	10.9135	0.1782	0.4752	37.0
Pixel-level translation	CycleGAN [29]	0.3981	12.2493	0.1578	0.4655	33.6
	CUT [18]	0.3776	12.1043	0.1668	0.4833	29.3
	Santa [23]	0.3920	12.0863	0.1374	0.4770	25.7
Low-light enhancement	EnlightenGAN [9]	0.3905	11.8725	0.1651	0.4649	37.2
	Zero-DCE [8]	0.3160	10.6015	0.2726	0.4428	32.4
	SCI [16]	0.3239	8.5693	0.2600	0.5149	24.4
	PairLIE [7]	0.3577	8.9133	0.1564	0.4598	28.3
All weather enhancement	Ours	0.3983	11.6618	0.1257	0.4619	38.2

Semantic Segmentation. The effectiveness of our image enhancement model for semantic segmentation is assessed by performing a direct evaluation using the pre-trained PSPNet model [28]. We apply the model to datasets enhanced by our model as well as those enhanced by others. As shown in last column

Table 2. Qualitative evaluation of image quality and semantic segmentation performance on Dark Zurich dataset.

Models	Metrics				
	SSIM \uparrow	PSNR \uparrow	NIQE \downarrow	LPIPS \downarrow	mIoU \uparrow
EnlightenGAN [9]	0.3791	10.2460	0.3186	0.4766	10.8
Zero-DCE [8]	0.3324	8.6448	0.3425	0.5109	10.6
SCI [16]	0.3519	8.5735	0.2554	0.5115	11.0
PairLIE [7]	0.3827	9.1442	0.2083	0.4844	7.1
Ours	0.3849	9.6121	0.1850	0.4589	17.6

in Table 1 and Table 2, our method demonstrates superior performance in both adverse weather and nighttime scenes. This indicates that our model can enhance the performance of semantic segmentation models by significantly improving visual quality and visibility compared to other image processing models. In the visualization results shown in Fig. 10, our method improves the detail recognition of road elements such as trees, grass, and pedestrians in all conditions.

**Fig. 10.** Semantic segmentation results in comparison with other state-of-the-art methods of weather effect removal, pixel-level translation, and low-light enhancement, using zoomed-in white regions to highlight visual distinctions.

Generalization to Unseen Datasets. We evaluate our trained model’s performance in scenarios not seen during training using the Foggy Zurich and Nighttime Driving datasets. The results in Fig. 11 show that our model enhances clarity of cars, traffic lights, and road signs in the Foggy Zurich dataset, and corrects the yellowish glow on buildings and trees in the Nighttime Driving dataset, restoring their true colors and visibility. The mIoU comparison (Table 3) shows improvements of 1.8% and 3.9% respectively, highlighting the remarkable generalization capability of our model and demonstrating its ability to enhance semantic segmentation performance in unseen domains without re-training.



Fig. 11. The generalization performance of our model on the Foggy Zurich and Nighttime Driving datasets. The red and green box corresponds to the zoomed-in patches. (Color figure online)

Table 3. MIOU scores tested on pretrained PSPNet [28] for original and enhanced versions of Foggy Zurich and Nighttime Driving test datasets.

Datasets	Original	Enhanced
Foggy Zurich [19]	26.3	28.1
Nighttime Driving [6]	23.0	26.9

4.3 Ablation Studies

To demonstrate the impact of each component in our method, we conduct ablation experiments with a focus on image quality improvement. These experiments examine different levels of discrimination, Ranked Adaptive Window Pairing (RAWP), and the Scaled Illumination Attention mechanism (SIAM).

Table 4. Qualitative comparison of model components using SSIM. A higher SSIM value indicates better image generation quality as it signifies greater similarity to a clear daytime image in terms of structure, luminance, and contrast.

Models	Components					SSIM \uparrow
	D_s	D_o	D_t	RAWP	SIAM	
M_1	✓					0.3864
M_2	✓	✓				0.3879
M_3	✓	✓	✓			0.3910
M_4	✓	✓	✓	✓		0.3922
M_5	✓	✓	✓	✓	✓	0.3983

In Table 4, we observe enhanced image quality, as indicated by higher SSIM values, with the incremental inclusion of discriminators: scene discriminator D_s , object discriminator D_o , and texture discriminator D_t . The introduction of RAWP further increases SSIM, indicating that the model learns finer details in local patches through pair-to-pair training. Subsequently, incorporating attention further elevates SSIM, demonstrating the effectiveness of attention in

enhancing image quality. For various adverse conditions, the scaled illumination attention mechanism effectively focuses on road elements, as demonstrated in Fig. 12. The enhanced image quality in Fig. 13 indicates that scaled attention addresses uneven lighting issues by allocating attention appropriately, particularly to areas overlaid with high illumination. This highlights the capability of the scaled attention mechanism to focus on both low and high-illumination regions within road elements and adapt to all adverse conditions.

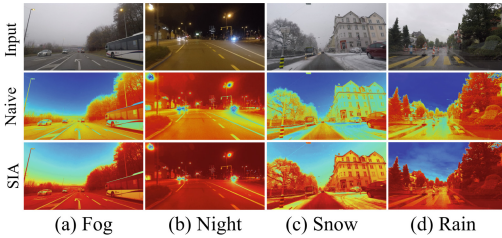


Fig. 12. Naive attention and SIAM maps for various input adverse condition images. Darker regions indicate higher attention scores.

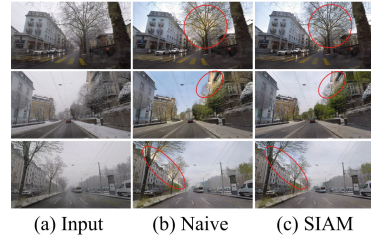


Fig. 13. Results generated by models trained with naive attention and scaled attention.

5 Conclusions

In this work, we introduced AllWeatherNet, a unified framework designed to enhance image quality under various adverse conditions such as snow, rain, fog, and nighttime. Our objective was to develop a singular model capable of simultaneously addressing these four conditions without introducing artifacts that degrade image quality. The model can adjust lighting, brightness, and color in images in both adverse and normal weather conditions, transforming them into clear, daytime-like visuals. We implemented a hierarchical framework to recover color and texture details, along with a ranked adaptive window pair-to-pair training strategy to boost performance. We also developed a scaled-illumination attention mechanism to direct the learning process towards low and high-illumination areas, making it adaptable to different adverse scenarios. We performed semantic segmentation experiments using our enhanced dataset and observed notable improvements. Additionally, the model demonstrated exceptional generalization capability across a range of datasets without requiring re-training.

References

1. Abdullah-Al-Wadud, M., Kabir, M.H., Dewan, M.A.A., Chae, O.: A dynamic histogram equalization for image contrast enhancement. *IEEE Trans. Consum. Electron.* **53**(2), 593–600 (2007)

2. Anoosheh, A., Sattler, T., Timofte, R., Pollefeys, M., Van Gool, L.: Night-to-day image translation for retrieval-based localization, March 2019. <http://arxiv.org/abs/1809.09767>, arXiv:1809.09767 [cs]
3. Berman, D., Avidan, S., et al.: Non-local image dehazing. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1674–1682 (2016)
4. Chen, W.T., Huang, Z.K., Tsai, C.C., Yang, H.H., Ding, J.J., Kuo, S.Y.: Learning multiple adverse weather removal via two-stage knowledge learning and multi-contrastive regularization: toward a unified model. In: CVPR, pp. 17653–17662 (2022)
5. Dai, D., Sakaridis, C., Hecker, S., Van Gool, L.: Curriculum model adaptation with synthetic and real data for semantic foggy scene understanding, May 2019. arXiv:1901.01415 [cs]
6. Dai, D., Van Gool, L.: Dark model adaptation: semantic image segmentation from daytime to nighttime. In: International Conference on Intelligent Transportation Systems (ITSC), pp. 3819–3824. IEEE (2018)
7. Fu, Z., Yang, Y., Tu, X., Huang, Y., Ding, X., Ma, K.K.: Learning a simple low-light image enhancer from paired low-light instances. In: CVPR, pp. 22252–22261 (2023)
8. Guo, C., et al.: Zero-reference deep curve estimation for low-light image enhancement. In: CVPR, Seattle, WA, USA, pp. 1777–1786. IEEE, June 2020
9. Jiang, Y., et al.: EnlightenGAN: deep light enhancement without paired supervision. IEEE Trans. Image Process. **30**, 2340–2349 (2021)
10. Jobson, D.J., Rahman, Z.U., Woodell, G.A.: Properties and performance of a center/surround retinex. Trans. Image Process. **6**(3), 451–462 (1997)
11. Jolicœur-Martineau, A.: The relativistic discriminator: a key element missing from standard GAN. arXiv preprint arXiv:1807.00734 (2018)
12. Jose Valanarasu, J.M., Yasarla, R., Patel, V.M.: TransWeather: transformer-based restoration of images degraded by adverse weather conditions. In: CVPR, pp. 2343–2353. IEEE, June 2022
13. Kwak, J.g., Jin, Y., Li, Y., Yoon, D., Kim, D., Ko, H.: Adverse weather image translation with asymmetric and uncertainty-aware GAN. arXiv preprint arXiv:2112.04283 (2021)
14. Li, R., Tan, R.T., Cheong, L.F.: All in one bad weather removal using architectural search. In: CVPR, pp. 3175–3185 (2020)
15. Liu, Y.F., Jaw, D.W., Huang, S.C., Hwang, J.N.: DesnowNet: context-aware deep network for snow removal. IEEE Trans. Image Process. **27**(6), 3064–3073 (2018)
16. Ma, L., Ma, T., Liu, R., Fan, X., Luo, Z.: Toward fast, flexible, and robust low-light image enhancement. In: CVPR, pp. 5637–5646 (2022)
17. Mao, X., Li, Q., Xie, H., Lau, R.Y., Wang, Z., Paul Smolley, S.: Least squares generative adversarial networks. In: ICCV, October 2017
18. Park, T., Efros, A.A., Zhang, R., Zhu, J.-Y.: Contrastive learning for unpaired image-to-image translation. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12354, pp. 319–345. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58545-7_19
19. Sakaridis, C., Dai, D., Hecker, S., Van Gool, L.: Model adaptation with synthetic and real data for semantic dense foggy scene understanding. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11217, pp. 707–724. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01261-8_42
20. Sakaridis, C., Dai, D., Van Gool, L.: ACDC: the adverse conditions dataset with correspondences for semantic driving scene understanding, September 2021. <http://arxiv.org/abs/2104.13395>, arXiv:2104.13395 [cs]

21. Wang, Y., Ma, C., Liu, J.: SmartAssign: learning a smart knowledge assignment strategy for deraining and desnowing. In: CVPR, pp. 3677–3686 (2023)
22. Wang, Z., Cun, X., Bao, J., Zhou, W., Liu, J., Li, H.: Uformer: a general U-shaped transformer for image restoration. In: CVPR, pp. 17683–17693 (2022)
23. Xie, S., Xu, Y., Gong, M., Zhang, K.: Unpaired image-to-image translation with shortest path regularization. In: CVPR, pp. 10177–10187 (2023)
24. Yang, W., Tan, R.T., Feng, J., Guo, Z., Yan, S., Liu, J.: Joint rain detection and removal from a single image with contextualized deep networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **42**(6), 1377–1393 (2019)
25. Zhang, H., Sindagi, V., Patel, V.M.: Joint transmission map estimation and dehazing using deep networks. *IEEE Trans. Circuits Syst. Video Technol.* **30**(7), 1975–1986 (2019)
26. Zhang, J., et al.: Hierarchical density-aware dehazing network. *IEEE Trans. Cybern.* **52**(10), 11187–11199 (2021)
27. Zhang, K., Li, R., Yu, Y., Luo, W., Li, C.: Deep dense multi-scale network for snow removal using semantic and depth priors. *IEEE Trans. Image Process.* **30**, 7419–7431 (2021)
28. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network, April 2017. <http://arxiv.org/abs/1612.01105>, [arXiv:1612.01105](https://arxiv.org/abs/1612.01105) [cs]
29. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: ICCV, pp. 2223–2232 (2017)
30. Zhu, Y., et al.: Represent, compare, and learn: a similarity-aware framework for class-agnostic counting. In: CVPR (2023)



Uni4DAL: A Unified Baseline for Multi-dataset 4D Auto-Labeling

Zhiyuan Yang¹, Xuekuan Wang², Wei Zhang², Xiao Tan², Jinchen Lu²,
Jingdong Wang², Errui Ding², Zhihui Lai³, and Cairong Zhao¹(✉)

¹ Department of Computer Science and Technology, Tongji University,
Shanghai 201804, China
zhaocairong@tongji.edu.cn

² Department of Computer Vision Technology (VIS), Baidu Inc., Beijing, China

³ College of Computer Science and Software Engineering, Shenzhen University,
Shenzhen 518060, China

Abstract. The 4D auto-labeling system, with its potential to enhance data annotation efficiency for 3D object detection, has garnered significant attention. However, its adoption has been hampered by the high costs associated with temporally annotated long-sequential training data and limited generalization capabilities across diverse scenarios. In this paper, we hypothesize that a multi-dataset approach can address these challenges and, accordingly, first introduce a Unified training pipeline for multi-dataset 4D Auto-Labeling, namely Uni4DAL. We recognize that this is a challenging task, primarily due to data-level variations and feature-level inconsistencies among various datasets. Motivated by this understanding, we initially propose a series of Data-Level Alignment (DLA) operations to mitigate potential discrepancies between diverse datasets and ensure synchronized training progress across samples from multiple datasets. Furthermore, to address feature-level inconsistencies, we introduce the Mixed Expert Models Voxel Feature Encoding (MoE-VFE) module, which aims to extract both domain-specific and domain-generalizable features. Additionally, we employ a Domain-Adaptive Hard Example Mining (DA-HEM) technique to leverage both data-level and feature-level consistencies, ensuring that the model pays enhanced attention to the challenging samples during multi-dataset training. Finally, comprehensive experiments demonstrate that Uni4DAL significantly improves performance on the nuScenes, Argoverse2, and Waymo datasets, and exhibits greater robustness with insufficient training data.

Keywords: Offline 3D object detection · Multi-dataset training · 4D auto-labeling

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-78113-1_12.

1 Introduction

3D object detection for autonomous driving has gathered significant interest and undergone rapid technological advancements in recent years. However, the training of deep learning-based 3D object detectors often requires a substantial amount of manually annotated data, which is both time-consuming and labor-intensive. To address this issue and reduce the dependency on manual labor, researchers [5, 11, 13, 21] have increasingly focused on the development of high-quality offline detectors for auto-labeling purposes. In a 4D auto-labeling system, the long-term offline 3D object detector serves as a critical component. As emphasized in the work of CTRL [5], these detectors leverage the entire sequential point cloud and utilize temporal context to achieve higher performance compared to state-of-the-art (SOTA) online 3D object detectors.

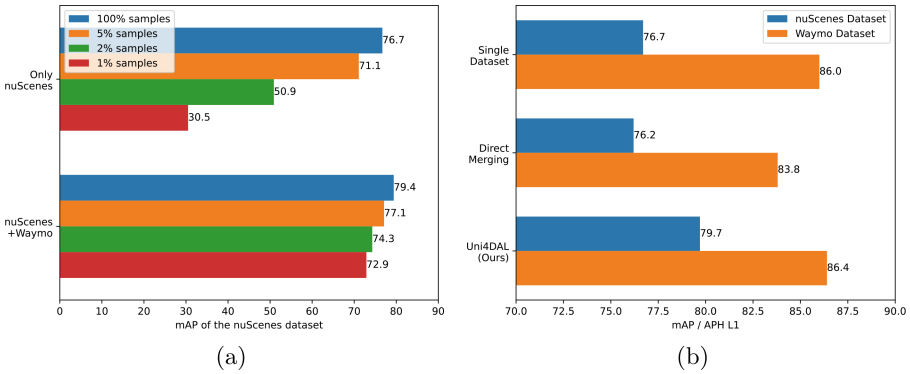


Fig. 1. Challenges in training a 4D auto-labeling detector. We conduct our experiments on the cyclist category of the nuScenes dataset and the Waymo dataset. Figure (a) illustrates a comparative analysis between single-dataset training and our proposed Uni4DAL method when confronted with inadequate training samples. The results indicate that 4D auto-labeling detectors trained solely on single dataset exhibit considerably degraded performance, while our proposed Uni4DAL achieves significant improvements. Figure (b) compares the performance of single-dataset training, direct merging, and our Uni4DAL approach. Notably, the performance of directly merging datasets is inferior to single-dataset training, while our proposed Uni4DAL surpasses other training methodologies.

However, two major challenges hinder the widespread adoption of 4D auto-labeling systems: (1) the requirement for a vast quantity of high-quality, temporally annotated training data; (2) the limited generalization capability of the resulting models, which hinders their overall performance [8, 16, 23]. As depicted in Fig. 1a, the lack of sufficient training data significantly degrades the detection performance, thereby limiting the effectiveness of these methods in diverse autonomous driving scenarios.

To enhance the generalization capability of models, research on object detection has demonstrated that multi-dataset training strategies [3, 8, 16, 19, 23] migrate the domain gap between various datasets. However, based on our experiments, directly combining different datasets and conducting multi-dataset training adversely affects the overall performance, as depicted in Fig. 1b. We attribute this to data-level variations and feature-level inconsistencies across various datasets, caused by the domain gap shift among distinct LiDAR point clouds across datasets.

In this work, we propose a Unified training pipeline for multi-dataset 4D Auto-Labeling, namely Uni4DAL. Firstly, to address the inferior performance caused by data-level variations, we implement a series of Data-Level Alignment (DLA) strategies to mitigate potential discrepancies between point cloud data from various datasets and ensure synchronized training progress across multiple datasets. Next, to mitigate feature-level inconsistencies, unlike the common practices of previous research [8, 19, 23], we propose the Mixed Expert Models Voxel Feature Encoding (MoE-VFE) module to extract both domain-specific and domain-generalizable features from the aligned point cloud data. Finally, a Domain-Adaptive Hard Example Mining (DA-HEM) technique is employed to identify and select challenging, or “hard” examples from the datasets. The selected hard examples are over-sampled during multi-dataset training, focusing the model’s attention on those instances that are most difficult to classify or localize.

Comprehensive experiments demonstrate that Uni4DAL significantly enhances performance on the nuScenes, Argoverse2, and Waymo datasets. Furthermore, Uni4DAL exhibits greater robustness with insufficient training data, thereby emphasizing its aptness for real-world applications in the context of autonomous driving. To the best of our knowledge, we are the first to investigate the integration of multi-dataset training for offline 4D detection. Our key contributions are summarized as follows:

- We introduce a novel offline multi-dataset 3D object detection method, termed Uni4DAL, which utilizes long-term multi-dataset sequences for 4D auto-labeling.
- Uni4DAL incorporates a series of Data-Level Alignment (DLA) techniques to mitigate potential discrepancies in data-level variations, alongside the Mixed Expert Models Voxel Feature Encoding (MoE-VFE) module, which addresses feature-level inconsistencies.
- Uni4DAL proposes the Domain-Adaptive Hard Example Mining (DA-HEM) technique that enhances the attention to the challenging samples during multi-dataset training, leveraging data-level and feature-level consistencies.
- Extensive experiments demonstrate that Uni4DAL significantly improves performance on the nuScenes, Argoverse2, and Waymo datasets, and exhibits greater robustness with insufficient training data.

2 Related Work

2.1 LiDAR-Based 3D Object Detection

Based on their spatial sparsity, previous LiDAR-based 3D object detectors can be classified into three categories: dense detectors, semi-dense detectors, and sparse detectors. VoxelNet [24] pioneered the use of dense convolutions for voxel feature extraction. PointPillars [7] applies 2D dense convolutions on Bird’s Eye View (BEV) feature maps to enhance computational efficiency. As a pioneering work in the field of semi-dense detectors, SECOND [20] employs sparse convolution to extract 3D sparse voxel features. CenterPoint [22] further established a robust baseline with improved performance. FSD [4], being the first work to adopt a fully sparse architecture, develops a comprehensive sparse pipeline and mitigates time-consuming operations in purely point-based methods. Furthermore, FSD has been adapted to process tracklet data with extensive spatial spans for offline auto-labeling in CTRL [5], which is also utilized in our proposed Uni4DAL.

2.2 4D Object Automated Labeling

Accurate auto-labeling for data-driven models, which is critical due to costly annotations, has attracted increasing attention in both academia and industry. However, online detectors, although real-time, suffer from limited performance as they cannot fully utilize temporal context. Conversely, offline detectors analyze the entire sequence data, capturing temporal patterns and dependencies that improve detection accuracy. Here, the pioneering work, 3DAL [13], proposes an object-centric offline 3D object automated labeling pipeline, utilizing an off-the-shelf 3D object detector and 3D multi-object tracker to process sequential point cloud data, followed by refinement through a well-designed network. Subsequently, Auto4D [21] presents a trajectory-centric pipeline to refine the size and motion path of objects. Recently, DetZero [11] proposes a pipeline with an attention-mechanism-based refining module to leverage long-term temporal contextual information. Additionally, CTRL [5] introduces a track-centric perspective which incorporates all points and proposals from each tracklet, refining all boxes simultaneously. However, the requirements of temporally annotated training data and limited generalization capabilities hinder the application of 4D auto-labeling.

2.3 Multi-dataset Training

Employing a unified model trained on multiple datasets has been a prevalent approach for traditional 2D perception tasks, including object detection [3] and semantic segmentation, due to its demonstrated enhancement in robustness and generalization capabilities. Various methods have been introduced to integrate image datasets for applications such as object detection, image segmentation, and depth estimation. In the realm of 3D vision, Uni3D [23] employs dataset-specific detection heads and feature re-coupling techniques to train a unified 3D

object detector. PPT [19] proposes a methodology to pre-train a point cloud segmentation network utilizing data from multiple datasets. However, the subsequent fine-tuning of the pre-trained weights on individual datasets undermines the universal learning approach. M3Net [8] designs a unified framework for multi-task, multi-dataset, multi-modality LiDAR segmentation. In our work, we are the first to introduce a multi-dataset training strategy into the realm of offline 3D object detection.

3 Method

3.1 Framework Overview

We select CTRL [5] as our baseline, which is a LiDAR-based offline 3D detector with high performance and low resource requirements. Moreover, its simpler pipeline is more suitable for engineering implementation.

The overall pipeline of our proposed Uni4DAL framework is illustrated in Fig. 2. Initially, the base 3D object detector and 3D multi-object tracker process LiDAR points (and/or multi-view images) to yield tracking results, wherein each tracked object is assigned a unique tracking ID. Secondly, the Uni4DAL framework takes Object-Centric Long-Term Sequential Point Cloud (OCLT-PC) data as its input. This data serves as the foundation for subsequent processing steps. Further details regarding OCLT-PC are provided in the supplementary material.

Subsequently, the DA-HEM technique is utilized to identify and select challenging, or “hard” examples from the datasets. The selected hard examples are then over-sampled during multi-dataset training, directing the model’s attention towards those instances that are more difficult to classify or localize. Following the DA-HEM step, DLA strategies are implemented to mitigate potential discrepancies between the various datasets. Next, the MoE-VFE module is introduced to extract both domain-specific and domain-generalizable features from the aligned point cloud data. Finally, a 3D backbone network and a detection head are employed to extract relevant features from both datasets. The 3D backbone functions as the primary feature extractor, while the detection head utilizes these features to generate classification and localization results. A more detailed description is provided as follows.

3.2 Data-Level Alignment

Contrary to traditional 2D image-based perception tasks, 3D point clouds are captured by a diverse array of LiDAR sensors, introducing variations such as differences in reflection intensity distributions and perspective ranges. These inconsistencies significantly hinder the effectiveness of multi-dataset training. Additionally, the varying positions of LiDAR sensors across datasets result in disparities in ground height, further complicating the challenges associated with multi-dataset training. To mitigate these issues and enhance the performance of multi-dataset training, we employ three Data-Level Alignment (DLA) strategies:

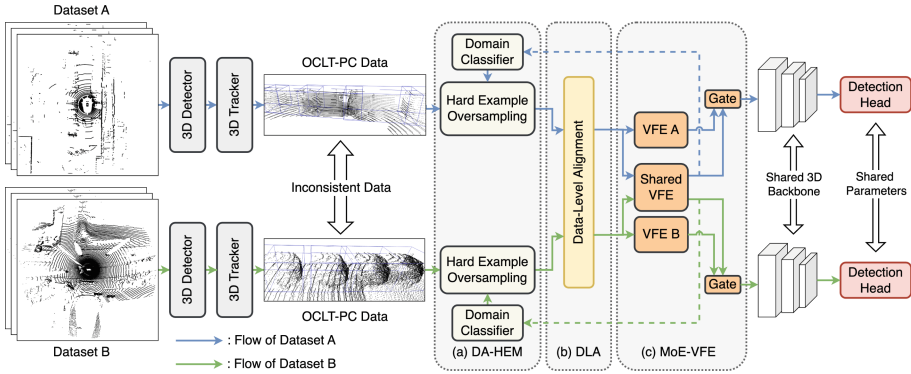


Fig. 2. Overview of the framework of Uni4DAL. Firstly, the Uni4DAL framework takes the Object-Centric Long-Term sequential Point Cloud data (OCLT-PC) as its inputs. Subsequently, a Domain-Adaptive Hard Example Mining (DA-HEM) technique is employed to identify and select hard examples. These selected examples are over-sampled during multi-dataset training. Following the DA-HEM step, Data-Level Alignment (DLA) strategies are implemented to mitigate potential discrepancies between different datasets. Next, the Mixed Expert Models Voxel Feature Encoding (MoE-VFE) module is designed to extract both domain-specific and domain-generalizable features. Finally, a 3D backbone and a detection head are utilized to extract features from both datasets and output classification and localization results.

point cloud reflection intensity alignment, point cloud range alignment, and point cloud ground height alignment. In this section, we primarily focus on the latter strategy, namely ground height alignment. A more detailed description of these alignment strategies is provided in the supplementary material.

Furthermore, given the diverse hyper-parameters and data processing pipelines across different datasets, some methods [23] utilize multiple data loaders during the training process, each fetching an equal number of samples from their respective datasets. However, the imbalance in sample scales across datasets frequently results in an imbalanced training process and sub-optimal performance. In this section, we introduce a simple yet effective method, named the Balanced Batch Size Strategy (BBSS), for aligning the sample scales at the data level across diverse datasets.

Point Cloud Ground Height Alignment. To mitigate discrepancies in sensor installations across diverse datasets, we introduce a point cloud ground height alignment operation. Specifically, we calculate the mean ground height of labels in the training set for various datasets and categories. Subsequently, we adjust the coordinate origins of both point clouds and labels to align them to a common ground height reference. Furthermore, Z-axis data augmentation is applied to point clouds and labels to enhance detection robustness. Empirical results demonstrate that this alignment technique significantly mitigates the degradation caused by variations in sensor configurations.

Balanced Batch Size Strategy. To address the problems of imbalanced sample scales, we introduce the Balanced Batch Size Strategy (BBSS), a novel training approach designed for alignment across diverse datasets. In this strategy, the training batch size is dynamically adjusted based on the proportional sample counts from multiple datasets. Specifically, if dataset A contains N_A samples and dataset B comprises N_B samples, the balanced batch sizes for datasets A and B are defined as $B_A = N_A/(N_A + N_B) \times B_S$ and $B_B = N_B/(N_A + N_B) \times B_S$, where B_A and B_B represent the balanced batch sizes for datasets A and B, respectively, and B_S denotes the total batch size after balancing. The Balanced Batch Size Strategy ensures synchronized training across multiple datasets, thereby achieving superior performance compared to using a uniform batch size across multiple datasets.

3.3 Cross-Domain Feature Extraction

To enhance cross-domain feature interaction, previous studies [8, 19, 23] have introduced domain-decoupled normalization modules for learning generalized representations across diverse datasets. However, while mitigating feature-space discrepancies through conventional 2D or 3D backbones, these approaches overlook the beneficial role of domain-specific features, thus limiting their overall performance. A similar concern has been observed by [1], which exploits the domain-specific features to enhance domain generalization ability.

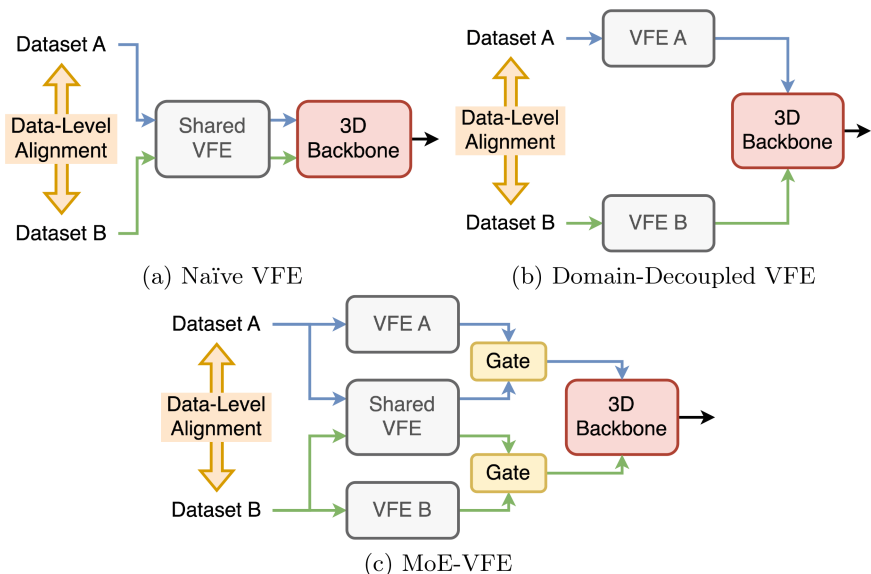


Fig. 3. Framework of VFE modules in Uni4DAL. VFE denotes Voxel Feature Encoding module.

To address this limitation and effectively extract both domain-specific and domain-generalizable features, we introduce a novel yet efficient module, named the Mixed Expert Models Voxel Feature Encoding (MoE-VFE), as illustrated in Fig. 3c. Inspired by the Mixed Expert Models (MoE) [6, 14], we design a comprehensive framework that incorporates a mixture of multiple VFE modules. These VFE modules can be categorized into two distinct parts: (1) the Naïve VFE module, depicted in Fig. 3a, which serves as a baseline for common 3D point cloud networks, extracting general features from the input data; (2) the Domain-Decoupled VFE module, presented in Fig. 3b, which can be considered as the specialized VFE modules within the MoE-VFE framework (depicted in Fig. 3c). Specifically, one VFE module is employed to extract shared features F_{share} , whereas N VFE modules are dedicated to extracting domain-specific features from each of the N datasets. Subsequently, a gating mechanism is employed to effectively fuse domain-specific and domain-generalizable features.

$$F_{sep} = [F_1, F_2, \dots, F_N] \quad (1)$$

$$G_\sigma(x) = \text{Softmax}(MLP_\sigma(x)) \quad (2)$$

$$F_m = G_{share}([F_{share}, F_{sep}])F_{share} + G_{sep}([F_{share}, F_{sep}])F_{sep} \quad (3)$$

Where $[\cdot]$ denotes the concatenate operation, F_{share} denotes the shared features from the shared VFE, F_{sep} denotes the specialized features from N specialized VFES. These fused features F_m are then fed into a 3D backbone network (namely, Sparse UNet [15] in our framework) without any domain-decoupled normalization modules. Our proposed MoE-VFE module ensures that both the generalized and specialized information are leveraged optimally, thereby enhancing the overall performance of the model across different domains. It outperforms both the Naïve VFE module and the Domain-Decoupled VFE module, while maintaining a straightforward framework and high computational efficiency.

3.4 Domain-Adaptive Hard Example Mining

Despite the capabilities of data-level alignment strategies and the MoE-VFE module in extracting domain-specific and domain-generalizable features, challenges persist with respect to a subset of samples. Unlike conventional positive and negative samples, these samples typically exhibit unique characteristics, including longer tracklet lengths, higher detection confidence, but sparser point clouds. Further statistical analysis is available in the supplementary materials. Such samples pose significant challenges during multi-dataset training, as they have the potential to disrupt the training process for other datasets.

To mitigate this issue and facilitate the identification of these challenging samples, we define them as “hard examples” and propose a domain-adaptive mining strategy, specifically, the Domain-Adaptive Hard Example Mining (DAHEM) strategy. Firstly, we develop a lightweight framework that comprises solely the MoE-VFE module and an MLP-based classifier, as depicted in Fig. 4. The

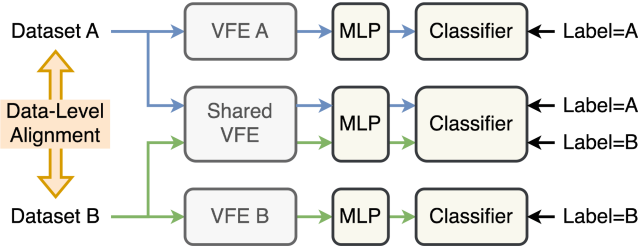


Fig. 4. Framework of Domain-Adaptive Hard Example Mining (DA-HEM) in Uni4DAL. VFE denotes Voxel Feature Encoding module.

classifier’s task is to assign domain labels to voxel features originating from various domains. Thanks to the lightweight structures of the MoE-VFE module and the MLP-based classifier, the training process does not consume excessive resources. Secondly, we identify and select the samples with the top-k lowest mean classification confidence scores across all voxel features within these samples. Finally, during the multi-dataset training process, these hard examples are over-sampled to enhance the models’ ability to learn from these potentially ambiguous samples.

4 Experiments

4.1 Experimental Setups

Dataset and Metrics. We conduct experiments on the nuScenes dataset [2], the Waymo Open Dataset [17], and the Argoverse2 dataset [18]. The nuScenes dataset comprises 1000 scenes, each capturing a duration of 20 s. These scenes are recorded by six cameras operating at 12 Hz and a 32-beam LiDAR sensor at 20 Hz. To maintain synchronization between the cameras and LiDAR, the annotated key frames are set to a frequency of 2 Hz. We evaluate our approach using mean Average Precision (mAP) for each category. The Waymo Open Dataset contains 1150 LiDAR scenes, each providing point cloud data captured over 20 s, sampled at a frequency of 10 Hz. We adhere to the official evaluation protocol and assess our approach using average precision (AP) and average precision weighted by heading (APH) on LEVEL 1 (L1) difficulty levels for each category. The Argoverse2 dataset encompasses 1000 scenes, where each scene provides point cloud data captured by two 32-beam LiDAR sensors over 15 s, sampled at a frequency of 10 Hz. We evaluate our approach using mean Average Precision (mAP) for each category.

Training Strategy. For our proposed approach, we follow the pipeline of CTRL [5], incorporating various data augmentation strategies, including global random flipping, rotation, scaling, and the injection of random box noise. Furthermore, we concatenate box size and rotation information with the point cloud to enrich

the feature representation. We employ AdamW [10] optimizer with a weight decay of 1×10^{-2} to optimize our model, and employ a cosine annealing strategy [9] to decay the learning rate. The models are trained on 8 Nvidia V100 GPUs. Further details are provided in the supplementary material.

Table 1. Specifics of the merged categories from different datasets.

Waymo [17]	nuScenes [2]	Argoverse2 [18]
Vehicle	Bus, Construction Vehicle, Car, Trailer, Truck	Articulated Bus, Box Truck, Bus, Large Vehicle, Regular Vehicle, School Bus, Truck, Vehicular Trailer
Pedestrian	Pedestrian	Pedestrian
Cyclist	Bicycle, Motorcycle	Bicycle, Bicyclist, Motorcycle, Motorcyclist

Merging Similar Categories. To maintain label space consistency across multi-dataset training workflows and address the issue of insufficient training data for long-tail categories within the nuScenes [2] and Argoverse2 [18] datasets, we implement a strategy to merge comparable categories during the training phase. Specifically, we consolidate the tracklets belonging to vehicle-like categories into a unified group, and similarly, combine the tracklets of cyclist-like entities into another distinct group. This approach ensures that the model can leverage a more balanced and comprehensive representation of these categories. Notably, during the inference stage, these merged categories are evaluated individually to preserve their distinctiveness and accuracy. The specifics of the merged categories are detailed in Table 1.

4.2 Main Results of Multi-dataset 4D Auto Labeling

Joint Training on nuScenes and Waymo Datasets. To investigate the effectiveness of multi-dataset 4D automatic labeling, we conduct a comparative analysis between our proposed Uni4DAL approach and the single-dataset 4D auto labeling strategy, specifically CTRL [5]. This evaluation is performed on both the nuScenes [2] validation set and the Waymo [17] validation set, and the results are shown in Table 2. Our proposed Uni4DAL outperforms CTRL (trained using a single-dataset strategy) by 0.8%/0.6%/3.0% in mAP on the nuScenes dataset and 0.3%/0.5%/0.4% in APH L1 on the Waymo dataset, as well as CTRL (trained using a multi-dataset strategy) by 0.9%/0.4%/3.5% in mAP on the nuScenes dataset and 0.1%/3.2%/2.6% in APH L1 on the Waymo dataset, demonstrating the effectiveness of our proposed Uni4DAL on both datasets owing to the well-designed multi-dataset training strategies.

Table 2. Results of joint training on the nuScenes, Argoverse2 and Waymo datasets. We report mAP metrics on the nuScenes dataset, mAP metrics on the Argoverse2 dataset, and AP and APH of LEVEL 1 metrics on the Waymo dataset. The superscript [†] indicates that we modified the input of CTRL [5] to accommodate the multi-dataset pipeline, without any improvements from our paper. The notion of category: Vehicle (Veh.), Pedestrian (Ped.), Cyclist (Cyc.). The notion of dataset: nuScenes (N), Argoverse2 (A), Waymo (W).

Trained on	Method	nuScenes			Argoverse2			Waymo		
		Veh.	Ped.	Cyc.	Veh.	Ped.	Cyc.	Veh.	Ped.	Cyc.
only nuScenes	CTRL [5]	65.0	88.4	76.7	–	–	–	–	–	–
only Argoverse2	CTRL [5]	–	–	–	36.2	65.7	52.3	–	–	–
only Waymo	CTRL [5]	–	–	–	–	–	–	87.2/86.6	87.3/84.6	86.9/86.0
N+W	CTRL [†]	64.9	88.6	76.2	–	–	–	87.4/86.8	85.1/81.9	84.7/83.8
N+A	CTRL [†]	65.2	88.3	76.3	36.4	65.1	46.5	–	–	–
A+W	CTRL [†]	–	–	–	37.0	65.1	53.9	86.3/85.7	87.0/84.0	85.4/84.5
N+W	Uni4DAL(Ours)	65.8	89.0	79.7	–	–	–	87.5/86.9	87.9/85.1	87.3/86.4
N+A	Uni4DAL(Ours)	65.7	88.8	80.2	36.6	65.7	55.4	–	–	–
A+W	Uni4DAL(Ours)	–	–	–	37.4	66.1	55.5	87.4/86.8	87.5/84.8	87.2/86.2

Joint Training on nuScenes and Argoverse2 Datasets. We also conduct experiments on both the nuScenes [2] validation set and the Argoverse2 [18] validation set. The results, summarized in Table 2, indicate that Uni4DAL surpasses CTRL (single-dataset) by 0.7%/0.4%/3.5% in terms of mAP on the nuScenes dataset and by 0.4%/0.0%/3.1% on the Argoverse2 dataset, as well as CTRL (multi-dataset) by 0.5%/0.5%/3.9% in terms of mAP on the nuScenes dataset and by 0.2%/0.6%/8.9% on the Argoverse2 dataset.

Joint Training on Argoverse2 and Waymo Datasets. We further conduct experiments on both the Argoverse2 [18] validation set and the Waymo [17] validation set. The results, summarized in Table 2, indicate that Uni4DAL surpasses CTRL (single-dataset) by 1.2%/0.4%/3.2% mAP on the Argoverse2 dataset, and by 0.2%/0.2%/0.2% APH L1 on the Waymo dataset. Furthermore, Uni4DAL surpasses CTRL (multi-dataset) by 0.4%/1.0%/1.6% mAP on the Argoverse2 dataset, and by 1.1%/0.8%/1.7% APH L1 on the Waymo dataset, indicating that directing merging multiple dataset may degrade the performance.

4.3 Ablation Studies

Ablation Studies of Components in Multi-dataset Training on NuScenes and Waymo. To validate the effectiveness and universality of our proposed Uni4DAL framework, we conducted ablation studies of all components on the nuScenes and Waymo datasets, as presented in Table 3. The first

Table 3. Ablation of each component on joint training on the nuScenes and Waymo dataset. We report mAP metrics on the nuScenes dataset, and AP and APH of LEVEL 1 metrics on the Waymo dataset. DLA represents Data-Level Alignment strategies. MoE-VFE represents Mixed Expert Models Voxel Feature Encoding module. DA-HEM represents Domain-Adaptive Hard Example Mining. The notion of category: Vehicle (Veh.), Pedestrian (Ped.), Cyclist (Cyc.). The notion of dataset: nuScenes (N), Waymo (W).

Trained on	DLA	MoE-VFE	DA-HEM	nuScenes			Waymo		
				Veh.	Ped.	Cyc.	Veh.	Ped.	Cyc.
only nuScenes				65.0	88.4	76.7	–	–	–
only Waymo				–	–	–	87.2/86.6	87.3/84.6	86.9/86.0
N+W				64.9	88.6	76.2	87.4/86.8	85.1/81.9	84.7/83.8
N+W	✓			65.1	88.6	79.2	87.4/86.8	87.4/84.7	86.8/85.9
N+W	✓	✓		65.2	88.9	79.4	87.5/86.9	87.9/85.1	87.3/86.4
N+W	✓	✓	✓	65.8	89.0	79.7	87.5/86.9	87.9/85.1	87.3/86.4

row, labeled “N+W”, demonstrates that directly merging the two datasets without proper alignment or module enhancements results in significantly inferior performance compared to training on a single dataset. The second row indicates that employing data-level alignment strategies enhances performance by 0.2%/0.0%/3.0% on the nuScenes dataset and 0.0%/2.3%/2.8% on the Waymo dataset. Notably, the slightly sub-optimal performance on the vehicle category in the nuScenes dataset may be attributed to the varying average heights among its five sub-categories. The third row shows that our proposed MoE-VFE module improves performance by 0.1%/0.3%/0.2% on the nuScenes dataset and 0.1%/0.4%/0.5% on the Waymo dataset. These results suggest the module’s ability to extract domain-specific and domain-generalizable features from point cloud data. Lastly, the fourth row demonstrates that our DA-HEM module enhances performance by 0.6%/0.1%/0.3% on the nuScenes dataset, while it does not yield significant improvements on the Waymo dataset. We attribute this sub-optimal performance to the relatively higher domain classification confidence scores achieved on the Waymo dataset, which subsequently results in fewer hard examples being selected for training.

Ablation Studies of the MoE-VFE Module. To thoroughly evaluate the effectiveness of the Mixed Expert Models Voxel Feature Encoding (MoE-VFE) module, we conduct an ablation study comparing its performance with several alternative VFE modules on the cyclist category within the nuScenes, Argoverse2 and Waymo datasets. Specifically, we analyze the performance of the MoE-VFE module with the following VFE modules: (1) the Naïve VFE module, depicted in Fig. 3a; (2) the Domain-Decoupled VFE module, presented in Fig. 3b. As summarized in Table 4, our findings indicate that the Domain-Decoupled VFE module outperforms the Naïve VFE module. Furthermore, the proposed MoE-VFE

module exhibits even superior performance, thereby validating the effectiveness of extracting both domain-specific and domain-generalizable features.

Table 4. Ablation on the Mixed Expert Models Voxel Feature Encoding (MoE-VFE) module and other VFE modules.

Trained on	Setting	nuScenes	Argoverse2	Waymo
		Cyclist	Cyclist	Cyclist
nuScenes+Waymo	Naïve VFE	79.2	–	86.8/85.9
nuScenes+Waymo	Domain-decoupled VFE	79.1	–	87.2/86.2
nuScenes+Waymo	MoE-VFE (ours)	79.4	–	87.3/86.4
nuScenes+Argoverse2	naïve VFE	79.0	53.3	–
nuScenes+Argoverse2	domain-decoupled VFE	79.8	54.4	–
nuScenes+Argoverse2	MoE-VFE (ours)	80.0	54.9	–

4.4 Further Analyses

Experiments on Reduced Training Samples. Given the substantial costs associated with human-annotated long-sequential training data, in practical applications, collecting sufficient LiDAR data for training 4D auto-labeling models may be impractical, resulting in inferior performance. In this context, Uni4DAL proposes an alternative approach to alleviate the requirement for extensive training data by leveraging multi-dataset training, utilizing both the target dataset and an additional source dataset. As demonstrated in Table 5, joint training with few-shot data from the nuScenes dataset and the Waymo

Table 5. Results of reducing the number of samples in the nuScenes dataset under the setting of nuScenes-Waymo joint training. The notion of category: Vehicle (Veh.), Pedestrian (Ped.), Cyclist (Cyc.).

Trained on	Sample Ratio	nuScenes		
		Veh.	Ped.	Cyc.
only nuScenes	100%	65.0	88.4	76.7
only nuScenes	5%	60.3	87.1	71.1
only nuScenes	2%	58.0	86.1	50.9
only nuScenes	1%	54.8	43.6	30.5
nuScenes+Waymo	100%	65.2 (+0.2)	88.9 (+0.5)	79.4 (+2.7)
nuScenes+Waymo	5%	61.5 (+1.2)	88.1 (+1.0)	77.1 (+6.0)
nuScenes+Waymo	2%	60.8 (+2.8)	87.4 (+1.3)	74.3 (+23.4)
nuScenes+Waymo	1%	59.3 (+4.5)	86.4 (+42.8)	72.9 (+42.4)

dataset significantly improves performance on the nuScenes dataset, in comparison to training solely on the nuScenes dataset. This improvement is primarily attributed to Uni4DAL’s capability to extract generalized features, which reduces the risk of over-fitting when dealing with limited nuScenes data. These experiments validate Uni4DAL’s ability to reduce the data dependency of 4D auto-labeling models in scenarios where only insufficient training data is available for the target dataset.

t-SNE Visualization of the MoE-VFE Module. To further investigate the effectiveness of our proposed MoE-VFE module, we gather voxel-level features from various VFE modules and visualize them using the t-SNE [12] technique. Figure 5a depicts the features extracted from the Naïve VFE (as demonstrated in Fig. 3a), where features from distinct domains share a relatively unified feature space. A similar trend is observed in Fig. 5b, indicating that the shared component within the MoE-VFE (as depicted in Fig. 3c) is also capable of extracting a more coherent set of features across diverse datasets. In contrast, as illustrated in Fig. 5c, the features from the two specialized components within the MoE-VFE exhibit a notably diverse distribution. These experimental results validate the capability of our proposed MoE-VFE module in extracting both domain-specific and domain-generalizable features, ultimately resulting in enhanced performance.

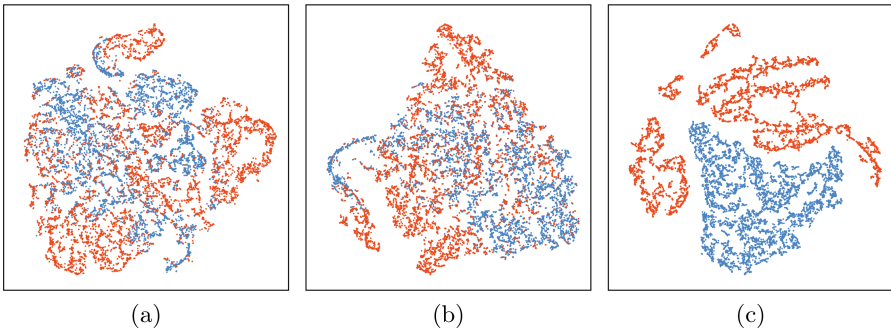


Fig. 5. t-SNE Visualization of the MoE-VFE Module. Figure (a) represents the features extracted by the Naïve VFE module. Figure (b) depicts the features of the shared component within the MoE-VFE module, whereas Fig. (c) illustrates the features of the specialized components within the MoE-VFE module.

5 Conclusion

This paper proposed a novel and high-performance offline multi-dataset 3D object detection method for processing long-term sequential multi-modal data,

named Uni4DAL. Aiming to mitigate the data-level variations and feature-level inconsistencies among various datasets, Uni4DAL leverages data-level alignment operations, enhanced cross-domain feature extraction and improved training strategies to mitigate the inconsistencies among various dataset during multi-dataset training process. The improved generalization ability and greater robustness, even with insufficient training data, demonstrate the effectiveness of Uni4DAL. We hope that this simple pipeline design can provide further insights into multi-dataset offline 3D object detection and 4D auto-labeling systems.

References

1. Bui, M.H., Tran, T., Tran, A., Phung, D.: Exploiting domain-specific features to enhance domain generalization. *Adv. Neural. Inf. Process. Syst.* **34**, 21189–21201 (2021)
2. Caesar, H., et al.: nuScenes: a multimodal dataset for autonomous driving. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11621–11631 (2020)
3. Chen, Y., et al.: ScaleDet: a scalable multi-dataset object detector. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7288–7297 (2023)
4. Fan, L., Wang, F., Wang, N., ZHANG, Z.X.: Fully sparse 3D object detection. In: Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., Oh, A. (eds.) *Advances in Neural Information Processing Systems*, vol. 35, pp. 351–363. Curran Associates, Inc. (2022)
5. Fan, L., et al.: Once detected, never lost: surpassing human performance in offline lidar based 3D object detection. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 19820–19829, October 2023
6. Fedus, W., Zoph, B., Shazeer, N.: Switch transformers: scaling to trillion parameter models with simple and efficient sparsity. *J. Mach. Learn. Res.* **23**(120), 1–39 (2022)
7. Lang, A.H., Vora, S., Caesar, H., Zhou, L., Yang, J., Beijbom, O.: PointPillars: fast encoders for object detection from point clouds. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12697–12705 (2019)
8. Liu, Y., et al.: Multi-space alignments towards universal lidar segmentation. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14648–14661 (2024)
9. Loshchilov, I., Hutter, F.: SGDR: stochastic gradient descent with warm restarts. In: *International Conference on Learning Representations* (2017)
10. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: *International Conference on Learning Representations* (2019)
11. Ma, T., et al.: DetZero: rethinking offboard 3D object detection with long-term sequential point clouds. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6736–6747, October 2023
12. Van der Maaten, L., Hinton, G.: Visualizing data using t-SNE. *J. Mach. Learn. Res.* **9**(11), 2579–2605 (2008)
13. Qi, C.R., et al.: Offboard 3D object detection from point cloud sequences. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6134–6144 (2021)
14. Shazeer, N., et al.: Outrageously large neural networks: the sparsely-gated mixture-of-experts layer. In: *International Conference on Learning Representations* (2017)

15. Shi, S., Wang, Z., Shi, J., Wang, X., Li, H.: From points to parts: 3D object detection from point cloud with part-aware and part-aggregation network. *IEEE Trans. Pattern Anal. Mach. Intell.* **43**(8), 2647–2664 (2020)
16. Soum-Fontez, L., Deschaud, J.E., Goulette, F.: MDT3D: multi-dataset training for lidar 3D object detection generalization. In: 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 5765–5772. IEEE (2023)
17. Sun, P., et al.: Scalability in perception for autonomous driving: Waymo open dataset. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2446–2454 (2020)
18. Wilson, B., et al.: Argoverse 2: next generation datasets for self-driving perception and forecasting. In: Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks (NeurIPS Datasets and Benchmarks 2021) (2021)
19. Wu, X., et al.: Towards large-scale 3D representation learning with multi-dataset point prompt training. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 19551–19562 (2024)
20. Yan, Y., Mao, Y., Li, B.: Second: sparsely embedded convolutional detection. *Sensors* **18**(10), 3337 (2018)
21. Yang, B., Bai, M., Liang, M., Zeng, W., Urtasun, R.: Auto4D: learning to label 4D objects from sequential point clouds. arXiv preprint [arXiv:2101.06586](https://arxiv.org/abs/2101.06586) (2021)
22. Yin, T., Zhou, X., Krahenbuhl, P.: Center-based 3D object detection and tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11784–11793 (2021)
23. Zhang, B., Yuan, J., Shi, B., Chen, T., Li, Y., Qiao, Y.: Uni3D: a unified baseline for multi-dataset 3D object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9253–9262 (2023)
24. Zhou, Y., Tuzel, O.: VoxelNet: end-to-end learning for point cloud based 3D object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4490–4499 (2018)



Dual-Attention Fusion Network with Edge and Content Guidance for Remote Sensing Images Segmentation

Shuaipeng Ding, Jianan Shui, Xin Li, and Mingyong Li^(✉)

School of Computer and Information Science, Chongqing Normal University,
Chongqing 401331, China
limingyong@cqnu.edu.cn

Abstract. This paper investigates the problem of semantic segmentation in high-resolution remote sensing images, aiming to predict semantic labels at a pixel-level granularity. Faced with the complexity and heterogeneity inherent in high-resolution remote sensing images, which lead to challenges such as misclassification of edges and confusion in contextual information, we propose a Dual-Attention Fusion Network with Edge and Content Guidance (DAF-Net). The DAF-Net consists of three modules: (1) the edge feature extraction module, responsible for extracting boundary information; (2) the edge fusion module, which thoroughly integrates the extracted edge features with the original features to improve intra-class semantic consistency, particularly in pixels containing boundaries; (3) the content guided attention fusion module (CGA), which produces unique spatial importance maps for each channel, thereby highlighting more useful information within the features and reducing redundancy. Additionally, we introduce a CGA-based fusion strategy that more effectively integrates the features from both the encoder and the decoder. The effectiveness of DAF-Net is demonstrated through extensive experimental evaluations and ablation studies conducted on the ISPRS Vaihingen and Potsdam datasets. DAF-Net achieves notable mIoU scores of 78.73% and 83.81% on the Vaihingen and Potsdam datasets, respectively.

Keywords: Remote sensing · Semantic segmentation · Edge fusion · Content-Guided attention fusion · Transformer

1 Introduction

The semantic segmentation of remote sensing images is essential for numerous remote sensing applications, such as monitoring environmental changes, precision agriculture, environmental conservation, urban planning, and 3D modeling. The goal of semantic segmentation is to assign pixels belonging to the same category with consistent color labels. However, current high-resolution remote sensing image applications demand higher requirements for semantic segmentation. Traditional methods such as support vector machines and random forests

have become inadequate to meet these demands. Therefore, there is significant room for improvement in the performance of semantic segmentation for remote sensing images. With the rapid advancement of deep learning techniques, Convolutional Neural Networks (CNNs) have achieved outstanding performance in semantic segmentation by learning complex feature representations of images. In particular, Long et al. [11] proposed Fully Convolutional Networks (FCNs), which greatly enhance the performance of pixel-level segmentation. Ronneberger et al. [12] developed UNet, an encoder-decoder network characterized by its symmetric structure, which employs skip connections to reduce the loss of feature information during downsampling and improve segmentation accuracy. Liu et al. [9] introduced the Scale Feature Attention Module (SFAM) to increase the network's depth. However, due to the complex scenes of remote sensing images, the performance of CNN seems to reach a threshold. Because the convolution kernel of feature extraction has great limitations, it is difficult to capture the global information correlation. Global context information is crucial to fully extract features. With the swift advancement of Transformers [4] in the realm of Natural Language Processing (NLP), some researchers have attempted to apply Transformers to semantic segmentation, [4, 24]. Due to its remarkable ability in global modeling, transformer-based models have the potential to continuously improve performance. Xu et al. [20] proposed an efficient transformer to overcome the computational burden associated with Transformers.

Despite the progress made by the above methods, there are still some limitations. First, these methods largely ignore the importance of edge learning, and have not considered fully integrating the supervised edge information with the feature maps. Secondly, simply concatenating the encoder and decoder features in the decoder stage directly increases the model's parameter count, which may hinder effective information extraction and impede the free flow of information within the network. Although weighted feature fusion In the study by Tan et al. [14] can reduce computational complexity, we posit that spatial importance varies across different channels. Each channel in the feature space should have distinct semantic significance, and direct weighted fusion might disrupt the spatial specificity of these channels.

Inspired by the above literature, we propose a Dual-Attention Fusion Network with Edge and Content Guidance (DAF-Net) for remote-sensing image semantic segmentation. The main work of this paper is as follows:

- 1) We propose an edge extraction module (EEM), which uses shallow information to accurately identify edges, and generates an edge guide graph while deeply supervised learning edge prediction graph, so that it can further learn edge features from rich features.
- 2) The simple use of edge prediction graphs to enhance the edge information of feature maps may lead to spatial and semantic confusion of feature maps. To overcome this problem, we proposed an edge fusion module (EFM), which uses the edge guide graph in the edge extraction module to adaptively fuse edge information, and introduces a spatial attention mechanism to filter the fused redundant features.

- 3) We propose a Content-Guided Attention (CGA) mechanism to generate channel-specific spatial importance maps in a coarse-to-fine manner. CGA produces unique spatial importance maps for each channel, thereby highlighting more useful information within the features and reducing redundancy. Additionally, we introduce a CGA-based fusion strategy that more effectively integrates the features from both the encoder and the decoder.

2 Related Work

2.1 CNN-Based Remote Sensing Image Semantic Segmentation

The advent of FCN [11] an end-to-end semantic segmentation network, has fundamentally disrupted traditional segmentation networks. Subsequently, CNN-based networks have dominated the field. The encoder-decoder structure, with its simple design and excellent performance, has become the primary framework for semantic segmentation of remote sensing images. However, CNN's limitations in capturing fine-grained spatial and global semantic information, caused by the complexity and blurry edges of remote sensing images, have prompted the introduction of multi-scale aggregation and attention mechanisms into the networks. Li et al. [6] proposed a Multi-level Attention Reconstruction Network (MAResUNet) that incorporates LLM attention mechanism into the skip connections of UNet to address semantic disparities across different scale feature maps and establish long-range dependencies. Li et al. [7] introduced a Multi-Attention Network (MANet) that partially compensates for CNN's deficiency in capturing global information. Despite various methods being developed to alleviate the limitations of CNN, the majority of them still rely on aggregating local features extracted by CNN to form global information.

2.2 Transformer-Based Remote Sensing Image Semantic Segmentation

In recent years, Transformer [24]-based models have exhibited remarkable performance in numerous computer vision and natural language processing (NLP) tasks. With exceptional sequence modeling capabilities, Transformers have swiftly made their way into the field of semantic segmentation. Their unique self-attention mechanism and parallel computing abilities enable them to capture global contextual information, surpassing the limitations of local convolutions in CNNs and extracting more valuable information. Consequently, a plethora of segmentation methods based on Transformers have emerged, including Swin-UNet [2] and SegFormer [19]. Additionally, there are approaches that combine the advantages of both CNNs and Transformers. For example, Zhang et al. [22] introduced a hybrid architecture in which the encoder uses a Swin Transformer to capture long-range dependencies, while the decoder leverages CNN-based techniques to effectively maintain local information in the image. Dense Connection Swin (DC-Swin) [15] leverages Transformers for feature extraction and utilizes a designed feature aggregation module (DCFM) to extract multi-scale semantically enhanced feature relationships.

3 Proposed Method

In this study, we adopt Swin Transformer Base as the backbone network for extracting image features. This allows us to obtain feature maps with varying receptive fields. We leverage the first three shallow-level features to extract edge features. These features are then inputted into the Edge Extraction Module (EEM). To better capture edge information, the Edge Truth Supervision Module is employed. It facilitates the learning of edge information by supervising the boundaries produced by the labels. The extracted edge features are subsequently merged with the original feature map using the Edge Fusion Module (EFM). Finally, in the decoder, the aggregated information is passed through skip connections for feature integration and is further processed by the Content-Guided Attention Fusion module (CGAF) to extract key regions. The overall architecture of DAF-Net is illustrated in Fig. 1.

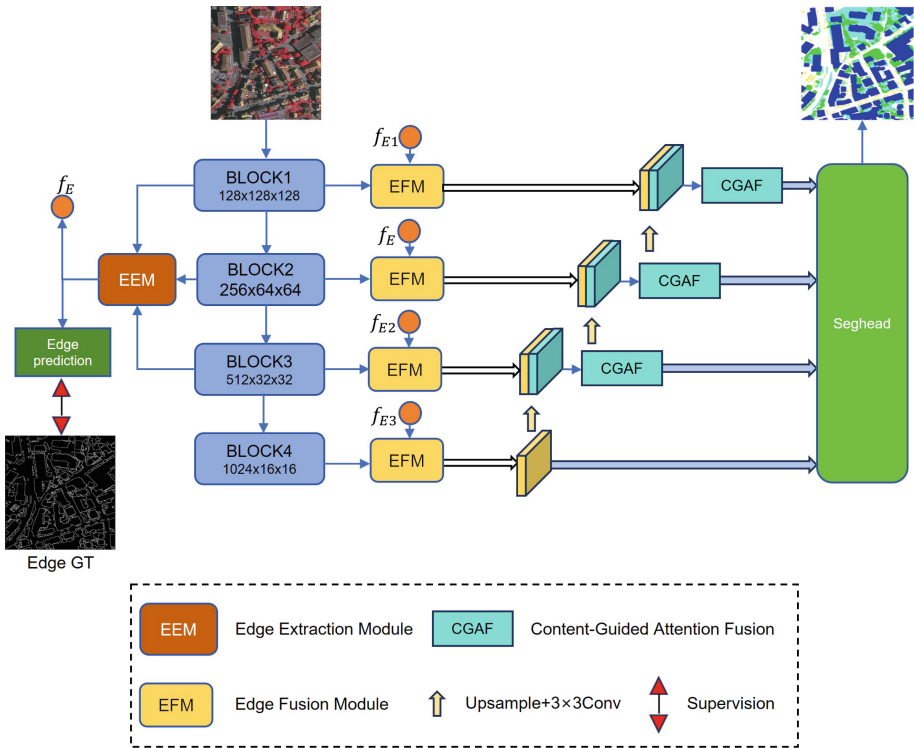


Fig. 1. The overall architecture of the proposed DAF-Net consists of four main components: an encoder, a decoder, an edge extraction and fusion module, and content-guided attention fusion module.

3.1 EEM

Edge information significantly contributes to enhancing the segmentation performance of remote sensing images. To achieve accurate extraction of edge information, we introduce the EEM as depicted in the Fig. 2.

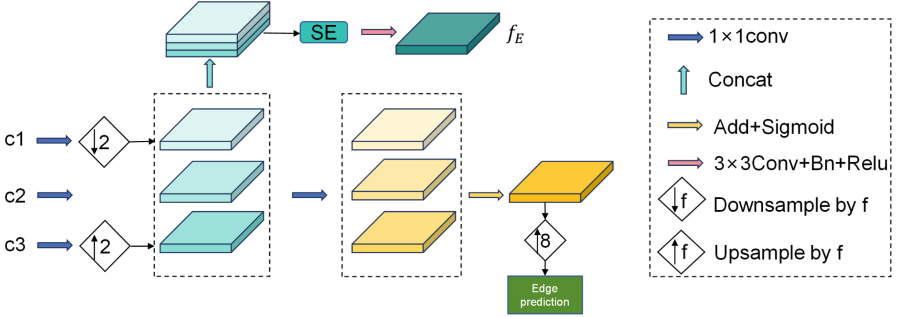


Fig. 2. Illustration of EEM. Monitor learning edges and generate edge guide graphs.

The EEM module is designed to extract precise edge information from remote sensing images. To maintain a higher resolution while preserving more semantic information, we utilize the first three feature maps (c_1 , c_2 , c_3) to extract edge information. In the EEM module, these three feature maps are first unified to the channel dimension of c_2 through 1×1 convolutions. Subsequently, c_1 undergoes downsampling while c_3 undergoes upsampling to match the size of c_2 . They are then concatenated and passed through a Channel Attention Module (SE) [5], to adaptively learn feature channel information from different feature maps. Finally, a 3×3 convolutional layer is applied to extract edge guide graph. To better learn edge information, we incorporate boundary ground truth supervision generated from labels. With shared parameters, each of the three feature maps of the same scale is separately convolved by a 1×1 kernel to produce single-channel feature maps, which are element-wise summed and passed through a sigmoid function to generate the boundary probability map. The aforementioned process can be represented as follows.

$$C_{total} = \text{Cat}(\text{Down}_2(\text{Conv}_{1 \times 1}(C_1)), \text{Conv}_{1 \times 1}(C_2), \text{Up}_2(\text{Conv}_{1 \times 1}(C_3))) \quad (1)$$

$$f_E = (\text{Conv}_{3 \times 3}(\text{SE}(C_{total}))) \quad (2)$$

$$E_{pred} = \text{Up}_8(\text{Sigmoid} \sum_{k=1} \text{Conv}_{1 \times 1}((C_{total}[k]))) \quad (3)$$

Where C_1 , C_2 , C_3 represent input features, $\text{Conv}_{1 \times 1}$ denotes a 1×1 convolution, Up_2 and Down_2 represent nearest-neighbor interpolation down-sampling and up-sampling by a factor of two, respectively, and Cat denotes channel concatenation. After these operations, the difference between E_{pred} , calculated using a loss function, and the ground truth edge labels is used to supervise the network for better learning of f_E .

3.2 EFM

To integrate the edge information extracted by the EEM module into the original feature maps to enhance edge information and reduce semantic confusion, we propose the EFM (Edge Fusion Module) module as shown in the Fig. 3. The EFM first applies weighting to the feature maps based on the extracted boundary weight information while preserving the diversity of residual connections to mitigate the impact of erroneous boundary information on the original feature maps. After an initial fusion, to emphasize the spatial learning of the feature maps, the fused features are passed through a spatial attention (SA [25]) to generate a spatial weight map. The original features and the preliminary fused features are then weighted separately and combined through simple addition to obtain the final fused features. As for the feature maps from other layers, which have different sizes compared to the extracted boundary features, the boundary features are upsampled, downsampled, and convolved with a 1×1 kernel before being fused with the other feature maps. The aforementioned process is represented as follows.

$$f_i = (C_i \otimes f_E) + C_i \quad (4)$$

$$C_{Ei} = SA(f_i) \otimes (C_i + f_i) \quad (5)$$

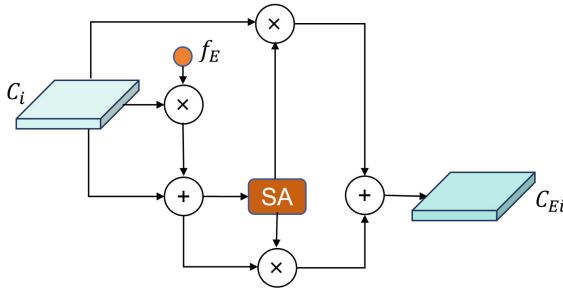


Fig. 3. Illustration of EFM. The edge guide map is fused with the original feature map.

3.3 CGA and CGA-Based Fusion Module

The Convolutional Block Attention Module (CBAM) [17] utilizes separate mechanisms for channel attention and spatial attention, treating these two forms of attention independently. This separation can limit the potential for integrated feature enhancement. To address this limitation, we propose a Content-Guided Attention (CGA) mechanism. The CGA mechanism creates specialized spatial importance maps for each channel using a progressive refinement approach. This method achieves a thorough integration of channel and spatial attention

and enhances the effective exchange of information between these two attention types.

The detailed steps of the Content-Guided Attention (CGA) are illustrated in the Fig. 4. For input features, we apply two branches to extract the weights for channel attention and spatial attention, denoted as W_c and W_s respectively. Specifically, in the channel branch, global average pooling is initially employed to generate the channel attention map. The channel dimension is then reduced through a 1×1 convolution, processed with a ReLU activation function, and restored to the original channel dimension with another 1×1 convolution. In the spatial branch, max pooling and average pooling are first applied to generate feature maps, which are then concatenated along the channel dimension to form the spatial attention map. The spatial weight distribution is then acquired through a 7×7 convolution. Next, a simple addition operation is used to fuse W_c and W_s , adhering to the broadcasting rule to produce the coarse spatial importance map. To obtain the final refined Spatial Importance Maps, each channel is adjusted based on the relevant input features. Specifically, the content of the input features guides the generation of the Spatial Importance Maps. Channel shuffle operations are used to alternately rearrange every channel. Here, σ represents the Sigmoid operation, $CS(\cdot)$ represents is the operation for channel shuffling, and $GC_{7 \times 7}(\cdot)$ indicates the group convolution. This approach emphasizes the more useful information encoded within the features. The aforementioned process is represented as follows.

$$X = C_{1 \times 1}([F_{low}, F_{high}]) \quad (6)$$

$$W_c = C_{1 \times 1}(\max(0, C_{1 \times 1}(X_{GAP}^c))) \quad (7)$$

$$W_s = C_{7 \times 7}([X_{GAP}^s, X_{GMP}^s]) \quad (8)$$

$$W_{coa} = W_c + W_s \quad (9)$$

$$W = \sigma(GC_{7 \times 7}(CS([X, W_{coa}]))) \quad (10)$$

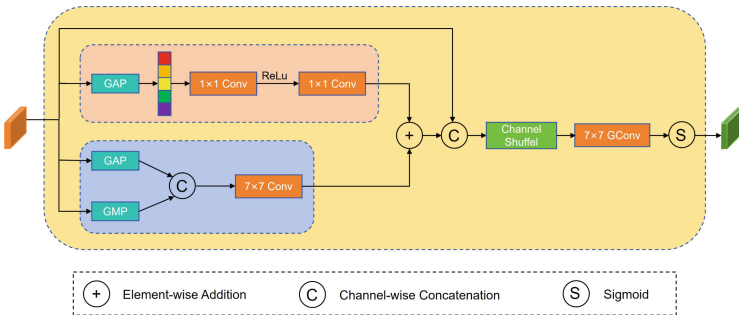


Fig. 4. Illustration of CGA. The CGA assigns unique spatial importance maps to every channel.

In the decoder, each pixel in the deep features corresponds to a pixel region derived from the shallow features. Basic operations such as addition, concatenation, or other combinations are inadequate for resolving the inconsistencies before fusion. To address this issue, we introduce a CGA-based fusion module, as illustrated in Fig. 5. This module fuses and adjusts the low-level features from the encoder with their matching high-level counterparts through the use of learned spatial weights, enabling an adaptive fusion mechanism. The core method utilizes CGA to compute the spatial weights necessary for feature modulation. The low-level and corresponding high-level features from the encoder are processed by CGA to generate these weights, which are then combined through weighted summation. Additionally, skip connections are introduced to incorporate input features. Finally, the fused features are passed through a 3×3 convolutional layer to generate the output features.

$$F_{fuse} = C_{3 \times 3} (F_{low} \cdot W + F_{high} \cdot (1 - W) + X) \tag{11}$$

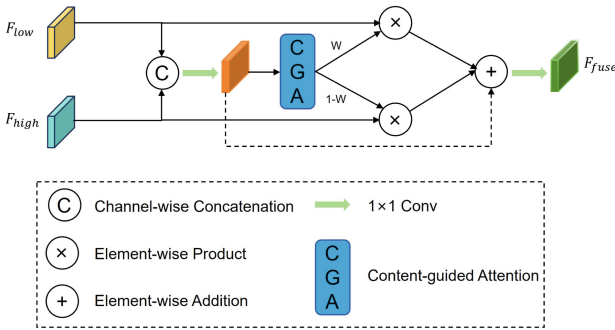


Fig. 5. Illustration of CGA based Fusion Module. Used to fully integrate features at different levels.

3.4 Loss Function

We define the loss function L_{seg} as cross-entropy loss (CE). The formulation is as follows:

$$L_{ce} = -\frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K y_k^{(n)} \log \hat{y}_k^{(n)} \tag{12}$$

In edge-supervised learning, we utilize a combination of binary cross-entropy (BCE) loss and binary Dice loss, formulated as follows:

$$L_{edg} = L_{bce} + L_{dice} \tag{13}$$

Therefore, our total loss is defined as follows:

$$L_{total} = L_{seg} + \lambda \cdot L_{edg} \tag{14}$$

where λ represents the weight, which was set to 0.2 in our experiment.

4 Datasets and Experimental Settings

4.1 Datasets

ISPRS Vaihingen. This dataset comprises 33 orthorectified images with an average size of 2494×2064 . Each image consists of three bands: near-infrared, red, and green, along with the corresponding Digital Surface Model (DSM) and Normalized DSM (NDSM). It consists of six classes: impervious surfaces, buildings, low vegetation, trees, cars, and background.

ISPRS Potsdam. This dataset consists of 38 orthorectified images with an image size of 6000×6000 and a Ground Sampling Distance (GSD) of 5 cm. Each image contains near-infrared, red, green, and blue bands, as well as the corresponding DSM and NDSM. Similar to the Vaihingen dataset, it includes six classes: impervious surfaces, buildings, low vegetation, trees, cars, and background.

4.2 Evaluation Metrics

To evaluate the performance of our proposed model, we utilized three evaluation metrics: Overall Accuracy (OA), mean Intersection over Union (mIoU), and mean F1 score (mF1). These metrics were compared against state-of-the-art methods. OA, mIoU, and mF1 were calculated based on the cumulative confusion matrix as follows:

$$\text{OA} = \frac{\sum_{k=1}^N \text{TP}_k}{\sum_{k=1}^N \text{TP}_k + \text{FP}_k + \text{TN}_k + \text{FN}_k} \quad (15)$$

$$\text{mIoU} = \frac{1}{N} \sum_{k=1}^N \frac{\text{TP}_k}{\text{TP}_k + \text{FP}_k + \text{FN}_k} \quad (16)$$

$$\text{precision} = \frac{1}{N} \sum_{k=1}^N \frac{\text{TP}_k}{\text{TP}_k + \text{FP}_k} \quad (17)$$

$$\text{recall} = \frac{1}{N} \sum_{k=1}^N \frac{\text{TP}_k}{\text{TP}_k + \text{FN}_k} \quad (18)$$

$$\text{F1} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \quad (19)$$

Where TP_k , FP_k , TN_k and FN_k represent true positives, false positives, true negatives, and false negatives, respectively, for objects with an index of class k . OA is the ratio of correctly predicted pixels to the total number of pixels.

4.3 Experimental Setup

We followed the official recommendations to partition the datasets. For the ISPRS Vaihingen dataset, we used 16 images for training and 17 images for testing. For the ISPRS Potsdam dataset, 24 images were used for training and 14 images for testing. To prevent overfitting, we employed a sliding window operation to crop the training data into 512×512 patches with a stride of 256. Data augmentation was applied using random horizontal flips, random vertical flips, and random multi-scale $[0.5\times, 0.75\times, 1\times, 1.25\times, 1.5\times]$ cropping. The experiments on the Vaihingen dataset were conducted on a single NVIDIA 3060 GPU using the PyTorch framework, while the experiments on the Potsdam dataset employed the same data cropping and augmentation methods on a single NVIDIA 3090 GPU with the PyTorch framework. During the training phase, we utilized the AdamW optimizer with a weight decay parameter of 0.01 and an initial learning rate of $6e-5$. The learning rate was updated using the “poly” learning policy with a power of 0.9. The batch size was set to 4, and the maximum training iteration was 105. During the testing phase, we employed a test time augmentation (TTA) strategy using multiple scales $[0.5\times, 0.75\times, 1\times, 1.25\times, 1.5\times]$.

5 Experimental Results and Analysis

5.1 Comparison With State-of-the-Art Methods on ISPRS Vaihingen

The experimental results of different methods on the ISPRS Vaihingen dataset are presented in Table 1. Since the background occupies a relatively small pro-

Table 1. Quantitative comparison with the latest models on the ISPRS Vaihingen dataset. The best values in each column are indicated in bold. All scores are reported as percentages (%), measured in F1 scores for all categories.

Method	Imp.surf	Building	Lowveg.	Tree	Car	MeanF1	mIoU	OA
UNet [12]	89.86	93.41	80.82	86.93	81.50	86.51	76.55	87.79
SegNet [1]	89.35	93.14	80.57	86.75	74.37	84.84	74.23	87.41
DeepLabv3+ [3]	90.40	94.06	81.45	87.20	81.23	86.87	77.13	88.36
PSPNet [23]	89.99	93.78	81.51	87.41	79.28	86.39	76.44	88.05
MAResU-Net [6]	90.43	94.13	81.73	87.17	80.18	86.73	76.95	88.39
Swin-UperNet [10]	90.26	94.07	81.13	87.04	81.58	86.82	77.05	88.21
BANet [16]	90.28	93.86	80.99	87.08	80.31	86.50	76.60	88.12
ABCNet [8]	89.73	93.38	80.91	87.23	78.72	85.99	75.83	87.86
CMTFNet [18]	90.61	94.21	81.93	87.56	82.77	87.42	77.95	88.71
MSGCNet [21]	90.75	94.43	81.48	87.13	83.19	87.4	77.93	88.58
DAF-Net(ours)	90.69	94.38	82.1	88.64	83.77	87.92	78.73	89.07

portion of the pixels, the accuracy of the background was not included in the experiments. The results in Table 1 demonstrate that our proposed DAF-Net method achieved the highest MeanF1/mIoU/OA scores.

5.2 Comparison With State-of-the-Art Methods on ISPRS Potsdam

The experimental results of different methods on the ISPRS Potsdam dataset are presented in Table 2. The results in Table 2 demonstrate that our proposed EGMF-Net method achieved the highest MeanF1/mIoU/OA scores.

Table 2. Quantitative comparison with the latest models on the ISPRS Potsdam dataset. The best values in each column are indicated in bold. All scores are reported as percentages (%), measured in F1 scores for all categories.

Method	Imp.surf	Building	Lowveg.	Tree	Car	MeanF1	mIoU	OA
UNet [12]	90.73	95.35	85.05	85.97	91.49	89.72	81.57	88.36
SegNet [1]	91.27	95.18	85.10	86.05	91.10	89.74	81.60	88.54
DeepLabv3+ [3]	91.76	96.33	85.74	86.87	92.23	90.59	83.02	89.45
PSPNet [23]	91.74	96.32	85.80	86.97	91.86	90.54	82.94	89.42
MAResU-Net [6]	91.79	96.33	85.69	87.03	92.19	90.61	83.05	89.46
Swin-UperNet [10]	91.60	96.04	86.09	87.00	91.70	90.49	82.82	89.43
BANet [16]	91.42	95.65	85.67	86.88	91.40	90.20	82.35	89.14
ABCNet [8]	91.21	95.92	85.28	86.56	90.74	89.94	81.94	88.94
CMTFNet [18]	92.12	96.41	86.43	87.26	92.41	90.93	83.57	89.89
MSGCNet [21]	92.2	96.62	86.34	87.15	92.51	90.91	83.49	89.87
DAF-Net(ours)	92.27	96.57	86.00	87.52	92.93	91.06	83.81	89.97

To visualize the differences between our method and other popular approaches, we display the visual results of several methods in the Fig. 6. It can be observed that DAF-Net achieves better segmentation results compared to other methods. In environments with complex edges, DAF-Net can accurately identify the complete contours of objects. For the small object category, such as “car,” DAF-Net can segment each car more accurately than other methods. It is evident that DAF-Net performs well in segmenting objects of different scales.

5.3 Ablation Experiment

To evaluate the performance of the modules included in our proposed DAF-Net, ablation experiments were conducted on the ISPRS Vaihingen dataset. Due to the interdependence of the edge extraction module and the edge fusion module, separate experiments were not performed on these two modules. The experimental results are shown in Table 3, as follows: Baseline: The baseline consists of the swin-transformer-Base as the backbone network, with the seghead [13] serving as the segmentation head.

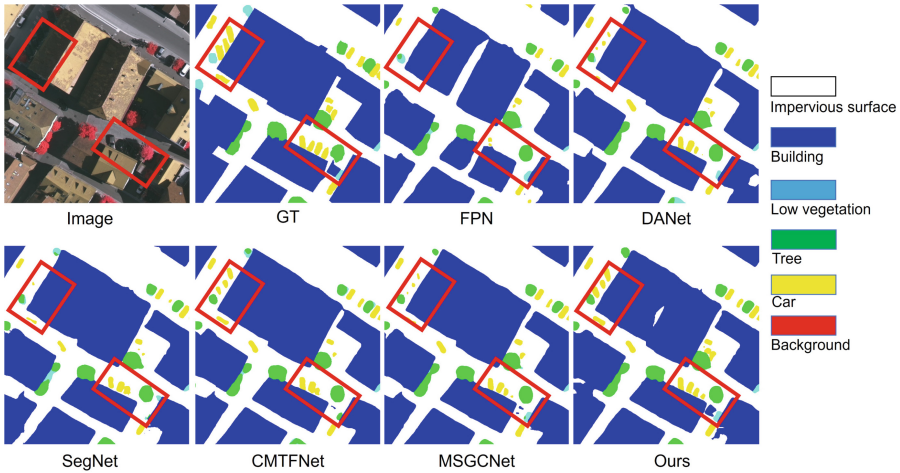


Fig. 6. Qualitative comparison of the visualization results of our method with other methods on the Vaihingen dataset.

Baseline+EEM+EFM. utilizes shallow-level features for edge feature extraction and integrates edge information through the designed module. As shown in Table 3, by incorporating EEM+EFM, the segmentation performance for all categories is improved compared to the baseline. Particularly, there is a significant improvement in the segmentation accuracy of small objects, such as cars. MeanF1 is enhanced by 1.74%, resulting in an overall improvement of 0.68% in MeanF1, 1.05% in mIoU, and 0.5% in OA.

Baseline+CGAF. CGAF mitigates the issue of detail loss during upsampling by incorporating a carefully designed content-guided attention fusion module. Additionally, it helps to bridge the semantic gap present in skip connections. As shown in Table 3, by incorporating CGAF, the segmentation performance for all categories is improved compared to the baseline, resulting in an overall improvement of 0.74% in MeanF1, 1.12% in mIoU, and 0.55% in OA.

Table 3. Performance Analysis of DAF-Net Modules. The best values in each column are highlighted in bold. All scores are presented as percentages (%), with F1 scores used for all categories.

Method	Imp.surf	Building	Lowveg.	Tree	Car	MeanF1	mIoU	OA
Baseline	90.26	94.07	81.13	87.04	81.58	86.82	77.05	88.21
Baseline+EEM+EFM	90.96	94.43	81.6	87.19	83.32	87.5	78.1	88.71
Baseline+CGAF	90.77	94.43	82.1	88.26	83.23	87.56	78.17	88.76
Baseline+total	91.31	94.87	82.38	87.29	84.04	87.98	78.84	89.20

Effect of CGA. To verify the effectiveness of our proposed CGA, we conducted comparative experiments. In order to ensure the uniqueness of the variables, we only replace the CGA module in the CGAF with the CBAM to verify its validity. As shown in Table 4, the results show that CGAF achieves better results, which proves that CGAF can better guide feature fusion.

Table 4. Verify the effectiveness of content-guided attention module. The best values in each column are highlighted in bold.

Method	Imp.surf	Building	Lowveg.	Tree	Car	MeanF1	mIoU	OA
Baseline	90.26	94.07	81.13	87.04	81.58	86.82	77.05	88.21
Baseline+CBAMF	90.95	94.55	81.53	87.05	82.84	87.38	77.93	88.66
Baseline+CGAF	90.77	94.43	82.1	88.26	83.23	87.56	78.17	88.76

6 Conclusion

The aim of this paper is to improve segmentation results by fusing edge information as accurately as possible, especially position pixels with similar edge colors. The features of high and low levels are gradually upsampled by a Content-Guided Attention fusion module containing jump connections. We have demonstrated through experiments the superiority of the proposed network architecture and the effectiveness of each module. However, using Swin Transformer as the encoder has the drawbacks of high computational complexity and large memory consumption. In future research, we will further optimize our network architecture, focusing on model compression and addressing the issue of high computational complexity.

Acknowledgements. This work was supported by the Chongqing social science planning project(Grant No. 2023BS085).

References

1. Badrinarayanan, V., Kendall, A., Cipolla, R.: SegNet: a deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **39**(12), 2481–2495 (2017)
2. Cao, H., et al.: Swin-Unet: Unet-like pure transformer for medical image segmentation. In: Karlinsky, L., Michaeli, T., Nishino, K. (eds.) *ECCV 2022*. LNCS, vol. 13803, pp. 205–218. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-25066-8_9
3. Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) *ECCV 2018*. LNCS, vol. 11211, pp. 833–851. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01234-2_49

4. Dosovitskiy, A., et al.: An image is worth 16x16 words: transformers for image recognition at scale. arXiv preprint [arXiv:2010.11929](https://arxiv.org/abs/2010.11929) (2020)
5. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7132–7141 (2018)
6. Li, R., Zheng, S., Duan, C., Su, J., Zhang, C.: Multistage attention ResU-Net for semantic segmentation of fine-resolution remote sensing images. *IEEE Geosci. Remote Sens. Lett.* **19**, 1–5 (2021)
7. Li, R., et al.: Multiattention network for semantic segmentation of fine-resolution remote sensing images. *IEEE Trans. Geosci. Remote Sens.* **60**, 1–13 (2021)
8. Li, R., Zheng, S., Zhang, C., Duan, C., Wang, L., Atkinson, P.M.: ABCNet: attentive bilateral contextual network for efficient semantic segmentation of fine-resolution remotely sensed imagery. *ISPRS J. Photogramm. Remote. Sens.* **181**, 84–98 (2021)
9. Liu, R., Mi, L., Chen, Z.: AFNet: adaptive fusion network for remote sensing image semantic segmentation. *IEEE Trans. Geosci. Remote Sens.* **59**(9), 7871–7886 (2020)
10. Liu, Z., et al.: Swin transformer: hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 10012–10022 (2021)
11. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431–3440 (2015)
12. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
13. Sun, K., et al.: High-resolution representations for labeling pixels and regions. arXiv preprint [arXiv:1904.04514](https://arxiv.org/abs/1904.04514) (2019)
14. Tan, M., Pang, R., Le, Q.V.: EfficientDet: scalable and efficient object detection (2020). 1911.09070
15. Wang, L., Li, R., Duan, C., Zhang, C., Meng, X., Fang, S.: A novel transformer based semantic segmentation scheme for fine-resolution remote sensing images. *IEEE Geosci. Remote Sens. Lett.* **19**, 1–5 (2022)
16. Wang, L., Li, R., Wang, D., Duan, C., Wang, T., Meng, X.: Transformer meets convolution: a bilateral awareness network for semantic segmentation of very fine resolution urban scene images. *Rem. Sens.* **13**(16), 3065 (2021)
17. Woo, S., Park, J., Lee, J.-Y., Kweon, I.S.: CBAM: convolutional block attention module. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11211, pp. 3–19. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01234-2_1
18. Wu, H., Huang, P., Zhang, M., Tang, W., Yu, X.: CMTFNet: CNN and multiscale transformer fusion network for remote sensing image semantic segmentation. *IEEE Trans. Geosci. Rem. Sens.* **61** (2023)
19. Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J.M., Luo, P.: SegFormer: simple and efficient design for semantic segmentation with transformers. *Adv. Neural. Inf. Process. Syst.* **34**, 12077–12090 (2021)
20. Xu, Z., Zhang, W., Zhang, T., Yang, Z., Li, J.: Efficient transformer for remote sensing image segmentation. *Rem. Sens.* **13**(18), 3585 (2021)
21. Zeng, Q., Zhou, J., Tao, J., Chen, L., Niu, X., Zhang, Y.: Multiscale global context network for semantic segmentation of high-resolution remote sensing images. *IEEE Trans. Geosci. Remote Sens.* **62**, 1–13 (2024)

22. Zhang, C., Jiang, W., Zhang, Y., Wang, W., Zhao, Q., Wang, C.: Transformer and CNN hybrid deep neural network for semantic segmentation of very-high-resolution remote sensing imagery. *IEEE Trans. Geosci. Remote Sens.* **60**, 1–20 (2022)
23. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2881–2890 (2017)
24. Zheng, S., et al.: Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6881–6890 (2021)
25. Zhu, X., Cheng, D., Zhang, Z., Lin, S., Dai, J.: An empirical study of spatial attention mechanisms in deep networks. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6688–6697 (2019)



Distortion Correction Sub-network for Semantic Segmentation Based on Deep Hough Transform

Wanpeng Geng, Jing Liu, Dexin Zhang, and Hui Zhang(✉)

Tianjin University of Science and Technology, Tianjin, China
zhanghui2022@tust.edu.cn

Abstract. This study proposes a novel method for semantic line correction utilizing deep Hough transform, aimed at tackling the challenges associated with detecting semantic straight lines and correcting image distortions in natural scenes. Traditional approaches frequently consider semantic straight line detection as a subset of object detection or simply adapt conventional object detection techniques, thereby neglecting the inherent characteristics of straight lines and consequently leading to suboptimal performance. Herein, we employ a deep Hough transform-based algorithm to achieve semantic line detection in images. The adopted approach utilizes parameterization and the Hough transform to map depth representations into parameter space for straight line detection, effectively exploiting the geometric properties of lines. Innovatively, we introduce the Distortion Correction Sub-network (DTN) to mitigate image distortion and enhance the success rate of deep Hough transform line detection. Furthermore, the DTN can dynamically adjust its spatial transformation according to various image transformations, thereby achieving effective image distortion correction. Experimental results demonstrate that the proposed method outperforms previous state-of-the-art methods on both self-constructed and publicly available datasets, thus substantiating its efficacy and superiority in addressing the challenges of semantic line detection and image distortion correction.

Keywords: Distortion correction, Hough transform, Semantic line detection, Deep Learning

1 Introduction

The detection of line structures in digital images has long been an enduring challenge in the realm of computer vision. The organization of line structures

W. Geng and J. Liu—These authors contributed equally to this work
This work is partially supported by National Natural Science Foundation of China (62076232, 62106015) and Beijing Nova Program (Z211100002121113).

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-78113-1_14.

represents a crucial stage in converting visual signals into intermediary concepts for visual interpretation. However, in real-world natural scenes, the detection of line structures becomes increasingly intricate and challenging owing to the presence of image distortions and deformations. These distortions may stem from various factors including camera lens shapes, shooting angles, and lens distortions, among others.

Presently, a plethora of techniques have been proposed for detecting salient objects [1] [2] and regions [3] [4] in digital images; however, scant attention has been devoted to detecting prominent line structures that unveil the image’s structure [5] [6] [7]. Traditional methods for line detection typically employ techniques such as the Hough transform [8] [9]; nevertheless, they frequently disregard the semantic information embedded within line structures, thereby demonstrating diminished robustness against distortions and deformations in images.

In recent years, deep learning has made significant strides in the field of computer vision, offering fresh insights and solutions for detecting semantic segmentation line structures. Nevertheless, existing deep learning methods still encounter challenges when confronted with image distortions [10]. Owing to the intricacy of distortions, traditional methods for line detection [11] frequently falter in precisely identifying and rectifying line structures in images, thereby yielding suboptimal detection results.

Consequently, this paper introduces an innovative correction approach for semantic segmentation lines leveraging the deep Hough transform, with the explicit objective of mitigating the effects of image distortion on semantic segmentation lines. This method integrates deep learning with classical Hough transform [12], thereby achieving effective detection and distortion correction of semantic line structures in images. In comparison to traditional methods, our approach not only facilitates more precise identification of line structures but also adeptly manages distortions and deformations in images, consequently enhancing the performance and robustness of line detection.

The main contributions of this paper include:

1. Based on the combination of deep learning and Hough transform, we propose a novel line detection method for the correction deep Hough transform model, which realizes the efficient detection and distortion correction of the semantic line structure in the image.
2. A spatial transformation network is devised to dynamically alter images based on diverse transformations, serving not only for rectifying image distortions but also for facilitating other tasks such as classification, detection, and segmentation.
3. Extensive experiments on public datasets validate the effectiveness and superiority of our method, demonstrating its capability in handling image distortion and warping.
4. The DTN exhibits versatility by not only seamlessly integrating with the deep Hough transform, but also providing a convenient integration pathway into the network architectures of diverse tasks including classification, detection, and segmentation.

2 Related Work

Within the realm of computer vision, the detection of image lines and the correction of distortion stand as two pivotal quandaries pivotal to the spectrum of image processing and analysis. This section will delve into an array of methodologies and techniques concerning Hough transform, CNN-based line detection, semantic line detection, edge activation function and distortion correction.

2.1 Hough Transform

The Hough transform, a classical image processing technique [13], is devised for detecting geometric shapes within images, with a primary focus on lines. Originally proposed by Hough in 1962, this technique serves as a foundation for line detection methods. Its fundamental principle involves mapping image pixels to curves in parameter space, thereby transforming the task into one of curve detection. Through voting within parameter space, it identifies parameters corresponding to collinear pixels in the image, typically expressed as slope and intercept. Traditional approaches utilize accumulator arrays for parameter space representation, extracting lines in the image via thresholding or non-maximum suppression techniques.

While the Hough transform demonstrates efficacy in tasks such as line detection, it exhibits limitations in mitigating image distortion and deformation. Owing to factors like lighting, noise, and geometric distortion affecting line appearances in images, conventional Hough transform methods frequently encounter challenges in accurately detecting and describing line structures. Consequently, additional enhancements and refinements of the Hough transform method are necessary to tackle issues related to semantic line detection and distortion correction in images [14], thereby augmenting its efficacy and resilience.

2.2 Line Detection Based on CNN

Convolutional Neural Networks (CNNs) [15] are a leading technique in image processing, widely used for tasks like line detection. CNNs incrementally extract features from images through convolutional and pooling layers, followed by classification or regression in fully connected layers.

Numerous studies explore CNNs in online detection. LeNet, introduced by LeCun et al. [16], was one of the earliest successes in CNN-based image processing, excelling in handwritten digit recognition. AlexNet, by Alex Krizhevsky et al. [17], further advanced CNNs by achieving significant success in the 2012 ImageNet competition, using deeper networks and larger datasets.

For online detection tasks, several refined CNN architectures have emerged. HoughNet, proposed by Samet N et al. [18], combines the traditional Hough transform with deep learning to effectively detect lines and curves. Additionally, semantic segmentation networks like U-Net are extensively used in line detection, offering precise and robust position and shape information.

2.3 Semantic Line Detection

Semantic line detection constitutes a pivotal subfield within line detection, directing attention not solely to the geometric configurations of lines but also to the comprehension of the semantic information they encapsulate. Among these advancements, in 2015, Long et al. [19] introduced the concept of Fully Convolutional Networks (FCN) and pioneered its application to semantic line detection, culminating in end-to-end pixel-level prediction. FCN transforms the traditional convolutional neural network structure into a fully convolutional form, empowering it to concurrently generate semantic labels for every pixel within the image. Consequently, it introduces a pioneering approach and methodology for semantic line detection.

Apart from FCN, several enhanced neural network architectures have emerged. For instance, the DeepLab series of networks, introduced by Chen et al. [20] in 2016, attained heightened precision in semantic segmentation outcomes by integrating dilated convolutions and multi-scale feature fusion mechanisms, heralding novel breakthroughs in the realm of semantic line detection tasks. Moreover, Mask R-CNN, devised by He et al. [21] in 2017, amalgamates principles from both object detection and semantic segmentation, facilitating simultaneous object detection and semantic line generation in images, thereby furnishing a more holistic solution.

2.4 Edge Activation Function

Edge activation functions, such as ReLU and its adaptations like Leaky ReLU, Parametric ReLU, and ELU, are crucial in neural networks for their ability to tackle gradient vanishing or exploding. They exhibit significant gradients near specific values, enhancing training speed and efficacy. These functions, characterized by their ease of implementation and practical performance, play a pivotal role in improving neural network training.

ReLU, one of the most elementary edge activation functions, was initially introduced by Hahnloser et al. [22] in 2000 and has undergone continual refinement and expansion in subsequent studies. Conversely, Leaky ReLU was proposed by Maas et al. [23] in 2013, offering the benefit of mitigating certain challenges associated with ReLU, including the issue of “neuron death”.

2.5 Distortion Correction

Within the realm of distortion correction, substantial research endeavors are immersed in deep learning methodologies. As an example, Noh et al. [24] introduced a convolutional neural network (CNN)-based end-to-end distortion correction method in 2018, referred to as the Deep Homography Estimation Network (Deep Homography). By learning homographic transformations between images, this method achieves precise rectification of image distortions, thus furnishing lucid inputs for subsequent image processing tasks.

Furthermore, the fully convolutional network (FCN) proposed by Kendall et al. [25] in 2017 has also found extensive application in tasks related to image distortion correction. FCN has the capability of mapping each pixel in an image to the output image, thus enabling end-to-end processing of images and providing an efficient solution for tasks related to distortion correction.

3 Method

This section offers an elaborate exposition on the deep Hough transform (DHT) and the Distortion Correction Sub-network (DTN). Our approach comprises the following key components: 1) Line parameterization using polar coordinates; 2) Mapping the image coordinate space to the parameter space of the Hough transform through the deep Hough transform; 3) Leveraging the parameter space to characterize linear features in the image and employing specific techniques for line detection; 4) Reverting the detected lines to the image space using the Hough transform; 5) Semantic secant line acquisition for deep Hough transform; 6) Integration of the DTN into the deep object detection network.

3.1 Parameterization and Reverse Parameterization

Within the framework of the deep Hough transform, the parameterization of a straight line can be accomplished utilizing polar coordinates, $r = x\cos\theta + y\sin\theta = x$. In a two-dimensional Cartesian coordinate system, a straight line can be delineated through two parameters: A directional parameter, denoted as $\theta \in [0, \pi)$, represents the angle between r and the x -axis, and a distance parameter, denoted as r , signifies the distance from line segment l to the origin, as shown in Fig. 1.

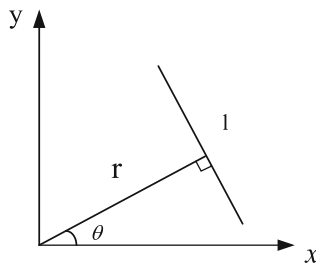


Fig. 1. Illustrates the coordinate parameter diagram.

3.2 Deep Hough Transform

The deep Hough transform (DHT) constitutes an image processing methodology that meticulously maps the image coordinate space onto the Hough parameter space. It adeptly employs a voting mechanism to meticulously estimate the parameters of the desired detection objects, finding widespread application in the realm of line detection.

In the Cartesian coordinate system, lines are commonly expressed in the format of $y = kx + b$. However, in cases where the line is perpendicular to the x-axis, the slope k tends towards infinity, posing challenges for subsequent processing and computation tasks. Therefore, we first convert the Cartesian coordinate system to the polar coordinate system, where the polar expression of the line is $\rho = x \cos \theta + y \sin \theta$. Subsequently, the two parameters ρ and θ serve as axes to construct the Hough space. Given the sum of parameters ρ and θ for a line, it corresponds to a point within the Hough space. With multiple points existing in the polar coordinate system, each point may correspond to numerous lines, each characterized by a distinct pair of parameters ρ and θ , thereby forming curves within the Hough space, as illustrated in Fig. 2. When aiming to determine a line that encompasses as many points as feasible from multiple points within the polar coordinate system, one can identify the maximum intersections of curves within the Hough space, depicted in Fig. 3.

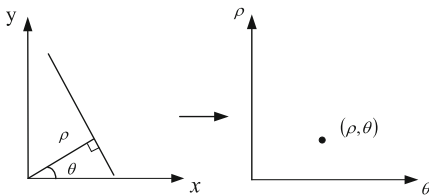


Fig. 2. Illustrates how a line in polar coordinates corresponds to a point in the Hough space.

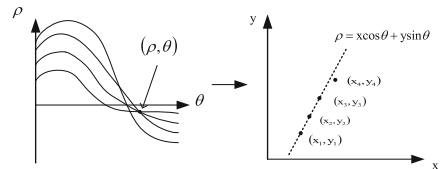


Fig. 3. Shows how a point in the Hough space corresponds to a line in polar coordinates.

3.3 Line Detection in Parameter Space

Line detection in parameter space entails the description of straight-line features within images by employing parameter space and detecting these lines through specific methodologies. Initially, through the utilization of DHT, the features within the image undergo transformation into parameter space, wherein each grid position corresponds to the parameters of a straight line l depicted in the image. The objective of feature transformation into parameter space is to depict the geometric attributes of lines more efficiently, thus rendering the representation of lines in parameter space more succinct.

Features within parameter space are aggregated utilizing convolution operations and other methodologies to facilitate the detection of straight-line features within images. At different stages of the Feature Pyramid Network (FPN) [26], convolution layers are employed to gather contextual line features, subsequently followed by interpolation operations to align the resolution of features across distinct stages. Ultimately, the interpolated features are concatenated to generate predictions for lines. For model training, it is imperative to transform the ground truth straight lines into parameter space and represent them as binary maps using specific methodologies. To expedite model convergence, the ground truth straight lines can be smoothed utilizing Gaussian kernels for both smoothing and expansion operations. Lastly, the cross-entropy loss between the smoothed ground truth straight lines and the model-predicted lines can be computed within parameter space to optimize the model parameters.

3.4 Reverse Mapping

The reverse Hough transform (RHT) denotes the procedure of converting the linear representation from parameter space back to the representation of straight lines in image space. In our line detector, initially, a prediction map is generated using parameter space, which represents the likelihood of the presence of lines. Subsequently, the positions where lines exist are determined by applying thresholding and binarization to the prediction map. Afterwards, the centroids of each connected region are computed, serving as the parameters for the detected lines. Next, utilizing the reverse Hough transform, these parameters are remapped to revert to the representation of straight lines in the image space. This process, termed as the “reverse mapping of the Hough transform” in the field of image processing, is aimed at remapping the linear parameters from parameter space back to the original image space to achieve precise localization and description of the detected lines.

3.5 Semantic Dividing Line Acquisition of the Deep Hough Transformation

In the Fig. 4, DHT denotes the deep Hough transform, responsible for converting the Cartesian coordinates of the image into polar coordinates; RHT, conversely, undertakes the reverse transformation, converting from polar coordinates back to Cartesian coordinates of the image, with RHT yielding the actual results of semantic segmentation line detection. The DHT module employed in this study is inspired by the deep Hough transform module outlined in reference [27]. The disparity emerges in the coordinate transformation process, wherein the image center is adopted as the origin, the polar angle range spans from 0 to 2π , and a consistent sampling rate is applied. The maximum polar radius fluctuates across diverse scales of feature layers in the image feature pyramid; consequently, a diverse set of polar radii is devised $[\nabla r_1, \nabla r_2, \dots, \nabla r_n]$ to align various feature layers with polar coordinate space layers of uniform dimensions, thereby expediting the subsequent convolutional computations.

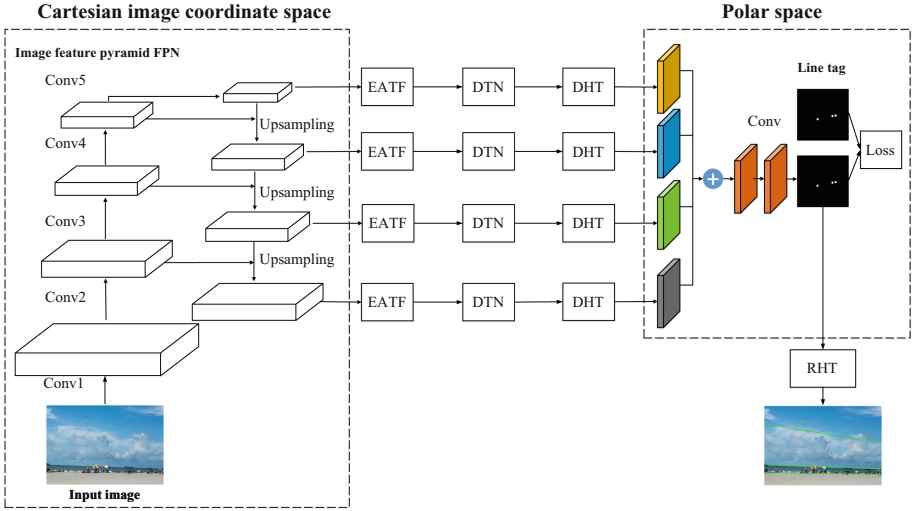


Fig. 4. Deep Hough transform agent semantic line detection network.

The abbreviation EATF denotes the Edge Activation Function Network. Traditional line detection methods relying on binary edge images as input, obtained from edge detection results based on the Hough transform, often encounter challenges in achieving optimal performance for complex real-world scene images. Nonetheless, the primary distribution of semantic segmentation lines tends to occur at the boundaries delineating different regions, where the edge information of the image assumes a pivotal role in semantic line segmentation tasks. To harness the significance of edge information in semantic line localization, we introduce the EATF. This network is designed to fulfill dual objectives: firstly, it incorporates a channel self-attention mechanism to adaptively weight convolutions, thus implicitly guiding the learning process towards convolutions beneficial for semantic segmentation lines, drawing inspiration from the self-attention mechanism employed in the SENet [28] model; secondly, it integrates the Tanh activation function to mitigate the influence of sharp edges, thereby diminishing the impact of short sharp lines, enabling the network to prioritize large-scale semantic segmentation lines with distinct semantics. The detailed architecture of EATF is illustrated in the subsequent Fig. 5. In contrast to a typical SENet that employs the Sigmoid function as the final activation function, we incorporate the Tanh activation function after the module. Consequently, we adopt the Relu function in the channel attention mechanism to mitigate the issue of gradient disappearance during network training, thereby expediting convergence.

Semantically distinct segmentation lines can be categorized into multiple classes, and the optimization target loss function, designated during training, is set to $L = L_{line} + L_{cls}$. Within polar coordinate space, a line is projected as a point. Throughout the training process, the annotation for line projection manifests as a point. Reference [27] introduces the process of smoothing a labeled

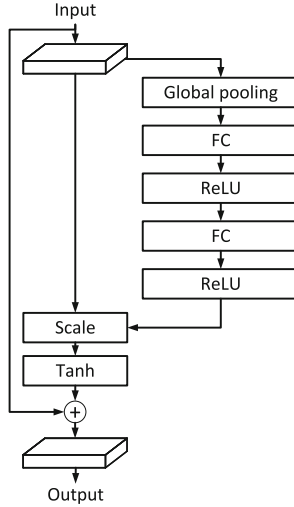


Fig. 5. Edge activation function module EATF.

point using a Gaussian kernel to acquire a distribution within a neighborhood range. L_{line} denotes the cross-entropy loss function between the predicted outcome and the smoothed annotation map. L_{cls} denotes the loss associated with the line category, wherein employing multi-class cross-entropy loss (softmax loss) proves to be adequate.

3.6 Distortion Correction Sub-network

Given that distortions frequently manifest in cameras, with variation across different models, rectifying distortions for surveillance cameras exhibiting minor aberrations is deemed unnecessary. Nonetheless, even subtle distortions possess the potential to induce image aberrations, thereby hindering the extraction of pivotal information crucial for object detection.

Despite lenses enhancing image fidelity, they represent the primary contributors to image distortion. The geometrical configuration of a lens influences light propagation, and misalignment with the imaging plane induces displacement of light positions. Radial distortion, stemming from the lens shape, exacerbates with distance from the image periphery, encompassing barrel and pincushion distortions, while tangential distortion arises from non-parallel imaging.

Radial distortion can be characterized through polynomial functions, with its expression contingent on the distance from the center. Polynomial functions of the second order or higher are applicable for rectifying coordinate alterations. On the normalized imaging plane, the coordinates $[x, y]$ of the uncorrected point are depicted in polar notation as $[r, \theta]$, with r signifying the point's distance from the origin, representing the angle relative to the horizontal axis, and $[x_{dis}, y_{dis}]$

delineating the coordinates where distortion manifests, as delineated below:

$$x_{dis} = x(1 + k_1r^2 + k_2r^4 + k_3r^6) \quad (1)$$

$$y_{dis} = y(1 + k_1r^2 + k_2r^4 + k_3r^6) \quad (2)$$

Typically, cameras can utilize these two parameters to effectively rectify radial distortion. However, in the case of cameras exhibiting significant distortion, such as those equipped with fisheye lenses, alternative methodologies become imperative for distortion rectification.

Tangential distortion is conventionally rectified employing the subsequent two formulas, wherein p_1 and p_2 denote the respective parameters:

$$x_{dis} = x + 2p_1xy + p_2(r^2 + 2x^2) \quad (3)$$

$$y_{dis} = y + p_1(r^2 + 2y^2) + 2p_2xy \quad (4)$$

Combining equations (1)(2) and (3)(4), we have:

$$x_{dis} = x(1 + k_1r^2 + k_2r^4 + k_3r^6) + 2p_1xy + p_2(r^2 + 2x^2) \quad (5)$$

$$y_{dis} = y(1 + k_1r^2 + k_2r^4 + k_3r^6) + p_1(r^2 + 2y^2) + 2p_2xy \quad (6)$$

Points on the normalized imaging plane are projected onto the pixel plane by the camera's intrinsic matrix, resulting in the derivation of actual pixel coordinates in the image, as depicted below:

$$u = f_x x_{dis} + c_x \quad (7)$$

$$v = f_y x_{dis} + c_y \quad (8)$$

Since the intrinsic matrices of different cameras vary, the parameters of f_x , f_y , c_x and c_y also differ.

In order to alleviate the influence of distortion on the determination of bounding boxes for object detection, we propose the incorporation of a Distortion Correction Subnetwork (DTN) into the architecture of the deep object detection network. When presented with a distorted image (depicted in pixel coordinates (u, v)), the process of acquiring an undistorted image (also depicted in pixel coordinates (u', v')) necessitates computations according to equations (5)(6) and (7)(8). The objective of the DTN is to acquire knowledge of the parameter matrix $\rho = (k_1, k_2, k_2, p_1, p_2, f_x, f_y, c_x, c_y)$, illustrated in Fig. 6. The DTN comprises multiple convolutional layers and fully connected layers. Representing the computation of distortion correction transformation, based on equations (5)(6) and (7)(8), at each pixel position of (u', v') the corrected output (V), the value corresponds to the pixel position of (u'', v'') the uncorrected image (U). Should the calculated result not yield an integer value, interpolation sampling based on the pixel values of U is necessary to derive the output V.

The Dynamic Transformation Network exhibits the capability to dynamically execute spatial transformations on images, accommodating various types of transformations. As a modular entity independent of the Neural Network

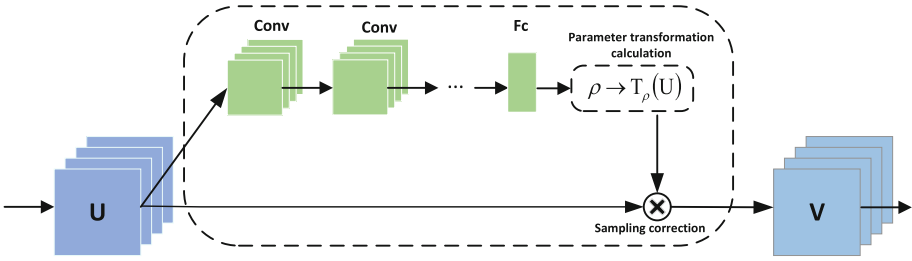


Fig. 6. Distortion Correction Sub-Network (DTN).

(NN), DTN has the flexibility to be integrated at any juncture within the NN architecture. DTN integration options span from pre-object detection stages to post-image processing or post-convolutional feature mapping stages. Moreover, this DTN module holds potential utility in diverse tasks such as classification and segmentation, culminating in comprehensive image distortion correction capabilities.

The DTN module embodies a sophisticated network architecture, comprising components such as a spatial transformer, a channel self-attention mechanism, matrix transformations, sampling correction, and the EATF module. Notably, the allocation of parameters primarily centers on the channel self-attention mechanism and the linear layers, commonly referred to as fully connected layers. Specifically, within the channel self-attention mechanism, the parameter volume is approximately one-sixteenth of the input channel count, while the parameter volume associated with convolutional operations and fully connected layers is contingent upon factors such as the input and output channel dimensions alongside the convolution kernel size. The computational burden primarily arises from convolutional operations and fully connected layers, the demands of which are intricately linked to the dimensions of the input feature maps and the parameter count. The DTN module is characterized by a substantial parameter count and computational demands; however, the incorporation of the channel self-attention mechanism and the EATF module serves to bolster the model’s expressiveness and performance.

DTN training intricately intertwines with the overarching network training process, with its primary optimization objective being the augmentation of image fidelity in object detection outcomes, without necessitating supplementary annotations for distorted images. As an integral component of the overall network architecture, DTN seamlessly integrates without impeding the end-to-end training protocol for object detection.

4 Experiments

This section comprehensively evaluates the potential of integrating deep Hough transform with DTN through an array of comparative experiments and ablation

studies, thereby showcasing the reliability and feasibility of our proposed model. A comparative analysis was conducted between the proposed method and both classical algorithms as well as state-of-the-art approaches.

4.1 Experimental Setup

Dataset: At present, our repository comprises solely of a solitary semantic line detection dataset denoted as SEL [29], encompassing 1,715 images, with 175 designated for testing purposes, and the remainder allocated for training. Recognizing the disparity between the capacities of CNN models and the incumbent dataset sizes, we unveil a novel semantic line detection dataset.

Termed NKL (abbreviated from NanKai Lines) [27], this fresh dataset encompasses 6,500 images portraying more intricate scenarios and a broader spectrum of lines. Within the NKL dataset, an impressive majority of images (67%, 4,356 out of 6,500) feature at least one semantic line, contrasting with a proportion of merely 45.5% observed in the SEL dataset. For an in-depth analysis of the diversity inherent in the SEL and NKL datasets, we subject all images to a ResNet50 network pre-trained on Place365, capturing the resultant outputs as classification labels.

The Place365 dataset comprises 365 categories in total, of which we acquired 167 categories in the SEL dataset and 327 categories in the NKL dataset. Moreover, the distribution of scene labels in the NKL dataset exhibits a more equitable distribution compared to that in the SEL dataset. For instance, within the SEL dataset, the top three primary categories (sky, wilderness, desert) collectively constitute over 25% of the dataset. Conversely, within the NKL dataset, the top three primary categories comprise less than 20% of the dataset.

Training: Our proposed methodology is realized within the PyTorch framework. The architecture of our neural network is trained from scratch, specifically tailored for line segmentation data, and does not rely on pre-trained weights. In this study, all experiments were conducted utilizing Python 3.8, PyTorch 1.10.0, CUDA 11.3, and Ubuntu 20.04. We set the number of training to 30 rounds, selected ResNet50 as the backbone network, with the initial learning rate set to $2e-4$ and the weight decay set to 0.9. To ensure equitable comparisons, we standardized the random seed across all experiments. The training utilized a batch size of 40 and was executed exclusively on the NVIDIA GeForce RTX 3090 GPU.

Evaluation Metrics: The quality of line detection is assessed through precision, recall, and F-measure. Initially, bipartite graph matching is employed to align predicted lines with ground truth lines. Subsequently, true positives (TP), false positives (FP), and false negatives (FN) can be computed post-matching. The matching process is carried out utilizing the Hungarian algorithm, known

for its polynomial time complexity. The matching outcomes determine the correspondence between each ground truth line and a predicted line. This approach effectively assesses the consistency between detection outcomes and ground truth.

The Precision (P), Recall (R), and F-measure (F) are defined as:

$$P = \frac{TP}{TP + FP}, R = \frac{TP}{TP + FN}, F = \frac{2PR}{P + R} \quad (9)$$

Through a series of meticulously conducted experiments, we acquire the average precision, recall, and F-measure metrics, and assess performance by meticulously comparing these three values.

4.2 Comparative Experiment

We use different methods for comparative experiments, including SLNet with different iterations, classical Hough transform, deep Hough transform and our proposed distortion correction deep Hough transform, which includes EATF modules and non-EATF modules (DHT+DTN+EATF and DHT+DTN).



Fig. 7. Examples of detection results of different methods on SEL dataset.

Both SLNet and HT methods necessitate the utilization of the HED edge detector for image preparation. In addition, SLNet incorporates non-maximum

suppression (NMS), and classical Hough transform similarly demands edge images for processing. Moreover, SLNet adopts iterative fine-tuning of the network to refine results, potentially affecting inference speed based on the number of iterations. Conversely, our approach mandates solely one forward pass to yield results, and NMS simplification facilitates computing the centroid of connected regions in parameter space. Figure 7 illustrates the comparative outcomes of diverse methodologies on the SEL dataset.

Table 1 encapsulates the test outcomes of multiple methodologies on the SEL dataset. Notably, the experimental outcomes of both HED+Hough and SLNet methodologies are drawn from the empirical data presented in the referenced paper [27], whereas the residual experiments are conducted employing ResNet50 as the fundamental network architecture. Our proposed approach demonstrates a noteworthy enhancement over the SLNet and Hough transformation results, attaining superior F-measure across various thresholds when compared to alternative methodologies. The comprehensive index containing the EATF module is the best, followed by the method without adding the EATF module.

Table 1. Depicts the detection outcomes of diverse methodologies on the SEL dataset.

Dataset	Method	Precision	Recall	F-measure
SEL	HED+Hough [27]	35.60	42.00	38.50
	SLNet [27]	76.20	72.90	74.50
	DHT	82.86	74.52	78.47
	DHT+DTN	77.76	80.37	79.04
	DHT+DTN+EATF	78.27	81.69	79.95

Table 2. Detection results of different methods on NKL dataset.

Dataset	Method	Precision	Recall	F-measure
NKL	HED+Hough [27]	21.30	62.20	31.80
	DHT	68.42	76.65	72.30
	DHT+DTN	70.16	80.12	74.81
	DHT+DTN+EATF	72.92	77.43	75.10

Table 2 delineates the test findings of various methodologies on the NKL dataset. Due to the unavailability of training code for SLNet, our comparison is limited to the results obtained from Hough transformation, DHT+DTN and DHT+DTN+EATF. Moreover, the experimental data pertaining to the HED+Hough method is cited from the empirical results presented in the referenced paper [27]. Figure 8 provides a visual representation of the detection outcomes achieved by DHT+DTN+EATF on the NKL dataset.

The experimental findings obtained from the SEL and NKL datasets signify a notable enhancement in the performance outcomes of DHT + DTN, but adding the EATF module works better. It effectively improves the indicators in the backbone network, further verifying the effectiveness of our proposed method.

In order to provide additional validation for the efficacy of our proposed methodology, we elected to utilize the NKL dataset as our sample corpus and implemented the radial distortion technique for image processing. Radial distortion, a frequently employed image manipulation technique, serves to simulate the effects of lens deformation on the captured images. Within the distortion model, pixels located in proximity to the image’s central axis undergo stretching or compression in comparison to those situated at the periphery, thereby

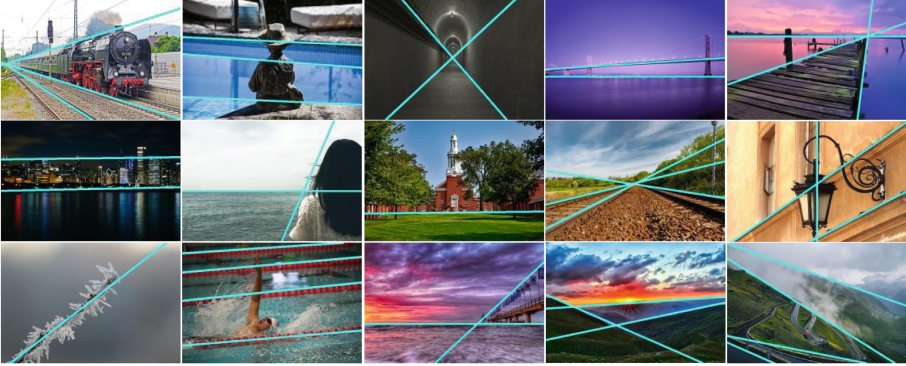


Fig. 8. Detection effect on NKL dataset.

inducing a perceptible curvature in straight lines within the image. Control over this distortion model is governed by the radial distortion parameters k_1 and k_2 , pivotal factors that dictate the magnitude of distortion exhibited.

Initially, we delineated two distinct sets of radial distortion parameters, denoted as k_1 and k_2 . Subsequently, we constructed an array for distortion mapping to determine the new position of every pixel through the application of these radial distortion parameters. Following that, we calculated the distortion factor for each pixel position based on the formula derived from the radial distortion model. Lastly, we applied the distortion mapping to the original images utilizing the `remap` function in OpenCV, thereby producing the distorted NKL1 and NKL2 datasets. However, it's noteworthy that the labels for the NKL dataset remained unaltered. Given that only result testing was performed on the NKL1 and NKL2 datasets without prior training on the distorted dataset, only images featuring distortion are essential, thus obviating the necessity of label processing.

The NKL1 dataset, characterized by the utilization of a smaller radial distortion parameter ($k_1=2 \times 10^{-7}$), manifests relatively mild image distortion; conversely, the NKL2 dataset employs a larger radial distortion parameter ($k_1=1 \times 10^{-6}$), thereby eliciting a more pronounced distortion effect on the images.

Distortion processing was conducted on the NKL dataset to emulate the deformations and distortions potentially encountered during real-world image acquisition processes. Subsequently, our method and the DHT method were employed to detect the distorted dataset, with the results compared against the Ground Truth of the NKL dataset. Figure 9 illustrates that our semantic segmentation line closely aligns with the Ground Truth. Distortion may engender diminished detection performance, particularly at the peripheries where distortions are pronounced, thereby engendering false positives or instances of undetected phenomena. Nonetheless, our proposed methodology exhibits superior performance compared to the DHT approach in managing distorted datasets.

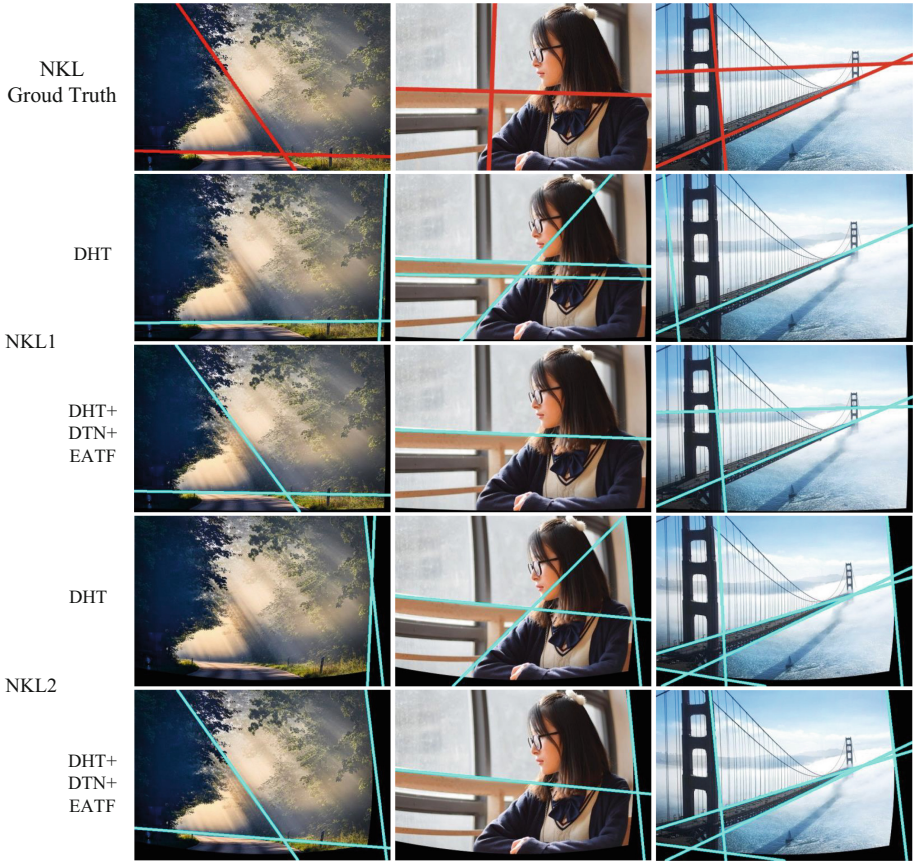


Fig. 9. Comparison of different methods in NKL distorted dataset.

As per the findings delineated in Table 3, our method surpasses the DHT method concerning accuracy, recall, and F-measure. This implies that our method possesses the capability to reconstruct the target information in the original dataset with greater precision, notwithstanding the distortions applied to the dataset. Furthermore, the disparity in F-measure post-distortion between our method and the DHT method exhibits a greater magnitude compared to the pre-distortion scenario. Deeper scrutiny of the data presented in Table 3 unveils the veracity that varying degrees of distortion do impact the detection outcomes. Specifically, it emerges that the NKL2 dataset is subject to more pronounced distortion in contrast to its NKL1 counterpart, thereby culminating in diminished detection efficacy. This outcome aligns with our anticipated hypotheses, suggesting that heightened dataset distortion correlates with increased detection complexity, hence concomitantly yielding diminished performance outcomes.

Nonetheless, in the presence of more severe distortion, our proposed methodology continues to exhibit superior performance relative to conventional

Table 3. Quantitative comparison on the NKL data.

Dataset	Method	Precision	Recall	F-measure
NKL1	DHT	62.37	72.75	67.16
	DHT+DTN+EATF	67.63	74.47	70.89
NKL2	DHT	29.09	58.32	38.82
	DHT+DTN+EATF	37.27	63.12	46.86

approaches. The findings suggest that our approach demonstrates enhanced robustness and generalization capabilities, effectively mitigating the effects of diverse levels of data distortion.

4.3 Ablation Experiment

A comprehensive series of ablation studies was undertaken to meticulously assess the efficacy of the proposed DHT+DTN +EATF architecture and verify its performance across various scenarios. These scenarios encompassed: 1) variances in DTN model architectures; 2) the incorporation of optimizer loading; 3) edge alignment training procedures.

Impact of Different DTN Model Structures: During the experimentation phase, we devised two model configurations for comparative analysis: a singular convolutional layer framework and a multi-layer convolutional structure, comprising convolutional layers, ReLU activation functions, and adjustments in channel quantity. These configurations underwent identical training protocols and parameter settings, followed by performance evaluation on the test dataset.

As delineated in Table 4, the multi-layer convolutional structure model demonstrated pronounced performance benefits on the test dataset, achieving an accuracy of 72.92%, surpassing that of the single convolutional layer model by 4% points. This underscores the efficacy of augmenting convolutional layers and integrating non-linear activation functions to bolster model performance for specific tasks. In terms of training duration, despite the additional parameters and computational load inherent in the multi-layer convolutional structure model, it incurred a mere 6-minute extension compared to the single convolutional layer framework. However, this supplementary training interval yielded discernible performance enhancements.

Table 4. Segmentation Performance of Various Models on the NKL Dataset.

Dataset	Model	Train time(<i>min</i>)	Precision	Recall	F-measure	F@0.95
NKL	Single Convolutional	160	68.95	78.78	73.54	47.62
	Multi-layer Convolutional	166	72.92	77.43	75.10	49.16

The Impact of Loading Optimizer: In the first scenario, an optimizer is incorporated into the model training process, employing a variant of the Adam gradient descent method to iteratively update model parameters; conversely, the second scenario omits the use of an optimizer, relying solely on the original gradient update rules.

According to the findings presented in Table 5, the inclusion of the optimizer led to discernible alterations in the model’s performance metrics. Following the incorporation of the optimizer, there was a notable augmentation in the F-measure metric. This suggests an enhancement in the overall performance attributable to the amelioration of the balance between precision and recall. Nevertheless, the precision marginally declined with the incorporation of the optimizer, juxtaposed with a slight increase in recall. This phenomenon implies a propensity towards positive class predictions induced by the optimizer, thereby augmenting recall while concurrently engendering more false positives, thereby diminishing precision. Furthermore, the introduction of the optimizer resulted in a slight augmentation of training duration. In summation, the optimizer yielded a discernible enhancement in the model’s performance within the scope of this experimental analysis.

Table 5. Evaluation of Optimizers’ Effects on Experiment.

Dataset	Optimizer	Train time (<i>min</i>)	Precision	Recall	F-measure	F@0.95
NKL	√	170	72.07	78.55	75.17	49.73
		166	72.92	77.43	75.10	49.16

The Influence of Edge Alignment: This study investigates the impact of edge alignment on the performance of the DHT+DTN+EATF model. Experiments are carried out under two conditions: with and without edge alignment training.

Table 6 illustrates that enabling the edge alignment operation has the potential to enhance the precision of the model moderately. The rise in model precision suggests a reduction in false detections facilitated by this operation. However, an observed marginal decline in model recall post-enablement of the edge alignment operation implies that certain targets might not be detected accurately, thereby diminishing the model’s recall rate. The comprehensive evaluation metrics, F-measure and F@0.95, demonstrate that enabling the edge alignment operation does not substantially influence the overall model performance but contributes modestly to its enhancement. Particularly noteworthy is the marginal enhancement observed in the model performance concerning F@0.95. Nonetheless, it is important to note that the utilization of edge alignment leads to a twofold increase in training time. Hence, the decision to enable edge alignment operation in practical scenarios should be carefully deliberated based on specific task demands and data characteristics to attain optimal detection performance.

Table 6. Impact of Edge Alignment on Experiments.

Dataset	EDGE-ALIGN	Train time(<i>min</i>)	Precision	Recall	F-measure	F@0.95
NKL	√	296	74.75	74.18	74.46	49.32
		166	72.92	77.43	75.10	49.16

Integrating the EATF and DTN modules inevitably increases computational demands. Nevertheless, these modules are specifically designed to enhance the network’s representational capacity and image processing accuracy, particularly in complex scenarios. Despite the increased computational complexity, the overall computational cost remains manageable, especially with optimized network architecture. By appropriately adjusting the number of convolutional kernels or reducing the input image resolution, computational load can be effectively managed, thus preserving high processing efficiency in resource-constrained or real-time applications. Overall, the increased computational burden facilitates a more effective balance between accuracy and performance.

The proposed method exhibits superior performance in both image distortion correction and semantic line detection, although it is not without its limitations. While the DTN module markedly enhances correction accuracy, it concurrently introduces additional computational overhead, potentially presenting challenges for real-time applications. Nevertheless, strategic hardware optimizations and algorithmic refinements can alleviate the computational burden while preserving high accuracy. The design of the Tanh activation function within the EATF module serves to mitigate the impact of short, sharp edges on the network. Although this approach may influence certain details under specific conditions, it predominantly enhances the detection of large-scale semantic lines.

5 Conclusions

This paper introduces the integration of the deep Hough transform with distortion correction subnetworks for the general task of detecting semantic segmentation lines. The proposed distortion correction subnetwork fully leverages deep learning and classical Hough transform to establish a robust global context, while allowing the network to dynamically adapt its feature extraction process to various image transformations. Extensive experimental results unequivocally demonstrate that the fusion of deep Hough transform with DTN yields superior performance and exceptional generalization capability when juxtaposed against extant methodologies. Furthermore, the proposed method demonstrates a trade-off between accuracy and efficiency. In future work, we will incorporate large language models and use CLIP to enhance the description of semantic segmentation lines, generating the required semantic segmentation lines.


References

1. Q. Hou, M.-M. Cheng, X. Hu, A. Borji, Z. Tu, and P. H. Torr, "Deeply supervised salient object detection with short connections," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 3203–3212
2. S.-H. Gao, Y.-Q. Tan, M.-M. Cheng, C. Lu, Y. Chen, and S. Yan, "Highly efficient salient object detection with 100k parameters," in *European Conference on Computer Vision*. Springer, 2020, pp. 702–721
3. Cheng, M.-M., Mitra, N.J., Huang, X., Torr, P.H., Hu, S.-M.: Global contrast based salient region detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **37**(3), 569–582 (2014)
4. W. Zhu, S. Liang, Y. Wei, and J. Sun, "Saliency optimization from robust background detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 2814–2821
5. L. Liu, R. Chen, L. Wolf, and D. Cohen-Or, "Optimizing photo composition," in *Computer graphics forum*, vol. 29, no. 2. Wiley Online Library, 2010, pp. 469–478
6. L. Mai, H. Jin, and F. Liu, "Composition-preserving deep photo aesthetics assessment," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 497–506
7. Fan, R., Wang, X., Hou, Q., Liu, H., Mu, T.-J.: Spinnet: Spinning convolutional network for lane boundary detection. *Computational Visual Media* **5**, 417–428 (2019)
8. V. Chhor and T. Kondo, "Illumination-invariant line detection with the gray-scale hough transform," in *2015 IEEE 7th International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM)*. IEEE, 2015, pp. 19–23
9. A. Nickfarjam, H. Ebrahimpour-Komleh, and A. A. Tehrani, "Binary image matching using scale invariant feature and hough transforms," in *2018 Advances in Science and Engineering Technology International Conferences (ASET)*. IEEE, 2018, pp. 1–5.
10. P. C. Teo and D. J. Heeger, "Perceptual image distortion," in *Proceedings of 1st International Conference on Image Processing*, vol. 2. IEEE, 1994, pp. 982–986
11. Fernandes, L.A., Oliveira, M.M.: Real-time line detection through an improved hough transform voting scheme. *Pattern Recogn.* **41**(1), 299–314 (2008)
12. Milletari, F., Ahmadi, S.-A., Kröll, C., Plate, A., Rozanski, V., Maiostre, J., Levin, J., Dietrich, O., Ertl-Wagner, B., Bötzel, K., et al.: Hough-cnn: Deep learning for segmentation of deep brain regions in mri and ultrasound. *Comput. Vis. Image Underst.* **164**, 92–102 (2017)
13. Ballard, D.H.: Generalizing the hough transform to detect arbitrary shapes. *Pattern Recogn.* **13**(2), 111–122 (1981)
14. Hu, X., An, Y., Shao, C., Hu, H.: Distortion convolution module for semantic segmentation of panoramic images based on the image-forming principle. *IEEE Trans. Instrum. Meas.* **71**, 1–12 (2022)
15. Kattenborn, T., Leitloff, J., Schiefer, F., Hinz, S.: Review on convolutional neural networks (cnn) in vegetation remote sensing. *ISPRS J. Photogramm. Remote. Sens.* **173**, 24–49 (2021)
16. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. *Proc. IEEE* **86**(11), 2278–2324 (1998)
17. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. *Commun. ACM* **60**(6), 84–90 (2017)

18. Samet, N., Hicsonmez, S., Akbas, E.: Houghnet: Integrating near and long-range evidence for visual detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **45**(4), 4667–4681 (2022)
19. J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440
20. Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans. Pattern Anal. Mach. Intell.* **40**(4), 834–848 (2017)
21. K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask r-cnn ieee international conference on computer vision (iccv),” *IEEE*, 2017
22. R. H. Hahnloser, R. Sarpeshkar, M. A. Mahowald, R. J. Douglas, and H. S. Seung, “Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit,” *nature*, vol. 405, no. 6789, pp. 947–951, 2000
23. A. L. Maas, A. Y. Hannun, A. Y. Ng *et al.*, “Rectifier nonlinearities improve neural network acoustic models,” in *Proc. icml*, vol. 30, no. 1. Atlanta, GA, 2013, p. 3
24. H. Noh, S. Hong, and B. Han, “Learning deconvolution network for semantic segmentation,” in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1520–1528
25. A. Kendall, Y. Gal, and R. Cipolla, “Multi-task learning using uncertainty to weigh losses for scene geometry and semantics,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7482–7491
26. T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, “Feature pyramid networks for object detection,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2117–2125
27. Zhao, K., Han, Q., Zhang, C.-B., Xu, J., Cheng, M.-M.: Deep hough transform for semantic line detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **44**(9), 4793–4806 (2021)
28. J. Hu, L. Shen, and G. Sun, “Squeeze-and-excitation networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141
29. J.-T. Lee, H.-U. Kim, C. Lee, and C.-S. Kim, “Semantic line detection and its applications,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 3229–3237



MemoFlow: Modifying Explicit Motion of Inconsistency in Optical Flow

Mengfei Wang^{1,2} , Wenjun Shi¹ , Dongchen Zhu^{1,2} , Lei Wang^{1,2} ,
and Jiamao Li^{1,2} 

¹ Bionic Vision System Laboratory, State Key Laboratory of Transducer Technology, Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences, Shanghai 200050, China

² University of Chinese Academy of Sciences, Beijing 100049, China
dchzhu@mail.sim.ac.cn

Abstract. Due to the limitation of training datasets and global motion understanding to optical flow estimation, current methods only focus on local clues and ignore the motion continuity of consecutive frames, resulting in an inconsistent motion problem. After theoretical analysis, we find the multi-frame methods need to pay more attention to the continuity between cost volumes than two-frame methods. Thus, our method is based on a multi-frame framework and introduces extra object coordinates in each frame by the segmentation model to revise the matching pairs in cost volume. Specifically, we introduce a Cost Volume Adaptation Module, including a Bbox Spatial Queries to store coordinates information and a Correlation Query Queue to query the object position of different frames. On the Sintel and KITTI test benchmark, our proposed MemoFlow achieves **1.00** and **1.69** average endpoint error (AEPE) on the clean and final passes and an F1-all error of **4.43%**, **ranking 1st** among all three-frame methods and two-frame methods.

Keywords: Inconsistency Motion · Correlation Query Queue · Bbox Spatial Queries

1 Introduction

Optical flow estimation is a fundamental computer vision task that involves estimating the pixel-level displacement field between consecutive frames. It is extensively applied in a variety of downstream video-related tasks, such as video interpolation, motion recognition, object detection, video understanding and analysis.

With the update of advanced neural network architectures, many effective optical flow methods have emerged [4, 26, 28]. However, existing optical flow estimations are limited by two problems: (1) **Limitations of simple motion scene in datasets.** Since it is difficult to label the ground truth of optical flow in the real world, most existing methods use synthetic methods [3, 16, 20]. But some methods [9] try to generate motions with simple 2D transformations. It greatly

limits the robustness of the optical flow estimation model in the face of complex motions. (2) **Lack of global understanding of motion continuity.** In general, the motion of an object in continuous time under the same scene should have continuity. The current methods ignore the global motion continuity and only focus on local clues, leading to **“inconsistent motion”**. Here, inconsistent motion refers to inconsistent optical flow of the same object in consecutive frames. Figure 1(a) shows examples of inconsistent optical flow in consecutive frames. It shows that the existing multi-frame methods have not paid attention to this problem.

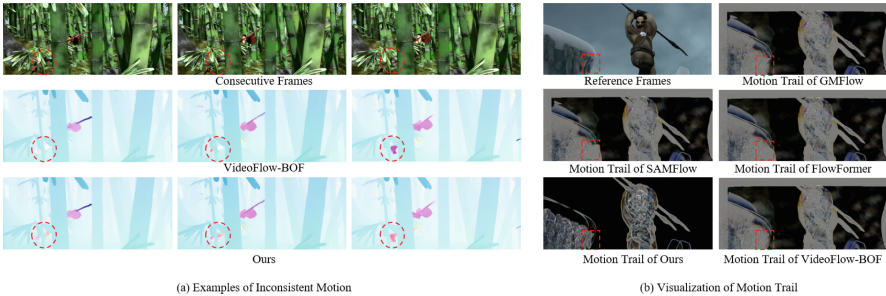


Fig. 1. (a) Examples of Inconsistency Motion in Optical Flow Estimation. We observed that our MemoFlow continuously estimates the optical flow of the same object in different frames. (b) Visualization of Motion Trail of Different Models.

Recent approaches have attempted to address these two problems by introducing some extra information. (1) Some methods [13, 15, 37, 39] find that combining large vision models for feature enhancement can improve optical flow estimation. However, these methods based on feature level cannot solve the problem caused by the motion structure. (2) MatchFlow [8] shows that the introduction of extra matching relationship information can enhance the optical flow estimation. And multi-frame optical flow methods [4, 26] strengthen the motion matching relationship of cost volume. In summary, we combine these components by incorporating extra matching information generated by the segmentation model into the cost volume of the multi-frame optical flow framework.

However, directly incorporating large segmentation models into the multi-frame optical flow framework would significantly reduce the model’s efficiency, especially considering that the multi-frame optical flow framework already has a substantial number of parameters. To solve this problem, we divide the training process into two stages. Specifically, for the first stage, we utilize a segmentation model to segment the objects in the training image data and store their bounding box information named Bbox Spatial Queries (BSQ). In the second stage, the stored BSQ and corresponding flow data are fed into the optical flow training framework to learn the optical flow with motion continuity. Specifically, we build a Correlation Query Queue (CQQ) with BSQ and image features. Then, the

CQQ and Cost Volume are transferred to the Cost Volume Adaption Module (CVAM) to repair the matching relationship. With the above designs, we upload our fine-tuned models to the benchmark sites of Sintel and KITTI-15, which show significant superiority. Our contributions are encapsulated in three key areas:

- For the first time, we introduce the segmentation information into multi-frame optical flow estimation, and we thus propose MemoFlow, a novel multi-frame approach aimed at improving optical flow estimation by effectively addressing issues of inconsistent motion.
- To prevent segmentation information from not being efficiently utilized, we propose a two-stage training scheme by introducing BSQ information stored in CQQ for subsequent query, and the CVAM to improve the cost volume matching accuracy.
- MemoFlow achieves remarkable results on the Sintel and KITTI-2015 tests, with AEPE of **1.00** and **1.69** on the clean and final passes and an F1-all error of **4.43%**, **ranking 1st** among all three-frame methods.

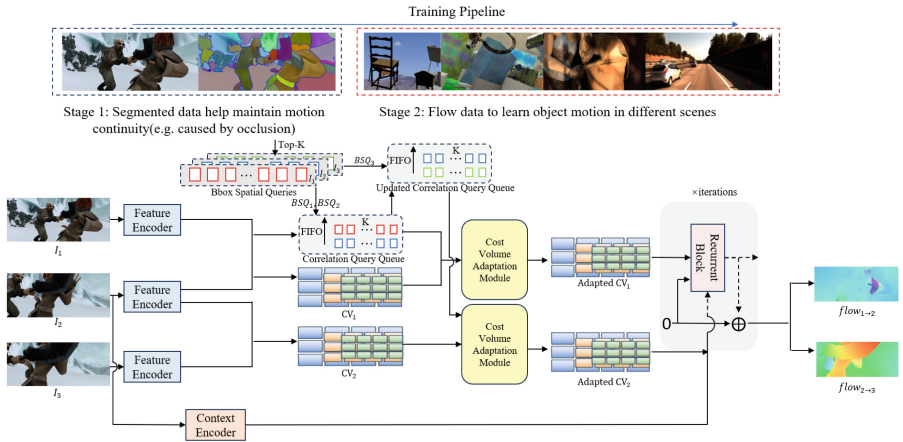


Fig. 2. Overview of our MemoFlow. The simplified training pipeline is shown at the top, while details are specifically listed below. We design two modules to adapt cost volumes, including the BSQ which stores bbox of objects, and the CQQ to perform temporal interaction of BSQ. The boxes with different colors in the BSQ represent objects stored in different frames.

2 Related Work

2.1 Multi-frame Optical Flow Framework

In the era of optimization-based optical flow, researchers use Kalman filters [1,6,11] to estimate optical flow through the temporal dynamics of motion.

PWC-Fusion [24, 30] fuses information from frames with GRU-RCN at the U-Net bottleneck. However, due to coarse feature encoding, the performance gain is weak (0.65% higher than PWC-Net). MFCFlow [4] constructs a Motion-Guided Feature Compensation unit to enhance blurry motion features based on the correlation of previous features. VideoFlow [26] utilizes temporal cues to simultaneously estimate motion features of bidirectional flow for multiple frames available in the video. Furthermore, SplatFlow [33] proposes a motion feature alignment method based on splatting to replace commonly used backward flow computation or non-differentiable forward warping transformations. However, these multi-frame methods focus more on network modules to learn and aggregate feature information between frames, ignoring their correlation.

2.2 Optical Flow with Segmentation

Semantic segmentation has been widely integrated with optical flow estimation [2, 17, 22]. These methods decompose optical flow estimation into motion segmentation networks to estimate optical flow from the perspective of motion segmentation. Some methods [5, 34, 36] guide the algorithm to determine smooth positions of optical flow in the scene based on objective information. Furthermore, research on joint learning algorithms for video semantic segmentation and optical flow is proposed [7]. However, these methods share the same drawbacks: they neglect consecutive frame information interactions and motion continuity.

2.3 Motion Information Propagation

Motion guidance information is commonly applied to inter-frame target tracking and multi-view 3D reconstruction tasks. CAMOT [23] employs stereo information for mask-based tracking of generic objects on the KITTI dataset. Lu et al. [18] achieve tracking by aggregating position and appearance features for each frame using LSTM. MaskFusion [25] proposes a 3D mapping system that builds entire geometric maps using object-aware fusion between multiple perspectives at different times. Building upon these methods, we introduce motion continuity information into the optical flow estimation task for the first time.

3 Method

3.1 Theoretical Analysis

Multi-frame optical flow estimation methods aim to find the mapping relation $F : (I_1, I_2, I_3) \mapsto (f_1, f_2)$, where I_1, I_2 and I_3 are three consecutive frames, and (f_1, f_2) are the 2D optical flow of (I_1, I_2) and (I_2, I_3) . Through the perspective of a probability density function, the optical flow network can be formulated as:

$$(f_1, f_2)^* = F_\theta(I_1, I_2, I_3) = \arg \max_{(f_1, f_2)} p(f_1, f_2 | I_1, I_2, I_3), \quad (1)$$

where $(f_1, f_2)^*$ represent the most likely optical flows, F_θ is the network with parameters θ and $p(f_1, f_2 | I_1, I_2, I_3)$ represents a posterior probabilistic distribution of optical flow.

According to Bayes' theorem, we expand $p(f_1, f_2 | I_1, I_2, I_3)$ as follows:

$$\begin{aligned} p(f_1, f_2 | I_1, I_2, I_3) &= \frac{p(f_1, f_2)p(I_1, I_2, I_3 | f_1, f_2)}{p(I_1, I_2, I_3)} \\ &= \frac{p(f_1, f_2)p(I_2 | f_1, f_2)p(I_1 | I_2, f_1, f_2)p(I_3 | I_1, I_2, f_1, f_2)}{p(I_1, I_2, I_3)} \\ &= \frac{p(I_2)p(f_1, f_2 | I_2)p(I_1 | I_2, f_1)p(I_3 | I_1, I_2, f_1, f_2)}{p(I_1, I_2, I_3)}. \end{aligned} \tag{2}$$

In search of optimal solution of (f_1, f_2) , we omit irrelevant items $p(I_2)$ and $p(I_1, I_2, I_3)$, and apply maximum likelihood estimation. Then we get a new formulation (Formula 3) of Formula 1 as follows:

$$\begin{aligned} (f_1, f_2)^* &= \arg \max_{(f_1, f_2)} \underbrace{\{\log(p(f_1, f_2 | I_2))\}}_{\text{context}} \\ &\quad + \underbrace{\{\log(p(I_1 | I_2, f_1))\}}_{\text{cost volume 1}} + \underbrace{\{\log(p(I_3 | I_1, I_2, f_1, f_2))\}}_{\text{cost volume 2}}. \end{aligned} \tag{3}$$

The context term provides auxiliary information for optical flow estimation and requires the model to have a deep understanding of image features. To achieve this, optical flow models such as [39, 40] enhance context features to help the model to comprehend the image feature more deeply.

However, as the number of input frames increases, the influence of **cost volume items** on optical flow estimation gradually increases. These continuous motion relationships are much more helpful to the model than the context item. Earlier multi-frame approaches [4, 26, 31] overlook this aspect, with the continuity of motion constrained to local clues. In contrast, our endeavor involves integrating consecutive cost volumes and introducing extra spatial information to enhance motion continuity. However, not all the spatial information can help to realize the continuity of motion. Empirically, segmentation information is a suitable candidate with its capability to maintain the integrity of moving objects.

3.2 Pipeline Overview

As depicted in Fig. 2, the training pipeline includes two stages. For stage one, the images of optical flow data are segmented panoramically. Information such as the bounding box of objects with high confidence are saved as Bbox Spatial Queries (BSQ) information (Sect. 3.3). Therefore, given a set of images (I_1, I_2, I_3) as inputs, the position coordinates of the objects with the highest confidence in different frames are recorded as (BSQ_1, BSQ_2, BSQ_3) . For stage two, to better fuse spatial information, we construct a Correlation Query Queue (CQQ) to perform temporal interaction. Moreover, since CQQ has a recursive matching

query ability, we design a Cost Volume Adaptation Module to revise the cost volume. This progressive pipeline allows our MemoFlow to learn motion continuity better.

3.3 Bbox Spatial Queries

First, we elaborate on the concept of establishing Bbox Spatial Queries (BSQ) using a panoramic segmentation model. Previous methods are trained directly on optical flow data without incorporating additional information to learn scene structure. MatchFlow [8] demonstrates that introducing simple scene-consistent matching information early in training can aid the network in learning its representation. However, MatchFlow requires additional pre-training, which increases the model’s parameters and affects its inference efficiency. In contrast, we utilize a pre-trained segmentation model with generalization capability [21, 42] to segment the training images and save the Top-K bounding box information for each frame. This information is then directly input into the optical flow model along with the images for training, thus maintaining the model’s inference efficiency.

In the segmentation process, it is necessary to align the image size of segmentation with the image size in the optical flow training. Specifically, we construct the Bbox Spatial Queries as follows:

$$BSQ = \varphi(I) \in R^{K \times 4}, \quad (4)$$

where I is the training image of optical flow data, φ is the segmentation model, K is the number of object bounding boxes stored in the BSQ (with Top-K highest confidence score) and Dimension “4” is information ($bbox_{x_0}, bbox_{y_0}, bbox_w, bbox_h$) for each bounding box. According to the experimental results of Table 4, we set $K = 5$.

3.4 Correlation Query Queue

We design a correlation query queue $CQQ \in R^{2 \times K \times 4}$ for effective motion continuity modeling. Dimension “2” is the number of stored frames and “ $K \times 4$ ” is the stored **BSQ** of every frame. When the input is I_1, I_2 , the Correlation Query Queue CCQ_1 is as follows:

$$CCQ_1 = [BSQ_1, BSQ_2]^T \in R^{2 \times K \times 4}. \quad (5)$$

CCQ_1 can help to revise the matching relationship in the cost volume CV_1 of I_1, I_2 to ensure the continuity of motion (Sect. 3.5).

Update of the CQQ: The entrance and exit of the correlation query queue follow the first-in, first-out (FIFO) rule. As information BSQ_3 from a new frame I_3 is added to the queue, the oldest information BSQ_1 of I_1 is discarded. The formula of the new CCQ_2 is as follows:

$$CCQ_2 = [BSQ_2, BSQ_3]^T \in R^{2 \times K \times 4}. \quad (6)$$

CCQ_2 can help to revise the matching relationship in the cost volume CV_2 of I_2, I_3 (Sect. 3.5).

Algorithm 1. The Process of the Cost Volume Adaptation Module**Input:** Cost Volume CV_1 , CQQ_1 with (BSQ_1, BSQ_2) and image features (Φ_1, Φ_2) .**Output:** The Adapted Cost Volume ACV_1 .

- 1: Compute the object feature queries (Ω_1, Ω_2) by Formula 7.
- 2: Compute the matching relationship M_1 between (Ω_1, Ω_2) by Formula 8:

$$M_1 \leftarrow \text{softmax}\left(\frac{\Omega_1 \Omega_2^T}{\sqrt{K}}\right) \in R^{K \times K}.$$
- 3: Compute the column index of the maximum value of each row of the M_1 matrix and the matching index pairs are stored in the matrix $P_1 \in R^{K \times 2}$.
- 4: Set the first row of P_1 be each index of object in Ω_1 and the second row of P_1 be the index of each object in Ω_2 that matches the object in Ω_1 .
- 5: Replace the coordinate matching relations of CV_1 by the coordinate matching relation queried by (BSQ_1, BSQ_2) in P_1 by Formula 9.
- 6: Adapt the Cost Volume:

$$ACV_1 \leftarrow CV_1$$
- 7: **return** ACV_1

3.5 Cost Volume Adaption Module

To better utilize the object query information in BSQ to achieve motion continuity, we propose a Cost Volume Adaptation Module. Inspired by GMFlow [37] and ReID [10] method, we calculate the matching of objects between frames in CQQ and replace the optimized matching into the Cost Volume according to the retained query information, which consists of three steps. As shown in Algorithm 1, we show the adaption to Cost Volume CV_1 of I_1, I_2 as an example:

(1) Object Feature Query: As shown in Fig. 2, K objects corresponding to each frame feature are queried through the corresponding BSQ information stored by CQQ :

$$\begin{aligned}\Omega_1 &= BSQ_1(\Phi_1) \in R^{K \times h \times w}, \\ \Omega_2 &= BSQ_2(\Phi_2) \in R^{K \times h \times w},\end{aligned}\tag{7}$$

where (Φ_1, Φ_2) are the features of (I_1, I_2) , (Ω_1, Ω_2) are bounding box corresponding object feature set in (BSQ_1, BSQ_2) , K is the number of objects stored per frame and $h \times w$ is the shape of each object. In the experiment, the shape is determined by the bounding box with a uniform set size.

(2) Matching Relationship Between (Ω_1, Ω_2) : As shown in Formula 8, we use the cross-attention layer to calculate the matching relationship. Before the calculation, we compress the dimension of Ω_1, Ω_2 into the two-dimensional $R^{K \times hw}$:

$$M_1 = \text{softmax}\left(\frac{\Omega_1 \Omega_2^T}{\sqrt{K}}\right) \in R^{K \times K},\tag{8}$$

where matrix M_1 is the correlation between K objects in Ω_1 and K objects in Ω_2 . By taking out the column index corresponding to the maximum value of each row of M_1 , the index matching relation matrix $P_1 \in R^{K \times 2}$ can be constructed.

The first row of the matrix P_1 represents each index of each object in Ω_1 . The second row of the matrix P_1 represents the index of each object in Ω_2 that matches the object in Ω_1 .

(3) Adaptation of the Cost Volume: Through the index correspondence contained in P_1 and bounding box information stored by BSQ , the corresponding coordinate matching can be obtained. Then replace the matching pairs in the Cost Volume CV_1 with the matching pairs of these objects.

$$CV_1[BSQ_1(P_1[0, :])] = BSQ_2(P_1[1, :]). \quad (9)$$

The Adapted Cost Volume CV_1 is defined as ACV_1 .

4 Experiments

4.1 Implementation Details

According to the FlowFormer series [13, 27], the image feature encoder and context feature encoder are chosen as the first two stages of the ImageNet-pretrained Twins-SVT. They are frozen during pretraining to achieve better performance. Since the FlyingChairs [9] dataset contains only pairs of training frames, we modified it by taking multi-frames in a group and fine-tuned it on the FlyingThings [20] dataset. For our model, we pretrain it for 300k iterations on the FlyingChairs and FlyingThings dataset (denoted as “ $C + T$ ”). Then, we fine-tune it for 120k iterations on data from FlyingChairs, FlyingThings, Sintel [3], KITTI-2015 [12], and HD1K [16] (denoted as “ $C + T + S + K(+H)$ ”). We further fine-tuned the model for 50k iterations on the KITTI-2015 dataset. AdamW optimizer and a one-cycle learning rate scheduler are employed. The batch size for all stages is set to 6. The highest learning rate is set to 2×10^{-4} for FlyingChairs and 1.2×10^{-4} for other training datasets. We use Average Endpoint Error (AEPE) and F1-All (%) as evaluation metrics. F1-All calculates the percentage of pixels with flow errors greater than 3 pixels or exceeding 5% of the ground truth.

4.2 Comparison with State-of-the-Art Methods

Generalization Performance: In Table 1, the “ $C + T$ ” setting reflects the model’s cross-dataset generalization ability. Specifically, in the challenging final pass, MemoFlow ranks first and surpasses the VideoFlow series. It is noteworthy that VideoFlow and FlowFormer++ have 37.7% and 85.7% more parameters than MemoFlow (13.5M vs. 18.2M vs. 9.8M). Additionally, FlowFormer++ undergoes pre-training using a masked autoencoder strategy on the YouTube-VOS dataset [38]. And for the clean pass and KITTI-2015, MemoFlow achieves performance on par with the state-of-the-art methods.

Table 1. Experiments on Sintel [3] and KITTI [12] datasets. “*” denotes the multi-frame methods. We use **bold** and to highlight the methods that rank 1st and 2nd.

Training Data	Method	Sintel(train)		KITTI-15(train)		Sintel(test)		KITTI-15(test)
		Clean	Final	EPE	F1-all	Clean	Final	F1-all
C+T	PWC-Net [29]	2.55	3.93	10.35	33.7	—	—	—
	FlowNet2 [14]	2.02	3.54	10.08	30.0	3.96	6.02	—
	SKFlow [31]	1.22	2.46	4.27	15.5	—	—	—
	RAFT [32]	1.43	2.71	5.04	17.4	—	—	—
	FlowFormer [13]	<u>0.94</u>	2.33	4.09	14.72	—	—	—
	FlowFormer++ [27]	0.90	2.30	3.93	14.13	—	—	—
	CRAFT [28]	1.27	2.79	4.88	17.5	—	—	—
	GMFlow [37]	1.08	2.48	—	—	—	—	—
	GMA [15]	1.30	2.74	4.69	17.1	—	—	—
	Videoflow-BOF [26]	1.03	<u>2.19</u>	3.96	15.33	—	—	—
	Videoflow-MOF [26]	1.18	2.56	3.89	<u>14.20</u>	—	—	—
	Ours	0.98	2.09	<u>3.91</u>	14.39	—	—	—
C+T+S+K(+H)	PWC-Net [29]	(1.71)	(2.34)	(1.50)	(5.3)	3.44	4.58	7.71
	FlowNet2 [14]	(1.45)	(2.01)	(2.29)	(6.79)	4.14	5.73	11.47
	GMFlow [37]	—	—	—	—	1.74	2.90	9.32
	SKFlow* [31]	(0.52)	(0.78)	(0.51)	(0.94)	1.28	2.27	4.48
	RAFT* [32]	(0.75)	(1.21)	(0.63)	(1.5)	1.94	3.18	5.11
	CRAFT [28]	(0.60)	(1.06)	(0.58)	(1.33)	1.45	2.40	4.81
	GMA [15]	(0.62)	(1.07)	(0.57)	(1.2)	1.39	2.48	5.14
	PWC-Fusion* [24]	—	—	—	—	3.43	4.57	7.17
	FlowFormer [13]	(0.48)	(0.74)	(0.53)	(1.11)	1.16	2.09	4.68
	FlowFormer++ [27]	(0.40)	(0.60)	(0.57)	(1.16)	1.07	1.94	4.52
	MatchFlow(R) [8]	(0.51)	(0.81)	(0.59)	(1.3)	1.33	2.64	4.72
	SAMFlow [39]	—	—	—	—	1.00	2.08	4.49
	MatchFlow(G) [8]	(0.49)	(0.78)	(0.55)	(1.1)	1.16	2.37	4.63
	Videoflow-BOF* [26]	<u>(0.37)</u>	<u>(0.54)</u>	<u>(0.52)</u>	(0.85)	<u>1.02</u>	<u>1.84</u>	<u>4.44</u>
	SplatFlow* [33]	(0.53)	(0.91)	(0.80)	(2.4)	1.12	2.07	4.61
	Ours	(0.30)	(0.42)	(0.49)	(0.87)	1.00	1.69	4.43

Dataset-Specific Performance: After the “ $C + T + S + K(+H)$ ” stage, we submit results to the online Sintel and KITTI benchmark test. As shown in Table 1, our model has outperformed most of the published methods, achieving **1.0** and **1.69** AEPE on the Sintel clean and final passes. In particular, our model has a great advantage over SAMFlow and Matchflow on the final pass. **It is proved that although additional information is introduced, the effect on motion continuity matching is superior to feature enhancement.** MemoFlow achieves a 4.43% F1-all error, surpassing the SOTA two-frame and multi-frame methods, FlowFormer++ and VideoFlow-BOF on KITTI-2015. Using the same three-frame setup, our model has achieved an overall improve-

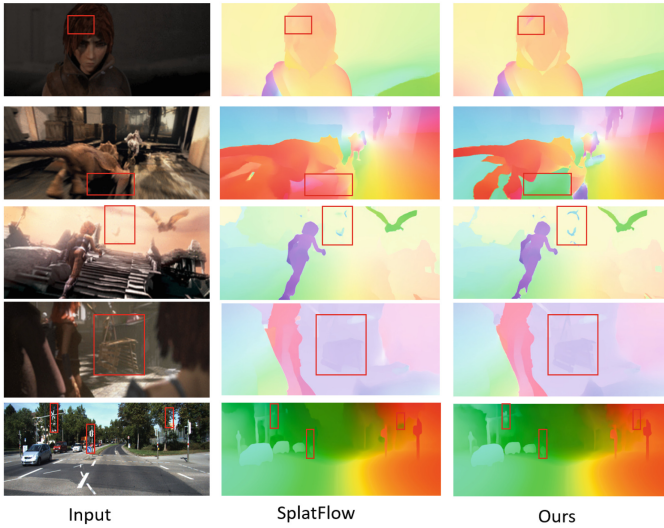


Fig. 3. It can be seen from the red box that MemoFlow better preserves the continuity of motion and the integrity of the objects. It is helpful to use optical flow information to understand the scene. (Color figure online)

Table 2. Results on the Sintel Test Set with Different Evaluation Metrics. ‘Unmatched’ refers pixels failed to match in adjacent frames and $s_{0-10}, s_{10-40}, s_{40+}$ denote pixels with ground truth flow motion magnitude falling in 0 – 10, 10 – 40, and more than 40 pixels, respectively.

Method	Sintel Test(clean)						Sintel Test(final)					
	All	Matched	Unmatched	s_{0-10}	s_{10-40}	s_{40+}	All	Matched	Unmatched	s_{0-10}	s_{10-40}	s_{40+}
SKFlow [31]	1.298	0.567	7.251	0.282	0.950	7.173	2.261	1.138	11.415	0.577	1.681	12.015
SAMFlow [39]	1.000	0.384	5.966	0.252	0.760	5.245	2.080	1.036	10.60	0.515	1.488	11.278
VideoFlow-BOF [26]	1.005	<u>0.389</u>	6.023	0.229	0.695	5.605	<u>1.713</u>	<u>0.812</u>	<u>9.054</u>	0.387	1.242	<u>9.422</u>
FlowFormer++ [27]	1.073	0.390	6.635	0.252	0.796	5.810	1.943	0.878	10.627	0.438	1.404	10.712
Ours(MemoFlow)	1.000	0.399	5.889	<u>0.233</u>	<u>0.708</u>	<u>5.503</u>	1.692	0.805	8.917	<u>0.407</u>	<u>1.262</u>	9.098

ment over VideoFlow-BOF. It demonstrates that our method brings significant accuracy improvements for multi-frame optical flow estimation. These results can be found on Sintel and KITTI-15 benchmark websites.

Performance Analysis for Different Regions: To fully investigate the performance of MemoFlow, additional metrics are provided in Table 2, where “**Unmatched**” refers to the EPE on **pixels failed to match in adjacent frames**, and $s_{0-10}, s_{10-40}, s_{40+}$ respectively indicate the EPE on pixels with ground truth flow magnitudes reduce to 0 – 10, 10 – 40, and over 40 pixels. We select four of the most competitive methods, SKFlow, SAMFlow, VideoFlow, and FlowFormer++ for comparison. Unmatched pixels pose challenges as they

Table 3. Results of Different Segmentation Models.

Methods	Sintel(train)		KITTI(train)	
	clean	final	F1-epe	F1-all
w/o Segmentation Model	0.48	0.74	0.69	1.11
MobileSAM [42]	0.30	<u>0.42</u>	0.48	<u>0.92</u>
EfficientPS [21]	<u>0.34</u>	0.49	0.54	0.91
YOSO [41]	0.37	0.40	<u>0.49</u>	0.87

Table 4. Results of Calculation Method of Matching Relationship in CVAM and Number of Objects Stored in BSQ.

Calculation Method	Length of the BSQ	Sintel(train)		Things(val)		KITTI(train)	
		clean	final	clean	final	F1-epe	F1-all
None	0	1.46	2.48	2.64	2.51	4.63	16.57
DeepSort [10, 35]	1	1.37	2.34	2.29	2.04	4.32	15.11
	3	1.24	2.25	2.01	1.69	4.05	14.86
	<u>5</u>	1.12	2.12	1.67	1.42	3.89	14.44
	7	1.14	2.11	1.52	1.44	3.91	14.52
Cosine Similarity [19]	1	1.41	2.46	2.17	2.04	4.29	15.31
	3	1.33	2.25	1.64	1.57	4.03	14.92
	<u>5</u>	1.09	2.14	1.45	1.36	3.92	14.53
	7	1.1	2.13	1.44	1.39	3.91	<u>14.4</u>
Cross-Attention(Ours)	1	1.29	2.31	2.15	2.07	4.25	15.23
	3	1.1	2.16	1.69	1.46	4.01	14.66
	<u>5</u>	0.98	<u>2.09</u>	<u>1.44</u>	1.32	<u>3.91</u>	14.39
	7	<u>1.01</u>	2.08	1.43	<u>1.35</u>	3.93	14.41

are invisible in the image compared to matched pixels. However, both on clean pass and final pass, our MemoFlow achieves the **best performance in the “Unmatched” region**. It is a strong validation of the effectiveness of our approach. **It proves that focusing on motion continuity can alleviate the occlusion problems such as frame mismatch.** As shown in Fig. 1 (b), MemoFlow can describe the trail of objects more accurately.

Similarly, pixels with larger flow magnitudes are challenging, particularly on the final pass, as faster motion leads to more severe motion blur. For large displacement, feature enhancement has certain advantages. However, the performance of our Memoflow on large displacement scenes is not inferior to the method of enhancing features. It’s even better than SAMFlow and SKFlow on the final pass. As shown in red boxes in Fig. 1 (b), MemoFlow can show accurate motion trails with more precise details.

Table 5. The Parameters and Inference of Different Models.

Model	Para.	Infer.
GMA [15]	5.9M	74 ms
FlowFormer [13]	18.2M	149 ms
SAMFlow [39]	–	450 ms
CRAFT [28]	6.4M	116ms
MatchFlow(G) [8]	15.4M	126 ms
MemoFlow(Ours)	9.8M	163 ms

Qualitative Results: In Fig. 3, we present examples of visualization results on the final pass of the Sintel and the KITTI test sets. Each row, from left to right, represents input images and predicted flows of the multi-frame method SplatFlow and our MemoFlow. Compared to SplatFlow, MemoFlow performs better in regions with small object details (third row, fifth row) and areas prone to unmatched issues such as occlusions (second row, fifth row). At the same time, MemoFlow also demonstrates advantages in terms of object integrity (first row, fourth row). It shows that motion continuity is still valid in large displacement scenarios.

4.3 Ablation Experiment

Segmentation Model: We establish a baseline based on Flowformer with the encoder replaced by a combination of SKFlow [31], Flowformer [13], and GMFlow [37]. To verify the quality of Bbox Spatial Queries obtained by different segmentation models, we perform the ablation. To prevent the segmentation in stage one from taking too long time, we choose some efficient and lightweight segmentation models, such as EfficientPS [21], YOSO [41], and MobileSAM [42].

As shown in Table 3, due to performance differences among different segmentation models, it is evident that each segmentation model performs better than the baseline (w/o Segmentation Model). It shows the efficiency of segmented information. In different datasets, the performance ability of the model is different. In Sintel, “MobileSAM” performs best. In KITTI, “YOSO” performs best. Therefore, in the second training stage, different datasets can adopt different segmentation results.

Number of Objects Stored in BSQ: As shown in Table 4, we validate the number of objects stored in Bbox Spatial Queries as 3, 5, and 7, analyzing the impact of different numbers K (Sect. 3.3, 3.4, 3.5) on optical flow estimation. These results indicate that increasing the number of objects is not necessarily beneficial for performance. The optimal value of K varies across different datasets. We find that the features contained in multi-frames are sufficient to capture the motion information of self-moving objects, so increasing the query

length beyond this does not yield significant improvement. Considering computational efficiency and storage requirements, we set K to 5.

Calculation Method of Matching Relationship in CVAM: In the Cost Volume Adaptation Module (CVAM), the method used to calculate inter-frame object feature correlation significantly affects model performance. In the upper, middle, and lower columns of Table 4, we present the results of various calculation methods, including DeepSort [10,35], Cosine Similarity [19], and our Cross-Attention approach. DeepSort and Cosine Similarity are commonly used for recognizing similar pedestrian features in the ReID task. Our Cross-Attention method outperforms the others due to its integration into the training stage.

Parameters and Runtime Analysis: We present the computational overhead of our models in Table 5. Despite the inclusion of additional BSQ information, MemoFlow has a relatively low parameter count (9.8M). It is particularly notable when compared to MatchFlow (15.4M), which also incorporates additional information. In terms of inference time, MemoFlow demonstrates shorter run times (163ms) compared to SAMFlow (450ms), which shares a baseline (FlowFormer) with MemoFlow.

5 Conclusion

This paper focuses on the challenging problem of motion inconsistency in optical flow estimation. Firstly, the importance of matching relations in multi-frame optical flow estimation is analyzed theoretically. Therefore, we propose MemoFlow, which inputs additional matching information into the optical flow estimation network. Next, to integrate the new matching information into the optical flow estimation, we introduce a specific multi-frame Cost Volume Adaptation Module, including BSQ and CQQ. In experiments, we demonstrated the effectiveness of MemoFlow in motion consistent matching and its superiority in optical flow estimation accuracy, achieving state-of-the-art performance and ranking 1st among all three-frame methods on the Sintel and KITTI test sets. However, our approach currently falls short in effectively encoding segmentation information into the well-established knowledge domain of optical flow. In future work, we consider how to take both advantages of segmentation and cost volume.

Acknowledgements. This work was supported by National Science and Technology Major Project from Minister of Science and Technology, China (2021ZD-0201403), National Natural Science Foundation of China (62103399), Youth Innovation Promotion Association, Chinese Academy of Sciences (2021233, 202324-2), Shanghai Academic Research Leader (22XD1424500).

References

1. Bao, Wenbo, et al. “KalmanFlow 2.0: Efficient video optical flow estimation via context-aware kalman filtering.” *IEEE Transactions on Image Processing* 28.9 (2019): 4233-4246
2. Brox, Thomas, and Jitendra Malik. “Object segmentation by long term analysis of point trajectories.” *European conference on computer vision*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010
3. Butler, D., et al. “MPI-Sintel optical flow benchmark: Supplemental material.” MPI-IS-TR-006, MPI for Intelligent Systems (2012). Citeseer, 2012
4. Chen, Yonghu, et al. “MFCFlow: A Motion Feature Compensated Multi-Frame Recurrent Network for Optical Flow Estimation.” *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2023
5. Cheng, Jingchun, et al. “Segflow: Joint learning for video object segmentation and optical flow.” *Proceedings of the IEEE international conference on computer vision*. 2017
6. Chin, Toshio M., William Clement Karl, and Alan S. Willsky. “Probabilistic and sequential computation of optical flow using temporal coherence.” *IEEE Transactions on Image Processing* 3.6 (1994): 773-788
7. Ding, Mingyu, et al. “Every frame counts: Joint learning of video segmentation and optical flow.” *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. No. 07. 2020
8. Dong, Qiaole, Chenjie Cao, and Yanwei Fu. “Rethinking Optical Flow from Geometric Matching Consistent Perspective.” *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023
9. Dosovitskiy, Alexey, et al. “FlowNet: Learning optical flow with convolutional networks.” *Proceedings of the IEEE international conference on computer vision*. 2015
10. Du, Yunhao, et al. “StrongSORT: Make deepSORT great again.” *IEEE Transactions on Multimedia* (2023)
11. Elad, M., Feuer, A.: Recursive optical flow estimation—adaptive filtering approach. *J. Vis. Commun. Image Represent.* **9**(2), 119–138 (1998)
12. Geiger, Andreas, et al. “Vision meets robotics: The kitti dataset.” *The International Journal of Robotics Research* 32.11 (2013): 1231-1237
13. Huang, Zhaoyang, et al. “Flowformer: A transformer architecture for optical flow.” *European Conference on Computer Vision*. Cham: Springer Nature Switzerland, 2022
14. Ilg, Eddy, et al. “FlowNet 2.0: Evolution of optical flow estimation with deep networks.” *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017
15. Jiang, Shihao, et al. “Learning to estimate hidden motions with global motion aggregation.” *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021
16. Kondermann, Daniel, et al. “The hci benchmark suite: Stereo and flow ground truth with uncertainties for urban autonomous driving.” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 2016
17. Wang, Xianshun, et al. “Camera Parameters Aware Motion Segmentation Network with Compensated Optical Flow.” 2021 *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021
18. Lu, Yongyi, Cewu Lu, and Chi-Keung Tang. “Online video object detection using association LSTM.” *Proceedings of the IEEE International Conference on Computer Vision*. 2017

19. Luo, C., Zhan, J., Xue, X., Wang, L., Ren, R., Yang, Q.: Cosine Normalization: Using Cosine Similarity Instead of Dot Product in Neural Networks. In: Kůrková, V., Manolopoulos, Y., Hammer, B., Iliadis, L., Maglogiannis, I. (eds.) ICANN 2018. LNCS, vol. 11139, pp. 382–391. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01418-6_38
20. Mayer, Nikolaus, et al. “A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation.” Proceedings of the IEEE conference on computer vision and pattern recognition. 2016
21. Mohan, R., Valada, A.: Efficientps: Efficient panoptic segmentation. *Int. J. Comput. Vision* **129**(5), 1551–1579 (2021)
22. Ochs, Peter, and Thomas Brox. “Higher order motion models and spectral clustering.” 2012 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2012
23. Ošep, Aljoša, et al. “Track, then decide: Category-agnostic vision-based multi-object tracking.” 2018 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2018
24. Ren, Zhile, et al. “A fusion approach for multi-frame optical flow estimation.” 2019 IEEE Winter Conference on Applications of Computer Vision (WACV). IEEE, 2019
25. Runz, Martin, Maud Buffier, and Lourdes Agapito. “Maskfusion: Real-time recognition, tracking and reconstruction of multiple moving objects.” 2018 IEEE International Symposium on Mixed and Augmented Reality (ISMAR). IEEE, 2018
26. Shi, Xiaoyu, et al. “Videoflow: Exploiting temporal cues for multi-frame optical flow estimation.” arXiv preprint [arXiv:2303.08340](https://arxiv.org/abs/2303.08340) (2023)
27. Shi, Xiaoyu, et al. “Flowformer++: Masked cost volume autoencoding for pre-training optical flow estimation.” Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023
28. Sui, Xiuchao, et al. “Craft: Cross-attentional flow transformer for robust optical flow.” Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition. 2022
29. Sun, Deqing, et al. “Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume.” Proceedings of the IEEE conference on computer vision and pattern recognition. 2018
30. Sun, Deqing, et al. “Models matter, so does training: An empirical study of cnns for optical flow estimation.” *IEEE transactions on pattern analysis and machine intelligence* **42.6** (2019): 1408-1423
31. Sun, Shangkun, et al. “Skflow: Learning optical flow with super kernels.” *Advances in Neural Information Processing Systems* **35** (2022): 11313-11326
32. Teed, Z., Deng, J.: RAFT: Recurrent All-Pairs Field Transforms for Optical Flow. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12347, pp. 402–419. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58536-5_24
33. Wang, Bo, et al. “SplatFlow: Learning Multi-frame Optical Flow via Splatting.” arXiv preprint [arXiv:2306.08887](https://arxiv.org/abs/2306.08887) (2023)
34. Weinzaepfel, Philippe, et al. “DeepFlow: Large displacement optical flow with deep matching.” Proceedings of the IEEE international conference on computer vision. 2013
35. Wojke, Nicolai, Alex Bewley, and Dietrich Paulus. “Simple online and realtime tracking with a deep association metric.” 2017 IEEE international conference on image processing (ICIP). IEEE, 2017

36. Wulff, Jonas, and Michael J. Black. “Efficient sparse-to-dense optical flow estimation using a learned basis and layers.” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015
37. Xu, Haoifei, et al. “Gmflow: Learning optical flow via global matching.” *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022
38. Xu, Ning, et al. “Youtube-vos: A large-scale video object segmentation benchmark.” *arXiv preprint [arXiv:1809.03327](https://arxiv.org/abs/1809.03327)* (2018)
39. Zhou, Shili, et al. “SAMFlow: Eliminating Any Fragmentation in Optical Flow with Segment Anything Model.” *arXiv preprint [arXiv:2307.16586](https://arxiv.org/abs/2307.16586)* (2023)
40. Cheng, Ri, et al. “Context-Aware Iteration Policy Network for Efficient Optical Flow Estimation.” *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 38. No. 2. 2024
41. Hu, Jie, et al. “You only segment once: Towards real-time panoptic segmentation.” *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023
42. Zhang, Chaoning, et al. “Faster segment anything: Towards lightweight sam for mobile applications.” *arXiv preprint [arXiv:2306.14289](https://arxiv.org/abs/2306.14289)* (2023)



Enhanced Brain Tumor Segmentation Using Preprocessing Techniques and 3D U-Net

Abdelrahman Telib^(✉) and Mohamed Gabr

German University in Cairo, New Cairo, Egypt
abdelrahman.telib@gmail.com

Abstract. Accurate brain tumor segmentation in magnetic resonance imaging (MRI) scans is crucial for diagnosis and treatment planning. This paper presents an enhanced approach to brain tumor segmentation by combining unsharp masking, normalization, and histogram equalization with a 3D U-Net architecture. When compared to several state-of-the-art techniques, our strategy considerably improves the Dice Score for Whole Tumor (WT), Enhancing Tumor (ET), and Tumor Core (TC) regions. In particular, we obtain Dice Scores of 91.84% for WT, 84.58% for ET, and 85.00% for TC on the BraTS2020 dataset. The MRI images quality is enhanced by the suggested preprocessing techniques, which helps the model train and make more accurate predictions. Our approach shows significant gains in segmentation accuracy, especially in the difficult Enhancing Tumor region. These findings support the efficacy of our method and offer significant improvements over the state-of-the-art methods for brain tumor segmentation. Better contrast and feature improvement in the images are also a result of the preprocessing stages of normalization, unsharp masking, and histogram equalization, which boost model performance. All things considered, our research validates the usefulness of the suggested approach in clinical settings, providing a solid brain tumor segmentation solution that can greatly facilitate patient diagnosis and treatment planning.

Keywords: Brain Tumor Segmentation · 3D U-Net · Histogram Equalization

1 Introduction

In medical image analysis, brain tumor segmentation from MRI images is essential because it helps with brain tumor diagnosis, therapy planning, and monitoring. The size, location, and type of brain tumors are critical determinants in patient prognosis and therapy, and accurate segmentation of these tumors is necessary to ascertain these details. Nevertheless, brain tumor manual segmentation is a labor-intensive, time-consuming procedure that is subject to inter-observer variability. As a result, the demand for automated and trustworthy techniques for brain tumor segmentation is rising.

Deep learning methods, in particular convolutional neural networks (CNNs), have demonstrated considerable promise in medical picture segmentation tasks in recent years. The introduction of topologies like U-Net and its variations has resulted in notable advancements in accuracy of segmentation. Nevertheless, a number of obstacles still need to be overcome before these models can be applied to brain tumor segmentation. These include the unevenness of the tumor and background regions, the variability of tumor appearance, and the existence of noise and artifacts in MRI images.

A class of models known as transformers was first created for natural language processing applications, but because of their capacity to capture contextual information and long-range dependencies, they have lately been modified for use in medical picture analysis. Transformers have great potential, but the amount of processing power and training data they require can be a major obstacle to their practical application. Transformers are less practical for application in many clinical settings where access to high-performance computing resources may be limited due to their high processing requirements.

In order to overcome these obstacles, we suggest a strategy that makes use of strong CNN architecture and effective preprocessing methods to enhance brain tumor segmentation performance without requiring the high processing overhead of transformers. Our method involves improving the quality of MRI images using histogram equalization, unsharp masking and normalization before feeding them into a 3D U-Net model. These preprocessing techniques aid in enhancing the contrast and emphasizing significant details in the photos, which helps the model learn and predict more accurately.

The main contributions of this paper are as follows:

- Introduction of a preprocessing pipeline that includes histogram equalization, unsharp masking, and normalization to enhance MRI images for better segmentation.
- Implementation of a 3D U-Net architecture optimized for brain tumor segmentation.
- Thorough testing of the suggested approach using the BraTS2020 dataset, demonstrating significant improvements over state-of-the-art methods.

The rest of this paper is organized as follows: Sect. 2 provides an overview of related work in brain tumor segmentation. Section 3 describes the preprocessing techniques and the network architecture used in our method. Section 4 presents the experimental setup, evaluation metrics and discusses the results and compares our method with existing approaches. Finally, Sect. 5 concludes the paper and outlines potential directions for future research.

2 Related Works

Numerous studies on brain tumor segmentation are included in this part, with a focus on various methodologies and their results. The next papers provide an in-depth examination of techniques and their efficacy in brain tumor segmentation using MRI data.

Ullah et al. [25] explored the impact of various pre-processing techniques on brain MRI enhancement specifically for tumor segmentation using a 3D U-Net architecture. Their study demonstrated that applying Gibbs ringing artifact removal significantly improves segmentation accuracy. The work by Ullah et al. underlines the importance of a comprehensive pre-processing framework, similar to the approach proposed in our study, where multiple techniques are combined to enhance the input data quality and consequently the performance of deep learning models for brain tumor segmentation.

Messaoudi et al. [23] include the EfficientNet model in the encoding branch for 3D brain tumor segmentation, hence proposing an asymmetric U-Net. Futrega et al. [24] conducted in-depth ablation research on various components and training regimens to optimize U-Net architecture for brain tumor segmentation; they were successful in the validation phase and placed third in the BraTS21 challenge.

Peng et al. [17] publish Multi-Scale 3D U-Nets for automatic brain tumor segmentation, which leverages multi-scale feature extraction to increase segmentation performance. To create an end-to-end brain tumor segmentation system using multi-inception-UNET, Latif and colleagues [18] concentrated on combining inception modules to record various receptive fields. Chandra et al. [19] created the Contextual Efficient Capsule Network, which employs capsule networks to record spatial hierarchies in brain tumor segmentation.

Combining deep learning with semi-supervised learning techniques improved the accuracy of brain tumor segmentation, according to Mlynarski et al. [20]. Zhou et al. [21] integrated features at different scales for brain tumor segmentation using a multi-scale fusion convolutional neural network. Liu et al. [22] introduced a deep convolutional neural network for brain tumor segmentation using multi-modality MRI data, with the goal of improving the segmentation accuracy of different tumor locations.

Qamar et al. [9] presented HI-Net, a hyperdense inception 3D UNet that use factorized convolutional layers and dense connections to gather multi-scale information. It was confirmed with notable performance gains on the BRATS 2020 dataset. Islam et al. [10] developed a 3D attention UNet that combines radiomic and clinical characteristics with channel and spatial attention mechanisms for enhanced segmentation, utilizing machine learning techniques to predict survival.

Sinha et al. [11] introduced a memory-efficient cascade 3D UNet for brain tumor segmentation that reduces memory usage without compromising segmentation accuracy. Fang et al. [12] describe a transformer-based model for brain tumor segmentation that makes advantage of self-attention mechanisms.

Chen et al. [13] looked into anisotropic diffusion-based unsharp masking for denoising and MRI image enhancement, and they found improved segmentation results. Han et al. [14] created a novel feature improvement framework that blends multi-scale feature extraction with an attention mechanism to improve brain tumor segmentation accuracy.

Kaur et al. [15] developed a multi-scale lightweight 3D segmentation approach with an attention mechanism for brain tumor segmentation with the goal

of focusing on both computational efficiency and accuracy. Zhang et al. [16] demonstrated the significance of attention techniques in improving model performance by putting forth a 3D convolutional neural network with an attention mechanism for better brain tumor segmentation.

Eman Sami et al. [6] present a comparison study of threshold segmentation algorithms and find that k-means clustering outperforms other techniques on the TCIA dataset in terms of RMSE, PSNR, and segmentation accuracy.

Liu et al. [5] offer a multiscale lightweight 3D segmentation method with an attention mechanism, demonstrating significant improvements in computation efficiency and segmentation accuracy.

Tahir et al. [7] provide a feature enhancement framework for brain tumor segmentation and classification that incorporates advanced segmentation techniques, contrast enhancement, and noise reduction in order to significantly increase accuracy.

Ajai and Gopalan [8] propose an eight-direction Sobel edge detection algorithm and demonstrate its superiority over traditional methods in detecting irregular tumor edges with reduced error metrics and higher accuracy.

The collective findings of these studies show the advancements in MRI image preprocessing and brain tumor segmentation techniques. Each method improves the consistency and precision of brain tumor segmentation in clinical settings, and each has advantages of its own.

3 Methods

3.1 Preprocessing Techniques

To improve the quality and standardize the inputs, we used multiple strategies to preprocess the MRI images for brain tumor segmentation. Histogram equalization, unsharp masking, and normalization with rescaling are some of the techniques employed. A thorough explanation of each strategy is provided below.

Histogram Equalization and Rescaling. To enhance the contrast of the photographs, each slice of a 3D volume tensor is subjected to histogram equalization. To keep intensity levels consistent, the equalized slices are rescaled to the original range. The definition of the transformation is as follows:

Given a 3D volume tensor V with slices S_i , the transformation can be expressed as:

$$S'_i = \text{rescale}(\text{equalize}(S_i), \min(S_i), \max(S_i)) \quad (1)$$

where $\text{equalize}(S_i)$ is the histogram equalized slice, and $\text{rescale}(S'_i, \min(S_i), \max(S_i))$ scales the slice back to its original intensity range.

Unsharp Masking. By processing each 2D slice individually along a predetermined axis, unsharp masking is used to improve edge contrast in 3D medical pictures. Each slice is subjected to Gaussian blur, the unsharp mask is computed, and the original slice is enhanced by the addition of the scaled mask. Mathematically, the procedure is expressed for a slice S_i as follows:

$$S'_i = S_i + \alpha(S_i - \text{GaussianBlur}(S_i, \sigma)) \quad (2)$$

where α is the strength factor, σ is the standard deviation for the Gaussian kernel, and $\text{GaussianBlur}(S_i, \sigma)$ is the blurred slice.

Normalization and Rescaling. By using normalization, each slice's intensity levels are guaranteed to fall inside a predetermined range. First, the values must be scaled to a range of $[0, 1]$, and then they must be rescaled to a desired range of $[\text{min_val}, \text{max_val}]$. The normalized and rescaled slice S'_i for a slice S_i is calculated as follows:

$$S'_i = \text{rescale} \left(\frac{S_i - \min(S_i)}{\max(S_i) - \min(S_i)}, \text{min_val}, \text{max_val} \right) \quad (3)$$

This transformation ensures uniform intensity distribution across the dataset, facilitating better training of the segmentation model.

3.2 Network Architecture

We used a 3D U-Net model for the brain tumor segmentation task. Because of its encoder-decoder layout, which allows it to collect both local and global information, the U-Net architecture is highly suited for biomedical picture segmentation. The following describes our U-Net model's exact configuration:

The U-Net model is defined with the following parameters:

- **Spatial dimensions:** 3D
- **Input channels:** 4 (corresponding to the different MRI modalities)
- **Output channels:** 4 (corresponding to the segmented regions: whole tumor, enhancing tumor, tumor core, and background)
- **Channels:** (16, 32, 64, 128) - indicating the number of feature maps at each layer in the encoder
- **Strides:** (1, 2, 2, 2) - specifying the downsampling factors at each layer
- **Kernel size:** 3 - the size of the convolutional kernels used throughout the network

The model architecture is illustrated in Fig. 1.

The U-Net model consists of an encoder path, where the input is progressively downsampled to capture context, and a decoder path, where the feature maps are upsampled and combined with corresponding feature maps from the encoder path through skip connections. This structure allows the network to effectively learn and predict segmentation maps with high accuracy.

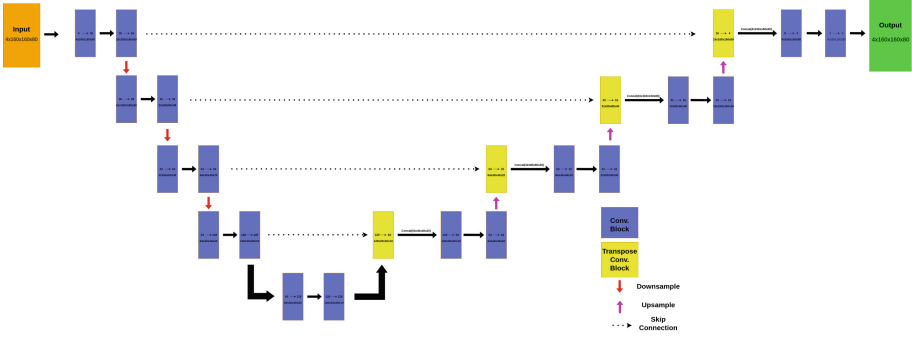


Fig. 1. The architecture of the 3D U-Net model used for brain tumor segmentation.

3.3 Loss Function

To train the U-Net model for brain tumor segmentation, we employed the Dice Loss function. The Dice Loss is particularly effective for segmentation tasks as it directly optimizes the Dice coefficient, a measure of overlap between the predicted segmentation and the ground truth.

The Dice coefficient D is defined as:

$$D = \frac{2|P \cap T|}{|P| + |T|} \tag{4}$$

where $|P \cap T|$ is the intersection of the predicted output P and the target T , $|P|$ is the sum of the predicted output, and $|T|$ is the sum of the target. The Dice coefficient ranges from 0 to 1, where 1 indicates perfect overlap.

However, to make this differentiable and suitable for optimization, we modify it slightly and introduce a smoothing term ϵ to prevent division by zero. The modified Dice coefficient is given by:

$$D = \frac{2 \sum_i (P_i T_i) + \epsilon}{\sum_i P_i + \sum_i T_i + \epsilon} \tag{5}$$

where the summation \sum_i is over all the elements in the predicted and target tensors, and ϵ is a small constant (typically 1×10^{-6}).

The Dice Loss L is then defined as:

$$L = 1 - D \tag{6}$$

This formulation ensures that minimizing the Dice Loss maximizes the Dice coefficient, thereby improving the overlap between the predicted and target segmentations. The Dice Loss effectively handles the class imbalance often present in medical imaging datasets, making it well-suited for brain tumor segmentation tasks.

4 Experiments

4.1 Dataset

The BraTS2020 dataset, a large collection of multi-modal MRI scans used for brain tumor segmentation, was employed for our investigations. Four distinct MRI modalities—T1-weighted, post-contrast T1-weighted (T1ce), T2-weighted, and Fluid Attenuated Inversion Recovery (FLAIR)—are included in the annotated data of the BraTS2020 dataset for every patient. The complementary information provided by these modalities is essential for precise tumor segmentation.

Dataset Details. The BraTS2020 dataset contains 369 cases with high-grade gliomas (HGG) and low-grade gliomas (LGG) annotated by experts. Each MRI scan in the dataset is provided as a 3D volume with a size of $240 \times 240 \times 155$ voxels. The dataset includes annotations for three tumor subregions:

- **Whole Tumor (WT):** Includes all tumor regions.
- **Tumor Core (TC):** Excludes the edema region.
- **Enhancing Tumor (ET):** Includes only the active tumor regions that show up brightly on the T1Gd sequence.

Challenges. The BraTS2020 dataset presents several challenges:

- **Class imbalance:** The tumor regions occupy a much smaller volume compared to the background, making it challenging to accurately segment the tumor.
- **Variability:** Significant variability in tumor appearance across different patients and MRI modalities.
- **Artifacts:** Presence of noise and artifacts in the MRI scans that can affect the segmentation accuracy.

Despite these challenges, the BraTS2020 dataset remains a valuable resource for developing and evaluating advanced brain tumor segmentation algorithms due to its rich diversity and comprehensive annotations.

4.2 Evaluation Metrics

The performance of the segmentation model was evaluated using the Dice Similarity Coefficient (DSC), a commonly used metric for image segmentation tasks. We calculated the Dice coefficients for three specific regions:

- **Whole Tumor (WT):**

$$DSC_{\text{whole}} = \frac{2 \sum_i (P_i^{\text{WT}} \cdot T_i^{\text{WT}})}{\sum_i P_i^{\text{WT}} + \sum_i T_i^{\text{WT}}} \quad (7)$$

– **Enhancing Tumor (ET):**

$$DSC_{\text{enhancing}} = \frac{2 \sum_i (P_i^{\text{ET}} \cdot T_i^{\text{ET}})}{\sum_i P_i^{\text{ET}} + \sum_i T_i^{\text{ET}}} \quad (8)$$

– **Tumor Core (TC):**

$$DSC_{\text{core}} = \frac{2 \sum_i (P_i^{\text{TC}} \cdot T_i^{\text{TC}})}{\sum_i P_i^{\text{TC}} + \sum_i T_i^{\text{TC}}} \quad (9)$$

A custom Dice metric class was used to aggregate these metrics over the entire validation dataset, providing a comprehensive evaluation of the model’s performance.

4.3 Training Details

The U-Net model was trained using the Dice Loss function, which directly optimizes the Dice coefficient. The Dice Loss L is defined as in (6).

This formulation ensures that minimizing the Dice Loss maximizes the Dice coefficient, thereby improving the overlap between the predicted and target segmentations.

The training process was carried out with the following details:

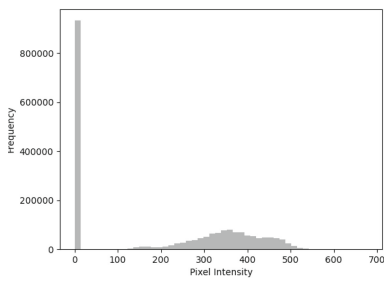
- **Model:** 3D U-Net with spatial dimensions of 3, input channels of 4, output channels of 4, channels of (16, 32, 64, 128), and strides of (1, 2, 2, 2).
- **Loss Function:** Dice Loss.
- **Optimizer:** Adam optimizer with a learning rate of 1×10^{-4} .
- **Epochs:** The model was trained for 100 epochs.
- **Data Augmentation:** Random transformations such as rotation, scaling, and flipping were applied to increase the diversity of the training data.
- **Validation:** The model’s performance was evaluated on a separate validation set at the end of each epoch.

During training, the Dice coefficients for the whole tumor, enhancing tumor, and tumor core were monitored to ensure the model’s effectiveness across different tumor regions.

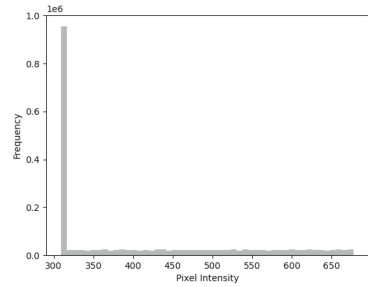
4.4 Visualizations

Understanding the consequences of preprocessing methods and the segmentation model’s performance depends heavily on visualizations. This section includes a number of visualizations that show how the final segmentation results, unsharp masking, and histogram equalization affect each other.

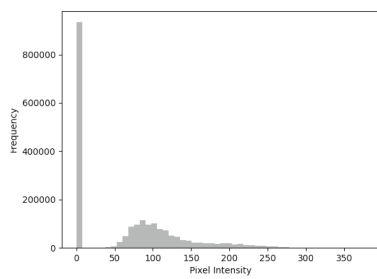
Histogram Equalization. Figures 2, 3, 4 and 5 show the comparison of MRI slices before and after applying histogram equalization in the four modalities. Histogram equalization enhances the contrast of the images by redistributing the intensity values, which can improve the visibility of structures within the brain.



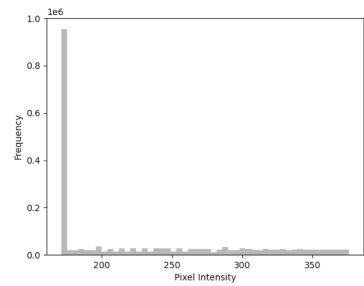
(a) Before Histogram Equalization



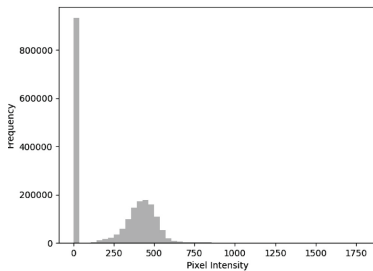
(b) After Histogram Equalization

Fig. 2. Comparison of MRI slices T1 Modality before and after histogram equalization.

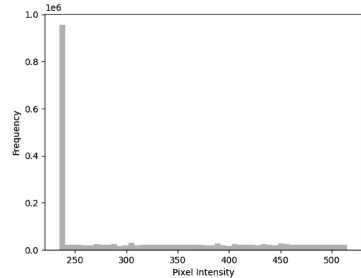
(a) Before Histogram Equalization



(b) After Histogram Equalization

Fig. 3. Comparison of MRI slices T2 Modality before and after histogram equalization.

(a) Before Histogram Equalization



(b) After Histogram Equalization

Fig. 4. Comparison of MRI slices T1ce Modality before and after histogram equalization.

Combined Preprocessing Effects. Figures 6, 7, 8 and 9 demonstrates the effects of combined preprocessing techniques. It includes MRI slices for the four modalities before preprocessing, after histogram equalization, and finally after applying 3D unsharp masking. The combined preprocessing steps enhance the image quality and highlight important features, aiding in better segmentation.

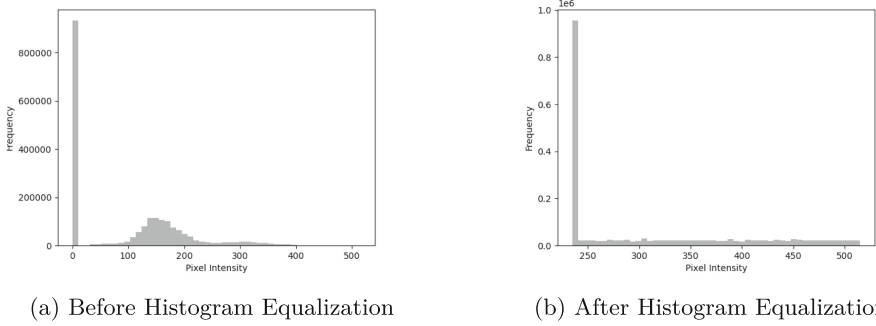


Fig. 5. Comparison of MRI slices Flair Modality before and after histogram equalization.

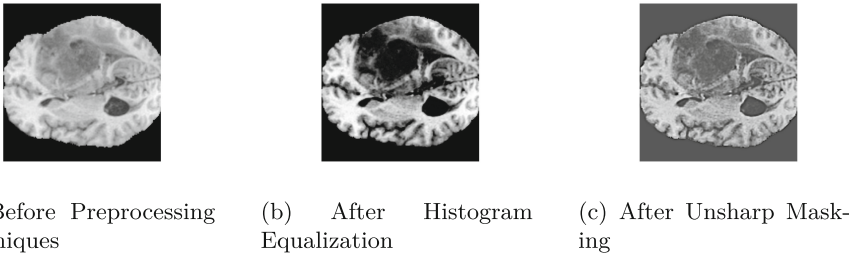


Fig. 6. Comparison of MRI slices T1 Modality before preprocessing, after histogram equalization, and after applying 3D unsharp masking.

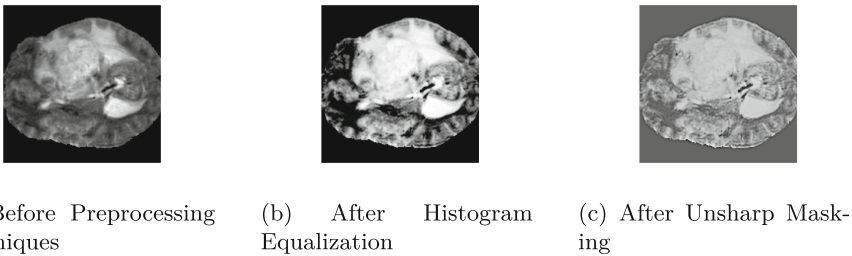
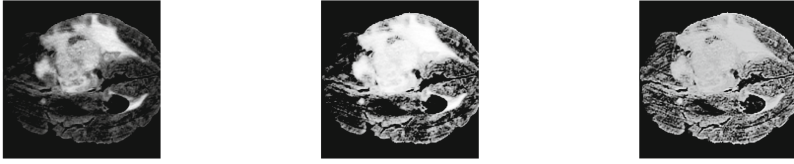


Fig. 7. Comparison of MRI slices T2 Modality before preprocessing, after histogram equalization, and after applying 3D unsharp masking.

Segmentation Results. Figure 10 presents the ground truth segmentation and the corresponding prediction by the U-Net model. These visualizations allow us to evaluate the accuracy of the model in identifying and segmenting the tumor regions.

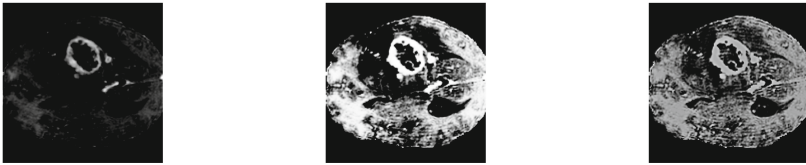


(a) Before Preprocessing Techniques

(b) After Histogram Equalization

(c) After Unsharp Masking

Fig. 8. Comparison of MRI slices Flair Modality before preprocessing, after histogram equalization, and after applying 3D unsharp masking.



(a) Before Preprocessing Techniques

(b) After Histogram Equalization

(c) After Unsharp Masking

Fig. 9. Comparison of MRI slices T1ce Modality before preprocessing, after histogram equalization, and after applying 3D unsharp masking.



(a) Ground Truth Segmentation

(b) Predicted Segmentation

Fig. 10. Comparison of ground truth segmentation and predicted segmentation by the U-Net model.

4.5 Results

We compared our method's performance with several state-of-the-art methods for brain tumor segmentation based on Dice Score for Whole Tumor (WT),

Table 1. Comparison of Dice Scores with selected methods on Brats2020

Model	Whole Tumor (%)	Enhancing Tumor (%)	Tumor Core (%)
nnU-Net [1]	91.07	81.37	87.97
H2NF-Net [2]	91.3	78.8	85.5
nnUnet Ensemble [3]	91.0	77.6	84.4
dResU-Net [4]	86.6	80.04	83.57
ADHDC-Net [7]	89.75	78.01	83.31
Our Method	91.84	84.58	85.00

Tumor Core (TC), and Enhancing Tumor (ET) regions. Table 1 shows the comparison results.

The results in Table 1 demonstrate that our method outperforms several state-of-the-art approaches in terms of Dice Score for brain tumor segmentation on Brats2020. Specifically, our method achieves a Dice Score of 91.84% for the Whole Tumor (WT), 84.58% for the Enhancing Tumor (ET), and 85.00% for the Tumor Core (TC). Compared to nnU-Net [1], which achieves 91.07% for WT, 81.37% for ET, and 87.97% for TC, our method shows a significant improvement in the Enhancing Tumor region. Similarly, H2NF-Net [2] and nnUnet Ensemble [3] also demonstrate lower performance in the Enhancing Tumor region compared to our method.

Our method shows how to improve MRI images prior to segmentation by applying unsharp masking, normalizing, and histogram equalization. These preprocessing techniques aid in enhancing the contrast and emphasizing significant details in the photos, which helps the model learn and predict more accurately. Overall, the findings indicate that our technique to brain tumor segmentation is highly competitive and offers notable advantages over current state-of-the-art methods.

5 Conclusion

In this work, we provided an improved approach to brain tumor segmentation that combines a solid 3D U-Net architecture with effective preprocessing methods. Our method involves enhancing MRI images with unsharp masking, normalization, and histogram equalization. This greatly improves contrast and highlights key characteristics for more accurate segmentation. We show that our method outperforms various state-of-the-art approaches with the experimental findings on the BraTS2020 dataset, attaining Dice Scores of 91.84% for Whole Tumor (WT), 84.58% for Enhancing Tumor (ET), and 85.00% for Tumor Core (TC).

The noteworthy enhancements in the segmentation accuracy, namely in the Enhancing Tumor area, highlight the efficacy of our preprocessing procedure. Our approach improves the quality of the input images, which helps the 3D

U-Net model train and predict better, which results in higher segmentation performance.

In order to increase segmentation accuracy, future work will concentrate on refining preprocessing methods and investigating the incorporation of cutting-edge deep learning models. To confirm the generalizability and robustness of our method, we also intend to apply it to other medical imaging modalities and segmentation tasks.

All things considered, our suggested approach offers a viable means of precisely and effectively segmenting brain tumors, and it may considerably help with clinical diagnosis and treatment planning.

References

1. Isensee, F., Jaeger, P.F., Full, P.M., Vollmuth, P., Maier-Hein, K.H.: nnU-Net for Brain Tumor Segmentation (2020). [arXiv:2011.00848](https://arxiv.org/abs/2011.00848)
2. Jia, H., Cai, W., Huang, H., Xia, Y.: H2NF-Net for Brain Tumor Segmentation Using Multimodal MR Imaging: 2nd Place Solution to BRATS Challenge 2020 Segmentation Task (2020). [arXiv:2012.15318](https://arxiv.org/abs/2012.15318)
3. Fidon, L., Ourselin, S., Vercauteren, T.: Generalized Wasserstein dice score, distributionally robust deep learning, and ranger for brain tumor segmentation: BraTS 2020 challenge. In: Crimi, A., Bakas, S. (eds.) BrainLes 2020. LNCS, vol. 12659, pp. 200–214. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-72087-2_18
4. Raza, R., Bajwa, U.I., Mehmood, Y., Anwar, M.W., Jamal, M.H.: DRESU-net: 3D deep residual U-net based brain tumor segmentation from multimodal MRI. *Biomed. Sig. Process. Control* **79**, 103861 (2023)
5. Liu, H., Huo, G., Li, Q., Guan, X., Tseng, M.-L.: Multiscale lightweight 3D segmentation algorithm with attention mechanism: brain tumor image segmentation (2023)
6. Sami, E., et al.: Brain Tumor Segmentation: A Comparative Analysis (2021)
7. Tahir, M., et al.: Feature enhancement framework for brain tumor segmentation and classification. *Microsc. Res. Tech.* **82**(6), 741–749 (2019). <https://doi.org/10.1002/jemt.23219>
8. Ajai, R., Gopalan, S.: Comparative analysis of eight direction Sobel edge detection algorithm for brain tumor MRI images. *Procedia Comput. Sci.* **201**, 487–494 (2022). <https://doi.org/10.1016/j.procs.2022.03.063>
9. Qamar, S., Ashraf, H., Khan, M.A., Siddiqui, L.: HI-net: hyperdense inception 3D UNet for brain tumor segmentation. In: BRATS 2020 Challenge (2020)
10. Islam, M., Zhang, Y., Feng, F., et al.: Brain Tumor Segmentation and Survival Prediction Using 3D Attention UNet. *arXiv preprint arXiv:2012.04653* (2020)
11. Cheng, X., Jiang, Z., Sun, Q., Zhang, J.: Memory-efficient cascade 3D U-net for brain tumor segmentation. In: Crimi, A., Bakas, S. (eds.) BrainLes 2019. LNCS, vol. 11992, pp. 242–253. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-46640-4_23
12. Fang, L., Yu, K., Chen, X., et al.: Brain Tumor Segmentation using Transformer. *arXiv preprint arXiv:2104.00985* (2021)
13. Chen, S., Li, X., Liu, Y.: Anisotropic diffusion based unsharp masking and crispening for denoising and enhancement of MRI images. *Sensors* **19**(11), 2501 (2019)
14. Han, X., Liu, H., Wu, J., et al.: Feature enhancement framework for brain tumor segmentation and classification. *Future Gener. Comput. Syst.* **89**, 180–191 (2018)

15. Kaur, H., Huo, G., Tseng, M.L.: Multiscale Lightweight 3D Segmentation Algorithm with Attention Mechanism: Brain Tumor Image Segmentation. arXiv preprint [arXiv:2012.06760](https://arxiv.org/abs/2012.06760) (2023)
16. Zhang, J., Cheng, W., Liu, G.: Attention mechanism in 3D convolutional neural network for brain tumor segmentation. *Sensors* **20**(19), 5281 (2020)
17. Peng, Y., et al.: Multi-scale 3D U-nets: an approach to automatic segmentation of brain tumor. *Int. J. Imaging Syst. Technol.* **29**(1), 3–11 (2019)
18. Latif, A., et al.: An end-to-end brain tumor segmentation system using multi-inception-UNET. *Int. J. Imaging Syst. Technol.* **31**(1), 70–80 (2021)
19. Chandra, S., et al.: Contextual Efficient Capsule Network for brain tumor segmentation. arXiv preprint [arXiv:2012.06760](https://arxiv.org/abs/2012.06760) (2020)
20. Mlynarski, P., et al.: Deep Learning with semi-supervised learning techniques for brain tumor segmentation. arXiv preprint [arXiv:2011.01045](https://arxiv.org/abs/2011.01045) (2020)
21. Zhou, Y., et al.: Multi-scale fusion convolutional neural network for brain tumor segmentation. *Sensors* **21**(21), 7528 (2021)
22. Liu, Z., et al.: Deep convolutional neural network for brain tumor segmentation using multi-modality MRI data. *Electronics* **9**(12), 2203 (2020)
23. Messaoudi, H., et al.: Efficient embedding network for 3D brain tumor segmentation. arXiv preprint [arXiv:2011.11052](https://arxiv.org/abs/2011.11052) (2020)
24. Futrega, M., et al.: Optimized U-Net for Brain Tumor Segmentation. arXiv preprint [arXiv:2110.03352](https://arxiv.org/abs/2110.03352) (2021)
25. Ullah, F., et al.: Brain MR image enhancement for tumor segmentation using 3D U-net. *Sensors* **21**(22), 7528 (2021)



Joint Top-Down and Bottom-Up Frameworks for 3D Visual Grounding

Yang Liu^(✉), Daizong Liu, and Wei Hu

Wangxuan Institute of Computer Technology, Peking University, Beijing, China
{20181iuyang,forhuwei}@pku.edu.cn, dzliu@stu.pku.edu.cn

Abstract. This paper tackles the challenging task of 3D visual grounding—locating a specific object in a 3D point cloud scene based on text descriptions. Existing methods fall into two categories: top-down and bottom-up methods. Top-down methods rely on a pre-trained 3D detector to generate and select the best bounding box, resulting in time-consuming processes. Bottom-up methods directly regress object bounding boxes with coarse-grained features, producing worse results. To combine their strengths while addressing their limitations, we propose a joint top-down and bottom-up framework, aiming to enhance the performance while improving the efficiency. Specifically, in the first stage, we propose a bottom-up based proposal generation module, which utilizes lightweight neural layers to efficiently regress and cluster several coarse object proposals instead of using a complex 3D detector. Then, in the second stage, we introduce a top-down based proposal consolidation module, which utilizes graph design to effectively aggregate and propagate the query-related object contexts among the generated proposals for further refinement. By jointly training these two modules, we can avoid the inherent drawbacks of the complex proposals in the top-down framework and the coarse proposals in the bottom-up framework. Experimental results on the ScanRefer benchmark show that our framework is able to achieve the state-of-the-art performance.

Keywords: 3D visual grounding · top-down · bottom-up

1 Introduction

The 3D visual grounding (3DVG) [2, 12] is a fundamental yet important task in 3D understanding, which has recently received increasing attention due to its wide range of applications, such as in robotics and AR/VR systems. The goal of this task is to locate the target object in a 3D point cloud scene based on a given free-form query text description. Different from previous mature 2D grounding methods [5, 10, 24, 27, 28, 30–32], 3D visual grounding has two more challenging

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-78113-1_17.

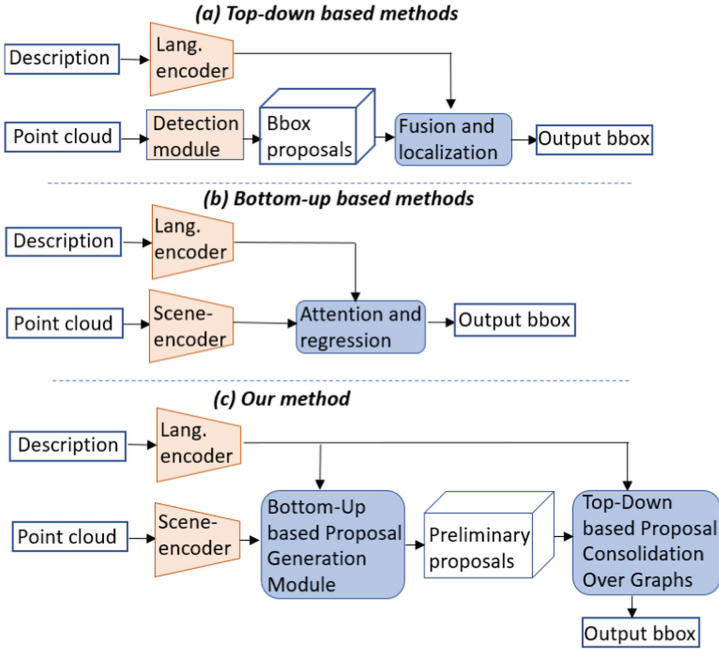


Fig. 1. (a): Typical procedure of top-down based method. (b): Typical procedure of bottom-up based method. (c): Procedure of our proposed method, where we generate initial proposals in an efficient bottom-up manner, and subsequently consolidate the proposals over graphs via an effective top-down approach.

aspects: Firstly, 3D visual grounding takes sparse, noisy, and information-dense 3D point clouds as input, making it more difficult to obtain visual information. Secondly, the object-to-object and object-to-scene relationships in 3D space are more complex than that in 2D images, further increasing the difficulty of 3D visual grounding task.

Existing methods for the 3D visual grounding (3DVG) task can mainly be grouped into two categories according to their model designs: (1) Top-down approaches: these methods [1, 2, 6, 8, 14, 33, 34] typically first utilize pre-trained 3D object detection models [4, 15, 18, 19, 26] or segmenter [3, 9] to generate a large number of candidate proposals in the entire point cloud scene, and then select the one that best matches the semantic meaning of the query text. Although they are able to obtain high-quality proposals, they need to enumerate all possible objects to ensure that the generated proposals contain the object required by the text, leading to a large number of redundant proposals. The general procedure of the top-down based methods is illustrated in Fig. 1(a). (2) Bottom-up approaches: these methods [7, 13, 17] generally first interact the point set with textual description via early-fusion strategy, then directly predict the target bounding box from the learned query-related point-wise features. Compared to the top-down methods, they do not rely on complex proposal generation and

selection, and thus can achieve end-to-end efficient training. The general procedure of the bottom-up based methods is illustrated in Fig. 1(b).

However, the above two types of methods have their own advantages and disadvantages. For the top-down approaches, this category of methods can typically yield high-quality proposals and facilitate the better capture of relationship information among a large number of proposals. However, its disadvantage is that in order to ensure that the generated proposals contain the proposal required by the query text, a large number of proposals need to be generated in abundance, while the vast majority of the generated proposals are not related to the description in the query text, which reduces the efficiency of the method. However, if the number of generated proposals is reduced, there is a higher risk of ignoring the object required by the query. For the bottom-up approaches, because they avoid the issue of generating and processing an excessive number of redundant proposals, higher computational efficiency is achievable. At the same time, since the bounding box is directly regressed and not limited to the proposals generated by the pre-trained model, this category of methods can capture smaller objects that may be easily neglected by the pre-trained model. However, they overlook the rich information between the global points as they struggle to model object-level interactions. Therefore, their predicted object proposals obtained are relatively coarse, and there is no additional design for further proposal refinement.

Through the analysis above, we find that the advantages and disadvantages of these two methods actually complement each other. Specifically, bottom-up approaches can generate a few proposals that closely relate to the query text, thereby reducing redundancy caused by top-down approaches and significantly improving computational efficiency. On the other hand, top-down approaches can alleviate the issue of coarse proposals generated by bottom-up approaches by capturing the relationship information among proposals and refining them.

This inspires us to propose a method that integrates the advantages of both approaches while mitigating their limitations. Our proposed method consists of two stages. In the first stage, we develop a *proposal generation module* similar to bottom-up methods. This module aggregates from both 3D point clouds and query texts, enabling us to predict bounding boxes for objects highly relevant to the query text directly based on these features, and extract the corresponding object features. This enables us to use the guidance of the query text information to avoid the inefficiency caused by detecting and analyzing a large number of redundant objects simultaneously, and to identify objects that may be overlooked by pre-trained detectors. In the second stage, we address the issue of rough proposals generated by bottom-up methods by developing a *graph-based proposal consolidation module* inspired by top-down methods. This module further captures object-to-object and object-to-scene relationships and updates the features of objects accordingly. Subsequently, we generate more refined bounding boxes for the objects based on these updated features. We conducted evaluations on the commonly adopted ScanRefer [2] datasets, and the experimental results demonstrated that our proposed method achieved the state-of-the-art performance when compared to existing methods.

In summary, the contributions of our work are:

1. We conduct an in-depth analysis of the strengths and weaknesses of existing top-down and bottom-up-based methods for 3D visual grounding. Through this analysis, we provide valuable insights into how to leverage their respective advantages while mitigating their limitations effectively.
2. We propose a novel framework, which first develops a bottom-up based strategy to generate a few object proposals and then devises a top-down based strategy over graphs to consolidate and refine the proposals for final grounding.
3. Through comprehensive experiments, we demonstrate the effectiveness of our proposed method and shed light on the rationale behind the successful integration of bottom-up and top-down approaches.

2 Related Work

Top-Down Based 3D Visual Grounding. Most approaches for 3DVG are top-down based. For example, ScanRefer [2] utilizes VoteNet [19] to extract numerous proposals and combine their features with textual features to select the matched proposal. Subsequently, several top-down approaches emerged. TGNN [8] generates candidate proposals as graph nodes, leveraging object features and relationships to generate attention heatmaps for sentence expressions. InstanceRefer [34] uses a language model to determine the target object category and identifies instances with the same category in the scene as candidates. The final instance is chosen through a matching process. SAT [33] enhances understanding of 3D scenes by learning alignments between 2D object representations and corresponding objects in 3D scenes. While these methods aim to include desired proposals specified by textual requirements, generating a large number of proposals often leads to inefficiencies. Besides, reducing generated proposals increases the risk of neglecting query-required objects.

Bottom-Up Based 3D Visual Grounding. To address the challenge of generating numerous irrelevant proposals in top-down methods, bottom-up method 3D-SPS [17] is proposed to progressively select key points based on language guidance and directly regresses the bounding box. Similarly, Refer-it-in-RGBD [13] first constructs a confidence heat map from the input sentence and voxels, then samples seed points according to the heat map, and regresses the object's bounding box. However, these bottom-up methods often produce coarse proposals, limiting their ability to exploit complementary information among different bounding boxes for refinement.

3 The Proposed Method

Previous works generally follow a top-down or bottom-up framework, both of which come with inherent limitations within their respective designs. In this

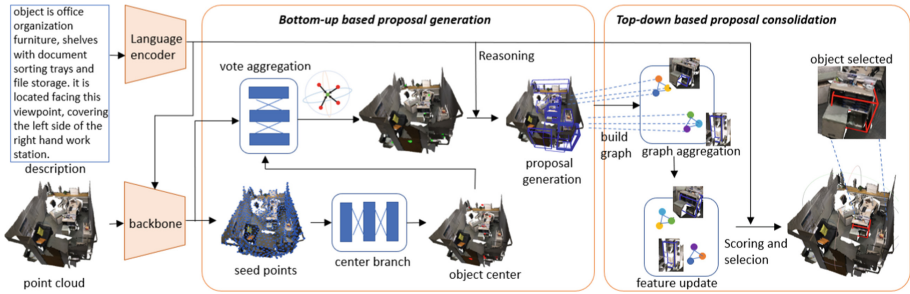


Fig. 2. The pipeline of our proposed method. Initially, we encode the input 3D point cloud and text with pre-trained encoders. In the bottom-up stage, our module fuses these features for language-guided object proposals. In the top-down stage, our refinement module enhances these proposals by graph-based features, followed by predicting matching scores to select the best-matching bounding box.

paper, we propose a novel approach that leverages the strengths of both top-down and bottom-up frameworks while mitigating their individual limitations through a unified structure. In this section, we provide a comprehensive description of our method. We begin by offering an overview of the 3D visual grounding task and our proposed framework. Then, we describe the multi-modal encoders used in our method. After that, we elaborate on our proposed bottom-up based 3D proposal generation module and top-down based 3D proposal consolidation module respectively. At last, we present the training objectives of our method.

3.1 Overview

Notation Definition. We first define 3D visual grounding task as follows. Given point clouds $\mathbf{P} \in \mathbb{R}^{N \times (3+F)}$ and free-form language query text $\mathbf{D} = \{w_n\}_{n=1}^W$, where N is the number of the points, F is the dimension of the additional features of the point clouds such as colors and normals, and W denotes the number of the input words. Our task is to predict the 3D bounding box of the object that matches the input description.

Overall Pipeline. As illustrated in Fig. 2, our proposed method consists of two stages: the bottom-up stage and the top-down stage. Firstly, we use pre-trained encoders to independently encode the 3D point cloud and the query text information. In the bottom-up stage, we feed these two types of features into our proposed bottom-up based proposal generation module for feature fusion and updating, which yields language-guided object proposals. Subsequently, in the top-down stage, these proposals are input into our proposed top-down based proposal consolidation module for further refinement, resulting in improved proposals. Finally, we predict the matching scores between these proposals and the query language, selecting the bounding box that best matches the query language as the final output.

3.2 Preliminaries

3D Scene Encoder. There have been many works [20, 21, 23, 29] on encoding 3D point clouds, and theoretically, they can all be used for encoding the input 3D scene point clouds. For consistency with previous works on 3D visual grounding [2, 13, 33], we adopt the same pre-trained PointNet++ [21] for encoding the 3D point cloud information. Let $\mathbf{V} \in \mathbb{R}^{M_0 \times (3+C_v)}$ denote the output of PointNet++, where M_0 is the number of the seed points obtained by PointNet++ and C_v is the dimension of point features. Each point’s feature V_i can be divided into two parts, which represent its 3D coordinates $\mathbf{x}_i \in \mathbb{R}^3$ and other features $\mathbf{f}_i \in \mathbb{R}^{C_v}$.

Description Encoder. We use a pre-trained CLIP model [22] to encode the language information from the query text $\mathbf{D} = \{w_n\}_{n=1}^W$. The output of the text encoder is denoted as $\mathbf{L} \in \mathbb{R}^{W \times C_l}$, where C_l here represents the dimension of language features.

3.3 Bottom-Up Based Proposal Generation Module

To locate the object, previous top-down based methods directly utilize 3D detection models to produce all possible proposals of all objects in the 3D scene, which not only severely rely on the proposal quality but also result in low computational efficiency. Although some recent bottom-up based methods try to directly regress query-related proposals, they still rely on the complex decoding modules.

Therefore, we propose a simple yet lightweight bottom-up based proposal generation module that generates candidate proposals guided by the query text. This approach achieves higher efficiency compared to traditional top-down based methods and is simpler than previous bottom-up based approaches.

To achieve this goal, we first utilize a multi-modal transformer-based module [25] to fuse language and 3D visual information and eliminate points that are irrelevant to the given language query. By leveraging the features generated by the Transformer, we predict the center coordinates of the objects to which each point in the point cloud belongs. Subsequently, based on each center, we employ a vote-aggregation module to combine features of its neighboring points belonging to the same object. Using these aggregated features, we can regress language-guided object proposals.

Specifically, as for the multi-modal transformer-based module, we employ two separate self-attention layers to encode the contexts in 3D point cloud features and query language features, along with a cross-attention layer to encode the correspondences between the two modalities. The self-attention mechanism allows the model to capture relevant relationships and dependencies within each modality, while the cross-attention mechanism helps in aligning the language and visual features, enabling the model to focus on the most relevant information for the task of generating object proposals guided by the language query. The output of the transformer module is denoted as $\mathbf{V}^T \in \mathbb{R}^{M_0 \times C_l}$, and the computation process can be represented as follows:

$$\mathbf{V}^T = \text{Transformer}(\mathbf{V}, \mathbf{L}). \quad (1)$$

By examining the attention scores produced by the cross-attention mechanism, we can filter out points that have weak associations with the given query language. Denoting the attention scores as att_{lang} , we obtain the filtered set of points $\mathbf{V}^F \in \mathbb{R}^{M_f \times C_t}$ through the following computation:

$$\mathbf{V}^F = \mathbf{V}_t[\text{argtop}_k(\text{Mean}(att_{lang}), M_f)]. \quad (2)$$

Subsequently, for each of the points obtained above, we employ a center predictor based on its features \mathbf{V}^F to predict the object it belongs to. This process yields the coordinates of the center of the corresponding object for each point, denoted as $\mathbf{c}_f \in \mathbb{R}^{M_f \times 3}$. Here, \mathbf{c}_f represents the three-dimensional coordinates of M_f individual center points. The center predictor can be implemented using a Multi-Layer Perceptron (MLP). Then, to obtain object candidate regions from the points, we first employ farthest point sampling based on these center coordinates to obtain several candidate proposal centers $\{c_i\}_{i=1}^K$, where K is the number of candidates. Since points belonging to the same object have closely related center coordinates, farthest point sampling helps to select those points that belong to different objects. After obtaining the selected points using farthest point sampling, we perform max pooling on the points within a certain distance from each selected center. Let the set of points within a certain distance r from point c_i be denoted as $\{c_{ij}\}_{j=1}^n$ and their features denoted as $\{\mathbf{V}_{ij}^F\}_{j=1}^n$, the feature \mathbf{V}_i^G of point c_i can be computed as:

$$\mathbf{V}_i^G = \text{Maxpool}(\{\text{MLP}(\mathbf{V}_{ij}^F)\}_{j=1}^n). \quad (3)$$

The purpose of this max-pooling operation is to aggregate information from points belonging to the same object.

The aggregated features $\{\mathbf{V}_i^G\}_{i=1}^K$ are then fed into a Proposal Predictor, which employs an MLP to regress preliminary proposal results. The Proposal Predictor predicts the bounding box center $\hat{\mathbf{c}}_0$ and bounding box size $\hat{\mathbf{r}}_0$ for each point that defines the proposed regions corresponding to different objects in the 3D scene:

$$[\hat{\mathbf{c}}_0, \hat{\mathbf{r}}_0] = \text{MLP}(\mathbf{V}^G). \quad (4)$$

3.4 Top-Down Based Proposal Consolidation Module

Since the object proposals are not always accurate, *i.e.*, it may contain only a part of the object or include surrounding objects within the bounding box, it is crucial to refine them. However, previous bottom-up based methods directly output the regressed proposals as the final result, easily resulting in inaccurate localization. Although top-down based methods try to correlate all proposals for selecting the best one, they still haven't refined the proposals. To address these issues, we propose to construct a graph structure to learn the correlations between the proposals. Different from previous methods, we additionally develop a weighted edge for solely correlating the relevant proposals. Moreover, we further devise a novel proposal consolidation strategy to enrich and refine the

information in each proposal based on the contexts from its relevant proposals belonging to the same object.

Specifically, we aim to refine each proposal using a graph-based method that leverages information from its neighboring and relevant proposals. To achieve this, we start by constructing a fully connected graph among the generated proposals. However, we recognize that the correlations between different nodes (objects) in the graph can vary significantly. Objects that are close in spatial location and share similar categories often have stronger relationships. To make the most of these correlations between nodes while reducing interference from unrelated nodes, we define the edge weights W_{uv} between proposals as follows:

$$\mathbf{W}_{vu} = \begin{cases} \alpha \cos(\mathbf{V}_v^G, \mathbf{V}_u^G) + \beta \text{IoU}(\mathbf{c}_v, \mathbf{c}_u), & v \neq u \\ 1, & v = u. \end{cases} \quad (5)$$

Here, α and β are hyper-parameters that quantify the weights of the two terms. The first term represents the semantic correlation strength between two proposals, measured by their cosine similarity of features. The second term quantifies the spatial correlation between the two proposals based on the Intersection over Union (IoU) value of their preliminary bounding boxes. The bounding box can be computed using the center $\hat{\mathbf{c}}_0$ and size $\hat{\mathbf{r}}_0$.

Then, we employ the edge weights to update the features of the proposals \mathbf{V}^A through a weighted summation process:

$$\mathbf{V}_v^A = \sum_u \mathbf{W}_{vu} \mathbf{V}_u^G. \quad (6)$$

Through this step, we refine the representation of the proposals, taking into account their semantic and spatial associations with other relevant proposals in the scene.

3.5 Training Objectives

Based on the enriched features of the proposals \mathbf{V}^A , we utilize a proposal predictor to refine the bounding box information $[\hat{\mathbf{c}}, \hat{\mathbf{r}}]$ for each proposal along with the matching score \hat{s} between the proposal and the query text. We then select the proposal with the highest matching score as the final result.

To ensure the model achieves satisfactory results, we adopt a combination of multiple loss functions to supervise the entire pipeline.

During the preliminary proposal generation process, we utilize a center loss $\mathcal{L}_{\text{center}}$ to supervise the center predictor and ensure the correct prediction of object center coordinates. The center loss is computed by calculating the L1 loss between the predicted center coordinates \mathbf{c}_f and the ground truth center coordinates \mathbf{c}_{gt} corresponding to that point.

$$\mathcal{L}_{\text{center}} = \|\mathbf{c}_f - \mathbf{c}_{gt}\|_1. \quad (7)$$

To obtain the ground truth center for a point, we leverage the dataset’s object centers. For each point, we calculate its distance to all the ground truth object

centers in the dataset. The point is then assigned to the object with the closest center, making it the ground truth center for that point.

In the final selection stage, we employ a reference loss \mathcal{L}_{ref} to supervise the selection of the best matching proposal for the query text. For each proposal, with its matching score s_i and corresponding bounding box $[\hat{\mathbf{c}}, \hat{\mathbf{r}}]$, we determine its corresponding label using the following approach. First, we calculate the IoU between the predicted bounding box and the ground truth bounding box. Then, we set the label of the proposal y_i to 1 if it has the maximum IoU value among proposals and the IoU value is greater than a specified threshold. For all other proposals, the label is set to 0. The reference loss \mathcal{L}_{ref} is computed by calculating the cross-entropy between the predicted matching score s and the corresponding label y . The formula for the reference loss is as follows:

$$\mathcal{L}_{\text{ref}} = - \sum_i (y_i \log(s_i) + (1 - y_i) \log(1 - s_i)). \quad (8)$$

Additionally, similar to some previous works [2, 19], we incorporate an object detection loss \mathcal{L}_{det} to supervise the object detection process. This loss function helps the model accurately localize and identifies objects within the scene. Moreover, we utilize a Language to Object classification loss $\mathcal{L}_{\text{lang}}$ to aid in language understanding.

Overall Loss. By combining these different loss functions, our proposed approach aims to achieve improved performance in 3D visual grounding task. As a summary, the total loss used during the training process can be represented as:

$$\mathcal{L} = \lambda_1 \mathcal{L}_{\text{center}} + \lambda_2 \mathcal{L}_{\text{ref}} + \lambda_3 \mathcal{L}_{\text{det}} + \lambda_4 \mathcal{L}_{\text{lang}}, \quad (9)$$

where λ_1 , λ_2 , λ_3 and λ_4 are the weights assigned to different types of losses.

4 Experiments

4.1 Datasets and Evaluation Metric

To validate the effectiveness of our method and compare with previous works, we conducted experiments on the commonly used ScanRefer dataset [2] and two Referit3D dataset Nr3D and Sr3D [1]. ScanRefer is designed for the 3D visual grounding task. It contains a total of 51,583 textual descriptions corresponding to the objects provided in 806 scanned scenes from the ScanNet dataset. On average, each scene contains 13.81 objects, and each object is associated with 4.67 textual descriptions in the ScanRefer dataset.

To evaluate the performance, we utilize the metric Acc@kIoU, where ‘k’ represents the threshold for the IoU between the predicted bounding box and the ground truth. Following previous works, we set ‘k’ to 0.25 and 0.5 for our experiments.

Nr3D and Sr3D [1] provides 41.5K and 83.6K textual descriptions for scenes in ScanNet, respectively. We evaluate the effectiveness of our method on NR3D using the same metrics.

4.2 Implementation Details

During the training process, we employed 4 NVIDIA RTX3090 GPUs, with a batch size of 6 on each of the 4 GPU, resulting in a total effective batch size of 24. The training process was conducted for 32 epochs. We utilized the AdamW optimizer [16] with an initial learning rate of 0.001 for optimization. The pretrained language model is frozen. During the k -th epoch, the learning rate was calculated using the following formula:

$$lr(k) = 0.5 \times \left(1 + \cos \frac{(k-1)\pi}{32} \right) \times 0.001. \quad (10)$$

During the encoding stage, we employed pre-trained PointNet++ and CLIP models. The input information includes point cloud coordinates, normal vectors, color vectors and 2D multiview features. The number M_0 of the output points of pointnet++ is 2048. In the bottom-up proposal generation module, the number of points M_f filtered based on attention coefficients is set to 512. Afterward, the number of points K selected using the FPS is set to 128. In the top-down based proposal consolidation module, the coefficients α and β in Eq. 5 are set to 0.7 and 0.3, respectively. As for the loss function, the coefficients λ_1 , λ_2 , λ_3 and λ_4 are set to 5, 0.1, 5, and 0.1, respectively.

Table 1. Comparison on ScanRefer dataset.

Method	unique		multiple		overall	
	Acc@0.25	Acc@0.5	Acc@0.25	Acc@0.5	Acc@0.25	Acc@0.5
ScanRefer [2]	76.33	53.51	32.73	21.11	41.19	27.40
InstanceRefer [34]	75.72	64.66	29.41	22.99	38.40	31.08
SAT [33]	73.21	50.83	37.64	25.16	44.54	30.14
3DVG-Transformer [35]	81.93	60.64	39.30	28.42	47.57	34.67
3D-SPS (reported) [17]	84.12	66.72	40.32	29.82	48.82	36.98
3D-SPS (Re-imple) [17]	82.82	64.77	39.58	29.11	47.97	36.03
Ours	82.60	66.83	40.96	30.81	49.04	37.80

4.3 Comparison with SOTA

We compared our experimental results on the ScanRefer dataset with those of previous methods, as shown in Table 1. In our evaluation, we measured the proportion of predicted bounding boxes with an IoU greater than 0.25 and 0.5 concerning the ground truth bounding boxes. All methods in the table utilize 3D point cloud features combined with 2D multiview features as inputs. The methods we compare with encompass both top-down based approaches, such as ScanRefer [2], InstanceRefer [34], SAT [33], and 3DVG-Transformer [35], as well

as bottom-up based approaches, such as 3D-SPS [17]. To provide a comprehensive analysis, we followed the example of previous works [2] and presented the results separately for the “unique” and “multiple” subsets. The “unique” subset refers to scenes where there is only one object of the same category as the target object, while the “multiple” subset includes scenes with multiple objects of the same category. In the table, “reported” represents the results reported in the paper, while “our implementation” refers to the results of our reproduced experiments.

From Table 1, our method has achieved the best results in five out of the six metrics. Particularly, in the two primary metrics, “overall-Acc@0.25” and “overall-Acc@0.5”, our method has demonstrated improvements of 0.56 and 1.61% points respectively compared to the previously leading approach. The improvements in accuracy over the previous state-of-the-art method (our implementation) demonstrate the effectiveness of our proposed technique.

To further validate the effectiveness of our proposed method, we conducted experiments on two datasets from ReferIt3D [1], namely NR3D and SR3D. We adopted an experimental setup similar to ScanRefer [2], where we directly predict the bounding box of the object described in the text, without relying on the provided bounding box information. We used a similar evaluation metric, Acc@0.25, for assessment. The experimental results are presented in Table 2. The results for other methods in the table were also obtained using the same experimental setup. From the Table 2, it can be observed that our method outperforms all others on both of these datasets.

Table 2. Comparison on Referit3D dataset.

Method	SR3D/Acc@0.25	NR3D/Acc@0.25
ReferIt3D	27.7	24.0
InstanceRefer	31.5	29.9
LanguageRefer	39.5	28.6
SAT	35.4	31.7
Ours	41.4	32.0

4.4 Ablation Study

Main Ablation. To further demonstrate the effectiveness of our proposed bottom-up based proposal generation module and top-down based proposal consolidation module, we conduct the following experiments on the ScanRefer dataset as shown in Table 3. In the first row, we report the performance achieved using a pre-trained encoder and transformer module, along with a proposal generation method similar to ScanRefer, serving as our baseline for comparison. The second and third rows illustrate the results when our proposed bottom-up module and top-down module are incorporated, respectively. The last row presents

Table 3. Ablation study results on ScanRefer dataset using ‘overall-Acc@0.25’ and ‘overall-Acc@0.5’ metrics.

bottom-up	top-down	Acc@0.25	Acc@0.5
×	×	44.27	33.30
✓	×	45.30	34.32
×	✓	45.46	33.82
✓	✓	49.04	37.80

Table 4. Ablation study on the number K of points selected during the farthest point sampling step in the bottom-up based proposal generation module.

K	Acc@0.25	Acc@0.5
256	45.90	34.50
128	49.04	37.80
64	45.83	34.50

Table 5. Ablation study on the number n of graph-based information aggregation iterations.

n	Acc@0.25	Acc@0.5
0	45.30	34.32
1	49.04	37.80
2	44.81	33.28

the results achieved by the complete model, integrating all components. From Table 3, it is evident that both of the proposed modules contribute to improving the experimental results. This observation highlights the essential role of both bottom-up and top-down cues in the 3D visual grounding task. Notably, when these two components are combined, they achieve the highest accuracy, demonstrating their complementary roles in addressing the 3D visual grounding task.

Ablation on Number K in the Farthest Point Sampling. Furthermore, we conducted an ablation study on the number of points selected during the farthest point sampling step in the bottom-up based proposal generation module. The experimental results are presented in Table 4. From the results shown in Table 4, we can conclude that selecting $K=128$ points yields the optimal performance.

Ablation on Graph Layers in Proposal Consolidation Module. We also conducted experiments on the number n of graph-based information aggregation iterations during the top-down stage, and the results are presented in Table 5. It can be observed that performing graph-based information aggregation once led to a significant improvement in the results. However, increasing the number of aggregation iterations had a detrimental effect on the results, likely due to factors such as oversmoothing [11], where excessive aggregation on the graph data blurs features.

More experiments of ablation studies can be found in our supplementary.

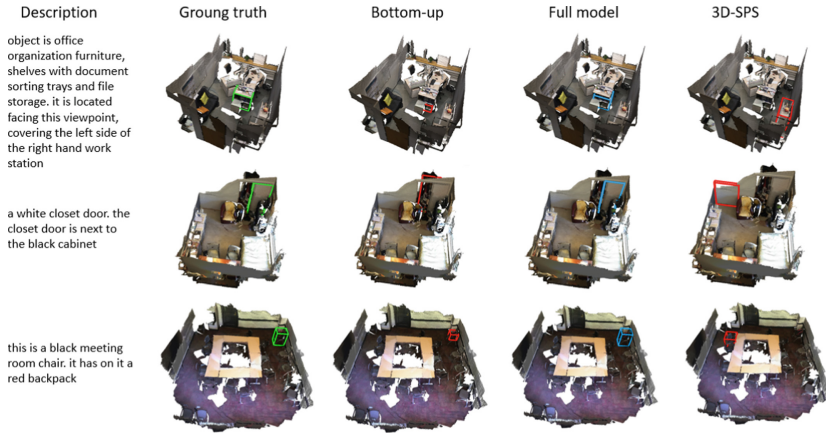


Fig. 3. Visualization of our method. The first column displays the ground truth bounding boxes provided by the ScanRefer dataset. The second and third columns represent the output results of our bottom-up based proposal generation module and the final output of the entire model, respectively. The last column shows the results obtained from the 3D-SPS method.

4.5 Visualization

In Fig. 3, we present a visual comparison of the output results from the two modules of our method, as well as the results obtained from the 3D-SPS [17] method. We can see that our bottom-up based proposal generation module, with the aid of query language information, can locate the target objects. However, the quality of the bounding boxes generated in this step often falls short of being optimal. For instance, in the first row, the bounding box only encompasses the lower part of the shelf, and in the third row, the bounding box only covers the upper part of the chair. Nevertheless, through the subsequent top-down based proposal consolidation module and by leveraging information from surrounding proposals, the bounding boxes can be refined to better represent the entire object’s position. This consolidation process allows our approach to provide more accurate and complete bounding box predictions, enhancing the overall localization performance. When comparing our method to the 3D-SPS [17] approach, we can observe that our method excels at precisely localizing the objects described in the query language. **More visualizations are in our supplementary.**

5 Conclusion

In this paper, we have analyzed two categories of methods used in 3D visual grounding: the top-down based method and the bottom-up based method, each with its respective strengths and weaknesses. Our proposed approach integrates the advantages of both methods effectively while alleviating their limitations. Firstly, we utilize a bottom-up based proposal generation module to

produce high-quality candidate proposals relevant to the query information. Subsequently, a top-down based consolidation module is employed to further enhance the performance. As a result, our proposed method demonstrates superior performance compared to the state-of-the-art results on the ScanRefer dataset. Furthermore, our approach can serve as a flexible framework, enabling the replacement of both the bottom-up based module and the top-down based module with more advanced methods to achieve even better results in future research.

References

1. Achlioptas, P., Abdelreheem, A., Xia, F., Elhoseiny, M., Guibas, L.: ReferIt3D: neural listeners for fine-grained 3D object identification in real-world scenes. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12346, pp. 422–440. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58452-8_25
2. Chen, D.Z., Chang, A.X., Nießner, M.: ScanRefer: 3D object localization in RGB-D scans using natural language. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12365, pp. 202–221. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58565-5_13
3. Chen, S., Fang, J., Zhang, Q., Liu, W., Wang, X.: Hierarchical aggregation for 3D instance segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 15467–15476 (2021)
4. Cheng, B., Sheng, L., Shi, S., Yang, M., Xu, D.: Back-tracing representative points for voting-based 3d object detection in point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8963–8972 (2021)
5. Deng, J., Yang, Z., Chen, T., Zhou, W., Li, H.: TranSVG: end-to-end visual grounding with transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 1769–1779 (2021)
6. Feng, M., et al.: Free-form description guided 3D visual graph network for object grounding in point cloud. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 3722–3731 (2021)
7. He, D., et al.: Transrefer3D: entity-and-relation aware transformer for fine-grained 3d visual grounding. In: Proceedings of the 29th ACM International Conference on Multimedia, pp. 2344–2352 (2021)
8. Huang, P.H., Lee, H.H., Chen, H.T., Liu, T.L.: Text-guided graph neural networks for referring 3d instance segmentation. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp. 1610–1618 (2021)
9. Jiang, L., Zhao, H., Shi, S., Liu, S., Fu, C.W., Jia, J.: Pointgroup: dual-set point grouping for 3D instance segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4867–4876 (2020)
10. Li, M., Sigal, L.: Referring transformer: a one-step approach to multi-task visual grounding. In: Advances in Neural Information Processing Systems, vol. 34, pp. 19652–19664 (2021)
11. Li, Q., Han, Z., Wu, X.M.: Deeper insights into graph convolutional networks for semi-supervised learning. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 32 (2018)

12. Liu, D., Liu, Y., Huang, W., Hu, W.: A survey on text-guided 3D visual grounding: elements, recent advances, and future directions. arXiv preprint [arXiv:2406.05785](https://arxiv.org/abs/2406.05785) (2024)
13. Liu, H., Lin, A., Han, X., Yang, L., Yu, Y., Cui, S.: Refer-it-in-RGBD: a bottom-up approach for 3D visual grounding in RGBD images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6032–6041 (2021)
14. Liu, Y., Liu, D., Guo, Z., Hu, W.: Cross-task knowledge transfer for semi-supervised joint 3D grounding and captioning. In: Proceedings of the 32st ACM International Conference on Multimedia. ACM (2024)
15. Liu, Z., Zhang, Z., Cao, Y., Hu, H., Tong, X.: Group-free 3D object detection via transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 2949–2958 (2021)
16. Loshchilov, I., Hutter, F.: Fixing weight decay regularization in ADAM (2018)
17. Luo, J., et al.: 3D-SPS: single-stage 3D visual grounding via referred point progressive selection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 16454–16463 (2022)
18. Misra, I., Girdhar, R., Joulin, A.: An end-to-end transformer model for 3D object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 2906–2917 (2021)
19. Qi, C.R., Litany, O., He, K., Guibas, L.J.: Deep Hough voting for 3D object detection in point clouds. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 9277–9286 (2019)
20. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: deep learning on point sets for 3D classification and segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 652–660 (2017)
21. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: deep hierarchical feature learning on point sets in a metric space. In: Advances in Neural Information Processing Systems, vol. 30 (2017)
22. Radford, A., et al.: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning, pp. 8748–8763. PMLR (2021)
23. Radosavovic, I., Kosaraju, R.P., Girshick, R., He, K., Dollár, P.: Designing network design spaces. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10428–10436 (2020)
24. Sadhu, A., Chen, K., Nevatia, R.: Zero-shot grounding of objects from natural language queries. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4694–4703 (2019)
25. Vaswani, A., et al.: Attention is all you need. In: Advances in Neural Information Processing Systems, vol. 30 (2017)
26. Vu, T., Kim, K., Luu, T.M., Nguyen, T., Yoo, C.D.: Softgroup for 3D instance segmentation on point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2708–2717 (2022)
27. Wang, L., Li, Y., Huang, J., Lazebnik, S.: Learning two-branch neural networks for image-text matching tasks. *IEEE Trans. Pattern Anal. Mach. Intell.* **41**(2), 394–407 (2018)
28. Wang, P., Wu, Q., Cao, J., Shen, C., Gao, L., Hengel, A.v.d.: Neighbourhood watch: Referring expression comprehension via language-guided graph attention networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1960–1968 (2019)

29. Wang, Y., Sun, Y., Liu, Z., Sarma, S.E., Bronstein, M.M., Solomon, J.M.: Dynamic graph CNN for learning on point clouds. *ACM Trans. Graph. (tog)* **38**(5), 1–12 (2019)
30. Yang, A., Miech, A., Sivic, J., Laptev, I., Schmid, C.: TubeDeTR: spatio-temporal video grounding with transformers. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16442–16453 (2022)
31. Yang, S., Li, G., Yu, Y.: Cross-modal relationship inference for grounding referring expressions. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4145–4154 (2019)
32. Yang, Z., Gong, B., Wang, L., Huang, W., Yu, D., Luo, J.: A fast and accurate one-stage approach to visual grounding. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4683–4693 (2019)
33. Yang, Z., Zhang, S., Wang, L., Luo, J.: Sat: 2D semantics assisted training for 3D visual grounding. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1856–1866 (2021)
34. Yuan, Z., et al.: InstanceRefer: cooperative holistic understanding for visual grounding on point clouds through instance multi-level contextual referring. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1791–1800 (2021)
35. Zhao, L., Cai, D., Sheng, L., Xu, D.: 3DVG-transformer: relation modeling for visual grounding on point clouds. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2928–2937 (2021)



Anticipating Future Object Compositions Without Forgetting

Youssef Zahran^{1,2}^(✉), Gertjan Burghouts², and Yke B. Eisma¹

¹ Department of Cognitive Robotics, Delft University of Technology,
Delft, The Netherlands

y.zahran09@gmail.com, Y.B.Eisma@tudelft.nl

² Intelligent Imaging Department, TNO, The Hague, The Netherlands
gertjan.burghouts@tno.nl

Abstract. Despite the significant advancements in computer vision models, their ability to generalize to novel object-attribute compositions remains limited. Existing methods for Compositional Zero-Shot Learning (CZSL) mainly focus on image classification. This paper aims to enhance CZSL in object detection without forgetting prior learned knowledge. We use Grounding DINO and incorporate Compositional Soft Prompting (CSP) into it and extend it with Compositional Anticipation. We achieve a 70.5% improvement over CSP on the harmonic mean (HM) between seen and unseen compositions on the CLEVR dataset. Furthermore, we introduce Contrastive Prompt Tuning to incrementally address model confusion between similar compositions. We demonstrate the effectiveness of this method and achieve an increase of 14.5% in HM across the pretrain, increment, and unseen sets. Collectively, these methods provide a framework for learning various compositions with limited data, as well as improving the performance of underperforming compositions when additional data becomes available.

Keywords: compositional zero-shot learning · prompt tuning · incremental learning

1 Introduction

Although humans have never seen a blue apple, they can easily picture it. This is due to the inherent human ability to generalize to novel concepts by combining the known entity “apple” with the color “blue”. However, do computer vision models possess this capability? This question has motivated the development of Compositional Zero-Shot Learning (CZSL) [5, 21, 22, 26]. In CZSL, the goal is to recognize unseen object-attribute combinations, referred to as compositions, based on the compositions seen during training. For this, models should understand the attributes and objects that compose these compositions to generalize to all possible compositions.

Vision Language Models (VLMs), pretrained on large-scale image-text pairs, are promising for CZSL due to their ability to understand the relationship between the visual content and the textual description [16, 22, 26]. For object

detection, VLMs such as Grounding DINO [10] and GLIP [7] learn to associate regions of text with regions of images by pulling the embeddings of paired image regions and text descriptions close while pushing others away [25]. These models perform cross-modality fusion throughout the whole architecture, which makes the textual features image-aware and the visual features text-aware. Liu et al. [10] argue that VLMs benefit from frequent cross-modality fusion, making Grounding DINO superior to GLIP. Therefore, throughout this paper, we will solely focus on Grounding DINO.

Unfortunately, these models tend to be biased towards object categories rather than attributes, which makes it suffer from feature misalignment when used directly for attribute recognition [3]. Fine-tuning VLMs can solve this but often leads to catastrophic forgetting of prior knowledge [28], thereby compromising their generalization ability. To address this, we explore how to fine-tune VLMs to perform well in CZSL without forgetting any prior knowledge. Nayak et al. [15] introduced Compositional Soft Prompting (CSP), which combats catastrophic forgetting by adding auxiliary tokens for all words in a given dataset and training only these tokens. This approach preserves the model’s original embeddings, allowing it to retain and revert to its initial knowledge when necessary, unlike full fine-tuning, which alters the model’s parameters. CSP improves model performance in CZSL for image classification. We incorporate CSP in Grounding DINO, to leverage it for object detection.

We consider CSP as a baseline and improve it for CZSL by introducing Compositional Anticipation (CA), which recognizes that additional compositions may exist beyond those present during training. In this context, the term “anticipation” does not refer to actively predicting new compositions. Instead, it involves enhancing the model’s ability to handle potential new compositions by adjusting how it processes partially correct predictions through Compositional Smoothing and by guiding the model to disentangle attributes from objects via Compositional Independence. Compositional Smoothing prepares the model for novel compositions by assigning soft labels when predictions are partially correct, e.g., the object is correct but the attribute is different. This approach deviates from conventional Label Smoothing [20], which assigns soft labels to all classes. Compositional Independence disentangles objects from attributes through Separation and Decorrelation. Separation introduces a separation loss to maximize the distinction between object and attribute classes by applying intra-class separation within objects and attributes and an inter-class separation between objects and attributes. Decorrelation minimizes the correlation between objects and attributes to reduce dependency between the two.

For incremental learning on newly added compositions, we use prior knowledge to address specific mistakes related to confusion between similar compositions. Inspired by recent developments in prompt tuning [17, 29, 30], we introduce a novel method called Contrastive Prompt Tuning, specifically tailored for object attributes. Contrastive Prompt Tuning addresses cases where the model confuses similar compositions, such as mistaking a blue apple for a red apple, by adding a trainable prompt in front of the confused class: “is not red apple but is blue

apple”. This approach utilizes our prior knowledge to harness the ability of a VLM to exploit language.

In summary, our main contributions are:

1. We incorporate CSP [15] into Grounding DINO [10] and extend it with Compositional Anticipation. Compositional Anticipation consists of:
 - Compositional Smoothing, which assigns soft labels when predictions are partially correct.
 - Compositional Independence, which disentangles objects from attributes.
2. We develop Contrastive Prompt Tuning, a method that adds a learnable prompt for compositions that are confused with each other during training. This technique harnesses the power of language and our understanding of the model to improve performance beyond simply training with additional data.

2 Related Work

In this section, we review literature related to our work. We cover Compositional Zero-Shot Learning (CZSL), Prompt Tuning in Vision-Language Models (VLMs), and Class Incremental Learning (CIL). Our research focuses on improving the CZSL capabilities of Grounding DINO [10], a VLM designed for object detection, by utilizing prompt tuning. Additionally, we address underperforming compositions in a class-incremental manner to further improve model performance.

Compositional Zero-Shot Learning. The main objective of CZSL is to recognize unseen compositions from the compositions encountered during training. In CZSL, individual objects and attributes are referred to as primitives. Misra et al. [13] use a limited set of compositions to learn linear classifiers for each primitive. Then, they learn a transformation network that takes these classifiers as input and composes them to produce a classifier for their combination. Since then, multiple works [8, 12, 14, 19] have been proposed to tackle the CZSL task.

Recent works focus on adapting pretrained VLMs for CZSL by fine-tuning primitive tokens. While CSP [15] only trains these tokens, others [11, 21] also introduce prompt disentangled tuning. This technique addresses entanglement, where optimizing one primitive’s embedding affects another. Prompt disentangled tuning divides the process into three phases with different prompts: one for the entire composition, one for the attribute, and one for the object. This ensures attributes and objects learn their optimal parameters independently.

While [11, 21] improve upon [15] with an average performance increment of 1.7% and 2.3%, respectively, the gains are marginal relative to the increased complexity. Our work is closely related to [11, 15] as we adapt CSP for object detection and address entanglement through Compositional Independence.

Prompt Tuning in VLMs. Ever since CLIP [16] demonstrated that prompt templates such as “a photo of a [CLASS]” improve the results of VLMs compared to using only the classname, several other works [17, 24, 29, 30] have been introduced to replace the hand-crafted prompt with learnable soft prompts. CoOP [30] introduces soft prompts that are shared across all classes, resulting in prompts like $[v_1], [v_2], \dots [v_M]$ for all images. CoCoOp [29] improves upon this by proposing soft prompts that are image-conditioned, generating prompts such as $[v_1(x)], [v_2(x)], \dots [v_M(x)]$ for each image x . Building upon these advancements, Rao et al. [17] use contextual information from the image to prompt the language model.

Our work is closely related to these works but is unique in its focus on improving the performance of confused compositions using learnable prompts that are initialized based on our knowledge of the model’s errors.

Class Incremental Learning. Class-Incremental Learning (CIL) refers to learning new classes while retaining previously learned classes [27, 28]. In typical CIL scenarios, learning occurs through a sequence of training tasks, each of which introduce new classes without any overlap of the classes from previous tasks. The main challenge is avoiding catastrophic forgetting, where learning new classes leads to a loss of knowledge from previous tasks. Our approach bears resemblance to Blurry CIL [1, 2], where former classes can be revisited during training. Similarly, we train incrementally with underperforming compositions while allowing former compositions to be revisited.

3 Method

3.1 Problem Definition

Compositional Zero-Shot Learning. We follow [21, 22, 26] and formalize the CZSL task as follows. Let \mathcal{A} denote the set of attributes, and \mathcal{O} the set of objects, and $\mathcal{C} = \mathcal{A} \times \mathcal{O}$ the set of all compositions. $\mathcal{T} = \{(x_j, c_j)\}_{j=1}^N$ denotes the train set where $x_j \in \mathcal{X}$ is a sample in the input (image) space \mathcal{X} and $c_j \in \mathcal{C}_s$ is a composition in the subset $\mathcal{C}_s \subseteq \mathcal{C}$. The seen set $\mathcal{C}_s \subseteq \mathcal{C}$ consists of all compositions encountered during training, whereas the unseen set $\mathcal{C}_u \subseteq \mathcal{C}$ consists of compositions not seen during training. Let \mathcal{C}_s and \mathcal{C}_u be two sets such that $\mathcal{C}_s \cap \mathcal{C}_u = \emptyset$. While \mathcal{C}_s and \mathcal{C}_u are disjoint, the objects \mathcal{O}_u and attributes \mathcal{A}_u are defined such that $\mathcal{O}_u \subseteq \mathcal{O}_s$ and $\mathcal{A}_u \subseteq \mathcal{A}_s$.

Catastrophic Forgetting. VLMs, such as Grounding DINO [10], are known for their ability to generalize well across diverse tasks due to the extensive and varied data used during pre-training. However, fine-tuning these models on a new dataset often compromises their generalization capability, as the rich features learned during pre-training are replaced by features specific to the new dataset. This can lead to catastrophic forgetting, where the model’s performance on previously learned tasks significantly deteriorates. In the context of

CZSL with VLMs, catastrophic forgetting is particularly problematic. While the model may perform well on the specific compositions present in the new dataset it was fine-tuned on, it risks becoming overly specialized. This specialization may result in a model that loses its ability to generalize to other compositions, objects, or concepts, and instead becomes exceptionally good at predicting the compositions seen during training. Such a limitation is especially undesirable for open-set object detectors like Grounding DINO, which are meant to recognize a wide range of concepts.

3.2 Incremental CZSL

In practical settings, models often encounter new data or need to improve performance on underperforming compositions after the initial training phase. To address this, we introduce an increment set $\mathcal{C}_i \subseteq \mathcal{C}$ to CZSL. Let \mathcal{C}_p be the set used for the initial fine-tuning of the model, with $\mathcal{C}_p = \mathcal{C}_s$. After introducing \mathcal{C}_i , the set of seen compositions becomes $\mathcal{C}_s = \mathcal{C}_p \cup \mathcal{C}_i$. The increment set \mathcal{C}_i consists of compositions introduced after the initial fine-tuning to improve performance on underperforming compositions. Improving these underperforming compositions with additional compositions is challenging because \mathcal{C}_p is designed to cover \mathcal{A} and \mathcal{O} with the minimum number of compositions. Extending \mathcal{C}_s with \mathcal{C}_i makes the attributes and objects in \mathcal{C}_i overrepresented in \mathcal{C}_s , which can bias the model towards these attributes and objects. In this paper, we focus solely on improving performance on compositions $c_j \in \mathcal{C}$ without extending the attribute set \mathcal{A} or the object set \mathcal{O} .

3.3 Compositional Soft Prompting

To prevent catastrophic forgetting in Grounding DINO [10], we follow CSP [15] and modify it for object detection. Objects and attributes that form compositions are treated as learnable tokens within the VLMs vocabulary. Each attribute $a_j \in \mathcal{A}$ and each object $o_j \in \mathcal{O}$ is represented as an auxiliary token t_{a_j} and t_{o_j} respectively, where $t_{a_j}, t_{o_j} \in \mathbb{R}^d$, with d being the dimension of the vocabulary embedding. During training, only these auxiliary tokens are tuned, resulting in $(|\mathcal{A}| + |\mathcal{O}|) \times d$ learnable parameters.

To illustrate, CSP creates auxiliary tokens for each attribute and object such as t_{blue} for the attribute “blue” and t_{apple} for the object “apple”. These tokens are adjusted during training while the rest of the weights, such as those of the encoder and decoder in Grounding DINO, remain unchanged. By doing this, CSP prevents catastrophic forgetting and preserves the pretrained weights of the model.

3.4 Compositional Smoothing

To combat bias during training for \mathcal{C}_s , where the model becomes overly confident with the seen classes, we assign soft labels rather than hard labels (0 and 1) in

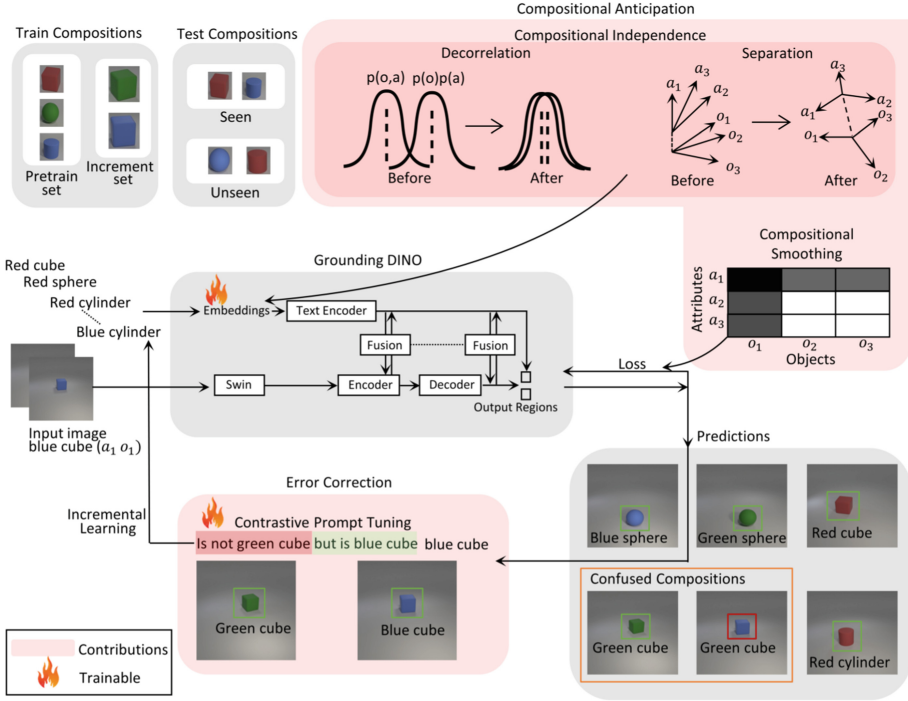


Fig. 1. Our method anticipates unseen, future object-attribute compositions through Compositional Independence and Compositional Smoothing. Forgetting is mitigated by creating auxiliary tokens for the language embeddings and refining only these tokens. Errors in compositions are incrementally corrected using Contrastive Prompt Tuning, which contrasts confused compositions.

the classification loss. This is referred to as Label Smoothing [20], and it prevents the model from becoming overly confident in its predictions, thereby improving its generalization capability. Conventional Label Smoothing adjusts the target labels by distributing a small portion of the probability mass to all other labels. For a given true label y in a classification problem with k classes, the smoothed label y_{smooth} is defined as:

$$y_{\text{smooth}} = (1 - \epsilon)y + \frac{\epsilon}{k}, \tag{1}$$

where ϵ is the smoothing parameter, and the term $\frac{\epsilon}{k}$ distributes the smoothing equally among all classes.

We deviate from conventional Label Smoothing [20] and assign soft labels based on the correctness of the object, attribute, or the entire composition. We refer to this as Compositional Smoothing. Let p_O , p_A , and p_C represent the probabilities for object, attribute, and overall composition predictions, respectively. For a given true composition c_t composed of object o_t and attribute a_t , and

predicted composition c_p composed of object o_p and attribute a_p , the smoothed label $y_{o,a}$ is defined as:

$$y_{o,a} = \begin{cases} p_{\mathcal{O}} & \text{if } o_p = o_t \wedge a_p \neq a_t, \\ p_{\mathcal{A}} & \text{if } a_p = a_t \wedge o_p \neq o_t, \\ p_{\mathcal{C}} & \text{if } a_p = a_t \wedge o_p = o_t, \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

Compositional Smoothing ensures that there is a difference between having partial correctness and no correctness in the prediction, guiding the model to learn what the compositions are composed of rather than learning the compositions themselves. This, in turn, should lead to better performance on \mathcal{C}_u .

Figure 1 (top right) illustrates Compositional Smoothing. For a given ground truth label (a_1, o_1) , predictions where both the attribute and object are correct are shown in black, and the smoothed label becomes $p_{\mathcal{C}}$. Partial correctness is depicted in gray, with the smoothed label being either $p_{\mathcal{O}}$ or $p_{\mathcal{A}}$. When both the object and attribute are completely wrong, no smoothing is applied.

3.5 Compositional Independence

In CZSL, it is important to disentangle objects from attributes and have clear distinctions within each category. For example, a cube and a cylinder should be easily distinguishable to prevent confusion. Additionally, colors should be distinguished from specific objects, such as cubes, to ensure their independence. This prevents similar attributes or objects to be confused with each other and helps the model treat attributes and objects as distinct concepts.

We achieve this independence through two components: Separation and Decorrelation. Separation enforces orthogonality within the embeddings of objects and attributes and maximizes the distance between their mean embeddings. Decorrelation minimizes the correlation between the embeddings of objects and attributes. This is achieved using the Hilbert-Schmidt Independence Criterion (HSIC) [4], a kernel statistical test commonly used to measure independence between two random variables, which proved to be effective for CZSL image classification [18] and is leveraged here for object detection.

Separation. To help the model differentiate between similar attributes or objects, we introduce an orthogonality loss. We achieve orthogonality within the groups of attributes and objects by minimizing the average absolute similarity between the normalized embeddings within each group:

$$\mathcal{L}_{\text{orth}}(\mathbf{E}) = \frac{1}{|\mathbf{E}|^2 - |\mathbf{E}|} \sum_{i=1}^{|\mathbf{E}|} \sum_{\substack{j=1 \\ j \neq i}}^{|\mathbf{E}|} |\mathbf{e}_i \cdot \mathbf{e}_j| \quad (3)$$

where \mathbf{E} is the set of normalized embeddings, and \mathbf{e}_i and \mathbf{e}_j are embeddings within this set. The summation $\sum_{j \neq i}$ ignores self-similarity, and $\frac{1}{|\mathbf{E}|^2 - |\mathbf{E}|}$ ensures

that self-similar terms are excluded during normalization. This orthogonality loss is applied to both the attributes and objects:

$$\mathcal{L}_{\mathcal{A}} = \mathcal{L}_{\text{orth}}(\mathbf{E}_{\mathcal{A}}) \quad (4)$$

$$\mathcal{L}_{\mathcal{O}} = \mathcal{L}_{\text{orth}}(\mathbf{E}_{\mathcal{O}}) \quad (5)$$

where $\mathbf{E}_{\mathcal{A}}$ and $\mathbf{E}_{\mathcal{O}}$ represent the sets of normalized embeddings for the attributes and objects, respectively.

Additionally, to enforce a clear distinction between attributes and objects, we ensure that the mean embeddings of attributes and objects are significantly separated:

$$\mathcal{L}_{\text{distance}} = -\log(\|\mu_{\mathcal{A}} - \mu_{\mathcal{O}}\|_2) \quad (6)$$

where $\mu_{\mathcal{A}} = \frac{1}{|\mathcal{A}|} \sum_{j=1}^{|\mathcal{A}|} \mathbf{e}_{\mathcal{A}_j}$ and $\mu_{\mathcal{O}} = \frac{1}{|\mathcal{O}|} \sum_{j=1}^{|\mathcal{O}|} \mathbf{e}_{\mathcal{O}_j}$ represent the mean embeddings of attributes and objects, respectively. The distance is computed using the L_2 norm between the mean embeddings of the two groups.

The total Separation loss is a weighted combination of the orthogonality and mean separation components:

$$\mathcal{L}_{\text{separation}} = \lambda_1 \mathcal{L}_{\text{distance}} + \lambda_2 \mathcal{L}_{\mathcal{A}} + \lambda_3 \mathcal{L}_{\mathcal{O}} \quad (7)$$

where λ_1, λ_2 and λ_3 are hyperparameters controlling the contribution of $\mathcal{L}_{\text{distance}}$, $\mathcal{L}_{\mathcal{A}}$ and $\mathcal{L}_{\mathcal{O}}$ to the final loss, respectively.

Decorrelation. To further ensure the independence between object and attribute embeddings, we introduce Decorrelation by using HSIC [4]. For an object o_j with attribute a_j , we formulate the HSIC loss as follows:

$$\mathcal{L}_{\text{hsic}} = \lambda_h \text{HSIC}(o_j, a_j) \quad (8)$$

Here, λ_h is a hyperparameter that controls the contribution of the HSIC term to the total loss.

3.6 Compositional Anticipation

Our method, which we refer to as Compositional Anticipation (CA), consists of both Compositional Smoothing and Compositional Independence. While CA does not actively predict unseen compositions, it prepares the model by refining how it handles potential new compositions and disentangles attributes from objects. Figure 1 shows how we implement CA in Grounding DINO [10].

3.7 Contrastive Prompt Tuning

To improve the performance of some underperforming composition after training with \mathcal{C}_p , we extend \mathcal{C}_s with an additional set \mathcal{C}_i to improve performance. Our approach begins with analyzing the predictions to identify compositions that are

frequently confused with each other. For instance, if c_j and c_k are often mixed-up, both compositions are included in \mathcal{C}_i , and a trainable prompt is added in front of the underperforming class(es). For example, if c_j performs poorly, we add the following prompt in front of the class: “*is not c_k but is c_j* ”. This prompt contains both a negative and an affirmative component.

We refer to this method as Contrastive Prompt Tuning and it does not modify any of the tokens present in the sets \mathcal{A} and \mathcal{O} . Instead, it focuses solely on the learnable prompt, which leads to fewer changes in the performance of other compositions and mitigates catastrophic forgetting. By doing this, we exploit the ability of a VLM to understand language and use a semantically meaningful initial prompt to learn to distinguish between similar compositions. This step is depicted as Incremental Learning in Fig. 1.

4 Experiments

4.1 Evaluation

Dataset. We evaluate our approach using a synthetic dataset generated following the CLEVR framework [6]. This dataset consists of three types of objects: cube, cylinder, and sphere. Each object is associated with six attributes: blue, red, green, purple, brown, and yellow.

The dataset intentionally excludes non-visual attributes (e.g., heavy) and attributes that exhibit significant variation across different objects (e.g., wet in wet dog versus wet car). This yields a dataset that is reliable for assessing a model’s performance in the CZSL task. Given that there are no ambiguous attributes present in this dataset, a poorly performing model would indicate that the model is bad in the CZSL task.

Train-Test Split. Throughout this section, all experiments for the CZSL task are trained using the set: {red cube, blue cube, green sphere, purple sphere, brown cylinder, yellow cylinder} as \mathcal{C}_p with 10 shots per composition. This split ensures that \mathcal{C}_s covers the entire set of objects \mathcal{O} and attributes \mathcal{A} . Testing is performed with the whole set of composition \mathcal{C} with 60 samples per composition.

Evaluation Metric. We adopt the NMS mAP evaluation metric introduced by Yoa et al. [23]. In this work they argued that the traditional COCO mAP [9] is deceiving for open vocabulary detection models, such as Grounding DINO [10]. Consider an image annotated with two ground-truth instances: a purple cylinder and a green cylinder, assuming these are the only cylinder categories in the model. These models tend to be able to detect and locate the presence of all cylinders in the image, but they struggle with the contextual description. They would predict two overlapping bounding boxes for each object, mistakenly assigning both ‘green’ and ‘purple cylinder’ labels to each object. All four of these boxes would be predicted with a high confidence score. Additionally, the highest scoring label is not necessarily the correct one. Consequently, the AP for

each category would misleadingly be 0.50, despite the model failing to correctly comprehend the target objects. Yao et al. [23] refer to this as the ‘inflated AP problem’.

To address this issue, Yao et al. [23] propose applying class-agnostic Non-Maximum Suppression (NMS) before calculating the mAP. This method suppresses redundant bounding boxes, ensuring that only the prediction with the highest confidence score is used in the calculation of the mAP. We adopt this NMS mAP metric to provide a more realistic measure of our model’s performance.

4.2 CSP Base

We adapt CSP [15] and modify it for Grounding DINO [10] and this integration serves as our baseline method. To assess its performance, we begin by training it with $\mathcal{C}_p = \mathcal{C}$. This yields an NMS mAP of 87.2 ± 6.8 , demonstrating that good performance can be achieved by only training the embeddings of \mathcal{O} and \mathcal{A} .

Table 1. Compositional Anticipation improves both object detection performance and generalization to unseen compositions. Compositional Smoothing contributes the most to these improvements, followed by Separation and Decorrelation.

Compositional Anticipation (CA)			Seen	Unseen	HM
Compositional Smoothing	Separation	Decorrelation			
✗	✗	✗	81.4 ± 7.6	4.5 ± 4.6	8.0 ± 8.1
✗	✗	✓	81.3 ± 7.7	10.8 ± 6.7	18.2 ± 10.5
✗	✓	✗	82.5 ± 6.8	15.1 ± 4.2	25.4 ± 6.1
✗	✓	✓	84.4 ± 7.1	20.8 ± 4.7	33.1 ± 6.4
✓	✗	✗	86.2 ± 6.8	64.3 ± 5.9	73.5 ± 5.3
✓	✗	✓	92.4 ± 3.0	61.6 ± 5.7	73.8 ± 4.5
✓	✓	✗	86.0 ± 6.1	67.7 ± 4.7	75.7 ± 5.0
✓	✓	✓	88.7 ± 4.9	70.6 ± 7.4	78.5 ± 6.0

Table 2. Our model does not forget. It achieves good performance on the fine-tuned CLEVR [6] dataset while preserving performance on MS-COCO [9], whereas conventional fine-tuning of Grounding DINO [10] leads to forgetting on MS-COCO.

Model	CLEVR [6]		MS-COCO [9]	
	Before	After	Before	After
Grounding DINO [10]	23.4	91.0 $\uparrow 67.6$	41.1	11.8 $\downarrow 29.3$
+ CSP [15] + CA (ours)		76.6 $\uparrow 53.2$		41.1 $= 0.0$

4.3 CZSL Comparison

We compare the CSP [15] baseline with our proposed method, Compositional Anticipation (CA) which extends CSP with Compositional Independence and Compositional Smoothing. The results, averaged over 13 experimental runs, are shown in Table 1 and are denoted using the NMS mAP metric [23]. Our results show that our method substantially improves upon the CSP baseline, with the harmonic mean (HM) between seen and unseen compositions improving by 70.5%. This improvement is predominately achieved on the unseen compositions, which improved by 66.1%.

Additionally, we showcase that our method does not suffer from catastrophic forgetting by evaluating its generalization ability compared to the conventional fine-tuning of Grounding DINO [10]. We compare the results on the MS-COCO [9] dataset before and after fine-tuning on the CLEVR [6] dataset for the CZSL task. Table 2 shows that conventional fine-tuning of Grounding DINO [10] achieves a 67.6% improvement on CLEVR, whereas our method achieves a 53.2% improvement. This suggests that conventional fine-tuning of Grounding DINO is superior in CZSL. However, conventional fine-tuning of Grounding DINO leads to a 29.3% performance drop on MS-COCO, whereas our method maintains stable performance with no drop at all. This demonstrates that conventional fine-tuning suffers from catastrophic forgetting, while our method does not.

4.4 Improving Incrementally

In this experiment, we incrementally learn new classes using the model initially trained with CSP [15] extended with Compositional Anticipation. We continue training the model using a dataset that includes both \mathcal{C}_p and \mathcal{C}_i .

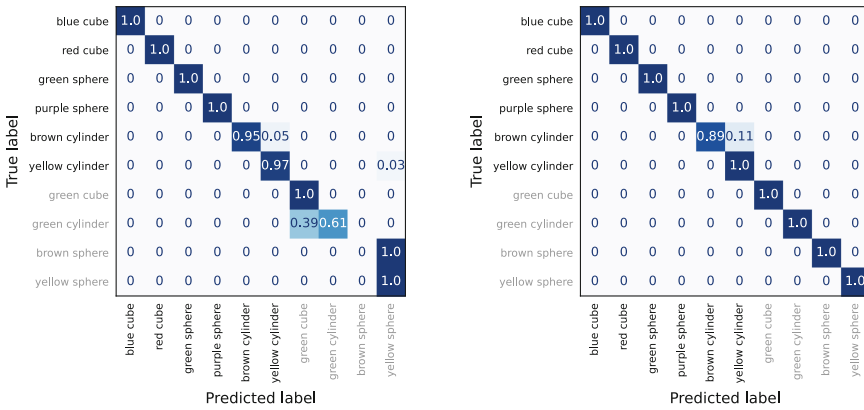
We explore two different fine-tuning methods: fine-tuning class-specific tokens and our proposed method, Contrastive Prompt Tuning. For fine-tuning class-specific tokens, we compare CSP [15] with CSP extended with Compositional Anticipation. Additionally, we conduct this fine-tuning in two ways: (1) allowing the fine-tuning of all tokens in the sets \mathcal{O} and \mathcal{A} , and (2) fine-tuning only objects and attributes present in \mathcal{C}_i , specifically \mathcal{O}_i and \mathcal{A}_i . For Contrastive Prompt Tuning, all tokens are frozen and only the prompt is fine-tuned. The prompt is initialized with semantically meaningful information, including both an affirmative and a negative component. For instance, if “green cylinder” is often confused with “green cube”, the prompt is initialized as “is not green cube but is green cylinder”. We also analyze the individual contributions of each component to the overall performance enhancements.

To evaluate performance, we compare the model’s results before and after introducing \mathcal{C}_i . Specifically, we determine performance across the sets \mathcal{C}_p , \mathcal{C}_i , and \mathcal{C}_u . Initially, \mathcal{C}_u is defined as $\mathcal{C} - \mathcal{C}_p$. After introducing \mathcal{C}_i , \mathcal{C}_u becomes $\mathcal{C} - \mathcal{C}_p - \mathcal{C}_i$. The results on these sets after introducing \mathcal{C}_i are shown in Table 3, with the absolute changes compared to the initial values indicated with arrows.

Our results show that our method, Contrastive Prompt Tuning, which fine-tunes a prompt initialized with prior knowledge to address specific mistakes

Table 3. Our Contrastive Prompt Tuning is effective for incremental learning. It improves performance across all classes including the unseen ones.

Class-Specific Tuning						
Method	Tunable Tokens	Pretrained	Increment	Unseen	HM	
CSP	$\mathcal{O} + \mathcal{A}$	88.1 ± 4.5 $\downarrow 2.9$	72.9 ± 20.3 $\uparrow 13.6$	0.0 ± 0.0 $\uparrow 74.3$	0.0 ± 0.0	$\uparrow 171.7$
+ CA	$\mathcal{O} + \mathcal{A}$	80.9 ± 5.0 $\uparrow 10.1$	60.5 ± 18.2 $\uparrow 11.2$	76.5 ± 8.4 $\uparrow 12.1$	69.6 ± 7.0	$\uparrow 12.1$
CSP	$\mathcal{O}_i + \mathcal{A}_i$	89.0 ± 2.3 $\uparrow 2.0$	64.1 ± 20.9 $\uparrow 14.8$	69.4 ± 8.3 $\uparrow 14.9$	70.6 ± 8.2	$\uparrow 11.1$
+ CA	$\mathcal{O}_i + \mathcal{A}_i$	89.2 ± 3.9 $\uparrow 1.8$	62.4 ± 19.6 $\uparrow 13.1$	78.6 ± 4.9 $\uparrow 14.3$	73.3 ± 9.1	$\uparrow 11.5$
Compositional Prompt Tuning (ours)						
Affirmation	Prompt	88.8 ± 5.5 $\uparrow 2.2$	82.8 ± 17.8 $\uparrow 23.5$	74.8 ± 6.6 $\uparrow 0.5$	80.2 ± 7.9	$\uparrow 8.5$
Negation	Prompt	91.2 ± 2.0 $\uparrow 0.2$	81.9 ± 13.8 $\uparrow 22.6$	76.6 ± 5.8 $\uparrow 2.3$	82.1 ± 6.0	$\uparrow 10.4$
Both	Prompt	92.6 ± 1.5 $\uparrow 1.6$	93.7 ± 2.1 $\uparrow 34.4$	75.2 ± 5.0 $\uparrow 0.9$	86.2 ± 2.1	$\uparrow 14.5$



(a) Before incremental learning (b) After incremental learning

Fig. 2. Our Contrastive Prompt Tuning method is effective in incremental learning. It improves performance on the increment compositions (in gray) while preserving performance of the pretrained compositions (in black). (Color figure online)

related to confusion between similar compositions, is superior to the class-specific tuning strategy. With Contrastive Prompt Tuning, we achieve a 12.9% enhancement in the HM across the pretrain, increment, and unseen sets compared to the best class-specific tuning method. This improvement is predominately achieved across the increment set, which improved by 31.3% compared to the best class-specific tuning method. Furthermore, Contrastive Prompt Tuning benefits from both the affirmative and negative components of the prompt.

Figure 2 shows the effects of Compositional Prompt Tuning on the model’s predictions. Figure 2a shows that before incremental learning, 39% of all instances of “green cylinder” are misclassified as “green cube”, and all instances of “brown sphere” are misclassified as “yellow sphere”. Figure 2b demonstrates that after applying Compositional Prompt Tuning, “green cube”, “green cylinder”, “yellow sphere”, and “brown sphere” are classified correctly on all instances.

5 Conclusion

In this paper, we demonstrated that conventional fine-tuning of Grounding DINO achieves an NMS mAP of 91.0 when fine-tuned on the CLEVR dataset for CZSL. However, this approach suffers from catastrophic forgetting, as confirmed by a 29.3% decrease in performance on the MS-COCO dataset post fine-tuning. To address this, we proposed incorporating CSP into Grounding DINO to mitigate forgetting by only fine-tuning auxiliary tokens. However, we observed that using CSP alone resulted in an NMS mAP of only 8.0 for the HM between seen and unseen compositions. Therefore, we extended CSP with Compositional Anticipation, which improved the HM by 70.5%. While our method improves upon the CSP baseline, it does not surpass conventional fine-tuning of Grounding DINO. Additionally, we introduced Contrastive Prompt Tuning to incrementally improve compositions that are confused with each other during training. With Contrastive Prompt Tuning, we improve performance on the HM across the pre-train, increment, and unseen sets by 12.9% compared to the best class-specific tuning method.

Given these findings, we recommend conventional fine-tuning of Grounding DINO for applications where performance on a specific dataset is prioritized, and our method for scenarios emphasizing overall performance across datasets. However, we acknowledge that our experiments are limited to the CLEVR dataset, and it remains unclear how the proposed methods will perform on real-world datasets beyond this toy dataset.

Considering that Grounding DINO excels in CZSL, likely due to the cross-modality fusion between image and text embeddings and our proposed methods involve strategically guiding the positioning of embeddings in the embedding space. Having demonstrated the benefits of our approach for CZSL, further investigation into the positioning of embeddings in the fused embedding space could potentially yield results approaching those achieved by conventional fine-tuning of Grounding DINO, but without encountering catastrophic forgetting.

References

1. Bang, J., Kim, H., Yoo, Y., Ha, J.W., Choi, J.: Rainbow memory: continual learning with a memory of diverse samples. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8218–8227 (2021)
2. Bang, J., Koh, H., Park, S., Song, H., Ha, J.W., Choi, J.: Online continual learning on a contaminated data stream with blurry task boundaries. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9275–9284 (2022)
3. Chen, K., et al.: Ovarnet: towards open-vocabulary object attribute recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 23518–23527 (2023)
4. Gretton, A., Fukumizu, K., Teo, C., Song, L., Schölkopf, B., Smola, A.: A kernel statistical test of independence. In: Advances in Neural Information Processing Systems, vol. 20 (2007)

5. Huang, S., Wei, Q., Wang, D.: Reference-limited compositional zero-shot learning. In: Proceedings of the 2023 ACM International Conference on Multimedia Retrieval, pp. 443–451 (2023)
6. Johnson, J., Hariharan, B., Van Der Maaten, L., Fei-Fei, L., Lawrence Zitnick, C., Girshick, R.: Clevr: a diagnostic dataset for compositional language and elementary visual reasoning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2901–2910 (2017)
7. Li, L.H., et al.: Grounded language-image pre-training. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10965–10975 (2022)
8. Li, X., Yang, X., Wei, K., Deng, C., Yang, M.: Siamese contrastive embedding network for compositional zero-shot learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9326–9335 (2022)
9. Lin, T.-Y., et al.: Microsoft COCO: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_48
10. Liu, S., et al.: Grounding Dino: marrying Dino with grounded pre-training for open-set object detection. arXiv preprint [arXiv:2303.05499](https://arxiv.org/abs/2303.05499) (2023)
11. Lu, X., et al.: DRPT: disentangled and recurrent prompt tuning for compositional zero-shot learning. arXiv preprint [arXiv:2305.01239](https://arxiv.org/abs/2305.01239) (2023)
12. Mancini, M., Naem, M.F., Xian, Y., Akata, Z.: Open world compositional zero-shot learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5222–5230 (2021)
13. Misra, I., Gupta, A., Hebert, M.: From red wine to red tomato: composition with context. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1792–1801 (2017)
14. Nagarajan, T., Grauman, K.: Attributes as operators: factorizing unseen attribute-object compositions. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11205, pp. 172–190. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01246-5_11
15. Nayak, N.V., Yu, P., Bach, S.H.: Learning to compose soft prompts for compositional zero-shot learning. arXiv preprint [arXiv:2204.03574](https://arxiv.org/abs/2204.03574) (2022)
16. Radford, A., et al.: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning, pp. 8748–8763. PMLR (2021)
17. Rao, Y., et al.: Denseclip: language-guided dense prediction with context-aware prompting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 18082–18091 (2022)
18. Ruis, F., Burghouts, G., Bucur, D.: Independent prototype propagation for zero-shot compositionality. In: Advances in Neural Information Processing Systems, vol. 34, pp. 10641–10653 (2021)
19. Saini, N., Pham, K., Shrivastava, A.: Disentangling visual embeddings for attributes and objects. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 13658–13667 (2022)
20. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2818–2826 (2016)
21. Wang, H., Yang, M., Wei, K., Deng, C.: Hierarchical prompt learning for compositional zero-shot recognition. In: Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, pp. 1470–1478 (2023)

22. Wang, Q., et al.: Learning conditional attributes for compositional zero-shot learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11197–11206 (2023)
23. Yao, Y., et al.: How to evaluate the generalization of detection? A benchmark for comprehensive open-vocabulary detection (2024)
24. Yao, Y., Zhang, A., Zhang, Z., Liu, Z., Chua, T.S., Sun, M.: CPT: colorful prompt tuning for pre-trained vision-language models. *AI Open* **5**, 30–38 (2024)
25. Zhang, J., Huang, J., Jin, S., Lu, S.: Vision-language models for vision tasks: a survey. *IEEE Trans. Pattern Anal. Mach. Intell.* (2024)
26. Zheng, Z., Zhu, H., Nevatia, R.: Caila: concept-aware intra-layer adapters for compositional zero-shot learning. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 1721–1731 (2024)
27. Zhou, D.W., Wang, Q.W., Qi, Z.H., Ye, H.J., Zhan, D.C., Liu, Z.: Deep class-incremental learning: a survey. arXiv preprint [arXiv:2302.03648](https://arxiv.org/abs/2302.03648) (2023)
28. Zhou, D.W., Zhang, Y., Ning, J., Ye, H.J., Zhan, D.C., Liu, Z.: Learning without forgetting for vision-language models. arXiv preprint [arXiv:2305.19270](https://arxiv.org/abs/2305.19270) (2023)
29. Zhou, K., Yang, J., Loy, C.C., Liu, Z.: Conditional prompt learning for vision-language models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 16816–16825 (June 2022)
30. Zhou, K., Yang, J., Loy, C.C., Liu, Z.: Learning to prompt for vision-language models. *Int. J. Comput. Vision* **130**(9), 2337–2348 (2022)



SPK: Semantic and Positional Knowledge for Zero-Shot Referring Expression Comprehension

Zetao Du¹, Jianhua Yang³, Junbo Wang⁴, Yan Huang^{2(✉)}, and Liang Wang²

¹ School of Information Science and Technology, ShanghaiTech University,
Shanghai 201210, China

duzt2022@shanghaitech.edu.cn

² Institute of Automation, Chinese Academy of Sciences (CASIA), Beijing, China

{yhuang, wangliang}@nlpr.ia.ac.cn

³ School of Artificial Intelligence, Beijing University of Posts and
Telecommunications, Beijing, China

youngjianhua@bupt.edu.cn

⁴ School of Software, Northwestern Polytechnical University, Xi'an, China

jbwang@nwpu.edu.cn

Abstract. Referring expression comprehension aims to localize an object in an image based on a natural language expression. This task is challenging due to the scarcity of large-scale annotated data, which prompts the research of zero-shot methods. In zero-shot scenarios, while existing models excel at grounding, they struggle to identify the target described by textual query due to the presence of multiple objects in a scene, as well as various spatial and attribute information. To address these issues, we propose a method called **Semantic and Positional Knowledge (SPK)**, which leverages multimodal knowledge for fine-grained cross-modal matching in the referring expression comprehension task. Specifically, we pair words with visual representations as multimodal knowledge to match the information of expressions and images, such as objects, attributes, and spatial information. This method can be directly integrated with existing multimodal grounding models for further performance improvement. Experiments on the RefCOCO+/g datasets demonstrate the effectiveness of our method, which can obtain consistent improvements.

Keywords: Multimodal Knowledge · Referring Expression
Comprehension (REC) · Zero-shot

1 Introduction

Referring expression comprehension(REC) [24] is a fundamental task in computer vision and natural language processing, requiring a model to localize the target object in the corresponding image according to the referring expression.

It has applications in various fields, including image content retrieval [6], robot interaction [47], image captioning [27, 32], etc. However, collecting large-scale and annotated data to train supervised models is expensive. It prompts researchers to find models pre-trained on widely available data and directly apply them to the REC task, thus alleviating the need for annotated data. This scenario is also known as zero-shot [25].

Due to the zero-shot generalization capacity of large-scale multimodal models [7, 10, 11, 25], some approaches [4, 28, 33] leverage them to serve as bridges connecting text and image domains. These approaches, however, fall short in fine-grained instance understanding [1], because they are pre-trained on coarse-grained image-text paired. This problem happens in ReCLIP [33], which pioneers the use of multimodal models [11, 25] for the zero-shot REC task, aligning individual image-text object pairs in a training-free manner. Though ReVLA [4] further pre-trains large-scale multimodal models on additional instance-level data to address the problem of data granularity mismatching, it requires significant computational resources and additional annotated data. Other models [12, 16, 23] like GLIP [12] are effective and scalable to learn instance-level and language-aware visual representations. However, these models also suffer from two issues when dealing with the zero-shot REC. Firstly, these models are pre-trained on datasets from domains different than REC. They merely perform simple grounding on objects rather than selecting the referred object among multiple grounded objects according to the description. Secondly, these models behave like “bags-of-words” [44] when tasks require fine-grained image-text understanding. This phenomenon indicates they can not handle the spatial information in referring expressions.

To deal with these two issues, a straightforward idea is to use domain-free knowledge to improve the generalization ability of pre-trained models. However, most of the existing knowledge-based methods are unimodal. They either leverage linguistic knowledge [29] to supplement the description or utilize image knowledge [22] to improve the visual representations. Although there are some works [19, 21, 30] that propose multimodal knowledge, they are designed for other tasks. Their effectiveness on the zero-shot REC task is unknown. More importantly, existing knowledge-based methods only focus on semantic knowledge, which focuses on mapping each word to an object. However, for the zero-shot REC, it is also necessary to model spatial knowledge, which focuses on distinguishing different words and different objects. Therefore, how to jointly model these two types of knowledge for the zero-shot REC is challenging and unstudied.

In this work, we propose an approach namely Semantic and Positional Knowledge (SPK) for zero-shot referring expression comprehension. The knowledge is collected from word-region pairs of public datasets and can bridge the images and texts in a fine-grained level. Specifically, for the modeling of semantic knowledge, we extract image region features of word-region pairs to obtain the semantic prototypes. The semantic prototypes can be used as knowledge matching to alleviate the mismatching between the referred text object and image region. For positional knowledge, we average all main object coordinates of the same

orientation in images to obtain the positional masks. The positional masks can be used as knowledge to represent the spatial information in referring expression. These two pieces of knowledge can be well-corporated to improve better the performance of pre-trained models [12, 16, 23]. In particular, the proposed SPK can achieve consistent improvements on the publicly available RefCOCO/+g datasets [20, 43] and outperform state-of-the-art approaches on RefCOCO/g.

Our contributions are summarized as follows: (1) We construct a new knowledge base containing both semantic and positional knowledge for multimodal alignment. (2) We propose the SPK method that leverages pre-defined knowledge to enhance the zero-shot performance of multimodal grounding models. (3) Our method achieves consistent improvements on the RefCOCO/+g datasets by re-ranking with multimodal grounding models.

2 Related Work

2.1 Referring Expression Comprehension

Numerous methods have been established for the REC task. They can be broadly categorized into one-stage [3, 13, 39] and two-stage [17, 34, 42] approaches. One-stage methods generate bounding boxes directly in an end-to-end manner, while two-stage methods utilize a proposal-query mechanism. In the zero-shot REC task, researchers naturally decompose the task into parsing and querying proposals steps [26], similar to two-stage methods. The parsing step often involves using existing grounding models to provide bounding boxes and external parsers to analyze the text. Then, the querying step leverages pre-trained models [7, 10, 11, 25] to align visual and language modalities. Building on this, CPT [41], ReCLIP [33], and ReVLP [4] harness the power of pre-trained models with strong visual-language alignment capabilities for the second step. Notably, there exists a traditional definition of zero-shot REC where prediction occurs on unseen objects during training, but training on a base dataset is still required. But in our setup, CPT [41], ReCLIP [33], and our method do not require retraining the pre-trained models. And different from these methods, our approach bridges the image-text modality from a knowledge modeling perspective.

2.2 Knowledge-Based Method

Leveraging knowledge has been applied in various vision-language tasks such as visual question answering [9, 30], image classification [21], and attempt reasoning [19]. Zhu et al. [50], Wang et al. [36], and Zhang et al. [46] have incorporated knowledge into visual-language tasks. However, only a few works introduce knowledge into visual grounding tasks. In traditional zero-shot scenarios, Singh et al. [31] encode external knowledge into image region proposals for object detection, while Zhan et al. [29] construct commonsense knowledge in the form of graphs, generating knowledge graphs with entities, relations, and objects as nodes, and utilizing this knowledge for zero-shot reasoning. These two methods

demonstrate different strategies for leveraging knowledge: embedding knowledge within the learning process during training [2, 37] or structuring knowledge as graphs to facilitate query-based access [18, 35, 51]. Unlike these two methods, we aim to leverage processed knowledge for instance-level image-text matching and avoid additional model training.

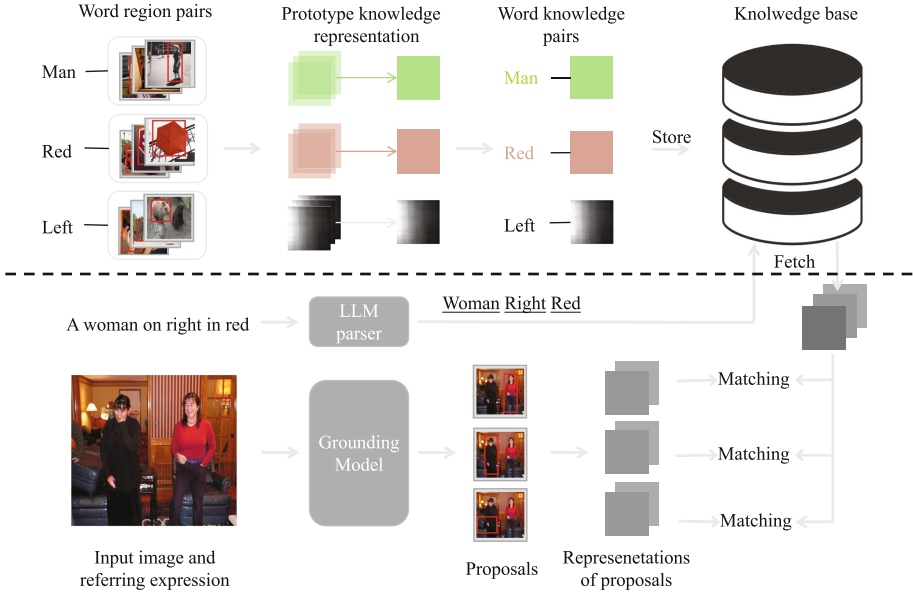


Fig. 1. Overview of SPK. The above details the construction of the knowledge base, while the below describes its reasoning. The **Matching** scores are used to re-rank the proposals.

3 Semantic and Positional Knowledge

As shown in Fig. 1, the proposed method SPK mainly consists of two parts: knowledge construction (above) and knowledge reasoning (below). When constructing knowledge, we incorporate semantic knowledge by collecting word-region pairs and extracting image-region features. Then, we obtain the semantic prototypes by averaging all image region features of the same semantic. These prototypes are then finetuned to improve discrimination. Additionally, we incorporate positional knowledge by collecting word-region pairs of orientation semantics, and we obtain positional knowledge by averaging all main object coordinates of the same orientation in images and projecting them to a mask.

During knowledge reasoning, we use Large Language Models (LLMs)¹ to parse referring expressions and retrieve relevant information from the knowledge

¹ We use Gemini as it has free and effective APIs.

base based on the description of the referred target. Subsequently, we assess the similarities between the features of proposals and the fetched semantic knowledge, as well as the response scores between the positions of proposals and the fetched positional knowledge. The proposals are generated by existing grounding models. Finally, both similarities and response scores are calculated as matching scores for re-ranking.

3.1 Knowledge Construction

Semantic Knowledge. We build the knowledge based on fine-grained image and text semantics, similar to unpaired image-text matching knowledge in MACK [5]. More specifically, it aligns semantic concepts with their corresponding visual features, establishing a one-to-one correspondence between textual and visual semantic concepts. Hence, the knowledge is represented as a set of semantic concepts with paired multi-modal representations, $(w_k, v_k), k = 1, \dots, K$, where w_k is k -th semantic concept and $v_k \in \mathbb{R}^F$ represent real-valued region representation of k -th semantic concept, K denotes the total number of semantic concepts. v_k can be computed as follows:

$$v_k = \frac{1}{J_k} \times \sum_{j=1}^{J_k} r_j \quad (1)$$

where J_k is the object number of k -th semantic concept and r_j is j -th feature of object belonging to k -th semantic concept. $r_j \in \mathbb{R}^F$ is obtained by applying global average pooling (GAP) [14] over the spatial dimensions (h, w) of $fr_j \in \mathbb{R}^{F \times h \times w}$, which is the output of the last layer of the feature extractor.

We compute the initial knowledge by the feature extractor in VinVL [48], which provides better features.

Positional Knowledge. We further extend our knowledge base to involve positional knowledge, which is absent in MACK [5]. When humans locate objects, they often form a preliminary impression of the image’s layout. If the description mentions “bottom left”, attention is automatically directed to the lower left corner of the image, implying heightened focus on that area. Based on this intuition, we incorporate positional knowledge into our knowledge base. Positional knowledge is divided into seven elements: top, bottom, left, right, front, back, and center, representing seven locations. Combinations of these elements can represent a broader range of locations.

To obtain the representation of positional knowledge, we utilize data from Visual Genome [8] containing predicates with spatial indications in format <subject predicate object>. In those data, the subject refers to the target, and the predicate indicates the position. The union of the subject and object regions is treated as the background and marked with 0; the subject’s position indicates the spatial position and is marked with 1. Then, a mask is formed by background(0) and subject position(1). After that, we accumulate all the masks within the same type of element to represent the positional knowledge for that



Fig. 2. From left to right, the elements represent above, under, left, right, front, and behind, respectively.

element. Figure 2 demonstrates the representation of the positional knowledge. When utilizing this positional knowledge, we project the mask to image and calculate the average response score within the region of the mask encompassed by proposals.

3.2 Knowledge Reasoning

Before utilizing knowledge, we employ an LLM to parse the referring expression. Unlike rule-based language parsers, the prompt's length greatly impacts the LLM's responses. The LLM may exhibit serious forgetting or logical confusion as the prompt length increases, leading to parsing failure. To avoid it, we designed a simple prompt to inquire about the primary entity within the referring expression, along with its descriptive words and orientation words, which are then used to query the knowledge base. Figure 3 shows an example of a prompt [4].

Mission Instruction: Given a sentence, first determine the main entity (with its attributes) that the sentence is referring to, it's the most important thing, think twice. And you should find all appearance entities (with its attributes) of main entity. Then find out the orientation words in the sentences.

Mission Details: Please return the answer in the JSON format. Like this format: {"main entity": "entity1", "appearance entities": ["appearance entity1", "appearance entity2"], "orientation": "orientation word1"} You should write the main entity and appearance entities with their attributes like "black T-shirt".

In-context Example: Given a sentence: the taller and black man wearing red T-shirt holding a black bag to the left of the door. Answer: {"main entity": "taller and black man", "appearance entities": ["red T-shirt", "black bag"], "other entities":["the door"],"orientation": "left"}

Your Instruction: **Your task sentence: #inputsentence. Please answer:**

Fig. 3. An example of the prompt for LLM.

Then, on the one hand, we retrieve the knowledge in the knowledge base by the entities and attributes parsed by LLMs. On the other hand, we aim to focus on the parts of the proposal's feature $\mathbf{x} \in \mathbb{R}^{h \times w \times F}$ that is described by the textual semantics. So, inspired by CAM [49], we calculate the knowledge-similar parts within the proposal's feature \mathbf{x} extracted by the feature extractor. We perform a weighted product on these knowledge-similar parts and apply the

weighted product matrix on the proposal’s feature \mathbf{x} , then use average pooling to the features after the weighting operation:

$$\mathbf{x}' = \frac{1}{hw} \sum_h \left(\sum_w (\mathbf{m}\mathbf{x}) \right), \quad \mathbf{m} = \text{softmax} \left(\sum_F (\mathbf{x}\mathbf{v}_k) \right) \quad (2)$$

where $\mathbf{m} \in \mathbb{R}^{h \times w}$ is a weighted product matrix, $\mathbf{x}' \in \mathbb{R}^F$ is the proposal’s feature after pooling.

Furthermore, we aim to fine-tune the knowledge to facilitate instance-level alignment between knowledge and features. Specifically, we introduce a linear layer to guide the fine-tuning process, where the refined knowledge \mathbf{v}'_k is obtained by $\mathbf{v}'_k = \sigma(\mathbf{W}\mathbf{v}_k)$, $\mathbf{W} \in \mathbb{R}^{F \times F}$, σ is activate function. Finally, we compute the cosine similarity between the fine-tuned knowledge \mathbf{v}'_k and the weighted regional features \mathbf{x}' to obtain the relevance score between feature and knowledge.

$$s_s = \text{cosine}(\mathbf{x}', \mathbf{v}'_k) \quad (3)$$

Besides semantic knowledge, when utilizing positional knowledge, we project the relevant mask to images and calculate the average response score within the region of the mask encompassed by proposals. Both semantic knowledge and positional knowledge are employed for re-ranking.

3.3 Knowledge Finetune

When performing a classification task, the features extracted by the feature encoder are fed into a multi-layer perceptron (MLP), yielding probabilities for specific classes. However, while these features perform well for classification, their inter-class discrimination for knowledge matching is unsatisfactory. For instance, opposite color semantics like black and white, despite being easily distinguished by the classifier, may be close in the feature vector space, hindering accurate differentiation in similarity calculations. To alleviate this, we employ contrastive learning to enhance the separation between knowledge. Specifically, we introduce a loss function termed Feature-Knowledge Contrastive (FKC) loss. It operates as follows: from each batch, we randomly select negative samples of two types. The first type consists of image features that do not match the currently queried knowledge, potentially sharing similarities with it. We call it soft negative samples. Soft negative loss can be calculated by positive knowledge \mathbf{v}'_p and negative image feature \mathbf{x}'_n .

$$\mathcal{L}_1^n = 1 - \text{cosine}(\mathbf{x}'_n, \mathbf{v}'_p) \quad (4)$$

The second type consists of knowledge that does not match the currently computed proposal feature, where we expect less similarity between the computed feature \mathbf{x}'_p and the mismatched knowledge \mathbf{v}'_n , we call it hard negative samples.

$$\mathcal{L}_2^n = 1 - \text{cosine}(\mathbf{x}'_p, \mathbf{v}'_n) \quad (5)$$

Ultimately, our contrastive learning loss can be formulated as:

$$\mathcal{L}^p = 1 - \text{cosine}(\mathbf{x}'_p, \mathbf{v}'_p) \quad (6)$$

$$\mathcal{L} = \max(\mathcal{L}^p - \mathcal{L}_1^n + \beta_1, 0) + \max(\mathcal{L}^p - \mathcal{L}_2^n + \beta_2, 0) \quad (7)$$

where β_1 and β_2 are hyper-parameters satisfying $\beta_1 < \beta_2$. The loss function enforces that the loss on the positive sample is at least β_1 smaller than the loss on the soft negative sample and at least β_2 smaller than the loss on the hard negative sample.

3.4 Knowledge Re-rank

Multimodal grounding models have evolved from multimodal image-text matching models. For example, GLIP [12] adapts CLIP [25] by replacing image-text pairs with bounding box-entity pairs. These models require extensive training on large datasets to achieve zero-shot capabilities for image-text understanding. What’s more, although they are trained on lots of instance-level object-text pairs but perform unsatisfying on the REC task in a zero shot manner. To improve this situation, we use SPK to complement existing grounding models. Specifically, current grounding models typically generate N proposals with corresponding scores S for each proposal, representing the model’s localization and confidence for the proposals. After the post-processing, we re-rank the proposals by combining the semantic scores s_s and position scores s_p with the original grounding model’s scores, i.e. $S + s_s + s_p$.

4 Experiment

4.1 Datasets

We evaluate our model on three referring expression comprehension datasets: RefCOCO [20], RefCOCO+, and RefCOCOg [43]. These datasets are derived from MSCOCO [15], providing images and referring expressions for identifying specific objects within the images. Each dataset exhibits distinct characteristics: RefCOCO comprises 19,994 images and 142,210 referring expressions, and its expressions emphasize simple feature descriptions and spatial relationships. RefCOCO+ contains 19,992 images and 141,564 referring expressions. It focuses on simple feature and state descriptions, excluding spatial relationships. RefCOCOg, with 25,799 images and 95,010 referring expressions, emphasizes diverse descriptions. Its expressions are more extended, averaging 8.43 words in length.

4.2 Implementation Details

In the pre-trained initial knowledge, we collected all adjectives and nouns from Visual Genome [19] as semantic concepts, resulting in a total of $K = 40,142$ concepts. The image features for each semantic concept are extracted using Faster R-CNN in VinVL [48], with each object feature having a dimension of $h \times w \times F = 7 \times 7 \times 2048$. During the fine-tuned phase, we perform the model training in batch size = 128; the optimizer is AdamW with a learning rate of $1e-5$ for 18 epochs. The super parameter $\beta_1 = 0.6$, $\beta_2 = 0.75$. Fine-tuning was performed on Visual Genome, excluding any data overlapping with the MSCOCO

dataset. The task accuracy refers to the percentage of instances where the proposal with the highest score has an Intersection over Union (IoU) of at least 0.5 with the ground truth.

4.3 Knowledge-Guided Re-ranking

We evaluate our approach on three visual grounding models, GLIP [12], GroundingDINO [16] and KOSMOS-2 [23]. It is important to note that the RefCOCO datasets were excluded from the training process of these models. During inference, they output N proposals, which are ranked based on their scores. The proposal with the highest score is selected as the target matching the referring expression. Our goal is to assess their zero-shot performance on REC, treating this as a zero-shot scenario due to the absence of training data overlap.

Table 1. Results on REC re-ranking three grounding models in a zero-shot manner. “+” denotes the re-ranking. SK and PK represent semantic knowledge and positional knowledge, respectively.

Method	RefCOCO			RefCOCO+			RefCOCOg	
	TestA	TestB	Val	TestA	TestB	Val	Test	Val
GLIP	54.69	43.06	49.96	53.44	43.42	49.01	66.08	65.58
+ SK	57.93	45.10	52.48	57.49	45.24	51.43	67.53	66.95
+ SK & PK	64.35	55.27	60.94	57.53	45.06	51.35	68.07	67.57
GroundingDINO	57.29	44.94	50.75	57.25	46.20	51.48	59.85	60.44
+ SK	59.18	45.85	52.59	59.19	47.58	53.76	68.36	67.46
+ SK & PK	69.12	60.75	64.96	59.40	48.21	53.78	70.12	69.24
Kosmos-2	57.41	47.26	52.32	50.73	42.24	45.48	60.57	60.57
+ SK	57.73	47.91	52.79	51.10	43.83	46.13	61.99	60.87
+ SK & PK	58.03	48.60	53.34	51.11	43.95	46.33	62.13	60.97

The results in Table 1 demonstrate consistent improvements across the RefCOCO datasets. Notably, we observed gains of 1–3% on RefCOCO, RefCOCO+, and RefCOCOg about GLIP. This suggests that our knowledge augmentation through referring expression decomposition is effective. Furthermore, re-ranking with scores derived from positional knowledge significantly boosted performance by about 10% on RefCOCO and 2% on RefCOCOg. The improvement is attributed to the abundance of location-related descriptions in these datasets, whereas RefCOCO+ lacks such spatial terms.

To demonstrate the generalizability of our knowledge-based approach, we apply it to the GroundingDINO, GLIP, and KOSMOS-2. We observe the patterns above across all three models, indicating the effectiveness of our method and its applicability to diverse grounding models.

4.4 Compare with SOTA Methods

We compare our approach with existing zero-shot methods in Table 2. We find that existing zero-shot methods achieve comparable performance to traditional supervised methods on RefCOCOg, but there is still a gap on RefCOCO+. Meanwhile, compared with existing zero-shot methods, our method can achieve 1%~6% improvement on RefCOCO by re-ranking on the more advanced GroundingDINO [16] model.

Compared with ReVLP [4], although it has additional training on CLIP [25] and thus gains significant improvement, it still cannot comprehensively outperform our method. Since our method does not model semantic relationships, it is not as good as SOTA on the RefCOCO+ dataset, which lacks spatial information.

Furthermore, even with SPK, the performance on Kosmos-2 [23] still cannot match GLIP and GroundingDINO. This is because the Kosmos-2 obtains the coordinate description of the proposals by generating text from the language model, which provides fewer proposals and thus limits the re-selection of re-ranking.

Table 2. Results comparing with other zero-shot manner. The highest score is in bold. Supervised SOTA refers to UNINEXT [38]. The Supervised method w/o VLP refers to TransVG [3].

Method	RefCOCO			RefCOCO+			RefCOCOg	
	TestA	TestB	Val	TestA	TestB	Val	Test	Val
Supervised SOTA [38]	94.33	91.46	92.64	89.63	79.79	85.24	89.37	88.73
Supervised w/o VLP [3]	82.72	78.35	81.02	70.70	56.94	64.82	67.73	68.67
CPT-Seg [40]	36.10	30.30	32.20	35.20	28.80	31.90	36.50	36.70
ReCLIP [33]	46.99	45.24	45.77	48.45	42.71	45.34	56.15	56.96
ReVLP w/o training	48.40	49.15	48.24	47.59	42.79	45.64	56.64	57.60
ReVLP [4]	66.52	54.86	60.62	62.56	45.69	55.52	59.90	59.87
GroundVLP [28]	61.30	43.53	52.58	64.77	47.43	56.38	63.54	64.30
Kosmos-2 + SPK	58.03	48.60	53.34	51.11	43.95	46.33	62.13	60.97
GLIP + SPK	64.35	55.27	60.94	57.53	45.06	51.35	68.07	67.57
GroundingDINO + SPK	69.12	60.75	64.96	59.40	48.21	53.78	70.12	69.24

4.5 Ablation Study

We conduct ablation study to verify the effectiveness of certain components in our method. The base model is GLIP.

Fine-Grained Alignment of Referring Objects. When performing knowledge matching, we parse the fine-grained information of the referring expression into entities and attributes. Table 3 presents the results on entities and attributes. Both entity and attribute information contribute to improved results. When we use only the target entity, the scores significantly increase. Combining both entity and attribute leads to further improvement. This demonstrates that semantic knowledge can serve as a bridge for aligning referential targets with image objects.

Table 3. Ablation study on the contributions of entity and attribute information. Ent, Attr, and Pos represent using entity, attribute, and orientation words to fetch knowledge in the knowledge base, respectively.

Ent	Attr	Pos	RefCOCO	RefCOCO+	RefCOCOfg
✓			56.76	55.92	66.74
	✓		54.81	53.20	65.98
✓	✓		57.93	57.49	67.53
✓	✓	✓	64.35	57.53	68.07

Effectiveness of Fine-Tuning. We aim to explore whether both the initial knowledge and the fine-tuned knowledge are effective. For comparison, we conducted two experiments: 1) Initial Knowledge: matching is performed using knowledge extracted directly from the feature extractor in VinVL [48]. 2) Fine-tuned Knowledge: knowledge is fine-tuned using cosine similarity as the loss function, with contrastive learning. Results in Table 4 denote that fine-tuning the knowledge is effective, but the improvement in scores is not substantial. This is partly because the initial knowledge is already effective for common objects, and the RefCOCO dataset contains relatively few uncommon objects. Moreover, larger values for β_1 and β_2 are preferred. As shown in Eq. 7, smaller values can lead to the convergence of positive and negative loss, hindering the training process.

Impact of Weighted Feature Extraction. At last, we investigate the effectiveness of weighted feature extraction. Table 5 presents results with and without weighted feature extraction. This method allows the model to focus on features relevant to the knowledge. We provide corresponding highlighted visualizations in Fig. 4 that show the detailed attention maps given to the target features.

4.6 Limitations

Our experiments are based on multimodal grounding models, which are more inclined towards grounding tasks compared to multimodal image-text models.

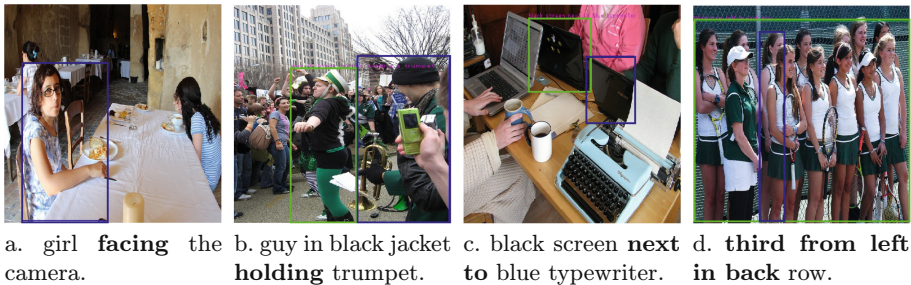
Table 4. Ablation study of knowledge finetuning.

Method	RefCOCO	RefCOCO+	RefCOCOG
Initial knowledge	57.22	56.14	67.42
Finetuned knowledge			
$\beta_1 = 0.30, \beta_2 = 0.45$	57.53	56.53	67.36
$\beta_1 = 0.60, \beta_2 = 0.75$	57.93	57.49	67.53

Table 5. Ablation study of weighted pooling on test parts.

Method	RefCOCO	RefCOCO+	RefCOCOG
w/o weighted pool	57.18	56.09	66.95
w weighted pool	57.93	57.49	67.53

This makes them sensitive to the precise location of specific targets, but their ability to understand semantics is limited. Although we use LLMs to parse irregular referring expressions, we do not model the semantic relationships that link objects and subjects. As illustrated in Fig. 5a and Fig. 5b, the model struggles to handle “facing” and “holding” relationships, leading to failures in these cases. While we model the spatial relationships in referring expressions, these are simple two-dimensional planar positions that cannot fully cope with complex real-world


Fig. 4. Visualization of model attention

Fig. 5. Incorrect examples, where blue box indicates ground truth, green box indicates prediction. (Color figure online)

positional descriptions. As illustrated in Fig. 5c and Fig. 5d, the model struggles to handle cases that require scene spatial understanding.

4.7 Conclusion

In this paper, we have introduced SPK, an approach designed to enhance the zero-shot performance of multimodal grounding models on the referring expression comprehension (REC) task. We have constructed a knowledge base encompassing semantic and positional knowledge, which is then integrated with existing multimodal grounding models. Our method effectively improves the performance of these models on the REC task without retraining.

However, there remains room for further improvement. A critical limitation is the lack of modeling relationships between two or more objects. Existing works like neural motifs [45] essentially rely on statistics to capture the co-occurrence probability of objects. However, using CNNs to extract holistic scene information while preserving individual object details is challenging. In future work, we will incorporate complex semantic relationships into our knowledge base, as well as, explore more effective ways to fuse the semantic and positional knowledge.

Acknowledgements. This work was jointly supported by the National Key R&D Program of China (2022ZD0116309), the National Natural Science Foundation of China (62236010, 62322607 and 62276261) and the Double First-Class Construction Foundation of China under Grant 23GH020227.

References

1. Bica, I., et al.: Improving fine-grained understanding in image-text pre-training. arXiv preprint [arXiv:2401.09865](https://arxiv.org/abs/2401.09865) (2024)
2. Chen, C., Li, O., Tao, D., Barnett, A., Rudin, C., Su, J.K.: This looks like that: deep learning for interpretable image recognition. In: Advances in Neural Information Processing Systems, vol. 32 (2019)
3. Deng, J., Yang, Z., Chen, T., Zhou, W., Li, H.: TranSVG: end-to-end visual grounding with transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 1769–1779 (2021)
4. Han, Z., Zhu, F., Lao, Q., Jiang, H.: Zero-shot referring expression comprehension via structural similarity between images and captions. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 14364–14374 (2024)
5. Huang, Y., Wang, Y., Zeng, Y., Wang, L.: Mack: multimodal aligned conceptual knowledge for unpaired image-text matching. In: Advances in Neural Information Processing Systems, vol. 35, pp. 7892–7904 (2022)
6. Jain, S., Pulaparthy, K., Fulara, C.: Content based image retrieval. *Int. J. Adv. Eng. Glob. Technol.* **3**(10), 1251–1258 (2015)

7. Jia, C., et al.: Scaling up visual and vision-language representation learning with noisy text supervision. In: International Conference on Machine Learning, pp. 4904–4916. PMLR (2021)
8. Krishna, R., et al.: Visual genome: connecting language and vision using crowd-sourced dense image annotations. *Int. J. Comput. Vision* **123**, 32–73 (2017)
9. Li, G., Wang, X., Zhu, W.: Boosting visual question answering with context-aware knowledge aggregation. In: Proceedings of the 28th ACM International Conference on Multimedia, pp. 1227–1235 (2020)
10. Li, J., Li, D., Xiong, C., Hoi, S.: Blip: bootstrapping language-image pre-training for unified vision-language understanding and generation. In: International Conference on Machine Learning, pp. 12888–12900. PMLR (2022)
11. Li, J., Selvaraju, R., Gotmare, A., Joty, S., Xiong, C., Hoi, S.C.H.: Align before fuse: vision and language representation learning with momentum distillation. In: Advances in Neural Information Processing Systems, vol. 34, pp. 9694–9705 (2021)
12. Li, L.H., et al.: Grounded language-image pre-training. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10965–10975 (2022)
13. Liao, Y., et al.: A real-time cross-modality correlation filtering method for referring expression comprehension. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10880–10889 (2020)
14. Lin, M., Chen, Q., Yan, S.: Network in network. arXiv preprint [arXiv:1312.4400](https://arxiv.org/abs/1312.4400) (2013)
15. Lin, T.-Y., et al.: Microsoft COCO: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_48
16. Liu, S., et al.: Grounding Dino: marrying Dino with grounded pre-training for open-set object detection. arXiv preprint [arXiv:2303.05499](https://arxiv.org/abs/2303.05499) (2023)
17. Liu, X., Li, L., Wang, S., Zha, Z.J., Su, L., Huang, Q.: Knowledge-guided pairwise reconstruction network for weakly supervised referring expression grounding. In: Proceedings of the 27th ACM International Conference on Multimedia, pp. 539–547 (2019)
18. Liu, Y., Li, H., Garcia-Duran, A., Niepert, M., Onoro-Rubio, D., Rosenblum, D.S.: MMKG: multi-modal knowledge graphs. In: Hitzler, P., et al. (eds.) ESWC 2019. LNCS, vol. 11503, pp. 459–474. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-21348-0_30
19. Lu, C., Krishna, R., Bernstein, M., Fei-Fei, L.: Visual relationship detection with language priors. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 852–869. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_51
20. Mao, J., Huang, J., Toshev, A., Camburu, O., Yuille, A.L., Murphy, K.: Generation and comprehension of unambiguous object descriptions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 11–20 (2016)
21. Marino, K., Salakhutdinov, R., Gupta, A.: The more you know: using knowledge graphs for image classification. arXiv preprint [arXiv:1612.04844](https://arxiv.org/abs/1612.04844) (2016)
22. Nauta, M., Van Bree, R., Seifert, C.: Neural prototype trees for interpretable fine-grained image recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 14933–14943 (2021)
23. Peng, Z., et al.: Kosmos-2: grounding multimodal large language models to the world. arXiv preprint [arXiv:2306.14824](https://arxiv.org/abs/2306.14824) (2023)
24. Qiao, Y., Deng, C., Wu, Q.: Referring expression comprehension: a survey of methods and datasets. *IEEE Trans. Multimedia* **23**, 4426–4440 (2020)

25. Radford, A., et al.: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning, pp. 8748–8763. PMLR (2021)
26. Sadhu, A., Chen, K., Nevatia, R.: Zero-shot grounding of objects from natural language queries. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4694–4703 (2019)
27. Sharma, H., Padha, D.: A comprehensive survey on image captioning: from hand-crafted to deep learning-based techniques, a taxonomy and open research issues. *Artif. Intell. Rev.* **56**(11), 13619–13661 (2023)
28. Shen, H., Zhao, T., Zhu, M., Yin, J.: GroundVLP: harnessing zero-shot visual grounding from vision-language pre-training and open-vocabulary object detection. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 38, pp. 4766–4775 (2024)
29. Shi, Z., Shen, Y., Jin, H., Zhu, X.: Improving zero-shot phrase grounding via reasoning on external knowledge and spatial relations. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 36, pp. 2253–2261 (2022)
30. Singh, A.K., Mishra, A., Shekhar, S., Chakraborty, A.: From strings to things: knowledge-enabled VQA model that can read and reason. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4602–4612 (2019)
31. Singh, K.K., Divvala, S., Farhadi, A., Lee, Y.J.: DOCK: detecting objects by transferring common-sense knowledge. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11217, pp. 506–522. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01261-8_30
32. Stefanini, M., Cornia, M., Baraldi, L., Cascianelli, S., Fiameni, G., Cucchiara, R.: From show to tell: a survey on deep learning-based image captioning. *IEEE Trans. Pattern Anal. Mach. Intell.* **45**(1), 539–559 (2022)
33. Subramanian, S., Merrill, W., Darrell, T., Gardner, M., Singh, S., Rohrbach, A.: Reclip: a strong zero-shot baseline for referring expression comprehension. arXiv preprint [arXiv:2204.05991](https://arxiv.org/abs/2204.05991) (2022)
34. Sun, M., Xiao, J., Lim, E.G., Liu, S., Goulermas, J.Y.: Discriminative triad matching and reconstruction for weakly referring expression grounding. *IEEE Trans. Pattern Anal. Mach. Intell.* **43**(11), 4189–4195 (2021)
35. Sun, R., et al.: Multi-modal knowledge graphs for recommender systems. In: Proceedings of the 29th ACM International Conference on Information and Knowledge Management, pp. 1405–1414 (2020)
36. Wang, S., Yue, J., Liu, J., Tian, Q., Wang, M.: Large-scale few-shot learning via multi-modal knowledge discovery. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12355, pp. 718–734. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58607-2_42
37. Xu, W., Xian, Y., Wang, J., Schiele, B., Akata, Z.: Attribute prototype network for zero-shot learning. In: Advances in Neural Information Processing Systems, vol. 33, pp. 21969–21980 (2020)
38. Yan, B., et al.: Universal instance perception as object discovery and retrieval. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 15325–15336 (2023)
39. Yang, Z., Gong, B., Wang, L., Huang, W., Yu, D., Luo, J.: A fast and accurate one-stage approach to visual grounding. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4683–4693 (2019)
40. Yao, Y., Zhang, A., Zhang, Z., Liu, Z., Chua, T.S., Sun, M.: CPT: colorful prompt tuning for pre-trained vision-language models. arXiv preprint [arXiv:2109.11797](https://arxiv.org/abs/2109.11797) (2021)

41. Yao, Y., Zhang, A., Zhang, Z., Liu, Z., Chua, T.S., Sun, M.: CPT: colorful prompt tuning for pre-trained vision-language models. *AI Open* **5**, 30–38 (2024)
42. Yu, L., et al.: Mattnet: modular attention network for referring expression comprehension. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1307–1315 (2018)
43. Yu, L., Poirson, P., Yang, S., Berg, A.C., Berg, T.L.: Modeling context in referring expressions. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) *ECCV 2016*. LNCS, vol. 9906, pp. 69–85. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46475-6_5
44. Yuksekogonul, M., Bianchi, F., Kalluri, P., Jurafsky, D., Zou, J.: When and why vision-language models behave like bags-of-words, and what to do about it? In: *The Eleventh International Conference on Learning Representations* (2023)
45. Zellers, R., Yatskar, M., Thomson, S., Choi, Y.: Neural motifs: scene graph parsing with global context. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5831–5840 (2018)
46. Zhang, B., Hu, H., Qiu, L., Shaw, P., Sha, F.: Visually grounded concept composition. *arXiv preprint [arXiv:2109.14115](https://arxiv.org/abs/2109.14115)* (2021)
47. Zhang, C., Chen, J., Li, J., Peng, Y., Mao, Z.: Large language models for human-robot interaction: a review. *Biomimetic Intell. Robot.* 100131 (2023)
48. Zhang, P., et al.: VinVL: revisiting visual representations in vision-language models. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5579–5588 (2021)
49. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2921–2929 (2016)
50. Zhu, X., et al.: Multi-modal knowledge graph construction and application: a survey. *IEEE Trans. Knowl. Data Eng.* **36**(2), 715–735 (2022)
51. Zhu, Y., Zhang, C., Ré, C., Fei-Fei, L.: Building a large-scale multimodal knowledge base system for answering visual queries. *arXiv preprint [arXiv:1507.05670](https://arxiv.org/abs/1507.05670)* (2015)



Can Language Improve Visual Features For Distinguishing Unseen Plant Diseases?

Jerad Zherui Liaw¹(✉), Abel Yu Hao Chai¹, Sue Han Lee¹, Pierre Bonnet²,
and Alexis Joly³

¹ Swinburne University of Technology Sarawak Campus, Kuching, Malaysia
101234758@students.swinburne.edu.my, {aychai,shlee}@swinburne.edu.my

² CIRAD, UMR AMAP, Montpellier, France
pierre.bonnet@cirad.fr

³ INRIA, Montpellier, France
alexis.joly@inria.fr

Abstract. Deep learning approaches have been pivotal in identifying multi-plant diseases, yet they often struggle with unseen data. The challenge of handling unseen data is significant due to the impracticality of collecting all disease samples for every plant species. This is attributed to the vast number of potential combinations between plant species and diseases, making capturing all such combinations in the field difficult. Recent approaches aim to tackle this issue by leveraging a zero-shot compositional setting. This involves extracting visual characteristics of plant species and diseases from the seen data in the training dataset and adapting them to unseen data. This paper introduces a novel approach by incorporating textual data to guide the vision model in learning the representation of multiple plants and diseases. To our knowledge, this is the first study to explore the effectiveness of a vision-language model in multi-plant disease identification, considering the fine-grained and challenging nature of disease textures. We experimentally prove that our proposed FF-CLIP model outperforms recent state-of-the-art models by 26.54% and 33.38% in Top-1 accuracy for unseen compositions, setting a solid baseline for zero-shot plant disease identification with the novel vision-language model. We release our code at <https://github.com/abelchai/FF-CLIP-Can-Language-Improve-Visual-Features-For-Distinguishing-Unseen-Plant-Diseases>.

Keywords: Unseen Plant Disease Identification · Zero-Shot · Vision Language Model

J. Z. Liaw and A. Y. H. Chai—These authors contributed equally to this work.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-78113-1_20.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
A. Antonacopoulos et al. (Eds.): ICPR 2024, LNCS 15330, pp. 296–311, 2025.
https://doi.org/10.1007/978-3-031-78113-1_20

1 Introduction

Pathogenic organisms like fungi, bacterium, mycoplasma, viruses, viroid, nematodes, or parasitic flowering plants are disease vectors, causing significant damage that can result in major yield losses, particularly in agronomy. Identifying plant diseases is a fundamental challenge for the general public, whose knowledge is limited, and for botanists and agronomists, who are experts in their field but not necessarily in plant pathologies. Furthermore, the task grows increasingly complex due to the vast number of plant species [11] and diseases worldwide, alongside the escalating risk of disease spread facilitated by globalization. Traditional methods of disease identification, based on manual inspection and expert knowledge, are not only time-consuming but also costly and are limited by geographical constraints and the availability of competent expertise [1]. In agriculture, there is an urgent need for rapid, accessible disease diagnosis. This identification plays a crucial role as a reference point, enabling farmers to implement appropriate mitigation measures promptly [6], thus avoiding substantial losses due to plant damage [10].

Deep learning (DL), a subset of machine learning, has emerged as an important approach in this field, driven by diverse datasets to learn discriminative features for plant diseases [15, 17, 24]. It has brought a new paradigm shift in the field of multi-plant disease identification, allowing faster and large-scale diagnosis. Researchers have recently started conceptualising multi-plant disease representation as compositions comprising individual plant and disease concepts. Plant-disease pairs found in the training data are termed as “seen compositions”, while those not present are termed as “unseen compositions”. Notably, individual concepts of the unseen compositions remain within the training dataset. The common approach to multi-plant disease identification tasks is learning the individual concepts’ features corresponding to their compositions [1, 26]. In

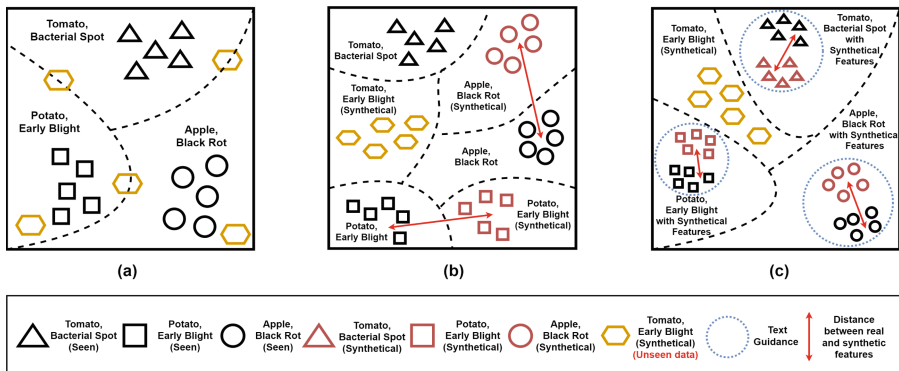


Fig. 1. (a) Depiction of traditional supervised discriminative models where features are not primarily learned to represent unseen data. (b) The recent FF-ViT [3] enhances the composition diversity of the data to encompass unseen data purely based on visual cues. It is effective, but the generated synthetic data is far from the real data distribution. (c) The proposed model aims to reduce the gap between the synthetic features and the real data distribution based on combined visual and textual language cues.

contrast, Lee et al. [14] have introduced a new framework called Conditional Multi-Task Learning (CMTL), which simultaneously learns individual plant and disease classifiers with a conditional link between them. CMTL can identify unseen plant disease compositions, provided that the individual plant or disease concept has been learned based on previously seen compositions. This is important as obtaining a comprehensive range of all plant disease compositions in real-world scenarios is not feasible.

This paper is also further motivated by the emerging trend of visual disentanglement, particularly in the context of Compositional Zero-Shot Learning (CZSL) [9, 30]. This involves breaking down visual concepts into smaller, more manageable parts and then combining them to recognize new compositions. Notably, the significance of visual disentanglement in CZSL has spurred other researchers to explore its application in the domain of plant disease analysis. Chai et al. [3] introduced a method using information from seen compositions to help the model understand unseen compositions better, improving recognition of new compositions. However, relying solely on visual cues poses challenges in identifying plant diseases [7, 27]. This is because plant diseases often appear as subtle variations in leaf texture, colour, shape, and overall morphology, unlike distinct objects typically encountered in general object recognition.

Recent work by El Banani et al. [5] proposed incorporating language guidance into the learning process to leverage descriptions capturing conceptual similarities between images. The emergence of Contrastive Language–Image Pre-training (CLIP) has further underscored the importance of integrating visual and language modalities in various domains [2, 16]. It leverages large-scale datasets comprising paired images and text captions to learn joint representations of images and corresponding textual descriptions. This prompts the question: *Can language enhance visual features for such fine-grained unseen plant disease identification?*

Inspired by a recent approach [20] demonstrating the potential of deploying CLIP in zero-shot compositional learning, we test our hypothesis by incorporating the concept of re-purposing joint pre-trained vision-language models into our zero-shot plant disease composition task. We show that exploiting textual descriptions can further improve the performance of zero-shot compositional tasks, outperforming state-of-the-art plant disease identification models. In summary, this paper makes two significant contributions. Firstly, our research introduces a novel approach that demonstrates the effectiveness of language cues in guiding visual features to improve the identification of unseen plant diseases. Secondly, we show that our proposed method enhances the identification performance of unseen plant disease compositions, surpassing state-of-the-art techniques in multiple plant disease identification tasks.

2 Related Work

2.1 Multi-plant Disease Identification

Deep learning has proven effective in multi-plant disease identification tasks. Existing multi-plant disease identification’s mainstream methodologies focus on transforming this issue into a general supervised recognition task [17, 24]. For instance,

Lee et al. [13, 15] tackled this challenge by training a classifier oriented towards visual symptoms, concentrating solely on diseases without considering plant species. Other approaches, as presented in [14], propose various configurations for learning plant species and disease features, employing either two-headed classifiers (separate plant and disease concepts) or single-headed classifiers (plant disease compositions). However, considerable room remains for enhancing model performance, particularly for compositions with limited or zero training data. This is crucial as, in real-world scenarios, collecting all plant disease samples exhaustively is not feasible due to the immense diversity of plant species and diseases worldwide.

Recently, Chai et al. [3] explored the application of CZSL [18] techniques in the domain by introducing the Feature Fusion Vision Transformer (FF-ViT) model with pairwise feature learning for unseen plant disease identification. This approach leverages the visual features of seen compositions and employs a feature fusion (FF) strategy to generate synthetic data, facilitating the acquisition of knowledge to identify seen and unseen compositions. However, since synthetic data lacks inherent relationships with real-world data, the performance of unseen data may be affected, leading to potential out-of-distribution issues, as illustrated in Fig. 1 (b). Therefore, this paper introduces a novel approach to enhance visual cues using language guidance. The idea is to improve the features of both the seen and unseen data by projecting them into a feature space that better aligns with the real data distribution, as shown in Fig. 1 (c). To the best of our knowledge, our study is the first to explore the potential of language cues in zero-shot plant disease identification.

2.2 Vision Language Models for Zero-Shot Classification

To address the limited representation of solely relying on visual features, we aim to enhance the feature representation of plants and diseases by incorporating language guidance into the learning process. In a previous study, Frome et al. [8] introduced DeVISE, a novel deep vision-semantic embedding model. DeVISE learns to identify visual objects by leveraging both labelled image data and semantic information extracted from unannotated text. Their model demonstrates its capability to enhance zero-shot predictions for labels unseen by the visual model.

Built on foundations in zero-shot transfer, natural language supervision, and multi-modal learning, the recently proposed CLIP [21] associates images with corresponding textual descriptions, allowing it to infer information based on natural language supervision. The introduction of CLIP-based visual language models marked a significant advancement, showcasing remarkable performance in zero-shot classification tasks by efficiently learning visual concepts from textual descriptions, making it highly adaptable across diverse tasks and domains [2, 16]. Two main methodologies have emerged to improve classification tasks by taking advantage of the CLIP model. Firstly, the CLIP model is used to refine the descriptive captions associated with images. This approach, illustrated by previous studies such as [25, 29], use the CLIP model to generate descriptive captions, enriching the information content of composition labels. Secondly, the use of the CLIP model as a guiding condition for other models. This is demonstrated

in recent studies where the CLIP model serves as conditioning information to guide generative models such as VQGAN [4] and decoder [23]. These approaches aim to produce more diverse images while retaining photorealism. Inspired by previous studies, our study utilises the CLIP model to enrich our composition labels while guiding our visual model to extract features that are closer towards real data distributions.

3 Dataset

We use the PlantVillage (PV) dataset [19], known as the largest publicly available dataset covering multiple plant species and diseases, to evaluate the performance of all models. The PV dataset comprises 38 distinct plant disease compositions, C , encompassing a total of 54,305 images. In line with previous [15, 19] studies, we divided the dataset into an 80% training dataset and a 20% testing dataset. In accordance with the experimental setup from [13, 15] for unseen plant disease identification, we strategically excluded the *Pepper bell_bacterial spot* class from the training data, while retaining *Pepper bell_healthy*. The *Pepper bell_healthy* samples drove our FF-CLIP model to generate synthetic *Pepper bell_bacterial spot* composition. Subsequently, we separated the testing dataset into the seen testing dataset (37 different plant disease compositions, with all compositions available in the training dataset) and the unseen testing dataset (only *Pepper bell_bacterial spot*). Given that the samples in the PV dataset have uniform background characteristics, our study focuses on examining the generalizability of our model by transferring knowledge learned from seen data to obtain effective performance on both seen and unseen data.

4 Methods

This section outlines our problem formulation regarding unseen plant disease identification tasks. We then delve into the details of our novel Feature Fusion Contrastive Language-Image Pre-Training (FF-CLIP) model and our training strategy.

4.1 Problem Formulation

This study conceptualises our multi-plant disease identification tasks as compositional tasks in which each sample, denoted $C = (P, D)$, encapsulates two distinct concepts. Specifically, we define these concepts as plant concept, $P = (p_0, p_1, \dots, p_m)$, and disease concept, $D = (d_0, d_1, \dots, d_n)$, where m and n represent the total number of unique plant entities and diseases in the training dataset respectively. The total unique plant and disease composition can be defined as $m \times n$. Furthermore, we split all available data into three subsets: a training dataset denoted by C^t , a seen testing dataset denoted by C^s and an unseen testing dataset denoted by C^u . Notably, the compositions present in C^u

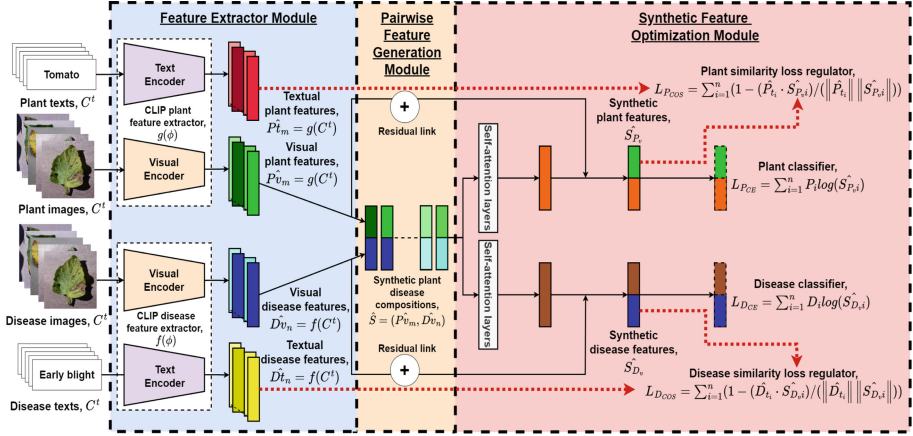


Fig. 2. This is the overview of the FF-CLIP model architecture, comprising three main modules. The first module is the feature extractor module, utilizing CLIP models to extract visual and textual features from input text-image pairs for both plant and disease concepts. The second module, the pairwise feature generation module, sourced from previous work [3], generates synthetic features of plant disease compositions using visual features from the previous module. The third module, the synthetic feature optimization module, consists of multiple self-attention layers and residual links to extract synthetic plant and disease features. The FF-CLIP model is trained with both cross-entropy and cosine similarity losses.

is not present in C^t and C^s , while all plant, P , and disease, D , concepts present in C^u are present in C^t and C^s . We refer to the compositions, C , available in C^s as seen data and the others as unseen data. The main goal of our study on unseen plant disease identifications is to efficiently recognize the two compositions in C^s and C^u using the knowledge acquired in C^t .

4.2 Feature Fusion Contrastive Language-Image Pre-training (FF-CLIP) Model

Our research is motivated by the framework proposed by [3], in which the model demonstrates the ability to generate synthetic features for both seen and unseen plant disease identification tasks. In this study, we observed that the previous model, as described in [3], relying solely on visual input, might not adequately capture relevant real data distributions from the local distribution within the training dataset. We then propose a new approach known as the Feature Fusion Contrastive Language Image Pre-Learning (FF-CLIP) model. This model exploits textual information as external features to guide the original and synthetic feature learning process, facilitating the acquisition of a generalized distribution that closely aligns with the real data distribution.

The architecture of our novel FF-CLIP model, as shown in Fig. 2, can be separated into three different modules, which are the feature extractor module,

pairwise feature generation module and synthetic feature optimization module. The feature extractor module extracts both visual and textual features from the training dataset, which are then fed into the pairwise feature generation module. This module generates both seen and unseen synthetic features. The synthetic feature optimization module optimizes the feature output from the pairwise feature generation module for the identification task.

Feature Extractor Module. The feature extractor module acts as a disentangler, separating the data into two concepts: plant and disease. Previous studies in [12, 14] adopt separate models to extract the visual features of plant and disease concepts. In the architecture of our FF-CLIP model, we include an additional cue, the natural language description, forming the two distinct vision-textual concepts: plants vision-textual and disease vision-textual concepts. We employ the CLIP model proposed by [22] as the backbone for both visual and textual feature extraction. The rationale behind choosing CLIP lies in its design to optimize both visual and textual extractors simultaneously, enabling seamless integration of visual and textual features related to plant and disease concepts. Moreover, CLIP is pre-trained on a vast dataset consisting of 400 million image-text pairs, a crucial advantage in our study, where our training dataset is limited.

Specifically, we use two CLIP models (ViT-B/32) from [22] as plant feature extractor, $g(\phi)$ and disease feature extractor, $f(\phi)$ for both visual and textual input. Each model consists of 12 transformer layers with 12 attention heads for visual input and 8 attention heads for textual input. Both models produce embeddings with a 512-dimensional feature space. These models are pre-trained on the WebImageText dataset as mentioned in [22], which comprises 400 million image-text pairs. The image-text pairs from our training dataset are used as input for both CLIP models. The plant feature extractor, $g(\phi)$ will extract original visual plant features, $P_{v_m} = g(C^t)$ and textual plant features, $\hat{P}_{t_m} = g(C^t)$. The disease feature extractor, $f(\phi)$ will extract original visual disease features, $D_{v_n} = f(C^t)$ and textual disease features, $\hat{D}_{t_n} = f(C^t)$.

Pairwise Feature Generation Module. The pairwise feature generation module is used to generate synthetic visual features of seen and unseen composition from the original visual plant, \hat{P}_{v_m} and disease, \hat{D}_{v_n} features. The synthetic data, denoted as S , encompasses a broader range of compositions compared to the available compositions within the original training dataset, C , where S contains $m \times n$ number of compositions. In particular, the module obtains the original visual features of the plant, \hat{P}_{v_m} and the disease, \hat{D}_{v_n} , from the feature extraction module. These two inputs are then combined using different feature fusion strategies to generate synthetic features of plant disease compositions, $\hat{S} = (\hat{P}_{v_m}, \hat{D}_{v_n})$.

For example, when the feature extraction module obtains *Potato_early blight* and *Corn_common rust* as input from the original training dataset. It extracts *Potato* and *Corn* as original plant visual features, and *Early blight* and *Common rust* as original disease visual features. The synthetic plant disease com-

positions from the pairwise feature generation module will consist of 4 compositions, that is, *Potato_early blight*, *Corn_common rust*, *Potato_common rust* and *Corn_early blight*, where *Potato_common rust* and *Corn_early blight* are unseen compositions. All synthetic plant disease compositions will be used as input for the following synthetic feature optimization module.

Synthetic Feature Optimization Module. The primary objective of our synthetic feature optimization module is to enhance synthetic plant and disease features, aligning the learned distributions of our model more closely with the real data distributions. To determine the optimal weight, we perform an optimization by gradient descent, minimizing the cross-entropy loss of the Concepts-Oriented Classifiers. Simultaneously, we perform Text-Guided Visual Modelling to bring the visual cues closer to the language cue through cosine similarity loss.

- **Concepts-Oriented Classifiers.** To train the classifier, we first deploy multiple attention layers to extract synthetic plant features, \hat{S}_{P_v} and disease features, \hat{S}_{D_v} . In addition, we introduce the residual link to regularize the classifier’s objective so that it aligns with the target’s original visual distribution. This is illustrated in Fig. 2 with residual links. The synthetic plant and disease features can be defined as $\hat{S}_{P_v} = Att_1(\hat{S})$ and $\hat{S}_{D_v} = Att_2(\hat{S})$. Next, we use two linear classifiers to refine the decision boundaries for plant and disease concepts, optimizing their discriminative ability through cross-entropy loss. The cross-entropy loss of the plant and disease classifier can be formulated as *Plant classifier loss*, $L_{PCE} = \sum_{i=1}^n P_i \log(S_{P_v,i})$ and *Disease classifier loss*, $L_{DCE} = \sum_{i=1}^n D_i \log(S_{D_v,i})$ where P_i and D_i represent the truth labels for the i^{th} sample of the plant and disease concepts respectively, and $\hat{S}_{P_v,i}$ and $\hat{S}_{D_v,i}$ are the i^{th} sample of the synthetic plant and disease features respectively.
- **Text-Guided Visual Modelling.** To further minimize the divergence between our synthetic and real-world data distributions, we utilize the textual features extracted by the feature extraction module to guide the synthetic data distribution. More specifically, we consider the distributions of seen training data as references for real-world data distributions. Thus, we calculate the cosine similarity between the synthetic visual features and textual features extracted from the seen data distribution for the plant and disease concept. The similarity loss is therefore formulated as *Plant similarity loss*, $L_{PCOS} = \sum_{i=1}^n (1 - (\hat{P}_{t_i} \cdot \hat{S}_{P_v,i}) / (\|\hat{P}_{t_i}\| \|\hat{S}_{P_v,i}\|))$ and *Disease similarity loss*, $L_{DCOS} = \sum_{i=1}^n (1 - (\hat{D}_{t_i} \cdot \hat{S}_{D_v,i}) / (\|\hat{D}_{t_i}\| \|\hat{S}_{D_v,i}\|))$ where \hat{P}_{t_i} and \hat{D}_{t_i} are the i^{th} sample of the plant and disease textual features, respectively, by the feature extraction module. \hat{S}_{P_v} and \hat{S}_{D_v} are the i^{th} sample of the synthetic plant and disease features respectively.

Table 1. Performance comparison between SOTA models and our proposed model on seen and unseen plant disease identification for PV dataset.

Model	Seen Top 1	Unseen Top 1	Harmonic Mean
ViT single network	99.52	4.17	8.00
CMTL-ViT [14]	99.49	6.94	12.97
FF-ViT (single head) [3]	99.58	19.44	32.53
FF-ViT (dual head) [3]	99.67	15.28	26.50
CLIP [22]	98.66	18.83	31.62
FF-CLIP (dual head)	99.22	41.82	58.84

Single head model performs plant and disease identifications with a single classifier. Conversely, the dual head model performs plant and disease identifications with different classifiers and derives plant disease identification through post-predictions.

4.3 Training Strategy

In this section, we will discuss in detail all the hyperparameters and training schemes for the FF-CLIP model. The model is trained end to end with a learning rate of 0.001. We use an SGD optimiser with a momentum of 0.9 and weight decay of 0.00001. We use NVIDIA A100 80 GB GPUs.

The model consists of three modules: feature extraction module, pairwise feature generation module and synthetic feature optimization module. First, using cross-entropy loss, the model learns two linear classifiers for the plant and disease concepts. The total classifier loss can be defined as $L_{CE} = L_{P_{CE}} + L_{D_{CE}}$ where $L_{P_{CE}}$ and $L_{D_{CE}}$ are defined in Sect. 4.2. Secondly, the model aligns the synthetic distribution with the real data distribution with a cosine loss of similarity. Total cosine similarity loss can be defined as $L_{COS} = L_{P_{COS}} + L_{D_{COS}}$ where $L_{P_{COS}}$ and $L_{D_{COS}}$ are also from Sect. 4.2. We assign α and β as weighting coefficients to regulate between the two losses. As a result, the final loss function for our FF-CLIP model can be defined as:-

$$L_{final} = \alpha(L_{CE}) + \beta(L_{COS}) \quad (1)$$

5 Experimental Results and Discussions

In this section, we first conduct a comprehensive performance analysis of our proposed FF-CLIP model with various SOTA models on both seen and unseen plant disease identification tasks. Subsequently, we examine the similarity between synthetic distribution and original distribution within similar architecture. Next, we present an in-depth exploration of our FF-CLIP model through different ablation studies.

5.1 Comparison Between Different SOTA

In Table 1, we compare the performance of our novel FF-CLIP model with various SOTA models. In addition, we use the harmonic mean to balance the accuracy of seen and unseen classes, highlighting the model’s generalizability to unseen data without being overly influenced by its performance on seen data [28]. For CLIP, we use two models: one for plant classification and another for disease classification. We obtain the accuracy for each model separately and then perform post-processing to derive the final results.

Our FF-CLIP (dual head) model significantly outperforms all other models in the unseen task, achieving the highest accuracy of 41.82% and maintaining comparable performance on the seen task with an accuracy of 99.22%. Notably, our novel architecture improved upon the pre-trained CLIP model (ViT-B/32) [22], especially in the unseen task, by a margin of 22.99%. This shows that our FF-CLIP model with an additional feature fusion pairwise module is able to learn more generalized features for both seen and unseen tasks. This is probably due to the fact that the features learned from the CLIP model exhibit high generalization but may lack the fine details needed to address the challenging nature of disease textures. Our study validates that incorporating a feature generation schema and synthetic feature optimization module enables us to discern these finer details, distinguishing visually similar plant and disease concepts more effectively.

In addition, we observed that while both the ViT single-network and CMTL-ViT [14] models demonstrate excellent performance on seen tasks, their efficiency decreases significantly when dealing with unseen tasks, achieving only 4.17% and 6.94% respectively for the unseen task. This observation underscores the critical role of our pairwise feature generation method, which enriches the composition diversity of the training samples. This enrichment enables the learned feature space to capture the features of the seen compositions and those of the unseen compositions. As a result, our FF-CLIP model outperforms the ViT single-network and CMTL-ViT models on the unseen task with a significant margin of 37.65% and 34.88%, respectively.

Furthermore, our FF-CLIP (dual head) model outperformed the recent FF-ViT (dual head) and FF-ViT (single head) [3] models with a margin of 26.54% and 33.38% respectively on the unseen task. This shows that while FF-CLIP and FF-ViT models can generate synthetic unseen data, incorporating textual features as a guide in our architecture results in synthetic data distribution that closely resembles real data distribution, effectively narrowing the performance gap between seen and unseen tasks. The harmonic mean results further prove the superior generalizability of FF-CLIP, as it outperforms both FF-ViT (single head) and FF-ViT (dual head) by significant margins of 26.31% and 32.34%, respectively. To quantitatively analyse the distributing gap, we conducted a similarity check between the synthetic distribution and original distribution for both FF-CLIP (dual head) and FF-ViT (dual head) models, the two best-performing models in the following section.

5.2 Analysis of Distribution Gap

Table 2. Similarity score between synthetic distribution and original distribution for both plant and disease concepts.

Model	Plant concepts	Disease concepts
	similarity score	similarity score
FF-ViT	0.1653	0.1255
FF-CLIP	0.4012	0.2954

Similarity scores range between -1 to 1 . Higher scores represent closer distributions.

Similarity scores range between -1 to 1 . Higher scores represent closer distributions.

This section aims to evaluate the similarity between synthetic and original distributions for FF-ViT and FF-CLIP models. While both models are capable of synthesizing data, the synthetic data produced by FF-CLIP is further refined using textual features, potentially enhancing its alignment with the original data. Specifically, the distribution obtained from the feature extraction module is referred to as the original distribution since it is obtained directly from the original visual plant, P_{v_m} and disease, D_{v_n} features. Next, we derive the synthetic distribution from the synthetic plant, S_{P_v} and disease, S_{D_v} features from the synthetic feature optimization module. We compare the original and synthetic features for all images within the testing dataset to obtain both distributions. Specifically, the plant concept similarity scores are defined as *Plant concept similarity score* = $\sum_{i=1}^n (P_{v_{mi}} \cdot S_{P_{vi}}) / (\|P_{v_{mi}}\| \|S_{P_{vi}}\|)$ and disease concept similarity scores as *Disease concept similarity score* = $\sum_{i=1}^n (D_{v_{ni}} \cdot S_{D_{vi}}) / (\|D_{v_{ni}}\| \|S_{D_{vi}}\|)$. $P_{v_{mi}}$ and $D_{v_{ni}}$ are the i^{th} sample of the original visual plant and disease features, respectively. $S_{P_{vi}}$ and $S_{D_{vi}}$ are the i^{th} sample of the synthetic plant and disease features respectively.

From Table 2, FF-CLIP has a higher similarity score for the plant and disease concepts, with a score of 0.4012 and 0.2954, respectively, compared to the FF-ViT model. This observation underlines that while both models are capable of mapping original visual images into their synthetic feature space via the pairwise feature generation module, the synthetic feature space resulting from the FF-ViT model may lack robustness and present a potential bias in favour of the seen compositions. In contrast, the FF-CLIP model exploits textual features as a guide between the original and synthetic distributions, enabling the model to exploit additional features beyond visual features. By leveraging the knowledge distilled from many image-text pairs, textual features potentially encompass salient information that can improve the model’s discrimination capabilities,

Table 3. Comparison between different feature fusion strategies

Feature Fusion	Seen Top 1	Unseen Top 1
Concatenation	99.06	34.26
Summation	99.12	29.48
Multiplication	99.13	24.85

These feature fusion strategies are performed in the pairwise feature generation module.

Table 4. Influence of different weighting coefficients

α	β	Seen Top 1	Unseen Top 1
1.0	0.0	99.16	26.16
1.0	0.5	99.16	31.64
1.0	1.0	99.06	34.26
0.5	1.0	99.22	41.82

α and β are weighting coefficient in Eq. 1.

particularly by distinguishing visually similar concepts. As a result, the FF-CLIP model generates synthetic distributions that show better alignment with the original distribution. This alignment is crucial, enabling the synthetic distribution to more effectively encapsulate the unseen distributions. This is also demonstrated by the result in Table 1, where FF-CLIP outperforms the FF-ViT model on the unseen task with an accuracy advantage of 26.54%.

6 Ablation Studies

In this section, we present the empirical evaluation of our FF-CLIP model with various ablation studies. Besides, we also include some studies of our model in the supplementary materials.

6.1 Comparison Between Feature Fusion Strategies

In Table 3, we analyze the impact of different feature fusion strategies. The results demonstrate that while all three techniques perform comparably in the seen task, concatenation strategies outperform others in the unseen task with a larger margin, achieving the highest accuracy of 34.26%. On the other hand, the multiplication strategy shows the lowest accuracy of 24.85%. This difference is likely due to the fact that while the multiplication strategy can amplify crucial features and dampen irrelevant ones, such as background noise, this specificity may unintentionally exclude generalized information crucial for unseen tasks. Conversely, the concatenation strategy can better retain all features, ensuring a comprehensive approach that is particularly effective in unseen tasks.

6.2 The Importance of Weighting Coefficients

In this experiment, we analyze the impact of the weighting coefficients, α and β , in Eq. 1. These parameters regulate the importance of the cross-entropy loss and cosine similarity loss components in our FF-CLIP model. In Table 4, the FF-CLIP model using only cross-entropy loss ($\alpha = 1.0$ and $\beta = 0.0$) performs lowest on unseen tasks, with an accuracy of 26.16%. However, a shift towards the cosine similarity loss component ($\alpha = 0.5$ and $\beta = 1.0$) yields substantial performance improvements, outperforming the previous configuration with accuracy gains of 0.06% and 15.66%, respectively. These results underline the importance of cosine similarity loss, which leverages textual features to enable the model to generate synthetic data that not only discriminates effectively between classes but also closely aligns with the distribution of real data.

Table 5. Comparative analysis for different input texts.

Model	Seen Top 1	Unseen Top 1
Description Text	98.96	29.94
Label Text	99.06	34.26

The description text reflects the characteristics and specificities of each disease label.

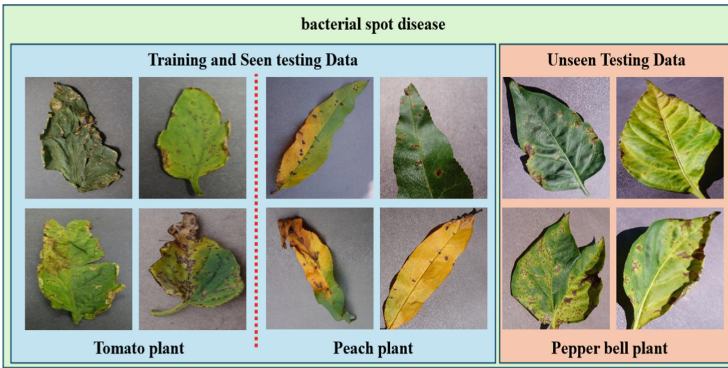


Fig. 3. The figure above shows various visual symptoms of bacterial spot disease which present distinct visual appearances from one plant species to another.

6.3 Analysis of Text Prompts

In Table 5, we evaluate the performance of the FF-CLIP model with different input texts. According to [21], the performance of the CLIP model can be improved by using an appropriate prompt for the label text. These templates

help overcome the polysemy challenge, whereby a text can have several possible meanings. By using the prompt, the model can effectively filter out textual features irrelevant to the domain, thus refining its scope. Therefore, we conducted a similar study on the FF-CLIP model, replacing original label text with disease descriptions from Bugwood.org¹. We detail the description text in the supplementary material.

However, we observe a decrease in the performance of our FF-CLIP model when using these descriptions compared to the original text. Specifically, the FF-CLIP model with disease descriptions shows accuracy decreases of 0.10% and 1.32% on seen and unseen tasks, respectively, compared to its counterpart using the label text. This may be due to the fact that disease descriptions, which describe characteristics and specificities, are too specific and not general enough. In fact, disease symptoms can differ based on environmental conditions and the growth stage of the disease, adding complexity. As a result, the text descriptions may lack the ability to encompass the broad visual appearance of symptoms associated with different plant species. Figure 3 shows the variability in the visual appearance of bacterial spot symptoms across different plants.

7 Conclusion

Our study highlights the significant impact of incorporating text guidance with visual features, particularly in distinguishing unseen plant diseases, which rely heavily on fine-grained features. FF-CLIP demonstrates superior performance over various state-of-the-art (SOTA) models in zero-shot plant disease identification, emphasizing the effectiveness of textual features in enhancing the visual representation of plant diseases.

Limitation. In some cases, we found that the improved performance of unseen tasks was associated with a degradation in the performance of seen tasks.

Future Work. Future work should focus on finding appropriate descriptions and achieving a balance where both seen and unseen compositions achieve optimal performance without sacrificing either. To address this challenge, it is essential to acquire extensive data and expertise to overcome the barriers of knowledge.

Acknowledgments. We appreciate the comments and advice from Hervé Goëau and Fei Siang Tay on our study and drafts. This research is supported by the FRGS MoHE Grant (Ref: FRGS/1/2021/ICT02/SWIN/03/2) from the Ministry of Higher Education Malaysia and Swinburne Sarawak Research Grant (Ref: RIF SSRG-Tay Fei Siang(30/12/24)). We gratefully acknowledged the support of NEUON AI for GPU workstation used for this research.

References

1. Ahmad, A., El Gamal, A., Saraswat, D.: Toward generalization of deep learning-based plant disease identification under controlled and field conditions. *IEEE Access* **11**, 9042–9057 (2023)

¹ <https://wiki.bugwood.org/>.

2. Cao, Y., Chen, L., Yuan, Y., Sun, G.: Cucumber disease recognition with small samples using image-text-label-based multi-modal language model. *Comput. Electron. Agric.* **211**, 107993 (2023). <https://doi.org/10.1016/j.compag.2023.107993>, <https://www.sciencedirect.com/science/article/pii/S0168169923003812>
3. Chai, A.Y.H., et al.: Pairwise feature learning for unseen plant disease recognition. In: 2023 IEEE International Conference on Image Processing (ICIP), pp. 306–310. IEEE (2023)
4. Crowson, K., et al.: VQGAN-clip: open domain image generation and editing with natural language guidance. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) ECCV 2022. LNCS, vol. 13697, pp. 88–105. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-19836-6_6
5. El Banani, M., Desai, K., Johnson, J.: Learning visual representations via language-guided sampling. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 19208–19220 (2023)
6. Fan, X., Luo, P., Mu, Y., Zhou, R., Tjahjadi, T., Ren, Y.: Leaf image based plant disease identification using transfer learning and feature fusion. *Comput. Electron. Agric.* **196**, 106892 (2022). <https://api.semanticscholar.org/CorpusID:247968352>
7. Feng, X., Zhao, C., Wang, C., Wu, H., Miao, Y., Zhang, J.: A vegetable leaf disease identification model based on image-text cross-modal feature fusion. *Front. Plant Sci.* **13** (2022). <https://api.semanticscholar.org/CorpusID:250034475>
8. Frome, A., et al.: Devise: a deep visual-semantic embedding model. In: Burges, C., Bottou, L., Welling, M., Ghahramani, Z., Weinberger, K. (eds.) *Advances in Neural Information Processing Systems*, vol. 26. Curran Associates, Inc. (2013)
9. Hao, S., Han, K., Wong, K.Y.K.: Learning attention as disentangler for compositional zero-shot learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 15315–15324 (2023)
10. Hassan, S.M., Maji, A.K.: Plant disease identification using a novel convolutional neural network. *IEEE Access* **10**, 5390–5401 (2022)
11. Joly, A., Bonnet, P., Affouard, A., Lombardo, J.C., Goëau, H.: Pl@ntnet - my business. In: Proceedings of the 25th ACM international conference on Multimedia (2017). <https://api.semanticscholar.org/CorpusID:34644257>
12. Lee, S.H., Chan, C.S., Remagnino, P.: Multi-organ plant classification based on convolutional and recurrent neural networks. *IEEE Trans. Image Process.* **27**(9), 4287–4301 (2018)
13. Lee, S.H., Goëau, H., Bonnet, P., Joly, A.: Attention-based recurrent neural network for plant disease classification. *Front. Plant Sci.* **11**, 1897 (2020)
14. Lee, S.H., Goëau, H., Bonnet, P., Joly, A.: Conditional multi-task learning for plant disease identification. In: 2020 25th International Conference on Pattern Recognition (ICPR), pp. 3320–3327. IEEE (2021)
15. Lee, S.H., Goëau, H., Bonnet, P., Joly, A.: New perspectives on plant disease characterization based on deep learning. *Comput. Electron. Agric.* **170**, 105220 (2020). <https://doi.org/10.1016/j.compag.2020.105220>, <https://www.sciencedirect.com/science/article/pii/S0168169919300560>
16. Ma, Y., Xu, G., Sun, X., Yan, M., Zhang, J., Ji, R.: X-clip: end-to-end multi-grained contrastive learning for video-text retrieval. In: Proceedings of the 30th ACM International Conference on Multimedia (2022). <https://api.semanticscholar.org/CorpusID:250607505>
17. Maurya, R., Pandey, N.N., Singh, V.P., Gopalakrishnan, T.: Plant disease classification using interpretable vision transformer network. In: 2023 International Conference on Recent Advances in Electrical, Electronics and Digital Healthcare

- Technologies (REEDCON), pp. 688–692 (2023). <https://api.semanticscholar.org/CorpusID:259179406>
18. Misra, I., Gupta, A., Hebert, M.: From red wine to red tomato: composition with context. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1792–1801 (2017)
 19. Mohanty, S.P., Hughes, D.P., Salathé, M.: Using deep learning for image-based plant disease detection. *Front. Plant Sci.* **7**, 1419 (2016)
 20. Nayak, N.V., Yu, P., Bach, S.: Learning to compose soft prompts for compositional zero-shot learning. In: The Eleventh International Conference on Learning Representations (2023). <https://openreview.net/forum?id=S8-A2FXnIh>
 21. Radford, A., et al.: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning (2021). <https://api.semanticscholar.org/CorpusID:231591445>
 22. Radford, A., et al.: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning, pp. 8748–8763. PMLR (2021)
 23. Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., Chen, M.: Hierarchical text-conditional image generation with clip latents (2022)
 24. Shoaib, M.A., et al.: An advanced deep learning models-based plant disease detection: a review of recent research. *Front. Plant Sci.* **14** (2023). <https://api.semanticscholar.org/CorpusID:257678708>
 25. Tewel, Y., Shalev, Y., Schwartz, I., Wolf, L.: Zerocap: zero-shot image-to-text generation for visual-semantic arithmetic. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 17897–17907 (2022). <https://doi.org/10.1109/CVPR52688.2022.01739>
 26. Thakur, P.S., Sheorey, T., Ojha, A.: VGG-ICNN: a lightweight CNN model for crop disease identification. *Multimedia Tools Appl.* **82**, 497–520 (2022). <https://api.semanticscholar.org/CorpusID:249479235>
 27. Wang, C., et al.: A plant disease recognition method based on fusion of images and graph structure text. *Front. Plant Sci.* **12** (2022). <https://api.semanticscholar.org/CorpusID:245908252>
 28. Yi, K., Elhoseiny, M.: Domain-aware continual zero-shot learning. *arXiv abs/2112.12989* (2021). <https://api.semanticscholar.org/CorpusID:245502766>
 29. Yu, J., Li, H., Hao, Y., Zhu, B., Xu, T., He, X.: CGT-GAN: clip-guided text GAN for image captioning. In: Proceedings of the 31st ACM International Conference on Multimedia (2023). <https://api.semanticscholar.org/CorpusID:261076397>
 30. Zhang, Y., Jia, Q., Fan, X., Liu, Y., He, R.: CSCnet: class-specified cascaded network for compositional zero-shot learning (2024). <https://api.semanticscholar.org/CorpusID:268349050>



Show Me the World in My Language: Establishing the First Baseline for Scene-Text to Scene-Text Translation

Shreyas Vaidya¹(✉), Arvind Kumar Sharma¹, Prajwal Gatti²,
and Anand Mishra¹

¹ Indian Institute of Technology Jodhpur, Jodhpur, India
{vaidya.2,sharma.126,mishra}@iitj.ac.in

² University of Bristol, Bristol, UK
prajwal.gatti@bristol.ac.uk

Abstract. In this work, we study the task of “visually” translating scene text from a source language (e.g., Hindi) to a target language (e.g., English). Visual translation involves not just the recognition and translation of scene text but also the generation of the translated image that preserves visual features of the source scene text, such as font, size, and background. There are several challenges associated with this task, such as translation with limited context, deciding between translation and transliteration, accommodating varying text lengths within fixed spatial boundaries, and preserving the font and background styles of the source scene text in the target language. To address this problem, we make the following contributions: (i) We study visual translation as a standalone problem for the first time in the literature. (ii) We present a cascaded framework for visual translation that combines state-of-the-art modules for scene text recognition, machine translation, and scene text synthesis as a baseline for the task. (iii) We propose a set of task-specific design enhancements to design a variant of the baseline to obtain performance improvements. (iv) Currently, the existing related literature lacks any comprehensive performance evaluation for this novel task. To fill this gap, we introduce several automatic and user-assisted evaluation metrics designed explicitly for evaluating visual translation. Further, we evaluate presented baselines for translating scene text between Hindi and English. Our experiments demonstrate that although we can effectively perform visual translation over a large collection of scene text images, the presented baseline only partially addresses challenges posed by visual translation tasks. We firmly believe that this new task and the limitations of existing models, as reported in this paper, should encourage further research in visual translation. We have publicly released the code and dataset on our project website: <https://vl2g.github.io/projects/visTrans/>.

Keywords: Visual Translation · Scene Text Synthesis · Evaluation Metrics.

S. Vaidya and A. K. Sharma—Equal Contribution.

P. Gatti—This work was done while Prajwal Gatti was affiliated with IIT Jodhpur.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
A. Antonacopoulos et al. (Eds.): ICPR 2024, LNCS 15330, pp. 312–328, 2025.
https://doi.org/10.1007/978-3-031-78113-1_21

1 Introduction

Machine Translation has shown remarkable growth in the last few years, partly attributed to the adoption of neural models [2, 6, 7, 30, 34]. In parallel, substantial advancements have also been made in speech-to-speech translation [11, 12, 17, 24] where the goal is to develop systems that are capable of accurately interpreting spoken language in one dialect and seamlessly translating it into another while preserving the voice of the original speaker, thus enabling effective cross-lingual communication in real-time. Drawing inspiration from these research directions, we present an analogous problem in the scene text domain, namely “scene-text to scene-text translation” or, in short, “visual translation”. The visual translation task aims to translate text present in images from a source language to the target language while preserving the visual characteristics of the text and background as illustrated in Fig. 1. Visual Translation has extensive applications, e.g., transforming the travel experience by allowing tourists to instantly understand sign boards in foreign languages and enabling seamless interaction with the visual world without language barriers.



Fig. 1. Imagine visiting Delhi, India, and arriving at the Rithala (Hindi: रिठाला) metro station. If you are not familiar with Hindi, the signboard on the left might be incomprehensible. The result of our proposed baseline solution, shown on the right, seamlessly transliterates the station name रिठाला to English. In our work, we aim to visually translate (or transliterate, when necessary, as in this case) text from the source language to the target language while preserving the visual attributes of the source scene text. Specifically, we focus on visual translation between Hindi and English in this work.

By drawing parallels with the speech-to-speech translation approaches, which comprise three components: automatic speech recognition (ASR), text-to-text machine translation (MT), and text-to-speech (TTS) synthesis, we propose a visual translation baseline that integrates scene-text recognition (STR), text-to-text machine translation (MT), and scene-text synthesis (STS). This cascaded system offers practical advantages over an end-to-end approach, as fully supervised end-to-end training necessitates a substantial collection of source and target scene text pairs, which can be challenging to obtain compared to parallel text pairs for MT or image-text pairs for STR. As STR and MT models are extensively explored in the literature, and several off-the-shelf methods are available, we prioritize enhancing the performance of the STS model. To this end, we extend a popular SRNet architecture [31] by decoupling background and foreground generation. For background generation, we employ a diffusion-

based model using ControlNet [33] to generate a text-erased image from an input containing scene texts. We further modify SRNet so that it only focuses on foreground generation on a plain background. Once foreground and background are independently generated, we blend them into the scene image. To improve the quality of visual translation further, we propose a set of design enhancements such as using regular expressions to filter special strings, grouping words and translating them together, and a planning strategy to blend the translated text in the scene image appropriately.

We extensively evaluate the proposed baselines for Hindi-to-English and English-to-Hindi visual translation using our new automatic and user evaluation metrics. While the baselines show promising results, the problem remains far from solved and requires further research.

We make the following contributions: (i) We study the under-explored task of visual translation that aims to translate text in images to a target language while preserving its font, style, position, and background. To the best of our knowledge, the comprehensive study of this problem has largely been unexplored in the existing literature. (ii) We introduce a generic cascaded approach for visual translation, and we design a set of baselines using state-of-the-art approaches for scene text recognition, machine translation, and scene text synthesis and their task-specific design enhancements. (iii) Training a visual translation model with real-world images is challenging due to the lack of large-scale paired scene text images in different languages. Therefore, we use synthetic images for training. We present a method to generate paired images with words sharing the same visual properties, creating VT-SYN, a synthetic dataset of 600K paired visually diverse English-Hindi scene-text images. To evaluate performance on real images, we provide extensive annotations of translated text from three users. These benchmark datasets will support future research in visual translation. (iv) Due to the lack of principled evaluation metrics for visual translation tasks in the literature, we propose a set of automatic and user evaluation metrics. We believe these metrics will help track the progress of visual translation tasks effectively.

2 Related Work

Machine Translation: It is a well-studied area [5, 6, 9, 10, 26, 30] that aims to convert a text from its source language to a target language. Current state-of-the-art models for machine translation are deep-learning based [6, 9, 30]. In the speech domain, Speech-to-Speech Translation (S2ST) aims to translate speech from one language to another while preserving the speaker’s voice and accent [11, 12, 17]. Inspired by these works, we focus on text translation in the visual modality, which brings newer research challenges, such as preserving font properties and integrity of the image background, which need to be addressed to produce visually appealing translations.

Translation of Text in Images: Recent years have seen growing interest in translating text within images, both in research and commercial domains. Current works primarily focus on recognition and translation methods for scene

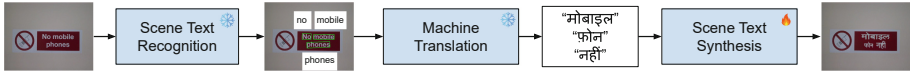


Fig. 2. Outline of proposed cascaded baseline for Visual Translation. We use state-of-the-art approaches for scene text recognition, machine translation, and scene text synthesis to design variants of our baseline. Moreover, we further investigate the scene-text synthesis and propose an extension to existing SRNet architecture.

text [16, 20], prioritizing accurate translation without addressing visually consistent text generation. A popular commercial product – Google lens¹ also falls into this category. These approaches often resort to simply overlaying translated text on source images. While some studies explore end-to-end methods for text translation in images that generate text directly in pixel space [15, 22], they typically deal with limited visual diversity in document-style images with plain backgrounds and fixed fonts – without tackling the complexities of scene text that we aim to address. The closest solution to our problem is Google Translate for images², a commercial product for visual translation of diverse scene text. However, its underlying technology remains proprietary and closed-source. We emphasize the need for the research community to study this problem openly, establish proper open-source solutions, create a benchmark, and define evaluation criteria – goals we pursue in this paper. Moreover, we observe that Google Translate still lacks translation quality and often fails to produce visually consistent results for complex cases, underlining the potential for better approaches. In Section 6.2, we provide a qualitative comparison between our work and Google Translate.

Editing Text in Images: The problem of editing text in images has witnessed significant research interest in recent years [14, 18, 27, 28, 31, 32]. This task aims to modify scene text to target content while retaining visual properties of the original text. SRNet [31] is one such method that learns an end-to-end GAN-based style-retention network. SwapText [32] improved upon the SRNet architecture by modeling the geometrical transformation of the text using spatial points. More recently, TextStyleBrush [14], RewriteNet [18], and MOSTEL [27] introduce a self-supervised training approach on real-world images for this task. Further, TextStyleBrush is evaluated on handwritten images as well. Authors in [28] proposed a character-wise text editor model for this task. However, their approach assumes source and target text instances are of the same length, which is not always true, especially in the translation task. A more recent approach, MOSTEL [27], also introduces stroke-level modifications to produce more photo-realistic generations. Despite these advances, these methods only address the cross-lingual editing problem, which is just a component of the visual translation process and is insufficient on its own for achieving visual translation. Our work aims to address the task of visual translation and its complexities more comprehensively.

¹ <https://lens.google/#translate>

² <https://translate.google.com/?op=images>

3 Proposed Visual Translation Baseline

The task of Visual Translation can be reduced to a sequence of sub-tasks: locating and reading text in scene images, translating the text into the target language, and generating the final image containing the translated scene text. Motivated by this observation, we propose a cascaded approach to visual translation by combining models for (i) scene text recognition, (ii) machine translation, (iii) scene-text synthesis, and (iv) seamlessly blending the generated scene text into the image. These sub-tasks are well-explored independently in computer vision literature; thus, we benefit from the availability of trained models. Further, such an approach can perform generation at the word or phrase level, which can help preserve the consistency of non-text regions in the image. The outline of our cascaded baseline is illustrated in Fig. 2 and a detailed illustration is provided in Fig. 3. We describe each module in detail in the following sections.

Training and evaluation of a visual translation baseline require real-world images in the form of (I, I') where I and I' are visually identical images containing corresponding scene text in two different languages with matching font and style. However, such instances are not easily available in the real world. We mitigate this data scarcity challenge by directly *synthesizing* the desired data: we generate paired scene-text images that are (i) identical in the image background and (ii) matching in font and style. A few examples are shown in Fig. 4.

In generating synthetic samples, we use a large corpus of words in both languages, as well as a diverse collection of fonts. To simulate real-world scene text, we also render the images on natural backgrounds, as well as vary the orientation, positioning, and size of the scene text in images. A more detailed procedure for generating the synthetic data is provided in Section 4.1.

(i) Locating and Recognizing Text in Images. The first step in our proposed baseline is locating scene text in images, followed by recognizing the detected text, which are both well-explored problems in computer vision literature. Given the source image, we use a scene-text detector to detect all occurrences of text in the image by predicting a bounding box around them. Next, we use a text recognition model that predicts the text content from the crops of words obtained from the previous step. In this work, we use DBNet [19] for text detection and ParSeq [3] for the text recognition step pretrained on English and Hindi language data, respectively.

(ii) Machine Translation of Text. After obtaining the recognized text in the source language L , we map each instance to the desired target language L' using an off-the-shelf neural machine translation method. We test our model with two state-of-the-art neural machine translation methods, namely IndicTrans2 [8] and M2M100 [7]. IndicTrans2 is trained on a large collection of Indic languages (including Hindi), whereas M2M100 is a more general translation method trained on a diverse collection of languages with support for Indic languages as well.

(iii) Scene-Text Synthesis. Our pipeline has thus far obtained source word bounding boxes, recognized text, and translated text. The final step is to generate the target word image containing the translated text while maintaining stylistic consistency with the source text.

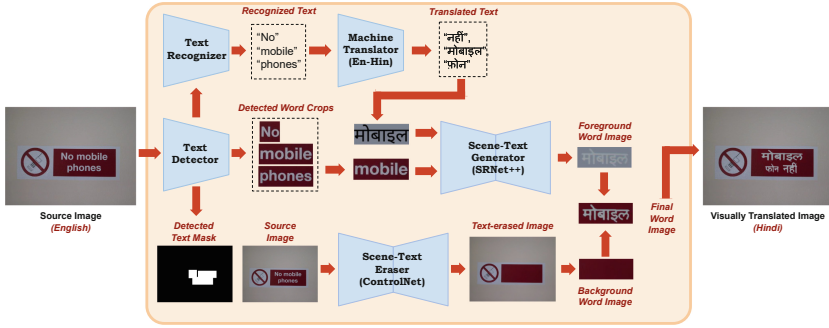


Fig. 3. Our proposed baseline extends the SRNet scene text synthesis approach by decoupling background and foreground generation. More details provided in Section 2.

Popularly, scene-text synthesis methods aim to generate this image using a single end-to-end trained model [14, 18, 27, 31]. However, in our empirical results, we consistently observed two frequent limitations with such methods: (i) incomplete erasure of source text in the generated image and (ii) unintended alterations to the background (non-text) regions. These errors often resulted in a "patchy" effect, among other unnatural artifacts, when the generated images were integrated into the full scene.

To address these issues, we propose a decoupled approach to scene-text synthesis, consisting of three independent steps: background generation and foreground generation, followed by a composition step. We collectively term this approach SRNet++ – an enhancement to the original SRNet architecture [31]. The proposed SRNet++ works as follows: (a) **Background Generation**: In this step, we employ a diffusion-based model using the ControlNet architecture to generate a text-erased image from an input containing scene texts using a publicly available implementation [29]. The model is conditioned on a binary text-masked image. It takes the full-sized source image and a mask image indicating detected text regions as input, producing a full-size image with text regions erased. These text-erased regions are then cropped to obtain clean background images. (b) **Foreground Generation**: In this, we modify the SRNet [31] architecture to generate only the foreground text information on a plain background. The model takes a source word crop and target text (rendered as black text on a gray background) as input, generating colored foreground scene text on a gray as output. During training, the model is optimized to generate both the target text and a skeletal image of the text. This model is trained from scratch using synthetic data. (c) **Composition Step**: This step combines the generated background and foreground images for each word. We apply Otsu’s method [25] to the foreground image to obtain a thresholded binary mask, which is used to extract the foreground text region. The extracted text is then composited onto the background image. This approach results in a clear, smooth image that maintains visual consistency with the source, avoiding the jagged-edge artifacts that can occur with simple overlay methods. The background generation model

utilizes pre-trained weights, while the foreground generation model is trained from scratch on our synthetic dataset. In Section 6, we compare the decoupled approach of our proposed SRNet++ with direct scene-text generation methods of MOSTEL [27] and SRNet [31].

Once we obtain all the target word images through this process, we compose them onto the full-sized input image at their respective positions. This final composition step yields the complete visually translated image.

3.1 Design Enhancements

To enhance the design, a series of refined and newly introduced steps have been implemented. The process begins with the detection and recognition of text, after which numbers, websites, and email addresses are filtered out using regular expressions. Note that these elements do not need to be translated. Words are then grouped into paragraphs and lines based on the geometry and coordinates of the bounding box in conjunction with a heuristic function. These paragraphs are translated and segmented into lines, ensuring alignment with the proportion of lines present in each original paragraph. Through cubic spline interpolation, new coordinates for each word within a line are determined, which are then linked back to the original crops of the words. Depending on the new width of the translated words, adjustments are made to the crops—either cutting or replicating them—to maintain the original style of the text.

The process is finalized by accurately positioning the new words on the image using the developed method. Although heuristically designed, this step shows a significant boost in translation quality, as shown in the experiments.

3.2 Baseline Variants

We present several baseline variants for visual translation, each incorporating different combinations of techniques for scene text detection, recognition, machine translation, and image synthesis. These variants are designed to evaluate the impact of each individual component and improvements in the pipeline. **B-1:** Utilizes ground truth scene text detection and recognition, pre-trained M2M100 [7] for machine translation, and SRNet [31] for scene text synthesis. **B-2:** Identical to B-1, but uses MOSTEL [27] instead of SRNet for scene text synthesis. **B-3:** Modifies B-1 by employing SRNet++ (our proposed enhancement of SRNet) for scene text synthesis. **B-4:** Modifies B-3 by replacing oracle bounding boxes with state-of-the-art DBNet [19] for detection and ParSeq [3] for recognition. **B-5:** Modifies B-3 by substituting M2M100 with IndicTrans2 [8], a state-of-the-art translation module for Indic languages. **B-6:** Identical to B-5 but uses DBNet and ParSeq instead of using Oracle bounding boxes. **B-7:** Addresses the limitations of word-level translation by incorporating the design enhancements proposed in Section 3.1. This variant is built upon the best-performing baseline from B-1 to B-6. B-7, in particular, represents a significant departure from the word-by-word translation approach, accounting for language-specific word ordering and context.



Fig. 4. VT-SYN dataset examples, which contains paired Eng \rightarrow Hin and Hin \rightarrow Eng images with diverse fonts, text colors, sizes, orientations, and background images of natural scenes, textures, and plain colors.

4 Dataset

The problem of visual translation has not been comprehensively studied in the literature. Therefore, no benchmark dataset currently exists for its comprehensive investigation. To fill this gap, we present the following datasets:

4.1 VT-SYN: Synthetic Training Data

For training the scene text synthesis components of our pipeline, we need paired images of text in different languages with identical visual properties (style, font, orientation, and size.). It is extremely difficult to get visually identical scene images with text in different languages in the real world, and it is even more difficult to generate accurate skeleton images required for training SRNet, MOS-TEL, as well as our proposed SRNet++ method. Thus, we rely on generating highly diverse synthetic images. We introduce **VT-SYN**, a synthetically generated corpus of 600K visually diverse paired bilingual word images in pairs of English-Hindi as well as Hindi-English.

We utilized an Indic-language scene-text image generator [23] and modified it to generate samples of scene-text in paired languages with controllable parameters for font, style, color, and spatial transformations to ensure visual diversity. Each sample contains a source image, a target word image, a background image, a foreground image, and a target image. We also generate source word images, source and target masks, and skeleton images based on the requirements of various scene-text synthesis architectures. We collect 291 publicly available fonts that support both Roman and Devanagari scripts and use a vocabulary of 3K commonly used words in both languages.

A few samples from VT-SYN are shown in Fig. 4. Note that the paired image words do not have to be translations as the STS module has to particularly learn to render the target word using the same style as the source image.

4.2 VT-REAL: Real Test Dataset

For the purpose of evaluation, we propose VT-REAL, which contains images from ICDAR 2013 [13] and Bharat Scene Text Dataset [1] to evaluate English-to-Hindi and Hindi-to-English translations, respectively. We filter images of moderate complexity³ from these two sources. In all, our dataset contains 269 images and 1021 words. These images were given to three human annotators to translate the text from Hindi to English and vice versa. A few example translation annotations of this dataset are shown in Fig. 5. Even though the above-mentioned datasets have no ground truths for Visual Translation (i.e., scene text in the target language), they are still useful for automatic evaluation proposed in the next section.

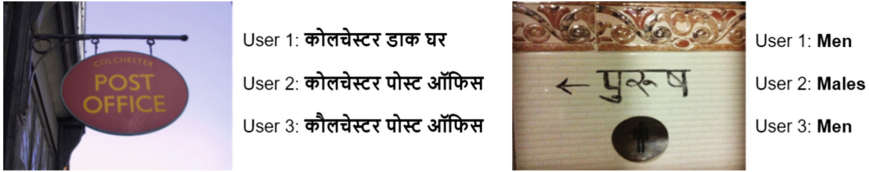


Fig. 5. A few examples from *VT-Real* dataset, showing image and Eng-Hin and Hin-Eng ground truth translations, manually annotated by three independent annotators (referred to as users here).

5 Performance Metrics

Evaluating visual translation methods is complex, even more so than evaluating machine translation. While the evaluation of machine translation has been a longstanding research area in the NLP community, recent research has saturated the use of metrics such as BLEU, METOER, and ROUGE; visual translation poses several additional challenges. Unlike machine translation, which is typically evaluated for a sentence or paragraph of text, in visual translation, one has to evaluate the correctness of translation for a single word or a small set of words or phrases. Further, It requires not only assessing the linguistic accuracy of the translation but also ensuring the preservation of background and font properties.

In this work, we propose automatic and user evaluation as follows:

5.1 Automatic Evaluation

We proposed the following three automatic evaluation metrics:

(i) **Translation Quality (TQ)**: To measure translation quality, we first detect and recognize scene text in the target language. We then group them and send them to an off-the-self machine translator, i.e., IndicTrans2. We then evaluate

³ As a first work on Hindi-to-English and English-to-Hindi translation, we have opted not to include highly complex curved and occluded text.

BLEU-1 and BLEU-2 to measure translation quality using reference translation annotations for each image and report mean scores for all images. It should be noted that BLEU-2 is not computed for those images where there is only one word in the target translation. Note that all BLEU scores are computed along with smoothing techniques as suggested in literature [4]

(ii) Perception Quality (PQ): Visually translating also requires the model to generate perceptually high-quality images without any patches or artifacts. To evaluate perception quality, we propose to use CONTRastive Image Quality Evaluator, CONTRIQUE [21], a recent approach for image quality assessment without any reference.

(iii) VT-score: For high-quality visual translation, it is important to have a high BLEU score, and perception quality along with font style preservation is required. Due to the absence of a robust automatic model that can verify cross-lingual font style similarity, we only consider Translation Quality and perception quality to compute combined *vt-score* as follows:

$$vt\text{-score} = \frac{2 \cdot TQ \cdot PQ}{TQ + PQ}. \quad (1)$$

Please note that for images that contain only one word, we employ BLEU-1 instead of BLEU-2 to assess translation quality. For the remaining images, BLEU-2 is utilized in the above mentioned scoring measure.

5.2 User Evaluation

Despite the availability of automatic evaluation metrics, as discussed above, user evaluation is crucial for assessing the accuracy, usability, and effectiveness of visual translations. User feedback is essential for evaluating the clarity, cultural appropriateness, and accessibility of translations. To this end, we conducted an extensive user evaluation with four human users aged 20 to 25 who hold graduate degrees and are proficient in both Hindi and English. They reviewed each visual translation baseline using Beamer slides: slides for metrics (ii) and (iii) featured single output images, while those for metrics (i) and (iv) displayed both input and output images together. The user evaluation metrics are described here:

(i) Translation Quality (TQ) (score range: 1-4): This criterion focuses on the accuracy of the translation. Users were asked to rate whether the translated text accurately conveys the meaning of the original text. A higher score indicates a more accurate translation. The different ratings by users convey the following: 4: Linguistically and culturally totally correct translation. 3: Some words are correct; translation can be improved. 2: Very few words are correct, and significant improvement is required. 1: Totally incorrect translation.

(ii) Readability (R) (score range: 1-4): This criterion evaluates how easily the translated text can be read within the scene image. Factors such as font size, contrast, and placement of the text may influence readability. A higher score

Table 1. Automatic Evaluation to evaluate baselines for visual translation. We report translation quality (TQ) using BLEU-1 (BL-1) and BLEU-2 (BL-2) metrics and perception quality (PQ). D.E.: Design Enhancements. More details in Section 5.1.

Method	STR	MT	STS	D.E.	TQ (BL-1)	TQ (BL-2)	PQ	VT-score
English-to-Hindi Translation								
B-7	DBNet+ParSeq	Indic SRNet++	✓		25.28	20.54	53.79	27.51
B-6	DBNet+ParSeq	Indic SRNet++	✗		22.57	15.69	53.93	25.59
B-5	Oracle	Indic SRNet++	✗		22.36	16.90	53.38	23.95
B-4	DBNet+ParSeq	M2MSRNet++	✗		19.09	14.51	54.02	21.52
B-3	Oracle	M2MSRNet++	✗		19.82	15.33	53.52	22.22
B-2	Oracle	M2MMostel	✗		14.13	10.44	46.98	16.58
B-1	Oracle	M2MSRNet	✗		15.00	12.25	46.71	16.56
Hindi-to-English Translation								
B-7	Oracle	Indic SRNet++	✓		38.30	29.30	55.49	40.08
B-6	DBNet+ParSeq	Indic SRNet++	✗		29.10	18.51	55.77	28.52
B-5	Oracle	Indic SRNet++	✗		31.31	19.70	55.62	32.27
B-4	DBNet+ParSeq	M2MSRNet++	✗		03.22	02.19	55.60	03.81
B-3	Oracle	M2MSRNet++	✗		04.20	02.89	55.58	04.97
B-2	Oracle	M2MMostel	✗		02.03	01.40	53.41	02.46
B-1	Oracle	M2MSRNet	✗		04.20	02.86	53.82	04.92

indicates better readability. The different readability ratings by users convey the following: 4: Clearly readable. 3: Can read with some effort. 2: Can read with significant effort; some words are not readable. 1: No text present in the target language.

(iii) Perceptual Quality (PQ) (score range: 1-4): This criterion assesses how well the translated text blends into the scene image, making it difficult to distinguish from a real image. A higher score indicates better integration of the translated text with the scene. Users were asked to rate approaches based on the following: 4: Very clear, looks like real image. 3: Clear image, but some patches are present if carefully seen. 2: There are a lot of patchy effects; looks like a fake image. 1: Too much patchy effect; for sure, it is a fake image.

(iv) Source Style Preservation (SSP) (score range: 1-4): This criterion examines whether the translated text preserves the style, font, color, and other visual attributes of the original text in the scene image. A higher score indicates that the translated text maintains consistency with the source text in terms of visual presentation. 4: Font style, size, color, and background are coherent to the source. 3: Only 2 or 3 of the following: font style, size, color, and background are coherent to the source. 2: Only 1 or 2 of the following: font style, size, color, and background are coherent to the source. 1: No source-style preservation.

Table 2. User Study to evaluate baselines for visual translation. We report mean Translation Quality (TQ), Readability (R), Perception Quality (PQ), and Source Style Preservation (SSP). Four fluent Hindi-English speakers rated the output on a four-point Likert scale, with 4 being the highest quality. D.E.: Design Enhancements. For more details please refer to Section 5.2.

Method STR		MT	STS	D.E.TQ	R	PQ	SSP
English-to-Hindi Visual Translation							
B-7	DBNet+ParSeq	Indic	SRNet++ ✓	2.25	2.60	2.27	1.85
B-6	DBNet+ParSeq	Indic	SRNet++ ✗	2.05	2.86	2.86	1.97
B-5	Oracle	Indic	SRNet++ ✗	2.13	3.00	2.92	1.96
B-4	DBNet+ParSeq	M2M	SRNet++ ✗	1.93	3.12	2.94	1.91
B-3	Oracle	M2M	SRNet++ ✗	1.94	3.27	2.72	1.92
B-2	Oracle	M2M	Mostel ✗	1.88	2.65	2.42	1.85
B-1	Oracle	M2M	SRNet ✗	1.94	2.51	2.50	1.88
Hindi-to-English Visual Translation							
B-7	Oracle	Indic	SRNet++ ✓	2.42	2.45	2.19	1.79
B-6	DBNet+ParSeq	Indic	SRNet++ ✗	1.92	2.11	2.05	1.67
B-5	Oracle	Indic	SRNet++ ✗	2.23	2.30	2.23	1.75
B-4	DBNet+ParSeq	M2M	SRNet++ ✗	1.36	2.07	1.95	1.42
B-3	Oracle	M2M	SRNet++ ✗	1.64	2.19	2.15	1.56
B-2	Oracle	M2M	Mostel ✗	1.38	2.03	1.94	1.63
B-1	Oracle	M2M	SRNet ✗	1.53	2.09	1.96	1.58

6 Experiments

In this section, we comprehensively evaluate scene-text to scene-text translation baseline approaches discussed in Section 3.2 using both automatic and user evaluation metrics proposed in Section 5. We use VT-REAL introduced in Section 4 for all our evaluation.

The automatic evaluation results are reported in Table 1. We observe that SRNet++ clearly emerges as the best scene text synthesis approach as compared to other existing architectures. The proposed design enhancements also significantly boost translation quality while maintaining nearly identical perceptual quality. We also observe that usage of IndicTrans2 as a translator consistently leads to an increase in translation quality. The state-of-the-art scene text recognition approaches are as good as ground truth annotations (Oracle) in the case of detecting and recognizing English text.

We further perform a rigorous user study using metrics presented in Section 5.2. As discussed in this section, we have collected user feedback from four qualified users and report mean scores of TQ, R, PQ, and SSP in Table 2. These scores nearly align with observations made via automatic evaluation. We

also observe that there is significant room for improvement on all these metrics, indicating the challenge associated with the task.

6.1 Qualitative Results

We show a selection of visual results for the proposed baseline variants in Fig. 6. The illustrated results indicate the merits/demerits of various choices. The use of IndicTrans2 as the translator instead of M2M improves translation to a large extent and also enables the transliteration of words when necessary. By using a ControlNet-based model for erasing scene-text regions in the image, in SRNet++ instead of precariously erasing text from word crops as done by MOSTEL or SRNet, we ensure complete erasing of the source text. The rendering of the target text is also clearer and has a less patchy effect. **Design enhancements**, particularly translating at paragraph level instead of word level, improve the translation correctness by taking care of language-specific ordering of words.

6.2 Comparison with Commercial Systems

Google Translate for images⁴ is a commercial system that also handles scene text-to-scene text translation. However, it is closed-source and only available through a web interface, with no free API support. As a result, we do not include it in our quantitative comparison. Nevertheless, Fig. 7 presents some qualitative comparisons, illustrating that even Google Translate is not without flaws.

6.3 Limitations

The limitations of the proposed baselines are as follows: (i) It has limited success in visually translating curved or occluded Hindi texts, partly because, unlike English, scene text detection and recognition for Indian languages are still in their infancy. (ii) There is a trade-off between image and translation quality. Design enhancements allow for sentence-level translation, but approximations in word positioning and size can cause slight blurring, as illustrated in Fig. 6. While these issues affect perceptual quality, we prioritize translation accuracy over image sharpness as long as the text remains readable. (iii) Ensuring the natural alignment of generated text in the scene is challenging. Therefore, the baselines have limited success in translating longer sentences or phrases. (iv) Our baselines do not utilize visual cues from the scene, which impairs their ability to choose between transliteration and translation, particularly for brand names. Additionally, the absence of an automatic metric for evaluating source style preservation or the visual consistency between source and generated scene text, such as font, orientation, and style, limits our current evaluation framework. Addressing these limitations is an important direction for future research.

⁴ <https://translate.google.com/?op=images>

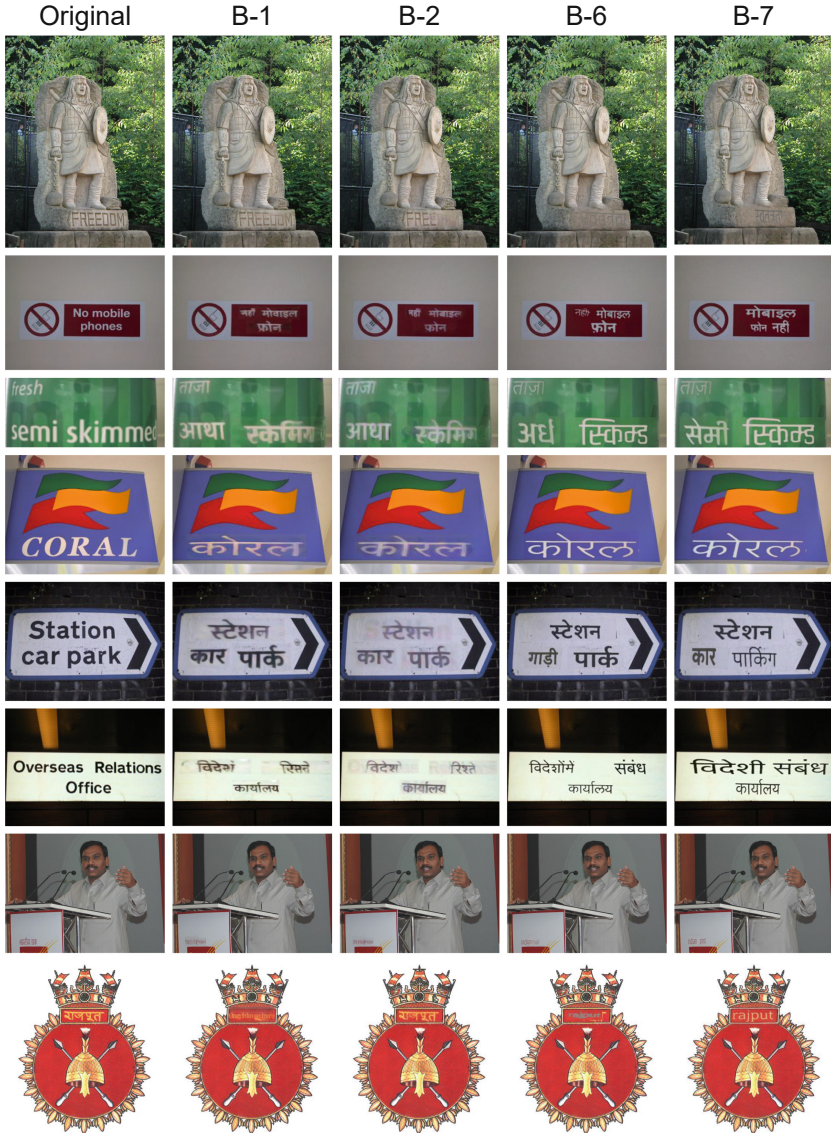


Fig. 6. A selection of visual translation results for proposed baseline variants for Eng \rightarrow Hin in row 1–6 and Hin \rightarrow Eng in row 7–8. Here, we show (left to right) the original image and results of best-performing baselines B-1, B-2, B-6, and B-7 (please refer to Section 3.2 for details about these baseline variants). We observe that B-7, which uses SRnet++ for scene text synthesis and proposed design enhancements, is clearly superior in visual translation. Native Hindi speakers can find that IndicTrans2 (used in B-6 and B-7) produces superior translations, and the design enhancements in B-7 result in grammatically correct translations.



Fig. 7. Comparison of our proposed baseline (B-7) with Google Translate for images (a commercial application). For more details please refer to Section 6.2.

7 Conclusion

We have presented a comprehensive study for the task of visual translation by proposing a series of baselines that utilize state-of-the-art approaches and their enhancements across various modules. Our baselines demonstrate promising results for translating scene text images between English and Hindi. However, it is evident that visual translation remains challenging, and addressing all of its complexities extends beyond the scope of this single paper. We hope that introducing this task, along with the dataset, baseline, and performance metrics, will inspire the research community to develop advanced models for visual translation.

Acknowledgement. This work was partly supported by MeitY, Government of India under NLTM-Bhashini.

References

1. Bharat Scene Text Dataset. <https://github.com/Bhashini-IITJ/BharatSceneTextDataset> (2024)
2. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. In: ICLR (2015)
3. Bautista, D., Atienza, R.: Scene text recognition with permuted autoregressive sequence models. In: ECCV (2022)
4. Chen, B., Cherry, C.: A systematic comparison of smoothing techniques for sentence-level bleu. In: WMT@ACL (2014)

5. Desai, P., Sangodkar, A., P. Damani, O.: A domain-restricted, rule based, english-hindi machine translation system based on dependency parsing. In: *ICON* (2014)
6. Edunov, S., Ott, M., Auli, M., Grangier, D.: Understanding back-translation at scale. In: *EMNLP* (2018)
7. Fan, A., Bhosale, S., Schwenk, H., Ma, Z., El-Kishky, A., Goyal, S., Baines, M., Celebi, O., Wenzek, G., Chaudhary, V., Goyal, N., Birch, T., Liptchinsky, V., Edunov, S., Auli, M., Joulin, A.: Beyond English-centric Multilingual Machine Translation. *J. Mach. Learn. Res.* **22**, 107:1–107:48 (2021)
8. Gala, J.P., Chitale, P.A., AK, R., Gumma, V., Doddapaneni, S., M., A.K., Nawale, J.A., Sujatha, A., Pudupully, R., Raghavan, V., Kumar, P., Khapra, M.M., Dabre, R., Kunchukuttan, A.: IndicTrans2: Towards high-quality and accessible machine translation models for all 22 scheduled indian languages. *Trans. Mach. Learn. Res.* **2023** (2023)
9. Gehring, J., Auli, M., Grangier, D., Yarats, D., Dauphin, Y.N.: Convolutional sequence to sequence learning. In: *ICML* (2017)
10. Hurskainen, A., Tiedemann, J.: Rule-based machine translation from english to finnish. In: *WMT* (2017)
11. Jia, Y., Weiss, R.J., Biadsky, F., Macherey, W., Johnson, M., Chen, Z., Wu, Y.: Direct speech-to-speech translation with a sequence-to-sequence model. In: *Inter-speech* (2019)
12. Kano, T., Sakti, S., Nakamura, S.: Transformer-based direct speech-to-speech translation with transcoder. In: *IEEE Spoken Language Technology Workshop (SLT)* (2021)
13. Karatzas, D., Shafait, F., Uchida, S., Iwamura, M., i Bigorda, L.G., Mestre, S.R., Mas, J., Mota, D.F., Almazán, J., de las Heras, L.: ICDAR 2013 robust reading competition. In: *ICDAR* (2013)
14. Krishnan, P., Kovvuri, R., Pang, G., Vassilev, B., Hassner, T.: Textstylebrush: Transfer of text aesthetics from a single example. *IEEE Trans. Pattern Anal. Mach. Intell.* **45**(7), 9122–9134 (2023)
15. Lan, Z., Niu, L., Meng, F., Zhou, J., Zhang, M., Su, J.: Translatotron-v (ison): An end-to-end model for in-image machine translation. In: *ACL (Findings)* (2024)
16. Lan, Z., Yu, J., Li, X., Zhang, W., Luan, J., Wang, B., Huang, D., Su, J.: Exploring better text image translation with multimodal codebook. In: *ACL* (2023)
17. Lee, A., Chen, P.J., Wang, C., Gu, J., Popuri, S., Ma, X., Polyak, A., Adi, Y., He, Q., Tang, Y., et al.: Direct speech-to-speech translation with discrete units. In: *ACL* (2022)
18. Lee, J., Kim, Y., Kim, S., Yim, M., Shin, S., Lee, G., Park, S.: Rewritenet: Reliable scene text editing with implicit decomposition of text contents and styles. In: *CVPRW* (2022)
19. Liao, M., Wan, Z., Yao, C., Chen, K., Bai, X.: Real-time scene text detection with differentiable binarization. In: *AAAI* (2020)
20. Ma, C., Zhang, Y., Tu, M., Han, X., Wu, L., Zhao, Y., Zhou, Y.: Improving end-to-end text image translation from the auxiliary text translation task. In: *ICPR* (2022)
21. Madhusudana, P.C., Birkbeck, N., Wang, Y., Adsumilli, B., Bovik, A.C.: Image quality assessment using contrastive learning. *IEEE Trans. Image Process.* **31**, 4149–4161 (2022)
22. Mansimov, E., Stern, M., Chen, M.X., Firat, O., Uszkoreit, J., Jain, P.: Towards end-to-end in-image neural machine translation. In: *EMNLP Workshop* (2020)
23. Mathew, M., Jain, M., Jawahar, C.: Benchmarking scene text recognition in devanagari, telugu and malayalam. In: *ICDAR* (2017)

24. Nakamura, S., Markov, K., Nakaiwa, H., Kikui, G., Kawai, H., Jitsuhiro, T., Zhang, J., Yamamoto, H., Sumita, E., Yamamoto, S.: The ATR multilingual speech-to-speech translation system. *IEEE Trans. Speech Audio Process.* **14**(2), 365–376 (2006)
25. Otsu, N.: A threshold selection method from gray-level histograms. *IEEE Trans. Syst. Man Cybern.* **9**(1), 62–66 (1979)
26. Pirinen, T.A.: Apertium-fin-eng-rule-based shallow machine translation for wmt 2019 shared task. In: *WMT (2019)*
27. Qu, Y., Tan, Q., Xie, H., Xu, J., Wang, Y., Zhang, Y.: Exploring stroke-level modifications for scene text editing. In: *AAAI (2023)*
28. Roy, P., Bhattacharya, S., Ghosh, S., Pal, U.: STEFANN: scene text editor using font adaptive neural network. In: *CVPR (2020)*
29. Susladkar, O.: Diff-Scene Text Eraser. https://github.com/Onkarsus13/Diff_SceneTextEraser (2023)
30. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: *NeurIPS (2017)*
31. Wu, L., Zhang, C., Liu, J., Han, J., Liu, J., Ding, E., Bai, X.: Editing text in the wild. In: *ACM-MM (2019)*
32. Yang, Q., Huang, J., Lin, W.: Swaptext: Image based texts transfer in scenes. In: *CVPR (2020)*
33. Zhang, L., Rao, A., Agrawala, M.: Adding conditional control to text-to-image diffusion models. In: *ICCV (2023)*
34. Zhao, G., Sun, X., Xu, J., Zhang, Z., Luo, L.: Muse: Parallel multi-scale attention for sequence to sequence learning. arXiv preprint [arXiv:1911.09483](https://arxiv.org/abs/1911.09483) (2019)



iGrasp: An Interactive 2D-3D Framework for 6-DoF Grasp Detection

Jian-Jian Jiang¹, Xiao-Ming Wu¹, Zibo Chen¹, Yi-Lin Wei¹,
and Wei-Shi Zheng^{1,2,3}(✉)

¹ School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, China

{jiangjj35,wuxm65,chenzb8,weiyin5}@mail2.sysu.edu.cn, wszheng@ieee.org

² Key Laboratory of Machine Intelligence and Advanced Computing, Ministry of Education, Guangzhou, China

³ Guangdong Province Key Laboratory of Information Security Technology, Sun Yat-sen University, Guangzhou, China

Abstract. For 6-DoF grasp detection, we aim at introducing a new interactive 2D-3D framework which filters out irrelevant information and makes both modalities collaborate effectively to generate robust grasps and accelerate inference speed greatly. This cannot be accomplished by existing works on learning to grasp that merely utilize 3D point clouds or leverage both 2D textures and 3D point clouds. Our framework is called iGrasp, a novel three-step design between 2D textures and 3D point clouds, where the interaction modelling enhances both modalities. Concretely, we propose the 2D-to-3D interaction to leverage objectness masks generated from 2D textures to filter out target-irrelevant information in 3D point clouds. Then, we introduce the 3D-to-2D interaction to leverage structural priors from 3D point cloud features with cross-attention and cylinder grouping to refine 2D texture features. Finally, we combine the refined 2D texture features and 3D point cloud features for generating high-quality 6-DoF grasp poses. Our experiments on the large-scale real-world dataset, namely GraspNet-1Billion, demonstrate that iGrasp surpasses state-of-the-art methods by 4.66/3.53 mAP on RealSense/Kinect and reduces the inference time by 28%. Real-world experiments further verify the effectiveness of iGrasp.

Keywords: 6-DoF Grasp Detection · Deep Learning in Grasping.

1 Introduction

As a fundamental problem in robotics community, 6-DoF grasp detection aiming at predicting grasp points and rotations in cluttered scenes has a wide range of applications in picking [1], stowing [2] and home servicing [3], *etc.*

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-78113-1_22.

In 6-DoF grasp detection, most regression-based methods [4–10] merely leverage 3D point clouds generated by depth cameras for predicting grasp poses and grasp scores. Although 3D point clouds comprise rich geometric information, the presence of heavy noise along with the numerous target-irrelevant background points (about 78% on average in GraspNet-1Billion [4]) causes interference in the perception of target objects. The noise degrades the performance and numerous irrelevant non-target points hinder the inference speed. As depicted in Fig. 2 (a) and Fig. 2 (b), the experiments on a strong baseline [6] show that numerous irrelevant information in redundant background points interfere with model training, harm performance and slow down inference speed.

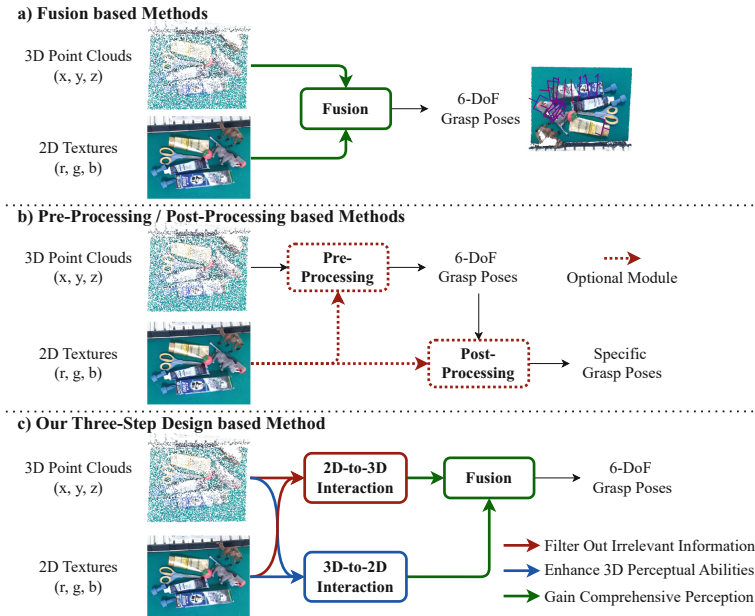


Fig. 1. Comparison of iGrasp with other 2D-3D 6-DoF grasping frameworks. Fusion based methods (a) propose multi-modal fusion modules (indicated by the green arrow) and use the fused features to predict grasp poses. Another pipeline (b) pre-processes the 3D point clouds or post-processes the predicted 6-DoF grasps with 2D textures (indicated by the red arrow). Our iGrasp model (c) adopts a three-step design (directed by the three colored arrows), filtering out numerous irrelevant information in redundant background points and generating high-quality grasp poses with reduced inference time. (Color figure online)

To address the problem caused by 3D point clouds, we delicately introduce 2D textures and propose a new interactive 2D-3D framework. This framework filters out numerous irrelevant information in redundant background points and makes both modalities collaborate effectively. As a result, it can generate robust

grasps and improve inference speed. This cannot be accomplished by previous works on learning to grasp by using 2D textures and 3D point clouds, where they only generate grasps with multi-modal and do not consider the target-irrelevant information problem. As shown in Fig. 1 (a), some methods [11, 12] combine 2D textures and 3D point clouds in a simple fusion way to enhance the perception of the grasp detector in target objects. While other methods depicted in Fig. 1 (b) [13–15] utilize 2D textures to pre-process input 3D points or to post-process predicted grasps. To sum up, existing 2D-3D based methods improve the quality of grasps with multi-modal in a simple unidirectional interaction way and do not consider the numerous irrelevant information in 3D point clouds.

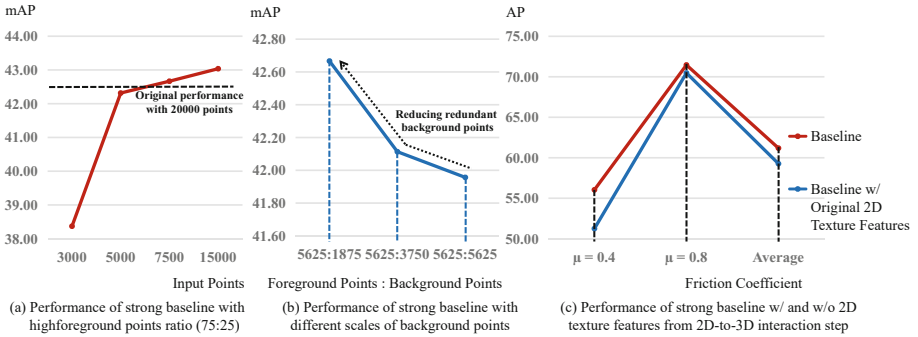


Fig. 2. Some experiments on a strong grasp detection baseline [6]. To verify the disadvantages of irrelevant information in redundant background points, we conduct experiments (a) and (b). Experiment (a) assesses the performance of the baseline with a high foreground to background ratio (75:25) across different scales of input points. The dashed line represents the original performance with 20000 input points at a ratio of 22:78. Increasing the ratio to 75:25 significantly improves performance with an increased number of input points, surpassing the original performance with fewer than half the number of input points. Experiment (b) analyzes the performance of the baseline with varying scales of background points, showing a continuous decline in performance as background points increase while foreground points remain fixed (5625). To verify the ineffectiveness of directly using original 2D texture features, we conduct experiment (c), which shows that integrating the original 2D texture features into the baseline is unsuitable and harms performance.

Our framework named iGrasp, contains three steps: the 2D-to-3D interaction step, the 3D-to-2D interaction step and the fusion step, as depicted in Fig. 1 (c). The 2D-to-3D interaction step leverages 2D textures to filter out target-irrelevant information in 3D point clouds, where the objectness masks generated from 2D textures are utilized to guide point cloud downsampling. This guidance can effectively filter out the disturbance caused by numerous irrelevant information in redundant background points and help the feature extractor focus more on target objects, thereby significantly decreasing the number of input points and inference time while maintaining competitive performance.

To further leverage the introduced 2D textures, we aim to use 2D texture features to assist the generation of grasp poses. However, as depicted in Fig. 2 (c), the experiment on a strong grasp detection baseline shows that merely combining the original 2D texture features is insufficient for 6-DoF grasp detection and even harms performance. We believe this is caused by the sensitivity and lack of 3D perception in the original 2D texture features. Therefore, in the second step, we design a novel 3D-to-2D interaction to utilize 3D point cloud features to refine 2D texture features. We employ cross-attention and cylinder grouping to refine the 2D texture features with structural priors from 3D point cloud features. This refinement enhances perceptual capabilities of 2D texture features in 3D space while reducing their sensitivity, mitigating the problems related to 2D texture sensitivity and their lack of 3D perceptual abilities.

Finally, the refined 2D texture features are combined with 3D point cloud features in the fusion step to generate 6-DoF grasp poses. This process alleviates the negative effects of point cloud noise. In this step, we employ a concatenation operation to integrate the refined 2D texture features with 3D point cloud features to predict grasp poses. Benefiting from these complementary modalities, the grasp detector can better understand targets from different perspectives, thereby generating robust grasp poses and effectively mitigating challenging scenarios caused by severe occlusions and point cloud noise.

In summary, benefiting from the three-step design, iGrasp can mitigate the limitations of each modality. It enables the generation of effective grasps in challenging scenarios and greatly reduces inference time by eliminating irrelevant 3D points through interaction modeling. Our extensive experiments on the large-scale real-world dataset, namely GraspNet-1Billion [4], demonstrate that iGrasp can greatly reduce the number of input 3D points (by about 80%). It outperforms the state-of-the-art method by 4.66/3.53 mAP on RealSense/Kinect, achieving a 28% reduction in inference time. Moreover, based on the three-step design, iGrasp is capable of addressing challenging scenarios where successful grasps cannot be generated and achieving a failure rate of 1/5 that of the SOTA method. Real-world experiments further verify the effectiveness of iGrasp.

2 Related Work

2.1 6-DoF Grasp Detection based on Point Clouds

Point clouds based 6-DoF grasp detection methods can be divided into three main categories. With the development of 6D object pose estimation, some model based methods [16, 17] typically predict 6D poses of objects and project the predefined grasps to the scene. However, they rely on 3D object models and cannot generalize to novel scenes involving unseen objects. Another pipeline is sampling-evaluation [18–20] that comprises two steps. They densely uniformly sample grasp candidates with heuristic sampling strategies at first and then evaluate them with deep neural networks. However, sampling-evaluation methods are usually time-consuming because they need to sample numerous grasps to cover the optimal grasp. Recently, lots of methods [4–10] predict grasp poses in an

end-to-end manner, where 3D point clouds are directly processed by point cloud backbones. Most of these methods focus on improving the quality of grasps, but waste computing resources on irrelevant non-target points.

2.2 6-DoF Grasp Detection based on 2D-3D

For 6-DoF grasp detection, the application of 2D-3D has not been fully explored. Recently, few methods [11–15] combine 2D textures and 3D point clouds simultaneously to accomplish this task. These approaches can be divided into two categories. With the development of multi-modal applications, some fusion based methods [11, 12] integrate 2D texture features with 3D point cloud features in a simple fusion way and use the fused features to predict 6-DoF grasps (See Fig. 1 (a)). Another pipeline [13–15] employs 2D textures to pre-process the 3D input point clouds to identify regions of interest or to post-process the predicted grasps to select specific grasp poses (See Fig. 1 (b)). However, all existing methods merely utilize the unidirectional interaction between modalities and do not consider the target-irrelevant information in 3D point clouds.

In this work, we propose a new interactive 2D-3D framework for 6-DoF grasp detection to filter out numerous target-irrelevant information so as to generate accurate grasps and accelerate inference speed greatly. (See Fig. 1 (c)). iGrasp is a novel three-step design between 2D textures and 3D point clouds, where the interaction modelling can enhance both modalities and enable fast inference speed and robust performance.

3 iGrasp: An Interactive 2D-3D Framework for 6-DoF Grasp Detection

3.1 Problem Statement

Given a single-view 2D texture (in this work we use a RGB image as example) $I \in \mathbb{R}^{H \times W \times 3}$ and a single-view point cloud $P \in \mathbb{R}^{N \times 3}$ captured by a depth camera, 6-DoF grasp detection task aims to generate a set of stable 6-DoF grasp poses in a cluttered scene. The 6-DoF grasp pose can be represented as:

$$G = [\mathbf{R}, \mathbf{t}, w], \quad (1)$$

where $\mathbf{R} \in \mathbb{R}^{3 \times 3}$ denotes the grasp rotations, $\mathbf{t} \in \mathbb{R}^3$ denotes the grasp positions, and $w \in \mathbb{R}$ denotes the grasp width that is suitable for grasping target objects. In this work, we follow the setting in GraspNet-1Billion [4] and decouple rotations \mathbf{R} into grasp views \mathbf{v} and angles θ , as shown in Fig. 3. In addition, grasp points are determined by the object point “obj” and the grasp depth d .

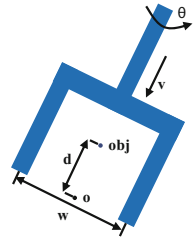


Fig. 3. The 6-DoF grasp pose representation used in iGrasp. A grasp can be represented as grasp view \mathbf{v} , grasp angle θ , grasp width w , grasp depth d from object point “obj” to the grasp origin o .

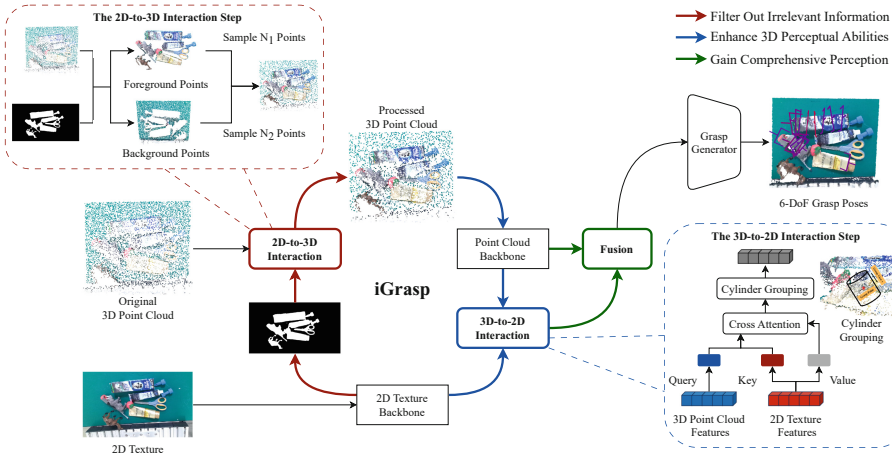


Fig. 4. The architecture of iGrasp that utilizes a three-step design between 2D textures and 3D point clouds, achieving improved performance with reduced inference time. The workflow of iGrasp is depicted by colored arrows. Before forward propagation, as directed by the red arrow, we use 2D textures to perform segmentation tasks. By employing the foreground-background mask, 2D-to-3D interaction filters out information irrelevant to targets. Before the grasp generation, as directed by the blue arrow, 3D-to-2D interaction enhances 2D texture features using 3D point cloud features. Finally, as directed by the green arrow, the fusion step combines the refined 2D texture and 3D point cloud features and uses them for 6-DoF grasp generation. (Color figure online)

3.2 Overview of iGrasp

In this work, we propose a new interactive 2D-3D framework for 6-DoF grasp detection to filter out irrelevant information in redundant background points so as to generate accurate grasps and improve inference speed. Our framework is called iGrasp, a novel three-step design between 2D textures and 3D point clouds, making both modalities enhanced and collaborate effectively. As shown in Fig. 4, before forward propagation, we utilize the 2D-to-3D interaction to filter out information irrelevant to target objects in 3D point clouds, thereby greatly reducing the number of input points and inference time while maintaining competitive performance. Then, before grasp generation, we leverage the 3D-to-2D interaction to further refine the original 2D texture features with 3D point cloud features, enhancing the robustness and 3D perceptual capabilities of them. Finally, in the fusion step, we combine the refined 2D texture features with 3D point cloud features and utilize the fused features to generate grasps.

3.3 The 2D-to-3D Interaction

Benefiting from the strong perception of 2D textures in object edges, colors, *etc.* and the development of lightweight 2D backbones [21–23], we use 2D textures to address issues of high computation and numerous irrelevant information in point clouds. In the 2D-to-3D interaction, we use objectness priors from 2D textures to interact with 3D point clouds. The key insight is that since 3D point clouds from depth maps and 2D textures correspond one-to-one, we utilize 2D textures for segmentation instead of point clouds and then use the predicted masks to filter out target-irrelevant information in input points. We opt for foreground-background segmentation over instance segmentation, which simplifies the problem and enables a lightweight CNN to finish it. With the help of the segmentation results, we can easily decrease the redundant background points, thereby removing numerous information irrelevant to targets.

Specifically, we build a lightweight CNN model to segment cluttered scenes. Given an image I , we can use our lightweight model to obtain objectness masks $M \in \{0, 1\}^{H \times W}$, where 0 denotes the background and 1 denotes the foreground. Based on the foreground and background mask, we separately select points from 3D point clouds for them with the uniform sampling algorithm. Concretely, we sample N_1 points in the foreground to basically preserve complete point clouds of target objects and we sample N_2 points in the background to maintain the relationship between objects and the environment. The ratio of N_1 to N_2 is much greater than 1 to filter out numerous irrelevant information.

It should be noted that it remains essential to preserve a limited number of background points to guarantee that the grasp generator can establish the relationship between target objects and the environment. This relationship offers implicit constraint crucial for estimating grasp poses.

With the 2D-to-3D interaction, we greatly reduce the number of input points, achieving competitive performance with a slight rise in parameters (only 3%). The loss function employed to train this step is as follows:

$$L_{seg} = L_{cls}(M, M^*), \quad (2)$$

M^* means the ground truth of M and L_{cls} denotes the Cross Entropy loss.

3.4 The 3D-to-2D Interaction

To further leverage the introduced 2D textures, we aim to use 2D texture features to assist the generation of grasps. However, as described in Sect. 1, the original 2D texture features are not well-suited for this task. We believe this is caused by the lack of 3D perception and sensitivity in original 2D texture features. To address this problem, in the 3D-to-2D interaction, we use 3D point cloud features to interact with 2D texture features before predicting 6-DoF grasps. This enables 2D texture features to have 3D perception, making them more robust and suitable for this task. The key insight is that we use 3D point cloud features to refine 2D texture features, introducing 3D structural priors into 2D texture features. For this purpose, we design two novel submodules.

Firstly, inspired by the cross-attention mechanism, we use 3D point cloud features to aggregate 2D texture features, thereby allowing 2D textures to better mitigate sensitivity and have 3D perception. Specifically, given M selected grasp points, we can obtain the corresponding 2D texture features $F_{2d} \in \mathbb{R}^{M \times C}$ from image feature volumes. Denote the 3D structural features as $F_{3d} \in \mathbb{R}^{M \times C}$, and we can generate the output 2D texture features by employing cross attention to F_{3d}, F_{2d} , where F_{3d} is the query and F_{2d} is the key and value. With the cross-attention mechanism, 2D texture features are endowed with the 3D perceptual abilities and are suitable for 6-DoF grasp detection.

Moreover, cylinder grouping is a widely used strategy in this task [4, 6, 8], which can integrate enough points into a cylinder, forming a suitable space for grasp detection. Thus, we also use this strategy in iGrasp. To align 2D texture features with 3D point cloud features grouped by cylinders, we apply this operation to 2D texture features. Given a height equal to the length of the gripper model, we sample neighboring points within the cylinder centered along the grasp view to a fixed number. Based on the sampled point set, we extract the corresponding 2D texture feature set from the cross-attention stage. Then, we use max pooling to process features in the 2D texture feature set, thereby obtaining 2D texture features grouped by cylinders. The resulting 2D texture features are more robust, 3D perceptive and aligned with 3D point cloud features.

3.5 Finalizing iGrasp

The appearance information in the refined 2D textures and the geometry information in 3D point clouds are complementary to enhance the perception of the grasp detector in targets. Therefore, in the fusion step, we use concatenation to combine the refined 2D texture and 3D point cloud features. Meanwhile, we use a multi-layer perceptron to map the concatenated features to a lower dimension, achieving the fusion of modalities for 6-DoF grasp detection.

Then, we constrain the output features by the following losses. Our loss is divided into three parts. Firstly, we utilize L_{seg} to supervise the foreground-background masks. Then, we follow previous methods [4, 6] to conduct downsampling and utilize L_o to guide the selection of grasp points. Finally, we supervise the prediction of grasp views, scores and widths with L_v , L_s and L_w respectively. To sum up, iGrasp is trained with a multi-task loss:

$$\begin{aligned} L_{pred} &= \lambda L_v + \beta(L_s + L_w), \\ L &= L_{seg} + \alpha L_o + L_{pred}. \end{aligned} \tag{3}$$

Previous works [6, 24] demonstrate that distinguishing graspable points from the input 3D point clouds is significant. Therefore, we further divide L_o into object point classification and graspable score regression.

4 Experiments

4.1 Dataset and Settings

Dataset. We choose GraspNet-1Billion [4], which is widely used in 6-DoF grasp detection [6, 8, 9, 11, 12, 14]. It is a large-scale real-world dataset, which contains 190 scenes with 256 distinct views captured by two cameras (RealSense/Kinect). Scenes are split into train and test set with 100 and 90 scenes respectively, where the test set is further divided into seen, similar and novel.

Evaluation Metric. We adopt AP_μ and AP as provided by GraspNet-1Billion [4] to evaluate the performance, where μ denotes the friction coefficient calculated by [25] and AP denotes the mean value of various AP_μ .

Table 1. GraspNet-1Billion quantitative results on RealSense.

Method	Input Points	Time (ms)	Seen AP			Similar AP			Novel AP		
			AP	AP _{0.8}	AP _{0.4}	AP	AP _{0.8}	AP _{0.4}	AP	AP _{0.8}	AP _{0.4}
GPD [18]	N/A	>1000	22.87	28.53	12.84	21.33	27.83	9.64	8.24	8.89	2.67
PointNetGPD [19]	N/A	>1000	25.96	33.01	15.37	22.68	29.15	10.76	9.23	9.89	2.74
Graspnet-baseline [4]	20000	151.58	27.56	33.43	16.95	26.11	34.18	14.23	10.55	11.25	3.98
TransGrasp [9]	20000	13.52	39.81	47.54	36.42	29.32	34.80	25.19	13.83	17.11	7.67
GNet [6]	20000	51.79	65.70	76.25	61.08	53.75	65.04	45.97	23.98	29.93	14.05
RGB-Matters [14]	N/A	554.36	27.98	33.47	17.75	27.23	36.34	15.60	12.25	12.45	5.62
Liu et al. [11]	20000	-	36.29	44.51	29.73	30.52	36.57	23.36	15.34	18.24	6.85
HGGD [12]	25600	38.11	59.36	-	-	51.20	-	-	22.17	-	-
Ours	5000	29.18	69.74	80.07	65.20	61.61	73.21	54.92	26.06	32.34	14.87

The gray area in the table represents point clouds based sampling-evaluation methods, the yellow area represents point clouds based regression methods, and the red area represents 2D-3D based regression methods.

Table 2. GraspNet-1Billion quantitative results on Kinect.

Method	Input Points	Time (ms)	Seen AP			Similar AP			Novel AP		
			AP	AP _{0.8}	AP _{0.4}	AP	AP _{0.8}	AP _{0.4}	AP	AP _{0.8}	AP _{0.4}
GPD [18]	N/A	>1000	24.38	30.16	13.46	23.18	28.64	11.32	9.58	10.14	3.16
PointNetGPD [19]	N/A	>1000	27.59	34.21	17.83	24.38	30.84	12.83	10.66	11.24	3.21
Graspnet-baseline [4]	20000	151.58	29.88	36.19	19.31	27.84	33.19	16.62	11.51	12.92	3.56
TransGrasp [9]	20000	13.52	35.97	41.69	31.86	29.71	35.67	24.19	11.41	14.42	5.84
GNet [6]	20000	51.79	61.19	71.46	56.04	47.39	56.78	40.43	19.01	23.73	10.60
RGB-Matters [14]	N/A	554.36	32.08	39.46	20.85	30.40	37.87	18.72	13.08	13.79	6.01
HGGD [12]	25600	38.11	60.26	-	-	48.59	-	-	18.43	-	-
Ours	5000	29.18	62.65	73.04	56.12	54.17	64.58	46.60	21.36	26.53	12.37

The gray area in the table represents point clouds based sampling-evaluation methods, the yellow area represents point clouds based regression methods, and the red area represents 2D-3D based regression methods.

4.2 Implementation Details

In this work, the input RGB image size is $1280 \times 720 \times 3$ and the number of input points is 5000. In the 2D-to-3D interaction step, we uniformly select $N_1 = 4500$ points and $N_2 = 500$ points from the foreground and background respectively. The point cloud backbone outputs $C = 512$ dimensional point cloud features to select suitable grasp points from the input point clouds and the best grasp view from the predefined 300 approaching directions. Notably, we utilize a 3D UNet built upon the Minkowski Engine [26] as the 3D point cloud backbone and MnasNet [23] as the 2D texture backbone.

Then, in grasp generator, we divide grasp angles into 12 classes and depths into 4 classes to regress grasp scores and widths of (angle-depth) combinations. In loss functions, we set $\alpha, \beta, \lambda = 10, 15, 100$ respectively.

iGrasp is implemented with PyTorch [27] and trained on an NVIDIA GTX 1080Ti GPU or NVIDIA RTX 3090 GPU for 10 epochs with Adam optimizer and the batch size of 2. The learning rate is 1.25×10^{-4} at the first epoch, and multiplied by 0.95 every one epoch.

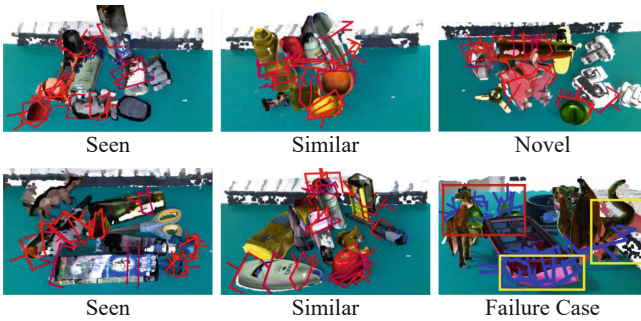


Fig. 5. Qualitative results of our predicted grasp poses. Scenes are constructed using RGB-D images captured by a Kinect camera. The red gripper indicates a viable grasp pose, while the blue one indicates an infeasible grasp pose. The figure at the bottom right presents a failure case of iGrasp, where it fails to predict a feasible grasp in this challenging scenario. In this figure, yellow boxes highlight areas with significant noise, with red and pink points representing the noise point clouds. The red box indicates an area that is more heavily shielded. Zoom in for the best view. (Color figure online)

4.3 Experiments on GraspNet-1Billion

Quantitative Results. We compare our method with previous representative 6-DoF grasp detection methods. GPD [18] and PointNetGPD [19] are sampling-evaluation based methods, which classify the grasp candidates generated by heuristic sampling strategies. GraspNet-baseline [4], TransGrasp [9] and GS-Net [6] are regression based methods, which directly process scene points with

point cloud backbones and predict grasp poses together with grasp scores. RGB-Matters [14] incorporates 2D textures and 3D point clouds, which predicts grasp orientations with 2D textures and searches grasp width-depth pairs with 3D point clouds. Liu *et al.* [11] propose a novel method for integrating 2D texture features with 3D point cloud features, utilizing the fused features for predicting grasp poses. HGGD [12] utilizes RGB-D images to generate grasp heatmaps, guiding local grasp generators to predict effective grasp poses.

We test iGrasp in three object categories. As shown in Table 1 and Table 2, we can see that: Our method utilizes fewer points (about 20%), surpassing previous methods and resulting in faster inference speed. Compared with GS-Net, the previous SOTA method, the results of our method are improved by 4.04/1.46 AP, 7.86/6.78 AP and 2.08/2.35 AP for Realsense/Kinect input on seen, similar and novel scenarios respectively. Achieving such good performance is due to our three-step design. The design filters out information irrelevant to targets in point clouds, endows 2D texture features with 3D perception and uses complementary data modalities to generate high-quality 6-DoF grasp poses.

Qualitative Results. We visualize top-20 grasps generated by iGrasp to qualitatively verify the performance of iGrasp, which can be seen in Fig. 5. It can be concluded that: (1) In most scenarios, iGrasp performs well and achieves nearly 100% success rate, no matter in seen, similar or in novel setting. (2) However, iGrasp may still fail when encountering challenging scenes, like the one depicted in the bottom right that has heavy noise and severe occlusions.

Table 3. The performance of ablation studies on each component on Kinect.

2D-to-3D Interaction	3D-to-2D Interaction	Time(ms)	Seen AP			Similar AP			Novel AP		
			AP	AP _{0.8}	AP _{0.4}	AP	AP _{0.8}	AP _{0.4}	AP	AP _{0.8}	AP _{0.4}
		19.82	48.41	59.64	37.41	42.88	53.74	31.68	16.01	20.22	7.65
✓		23.29	58.37	69.65	50.05	49.77	61.08	40.22	18.81	23.46	10.47
	✓	25.71	53.39	63.77	44.94	49.28	59.89	40.73	17.91	22.49	8.66
✓	✓	29.18	62.65	73.04	56.12	54.17	64.58	46.60	21.36	26.53	12.37

Table 4. The ablation studies of submodules in 3D-to-2D interaction on Kinect.

Cross Attention	Cylinder Grouping	Seen AP	Similar AP	Novel AP
		58.37	49.77	18.81
✓		57.64	49.35	19.51
	✓	58.66	51.05	19.82
✓	✓	62.65	54.17	21.36

Table 5. The ability of models to handle challenging scenarios.

Method	FailureScene Number	FailureScene Ratio
GS-Net [6]	17	0.074%
Ours	3	0.013%

Ablation Study. We conduct ablation studies for iGrasp. Concretely, we use 5000 input points and test the effectiveness of the 2D-to-3D interaction and 3D-

to-2D interaction. The results are shown in Table 3. We can conclude that: (1) Due to the decrease in input points, randomly sampling in the workspace cannot maintain the completeness of object points, resulting in low performance (line 1 in Table 3). (2) Utilizing the 2D-to-3D interaction to filter out target-irrelevant information improves grasping performance, which adds 9.96 AP, 6.89 AP and 2.80 AP on Kinect across the three categories respectively (line 2 in Table 3). (3) Then, to further leverage 2D textures, we use the 3D-to-2D interaction to process 2D texture features, assisting in the generation of 6-DoF grasp poses. This improves the model with 4.28 AP, 4.40 AP and 2.55 AP on Kinect across seen, similar, and novel scenarios respectively (line 4 in Table 3). Above all, by combining the 2D-to-3D and 3D-to-2D interaction, we achieve a high-performance and high-speed 6-DoF grasp detection framework.

Analysis of Submodules in 3D-to-2D Interaction. We also conduct a detailed ablation study for our 3D-to-2D interaction. Specifically, we fix the input points equal to 5000 and the ratio of foreground points to background points equal to 9:1 to test the effectiveness of our submodule in 3D-to-2D interaction. It should be noted that the 2D-to-3D interaction is used for all the experimental results recorded in the Table 4. The results are depicted in Table 4. From the table, we can conclude that: (1) The submodules complement with each other. Lacking either one cannot effectively improve the performance (line 2 in Table 4, line 3 in Table 4). (2) When we combine submodules together, we achieve a significant improvement in performance by aligning the 3D-perceptive 2D texture features and 3D point cloud features in the feature space (line 4 in Table 4).

Analysis of Mitigating Challenging Scenarios. We analyze the ability of iGrasp in challenging scenes that are often caused by severe noise and occlusions. We summarize all the scenarios in the test set of GraspNet-1Billion where successful grasp poses cannot be generated with iGrasp compared with GS-Net. Results are recorded in Table 5. This table shows that by using the three-step design, iGrasp significantly mitigates the impact of challenging scenarios. Compared with GS-Net, both the number and ratio of our failure cases are reduced to only 1/5. We believe this is due to the fact that 2D textures, as a complementary input, help iGrasp better understand the noisy environment.

Table 6. Illustration of the grasping performance with different foreground-background ratios on Kinect.

Ratio	Seen AP			Similar AP			Novel AP		
	AP	AP _{0.8}	AP _{0.4}	AP	AP _{0.8}	AP _{0.4}	AP	AP _{0.8}	AP _{0.4}
Ori. (22:78)	53.39	63.77	44.94	49.28	59.89	40.73	17.91	22.49	8.66
75:25	62.08	72.62	55.07	52.92	63.50	44.99	21.31	26.52	12.27
85:15	62.12	72.47	55.16	53.33	63.90	45.21	20.73	25.69	12.30
90:10	62.65	73.04	56.12	54.17	64.58	46.60	21.36	26.53	12.37
100:0	60.96	71.17	53.91	52.29	62.42	44.75	19.54	24.28	11.35

Table 7. Illustration of performance with varying numbers of input points on Kinect.

Input Points	Seen AP			Similar AP			Novel AP		
	AP	AP _{0.8}	AP _{0.4}	AP	AP _{0.8}	AP _{0.4}	AP	AP _{0.8}	AP _{0.4}
3000	58.46	68.50	51.54	50.79	60.92	42.69	19.92	24.77	11.50
5000	62.65	73.04	56.12	54.17	64.58	46.60	21.36	26.53	12.37
7500	62.73	72.94	56.36	53.27	64.01	44.86	20.77	25.76	12.48
10000	63.59	73.96	57.01	54.37	65.04	46.02	21.44	26.61	12.90

Analysis of the Foreground-Background Ratio. We conduct some analysis experiments on the foreground-background ratios. We maintain the default number of input points at 5000 and test the performance using different foreground-background ratios, as seen in Table 6. It can be observed that: (1) As the ratio of foreground points increases, performance continually improves, showing the importance of filtering out irrelevant information for target objects. (2) Notably, it is essential to preserve a limited number of background points. Completely removing them leads to a decrease in performance (line 5 in Table 6).

Analysis of the Number of Input Points. We analyze how the number of input points affects performance. We fix the foreground-background ratio at 9:1 and test different numbers of input points. Results are shown in Table 7. From this table, it can be concluded that: (1) While the number of input points is small, the performance is poor. (2) When the number of points reaches a certain level, the performance will remain at a relative high level. Considering both inference speed and performance, we choose 5000 input points as the default.

Analysis of Time Costs of Each Component. We report the time costs comprising all components used by iGrasp and GS-Net. Results are shown in Table 8. The reasons for our significantly lower time costs are: (1) Our 2D texture backbone is a lightweight neural network (5.46 MB). (2) The number of input points is significantly reduced (1/4 of GS-Net’s input points).

Table 8. The Time Costs of Each Component.

Components	Time (ms)	
	GS-Net [6]	Ours
2D-to-3DInteraction	-	5.89
3D-to-2DInteraction	-	3.47
Point CloudBackbone	48.15	16.69
Grasp PosesPrediction	3.64	3.13
Total	51.79	29.18

Table 9. Quantitative results of real world grasping experiments for normal scenarios.

Object IDs	Difficulty	Object Number	Attempt Number	Success Rate
3,6,8,11,13	Normal	5	7	71.43%
3,6,9,12,16	Normal	5	5	100.00%
1,5,6,8,13,14	Normal	6	7	85.71%
4,7,10,13,15,16	Normal	6	6	100.00%
3,4,5,6,8,13,15,16	Normal	8	9	88.89%
2,3,4,5,6,7,12,16	Normal	8	8	100.00%
Average		6.33	7	90.43%

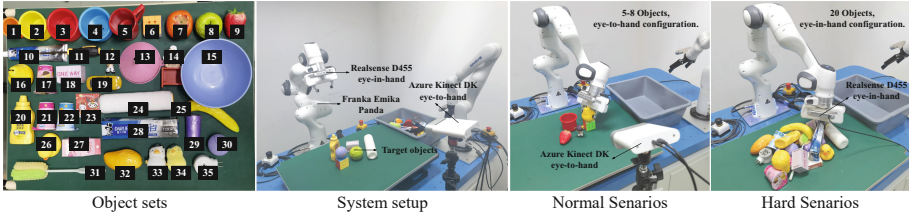


Fig. 6. The illustration of our real-world experiments settings.

4.4 Real World Grasping Experiments

Experimental Environment. To verify the real practical ability of iGrasp, we conduct real world grasping experiments for cluttered scenes. The experiments are conducted on a Franka Emika robot arm with an Azure Kinect DK and an Intel Realsense D455. To comprehensively verify the performance of our model, we assign the real world scenarios in two levels of difficulty: normal and hard. The more difficult the scenarios, the more complex the occlusions and stacking of the objects. We select 35 objects, which contain some of the objects in GraspNet-1Billion and some unseen objects from daily life as shown in Fig. 6.

Quantitative Results for Normal Scenarios. For the normal scenarios, we adopt an eye-to-hand configuration, positioning the Kinect DK camera at a fixed location to capture the scene’s 2D textures and 3D point clouds, where each scene contains 5 to 8 objects. As shown in Table 9, our model achieves a high success rate in real-world deployment, showing the effectiveness of our method.

Quantitative Results for Hard Scenarios. Directing at the hard scenarios, we use an eye-in-hand configuration, mounting the Realsense D455 camera on the robotic arm, with each scene consisting 20 color-rich objects. To verify the effectiveness of our method, we conduct the experiments five times and compare the average success rate with that of GS-Net. As shown in Table 10, iGrasp performs better, which we attribute to our novel three-step design.

Experiments on Objects with Special Materials. Benefiting from the introduced 2D textures, our iGrasp can better utilize complementary information for grasping black-body objects and transparent objects. The specific grasping process can be viewed in Fig. 7 and our video demo.

Table 10. Quantitative results of real world grasping experiments for hard scenarios.

Method	Object IDs	Difficulty	Exp.1	Exp.2	Exp.3	Exp.4	Exp.5	AVG
GS-Net [6]	16-35	Hard	83.33%	86.96%	80.00%	83.33%	83.33%	83.39%
Ours			80.00%	90.91%	83.33%	86.96%	86.96%	85.63%

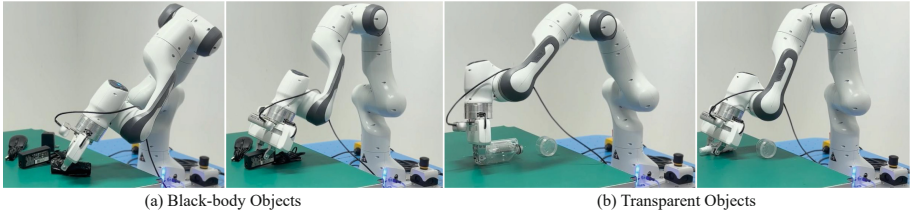


Fig. 7. The real-world experiments on objects with special materials.

5 Conclusion

In this paper, we propose a new interactive 2D-3D framework which filters out irrelevant information and makes both modalities collaborate effectively to generate robust grasps and accelerate inference speed. Our framework is called iGrasp, a novel three-step design between 2D textures and 3D point clouds where the interaction modelling enhances both modalities. Specifically, we propose the 2D-to-3D interaction to leverage objectness masks generated from 2D textures to filter out target-irrelevant information in 3D point clouds, and introduce the 3D-to-2D interaction to leverage structural priors from 3D point cloud features to refine 2D texture features. Finally, we combine the refined 2D texture features and 3D point cloud features for generating high-quality 6-DoF grasp poses. Our experiments on the large-scale real-world dataset namely GraspNet-1Billion demonstrate that iGrasp surpasses state-of-the-art methods and reduces the inference time greatly. Moreover, extensive experiments in the real world further verify the practical ability of iGrasp.

Acknowledgements. This work was supported partially by the Guangdong NSF Project (No. 2023B1515040025).

References









1. W. Yuan, A. Murali, A. Mousavian, and D. Fox, “M2T2: multi-task masked transformer for object-centric pick and place,” in *Conference on Robot Learning*, 2023
2. H. Chen, Y. Niu, K. Hong, S. Liu, Y. Wang, Y. Li, and K. R. Driggs-Campbell, “Predicting object interactions with behavior primitives: An application in stowing tasks,” in *Conference on Robot Learning*, 2023
3. H. Shi, H. Xu, S. Clarke, Y. Li, and J. Wu, “Robocook: Long-horizon elasto-plastic object manipulation with diverse tools,” in *Conference on Robot Learning*, 2023
4. H. Fang, C. Wang, M. Gou, and C. Lu, “Graspnet-1billion: A large-scale benchmark for general object grasping,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2020
5. P. Ni, W. Zhang, X. Zhu, and Q. Cao, “Pointnet++ grasping: Learning an end-to-end spatial grasp generation algorithm from sparse point clouds,” in *IEEE International Conference on Robotics and Automation*, 2020

6. C. Wang, H. Fang, M. Gou, H. Fang, J. Gao, and C. Lu, "Graspness discovery in clutters for fast and accurate grasp detection," in *IEEE International Conference on Computer Vision*, 2021
7. Y. Li, T. Kong, R. Chu, Y. Li, P. Wang, and L. Li, "Simultaneous semantic and collision learning for 6-dof grasp pose estimation," in *IEEE International Conference on Intelligent Robots and Systems*, 2021
8. D. Hoang, J. A. Stork, and T. Stoyanov, "Context-aware grasp generation in cluttered scenes," in *IEEE International Conference on Robotics and Automation*, 2022
9. Z. Liu, Z. Chen, S. Xie, and W. Zheng, "Transgrasp: A multi-scale hierarchical point transformer for 7-dof grasp detection," in *IEEE International Conference on Robotics and Automation*, 2022
10. X. Wu, J. Cai, J. Jiang, D. Zheng, Y. Wei, and W. Zheng, "An economic framework for 6-dof grasp detection," in *European Conference on Computer Vision*, 2024
11. X. Liu, Y. Zhang, H. Cao, D. Shan, and J. Zhao, "Joint segmentation and grasp pose detection with multi-modal feature fusion network," in *IEEE International Conference on Robotics and Automation*, 2023
12. S. Chen, W. Tang, P. Xie, W. Yang, and G. Wang, "Efficient heatmap-guided 6-dof grasp detection in cluttered scenes," *IEEE Robotics and Automation Letters*, 2023
13. M. Sundermeyer, A. Mousavian, R. Triebel, and D. Fox, "Contact-graspnet: Efficient 6-dof grasp generation in cluttered scenes," in *IEEE International Conference on Robotics and Automation*, 2021
14. M. Gou, H. Fang, Z. Zhu, S. Xu, C. Wang, and C. Lu, "RGB matters: Learning 7-dof grasp poses on monocular RGBD images," in *IEEE International Conference on Robotics and Automation*, 2021
15. Y. Shi, Z. Tang, X. Cai, H. Zhang, D. Hu, and X. Xu, "Symmetrygrasp: Symmetry-aware antipodal grasp detection from single-view RGB-D images," *IEEE Robotics and Automation Letters*, 2022
16. M. A. Roa and R. Suárez, "Computation of independent contact regions for grasping 3-d objects," *IEEE Transactions on Robotics*, 2009
17. X. Deng, Y. Xiang, A. Mousavian, C. Eppner, T. Bretl, and D. Fox, "Self-supervised 6d object pose estimation for robot manipulation," in *IEEE International Conference on Robotics and Automation*, 2020
18. A. ten Pas, M. Gualtieri, K. Saenko, and R. P. Jr., "Grasp pose detection in point clouds," *The International Journal of Robotics Research*, 2017
19. H. Liang, X. Ma, S. Li, M. Görner, S. Tang, B. Fang, F. Sun, and J. Zhang, "Pointnetgpd: Detecting grasp configurations from point sets," in *IEEE International Conference on Robotics and Automation*, 2019
20. A. Mousavian, C. Eppner, and D. Fox, "6-dof graspnet: Variational grasp generation for object manipulation," in *IEEE International Conference on Computer Vision*, 2019
21. M. Sandler, A. G. Howard, M. Zhu, A. Zhmoginov, and L. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018
22. X. Zhang, X. Zhou, M. Lin, and J. Sun, "Shufflenet: An extremely efficient convolutional neural network for mobile devices," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018
23. M. Tan, B. Chen, R. Pang, V. Vasudevan, M. Sandler, A. Howard, and Q. V. Le, "Mnasnet: Platform-aware neural architecture search for mobile," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019

24. B. Zhao, H. Zhang, X. Lan, H. Wang, Z. Tian, and N. Zheng, "Regnet: Region-based grasp network for end-to-end grasp detection in point clouds," in *IEEE International Conference on Robotics and Automation*, 2021
25. J. Mahler, J. Liang, S. Niyaz, M. Laskey, R. Doan, X. Liu, J. A. Ojea, and K. Goldberg, "Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics," in *Robotics: Science and Systems*, 2017
26. C. B. Choy, J. Gwak, and S. Savarese, "4d spatio-temporal convnets: Minkowski convolutional neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019
27. A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Z. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems*, 2019



Goal-Driven Transformer for Robot Behavior Learning from Play Data

Congcong Wen^{1,2,4}, Jiazhao Liang^{1,3}, Shuaihang Yuan^{1,2},
Hao Huang^{1,2}, Yu Hao^{1,2}, Hui Lin⁴, Yu-Shen Liu⁵, and Yi Fang^{1,2}

¹ Embodied AI and Robotics (AIR) Lab, New York University Abu Dhabi,
Abu Dhabi, UAE
hh1811@nyu.edu

² Center for Artificial Intelligence and Robotics, New York University Abu Dhabi,
Abu Dhabi, UAE

³ Tandon School of Engineering, New York University, New York, USA

⁴ University of Science and Technology of China,
Anhui, People's Republic of China

⁵ School of Software, Tsinghua University, Beijing, People's Republic of China

Abstract. Robot behavior learning has emerged as a crucial field, allowing robots to adapt and improve their actions based on experiential knowledge rather than being solely reliant on predefined instructions. However, the effectiveness of such learning is often hindered by the limitations of offline reinforcement learning, which relies on pre-defined reward labels, and traditional imitation learning, which depends on high-quality expert demonstrations. To address these challenges, in this paper, we propose a novel Goal-Driven Transformer (GDT) for robotic behavior learning from play data. The core module of the GDT is the inclusion of the Goal-Driven Attention Block (GDAB) that utilizes attention mechanisms to concentrate the model's focus on particular objectives, enabling the GDT to selectively focus on critical parts of the observation data to perform behavioral learning for specific goals. Moreover, we employ the Standard Attention Block (SAB) to ensure that this goal-directed learning occurs with a comprehensive understanding of the environment and the sequence of actions required. Experimental validation of the proposed GDT framework is conducted in two simulated environments: Block-pushing and Franka Kitchen. The results demonstrate that the GDT framework has achieved state-of-the-art performance in the realm of robot behavior learning from play data. Videos are available at: <https://gdt-bl.github.io/>.

Keywords: Robot Behavior Learning · Transformer · Robot Play Data · Attention Mechanism.

C. Wen, J. Liang—Equal contribution.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
A. Antonacopoulos et al. (Eds.): ICPR 2024, LNCS 15330, pp. 346–359, 2025.
https://doi.org/10.1007/978-3-031-78113-1_23

1 Introduction

Robot behavior learning, a pivotal area in robotics, involves training robots to autonomously execute a variety of tasks by learning from experiences. This field holds significant potential for advancing the autonomy of robots, enabling them to perform in unpredictable environments and adapt to new tasks without explicit programming. As robotic systems grow more complex and are expected to function in increasingly dynamic environments, the significance of efficient robot behavior learning techniques has become more evident than ever.

One approach to achieving robot behavior learning is through offline reinforcement learning [8, 13, 14], a method where a model learns to make decisions by optimizing a cumulative reward from a fixed dataset of previous interactions. However, this method heavily relies on the availability of reward labels, which are often difficult to specify accurately for complex tasks. Moreover, the dependence on pre-defined reward functions can limit the flexibility and generalizability of learned behaviors, making it challenging to adapt to tasks where the reward structure is not well understood or is difficult to encode.

Alternatively, imitation learning [11, 21] offers a pathway for robot behavior learning by mimicking expert demonstrations. This method bypasses the need for explicit reward functions by directly learning the actions demonstrated by an expert in similar situations. While imitation learning can effectively transfer expert knowledge to robots, it requires access to high-quality demonstrations, which can be costly and time-consuming to produce. Additionally, the approach may struggle with tasks that are not easily demonstrated or where expert knowledge is not readily available. In contrast, play data, which comprises unstructured and varied interactions within an environment, presents an opportunity to overcome these limitations. Play data allows for a richer exploration of possible behaviors, offering a broader learning spectrum compared to structured expert demonstrations.

Recent work has explored the utilization of play data in robot behavior learning, addressing the limitations of offline reinforcement learning and traditional imitation learning. For example, Shafiullah et al. [23] introduce a Behavior Transformer (BeT) that integrates action discretization and correction into standard transformers for predicting multi-modal continuous actions. However, the standard BeT is limited to unconditional behavior rollouts, preventing the selection of a specific behavior mode during policy deployment. To address this issue, some studies [3, 16] have utilized a combination of the observed state with the future or global state as inputs to their models for behavior learning. Instead of this simple concatenation, we introduce a goal-driven attention mechanism that significantly improves the robot’s proficiency in discerning and executing complex action sequences to reach the intended future or goal state.

In this paper, we propose the Goal-Driven Transformer (GDT), a novel framework specifically designed for robotic behavior learning from play data. The GDT architecture comprises three critical components: the *Embedding Layer*, the *Goal-Driven Attention Block (GDAB)*, and the *Standard Attention Block (SAB)*. Among these, the GDAB is our primary innovation, employing attention

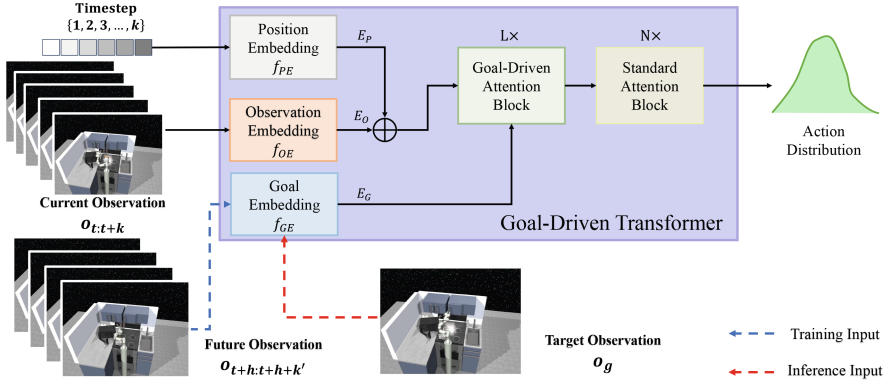


Fig. 1. The framework of the proposed Goal-Driven Transformer (GDT). The GDT model comprises three essential components: the *Embedding Layer*, the *Goal-Driven Attention Block*, and the *Standard Attention Block*. The Embedding Layer is first applied to produce Observation Embeddings E_o , Position Embeddings E_p , and Goal Embeddings E_g , respectively. Subsequently, E_o , E_p , and E_g are processed through L Goal-Driven Attention Blocks and N Standard Attention Blocks. The final output of the GDT is a probability distribution over potential actions.

mechanisms to direct the model’s focus towards specific objectives. This enables the GDT to streamline the decision-making process by emphasizing relevant data and omitting irrelevant information. Specifically, the GDT first employs embedding layers to transform raw data into a structured format amenable to analysis by neural networks. Subsequently, the model leverages a carefully orchestrated combination of GDABs and SABs to support the robot’s learning of actions. The GDAB refines the model’s concentration on achieving the set goals by accentuating features pertinent to those goals, while the SAB ensures that such goal-oriented learning is grounded in a comprehensive understanding of the surrounding context. The GDT’s effectiveness has been thoroughly evaluated in two simulated environments: Block-Pushing and Franka Kitchen. In these settings, the GDT achieved outstanding performance, surpassing other state-of-the-art models designed for robot behavior learning. These empirical findings demonstrate the GDT’s capability to generate robotic behaviors aligned with overarching goals, thereby underscoring the framework’s significant potential as a tool for behavior learning from play data.

2 Related Work

2.1 Demonstration and Play Data

Learning from Demonstrations (LfD) stands as a cornerstone in the realm of behavioral learning algorithms, offering a robust framework for assimilating expert knowledge into models for a variety of tasks [1]. Referred to as behavioral

datasets, the data utilized within this framework display a broad spectrum of variability: some datasets are enriched with goal or reward annotations [7], while others are more open-ended, lacking explicit reward or task labels [19]. Typically, these datasets presuppose that an expert executes a given task consistently and precisely, facilitating straightforward learning from high-quality examples.

In contrast, play datasets constitute a unique segment of unlabeled behavioral data, founded on the premise that demonstrations emanate from rational agents guided by some latent intent [10, 16]. The absence of explicit labels, coupled with the assumption of underlying intents, indicates that play datasets might feature a richer diversity in action distributions, presenting both challenges and opportunities for learning algorithms to infer intent and adapt accordingly. It is also pivotal to distinguish play-like behavior datasets from those used in standard Offline Reinforcement Learning (Offline RL), which typically encompass entirely random behaviors. Unlike the randomness in Offline RL datasets, play datasets offer a structured, yet exploratory, compilation of actions indicative of rational, albeit unspecified, objectives, providing a more nuanced substrate for behavior learning.

2.2 Behavior Learning

The feasibility of learning behavior from offline data using neural networks was first demonstrated by Pomerleau et al. [22]. Building upon these early applications of neural networks in behavior learning, the field has evolved towards more sophisticated methods, paving the way for innovative approaches that extend beyond traditional frameworks. Currently, behavior learning methods primarily fall into two categories: offline reinforcement learning (RL) and imitation learning.

Offline RL is characterized by its focus on learning from mixed-quality datasets that include reward labels. For instance, Fujimoto et al. [8] enhanced actor-critic methods in offline RL by integrating a Double Q-learning approach to mitigate overestimation biases. Kumar et al. [13] introduced the BEAR algorithm to stabilize off-policy Q-learning by addressing bootstrapping error, thus demonstrating robustness across various off-policy datasets and continuous control tasks without necessitating additional on-policy data collection. Subsequently, Kumar et al. [14] proposed the Conservative Q-learning (CQL) algorithm, which further advances offline RL by learning a conservative estimate of the Q-function to ensure lower-bound values for policies and counteract overestimation biases due to distributional shifts. In practice, CQL has significantly outperformed traditional offline RL methods, achieving substantial gains in environments with complex, multi-modal data distributions.

Imitation learning, alternatively, aims to model behavior from data without the necessity for explicit rewards. Ho and Ermon [11] innovatively combined imitation learning with generative adversarial networks (GANs), resulting in a model-free algorithm that can produce complex behaviors. Following this, Peng et al. [21] leveraged reinforcement learning to significantly improve agents' capabilities in mimicking varied movements and adapting to intricate environments.

However, the reliance on expert demonstrations for traditional imitation learning methods poses a challenge to their applicability in diverse scenarios. To address this issue, recent research has turned to play data for behavior learning. Lynch et al. [16] introduced Play-LMP, which learns to organize play behaviors in a latent space and reuses them at test time to achieve specific goals. Similarly, Shafullah et al. [23] developed the Behavioral Transformer (BeT) that enhances standard transformer architectures with action discretization and action correction tailored to learning from play data. Addressing BeT’s inability to select a targeted behavior mode during policy execution, Cui et al. [3] presented the Conditional Behavior Transformer (CBeT), refining BeT by integrating future-conditioned goal specification.

3 Methods

3.1 Problem Statement

Consider a play dataset comprising a sequence of paired tuples $(o_t, a_t) \in \mathcal{O} \times \mathcal{A}$, where o represents an observation at time t and a denotes the corresponding action. In this paper, to enhance the model’s focus on information pertinent to the goal state $g \in \mathcal{G}$, we augment the dataset with triple tuples (o_t, a_t, g) , enriching each original pair with a goal state. Our aim is to learn a policy $\pi : \mathcal{O} \times \mathcal{G} \rightarrow \mathcal{D}(\mathcal{A})$ that models the action distribution capable of transitioning the agent from current observation o_t to future goal state g . The ultimate objective of this behavior model is to optimize the policy, which is formulated as:

$$\pi^* = \operatorname{argmax}_{\pi} \prod_{(o_t, a_t, g) \in \mathcal{O} \times \mathcal{A} \times \mathcal{G}} \mathbb{P}[a \sim \pi(\cdot | o_t, g)] \quad (1)$$

3.2 Goal-Driven Transformer

In this paper, we propose a Goal-Driven Transformer that facilitates the behavior learning process by focusing on achieving specific goal states from play data. Figure 1 illustrates the overall framework of the Goal-Driven Transformer, which comprises three key components: the *Embedding Layer*, the *Goal-Driven Attention Block*, and the *Standard Attention Block*. Given a sequence of observations over k time steps, we utilize an observation embedding layer f_{OE} , a position embedding layer f_{PE} , and a goal embedding layer f_{GE} to encode the observations from time step t to $t+k$, denoted as $o_{t:t+k}$, the corresponding sequence of time steps, and the goal state into Observation Embeddings E_o , Position Embeddings E_p , and Goal Embeddings E_g , respectively. Notably, to enhance the training model’s stability, the goal state during the training process is defined as future observations from time step $t+h$ to $t+h+k'$ (where $h > k$). For inference, the goal state is determined by the observation at the final target time step. Subsequently, E_o , E_p , and E_g are processed through L Goal-Driven Attention Blocks and N Standard Attention Blocks. This configuration enables the aggregation of pertinent features and contextual information related to the

final goal through the Goal-Driven Attention Blocks, while the Standard Attention Blocks are employed to refine the understanding of contextual relationships between the observations and the ultimate goal. Ultimately, the Goal-Driven Transformer outputs an action probability distribution.

Embedding Layer. The embedding layer serves as the foundational entry-way into the Goal-Driven Transformer, translating the multifaceted input data into nuanced vector representations. This layer is segmented into three pivotal subcomponents, each dedicated to encoding different aspects of the input:

Observation Embedding Layer f_{OE} : Tasked with encoding the information contained within a sequence of observations $o_{t:t+k}$, this subcomponent transforms the discrete sequences of observations into dense, continuous vector representations (E_o).

Position Embedding Layer f_{PE} : The position embedding layer enhances the model by infusing it with essential positional information, achieved through the integration of positional encodings and observation embeddings. Specifically, this layer employs time steps to explicitly represent the sequential order of observations, granting the model the capability to discern and comprehend the sequence and relative timing of these observations.

Goal Embedding Layer f_{GE} : Focused on the higher-level goals (g) associated with the observation, this layer translates the specified goals into continuous vector representations (E_g). These goal embeddings distill the core objectives into a form that acts as a directional guide for the model, ensuring that the processing is consistently aligned with achieving the predefined goal states.

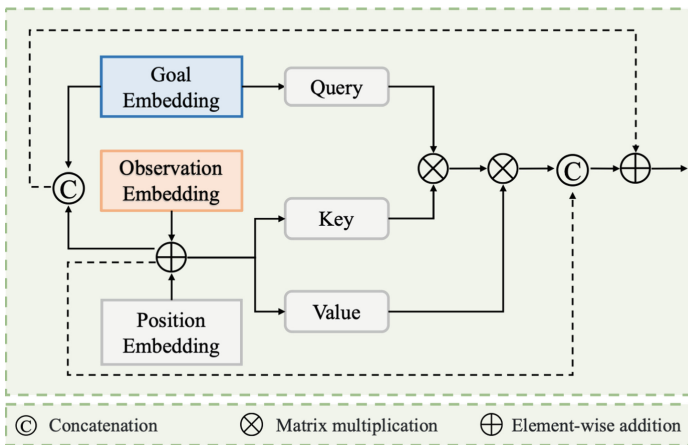


Fig. 2. Illustration of the proposed Goal-Driven Attention Block.

We can formulate the above layers as:

$$\begin{aligned}
 E_o &= f_{OE}(o_{t:t+k}), \\
 E_p &= f_{PE}(\{1, 2, \dots, k\}), \\
 E_g &= \begin{cases} f_{GE}(o_{t+h:t+h+k'}) & \text{for training,} \\ f_{GE}(o_g) & \text{for inference.} \end{cases}
 \end{aligned} \tag{2}$$

Goal-Driven Attention Block. The goal-driven attention block is a pivotal component within the Goal-Driven Transformer architecture, specifically designed to enhance the model’s focus on achieving predefined goal states. This module employs an advanced attention mechanism that dynamically weights the relevance of each input feature based on its significance towards accomplishing the goal at hand. By prioritizing the information that is directly pertinent to the goal, the goal-driven attention block enables the robot to more effectively discern and prioritize actions that contribute to the successful execution of action sequences leading to the goal state. Figure 2 illustrates the details of the Goal-Driven Attention Block. We first apply a linear layer to goal embedding E_g to generate query Q_g of attention mechanism and apply two linear layers to the addition of observation embedding and position embedding to generate key K_{op} and value V_{op} of attention mechanism. We formulate it as:

$$\begin{aligned}
 Q_g &= \text{Linear}(E_g), \\
 V_{op} &= \text{Linear}(E_o + E_p) \\
 K_{op} &= \text{Linear}(E_o + E_p)
 \end{aligned} \tag{3}$$

Then, we proceed to calculate the attention scores (S) using a scaled dot-product attention mechanism:

$$S = \text{softmax}\left(\frac{Q_g \cdot K_{op}^T}{\sqrt{D}}\right), \tag{4}$$

where the softmax function normalizes the dot products, providing attention scores that signify the relevance of each observation with respect to the goal. Next, we compute the goal-driven context vectors (C) by applying the attention scores (S) to the values (V_{op}):

$$C = S \cdot V_{op}. \tag{5}$$

These context vectors capture the goal-driven representations of the input data, emphasizing the parts of the observation that align with the specified high-level goal.

Finally, to ensure that we preserve valuable information from earlier stages of processing, we introduce a skip connection that combines the concatenation of goal-driven context and input embedding with the concatenation of goal and input embedding (O), which is formulated as:

$$F = [C, E_o + E_p] + [E_g, E_o + E_p] \tag{6}$$

This skip connection allows the model to retain the original contextual information while incorporating goal-driven insights. By integrating the goal embedding as the query, applying attention to the observation embedding, and adding a skip connection, the Goal-Driven Attention Block not only enhances the model’s focus on goal-related information but also maintains a rich contextual understanding of the input data. This comprehensive approach empowers the model to make well-informed decisions and actions based on both the high-level task objective and the underlying context.

3.3 Loss Functions

To train the Goal-Driven Transformer, we employ the methodologies outlined in prior work [3] by incorporating an action discretization module to partition ground-truth actions $\bar{\mathcal{A}}$ into discrete and continuous components. Specifically, the k-means clustering algorithm is utilized to identify K , the designated number of clusters, thereby generating a set of centroids $\{\mathcal{A}_1, \mathcal{A}_2, \dots, \mathcal{A}_K\}$. Each \mathcal{A}_i represents a cluster centroid within the action space $\bar{\mathcal{A}}$. The configurations for the k-means encoder and decoder are consistently maintained throughout the training and evaluation stages of the GDT. During the training phase of the GDT, the k-means encoder segments the ground-truth action $\bar{\mathcal{A}}$ into:

$$\begin{aligned}\mathcal{A}_c &= \min_{\mathcal{A}_i} \|\bar{\mathcal{A}} - \mathcal{A}_i\|_2, \\ \mathcal{A}_{\text{res}} &= \bar{\mathcal{A}} - \mathcal{A}_c,\end{aligned}$$

where \mathcal{A}_c denotes the nearest discrete action centroid to the ground-truth action, and \mathcal{A}_{res} represents the continuous residual component of the action. The GDT generates predictions characterized by $\pi(o, g)_d \in \mathbb{R}^K$ for the discrete aspect and $\pi(o, g)_c \in \mathbb{R}^{K \times |\bar{\mathcal{A}}|}$ for the continuous facet of actions. Consequently, the loss function is formulated as:

$$\mathcal{L} = L_{\text{focal}}(\pi(o, g)_d, \mathcal{A}_c) + \lambda \cdot L_{\text{MT}}(\langle \bar{\mathcal{A}} \rangle, \pi(o, g)_c),$$

where L_{focal} is the Focal loss predicated on negative log-likelihood [15], and L_{MT} is identified as the Masked Multi-Task loss [9].

4 Experiment

4.1 Experimental Datasets

Block-pushing Environment Drawing upon the multi-modal Block-pushing framework established by [6], our investigation explores intricate interaction demonstrations. In this context, an xArm robotic agent is designated the task of pushing two blocks, one red and one green, to their corresponding square targets, each matching the block’s color. The initial positioning of the blocks is subject to both randomness and noise. To ensure a fair comparison, we utilize a dataset, as introduced by [3], which consists of 1,000 demonstrations crafted via a deterministic controller.

Franka Kitchen Environment. Initially conceptualized by [10], the Franka Kitchen environment presents a sophisticated scenario for robotic tasks within a simulated kitchen, featuring seven possible tasks. This environment’s dataset comprises 566 demonstrations, meticulously assembled by humans using Virtual Reality (VR) controllers, that exemplify sequences encompassing four of the seven potential kitchen tasks.

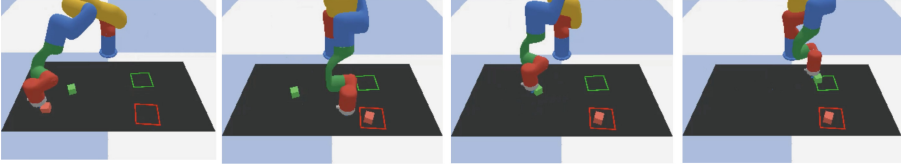


Fig. 3. Visualizations of key frames for completing a task in the Block Pushing environment, representing moments of reaching the red block, pushing the red block to the target red area, reaching the green block, and pushing the green block to the target green area. (Color figure online)

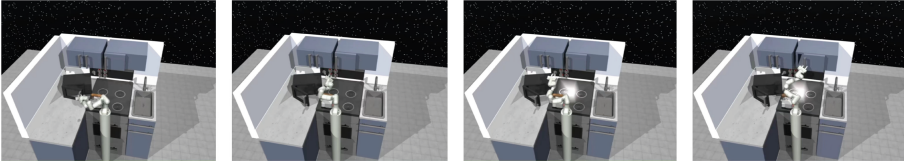


Fig. 4. Visualizations of key frames for completing four tasks in the Franka Kitchen environment, representing moments of completing the microwave, bottom burner, light switch, and slide cabinet tasks.

Table 1. Overall Results of the GDT Model and Other Baselines on the Block-pushing and Franka Kitchen Environments.

Environments	GCBC [16]	WGCSL [25]	Play-LMP [16]	RIL [10]	C-IBC [6]	GoFar [17]	GTI [18]	BeT [23]	C-BeT [3]	Ours
Block-pushing	0.06	0.10	0.02	0.07	0.01	0.04	0.04	0.34	0.90	0.96
Franka Kitchen	0.74	1.17	0.04	0.39	0.13	1.61	1.24	1.77	2.80	3.05

4.2 Baselines

The baselines of this study have been selected from state-of-the-art algorithms that learn from reward-free offline data. These include: Goal Conditioned BC

(GCBC) [5, 16], Weighted Goal Conditioned Supervised Learning (WGCSL) [25], Learning Motor Primitives from Play (Play-LMP) [16], Relay Imitation Learning (RIL) [10], Conditional Implicit Behavioral Cloning (C-IBC) [6], Generalization Through Imitation (GTI) [18], Goal-conditioned RL (GoFAR) [17], Behavior Transformers (BeT) [23] and Goal-Conditioned Behavior Transformers (C-BeT) [3]. Each baseline represents a unique approach to learning from reward-free offline data, providing a comprehensive comparison for evaluating the effectiveness of our proposed Goal-Driven Transformer.

4.3 Results

We initially evaluated the overall performance to compare our model with other models. For the Block-pushing environment, we report the success rate of accurately placing the green and red blocks in their respective target squares. For the Franka Kitchen environment, we report the average number of the four tasks that were successfully completed. Table 1 presents a comparison of our model with the baseline methods previously mentioned. The results show that our model achieved the highest performance in both settings. Specifically, in the Block-pushing environment, we achieved a success rate of 96%, outperforming all baseline methods. In the Franka Kitchen environment, we managed to complete an average of 3.05 out of four tasks, which is still above all the baseline models. Thus, it is evident that the proposed GDT model exhibits superior performance across different environments. To visually show the results of our model, we plotted key frames from a single task trajectory completed by our model in both the Block-pushing and the Franka Kitchen environments in Fig. 3 and Fig. 4. These visualizations demonstrate that our model has achieved satisfactory performance in accomplishing the tasks within both environments.

Subsequently, to conduct a more detailed analysis of our model’s performance in each environment, we refined the evaluation metrics for the respective settings. In the Block-pushing environment, we subdivided the results into “reaching” and “pushing” sub-tasks. We reported the success rates of reaching one and two blocks as R1 and R2, and the success rates of pushing one and two blocks into their respective targets as P1 and P2. For the Franka Kitchen environment, we detailed the success rates for completing one to five tasks. To ensure a fair comparison with prior models, we adhered to the baseline setup described in [23], selecting the following models as baselines: Multi-layer Perceptron with Mean Square Error (RBC) [24], Nearest Neighbor (NN) [2], Locally Weighted Regression (LWR) [20], Variational Autoencoders (VAE) [12], Normalizing Flow (Flow) [4], Implicit Behavioral Cloning (IBC) [6], Behavior Transformers (BeT) [23], and Goal-Conditioned Behavior Transformers (C-BeT) [3]. Table 2 lists the results of the proposed GDT in the Block-pushing and Franka Kitchen environments, respectively. As indicated in Table 2, for the relatively simpler reach tasks, several baselines achieved commendable results. However, their performance was limited for the more complex push tasks. In contrast, our model achieved a 99% success rate for pushing one block and a 97% success rate for pushing two blocks. Table 2 reveals that although the success rates of

all models gradually decreased as the number of tasks required to be completed increased, our model consistently maintained the highest success rates for completing 3, 4, and 5 tasks. These results reflect the effectiveness of applying an attention mechanism focused on the goal, which enables the model to concentrate on the final objective, thereby enhancing the success rate of task completion. This performance not only demonstrates the rationality of our model’s design but also its superiority, as it effectively utilizes goal-directed attention to streamline the path toward task completion.

Table 2. Results of the GDT model on Block-pushing and Franka Kitchen environments. R1 and R2 denote the success rates of reaching one and two blocks, respectively. P1 and P2 denote the success rates of pushing one and two blocks into their respective target areas. Numbers 1 to 5 denote the success rates for completing one to five tasks in the Franka Kitchen environment.

Models	Block-pushing				Franka Kitchen				
	Reach		Push		# Tasks completed				
	R1	R2	P1	P2	1	2	3	4	5
RBC [24]	0.67	0	0	0	0	0	0	0	0
1-NN [2]	0.49	0.05	0.01	0	0.90	0.72	0.44	0.17	0
LWR [20]	0.5	0.06	0	0	0.83	0.52	0.21	0	
VAE [12]	0.60	0.05	0	0	1	0	0	0	0
Flow [4]	0.59	0.02	0	0	0.04	0	0	0	0
IBC [6]	1	0.04	0.01	0	0.99	0.87	0.61	0.24	0
BeT [23]	1	0.99	0.96	0.71	0.99	0.93	0.71	0.44	0.02
CBeT [3]	1	1	0.91	0.91	1	0.8	0.53	0.41	0.05
Ours	1	1	0.99	0.97	1	0.86	0.76	0.51	0.09

4.4 Ablation Study

The Effectiveness of the Proposed Goal-Driven Attention Block.

To further validate the efficacy of our proposed Goal-Driven Attention Block (GDAB) i.e., its effectiveness in capturing goal-relevant features from observations, we conducted a comparison with a baseline model. This baseline integrates the goal into the model by simple concatenation with observations, instead of using attention mechanisms. To ensure a fair comparison, the baseline model was kept consistent with the GDT in all aspects except for the incorporation of the goal into the model. Table 3 lists the overall results of these two models in both the Block-pushing and Franka Kitchen environments. Our model achieved superior performance in both settings, thereby demonstrating that the GDAB enhances behavior learning by enabling the model to focus on features relevant to the goal.

Table 3. Comparison of GDT Performance with Different Methods of Integrating the Goal into the Model in the Block-pushing and Franka Kitchen Environments.

	Block-pushing Franka Kitchen	
GDT w/o GDAB	0.90	2.80
GDT w/ GDAB	0.96	3.05

Table 4. Results of GDT model under varying number of GDAB and SAB on the Block- pushing and Franka Kitchen Environments.

# GDAB	# SAB	Block-pushing	Franka Kitchen
0	7	0.87	2.6
1	6	0.96	3.05
2	5	0.92	2.85
4	3	0.89	2.7
6	1	0.8	2.5

The Number of GDAB and SAB. In Sect. 3.2, we discussed utilizing L Goal-Driven Attention Blocks (GDAB) and N Standard Attention Blocks (SAB) to generate the action distribution. This section primarily addresses the selection of values for L and N . Given the extensive range of possible combinations for L and N , it is impractical to examine each one exhaustively. Drawing from configurations commonly used in computer vision models, preliminary experiments suggested that employing a total of seven blocks might yield optimal results. Consequently, we focus on discussing the outcomes of the GDT model in the Block-pushing and Franka Kitchen environments under varying distributions of L and N . Table 4 presents the results for different configurations. It is observed that the configuration with one GDAB and six SABs performs better. The findings indicate that a greater quantity of GDABs doesn't automatically enhance the model's performance. This may occur because the model might become too goal-oriented, neglecting the need to grasp the complete environmental context and the action sequences that follow. An ideal configuration is achieved with a solitary GDAB that effectively grasps the goal's core, while a set of SABs provides comprehensive environmental information, ensuring the model's flexibility and adaptability across diverse situations. These insights reveal that although GDABs are critical in focusing the learning process on goals, they are not the only factor in the model's effectiveness but are nonetheless essential.

5 Conclusions

This paper proposes the Goal-Driven Transformer (GDT) framework for learning robot behavior from play data. GDT's key innovation is the inclusion of a

goal-driven attention block, enabling explicit integration of defined goals into the model’s attention mechanism. This facilitates selective focus on crucial segments of observational data for goal-specific behavior learning. Experimental validation, conducted in simulated environments such as the Block-pushing and Franka Kitchen scenarios, demonstrates that the GDT framework has achieved state-of-the-art performance in learning robot behavior from play data. We believe that the adoption of goal-driven paradigms will not only enrich the field of robot behavior learning but also encourage more sophisticated and adaptable robotic applications across diverse real-world scenarios.

References

1. Argall, B.D., Chernova, S., Veloso, M., Browning, B.: A survey of robot learning from demonstration. *Robot. Auton. Syst.* **57**(5), 469–483 (2009)
2. Arunachalam, S.P., Silwal, S., Evans, B., Pinto, L.: Dexterous imitation made easy: A learning-based framework for efficient dexterous manipulation. In: 2023 IEEE international conference on robotics and automation (icra). pp. 5954–5961. IEEE (2023)
3. Cui, Z.J., Wang, Y., Muhammad, N., Pinto, L., et al.: From play to policy: Conditional behavior generation from uncurated robot data. arXiv preprint [arXiv:2210.10047](https://arxiv.org/abs/2210.10047) (2022)
4. Dinh, L., Sohl-Dickstein, J., Bengio, S.: Density estimation using real nvp. arXiv preprint [arXiv:1605.08803](https://arxiv.org/abs/1605.08803) (2016)
5. Emmons, S., Eysenbach, B., Kostrikov, I., Levine, S.: Rvs: What is essential for offline rl via supervised learning? arXiv preprint [arXiv:2112.10751](https://arxiv.org/abs/2112.10751) (2021)
6. Florence, P., et al.: Implicit behavioral cloning. In: Conference on Robot Learning. pp. 158–168. PMLR (2022)
7. Fu, J., Kumar, A., Nachum, O., Tucker, G., Levine, S.: D4rl: Datasets for deep data-driven reinforcement learning. arXiv preprint [arXiv:2004.07219](https://arxiv.org/abs/2004.07219) (2020)
8. Fujimoto, S., Hoof, H., Meger, D.: Addressing function approximation error in actor-critic methods. In: International conference on machine learning. pp. 1587–1596. PMLR (2018)
9. Girshick, R.: Fast r-cnn. In: Proceedings of the IEEE international conference on computer vision. pp. 1440–1448 (2015)
10. Gupta, A., Kumar, V., Lynch, C., Levine, S., Hausman, K.: Relay policy learning: Solving long-horizon tasks via imitation and reinforcement learning. arXiv preprint [arXiv:1910.11956](https://arxiv.org/abs/1910.11956) (2019)
11. Ho, J., Ermon, S.: Generative adversarial imitation learning. *Advances in neural information processing systems* **29** (2016)
12. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint [arXiv:1312.6114](https://arxiv.org/abs/1312.6114) (2013)
13. Kumar, A., Fu, J., Soh, M., Tucker, G., Levine, S.: Stabilizing off-policy q-learning via bootstrapping error reduction. *Advances in Neural Information Processing Systems* **32** (2019)
14. Kumar, A., Zhou, A., Tucker, G., Levine, S.: Conservative q-learning for offline reinforcement learning. *Adv. Neural. Inf. Process. Syst.* **33**, 1179–1191 (2020)
15. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: Proceedings of the IEEE international conference on computer vision. pp. 2980–2988 (2017)

16. Lynch, C., Khansari, M., Xiao, T., Kumar, V., Tompson, J., Levine, S., Sermanet, P.: Learning latent plans from play. In: Conference on robot learning. pp. 1113–1132. PMLR (2020)
17. Ma, Y.J., Yan, J., Jayaraman, D., Bastani, O.: How far i’ll go: Offline goal-conditioned reinforcement learning via f -advantage regression. arXiv preprint [arXiv:2206.03023](https://arxiv.org/abs/2206.03023) (2022)
18. Mandlekar, A., Xu, D., Martín-Martín, R., Savarese, S., Fei-Fei, L.: Learning to generalize across long-horizon tasks from human demonstrations. arXiv preprint [arXiv:2003.06085](https://arxiv.org/abs/2003.06085) (2020)
19. Mandlekar, A., Zhu, Y., Garg, A., Booher, J., Spero, M., Tung, A., Gao, J., Emmons, J., Gupta, A., Orbay, E., et al.: Roboturk: A crowdsourcing platform for robotic skill learning through imitation. In: Conference on Robot Learning. pp. 879–893. PMLR (2018)
20. Pari, J., Shafiullah, N.M., Arunachalam, S.P., Pinto, L.: The surprising effectiveness of representation learning for visual imitation. arXiv preprint [arXiv:2112.01511](https://arxiv.org/abs/2112.01511) (2021)
21. Peng, X.B., Abbeel, P., Levine, S., Van de Panne, M.: Deepmimic: Example-guided deep reinforcement learning of physics-based character skills. ACM Transactions On Graphics (TOG) **37**(4), 1–14 (2018)
22. Pomerleau, D.A.: Alvin: An autonomous land vehicle in a neural network. Advances in neural information processing systems **1** (1988)
23. Shafiullah, N.M., Cui, Z., Altanzaya, A.A., Pinto, L.: Behavior transformers: Cloning k modes with one stone. Adv. Neural. Inf. Process. Syst. **35**, 22955–22968 (2022)
24. Torabi, F., Warnell, G., Stone, P.: Behavioral cloning from observation. arXiv preprint [arXiv:1805.01954](https://arxiv.org/abs/1805.01954) (2018)
25. Yang, R., et al.: Rethinking goal-conditioned supervised learning and its connection to offline rl. arXiv preprint [arXiv:2202.04478](https://arxiv.org/abs/2202.04478) (2022)



Adaptive Dynamic VSLAM: Refining Semantic-Geometric Fusion and Static Background Inpainting

Qi Mu^(✉), Baizhang Guo, Shuai Guo, and Zhanli Li

School of Computer Science and Technology, Xi'an University of Science and Technology,
Xi'an 710054, China
muqi@xust.edu.cn

Abstract. To tackle the challenges of adaptability and precision in VSLAM for dynamic environments, we propose a joint refined semantic-geometric approach that improves SLAM's performance across various dynamic settings. Our method integrates semantic segmentation networks with morphological processing to extract stable boundary features from potential dynamic objects accurately. By restoring depth information and applying geometric constraints that account for camera motion, we facilitate the precise identification and removal of dynamic objects. Additionally, we exploit static scene information to inpaint the background areas occluded by dynamic objects, thus enabling complete scene reconstruction. Quantitative evaluation using dynamic sequences from the TUM dataset reveals a significant reduction in RMSE for both high and low dynamic sequences compared to ORB-SLAM2, DynaSLAM, Yolo-SLAM and Blitz-SLAM. Specifically, there is 95.23%–32.30% decrease in RMSE for high dynamic sequences and 44.7%–5.52% decrease for low dynamic sequences, respectively. These results demonstrate the method's enhanced adaptability and localization accuracy across different levels of dynamic scenes. Furthermore, the dense reconstruction maps derived from the Static Background Inpainting process offer more complete static scene information than original maps, providing adequate technical support for autonomous localization and mapping of robots in dynamic environments.

Keywords: Adaptive Dynamic VSLAM · Refining Semantic-Geometric Fusion · Depth Image Restoration · Static Background Inpainting

1 Introduction

Simultaneous Localization and Mapping(SLAM) is a technique that enables robots to perceive and map their surroundings in real-time within unknown environments while simultaneously determining their position within the constructed maps [1]. This approach provides essential advantages, including real-time processing, scalability, and the autonomy to operate without relying on a prior map. Visual SLAM systems (VSLAM), which primarily rely on cameras as sensors, typically comprise several core modules, including front-end visual odometry, back-end optimization, loop closure detection, and

mapping. RGB-D cameras provide a significant benefit for VSLAM by offering per-pixel depth information directly, which greatly improves its real-time performance of VSLAM. As a result, VSLAM based on RGB-D has been increasingly adopted across various domains, including autonomous navigation [2, 3], virtual and augmented reality [4], and autonomous driving [5, 6].

In recent years, classical VSLAM systems [7–9] have demonstrated impressive performance on static environment datasets such as TUM, KITTI, and EuRoC. However, their effectiveness is typically based on the assumption of a static environment [10]. When dynamic objects are present, these systems may encounter challenges in matching feature points, adversely affecting the localization and mapping processes and even leading to system failure. For VSLAM systems to function effectively in dynamic environments, it is crucial to accurately identify and remove dynamic features, focusing solely on static features for localization and mapping. Currently, methods that integrate semantic and geometric information have proven effective. This approach first employs deep learning models to obtain semantic information in the image, accurately segmenting potential dynamic objects and enhancing the system’s comprehension and perception of the environment. Subsequently, by combining the geometric information of the scene, geometric constraint methods are used to analyze the correlation and consistency of feature points, distinguishing the motion states of potential dynamic objects. Finally, the features of genuine dynamic objects are removed, and the remaining static features are utilized for localization and mapping. However, current VSLAM methods that integrate semantic and geometric information still face several unresolved issues. Firstly, the features on the boundaries of dynamic objects obtained directly using semantic segmentation networks such as SegNet [11] and Mask-RCNN [12] have unstable dynamism. Secondly, holes in the depth images lead to inaccurate geometric information, impacting geometric constraint methods’ accuracy. Additionally, most geometric constraint methods struggle to maintain robustness across various dynamic scenes, such as camera movement, objects moving at high or slow speeds, and a high proportion of dynamic areas. Finally, the removal of dynamic objects leads to holes in dense mapping outcomes.

To address the issues above, we propose an adaptive dynamic VSLAM. This system accurately utilizes refined semantic and geometric information with varying levels of dynamics, such as different object velocities, varying proportions of dynamic area coverage, and varying states of camera motion or stillness. It effectively removes dynamic features and inpaints background information occluded by dynamic objects, utilizing only static information for localization and dense mapping. The main contributions of this paper are as follows:

- (1) **Dynamic Semantic Boundary Optimization.** We integrate pixel-level semantic segmentation networks with morphological methods to obtain stable boundary features for potential dynamic objects. This process is executed in a separate thread to ensure the visual odometry performs in real-time.
- (2) **Optimized Geometric Motion Recognition.** We restore the lost depth information due to holes. This refined geometric information is then integrated with geometric constraint methods that account for camera motion, enabling the accurate distinction of the actual motion states of potential dynamic objects.

- (3) **Static Background Inpainting.** We use optical flow between consecutive frames to complete the captured RGB and depth images, thereby inpainting the background information occluded by dynamic objects and reconstructing the static scene information. This process helps generate complete and accurate static dense mapping in dynamic environments.

2 Related Work

Currently, most dynamic SLAM systems treat dynamic features as outliers and remove them, relying solely on static features in the images for localization and mapping, which achieves particular effectiveness [13]. These systems primarily employ segmentation methods based on geometric constraints and segmentation methods integrating semantic-geometric information.

2.1 Segmentation Methods Based on Geometric Constraints

The segmentation method based on geometric constraints operates from the perspective of geometric information constraints, utilizing the static features remain unchanged between adjacent frames to identify dynamic objects. Yang [14] utilizes changes in the edges connecting the same pair of feature points in consecutive image frames to detect dynamic objects; Sun [15] employs sparse optical flow for dynamic object contour detection and further applies the Grab-Cut algorithm [16] for segmentation. Although these methods demonstrate effective dynamic feature removal, they are predicated on the assumption of a stationary camera, making them less suitable for scenarios involving camera motion.

To address the above issues, geometric constraint methods considering camera motion have been proposed. Wei [17] and Ai [18] initially calculate the homography matrix between adjacent frames using the RANSAC algorithm and then separate dynamic feature points using epipolar constraint and reprojection error. However, When most features in the image are dynamic, the RANSAC may fail to accurately compute the transformation between consecutive frames, which affects camera motion estimation; Additionally, when dynamic objects move slowly in the scene, the reprojection error between dynamic feature points in consecutive frames is small, and the variations in epipolar lines are not pronounced. Therefore, relying solely on the geometric relationship between adjacent frames makes it challenging to effectively distinguish objects in scenes with a high proportion of dynamic features and moving slowly. To address the challenges above, Islam [19] employs Multiple View Geometry, which identifies five local keyframes that are similar to the current frame's feature points and considers the geometric relationships of all feature points between multiple keyframes and the current frame. It effectively possesses strong geometric reliability even in scenarios with a high proportion of dynamic features and slow object motion.

The segmentation methods based on geometric constraints can ascertain the overall motion state based on the local features of objects. However, this method cannot provide semantic attributes or identification of objects in the scene. It can only judge the

actual dynamism of pixels in the current image frame, making it challenging to distinguish potential dynamic objects in the scene. This limitation results in issues such as localization drift and inaccurate mapping in SLAM systems [20].

2.2 Segmentation Methods Integrating Semantic-Geometric Information

Segmentation methods integrating semantic-geometric information combine deep learning based semantic labeling of feature points with geometric information to improve dynamic removal accuracy. The Dyna-SLAM system by Bescos [21] integrates semantic information from Mask R-CNN with Multiple View Geometry to remove dynamic features effectively. However, Mask R-CNN’s segmentation of dynamic object boundaries may be inaccurate, especially in scenes with a wide variety of objects. Additionally, when the camera rotation angle is too large, it is difficult to effectively inpaint the static background information occluded by dynamic objects, resulting in noise in dense mapping outcomes. The DS-SLAM system by Yu [22] divides dynamic objects into natural scenes using SegNet and constructs a semantic map. However, this system only utilizes epipolar constraint methods based on consecutive frames to distinguish object motion states. As a result, it struggles to achieve precise estimation in scenes with slow-moving objects. Moreover, the system lacks implementation of static background inpainting.

Deep learning can assign different semantic categories to each pixel in an image. However, this process requires pre-learning and training on sample data, thus emphasizing the judgment of the potential dynamicity of objects with prior knowledge. It cannot distinguish the actual motion states of objects in real-time [13]. Therefore, combining deep learning networks with geometric constraint methods to utilize semantic and geometric information [23] to filter out the scene’s dynamic features is an effective solution.

3 System Introduction

3.1 System Overview

We propose an adaptive dynamic VSLAM across various dynamic setting. This method accurately removes dynamic features and reconstructs static scenes by refining semantic-geometric fusion and inpainting static backgrounds. In the visual odometry stage, firstly, it integrates the YOLACT semantic segmentation network with morphological processing, which obtains semantic information about potential dynamic objects with stable boundary features. Then, it enhances the accuracy of geometric information by restoring scene depth information. Based on the refined semantic and geometric information, it employs Moving Consistency Check(MCC) and Multiple View Geometry(MVG) methods to distinguish object motion states, thereby removing dynamic features and using only static features to achieve localization. In the dense mapping stage, the FGVC hole-filling algorithm and Local Frames Static Information Completion are utilized to separately inpaint background for RGB and depth images, which achieves dense mapping containing only static scene information. The specific framework of this method is illustrated in Fig. 1.

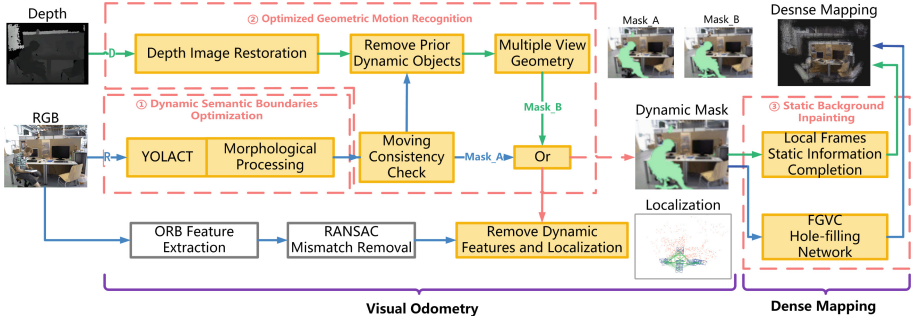


Fig. 1. The framework of adaptive dynamic VSLAM integrating refined semantic-geometric information and static background inpainting.

3.2 Dynamic Semantic Boundaries Optimization

We integrate the YOLACT semantic segmentation network [24] with morphological processing to segment and classify potential dynamic objects in the image. The YOLACT network enhances the segmentation capability of dynamic object edges, and morphological processing classifies the boundary features of objects as dynamic features, which enhances adaptability in scenes with a wide variety of objects.

Semantic Segmentation. The YOLACT network serves as the segmentation network, taking the RGB image as input to identify all potential dynamic objects in the scene and outputting them in the form of binary image masks. YOLACT generates pixel-level semantic segmentation and instance labels for each target box, producing a set of potential masks and their corresponding mask coefficients. Then, each potential mask is multiplied by its corresponding coefficient and summed to obtain the final semantic mask, as shown in Eq.(1).

$$mThreshold = \sum_{i=1}^n mask_i \times coefficient_i \tag{1}$$

Boundary Classification. In scenes with a wide variety of objects, the complexity of traditional feature extraction algorithms like SIFT and SURF might increase, which impacts the real-time performance of visual odometry. To tackle this, we employ ORB (Oriented Fast and Rotated Brief) for feature extraction due to its speed and low computational resource requirements. As shown in Fig. 2, ORB’s corner detection typically captures features spread along the object boundary. However, these features often exhibit dynamic object edge features and static background features, ultimately affecting the accuracy of visual odometry positioning.

To avoid this situation, a dilation layer is introduced for the masks obtained from the YOLACT. Equation (2) shows that a square structuring element B of size 10 × 10 is moved to the image pixel position(x, y). If the intersection between B and the maskM(x, y)is not empty, the M(x, y) in the output image is set to 1; otherwise, it is

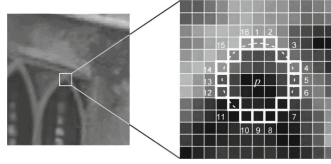


Fig. 2. Schematic diagram of ORB’s corner detection.

set to 0. Using this method to extend the boundaries of dynamic objects outward, the feature points on the boundaries are classified as dynamic features, thereby obtaining the semantic information of potential dynamic objects with stable boundary features.

$$M(x, y) \oplus B = \{x, y \mid (M)_{xy} \cap A \neq \emptyset\} \tag{2}$$

3.3 Optimized Geometric Motion Recognition

After marking potential dynamic objects, their motion states are ascertained by using geometric constraints. Initially, edge-preserving filling and curvature-driven diffusion methods are employed to restore the scene’s depth information, thereby enhancing the accuracy of geometric information. Subsequently, the MCC is utilized to preliminarily evaluate the motion states of potential dynamic objects and identify objects with a relatively large dynamic area in the scene. Then, the MVG is applied to more precisely identify objects with slow motion speeds. Applying this set of geometric optimization methods effectively enhances the performance of VSLAM across various dynamic settings.

Depth Image Restoration. Due to lighting conditions, depth images may contain holes, making it challenging to estimate scene geometry information accurately, which affects geometric constraint methods’ accuracy. We propose a depth image restoration algorithm based on edge-preserving filling and curvature-driven diffusion. This method effectively restores the missing information in depth images, thereby improving the accuracy of the obtained geometric information.

This method first uses the Canny operator to extract edges from the RGB image, as shown in Eq. (3), where $R(x, y)_{gray}$ is the grayscale image, and $I(x, y)$ is the edge structure map from Canny. Then, bitwise AND $I(x, y)$ with the binary mask to ascertain hole regions and edge positions. A neighborhood maximum value filling strategy is applied to fill the holes and edges, as shown in Eq. (4), where $I'(x, y)$ represents the hole and edge position maps, and $B(x, y)$ is the binary image of the hole region.

$$R(x, y)_{gray} \xrightarrow{\text{Canny}} I(x, y) \tag{3}$$

$$I(x, y) \perp B(x, y) = I'(x, y) \tag{4}$$

Finally, the CDD(curvature-driven diffusion) model is utilized to ascertain the diffusion information and diffusion strength, as shown in Eq. (5), where ∇u denotes the

gradient of the $u(x, y)$, which denotes the depth image pixel values, $\frac{1}{|\nabla u|}$ represents the diffusion coefficient, k denotes the curvature-driven factor, λ acting as the Lagrange multiplier in the constrained variational problem.

$$-\nabla u \cdot \left[\frac{g(k)}{|\nabla u|} \nabla u \right] + \lambda (u - u^0) = 0 \quad (5)$$

Dynamic Object Identification and Removal. To tackle the issue of segmentation networks failing to distinguish object motion states, we employ the combination of MCC and MVG, which consider camera motion. The MCC utilizes semantic information to assist geometric constraints, effectively removing dynamic objects with a large area proportion in the scene. MVG leverages the geometric relationships between multiple keyframes and the current frame to effectively remove slowly moving objects in the scene. This combination approach enhances the system's adaptability in scenes with different levels of dynamics.

The MCC takes the feature points detected by the semantic segmentation network as input and utilizes epipolar geometry constraints to ascertain the motion attributes of feature points. The fundamental matrix F and epipolar lines L are calculated using eq. (6). Then, eq. (7) is used to compute the distance D from the matched points in the previous frame to their corresponding epipolar lines. The point is considered a dynamic feature if D exceeds a threshold. When the number of dynamic feature points in a certain semantic category reaches a predefined value, the object corresponding to that semantic category is identified as a dynamic object. Here, f and $f - 1$ represent the current and previous frames, respectively. K denotes the camera's intrinsic parameters, H represents the homogeneous coordinates of feature points, and R is the rotation matrix.

$$F = K^{-T} t_{f(f-1)} \wedge R_{f(f-1)} K^{-1}, L = \begin{bmatrix} A \\ B \\ C \end{bmatrix} = FH_{f-1} \quad (6)$$

$$D = \frac{H_f^T F H_{f-1}}{\sqrt{A^2 + B^2}} \quad (7)$$

The MVG takes the RGB image and depth image after filtering out dynamic objects through the MCC as input. It selects five keyframes with the highest overlap with the current frame. As shown in Fig. 3, the feature point x in the keyframes is projected onto the current frame as x' and calculates the difference ∇z between the projected point's depth at the current frame position z_{proj} and the true depth z' . When $\nabla z > 0.4$, the feature point x is classified as a dynamic feature.

3.4 Static Background Inpainting

When dynamic objects are removed, the color and depth information occluded by these objects are lost, resulting in holes in the dense mapping results. We propose a method of integrating Flow-edge Guided Video Completion and Local Frame Static Information Completion, which complete RGB and depth images, respectively, to inpaint the static background information occluded by dynamic objects.

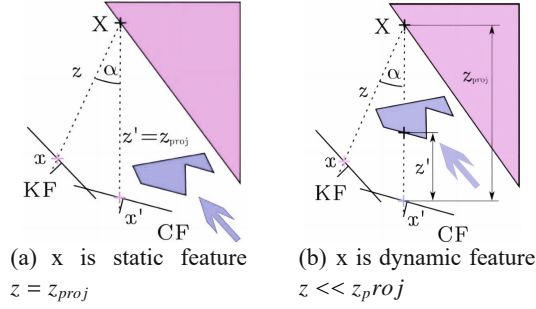


Fig. 3. Mathematical representation of multiple view geometry [21].

RGB Image Inpainting. Considering the abundant and prosperous texture information in RGB images, optical flow and edge information are used to assist in completing the missing regions in the scene to obtain more accurate and natural completion results. Optical flow can capture the motion information of objects in the scene; edge information can complete the structure and texture of the missing areas. It suits RGB images with diverse hierarchical structures and significant texture variations.

The collected RGB images are integrated with dynamic object masks using Flow-edge Guided Video Completion (FGVC) [25] to inpaint background information. This algorithm initially utilizes a FlowNet2 network to compute the optical flow. Then, it employs a Canny edge detector to extract object edge maps. After completing the optical flow edge using EdgeConnect [26], the algorithm performs completion by minimizing the gradients of all pixels in the region to be completed. Additionally, the algorithm establishes local and non-local neighborhoods and computes the color of the missing pixel c_k as candidate colors using Eq. (8), where p represents the given missing pixel, $k \in N(p)$ denotes the sets of local and non-local neighborhoods, w_k represents the optical flow cycle consistency error. Finally, it computes the weighted average of color gradients using Eq. (9) to address the seam issue at the edges of the completion result.

$$\tilde{I}(p) = \frac{\sum_k w_k c_k}{\sum_k w_k} \quad (8)$$

$$\tilde{G}_x(p) = \frac{\sum_k w_k A_x c_k}{\sum_k w_k}, \quad \tilde{G}_y(p) = \frac{\sum_k w_k A_y c_k}{\sum_k w_k} \quad (9)$$

Depth Image Inpainting. Considering the simple structural information and sparse textures in depth images, we utilize Local Frames Static Information Completion to fill in the depth holes created after removing dynamic objects. This approach enables the synthesis of a realistic depth image without altering its content, which ensures scene continuity and minimizes information loss in the completion results.

We utilize the static information from neighboring frames to complete the occluded information in the depth map. The images after Depth Image Restoration and the mask of dynamic objects serve as inputs. Following the methodology outlined in reference [21], we select the 20 frames following the current frame as local frames. With the

assistance of the dynamic object mask, only the static background information in the scene is retained. Then, we synthesize the static information from the local frames with that of the current frame, generating a depth map containing solely the static scene information.

4 Experiments and Analysis

4.1 Experimental Dataset

The TUM RGB-D dataset [27], released by the Technical University of Munich, evaluates visual SLAM localization and reconstruction in various scenes. It includes an extensive collection of indoor scene data captured by RGB-D sensors and ground truth trajectories for evaluating localization accuracy. The dataset comprises eight sequences with different object velocities, varying proportions of dynamic area coverage, and camera motion or stillness. We will conduct experiments using all dynamic sequences from the TUM dataset. Details of the experimental dataset are presented in Table 1:

Table 1. Introduction of the TUM dynamic scene dataset.

Sequence Name	Description
fr3_sitting_static	Low dynamic, stationary cameras
fr3_sitting_xyz	Low dynamic, camera movement along the XYZ axes
fr3_sitting_halfsphere	Low dynamic, camera movement along the semi-circular path
fr3_sitting_rpy	Low dynamic, camera rotation around the XYZ axes
fr3_walking_static	High dynamic, stationary cameras
fr3_walking_xyz	High dynamic, camera movement along the XYZ axes
fr3_walking_halfsphere	High dynamic, camera movement along the semi-circular path
fr3_walking_rpy	High dynamic, camera rotation around the XYZ axes

4.2 Evaluation Metrics for Localization Methods

Absolute Trajectory Error [27](ATE) refers to the deviation between the ground truth trajectory and the algorithm’s estimated trajectory, reflecting the computation’s accuracy and global consistency. It calculates a transformation matrix by fitting the best linear transformation between two distinct camera coordinate systems, allowing it to map computed poses to accurate poses. The ATE for the i -th frame is defined as follows in Eq. (10): Q_i represents the ground truth pose for the i -th frame, P_i represents the estimated pose for the i -th frame, $S \in SE(3)$ is the transformation matrix obtained through least squares fitting from the estimated pose to the ground truth pose.

$$F_i = Q_i^{-1} S P_i \quad (10)$$

Root Mean Square Error(RMSE) represents the error of all translational components over all time steps. It is defined as follows in Eq. (11): N is the number of poses, and $\text{trans}(F_i)$ represents the translational component of the absolute trajectory error F_i .

$$\text{RMSE}(F_{1:N}) = \left(\frac{1}{N} \sum_{i=1}^N \|\text{trans}(F_i)\|^2 \right)^{\frac{1}{2}} \quad (11)$$

4.3 Dynamic Semantic Boundaries Optimization

Semantic Segmentation. We utilize ResNet50-FPN as the backbone network for YOLACT, employing a pre-trained model trained on the COCO dataset [27]. The experimental results are shown in Fig. 4, which represents the original RGB image and the masks obtained using SegNet, Mask-RCNN, and YOLACT semantic segmentation networks, respectively. The experiments demonstrate that YOLACT can segment a greater variety of potential dynamic objects in scenes with diverse object categories. Moreover, the segmented objects exhibit more precise contour information compared to SegNet and Mask-RCNN.

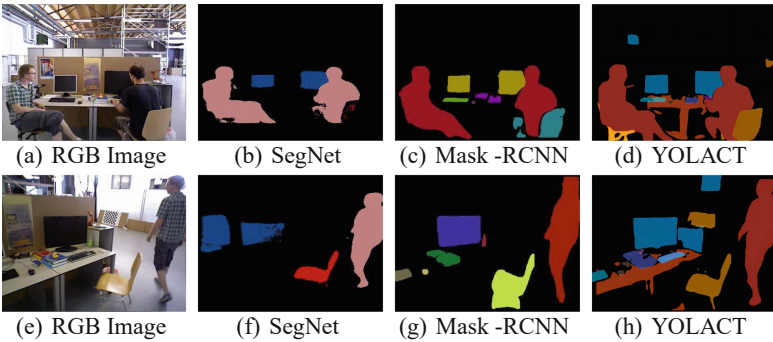


Fig. 4. Semantic segmentation network comparison experiment.

Boundary Classification. The classification results of boundary features of dynamic object masks are depicted in Fig. 5. The experiment utilizes images from the fr3_sitting_half and fr3_walking_half sequence with dynamic objects at different distances. Figures (a), (b), (e) and (f) present the masks generated by the semantic segmentation network along with the corresponding feature removal results. Green points denote static features in these figures. However, no matter whether the dynamic object is far away or near, its part of boundary features are defined as static features. Figures (c), (d), (g) and (f) show the masks after morphological processing and their corresponding feature removal results. The experiment demonstrates that the boundary classification method effectively partitions the feature points on the object boundary into dynamic features regardless of the distance of the dynamic object in the scene, thereby addressing the challenge of boundary feature classification.

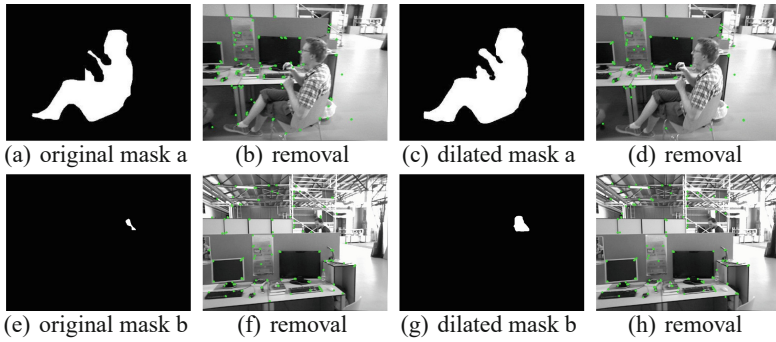


Fig. 5. Boundary classification result.

4.4 Optimized Geometric Motion Recognition

Depth Image Restoration. To acquire more precise geometric information, we employ edge-filling and curvature-driven diffusion methods to repair the depth images captured by the camera. Figure 6 shows the results of the restoration outcomes for the fr3_sitting_half dataset. The experiments indicate that the algorithm exhibits excellent restorative capabilities for holes in the depth images. Moreover, it achieves ideal filling effects around object edges, resulting in more precise edges and a more intact structure, effectively enhancing the precision of geometric information.

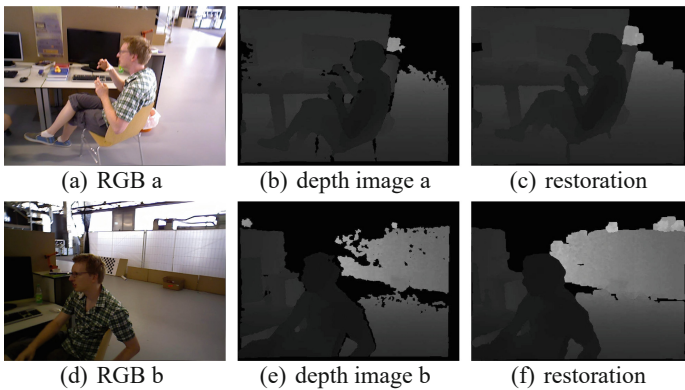


Fig. 6. Depth Image Restoration results.

Dynamic Object Identification and Removal. To demonstrate the adaptability of our method in scenes with large proportion of dynamic areas coverage, MCC is performed using the fr3_w_half dataset. The experimental results are shown in Fig. 7. It describes

the features classification and removal results obtained with and without MCC. Green points denote static features in these figures, while red points represent dynamic features. The experiment demonstrates that directly using semantic information for feature point classification in scenes with a large dynamic proportion may misclassify features of static objects such as computers and chairs as dynamic features. However, utilizing MCC can better identify and remove actual dynamic features in the scene, effectively enhancing system’s adaptability in scenes with high dynamic proportion.

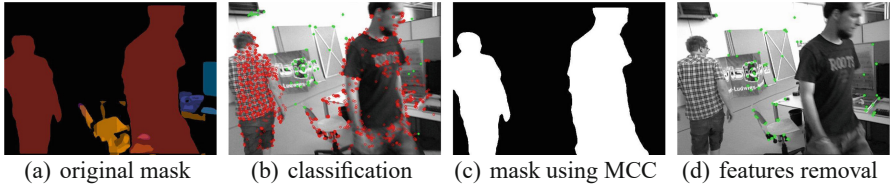


Fig. 7. Moving Consistency Check results.

To demonstrate the adaptability of our method in scenes with slowly moving objects, we conducted MVG on the fr3_s_static dataset. The experimental results are shown in Fig. 8, which shows the classification of feature points using the mask processed by MVG. The experiments indicate that the MVG method accurately distinguishes slowly moving chairs as dynamic objects, while the stationary chair on the right is still identified as a static object. The MVG method exhibits good removal capability for subtle dynamic features, effectively enhancing the system’s adaptability in scenes with slow object motion.

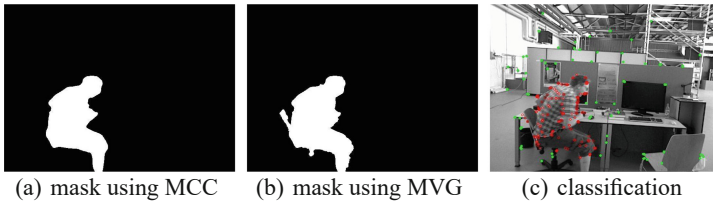


Fig. 8. Multiple View Geometry results.

4.5 Visual Odometry Localization

The visual odometry localization trajectories for dynamic scene sequences are presented in Tables 2. The red segments denote the differences between the actual trajectory(ground truth) and the localization trajectory(CameraTrajectory) obtained by our method. A more extended red segment indicates a more significant error. Objective

metrics, including the Root Mean Square Error(RMSE) of ATE and its standard deviation(S.D.), are summarized in Table 3.

Based on Tables 2, subjectively speaking, our method shows trajectories that are generally consistent with the actual trajectory across all eight data sequences. Our method’s results are consistently superior to ORB-SLAM2 [7] and ORB-SLAM3 [28]. Compared to the DynaSLAM [21], our method achieves better trajectory performance in the fr3_w_rpy and fr3_s_rpy sequences. Because DynaSLAM utilizes the Mask R-CNN semantic segmentation model, which has poor real-time performance and segments frames at intervals. This demonstrates that our method maintains better localization capabilities in scenes with camera motion.

Table 2. Trajectory maps of high dynamic sequence.

Sequence	ORB-SLAM2	ORB-SLAM3	DynaSLAM	Ours
fr3_w_xyz				
fr3_w_rpy				
fr3_w_static				
fr3_w_half				
fr3_s_rpy				
fr3_s_static				

As shown in Table 3, the RMSE and S.D. of ATE for our proposed method, are lower than those of ORB-SLAM2 [7], DynaSLAM [21], Yolo-SLAM [29] and Blitz-SLAM [30]. Specifically, on the high dynamic sequence dataset, the RMSE and S.D. of the proposed method decreased by 95.23%–32.30% and 95.62%–19.92% compared to other

methods respectively; On the low dynamic sequence dataset, the RMSE and S.D. of the proposed method decreased by 44.7%–5.52% and 56.7%–9.66% compared to other methods, respectively. This is because Yolo-SLAM, which uses YOLOv3, has difficulty to remove feature points located at the edges of dynamic objects. Blitz-SLAM removes features with the same depth values as dynamic features, which reduces the number of associated features. The conclusion is that the method proposed exhibits higher performance across various dynamic settings, such as scenes with different object velocities, varying proportions of dynamic area coverage, and camera motion or stillness.

Table 3. ATE comparison (unit: meters).

Sequence	ORB-SLAM2		Dyna-SLAM		Yolo-SLAM		Blitz-SLAM		Ours	
	RMSE	S.D.	RMSE	S.D.	RMSE	S.D.	RMSE	S.D.	RMSE	S.D.
fr3_w_xyz	0.8629	0.3116	0.0248	0.0085	0.0146	0.0070	0.0153	0.0078	0.0144	0.0072
fr3_w_rpy	0.8790	0.4035	0.0225	0.0282	0.2164	0.1001	0.0356	0.0220	0.0171	0.0183
fr3_w_static	0.1893	0.0059	0.0064	0.0048	0.0073	0.0035	0.0102	0.0052	0.0037	0.0031
fr3_w_half	0.7945	0.2051	0.0336	0.0115	0.0283	0.0138	0.0256	0.0126	0.0235	0.0125
fr3_s_xyz	0.0241	0.0062	0.0201	0.0058	/	/	0.0148	0.0069	0.0122	0.0053
fr3_s_rpy	0.0378	0.0230	0.0365	0.0516	/	/	/	/	0.0215	0.0349
fr3_s_static	0.0379	0.0472	0.0085	0.0051	0.0066	0.0033	/	/	0.0052	0.0039
fr3_s_half	0.0496	0.0171	0.0245	0.0112	/	/	0.0160	0.0076	0.0169	0.0078

4.6 Static Background Inpainting

As shown in Fig. 9, the experiment conducted background inpainting on the datasets fr3_walking_static and fr3_walking_xyz. Figures (b) and (f) depict the results after completion using the FGVC hole-filling algorithm. It can be observed that the repaired RGB images exhibit no ghosting, blurred boundaries, or uneven brightness, which demonstrates the algorithm’s capability to complete dynamic object edges and textures effectively. Figures (d) and (h) show the results using the Local Frames Static Information Completion method. The completed depth images have uniform grayscale values and the dynamic edges are aligned with the static background. The experiment confirms that the Static Background Inpainting method possesses excellent completion capability in dynamic scenes with camera motion or stillness.

4.7 Dense Mapping

Using the restored RGB images and depth images from the fr3_walking_static and fr3_walking_xyz datasets as input to achieve dense mapping. As shown in Fig. 10, Figures (a) and (c) depict the results of dense mapping without background inpainting. It can be observed that the static background information is occluded by the dynamic actions of the person, resulting in blurry mapping. Figures (b) and (d) show the results of

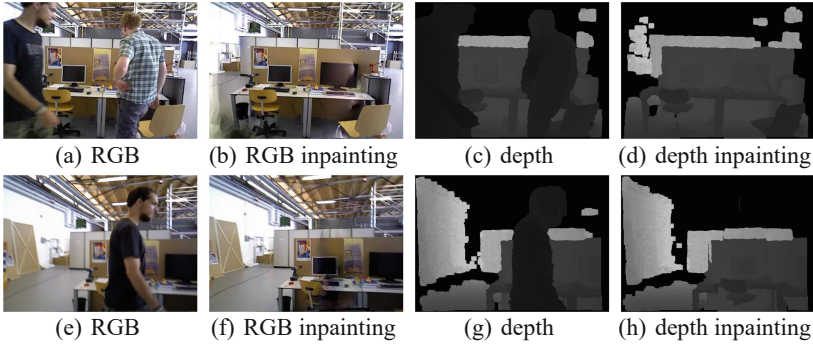


Fig. 9. Static Background Inpainting results.

dense mapping after applying the background inpainting. It is evident that the information on the mapping scene is richer than original maps, demonstrating the effectiveness of the Static Background Inpainting method.

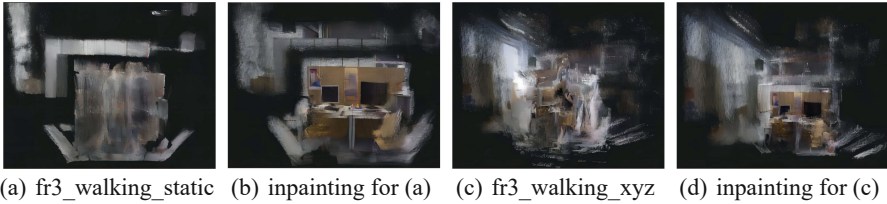


Fig. 10. Dense Mapping in dynamic scenes.

5 Conclusion

We propose an adaptive dynamic VSLAM, which demonstrates higher adaptability across various dynamic settings, such as scenes with different object velocities, varying proportions of dynamic area coverage, and camera motion or stillness. It accurately acquires and leverages high-precision semantic and geometric information effectively removes dynamic features, and inpaints background information occluded by dynamic objects, achieving localization and dense mapping solely using static information. Key factors contributing to its success include: (1) Integration of YOLACT semantic segmentation with morphological processing enables the acquisition of semantic information for potential dynamic objects with stable boundary features, which enhances adaptability in scenes with a wide variety of objects; (2) Depth Image Restoration provides accurate geometric information for geometric constraint methods, enhancing the ability to distinguish object motion states; Combining MCC and MVG effectively identifies and removes dynamic objects with large proportions of area and slow motion;

(3) Flow-edge Guided Video Completion and Local Frame Static Information Completion effectively complete static information occluded by dynamic objects in both RGB images and depth images, which provides richer scene information for dense mapping. The proposed method was quantitatively evaluated using dynamic sequences from the TUM dataset. The experimental results demonstrate that compared to ORB-SLAM2, DynaSLAM, Yolo-SLAM and Blitz-SLAM, our method effectively improves localization accuracy in both high and low-dynamic scenes. Additionally, utilizing the background inpainting method for dense 3D reconstruction maps more comprehensively represents scene static information compared to original map. Nonetheless, there are still some limitations in our method that require improvement. For instance, more precise background inpainting methods should be developed. Additionally, converting the dense maps drawn by the system, which currently occupy a high space rate, into octree maps with semantic information could be considered for future work.

References

1. Kazerouni, I.A., Fitzgerald, L., Dooly, G., Toal, D.: A survey of state-of-the-art on visual slam. *Expert Syst. Appl.* **205**, 117734 (2022)
2. Zhang, S., Zhao, S., An, D., Liu, J., Wang, H., Feng, Y., Li, D., Zhao, R.: Visual slam for underwater vehicles: a survey. *Comput. Sci. Rev.* **46**, 100510 (2022)
3. Geromichalos, D., Azkarate, M., Tsardoulis, E., Gerdes, L., Petrou, L., Perez Del Pulgar, C.: Slam for autonomous planetary rovers with global localization. *J. Field Robot.* **37**(5), 830–847 (2020)
4. Kuo, C.Y., Huang, C.C., Tsai, C.H., Shi, Y.S., Smith, S.: Development of an immersive SLAM-based VR system for teleoperation of a mobile manipulator in an unknown environment. *Comput. Ind.* **132**, 103502 (2021)
5. Cheng, J., Zhang, L., Chen, Q., Hu, X., Cai, J.: A review of visual slam methods for autonomous driving vehicles. *Eng. Appl. Artif. Intell.* **114**, 104992 (2022)
6. Zhang, H., Yang, Z., Tian, Y., Zhang, H., Di, B., Song, L.: Reconfigurable holographic surface aided collaborative wireless slam using federated learning for autonomous driving. *IEEE Trans. Intell. Veh.* **8**(8), 4031–4046 (2023)
7. Mur-Artal, R., Tardós, J.D.: ORB-SLAM2: an open-source SLAM system for monocular, stereo, and RGB-D cameras. *IEEE Trans. Rob.* **33**(5), 1255–1262 (2017)
8. Forster, C., Pizzoli, M., Scaramuzza, D.: SVO: fast semi-direct monocular visual odometry. In: 2014 IEEE International Conference on Robotics and Automation (ICRA), pp. 15–22. IEEE (2014)
9. Engel, J., Schöps, T., Cremers, D.: LSD-SLAM: large-scale direct monocular SLAM. In: European Conference on Computer Vision, pp. 834–849. Springer (2014)
10. Chappelle, K., Caron, G., Kanehiro, F., Sakurada, K., Kheddar, A.: Benchmarking cameras for open VSLAM indoors. In: 2020 25th International Conference on Pattern Recognition (ICPR), pp. 4857–4864. IEEE (2021)
11. Viswanathan, D.G.: Features from accelerated segment test (fast). In: Proceedings of the 10th Workshop on Image Analysis for Multimedia Interactive Services, London, UK, pp. 6–8 (2009)
12. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2961–2969 (2017)
13. Kesai, W., Xifan, Y., Yu, H., Min, L., Yuqian, L.: Review of visual slam in dynamic environment. *ROBOT* **43**(6), 715–732 (2021)

14. Yang, S., Fan, G., Bai, L., Li, R., Li, D.: MGC-VSLAM: a meshing-based and geometric constraint VSLAM for dynamic indoor environments. *IEEE Access* **8**, 81007–81021 (2020)
15. Sun, Y., Liu, M., Meng, M.Q.H.: Invisibility: a moving-object removal approach for dynamic scene modelling using RGB-D camera. In: 2017 IEEE International Conference on Robotics and Biomimetics (ROBIO), pp. 50–55. IEEE (2017)
16. Rother, C., Kolmogorov, V., Blake, A.: “GrabCut” interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph. (TOG)* **23**(3), 309–314 (2004)
17. Wei, T., Li, X.: Binocular vision slam algorithm based on dynamic region elimination in dynamic environment. *Robot* **42**(3), 336–345 (2020)
18. Ai, Q., Liu, G., Xu, Q.: An RGB-D SLAM algorithm for robot based on the improved geometric and motion constraints in dynamic environment. *Robot* **43**(2), 167–176 (2021)
19. Islam, Q.U., Ibrahim, H., Chin, P.K., Lim, K., Abdullah, M.Z.: MVS-SLAM: enhanced multi-view geometry for improved semantic RGBD slam in dynamic environment. *J. Field Robot.* **41**(1), 109–130 (2024)
20. Bojko, A., Dupont, R., Tamaazousti, M., Le Borgne, H.: Learning to segment dynamic objects using slam outliers. In: 2020 25th International Conference on Pattern Recognition (ICPR), pp. 9780–9787. IEEE (2021)
21. Bescos, B., Fàcil, J.M., Civera, J., Neira, J.: DynaSLAM: tracking, mapping, and inpainting in dynamic scenes. *IEEE Robot. Autom. Lett.* **3**(4), 4076–4083 (2018)
22. Yu, C., et al.: DS-SLAM: a semantic visual slam towards dynamic environments. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1168–1174. IEEE (2018)
23. Ai, Y., Rui, T., Lu, M., Fu, L., Liu, S., Wang, S.: DDL-SLAM: a robust RGB-D slam in dynamic environments combined with deep learning. *Ieee Access* **8**, 162335–162342 (2020)
24. Zhou, C.: Yolact++ Better Real-Time Instance Segmentation. University of California, Davis (2020)
25. Gao, C., Saraf, A., Huang, J.-B., Kopf, J.: Flow-edge guided video completion. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12357, pp. 713–729. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58610-2_42
26. Nazeri, K., Ng, E., Joseph, T., Qureshi, F.Z., Ebrahimi, M.: Edgeconnect: Generative image inpainting with adversarial edge learning. arXiv preprint [arXiv:1901.00212](https://arxiv.org/abs/1901.00212) (2019)
27. Sturm, J., Engelhard, N., Endres, F., Burgard, W., Cremers, D.: A benchmark for the evaluation of RGB-D SLAM systems. In: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 573–580. IEEE (2012)
28. Campos, C., Elvira, R., Rodríguez, J.J.G., Montiel, J.M., Tardós, J.D.: ORB-SLAM3: an accurate open-source library for visual, visual-inertial, and multimap slam. *IEEE Trans. Rob.* **37**(6), 1874–1890 (2021)
29. Wu, W., Guo, L., Gao, H., You, Z., Liu, Y., Chen, Z.: YOLO-SLAM: a semantic slam system towards dynamic environment with geometric constraint. *Neural Comput. Appl.* 1–16 (2022)
30. Fan, Y., Zhang, Q., Tang, Y., Liu, S., Han, H.: Blitz-SLAM: a semantic slam in dynamic environments. *Pattern Recogn.* **121**, 108225 (2022)



Hierarchical Visual Place Recognition with Semantic-Guided Attention

Wenwen Ming¹, Xucan Chen^{1,2(✉)}, Zhe Liu^{1,2(✉)}, Ruihao Li^{1,2}, and Wei Yi^{1,2}

¹ Academy of Military Sciences, Beijing 100091, China

² Intelligent Game and Decision Lab(IGDL), Beijing 100166, China
13973162892@139.com, liuzhe16@nudt.edu.cn

Abstract. Visual Place Recognition (VPR) is pivotal for navigation and robotic systems, facilitating accurate localization by recognizing previously visited places. In this paper, we present a novel hierarchical VPR approach that learns robust global and local features from semantic information. By leveraging semantic cues as prior information during the training process, our method implicitly guides the attention of the VPR model to focus on stable semantic features (e.g. buildings) while suppressing unreliable regions (e.g. persons, cars). Furthermore, we integrate the semantic-guided attention mechanism into the local matching process by extracting patch descriptors from the discriminative areas and prioritizing nearest neighbor matching on these patches, thereby reducing incorrect correspondences caused by dynamic or redundant patches. We evaluate the performance of our method against state-of-the-art techniques on public benchmark datasets with varying conditions and viewpoints. The experimental results demonstrate the superior performance of our proposed method, highlighting its robustness across diverse scenarios.

Keywords: Visual Place Recognition · Semantic Segmentation · Attention · Hierarchical VPR

1 Introduction

Visual Place Recognition (VPR) is an essential task for applications such as automatic navigation and mobile robots. It is a prerequisite for loop closure detection, which is a key component of the Simultaneous Localization And Mapping (SLAM) system. VPR is usually regarded as an image retrieval problem, involving the matching of an input query image from an unknown location to a set of reference images from known locations. Previous studies [8, 19, 32, 35] have adopted a hierarchical (two-stage) approach to the retrieval problem, involving global retrieval followed by re-ranking. This hierarchical approach aims to balance between matching accuracy and computational efficiency: initially using global descriptors to retrieve top-k candidate reference images, and subsequently refining these candidates through a re-ranking process.

However, VPR still faces a significant challenge: images captured from the real world have severe appearance changes due to lighting, weather variations,

dynamic object occlusions, etc. For a robust VPR system, it is necessary to extract features from discriminative and reliable objects (e.g. buildings) and suppress those from uninformative objects (e.g. persons, cars, sky). Some researches [16, 27] leverage semantic segmentation networks to identify long-term invariant objects for VPR tasks, but the effectiveness of these approaches hinges on the accuracy of the segmentation networks, which may struggle under challenging conditions such as low-light environments or nighttime scenarios. Instead, we propose a hierarchical VPR method that leverages semantic information to implicitly guide the model’s attention towards more robust global and local features. By incorporating semantic cues as supervision during the training process, our approach enables the VPR model to concentrate on features which are instrumental for discriminating the correct place (see Fig. 1), avoiding the generation and storage of additional semantic labels at test time. Additionally, we incorporate semantic-aware attention into local feature matching (patch-level) for re-ranking. By deriving task-related patch descriptors and enhancing the matching efficiency of patch pairs with higher attention scores, our method reduces false correspondences caused by dynamic or redundant patches in local matching (see Fig. 4).

In summary, we make specific contributions as follows:

- 1) We introduce a unified hierarchical VPR pipeline to learn robust global and local features for VPR tasks by incorporating high-level semantic information, which guides the attention of the VPR model to emphasize reliable areas while suppressing unreliable objects.
- 2) We improve local matching efficiency with semantic-guided attention, leveraging attention scores to filter out unreliable and redundant patch descriptors, thus eliminating wrong correspondences in local matching. Additionally, these scores weight the nearest neighbor matching, facilitating easier matching of critical patches.
- 3) We comprehensively test datasets with appearance variance and viewpoint variance, and the experimental results demonstrate the effectiveness of our method.

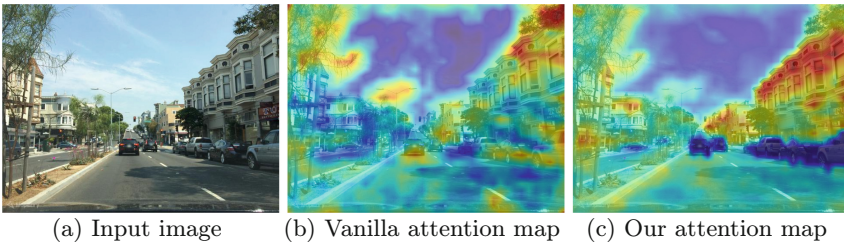


Fig. 1. Visualization of the attention maps of the baseline model and our model. The vanilla model’s attention is scattered and it pays attention to trees and cars. Our method is guided by semantic weight and focuses on robust areas such as buildings.

2 Related Work

Global and Local Descriptors for VPR: Early VPR global descriptors describe the entire image through a single feature representation [13,28], or by some algorithms [12,21,33] that aggregate local descriptors [5,25]. With the advent of deep learning, NetVLAD [4] is a new generalized VLAD [21] layer that can be pluggable into any CNN architecture and learned end-to-end. Building on the success of NetVLAD, its variant CRN [23] introduces context-aware feature reweighting, while SFRS [18] refines image-to-image similarities into self-supervised image-to-region similarities. Other global descriptor methods include AP-GeM [31], which optimizes the global mAP in listwise loss formulations. Recently, a large number of VPR methods have demonstrated superior performance by utilizing new training techniques on large-scale datasets. Notably, Cos-Place [6] casts the training as a classification task, while MixVPR [2] is trained on a large dataset called GSV-Cities [1] and incorporates a global relationship into each feature map through a series of Feature-Mixer blocks which consist of MLPs.

Hierarchical (two-stage) VPR methods [8,19,35] have gained popularity for enhancing the performance of global descriptors. These methods typically begin by obtaining the top-k candidates through global retrieval, followed by cross-matching using local descriptors. They subsequently employ geometric verification techniques like RANSAC [15] to eliminate false matches and re-rank candidates. Patch-NetVLAD [19] derives patch-level features from NetVLAD and introduces a multi-scale fusion of patch features with complementary scales in a complete feature space. TransVPR [35], based on the ViT [14] architecture, employs a multi-level attention mechanism to select key-patch descriptors via a fusion attention mask.

Semantic Information in VPR: Semantic information plays a crucial role in addressing challenges like appearance changes in VPR [10,16,17,22,24,27,32,38]. Some studies [16,27] utilize semantic segmentation networks to extract pre-defined stable semantic categories such as roads, buildings, etc. Based on [16,17] concatenates the appearance-based descriptor with the semantics-based descriptor to eliminate the inaccurate labeling of the segmentation network in extreme environments. StructVPR [29] inputs the segmentation images into the CNN network and employs the knowledge distillation method to enhance the structural knowledge of the RGB global features, thereby improving the stability of the features in a changing environment.

Some recent works [10,29,30] integrate semantic information with attention mechanisms. [29] employs a multi-scale attention module to guide segmentation process, enhancing robustness in global descriptor learning, albeit requiring additional training of the segmentation network. Closely related to our work is de-attention [10], which diminishes the influence of dynamic objects with semantic guidance by using binary semantic masks indicating their labels. Instead, our approach use semantic masks with varied predefined weights. This allows the

VPR model to suppress dynamic objects and reduce the influence of redundant elements (e.g., sky), while also emphasizing the importance of long-term features (e.g., buildings).

3 Method

We propose a unified hierarchical VPR pipeline, as illustrated in Fig. 2. By using semantic guidance during training, we focus the model’s attention on more stable regions, and the reweighted feature maps are then aggregated into global descriptors. In the re-ranking stage, we apply semantic-guided attention to both the extraction and matching of patch descriptors. We then use RANSAC [15] to calculate the spatial consistency score and obtain the final refined retrieval results. Next, we delve into the details of each stage of our pipeline, starting with the global retrieval process in Sect. 3.1, followed by the local matching for re-ranking process in Sect. 3.2.

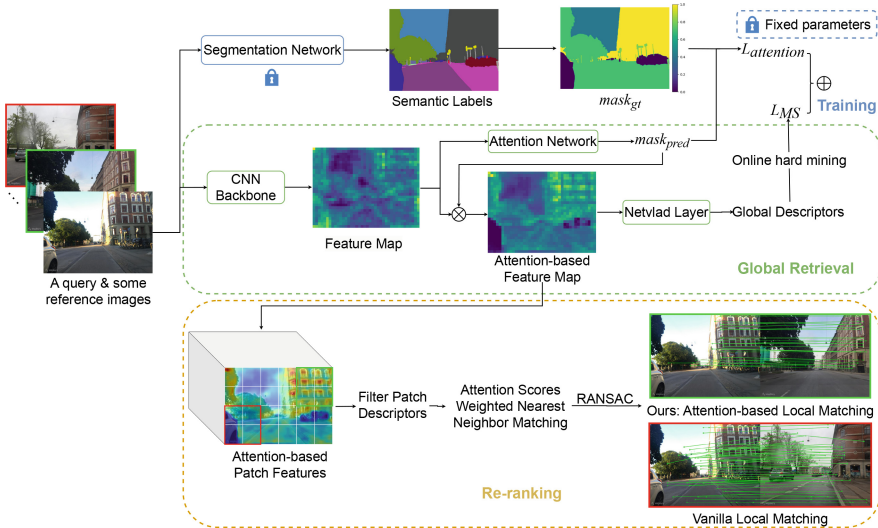


Fig. 2. Illustration of our unified hierarchical VPR pipeline.

3.1 Global Retrieval

To achieve robust global feature learning for VPR, we incorporate semantic information to guide attention during the training process. Notably, our VPR model autonomously generates semantic-aware descriptors at test time, operating in an end-to-end manner without requiring explicit semantic labels.

We utilize an advanced off-the-shelf segmentation network, Mask2Former [9], trained on Cityscapes dataset [11], to generate semantic labels for input images. Based on the robustness of different semantic categories to appearance changes, we group the 20 semantic labels from the Cityscapes dataset into five categories and assign different weights empirically to generate $mask_{gt}$. As shown in Fig. 3, for long-term stable building, we regard this semantic class as the most reliable and assign it a weight of 1.0. In contrast, we set the weights of all dynamic objects to 0 due to their constant movement, which makes visual place recognition difficult. Semantic categories that are relatively static but not as stable as building, such as pole, traffic light, and traffic sign—receive slightly lower weights of 0.8. The sky, which frequently changes appearance due to weather and day-night transitions and is often redundant in visual place recognition images, is given a lower weight of 0.4. Additional semantic categories are assigned a medium weight of 0.7.

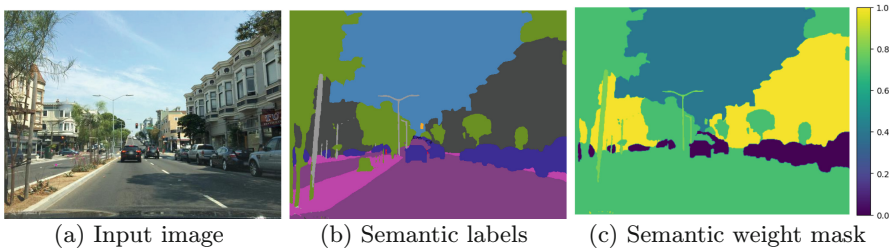


Fig. 3. The semantic labels of the image are used to generate the corresponding semantic weight mask.

With the supervision of semantic information, we train our VPR model, which consists of a CNN backbone network, an attention network, and a NetVLAD [4] layer. The attention network follows the de-attention [10] setting and comprises a Contextual Reweighting Network (CRN) [23] and a sigmoid function to generate a weight mask, $mask_{pred}$. This $mask_{pred}$ dynamically reweights the CNN feature map through element-wise multiplication across all channels. The resulting attention-based feature map serves as input to the NetVLAD layer to generate a global image representation.

To make the visual place recognition model highlight stable areas and suppress unreliable areas as we expect, the attention loss is defined as:

$$L_{attention} = MSE(mask_{pred}, mask_{gt}) \quad (1)$$

where MSE denotes the Mean Squared Error Loss.

Our VPR model is trained on the large-scale GSV-Cities [1] dataset and we use Multi-Similarity loss [36] due to its excellent performance.

$$L_{MS} = \frac{1}{N} \sum_{q=1}^N \left\{ \frac{1}{\alpha} \log \left[1 + \sum_{p \in \mathcal{P}_q} e^{-\alpha(S_{qp}-m)} \right] + \frac{1}{\beta} \log \left[1 + \sum_{n \in \mathcal{N}_q} e^{\beta(S_{qn}-m)} \right] \right\} \quad (2)$$

where for the query image q in the batch, \mathcal{P}_q and \mathcal{N}_q are the positive samples set and the negative samples set respectively, and α , β and m is the hyperparameters controlling the weight.

The overall VPR model is trained jointly by Multi-Similarity loss and attention loss, weighted by λ , as follows:

$$L_{total} = L_{MS} + \lambda L_{attention} \quad (3)$$

At test time, both query images and reference images undergo processing through our network to obtain robust global descriptors for distinguishing places. These descriptors are then used to retrieve the top-k candidates.

3.2 Local Matching for Re-ranking

In this section, we will introduce the integration of semantic-aware attention into local feature matching in two parts. The first part explains the method for filtering out non-discriminative regions during patch descriptor extraction. The second part details the strategy for assigning different weights through attention during patch matching.

Attention-Based Patch Filtering: We perform re-ranking using Patch-NetVLAD [19], which extracts densely-sampled patch descriptors within the full feature map. For the attention-based feature map $F \in \mathbb{R}^{H \times W \times D}$ from the global branch, we employ a sliding window mechanism to extract a set of $d_x \times d_y$ patches (where $d_y = d_x$ in square patches) with stride s_p . The total number of extracted patches is calculated as follows:

$$n_p = \left\lfloor \frac{H - d_y}{s_p} + 1 \right\rfloor * \left\lfloor \frac{W - d_x}{s_p} + 1 \right\rfloor, d_y, d_x \leq H, W \quad (4)$$

The original Patch-NetVLAD extracts patch descriptors from all regions of an image and performs exhaustive cross-matching, leading to high storage cost and long feature matching time per image. To mitigate these issues, we propose a method to extract task-relevant key patch descriptors. We filter the patch descriptors based on the previous attention network output $mask_{pred}$.

Specifically, we perform average pooling with kernel size d_y (or d_x) and stride $s_p = 1$ on the predicted attention mask of each image, as follows:

$$mask_{pooling}(i, j) = \frac{1}{d_x \cdot d_y} \sum_{p=0}^{d_x-1} \sum_{q=0}^{d_y-1} mask_{pred}(i+p, j+q) \quad (5)$$

where $mask_{pooling}(i, j)$ is the average pooling score at position (i, j) , corresponding to the top-left corner of the patch in the feature space.

The attention score corresponding to the k^{th} patch descriptor is as follows:

$$score(k) = mask_{pooling}(i, j) \quad (6)$$

We then retain only the patch descriptors with attention scores exceeding a specified threshold, effectively filtering out patches that are not useful for visual place recognition, as shown in Fig. 2.

Attention Scores Weighted Nearest Neighbor Matching: The original Patch-NetVLAD [19] performs mutual nearest neighbor matching by exhaustively comparing the descriptor sets of query image $\{\mathbf{f}_i^q\}_{i=1}^{n_p}$ and the descriptor sets of reference image $\{\mathbf{f}_i^r\}_{i=1}^{n_r}$. Matching patches are identified as follows:

$$\mathcal{P} = \{(i, j) : i = \text{NN}_r(\mathbf{f}_j^q), j = \text{NN}_q(\mathbf{f}_i^r)\} \quad (7)$$

where NN denotes the nearest neighbor matching obtained by calculating the minimum Euclidean distance between the image descriptor sets.

After filtering out unreliable patch descriptors, the Euclidean distance matrix is as follows:

$$D = \begin{bmatrix} d(f_1^q, f_1^r) & \cdots & d(f_1^q, f_{n_r}^r) \\ \vdots & \ddots & \vdots \\ d(f_{n_q}^q, f_1^r) & \cdots & d(f_{n_q}^q, f_{n_r}^r) \end{bmatrix} \quad (8)$$

where the number of query descriptors and reference descriptors after filtering is n_q and n_r respectively.

The original Patch-NetVLAD assumes equal importance for each patch descriptor, which can lead to false matches due to unreliable patch descriptors. To address this, we optimize the matching process by weighting the distance matrix with attention scores, making more discriminative patch descriptors easier to match. The weight of each patch descriptor is determined by its corresponding attention score (as described by Eq. 6):

$$\omega(f_i) = e^{-\alpha \cdot score} \quad (9)$$

where α is a hyperparameter that controls the influence of the attention score on the weight.

The weighted distance matrix is given by

$$D' = \begin{bmatrix} \omega(f_1^q)\omega(f_1^r) & \cdots & \omega(f_1^q)\omega(f_{n_r}^r) \\ \vdots & \ddots & \vdots \\ \omega(f_{n_q}^q)\omega(f_1^r) & \cdots & \omega(f_{n_q}^q)\omega(f_{n_r}^r) \end{bmatrix} \odot D \quad (10)$$

where each pair of descriptors is matched with their respective attention weights. This approach facilitates the matching of more stable patch descriptors while suppressing the matching of unreliable ones, as shown in Fig. 4.

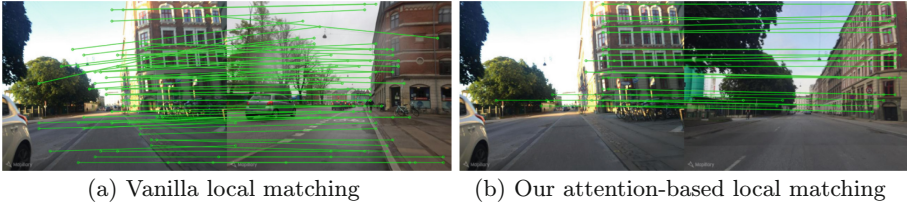


Fig. 4. Visualization of the top-1 results of the vanilla local matching and our attention-based local matching. Our method successfully retrieves the correct reference image, with the matching patches concentrated in the building area. While the vanilla local matching method produces incorrect match, with the matching patches including unreliable objects such as the bicycles and sky.

After performing mutual nearest neighbor matching, we apply RANSAC [15] for geometric verification. When fitting the homography, we take the center coordinates of each patch as the keypoints, filtering the keypoints according to the previously mentioned criteria, which reduces the computation time. Our spatial consistency score is determined by the number of inliers and is normalized by the number of patches after filtering. The final results are obtained by re-ranking the top-k global retrieval candidates using the spatial consistency score.

4 Experiments

4.1 Datasets and Evaluation

Our experiments primarily focus on urban scene datasets due to the use of the segmentation model pre-trained on the Cityscapes dataset [11]. These datasets encompass various challenges such as viewpoint changes, lighting changes, weather changes, etc. Table 1 and Table 2 summarize the information of the datasets list. Pitts30k-test [34] is a collection of Google Street View images of the city of Pittsburgh, and each place consists of 24 images from different perspectives with severe viewpoint variations and moderate condition variations. MSLS-val [37], a subset of the Mapillary Street-Level Sequences dataset, serves as

a public validation dataset, which covers a variety of challenging condition variations due to lighting, weather, dynamic objects, etc. The Tokyo 24/7 [3] includes reference images from Google Street View images of Tokyo, and query images captured via smartphones, showcasing significant lighting and viewpoint variations. More challenging datasets come from the recent Svox [7] of 5 domains—Snow, Rain, Sun, Night, and Overcast. Svox extracted images from Google Street View in the city of Oxford for the reference images, and images from Oxford RobotCar [26] as query images.

We use Recall@K as the evaluation metric for our experiments. Recall@K is defined as the percentage of queries for which at least one of the top K reference images retrieved is within the ground truth threshold. We set the threshold distance as 25 m, following precedents from previous work [1, 2, 7, 19].

Table 1. Summary of the popular datasets in experiments.

Dataset Name	Pitts30k-test	MSLS-val	Tokyo 24/7
# Query Images	6.8k	740	315
# Reference Images	10k	18.9k	76k
Description	viewpoint	weather, day/night	day/night

Table 2. Summary of the more challenging datasets in experiments.

Dataset Name	Svox	SnowSvox	SunSvox	NightSvox	RainSvox	Overcast
# Query Images	870	854	823	937	872	
# Reference Images	17k	17k	17k	17k	17k	
Description	weather	weather	day/night	weather	weather	

4.2 Implementation Details

We use ResNet50 [20] as the backbone and crop it to the *conv4_x* layer. Then CRN [23] is used as the attention module, and the weighted feature map is input to the aggregation layer NetVLAD [4]. For NetVLAD, the number of clusters is 16, resulting in a 16k-dimensional representation, and we use PCA for dimensionality reduction.

We train our model following the standard framework of GSV-Cities [1]. Our training batch consists of 80 places, each with 4 images, totaling 320 images per batch. We employ the Adam optimizer with a learning rate of $1.3e-4$, adjusted according to the batch size, and train for a maximum of 30 epochs. To ensure consistency between the training set images and semantic segmentation results, we removed the random augmentation used in the GSV-Cities framework.

For re-ranking, we adopt Patch-NetVLAD [19], using square patch sizes 2, 5 and 8 with corresponding weights of $w_i = 0.45, 0.15, 0.4$ for multi-scale fusion. Through experiment with the MSLS-val [37] dataset, we determined that 0.4 is the optimal threshold for filtering patch descriptors. Additionally, we set α to 0.01 to adjust the influence of the attention scores on the weighting of nearest neighbor matching.

4.3 Comparison to State-of-the-Art Methods

In this section, we compare our method against several state-of-the-art VPR methods, including global descriptors like NetVLAD [4], AP-GeM [31], SFRS [18], CosPlace [6], and MixVPR [2], using their official model checkpoints. Notably, CosPlace and MixVPR represent SOTA techniques trained on large-scale datasets. We also compare our approach against two-stage VPR methods such as Patch-NetVLAD [19], TransVPR [35], and StructVPR [32]. For Patch-NetVLAD, we use its performance-focused configuration, denoted as Patch-NetVLAD-p. TransVPR employs a transformer architecture. For StructVPR, we compare it using RANSAC [15] as the re-ranking backend for fairness, denoted as StructVPR-SP-RANSAC. Since StructVPR has not released code, we report results only on datasets available in its paper. Our results are presented both with and without re-ranking.

Table 3 presents the quantitative results for the Pitts30k-test, MSLS-val, and Tokyo 24/7 datasets. Our method demonstrates exceptional average performance across all datasets. On the Pitts30k-test and MSLS-val datasets, our method achieves results comparable to MixVPR and StructVPR. Notably, our method outperforms all other methods on the Tokyo 24/7 dataset with a 7% absolute increase in R@1.

Table 4 shows the performance on more challenging datasets under extreme weather and lighting conditions. Our method achieves impressive results on various datasets, improving R@1 by 5.1%, 5.7%, 2.2%, and 1.2% on Svox Sun, Svox Night, Svox Rain, and Svox Overcast, respectively, and achieving comparable performance on the Svox Snow dataset.

Qualitative results are shown in Fig. 5. Our examples contain challenging viewpoint changes, as well as extreme appearance changes such as dynamic object occlusion, weather changes, day-to-night illumination changes, etc. The results demonstrate the robustness of our method to complex environments.

4.4 Ablation Studies

Threshold for Filtering Patch Descriptors: We conduct an ablation experiment to determine the optimal threshold for filtering patch descriptors using attention scores. The Fig. 6 shows the R@1 results corresponding to thresholds ranging from 0 to 1, in increments of 0.1, on the MSLS-val and Svox Snow datasets. The recall rate (R@1) remains relatively stable when the threshold is set between 0 and 0.6. Beyond this point, the recall rate drops significantly

Table 3. Comparison to state-of-the-art methods on popular datasets. The best is highlighted in bold and the second is underlined.

Method	Pitts30k-test			MSLS-val			Tokyo 24/7			Average		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
NetVLAD [4]	85.0	92.1	94.4	58.5	70.5	74.7	65.1	75.6	78.1	69.5	79.4	82.4
AP-GeM [31]	80.7	91.4	94.1	64.6	75.1	77.8	54.3	68.3	75.2	66.5	78.3	82.4
SFRS [18]	89.0	94.6	95.9	69.7	79.6	82.3	76.5	86.3	88.6	78.4	86.8	88.9
CosPlace [6]	88.4	94.6	95.7	82.4	89.9	92.2	80.0	88.6	91.1	83.6	91.0	93.0
MixVPR [2]	91.6	95.6	<u>96.4</u>	88.1	93.2	94.1	85.7	<u>91.4</u>	<u>93.7</u>	<u>88.5</u>	<u>93.4</u>	<u>94.7</u>
Ours w/o Reranking	88.9	94.4	95.7	84.7	91.0	92.7	75.6	87.6	89.5	83.1	91.0	92.6
Patch-NetVLAD-p [19]	88.7	94.5	95.9	79.5	86.2	87.7	<u>86.0</u>	88.6	90.5	84.7	89.8	91.4
TransVPR [35]	89.0	94.9	96.2	86.8	91.2	92.4	79.0	82.2	85.1	84.9	89.4	91.2
StructVPR-SP-RANSAC [32]	89.4	<u>95.2</u>	96.5	<u>87.3</u>	<u>91.4</u>	92.8	-	-	-	-	-	-
Ours	<u>90.4</u>	<u>95.2</u>	96.3	87.0	<u>91.4</u>	<u>93.2</u>	93.0	94.6	95.2	90.1	93.7	94.9

Table 4. Comparison to state-of-the-art methods on more challenging datasets. The best is highlighted in bold and the second is underlined.

Method	Svox Snow			Svox Sun			Svox Night			Svox Rain			Svox Overcast		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
NetVLAD [4]	61.1	75.5	80.5	39.2	55.5	61.7	9.1	18.2	25.3	58.6	73.3	77.3	73.5	85.9	89.6
AP-GeM [31]	66.8	80.5	84.6	40.0	57.8	64.8	19.2	33.9	40.5	56.4	72.6	79.1	73.9	84.5	88.0
SFRS [18]	78.2	87.1	90.1	61.0	73.1	76.7	29.8	41.3	48.2	73.9	83.2	85.9	85.3	90.6	91.9
CosPlace [6]	89.3	93.2	95.2	69.9	81.6	85.9	48.6	63.7	71.4	85.8	91.9	94.3	89.0	94.3	95.2
MixVPR [2]	97.0	<u>98.2</u>	<u>98.6</u>	<u>84.0</u>	<u>92.6</u>	<u>94.5</u>	<u>62.1</u>	78.9	83.0	<u>92.0</u>	<u>96.6</u>	97.8	<u>96.2</u>	98.2	<u>99.0</u>
Ours w/o Reranking	92.9	97.6	98.1	71.4	83.0	87.2	29.7	45.1	54.4	85.5	92.9	95.3	96.1	<u>98.3</u>	98.7
Ours	<u>96.2</u>	98.999.1	89.1	94.495.1	67.8	<u>72.9</u>	<u>73.9</u>	94.2	97.2	<u>97.6</u>	97.498.999.1				

on both the MSLS-val and Svox Snow datasets. This demonstrates that filtering patch descriptors can reduce the storage cost of feature descriptors, while maintaining performance.

Attention Module: We perform ablation studies to verify the effectiveness of our proposed module. Considering the robustness of the semantic segmentation network, we present the ablation results on the weather change datasets, as shown in Table 5. Global Attention refers to the incorporation of a semantically guided attention module during the global retrieval stage, as detailed in Sect. 3.1. Local Attention pertains to the integration of semantic-aware attention into local feature matching, as discussed in Sect. 3.2. The use of Global Attention during the global retrieval stage enhances the model’s performance, with the most significant improvement observed on the Svox Overcast dataset, showing a 3.3% increase in R@1. The inclusion of Local Attention in the re-ranking stage, along with Global Attention, further boosts the performance. This suggests that

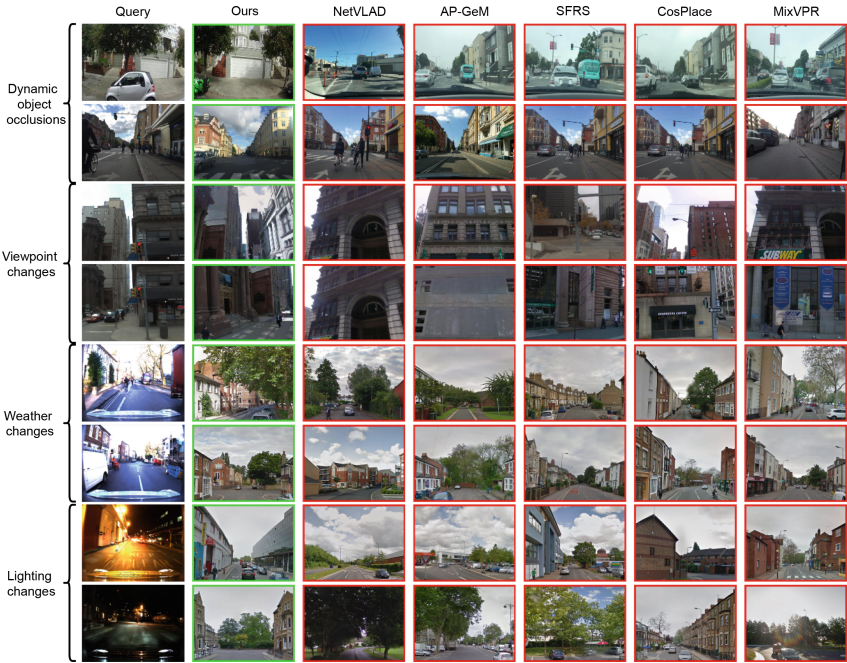


Fig. 5. Qualitative Results. These examples show that our method successfully identifies the correct place while all other methods retrieve incorrect results in different challenging scenarios.

optimizing local matching through attention mechanisms is crucial for refining retrieval results.

Table 5. Ablation study on different model components. Recall@1 is reported.

Method	Global Attention	Local Attention	MSLS-val	Svox	Snow Svox	Sun Svox	Rain Svox	Overcast
Global Retrieval			82.6	91.0	70.6	83.2	92.8	
	✓		84.7	92.9	71.4	85.5	96.1	
Re-ranking			85.1	94.9	88.2	93.4	96.4	
	✓		86.1	96.1	88.8	94.2	96.8	
	✓	✓	87.0	96.2	89.1	94.2	97.4	

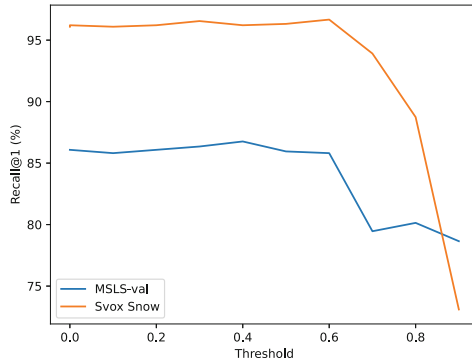


Fig. 6. Ablation study on different thresholds for filtering patch descriptors. Recall@1 is reported.

5 Conclusion

In this paper, we present an innovative hierarchical VPR approach to learn robust global and local features through the integration of semantic information to guide attention implicitly. Our method allows the VPR model to focus on critical information for accurately distinguishing places, without the need to generate and store additional semantic labels at test time. By embedding semantic-driven attention mechanisms into local matching, our approach derives discriminative patch descriptors and prioritizes the nearest neighbor matching of patch pairs with higher attention scores. Our experimental results achieve state-of-the-art performance across multiple benchmark datasets. We envision widespread practical applications for our proposed VPR method in real-world scenarios.

References

1. Ali-bey, A., Chaib-draa, B., Giguère, P.: GSV-Cities: toward appropriate supervised visual place recognition. *Neurocomputing* **513**, 194–203 (2022)
2. Ali-Bey, A., Chaib-Draa, B., Giguere, P.: MixVPR: feature mixing for visual place recognition. In: 2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), pp. 2997–3006. IEEE, Waikoloa, HI, USA (2023)
3. Arandjelovic, A.T.R., Okutomi, J.S.M., Pajdla, T.: 24/7 place recognition by view synthesis. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1808–1817 (2015)
4. Arandjelovic, R., Gronat, P., Torii, A., Pajdla, T., Sivic, J.: NetVLAD: CNN architecture for weakly supervised place recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5297–5307 (2016)
5. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-up robust features (SURF). *Comput. Vis. Image Underst.* **110**(3), 346–359 (2008)
6. Berton, G., Masone, C., Caputo, B.: Rethinking Visual Geo-localization for Large-Scale Applications. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4868–4878. IEEE, New Orleans, LA, USA (2022)

7. Berton, G.M., Paolicelli, V., Masone, C., Caputo, B.: Adaptive-attentive geolocalization from few queries: a hybrid approach. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 2918–2927 (2021)
8. Cai, Y., Zhao, J., Cui, J., Zhang, F., Feng, T., Ye, C.: Patch-NetVLAD+: learned patch descriptor and weighted matching strategy for place recognition. In: 2022 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), pp. 1–8 (2022)
9. Cheng, B., Misra, I., Schwing, A.G., Kirillov, A., Girdhar, R.: Masked-attention mask transformer for universal image segmentation. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1280–1289. IEEE, New Orleans, LA, USA (2022)
10. Choi, S.M., Lee, S.I., Lee, J.Y., Kweon, I.S.: Semantic-guided de-attention with sharpened triplet marginal loss for visual place recognition. *Pattern Recogn.* **141**, 109645 (2023)
11. Cordts, M., et al.: The cityscapes dataset for semantic urban scene understanding. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3213–3223. IEEE, Las Vegas, NV, USA (2016)
12. Cummins, M., Newman, P.: FAB-MAP: probabilistic localization and mapping in the space of appearance. *Int. J. Robot. Res.* **27**(6), 647–665 (2008)
13. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). vol. 1, pp. 886–893 vol. 1 (2005)
14. Dosovitskiy, A., et al.: An image is worth 16×16 words: transformers for image recognition at scale. arXiv preprint [arXiv:2010.11929](https://arxiv.org/abs/2010.11929) (2021)
15. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. In: Readings in Computer Vision, pp. 726–740. Morgan Kaufmann, San Francisco (CA) (1987)
16. Garg, S., Suenderhauf, N., Milford, M.: LoST? Appearance-invariant place recognition for opposite viewpoints using visual semantics. In: Robotics: Science and Systems XIV. Robotics: Science and Systems Foundation (2018)
17. Garg, S., Suenderhauf, N., Milford, M.: Semantic-geometric visual place recognition: a new perspective for reconciling opposing views. *Int. J. Robot. Res.* **41**(6), 573–598 (2022)
18. Ge, Y., Wang, H., Zhu, F., Zhao, R., Li, H.: Self-supervising fine-grained region similarities for large-scale image localization. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12349, pp. 369–386. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58548-8_22
19. Hausler, S., Garg, S., Xu, M., Milford, M., Fischer, T.: Patch-NetVLAD: multi-scale fusion of locally-global descriptors for place recognition. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 14136–14147. IEEE, Nashville, TN, USA (2021)
20. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
21. Jégou, H., Douze, M., Schmid, C., Pérez, P.: Aggregating local descriptors into a compact image representation. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 3304–3311 (2010)
22. Keetha, N.V., Milford, M., Garg, S.: A hierarchical dual model of environment- and place-specific utility for visual place recognition. *IEEE Robot. Autom. Lett.* **6**(4), 6969–6976 (2021)

23. Kim, H.J., Dunn, E., Frahm, J.M.: Learned contextual feature reweighting for image geo-localization. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3251–3260. IEEE, Honolulu, HI (2017)
24. Larsson, M., Stenborg, E., Toft, C., Hammarstrand, L., Sattler, T., Kahl, F.: Fine-grained segmentation networks: self-supervised segmentation for improved long-term visual localization. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 31–41. IEEE, Seoul, Korea (South) (2019)
25. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision* **60**(2), 91–110 (2004)
26. Maddern, W., Pascoe, G., Linegar, C., Newman, P.: 1 year, 1000 km: the oxford RobotCar dataset. *Int. J. Robot. Res.* **36**(1), 3–15 (2017)
27. Naseer, T., Oliveira, G.L., Brox, T., Burgard, W.: Semantics-aware visual localization under challenging perceptual conditions. In: 2017 IEEE International Conference on Robotics and Automation (ICRA), pp. 2614–2620. IEEE, Singapore (2017)
28. Oliva, A., Torralba, A.: Building the gist of a scene: the role of global image features in recognition. In: Martinez-Conde, S., Macknik, S.L., Martinez, L.M., Alonso, J.M., Tse, P.U. (eds.) *Progress in Brain Research, Visual Perception*, vol. 155, pp. 23–36. Elsevier (2006)
29. Paolicelli, V., Tavera, A., Masone, C., Berton, G., Caputo, B.: Learning semantics for visual place recognition through multi-scale attention. In: Sclaroff, S., Distanto, C., Leo, M., Farinella, G.M., Tombari, F. (eds.) *Image Analysis and Processing – ICIAP 2022*, pp. 454–466. Springer International Publishing, Cham (2022). https://doi.org/10.1007/978-3-031-06430-2_38
30. Peng, G., Yue, Y., Zhang, J., Wu, Z., Tang, X., Wang, D.: Semantic reinforced attention learning for visual place recognition. In: 2021 IEEE International Conference on Robotics and Automation (ICRA), pp. 13415–13422 (2021)
31. Revaud, J., Almazan, J., Rezende, R., Souza, C.D.: Learning with average precision: training image retrieval with a listwise loss. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 5106–5115. IEEE, Seoul, Korea (South) (2019)
32. Shen, Y., Zhou, S., Fu, J., Wang, R., Chen, S., Zheng, N.: StructVPR: distill structural knowledge with weighting samples for visual place recognition. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1217–11226 (2023)
33. Sivic, Zisserman: Video Google: a text retrieval approach to object matching in videos. In: *Proceedings Ninth IEEE International Conference on Computer Vision*, vol. 2, pp. 1470–1477 (2003)
34. Torii, A., Sivic, J., Pajdla, T., Okutomi, M.: Visual place recognition with repetitive structures. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition, pp. 883–890. IEEE, Portland, OR, USA (2013)
35. Wang, R., Shen, Y., Zuo, W., Zhou, S., Zheng, N.: TransVPR: transformer-based place recognition with multi-level attention aggregation. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 13638–13647. IEEE, New Orleans, LA, USA (2022)
36. Wang, X., Han, X., Huang, W., Dong, D., Scott, M.R.: Multi-similarity loss with general pair weighting for deep metric learning. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5017–5025. IEEE, Long Beach, CA, USA (2019)

37. Warburg, F., Hauberg, S., Lopez-Antequera, M., Gargallo, P., Kuang, Y., Civera, J.: Mapillary street-level sequences: a dataset for lifelong place recognition. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2623–2632. IEEE, Seattle, WA, USA (2020)
38. Xue, F., Budvytis, I., Cipolla, R.: SFD2: semantic-guided feature detection and description. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5206–5216. IEEE, Vancouver, BC, Canada (2023)



Dense Reconstruction and Localization in Scenes with Glass Surfaces Based on ORB-SLAM2

Zeyuan Chen^{1(✉)}, Ziquan Wang¹, Qiang Gao², Masahiko Mikawa³,
and Makoto Fujisawa³

¹ Graduate School of Comprehensive Human Sciences, University of Tsukuba,
Tsukuba, Japan

chen.zeyuan.tkb_gu@u.tsukuba.ac.jp

² School of Artificial Intelligence and Automation, Huazhong University of Science
and Technology, Wuhan, China

³ Institute of Library, Information and Media Science, University of Tsukuba,
Tsukuba, Japan

Abstract. In recent years, Visual Simultaneous Localization and Mapping (SLAM) research has made significant strides, particularly in the domain of RGB-D SLAM. However, the prevalent presence of glass surfaces poses a substantial challenge, impeding the effective performance of RGB-D SLAM in modern indoor environments. This challenge stems from the transparent, refractive, and reflective properties of glass surfaces, causing RGB-D cameras to struggle to obtain accurate depth information, consequently negatively impacting on the estimation of camera trajectories and the reconstruction of glass surfaces. In this paper, we propose a new network designed for simultaneous glass surface segmentation and depth estimation called CGSDNet-Depth. Employing a novel Context Guided Depth Decoder (CGDD), CGSDNet-Depth generates depth information guided by the contextual information of glass surfaces. Subsequently, based on ORB-SLAM2, we introduce a new method named ORB-SLAM2-GSD (Glass Surface Detection) that utilizes the segmentation and depth estimation results from CGSDNet-Depth to alleviate the adverse effects of glass surfaces on camera trajectory estimation and dense reconstruction. Additionally, we construct the first RGB-D dataset for glass surface scenes, comprising 8 image sequences, called GS RGB-D. Extensive experiments demonstrate that our method outperforms other State-of-the-Art (SOTA) methods in glass surface segmentation and improves ORB-SLAM2 performance in glass surface scenes. Code and dataset: <https://github.com/CZYQiYueShang/DRL-GSS>.

Keywords: Dense reconstruction and localization · Glass surface segmentation · Dataset for RGB-D SLAM

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-78113-1_26.

1 Introduction

Visual SLAM, utilizing its advantages of low cost and small size with cameras as input sensors, has found extensive applications in research fields such as autonomous driving. Among Visual SLAM, RGB-D SLAM utilizes RGB-D cameras as input sensors, allowing for direct depth information acquisition. This sets RGB-D SLAM apart from traditional monocular and stereo SLAM, making it particularly proficient in reconstructing dense point cloud maps and reducing scale drift, achieving significant progress over the past decade [1]. Numerous notable RGB-D SLAM methods [2–5] have demonstrated commendable performance on benchmark datasets like TUM RGB-D [6]. However, these methods are difficult to acquire exceptional performance in modern indoor scenes, mainly due to the prevalence of large glass surfaces such as glass doors, walls, and windows. The transparent, refractive, and reflective properties of these glass surfaces pose challenges for RGB-D cameras in accurately obtaining their depth and the depth of objects behind them as Fig. 1 (red box), introducing considerable noise into the dense reconstruction and camera trajectory estimation of RGB-D SLAM.

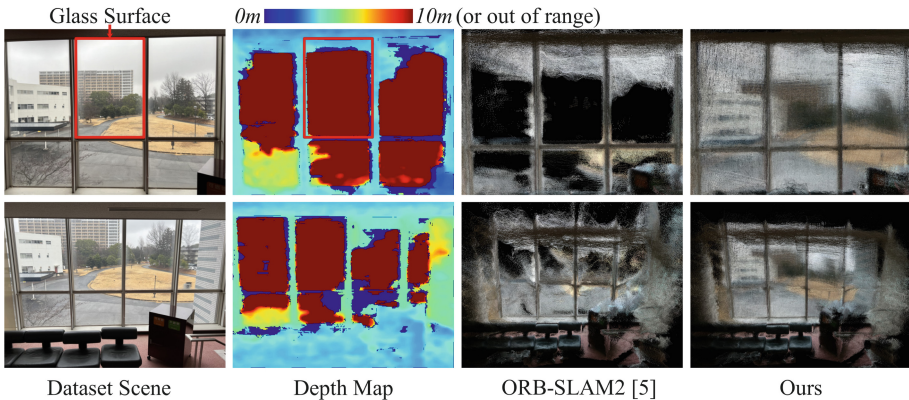


Fig. 1. Comparison between ORB-SLAM2 [5] and our method on the dense reconstruction results of *corridor_chair_day* sequence from our GS RGB-D dataset. The first column shows the scene of this sequence. The second column shows the visualized results of the depth maps captured by the Intel RealSense D435 depth camera. It is clear that the depth camera can not correctly output the depth values of glass surfaces when they act as obstacles. The third column displays the reconstruction results from ORB-SLAM2, which completely fails to reconstruct the glass surfaces. The fourth column presents the reconstruction results from our ORB-SLAM2-GSD, successfully reconstructing the glass surfaces with its mapped external scenery. (Color figure online)

In Fig. 1, we present the dense reconstruction results of ORB-SLAM2 [5] in a scene with a significant number of glass surfaces. It is evident that, due to the limitation of RGB-D cameras in measuring the depth of glass surfaces as obstacles, ORB-SLAM2 fails in reconstructing (map point generation) the glass

surfaces themselves. Furthermore, in this scene, objects positioned behind the glass surfaces are at a distance from the camera that exceeds the depth range measured by the RGB-D camera. Consequently, there is also a complete failure in reconstructing objects positioned behind the glass surfaces. These reconstruction failures can introduce erroneous map point information to ORB-SLAM2, further significantly impacting its accuracy in estimating camera trajectories.

The objective of this paper is to enhance the camera trajectory estimation and dense reconstruction performance of the current RGB-D SLAM in glass surface scenes. To achieve the above objective, we require precise positional information and depth information about the glass surfaces from input RGB images. However, existing State-of-the-Art (SOTA) methods from related fields face challenges in simultaneously achieving precise segmentation and depth estimation for glass surfaces, as shown in Fig. 2. To address this issue, we propose the CGSDNet-Depth network based on a glass surface segmentation network called CGSDNet [7]. It utilizes a novel Context Guided Depth Decoder (CGDD) to generate depth features guided by the contextual features of the glass surfaces extracted by CGSDNet. Therefore, CGSDNet-Depth can simultaneously output high-quality mask images and depth maps of the glass surfaces. As illustrated in Fig. 2, our proposed CGSDNet-Depth demonstrates excellent segmentation and depth estimation results for these challenging images.

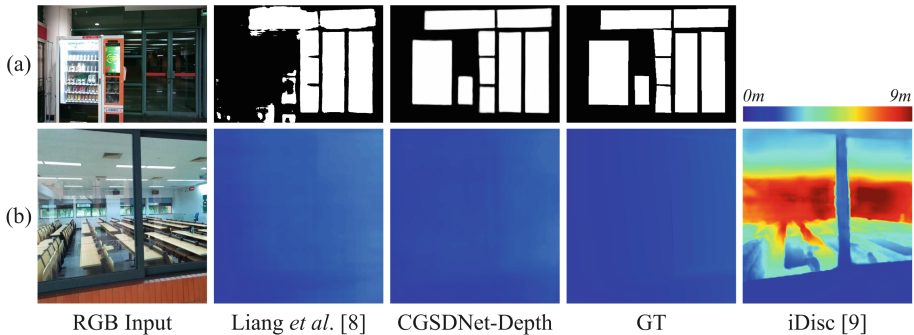


Fig. 2. Our CGSDNet-Depth compares with Liang *et al.*'s method [8] on the benchmark dataset GW-Depth [8]. CGSDNet-Depth and Liang *et al.*'s method are trained on GW-Depth, iDisc [9] is trained on the NYU Depth V2 [10]. In (a), it is evident that Liang *et al.*'s method shows noticeable defects in the segmentation of glass surfaces, particularly at the boundaries of glass surfaces. In (b), our CGSDNet-Depth and Liang *et al.*'s method correctly output the depth of glass surfaces instead of the depth of the object behind glass surfaces. However, iDisc trained on traditional depth estimation datasets fails to accurately acquire the depth of glass surfaces as obstacles.

Furthermore, we propose a new method based on ORB-SLAM2, which performs well in glass surface scenes, named ORB-SLAM2-GSD (Glass Surface Detection). Specifically, We present a Feature Point Filtering algorithm that

employs the results from CGSDNet-Depth. This algorithm filters ORB feature [11] points within the glass surface region, removing those feature points that would introduce significant noise to the camera trajectory estimation. We then develop a Glass Surface Reconstruction algorithm, also utilizing the output from CGSDNet-Depth, to achieve dense reconstruction of the glass surfaces. The reconstructed appearance of the glass surface is consistent with the pattern mapped by the glass surface in the input RGB image, as shown in Fig. 1.

Finally, to verify our proposed ORB-SLAM2-GSD and due to the lack of a dedicated RGB-D dataset for glass surface scenes, we construct the first RGB-D dataset specifically for glass surface scenes, named GS (Glass Surface) RGB-D. Our proposed GS RGB-D dataset comprises a total of 8 image sequences, each of which contains a minimum of 4 distinct independent glass surfaces.

In summary, the main contributions of our paper can be outlined as follows:

- We propose a novel Context Guided Depth Decoder (CGDD) and CGSDNet-Depth network, capable of simultaneously achieving high-quality segmentation and depth estimation for glass surfaces.
- We present a lightweight GSD (Glass Surface Detection) module, which improves ORB-SLAM2’s performance of dense reconstruction and camera trajectory estimation in glass surface scenes.
- We construct the first specialized dataset for RGB-D SLAM in scenes with abundant glass surfaces, called GS RGB-D.

2 Related Work

2.1 RGB-D SLAM

Traditional Visual SLAM methods typically utilize monocular or stereo images as input, which require substantial computation to obtain depth values. In contrast, RGB-D SLAM systems employ depth maps as input, allowing for the direct acquisition of depth information for map points and facilitating the construction of dense point cloud maps. Newcombe *et al.* [2] introduce KinectFusion, which preprocesses the original depth maps using a bilateral filter and constructs dense vertex and normal map pyramids for pose estimation. DVO-SLAM [3] combines dense visual odometry and Pose SLAM, incorporating global optimization to minimize accumulated drift. ElasticFusion [4] models variable scenes by categorizing surfels into active and non-active groups and fusing depth maps. ORB-SLAM2 [5] utilizes ORB features [11] for tracking, mapping, relocalization, and loop closing. It also employs bundle adjustment for pose optimization.

However, the aforementioned RGB-D SLAM methods encounter challenges in operating effectively in scenes with abundant glass surfaces like glass walls.

2.2 SLAM in Glass Surface Scenes

Linus *et al.* [12] have demonstrated that SLAM SOTAs still exhibit poor performance in modern indoor scenes due to the transparent and reflective properties

of glass surfaces. To improve the effectiveness of SLAM methods in glass surface scenes, various glass detection methods have been introduced and integrated with SLAM algorithms. Among them, both [13, 14] utilize Laser Range Finders to identify the reflective characteristics of glass surfaces to detect them. However, the mentioned glass surface detection methods are specifically tailored for Lidar SLAM, and unsuitable for RGB-D SLAM.

Ongoing research is improving the performance of RGB-D SLAM in scenes with transparent objects. Zhu *et al.* [15] propose a transparent object segmentation network to remove transparent objects, eliminating their negative impact on pose estimation. After calculating the camera pose, a visual hull-based method is employed to reconstruct the removed transparent object. Additionally, they introduce a specifically designed RGB-D dataset for scenes with transparent objects, called Trans-SLAM. However, Zhu *et al.*'s work primarily focuses on small transparent objects like glass bottles and is not suitable for reconstructing large-scale glass surfaces.

2.3 Glass Surface Segmentation

To address the issue of detecting glass surfaces from RGB images, Mei *et al.* [16] first propose a network specifically tailored for glass surface segmentation. Their method incorporates a Large-field Contextual Feature Integration module to extract rich contextual features. Lin *et al.* [17] utilize the reflection features of glass surfaces with their Reflection-based Refinement Module to aid in glass surface segmentation. We [7] have introduced a Cascade Atrous Pooling module and a Cascaded Network Architecture to aggregate denser large-field contextual features, reducing the generation of holes in glass surface segmentation results.

Nevertheless, simply knowing the position of glass surfaces in RGB images is insufficient for their reconstruction in the map. Accurate depth information for the glass surfaces, particularly when they act as obstacles, is also necessary.

2.4 Monocular Depth Estimation for Glass Surfaces

In recent years, there has been a rise in the development of high-quality monocular depth estimation methods, exemplified by NeWCRFs [18] and iDisc [9]. Although these methods demonstrate satisfactory performance on datasets like NYU Depth V2 [10], they encounter difficulties in accurately estimating the depth values of the glass surfaces when they act as obstacles. To address this issue, Liang *et al.* [8] introduce the first RGB-D dataset and monocular depth estimation method tailored for glass surfaces. Specifically, they propose a depth interpolation pipeline to effectively generate precise depth annotations for glass surfaces. In addition, they present a dual-context approach that utilizes both the structural context of glass surface bounding line segments and the reflective context of the glass to estimate its depth.

Despite Liang *et al.* have integrated glass surface segmentation as an auxiliary task, the segmentation performance of their method is not satisfactory.

This limitation may result in considerable errors when their method is used for detecting the position of glass surfaces in RGB-D SLAM.

3 CGSDNet-Depth

3.1 Overview

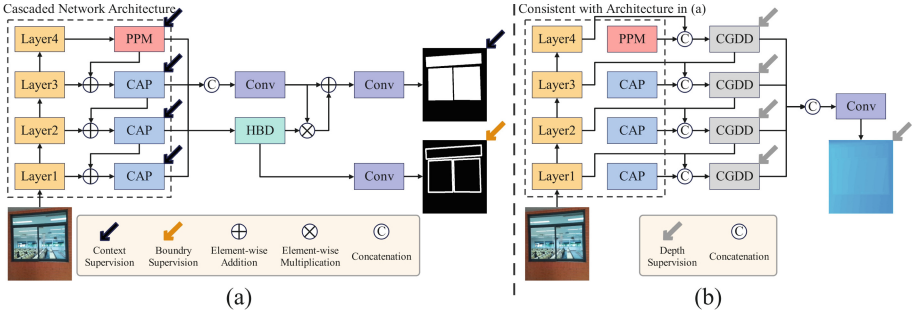


Fig. 3. The pipeline of our CGSDNet-Depth. (a) The pipeline for glass surface segmentation part in CGSDNet-Depth is consistent with [7]. (b) The pipeline for glass surface depth estimation part in CGSDNet-Depth. The CGDDs (gray blocks) extract depth features from the backbone features, contextual features of glass surfaces output by the PPM [19] and CAP [7] modules, and higher-level depth features (if available).

Inspired by Liang *et al.*'s work [8], we noticed that a single network can simultaneously generate mask images and depth maps of glass surfaces. However, we noted that Liang *et al.*'s network primarily focuses on depth estimation, leading to suboptimal glass surface segmentation results, as shown in Fig. 2(a). Additionally, we observed that the depth values of glass surfaces when acting as obstacles significantly differ from the depth values of objects behind them or their reflections, residing distinctly on a unified plane. This observation inspired us to employ more precise glass surface segmentation results and corresponding contextual features to guide the depth features, aiming to achieve smoother depth maps of glass surfaces. Therefore, we use the CGSDNet [7] with robust glass surface segmentation capability as a foundation and introduce a novel Context Guided Depth Decoder (CGDD), which utilizes the contextual features output by CGSDNet to guide the generation of depth maps for glass surfaces.

Figure 3 illustrates the pipeline of our CGSDNet-Depth. Initially, an RGB image is input into the Cascaded Network Architecture, which is identical to CGSDNet, as shown in Fig. 3(a). Subsequently, we concatenate the contextual features of glass surfaces output by the Pyramid Pooling Module (PPM) [19] and Cascade Atrous Pooling (CAP) [7] modules and corresponding backbone features, importing them into our CGDDs, as depicted in Fig. 3(b). The depth features output by CGDDs are further concatenated with lower-level contextual

features and backbone features, and then input into the lower-level CGDDs to guide the output of lower-level depth features. Finally, we fuse the depth features from four levels to generate the final depth map for glass surfaces.

3.2 Context Guided Depth Decoder

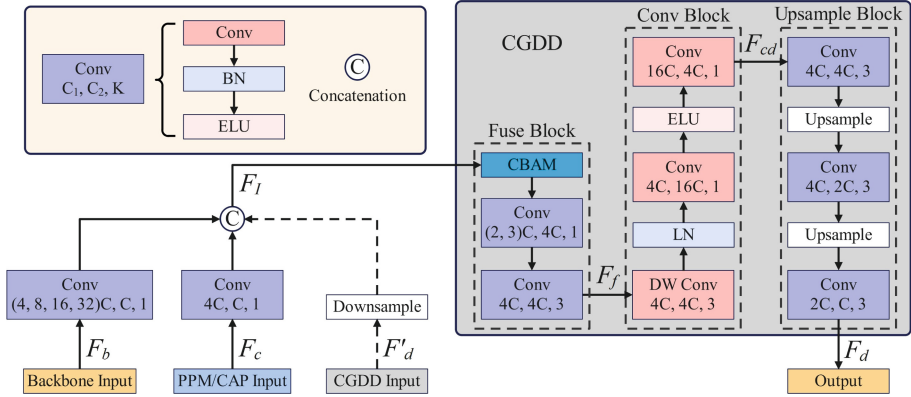


Fig. 4. The architecture of our CGDD. Conv C_1, C_2, K (purple blocks) represent a convolutional layer with an input channel number of C_1 , an output channel number of C_2 , and a kernel size of $K \times K$, accompanied by a Batch Normalization (BN) layer and an Exponential Linear Unit (ELU) layer. Conv C_1, C_2, K (red blocks) denote the corresponding convolutional layer alone. CBAM (blue block) signifies a Convolutional Block Attention Module (CBAM) [20]. The dashed line associated with CGDD Input indicates the potential absence of input from higher-level CGDD depth features. (Color figure online)

To enhance the overall smoothness of depth estimation results for the glass surfaces, we utilize the contextual features of glass surfaces to guide the generation of depth features. As shown in Fig. 4, after modifying the channel numbers of the contextual features of glass surfaces F_c extracted by the PPM or CAP module and the corresponding level of backbone features F_b , we concatenate them with the higher-level depth features F'_d (if available). Subsequently, these concatenated features F_I are input into our proposed CGDD.

Within the CGDD, we initially employ a Convolutional Block Attention Module (CBAM) [20] to enhance the channel-wise attention across F_I , which are fused into F_f through two convolutional layers. F_f is then fed into a Conv Block similar to the ConvNeXt Block [21], which retains the spatial information of the contextual features of glass surfaces through depthwise convolution, and generates deep features F_{cd} containing contextual information by two 1×1 convolutions that expand the feature dimension. F_{cd} is further input into the Upsampling Block, which utilizes two upsampling layers and convolutional layers

with gradually reduced output channel numbers, to restore the spatial details of depth features. Finally, high-resolution depth features F_d are output by CGDD.

3.3 Loss Function

During the training process, we utilize the same loss functions as [7] to optimize the glass surface segmentation part, Fig. 3(a), of CGSDNet-Depth. Including L_c for supervising the output of PPM and CAP modules, L_b for supervising the output of the HBD module, and L_{fc} for supervising the final output mask image. To optimize the glass surface depth estimation part, Fig. 3(b), of CGSDNet-Depth, we incorporate the scale-invariant (SI) loss [22] l_{si} , as the following (1):

$$l_{si}(P) = \frac{1}{T} \sum_i d_i^2 - \frac{\lambda}{T^2} \left(\sum_i d_i \right)^2 \quad (1)$$

where T represents the number of pixels with valid depth values in the ground truth (GT) for depth estimation. $d_i = \log(g_i) - \log(p_i)$, with g_i as the depth value of a pixel in the GT, and p_i as the depth value of a pixel in the predicted map. Referring [23], we set λ to 0.85 and employ the following (2) as loss functions to supervise the outputs of CGDDs (L_d) and the final output depth map (L_{fd}):

$$L_d = \sum_{j=1}^N \alpha \sqrt{l_{si}(P_j)}, \quad L_{fd} = \alpha \sqrt{l_{si}(P_{fd})} \quad (2)$$

where N denotes the total number of CGDDs, α is set to 10, P_j represents the depth map output by a specific CGDD, and P_{fd} represents the final output depth map.

Finally, the overall loss function is expressed as (3):

$$Loss = w_c L_c + w_b L_b + w_d L_d + w_{fd} L_{fd} + w_{fc} L_{fc} \quad (3)$$

where w_c , w_b , w_d , w_{fd} and w_{fc} represent the weight parameters for L_c , L_b , L_d , L_{fd} and L_{fc} , respectively.

4 ORB-SLAM2-GSD

Our method is built upon the RGB-D system of ORB-SLAM2 [5]. In addition to the inputs of original RGB images and depth maps obtained from the RGB-D camera, we also use CGSDNet-Depth for data preprocessing, incorporating mask images and depth maps of the glass surfaces as inputs. The newly introduced glass surface mask image and depth map inputs throughout both the Tracking thread of ORB-SLAM2 and our Dense Reconstruction thread. These inputs serve two main purposes: (1) Feature Point Filtering (FPF): Filtering ORB feature [11] points within the glass surface region. (2) Glass Surface Reconstruction (GSR): Completing the dense reconstruction of the glass surfaces.

Algorithm 1. Feature Point Filtering

Input: p - coordinate of the feature point, $mDepth$ - camera depth map, $mDepthGS$ - depth map of glass surfaces, $depthAvg$ - average depth of non-glass pixels.

Output: d - depth value of the feature point.

```

1:  $depth \leftarrow mDepth.at(p)$  ▷ Get coordinate  $p$ 's camera depth value
2:  $depthGS \leftarrow mDepthGS.at(p)$  ▷ Get coordinate  $p$ 's predicted depth value of glass surfaces
3:  $depthGap \leftarrow depth/depthGS$ 
4: if  $(1 - FPF DGR) < depthGap < (1 + FPF DGR)$  or  $depth < depthAvg * FPF MaxDW$  then
5:    $d \leftarrow depth$  ▷ Keep this feature point
6: else
7:    $d \leftarrow 0$  ▷ Logically delete this feature point
8: end if
9: return  $d$ 

```

4.1 Feature Point Filtering

To eliminate the significant noise introduced by glass surfaces in camera trajectory estimation, we need to filter out ORB feature points with inaccurate depth data within the glass surface region. Initially, during the extraction of ORB feature points from the input RGB image, we utilize the mask image to determine whether the feature points are within the glass surface region. When the mask value of a particular feature point is greater than our $FPFMinMV$ (FPF Min Mask Value, the greater the mask value, the more likely it is glass) threshold parameter, we identify the feature point as being within the glass surface region and assign its $class_id$ to 0. When calculating the depth value of the feature point, if its $class_id$ is not equal to 0, we directly use the camera depth value (the depth value from the depth map captured by the camera). If the $class_id$ is 0, we employ Algorithm 1 to compute the depth value for that feature point.

The primary objective of Algorithm 1 is to retain those feature points within the glass surface region that still have accurate camera depth values. Specifically, we keep feature points whose depth values are close to the glass surface because their camera depth values are not significantly affected by refraction. Feature points near the camera are also retained, as they are likely to be located between the glass surfaces and the camera, thus still having accurate camera depth values. As shown in lines 3 and 4 of Algorithm 1, we determine whether the feature point is close to the glass surface by checking if the ratio of its camera depth value to the predicted depth value of the glass surface falls within the range controlled by our $FPFDGR$ (Depth Gap Range) parameter. Additionally, we assess whether the feature point is close to the camera by checking if its depth value is less than the product of the average depth of non-glass pixels (mask value less than our $MaxMV$ threshold parameter) and our $FPFMaxDW$ (Max Depth Weight) parameter. Feature points that do not meet these conditions are removed.

4.2 Glass Surface Reconstruction

We use the camera poses of keyframes from ORB-SLAM2, with the corresponding RGB images and camera depth maps, to generate point clouds and perform dense reconstruction of each non-glass pixel. For achieving dense reconstruction of glass surfaces, generating point clouds at the position of the glass surfaces,

Algorithm 2 Border Pixel Depth Calculation

Input: p - coordinate of the border pixel, $mDepth$ - camera depth map, $mDepthGS$ - depth map of glass surfaces, $mMask$ - mask image, $depthAvg$ - average depth of non-glass pixels.

Output: d - depth value of the border pixel.

```

1: for all  $p'$  that  $p' \in$  All pixels within a range of  $GSRPR$  (Pixel Range) pixels around  $p$ . do
2:    $mask \leftarrow mMask.at(p')$   $\triangleright$  Get the mask value of pixel  $p'$ 
3:    $depth \leftarrow mDepth.at(p')$   $\triangleright$  Get the camera depth value of pixel  $p'$ 
4:    $depthGS \leftarrow mDepthGS.at(p')$   $\triangleright$  Get  $p'$ 's predicted depth value of glass surfaces
5:    $depthGap \leftarrow depth/depthGS$ 
6:   if  $mask < MaxMV$  then  $\triangleright p'$  is not a glass pixel
7:      $d' \leftarrow depth$   $\triangleright$  Take the camera depth value as  $p'$ 's depth value
8:   else if  $mask < GSRMinMV$  then  $\triangleright p'$  could be a glass pixel, further determine the gap
   between the camera depth value and the predicted depth value of glass surface
9:     if  $(1 - GSRDGR) < depthGap < (1 + GSRDGR)$  then  $\triangleright$  The gap is small
10:       $d' \leftarrow depth$ 
11:     else  $\triangleright$  The gap is large
12:       $d' \leftarrow depthGS$   $\triangleright$  Take the predicted depth value as  $p'$ 's depth value
13:     end if
14:   else  $\triangleright p'$  is highly likely to be a glass pixel
15:      $d' \leftarrow depthGS$ 
16:   end if
17:   if  $0 < d' < depthAvg * GSRMaxDW$  then  $\triangleright$  Discard  $p'$  with larger depth values
18:      $depthSum \leftarrow depthSum + d'$ 
19:      $n \leftarrow n + 1$ 
20:   end if
21: end for
22: if  $n \neq 0$  then  $\triangleright$  Any  $p'$  meets the criteria of line 17
23:    $d \leftarrow depthSum/n$ 
24: else  $\triangleright$  No  $p'$  meets the criteria of line 17
25:    $d \leftarrow mDepthGS.at(p)$ 
26: end if
27: return  $d$ 

```

we need to obtain the correct depth values for them. Considering that the camera depth values are more accurate compared to the depth values predicted by CGSDNet-Depth, we mainly utilize the camera depth values of the opaque borders around the glass surfaces, with the predicted depth values as auxiliary data, to calculate the depth values of glass surfaces. For a specific glass pixel (mask value greater than our $GSRMinMV$ threshold parameter), we first search for the four nearest non-glass pixels in the up, down, left, and right directions relative to the glass pixel on the image plane, named border pixels. Subsequently, we employ the depth values of these four border pixels and their surrounding pixels to calculate their final depth values, as illustrated in Algorithm 2.

After obtaining the depth values of the four border pixels, we calculate a weighted average based on their distances from the original glass pixel to obtain two average depth values in the up-down and left-right directions. If the gap between these two depth values is relatively small (similar to line 9 of Algorithm 2, adjusted through our $GSRDGRXY$ parameter), the average of them is taken as the final depth value for the glass pixel, and the corresponding point cloud is generated. Otherwise, the point cloud corresponding to that glass pixel is discarded. Additionally, we discard the point clouds corresponding to pixels with depth values that are greater than the product of the average depth of non-glass pixels and our $GSRMaxDW$ parameter, similar to line 17 in Algorithm 2. All

the point clouds are colored using the RGB values of the corresponding pixels in the original RGB image.

5 GS RGB-D Dataset

5.1 Overview

To evaluate RGB-D SLAM methods, various indoor scene datasets like TUM RGB-D [6] are widely available. However, most existing indoor datasets lack scenes where glass surfaces are the main element. To assess the effectiveness of our proposed ORB-SLAM2-GSD, we construct a new dataset that specifically focuses on indoor scenes with abundant glass surfaces, named GS (Glass Surface) RGB-D. We capture datasets in several different glass surface scenes, including 6 daytime scenes and 2 nighttime scenes. Figure 5 illustrates an example of the captured scene and the camera ground truth (GT) trajectories in our dataset.

Each sequence in our GS RGB-D dataset consists of RGB images, depth maps, and camera GT trajectory. The RGB images and depth maps are captured using an Intel RealSense D435 depth camera with a resolution of 640×480 pixels, while the camera GT trajectory is obtained using a Niryo One robot arm.

5.2 Camera GT Trajectory Acquisition

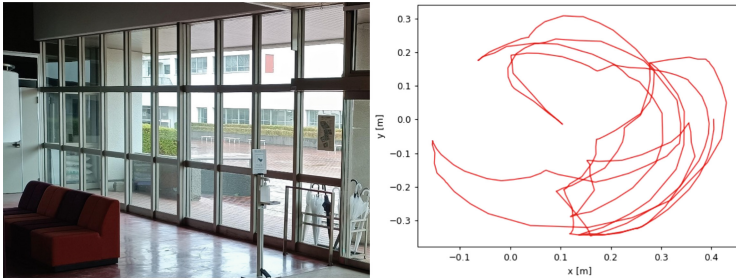


Fig. 5. Example of sequence in our GS RGB-D dataset. Left: The glass surface scene we captured. Right: The camera ground truth trajectories.

Our camera-robot arm system is presented in Fig. 6. In our dataset, the camera GT trajectory is a sequence of quaternion-format transformation matrix T_{camera}^{base} , representing the transformation of the RGB camera relative to the base of the robot arm. The calculation of T_{camera}^{base} is formulated in (4):

$$T_{camera}^{base} = T_{end}^{base} \cdot T_{camera}^{end} \quad (4)$$

where T_{end}^{base} is calculated utilizing forward kinematics, and T_{camera}^{end} can be defined based on the dimensions of the workpiece used to connect the camera and the robot arm's end and the camera as depicted in Fig. 6(b).



Fig. 6. Our camera-robot arm system. Left: The definition of coordinate systems for the robot arm’s base (green) and end (red). Right: The relationship between the coordinate system of the robot arm’s end (red) and the camera (yellow). (Color figure online)

During the dataset-capturing process, the robot arm is guided to follow the pre-obtained joint sequence while simultaneously capturing real-time camera images and recording the current pose of the camera. Throughout this entire process, the base of the robot arm remains stationary.

6 Experiment

6.1 Implementation Details

CGSDNet-Depth. We initially pre-trained the segmentation part of CGSDNet-Depth on 3 glass surface datasets: GDD [16], GSD [17], and HSO [24]. Subsequently, we train the whole CGSDNet-Depth network on the benchmark dataset GW-Depth [8] through multi-task joint training. The input image size is adjusted to 416×416 pixels, and the batch size is set to 6. Data augmentation includes horizontal flipping, vertical flipping, random cropping, and color jittering, consistent with [8]. The parameters of the backbone network are initialized using the pre-trained ConvNeXt-B [21], while other parameters are initialized with the default random initialization in PyTorch [25]. The weight parameters of the loss function w_c , w_b , w_d , w_{fd} and w_{fc} are empirically set to 1, 3, 1, 2, and 3, respectively. We utilize the same optimizer and learning strategy as in [7]. The model is trained for 200 epochs on an NVIDIA GTX 3090 Ti graphics card.

ORB-SLAM2-GSD. During testing, we employ a consistent set of parameter settings. For the mask value threshold parameters, $MaxMV$, $FPFM_{in}MV$, and $GSRMinMV$ are set to 5, 250, and 127.5, respectively. Regarding Feature Point Filtering, we set $FPFDGR$ and $FPFMaxDW$ to 0.3 and 1, respectively. For Glass Surface Reconstruction, $GSRPR$, $GSRDGR$, $GSRDGRXY$, and $GSRMaxDW$ are set to 2, 0.1, 0.1, and 2, respectively.

Table 1. Depth estimation and segmentation comparison result on the GW-Depth [8] test set. “*” indicates that the quantitative values for this method are sourced from [8]. Traditional depth estimation methods lack glass surface segmentation outputs, and the corresponding values are replaced by “-”. The best results are highlighted in **bold**.

Training set	Method	$A_{1.25} \uparrow$	$A_{1.25^2} \uparrow$	$A_{1.25^3} \uparrow$	REL \downarrow	RMS \downarrow	IoU \uparrow	$F_\beta \uparrow$	MAE \downarrow	BER \downarrow
NYU Depth V2 [10]	NeWCRFs [18]	0.364	0.678	0.863	0.460	1.558	-	-	-	-
	iDisc [9]	0.362	0.692	0.879	0.411	1.280	-	-	-	-
GW-Depth [8]	NeWCRFs* [23]	0.851	0.965	0.997	0.123	0.324	-	-	-	-
	Liang <i>et al.</i> [8]	0.902	0.991	0.998	0.097	0.279	92.79	0.965	0.055	8.65
GW-Depth [8]	CGSDNet-Depth	0.894	0.984	0.997	0.109	0.308	95.28	0.977	0.036	5.95

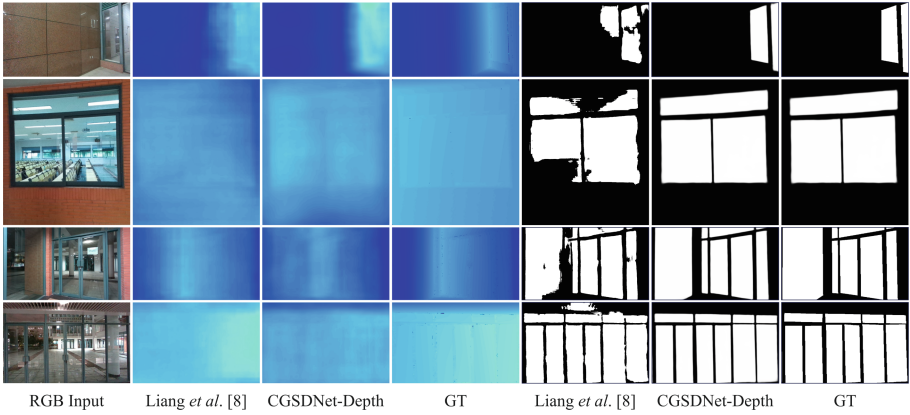


Fig. 7. Visual comparison between our proposed CGSDNet-Depth and Liang *et al.*’s method [8] on the GW-Depth [8] test set. The comparative results include two aspects: glass surface depth estimation and glass surface segmentation.

6.2 Evaluation Metrics

CGSDNet-Depth. To evaluate the glass surface segmentation performance of our proposed CGSDNet-Depth, we employ four widely used metrics in the glass surface segmentation field, including Intersection over Union (IoU), F-measure (F_β), Mean Absolute Error (MAE), and Balance Error Rate (BER). Subsequently, to assess the depth estimation performance of CGSDNet-Depth, we utilize three metrics as in [8], including Accuracy with Threshold (A_{thr}), Average Relative Error (REL), and Root Mean Squared Error (RMS), where A_{thr} is divided into three metrics ($A_{1.25}$, $A_{1.25^2}$, $A_{1.25^3}$) by setting different thresholds.

ORB-SLAM2-GSD. We utilize the widely used Absolute Trajectory Error (ATE) proposed in [6] to evaluate the performance of camera trajectory estimation. Additionally, we compare the dense reconstruction performance in the glass surface region between our proposed ORB-SLAM2-GSD and ORB-SLAM2 [5] through direct observation of the dense reconstruction results.

6.3 Comparison with the SOTAs

CGSDNet-Depth. We compare our CGSDNet-Depth with 3 SOTAs in related fields. As shown in Table 2, we first evaluate the performance of traditional depth estimation methods (NeWCRCFs [18] and iDisc [9] trained on the NYU Depth V2 [10]) on the GW-Depth [8] test set. It is evident that their one-shot ability for depth estimation on glass surfaces is subpar. However, after fine-tuning on the GW-Depth training set, the performance of NeWCRCFs and Liang *et al.*'s method [8] in glass surface depth estimation improved significantly. Unlike them, we first pre-train the glass surface segmentation part of our CGSDNet-Depth on the GDD, GSD, and HSO datasets, and then fine-tune the entire network on the GW-Depth training set. Table 1 demonstrates that our CGSDNet-Depth achieves depth estimation performance close to Liang *et al.*'s method, while significantly outperforming it in segmentation performance. The qualitative comparison results presented in Fig. 7 further highlight the superiority of our CGSDNet-Depth in glass surface segmentation.

Table 2. Camera trajectory estimation (RMSE (m) of ATE) comparison result on our GS RGB-D dataset and the TUM RGB-D [6] dataset. Each sequence is run 5 times, and the median result is recorded. The best results are highlighted in **bold**.

Dataset	Sequence	ORB-SLAM2 [5]	ours
GS RGB-D	corridor_chair_day	0.059	0.036
GS RGB-D	corridor_chair_night	0.021	0.019
GS RGB-D	corridor_bush_day	0.046	0.044
GS RGB-D	hall_door_day	0.037	0.034
GS RGB-D	hall_sofa_day	0.052	0.050
GS RGB-D	living_room_day	0.058	0.052
GS RGB-D	laboratory_night	0.053	0.049
GS RGB-D	office_day	0.087	0.083
TUM RGB-D [6]	fr2/desk	0.009	0.009

ORB-SLAM2-GSD. We conduct a comparative analysis between our proposed ORB-SLAM2-GSD and ORB-SLAM2 on our GS RGB-D dataset. Due to the unavailability of Transfusion's [15] code, it is not included in the comparison. As depicted in Table 2, our method outperforms ORB-SLAM2 in camera

trajectory estimation across all sequences in our GS RGB-D dataset. Additionally, in an example sequence of the TUM RGB-D [6] dataset, which lacks glass surfaces, our method does not negatively impact ORB-SLAM2. Furthermore, our method exhibits proficiency in the dense reconstruction of glass surfaces, as depicted in Fig. 8. In contrast, ORB-SLAM2 tends to neglect the glass surfaces and performs dense reconstruction of objects situated behind or reflections on them. However, our ORB-SLAM2-GSD effectively reconstructs the glass surfaces in both scenarios and successfully restores the appearance mapped by them.



Fig. 8. Comparison of dense reconstruction for glass surfaces between our proposed ORB-SLAM2-GSD and ORB-SLAM2 [5] on our GS RGB-D dataset.

Table 3. The results of the ablation study on our CGSDNet-Depth. “Pre-trained” refers to pre-training on the segmentation part of our CGSDNet-Depth, while “ConvBlock” denotes the Conv Block within our Context Guided Depth Decoder (CGDD). “✓” indicates the inclusion of the corresponding component, while “-” indicates the absence of the corresponding component. The best results are highlighted in **bold**.

+ConvBlock	+Pre-trained	$A_{1.25} \uparrow$	$A_{1.25^2} \uparrow$	$A_{1.25^3} \uparrow$	REL↓	RMS↓	IoU↑	$F_\beta \uparrow$	MAE↓	BER↓
-	-	0.858	0.983	0.996	0.121	0.359	94.13	0.956	0.044	6.75
✓	-	0.870	0.983	0.996	0.122	0.332	94.01	0.955	0.045	7.11
-	✓	0.878	0.980	0.997	0.110	0.316	94.90	0.962	0.039	6.29
✓	✓	0.894	0.984	0.997	0.109	0.308	95.28	0.977	0.036	5.95

6.4 Discussion and Future Work

CGSDNet-Depth. Due to the lack of pre-training on the NYU Depth V2, our CGSDNet-Depth performs slightly worse than Liang *et al.*'s method in terms of depth estimation. But in Fig. 7, it can be observed that our CGSDNet-Depth is better at distinguishing the depth values of glass surface regions from other regions, qualitatively showing the beneficial impact of more refined contextual features of glass surfaces on depth estimation for glass surfaces. Moreover, the results from the ablation study in Table 3 demonstrate that the glass depth estimation performance significantly improve when using the pre-trained model for glass surface segmentation, compared to not using the pre-trained model. This further proves that stronger glass surface segmentation capability can enhance depth estimation for glass surfaces. Table 3 also validates the effectiveness of the Conv Block in our Context Guided Depth Decoder (CGDD).

However, owing to the small data volume (only 1018 images in the training set) and repetitive scenes in the GW-Depth dataset, both our CGSDNet-Depth and Liang *et al.*'s method do not perform so well in real-world scenes, as Fig. 9. In the future, we plan to create a larger dataset for glass surface depth estimation and design a network that can be pre-trained on both glass surface segmentation and depth estimation datasets.

ORB-SLAM2-GSD. Experiments demonstrate that our lightweight GSD module can enhance the performance of ORB-SLAM2 in glass surface scenes. Additionally, by removing some feature points within the glass surface region, the tracking speed has even increased. The speed improvement varies depending on the proportion of removed feature points to the total number of feature points. For example, in the *corridor_chair_day* sequence, the speed improvement is approximately 58.16%. However, due to the potential risk of removing too many feature points, which reduces the number of map points that can be tracked, our method's camera trajectory estimation performance may also be affected. Considering that ORB-SLAM2 is not primarily designed for dense reconstruction, we plan to integrate our GSD module with direct methods in

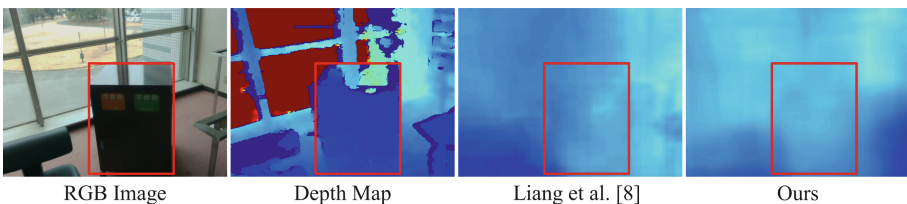


Fig. 9. Examples of our CGSDNet-Depth and Liang *et al.*'s method [8] in real-world scenes. Although both our CGSDNet-Depth and Liang *et al.*'s method can output the correct depth value of the glass surface acting as an obstacle, their sensitivity to depth estimation of other objects decreases, such as the trash can within the red box. (Color figure online)

RGB-D SLAM, such as ElasticFusion [4], to achieve better dense reconstruction results and global 3D consistency, while mitigating the side effects of filtering out glass surface points. Additionally, given the relatively small scale of our GS RGB-D dataset, we intend to introduce a larger and more diverse RGB-D SLAM dataset specifically for glass surface scenes in the future to provide a more reliable evaluation.

7 Conclusion

In this paper, we propose the CGSDNet-Depth network and a novel Context Guided Depth Decoder (CGDD), which utilizes contextual features of glass surfaces to guide depth feature generation, enabling high-quality simultaneous glass surface segmentation and depth estimation. We also present a lightweight GSD (Glass Surface Detection) module to enhance ORB-SLAM2’s camera trajectory estimation and dense reconstruction performance in glass surface scenes. Moreover, we construct the first RGB-D dataset specifically designed for glass surface scenes and validate the superiority of our method.

Acknowledgements. This work was supported by JSPS KAKENHI Grant Numbers 18K04041 and 21K12071.

References

1. Zhang, S., Zheng, L., Tao, W.: Survey and evaluation of RGB-D SLAM. In: IEEE Access (2021)
2. Newcombe, R.A.: et al.: KinectFusion: real-time dense surface mapping and tracking. In: IEEE ISMAR (2011)
3. Kerl, C., Sturm, J., Cremers, D.: Dense visual SLAM for RGB-D cameras. In: IROS (2013)
4. Whelan, T., Salas-Moreno, R.F., Glocker, B., Davison, A.J., Leutenegger, S.: ElasticFusion: real-time dense SLAM and light source estimation. In: IJRR (2016)
5. Mur-Artal, R., Tardós, J.D.: ORB-SLAM2: an open-source SLAM system for monocular, stereo and RGB-D cameras. In: IEEE T-RO (2017)
6. Sturm, J., Engelhard, N., Endres, F., Burgard, W., Cremers, D.: A benchmark for the evaluation of RGB-D SLAM systems. In: IROS (2012)
7. Chen, Z., Mikawa, M., Fujisawa, M.: CGSDNet: cascade network with ConvNeXt as backbone for glass surface detection. In: IEEE ICAICA (2023)
8. Liang, Y., Deng, B., Liu, W., Qin, J., He, S.: Monocular depth estimation for glass walls with context: a new dataset and method. In: TPAMI (2023)
9. Piccinelli, L., Sakaridis, C., Yu, F.: iDisc: internal discretization for monocular depth estimation. In: CVPR (2023)
10. Silberman, N., Hoiem, D., Kohli, P., Fergus, R.: Indoor segmentation and support inference from RGBD images. In: ECCV (2012)
11. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.: ORB: an efficient alternative to SIFT or SURF. In: ICCV (2011)
12. Linus, N., Rueckert, E.: Understanding why SLAM algorithms fail in modern indoor environments. In: RAAD (2023)

13. Koch, R., May, S., Koch, P., Kühn, M., Nüchter, A.: Detection of specular reflections in range measurements for faultless robotic SLAM. In: Robot 2015 (2016)
14. Yamaguchi, E., Higuchi, H., Yamashita, A., Asama, H.: Glass detection using polarization camera and LRF for SLAM in environment with glass. In: REM (2020)
15. Zhu, Y., Qiu, J., Ren, B.: Transfusion: a novel SLAM method focused on transparent objects. In: ICCV (2021)
16. Mei, H., et al.: Don't Hit Me! Glass Detection in Real-world Scenes. In: CVPR (2020)
17. Lin, J., He, Z., Lau, R.W.H.: Rich context aggregation with reflection prior for glass surface detection. In: CVPR (2021)
18. Yuan, W., Gu, X., Dai, Z., Zhu, S., Tan, P.: Neural window fully-connected CRFs for monocular depth estimation. In: CVPR (2022)
19. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: CVPR (2017)
20. Woo, S., Park, J., Lee, J.Y., Kweon, I.S.: CBAM: Convolutional Block Attention Module. In: ECCV (2018)
21. Liu, Z., Mao, H., Wu, C., Feichtenhofer, C., Darrell, T., Xie, S.: A ConvNet for the 2020s. In: CVPR (2022)
22. Eigen, D., Puhrsch, C., Fergus, R.: Depth map prediction from a single image using a multi-scale deep network. In: NeurIPS (2014)
23. Lee, J.H., Han, M.K., Ko, D.W., Suh, I.H.: From big to small: multi-scale local planar guidance for monocular depth estimation. [arXiv:1907.10326](https://arxiv.org/abs/1907.10326) (2019)
24. Yu, L., et al.: Progressive Glass Segmentation. In: TIP (2022)
25. Paszke, A., et al.: Pytorch: An Imperative Style. High-Performance Deep Learning Library, In NeurIPS (2019)



Content-Aware Feature Upsampling for Voxel-Based 3D Semantic Segmentation

Yu Song¹, Ruigang Fu^{1(✉)}, Qingyong Hu², Biao Li¹, and Ping Zhong¹

¹ National University of Defense Technology, Changsha 410073, China
{songyu22, furuigang08, libiao, zhongping}@nudt.edu.cn

² Chinese Academy of Military Sciences, Beijing 100000, China
huqingyong15@nudt.edu.cn

Abstract. Voxel-based sparse convolutional networks (sparse CNNs) are widely used in 3D point cloud semantic segmentation. In particular, feature upsampling, as one of the fundamental operations in the sparse CNNs, has been under-explored compared with other basic operations such as sparse convolution and pooling. Therefore, we dive deep into this area and focus on the upsampling design in sparse CNNs. 3D sparse deconvolution is the most representative feature unsampling in sparse CNNs. However, it applies the same kernel across the point cloud, regardless of the content of each point. To this end, we propose 3D Content-Aware Feature Upsampling (3DCAFU), a universal and effective module beyond sparse deconvolution in sparse CNNs. 3DCAFU has three appealing properties: (1) Content-aware processing. Instead of a fixed kernel for the point cloud feature, 3DCAFU generates point-wise kernels specific to each point for adaptive upsampling. (2) Context aggregation. Since the generation of the point-wise kernels aggregates the context of local neighborhoods, it makes the upsampled feature of 3DCAFU contain richer semantic information compared with sparse deconvolution. (3) Lightweight and efficient. 3DCAFU introduces little extra parameters and accelerates the computation on GPUs by gather-scatter paradigm. Extensive experiments on the SemanticKITTI, SemanticPOSS, nuScenes, and Waymo benchmarks validate the effectiveness of our approach. For instance, it outperforms the baseline by 1.7% mIoU in the SemanticKITTI dataset. SphereFormer with 3DCAFU has achieved state-of-the-art performance among voxel-based methods for 3D semantic segmentation. The code will be made publicly available soon.

Keywords: 3D Content-Aware Feature Upsampling · Voxel-based Sparse CNNs · 3D Point Cloud Semantic Segmentation

1 Introduction

Point cloud is a set of points obtained by 3D sensors. Compared with image, point cloud provides reliable and accurate depth information. 3D Point Cloud semantic segmentation is to assign a semantic label to each point, which acts as an essential component in autonomous driving, digital cities, and service robots.

Unlike 2D image, 3D point cloud is highly sparse and irregular. Therefore, how to learn effective representations from point clouds is a big challenge for semantic segmentation. With the advent of deep learning, an enormous amount of methods have been proposed. View-based methods flatten 3D point clouds into dense 2D representations using spherical projection [16] and bird’s-eye view projection [32]. However, the projections inevitably destroys the physical dimension distortion and height information. Point-based methods process original point clouds directly, based on PointNet [19] and PointNet++ [20] networks. However, the neighbor sampling and grouping operations are time-consuming due to the unstructured point locations. Voxe-based methods rasterize the point clouds into voxels that retain regular structure and apply 3D sparse CNNs for efficient feature extraction [6, 8, 9]. Voxel-based sparse CNNs achieve state-of-the-art performance in multiple large-scale outdoor point cloud semantic segmentation benchmarks, and we further research based on the method.

Voxel-based sparse CNNs mostly adopt encoder-decoder architecture in 3D semantic segmentation. Specifically, the encoder gradually downsamples the points using sparse convolution and captures high-level semantic features. The decoder focuses on recovering object details and spatial dimensions through upsampling the points. As shown in Fig. 1, 3D sparse convolution [8, 9] takes input from a specific region of points and outputs a point along with a hashmap when downsampling. The hashmap reflects a many-to-one mapping relationship between these points through their coordinates. If there is no point within the region, the corresponding position in the output feature map is a null value. 3D sparse convolution avoids the computation issue by restricting the output feature positions to the input. However, it results in sparse deconvolution failing to fully utilize neighborhood information during upsampling. Besides, similar to 2D deconvolution, 3D sparse deconvolution also cannot dynamically generate upsampling kernels based on the content of the input feature map.

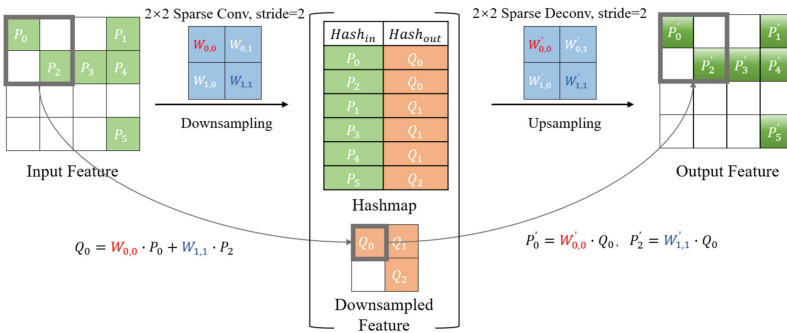


Fig. 1. Sparse convolution in downsampling and sparse deconvolution in upsampling.

To address the aforementioned problems, we propose a content-aware feature upsampling module named 3DCAFU for voxel-based sparse CNNs. Inspired by

CARAFE [27] in 2D images, 3DCAFU is capable of adapting to the feature content, aggregating contextual information, and maintaining computation efficiency. 3DCAFU consists of two parts: Kernel Generation Module and Upsampling Module. Specifically, the Kernel Generation Module predicts an upsampling kernel for each position of the output feature map, whose weights are generated from the content of the input feature map. Rather than being learned as network parameters, these weights are dynamically predicted using sparse convolution layers, which can aggregate the local neighborhood information of the input feature. In the Upsampling Module, we perform linear computation between input features at specific positions and their corresponding generated kernels, and then obtain the output features at the target positions according to the existing hashmap. Additionally, we perform dimensionality reduction in the Kernel Generation Module to reduce model complexity. At the same time, to maintain computational efficiency, we accelerate the computation of 3DCAFU on GPUs by the gather-scatter paradigm.

Our 3DCAFU can be easily integrated into voxel-based sparse CNNs. To demonstrate its effectiveness, we experiment with existing 3D semantic segmentation frameworks [6, 8, 33]. Our method achieves great enhancement on SemanticKITTI [1], SemanticPOSS [18], nuScenes [2], and Waymo [22] benchmarks. These results manifest that our approach has practical value. The key contributions of our paper are highlighted as follows:

- We propose a lightweight and efficient learnable feature upsampling module called 3DCAFU by generating content-aware kernels. It fully utilizes local neighborhood information and can be seamlessly plugged into various voxel-based sparse CNNs for 3D semantic segmentation.
- Extensive experiments are completed on four large-scale point cloud datasets using representative 3D sparse CNNs to verify the availability and scalability of our 3DCAFU.
- SphereFormer with 3DCAFU has achieved state-of-the-art performance among voxel-based methods for 3D semantic segmentation.

2 Related Work

2.1 3D Semantic Segmentation

The purpose of 3D semantic segmentation is to predict point-wise semantic labels for a given point cloud. This technology has developed rapidly in recent years, mainly thanks to the implementation of various deep neural networks. Approaches for LiDAR point cloud semantic segmentation can be roughly grouped into four categories, i.e., view-based, point-based, voxel-based, and hybrid-based methods. View-based methods [11, 30] transform the point cloud into a range view or a bird’s-eye view, and then use a 2D network for feature extraction. Point-based methods [10, 29] directly take the coordinates and features of points as input and design a variety of operators to aggregate neighborhoods. Voxel-based methods [5, 12, 24, 33] transform point clouds into regular

voxels, and then apply 3D sparse CNNs to extract features. Hybrid-based methods [15, 31] either combine the three modes of view, point, and voxel, or combine 2D images and 3D point clouds to achieve multi-modal feature fusion and obtain rich semantic information. Recently, state-of-the-art works for 3D semantic segmentation tend to rely largely or fully on sparse CNNs. We follow this line of research and propose an efficient feature upsampling operation in sparse CNNs.

2.2 Feature Upsampling

Feature Upsampling in 2D CNNs. Traditional interpolation-based upsampling approaches such as nearest and bilinear interpolation, have been extensively adopted in classical models for their simplicity. However, they fundamentally only leverage distances to measure the correlations between pixels and use hand-crafted upsampling kernels. These limitations have motivated researchers to explore learnable upsampling techniques, such as deconvolution [17], pixel shuffle [21], deformable convolution [7], dynamic convolution [3], CARAFE [27]. Deconvolution is an inverse operation of the convolution and widely used. However, the deconvolution has not considered the local variations explicitly in the images, since it applies the same kernel across different locations. Pixel shuffle reshapes depth on the channel space into width and height on the spatial space. Deformable convolution combines the idea of geometric transformations with regular convolutional layers to predict kernel offsets. CARAFE proposes a different upsampler that can generate upsampling kernels based on the input feature map.

Feature Upsampling in 3D CNNs. Similar to 2D, 3D interpolation methods determine weights based on the spatial distance between points to achieve upsampling. For point-based methods, such as KPConv [26], PointConv [28], and RS-Conv [14], there are several learnable feature upsampling operators being proposed. The convolution weights of KPConv are located in Euclidean space by kernel points, and applied to the input points close to them. PointConv treats convolution kernels as nonlinear functions of the local coordinates of 3D points comprised of weight and density functions. With respect to a given point, the weight functions are learned with multi-layer perceptron networks and the density functions through kernel density estimation. In RS-Conv, the convolutional weight for local point set is forced to learn a high-level relation expression from predefined geometric priors, between a sampled point from this point set and the others. However, they only operate on points, which is not suitable for voxel-based sparse convolutional networks.

In voxel-based sparse CNNs, the most commonly used feature upsampling operation is sparse deconvolution [8, 9], which is an inverse operation of sparse convolution. The problem of kernels not adapting to input feature content also exists in 3D sparse deconvolution. Due to the sparsity of point cloud, sparse deconvolution faces new challenges of being unable to fully obtain neighborhood information. Recently, there are some novel convolutions, such as

Focal Sparse Convolution(FocalSConv) [4], Sparse Depthwise Separable Convolution(SDSConv) [13]. However, there is not much research on specialized feature upsampling operator. Therefore, we would like to design a content-aware feature upsampling operator in 3D sparse CNNs.

Inspired by CARAFE, we design 3DCAFU, a feature upsampling operator in the 3D sparse CNNs, which can both expand the receptive field and adaptively generate upsampling kernels. 3DCAFU and dynamic convolution share similar design philosophy but with different focuses. Both dynamic convolution and 3DCAFU are content-aware operators, but a fundamental difference between them lies at their kernel generation process. Specifically, dynamic convolution works as a two-step operators, where the additional kernel prediction layer and convolution layer require heavy computation. On the contrary, 3DCAFU is simply a reassembly of features in local regions, without learning the feature transformation across channels. Thus, it is more efficient in memory and speed.

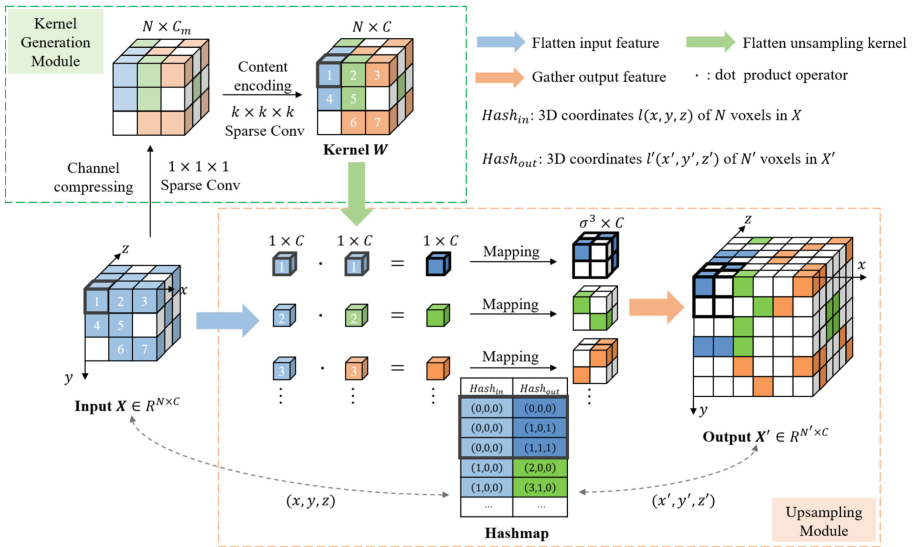


Fig. 2. The overall framework of 3DCAFU. It is composed of two key components, i.e., Kernel Generation Module and Upsampling Module. The kernel generation module generates upsampling kernels based on the input. The upsampling module first performs dot product of the input and the kernels, and then obtains the output through mapping with the hashmap. The hashmap records the coordinate mapping relationship of points during the downsampling process, which can be used directly while upsampling. The specific application of hashmap is in Sec. 3.3. White areas in point cloud and feature maps represent null values.

3 Method

Feature upsampling is a key operation in 3D sparse CNNs for point cloud semantic segmentation. In this paper, we propose a content-aware feature upsampling operator (3DCAFU). For each position in the output feature map, 3DCAFU can predict an upsampling kernel through the content of the input feature map. Therefore, 3DCAFU can use adaptive kernels at different locations and aggregate the local neighborhood information. Meanwhile, 3DCAFU introduces little extra parameters and maintains the computation efficiency on GPUs by optimized gather-scatter paradigm. Compared with mainstream feature upsampling operators, 3DCFU achieves better performance.

3.1 Formulation

As shown in Fig. 2, 3DCAFU generates upsampling kernels from the input feature map in Kernel Generation Module. Upsampling Module reorganizes the input feature with the predicted kernels and outputs upsampled feature through mapping with the existing hashmap. Given a feature map $X \in R^{N \times C}$ and an upsample ratio σ (supposing σ is an integer), 3DCAFU produces a new feature map $X' \in R^{N' \times C}$. N and N' represent the number of voxels in the input and output respectively, and C represents the feature dimension. For any target location $l' = (x', y', z')$ of the output feature map X' , there is a corresponding source location $l = (x, y, z)$ in the input feature map X . The location mapping relationship is stored in the hashmap which has been generated during down-sampling.

Kernel Generation Module Ψ is shown in Eqn. 1, where $R(X_l, k)$ is the $k \times k \times k$ sub-region of X centered at the location l . The module predicts a location-wise kernel W_l for each location l in the input X , based on the neighbor of X_l . Upsampling module Φ reassembles feature X_l at the source location with the kernel W_l , and then obtains feature $X'_{l'}$ at the target location through mapping with the hashmap, as shown in Eqn. 2.

$$W_l = \Psi(R(X_l, k)) \quad (1)$$

$$X'_{l'} = \Phi(X_l, W_l) \quad (2)$$

3.2 Kernel Generation Module

The purpose of the kernel generation module is to generate content-aware upsampling kernels. Each source location on X corresponds to σ^3 target locations on X' . Due to the sparsity of the point cloud, voxels at some locations are empty and have null feature values. We predict a $1 \times C$ kernel for X_l by sparse convolution layers. It not only adapts the upsampling kernels to the position and content of voxels, but also contains context information in the kernels.

Channel Compressing. We adopt a $1 \times 1 \times 1$ sparse convolution layer to compress the input feature channel from C to C_m . Reducing the channel of input feature map leads to less parameters and computational cost in the following steps, making 3DCAFU more efficient. Experimental results show that reducing the feature channel in an acceptable range will not harm the performance.

Content Encoding. In the encoding step, we adopt a $k \times k \times k$ sparse convolution to generate upsampling kernels. When k is constant, channel compression reduces the parameter of the encoder from $k^3 \times C \times C$ to $(k^3 + 1) \times C_m \times C$ because C_m is generally much smaller than C . It is also possible to use larger kernel sizes for the encoding step under the same budget. Intuitively, increasing k can expand the receptive field and exploit contextual information within a larger region, which is important for the prediction of upsampling kernels. However, the computational complexity grows with the cube of the kernel size, while the benefits from a larger kernel size do not. We need to choose appropriate C_m and k with a good trade-off between performance and efficiency.

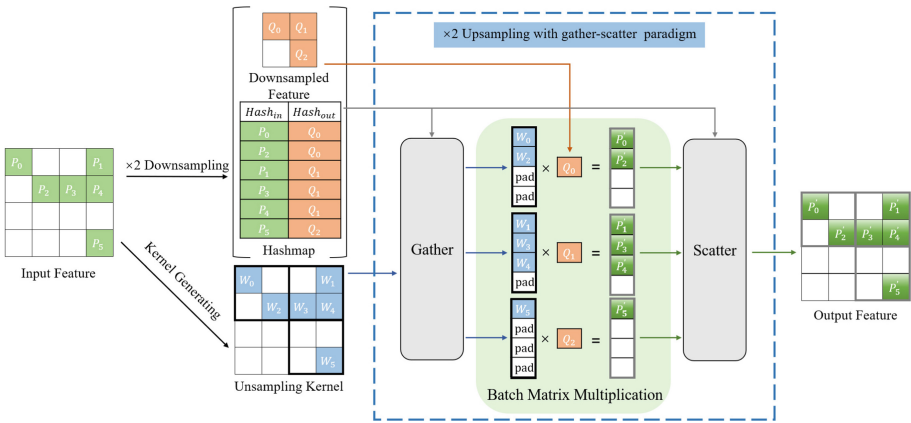


Fig. 3. The gather-scatter paradigm in 3DCAFU is to accelerate the computation on GPUs

3.3 Upsampling Module

With each kernel W_l , the upsampling module reassembles X_l via the function Φ . We adopt a simple form of Φ , which is just a dot product operator. As shown in Eqn. 3, after dot producting of the source feature X_l and the corresponding kernel W_l , the target feature X'_l is finally obtained through *Mapping* with with the hashmap. As shown in Fig. 2, for instance, the input feature with coordinates $(1, 1, 1)$ corresponds to three output features with coordinates $(0, 0, 0)$, $(1, 0, 1)$, and $(1, 1, 1)$. Therefore, the three feature values are all the results of the dot

product of the input feature with its corresponding upsampling kernel. With the predicted kernel, the same voxel contributes to the upsampled voxels differently. 3DCAFU employs more contextual information from local neighborhoods, so the semantics of the upsampled output feature map can be stronger.

$$X'_l = \text{Mapping}(X_l \cdot W_l) \quad (3)$$

Unlike conventional dense computation, sparse workload was not favored by modern high-parallelism hardware. On the one hand, the sparse nature of point clouds leads to irregular computation workloads: i.e., different kernels might correspond to drastically different numbers of matched input/output pairs. On the other hand, neighboring points do not lie contiguously in the sparse point cloud representation. In order to improve calculation and reduce memory, our upsampling module is based on torchsparse [23, 25], a high-performance inference engine library. Following the gather-scatter paradigm as shown in Fig. 3, we not only group the input feature vectors, but also the generated kernels according to the kernel offset. Then the corresponding features and kernels are processed in batches to achieve regular calculations. Finally, these dot product results are scattered and accumulated to the corresponding output feature vectors.

4 Experiments

4.1 Experimental Setting

Datasets and Evaluation Metrics. Following previous work, we evaluate methods on SemanticKITTI [1], SemanticPOSS [18], nuScenes [2], and Waymo Open Dataset [22] for 3D semantic segmentation. SemanticKITTI is a large-scale outdoor traffic scene dataset recorded with a Velodyne-64 LiDAR scanner. It consists of 43511 scans with point-wise annotations of 19 semantic classes. We follow the widely-adopted split and use sequences 00-07, 09-10 as the training set and sequence 08 for validation. SemanticPOSS consists of 2988 annotated point cloud scans of 14 semantic classes. We follow the official benchmark setting, i.e. sequence 03 for validation and the rest for training. NuScenes consists of 1000 driving scenes where 850 scenes are selected for training and validation, and the remaining 150 scenes are taken as the testing split. It is collected with a 32 beams LiDAR sensor at 20Hz frequency with point-wise annotations of 16 semantic classes. Waymo Open Dataset collects point cloud scans in 1150 scenes of 20s duration. It is collected with five LiDAR sensors at 20Hz frequency with point-wise annotations of 23 semantic classes. We follow the official split of training data and validation data.

Network Architecture. We evaluate 3DCAFU over three widely adopted semantic segmentation networks: 1) As shown in Fig 4, MinkUNet [6] is a typical voxel-based model that implements the idea of sparse convolutional networks originally presented by Graham et al. [8]. 2) SPVCNN [24] is a hybrid

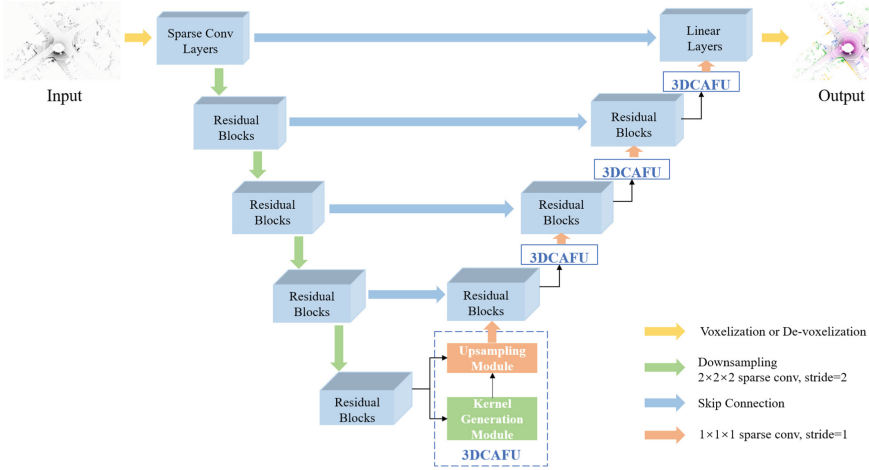


Fig. 4. 3DCAFU in MinkUNet for 3D semantic segmentation.

network with sparse convolutions and a point-based sub-network. With negligible overhead, the point-based branch can preserve the fine details even from large outdoor scenes. 3) RPNNet [31] proposes a range-point-voxel fusion network to utilize different view’s advantages and alleviate their shortcomings in segmentation task. 4) SphereFormer(SpTr) [12] is a voxel-based model, which significantly boosts the performance of sparse distant points by radial window self-attention and achieves state-of-the-art performance in voxel-based methods.

Table 1. Semantic segmentation results of MinkUNet using different feature upsampling methods on SemanticKITTI, Waymo, SemanticPOSS, and nuScenes val sets. And additional parameters related to the upsampling methods.

method	param.	Sem.KITTI	Waymo	Sem.POSS	nuScenes
Nearest(Near.)	0	58.8	56.9	55.5	71.7
Trilinear	0	59.0	57.8	56.0	72.1
Sparse Deconv	240k	60.6	59.5	57.1	73.3
Near.+SparseConv	240k	60.5	59.3	57.1	73.1
Near.+FocalSConv	240k	61.1	60.0	57.9	73.7
Near.+SDSConv	33k	59.7	58.5	56.4	72.8
3DCAFU	136k	62.3(+1.7)	60.8(+1.3)	58.9(+1.8)	74.6(+1.3)

Implementation Details. We conducted experiments with a TITAN RTX GPU for MinkUNet and a single RTX 4090 GPU for SPVCNN, RPNNet, and

Table 2. Semantic segmentation results on SemanticKITTI, Waymo, SemanticPOSS and nuScenes val sets. ★ means with 3DCAFU.

method	frames/s	Sem.KITTI	Waymo	Sem.POSS	nuScenes
MinkUNet	9.42	60.6	59.5	57.1	73.3
MinkUNet★	9.28	62.3	60.8	58.9	74.6
SPVCNN	7.59	63.0	61.3	60.6	77.4
SPVCNN★	7.48	63.9	62.1	61.8	78.1
RPVNet	6.45	65.2	63.0	63.7	77.6
RPVNet★	6.46	66.0	63.5	64.4	78.0
SpTr	4.21	67.8	64.2	78.4	69.9
SpTr★	4.09	69.2	70.0	64.6	78.9

SphereFormer. We adopt the default training hyper-parameters in the open-source libraries for the networks. We train the models for 30 epochs with sgd optimizer named OpenPCSeg [15] and cosine warmup scheduler where momentum is set to 0.9. The learning rate and weight decay are set to 0.24 and 0.0001, respectively. Batch size is set to 4 on Waymo Open Dataset, and 8 on SemanticKITTI, SemanticPOSS and nuScenes. If not otherwise specified, 3DCAFU adopts a fixed set of hyper-parameters in experiments, where the compressed channel C_m is 32, the upsampling rate σ is 2, and the convolution kernel size k is 3.

4.2 Semantic Segmentation Results

Effectiveness. We apply 3DCAFU in MinkUNet to evaluate the semantic segmentation results on SemanticKITTI, Waymo, SemanticPOSS, and nuScenes. As shown in Table 1, 3DCAFU has a significant improvement in the performance of semantic segmentation compared with the other upsampling methods. MinkUNet with 3DCAFU improves the mIoU by 1.7% and 1.8% in the SemanticKITTI and SemanticPOSS. 3DCAFU is effective for different baselines and datasets, indicating its robustness. Compared with interpolation methods, the kernels of 3DCAFU are learnable. Compared with sparse deconvolution, 3DCAFU has fewer parameters and better performance. Interpolation combined with convolution can also achieve feature upsampling. We conduct experiments using nearest-neighbor interpolation combined separately with sparse convolution, focal sparse convolution, and sparse depthwise separable convolution. The latter two are currently advanced 3D sparse convolutions. Compared with the methods, 3DCAFU achieves significant performance improvements with small parameter increments. The results show that our operator is lightweight and effective.

Universality. The semantic segmentation results on the SemanticKITTI, Waymo, SemanticPOSS and nuScenes datasets are shown in Table 2. When using 3DCAFU, the performance of baselines on these benchmarks has been improved. For the same baseline MinkUNet, 3DCAFU improves segmentation performance

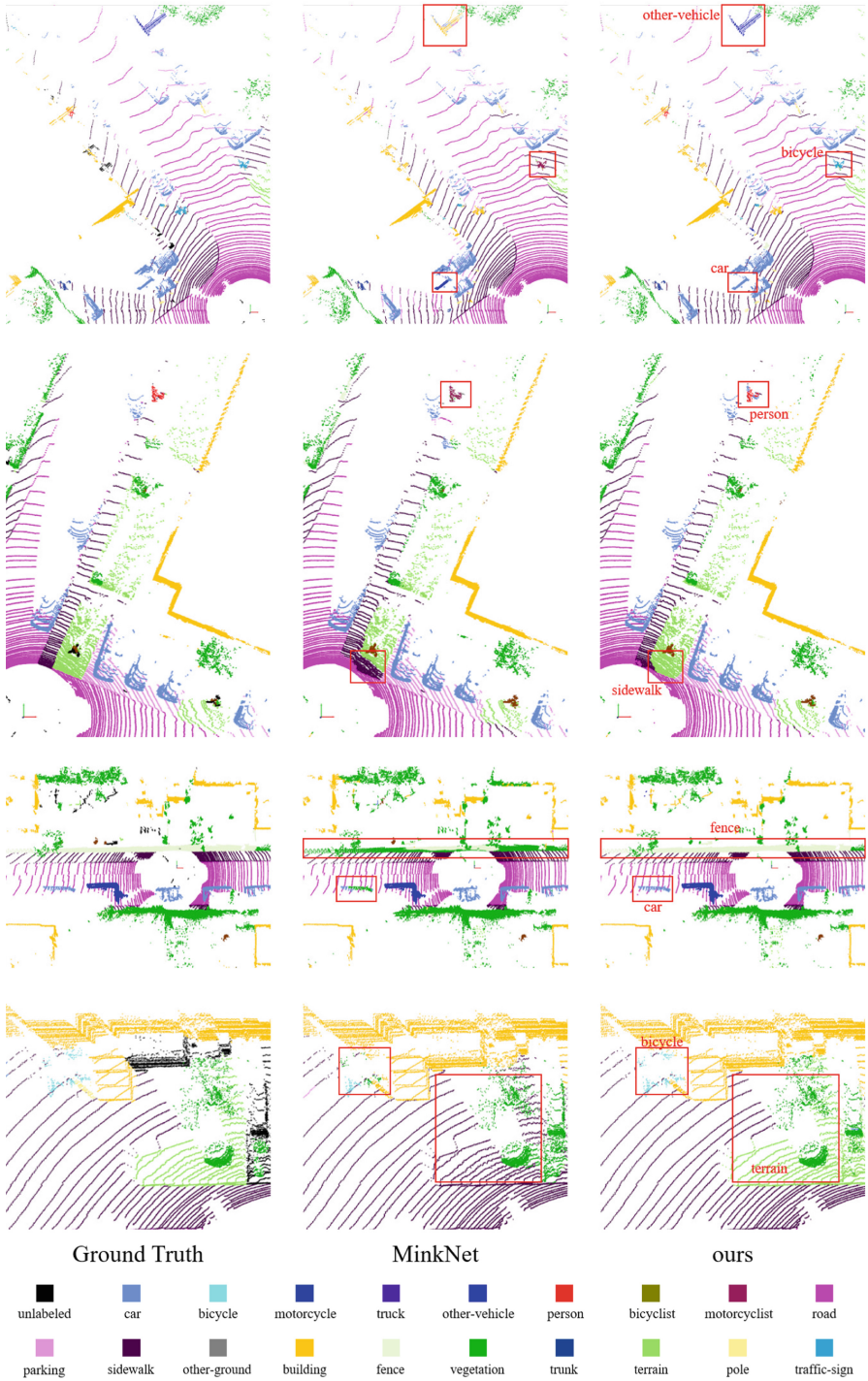


Fig. 5. Visual comparison between vanilla MinkUNet and ours(MinkUNet with 3DCAFU) on SemanticKITTI validation.

Table 3. Detailed semantic segmentation results on SemanticKITTI val set. \star means with 3DCAFU.

method	mIoU	car	bi.cle	mt.cle	truck	oth-ve	pers.	bi.clst	mt.clst	road
MinkUNet	60.6	95.8	12.8	59.1	66.8	57.9	60.5	78.6	0.0	93.4
MinkUNet \star	62.3	96.6	19.9	61.3	68.6	55.5	66.7	86.7	0.3	93.7
SPVCNN	63.0	96.0	32.4	66.4	67.1	52.9	74.8	84.3	0.0	93.3
SPVCNN \star	63.9	96.5	35.9	65.0	66.6	60.2	75.3	83.3	0.1	93.8
RPVNet	65.2	96.3	51.2	75.6	63.4	63.9	71.9	85.6	0.1	93.6
RPVNet \star	66.0	96.5	53.9	79.7	68.5	64.9	75.6	87.8	0.2	93.5
SpTr	67.8	96.6	54.4	77.4	65.1	62.0	79.8	89.9	1.6	91.7
SpTr \star	69.2	96.8	56.0	76.9	63.2	62.8	82.0	90.2	2.4	93.5

method	park.	sidew.	oth-gr.	build.	fence	veget.	trunk	terra.	pole	traf.
MinkUNet	48.7	79.9	0.0	91.0	62.8	89.1	67.6	76.7	63.9	48.6
MinkUNet \star	52.0	80.7	1.2	90.9	61.2	88.8	69.1	75.2	63.3	50.4
SPVCNN	46.9	80.2	1.4	91.1	64.1	88.1	67.0	73.9	64.0	51.6
SPVCNN \star	49.0	81.1	2.5	90.6	60.0	89.2	70.2	76.4	64.8	50.5
RPVNet	45.8	81.4	1.1	91.0	62.8	88.4	68.5	75.0	64.6	49.9
RPVNet \star	47.3	81.2	1.4	91.2	63.8	88.2	68.2	74.2	64.5	49.4
SpTr	45.8	80.9	2.9	94.0	70.6	90.4	67.2	80.8	62.5	69.5
SpTr \star	50.6	81.2	3.1	93.6	71.3	90.5	70.2	80.2	63.8	70.1

by 1.7%, 1.3%, 1.8%, and 1.3% mIoU on the SemanticKITTI, Waymo, SemanticPOSS, and nuScenes datasets respectively. On the SemanticKITTI dataset, 3DCAFU improves the segmentation performance of the MinkUNet, SPVCNN, RPVNet, and SpTr models by 1.7%, 0.9%, 0.8%, and 1.4% mIoU respectively. These results demonstrate that 3DCAFU has good robustness.

Advancement. The detailed results on the SemanticKITTI dataset are shown in Table 3. The segmentation accuracy of small objects such as car, bicycle, person, and motorcyclist has been greatly improved. At the same time, 3DCAFU also improves the segmentation accuracy of parking, other-ground. The improvements of both small and big objects are above 1% mIoU, which suggests that 3DCAFU is beneficial for various object scales. This shows that 3DCAFU obtains richer semantic features by aggregating contextual information, which is beneficial to the segmentation. Besides, SphereFormer with 3DCAFU(SpTr \star) has achieved **state-of-the-art** performance among voxel-based methods for 3D semantic segmentation.

As shown in Fig. 5, we visually compare the baseline(i.e., MinkUNet) and ours on SemanticKITTI validation. It visually indicates that with our proposed operation, more objects are segmented correctly, which are highlighted with

Table 4. Ablation study of various compressed channels C_m on SemanticPOSS val set.

C_m	mIoU
32	38.4
64	38.4
128	38.5
256	38.5
N/A	38.5

Table 5. Ablation study of the number of 3DCAFU layers on SemanticPOSS val set.

Num mIoU	
0	36.8
1	37.1
2	37.6
3	37.9
4	38.4

Table 6. Ablation study of upsampling rate σ and kernel size k on SemanticPOSS val set.

σ	k	mIoU
2	3	38.4
2	5	38.4
2	7	38.5
3	3	37.0
3	5	37.1
3	7	37.1

Table 7. Ablation study of gather-scatter paradigm in 3DCAFU on SemanticPOSS.

gather-scatter frames/s	
w/	9.28
w/o	1.57

red boxes. In the first and second rows of point cloud segmentation results, our method accurately segments small objects such as bicycles, persons and distant cars. As shown in the third and fourth rows, 3DCAFU also improves the segmentation accuracy of large-scale targets such as fences and sidewalks. In summary, the visualization results verify the effectiveness of our approach.

4.3 Ablation Study

We investigate the influence of hyper-parameters in the model design, i.e., the compressed channels C_m and sparse convolution kernel size k . We also test the impact of the number of 3DCAFU layers. We perform the ablation study on MinkUNet and train it with sequences 01-02 of SemanticPOSS.

We experiment with different values of C_m in the kernel generation module. In addition, we also try to remove the channel compressor and directly use input features to predict upsampling kernels. Experimental results in Table 4 show that C_m down to 128 leads to no performance decline, while being more efficient. A further smaller C_m will result in a slight drop in the performance. With no channel compressor, it can achieve the same performance, which proves that the step can speed up the kernel prediction without harming the performance.

Based on the above results, we set C_m to 32 by default as a trade-off between performance and efficiency.

Table 5 shows that as the number of 3DCAFU layers increases, the performance of the network also improves. These results prove the effectiveness of our operator. Therefore, we apply 3DCAFU in all upsampling layers of the model to achieve the best segmentation performance.

We investigate the relationship between upsampling rate σ and the sparse convolution kernel size k in the kernel generation module. As illustrated in Table 6, increasing σ needs a larger k since the kernel generation module requires a large

receptive field to predict upsampling kernels. We summarize an empirical formula that $k = 2\sigma - 1$, which is a good choice in all the settings.

As shown in Table 7, our model can process 9.28 frames of point cloud per second on GPU by using the gather-scatter paradigm. However, without this paradigm, it can only process 1.57 frames per second. These data significantly demonstrate that the gather-scatter paradigm ensures the computational efficiency of 3DCAFU on GPU.

5 Conclusion

In this paper, we propose an effective and lightweight content-aware feature upsampling (3DCAFU) for voxel-based sparse CNNs. The key idea is to generate upsampling kernels through input feature, hence achieving content-aware feature upsampling of sparse point clouds and aggregating context to improve the performance of 3D sparse CNNs. Extensive experimental results on the four datasets show that the proposed 3DCAFU can be seamlessly integrated into existing networks, and effectively improve the performance of 3D semantic segmentation.

References

1. Behley, J., Garbade, M., Milioto, A., Quenzel, J., Behnke, S., Stachniss, C., Gall, J.: Semantickitti: A dataset for semantic scene understanding of lidar sequences. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 9297–9307 (2019)
2. Caesar, H., Bankiti, V., Lang, A.H., Vora, S., Liong, V.E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., Beijbom, O.: nuscenes: A multimodal dataset for autonomous driving. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11621–11631 (2020)
3. Chen, Y., Dai, X., Liu, M., Chen, D., Yuan, L., Liu, Z.: Dynamic convolution: Attention over convolution kernels. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11030–11039 (2020)
4. Chen, Y., Li, Y., Zhang, X., Sun, J., Jia, J.: Focal sparse convolutional networks for 3d object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5428–5437 (2022)

5. Cheng, R., Razani, R., Taghavi, E., Li, E., Liu, B.: 2-s3net: Attentive feature fusion with adaptive feature selection for sparse semantic segmentation network. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 12547–12556 (2021)
6. Choy, C., Gwak, J., Savarese, S.: 4d spatio-temporal convnets: Minkowski convolutional neural networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 3075–3084 (2019)
7. Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., Wei, Y.: Deformable convolutional networks. In: Proceedings of the IEEE international conference on computer vision. pp. 764–773 (2017)
8. Graham, B., Engelcke, M., Van Der Maaten, L.: 3d semantic segmentation with submanifold sparse convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 9224–9232 (2018)
9. Graham, B., Van der Maaten, L.: Submanifold sparse convolutional networks. arXiv preprint [arXiv:1706.01307](https://arxiv.org/abs/1706.01307) (2017)
10. Hu, Q., Yang, B., Xie, L., Rosa, S., Guo, Y., Wang, Z., Trigoni, N., Markham, A.: Randla-net: Efficient semantic segmentation of large-scale point clouds. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11108–11117 (2020)
11. Kong, L., Liu, Y., Chen, R., Ma, Y., Zhu, X., Li, Y., Hou, Y., Qiao, Y., Liu, Z.: Rethinking range view representation for lidar segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 228–240 (2023)
12. Lai, X., Chen, Y., Lu, F., Liu, J., Jia, J.: Spherical transformer for lidar-based 3d recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 17545–17555 (2023)
13. Li, L., Shum, H.P., Breckon, T.P.: Less is more: Reducing task and model complexity for 3d point cloud semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9361–9371 (2023)
14. Liu, Y., Fan, B., Xiang, S., Pan, C.: Relation-shape convolutional neural network for point cloud analysis. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8895–8904 (2019)
15. Liu, Y., Chen, R., Li, X., Kong, L., Yang, Y., Xia, Z., Bai, Y., Zhu, X., Ma, Y., Li, Y., et al.: Uniseg: A unified multi-modal lidar segmentation network and the openpcseg codebase. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 21662–21673 (2023)
16. Milioto, A., Vizzo, I., Behley, J., Stachniss, C.: Rangenet++: Fast and accurate lidar semantic segmentation. In: 2019 IEEE/RSJ international conference on intelligent robots and systems (IROS). pp. 4213–4220. IEEE (2019)
17. Noh, H., Hong, S., Han, B.: Learning deconvolution network for semantic segmentation. In: Proceedings of the IEEE international conference on computer vision. pp. 1520–1528 (2015)
18. Pan, Y., Gao, B., Mei, J., Geng, S., Li, C., Zhao, H.: Semanticpos: A point cloud dataset with large quantity of dynamic instances. In: 2020 IEEE Intelligent Vehicles Symposium (IV). pp. 687–693. IEEE (2020)
19. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: Deep learning on point sets for 3d classification and segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 652–660 (2017)
20. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems* **30** (2017)

21. Shi, W., Caballero, J., Huszár, F., Totz, J., Aitken, A.P., Bishop, R., Rueckert, D., Wang, Z.: Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1874–1883 (2016)
22. Sun, P., Kretzschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., et al.: Scalability in perception for autonomous driving: Waymo open dataset. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2446–2454 (2020)
23. Tang, H., Liu, Z., Li, X., Lin, Y., Han, S.: Torchspase: Efficient point cloud inference engine. *Proceedings of Machine Learning and Systems* 4, 302–315 (2022)
24. Tang, H., Liu, Z., Zhao, S., Lin, Y., Lin, J., Wang, H., Han, S.: Searching efficient 3d architectures with sparse point-voxel convolution. In: European conference on computer vision. pp. 685–702. Springer (2020)
25. Tang, H., Yang, S., Liu, Z., Hong, K., Yu, Z., Li, X., Dai, G., Wang, Y., Han, S.: Torchspase++: Efficient point cloud engine. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 202–209 (2023)
26. Thomas, H., Qi, C.R., Deschard, J.E., Marcotegui, B., Goulette, F., Guibas, L.J.: Kpconv: Flexible and deformable convolution for point clouds. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 6411–6420 (2019)
27. Wang, J., Chen, K., Xu, R., Liu, Z., Loy, C.C., Lin, D.: Carafe: Content-aware reassembly of features. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 3007–3016 (2019)
28. Wu, W., Qi, Z., Fuxin, L.: Pointconv: Deep convolutional networks on 3d point clouds. In: Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition. pp. 9621–9630 (2019)
29. Wu, X., Jiang, L., Wang, P.S., Liu, Z., Liu, X., Qiao, Y., Ouyang, W., He, T., Zhao, H.: Point transformer v3: Simpler, faster, stronger. arXiv preprint [arXiv:2312.10035](https://arxiv.org/abs/2312.10035) (2023)
30. Xu, C., Wu, B., Wang, Z., Zhan, W., Vajda, P., Keutzer, K., Tomizuka, M.: SqueezeSegV3: Spatially-Adaptive Convolution for Efficient Point-Cloud Segmentation. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12373, pp. 1–19. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58604-1_1
31. Xu, J., Zhang, R., Dou, J., Zhu, Y., Sun, J., Pu, S.: Rpvnet: A deep and efficient range-point-voxel fusion network for lidar point cloud segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 16024–16033 (2021)
32. Zhang, Y., Zhou, Z., David, P., Yue, X., Xi, Z., Gong, B., Foroosh, H.: Polarnet: An improved grid representation for online lidar point clouds semantic segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9601–9610 (2020)
33. Zhu, X., Zhou, H., Wang, T., Hong, F., Ma, Y., Li, W., Li, H., Lin, D.: Cylindrical and asymmetrical 3d convolution networks for lidar segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9939–9948 (2021)



Enhancing 3D Referential Grounding by Learning Coarse Spatial Relationships

Soham Joshi¹✉, Aditay Tripathi², Viswanath Gopalakrishnan¹,
and Anirban Chakraborty²

¹ International Institute of Information Technology, Bangalore, India
soham.joshi@alumni.iiitb.ac.in, viswanath.g@iiitb.ac.in

² Indian Institute of Science, Bengaluru, India
{aditayt, anirban}@iisc.ac.in

Abstract. Large-scale pre-training is commonly used in 2D referential grounding tasks owing to the easy availability of a large number of image-text pairs with corresponding bounding box annotations. However, for 3D referential grounding, the unavailability of high-quality 3D scene-text pairs with annotations poses a significant challenge. To address this issue, we leverage the large corpus of 3D scenes with bounding box annotations of object instances and design an automated strategy to synthesize scene-text data for pre-training by utilising the coarse spatial relationships between the objects in the scene without any human supervision. The proposed strategy first clusters the 3D bounding boxes and then uses these clusters to create pairwise and triplet relations between the objects in the 3D scene. We achieved improved results consistently across various top-performing methods in 3D referential grounding, when the proposed pre-training strategy is deployed. In addition to pre-training with the samples containing coarse spatial relations, we also encode semantic relationships between the bounding boxes conditioned on the language utterance, using a compatibility measure between the box features and the language utterance. To evaluate the performance of our proposed techniques, we conduct experiments on large-scale publicly available datasets, namely ScanRefer and ReferIt3D (SR3D and NR3D). Our proposed techniques can be seamlessly integrated with any off-the-shelf 3D referential grounding method. Specifically, when integrated with BUTD-DETR, we observed notable improvements of 2.2% and 1% in performance on the SR3D and NR3D datasets, respectively.

Keywords: 3D Referential Grounding · Multimodal Learning · Pre-training 3D Grounding Models

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-78113-1_28.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
A. Antonacopoulos et al. (Eds.): ICPR 2024, LNCS 15330, pp. 427–442, 2025.
https://doi.org/10.1007/978-3-031-78113-1_28

1 Introduction

Pre-training on large-scale image datasets has been shown to improve the performance of various computer vision tasks, including image classification [18], object detection [19], and 2D referential grounding [5]. Typically, these models undergo initial training on a large-scale dataset, followed by fine-tuning on a downstream dataset tailored to a specific task.

3D referential grounding is a crucial area of research in computer vision, with applications in embodied agents and robotics [20, 22, 23], and autonomous navigation systems [21]. However, obtaining a large-scale 3D referential grounding dataset is challenging because: (i) Capturing 3D scenes requires specialized equipment (e.g. LiDAR or structured light scanners) and technical expertise and is time-consuming. Thus, 3D scenes aren't abundantly available as 2D images over the internet. (ii) Annotating 3D data often requires understanding 3D geometry [25, 26], as well as an understanding of how to interpret and annotate point clouds or other types of 3D data. (iii) 3D data annotation can be more time-consuming than 2D data annotation due to larger data volumes and multi-stage processing involving segmentation, labelling, and object positions and orientations annotation. (iv) 3D referential grounding models trained on datasets such as ReferIt3D [1] and ScanRefer [2] (both comprise scenes extracted from the ScanNet dataset [7]). The utterances in these datasets are generated either synthetically using rule-based templates or naturally through human free-form utterances. However, acquiring human annotations for 3D scenes is notoriously challenging, and synthetic textual utterances have limitations as template-based approaches struggle to generalize effectively to varying scene geometries and layouts. Thus, both approaches face limitations in scaling up to larger and more diverse datasets. These challenges, especially the prohibitive cost of 3D annotation, lead to a scarcity of annotated large-scale 3D datasets suitable for pre-training advanced 3D referential grounding models. Therefore, there is a need for a more efficient methodology to acquire aligned 3D scene-text data.

We utilize a large-scale 3D scenes dataset with bounding box annotations of all foreground objects and propose an algorithm that automatically synthesizes

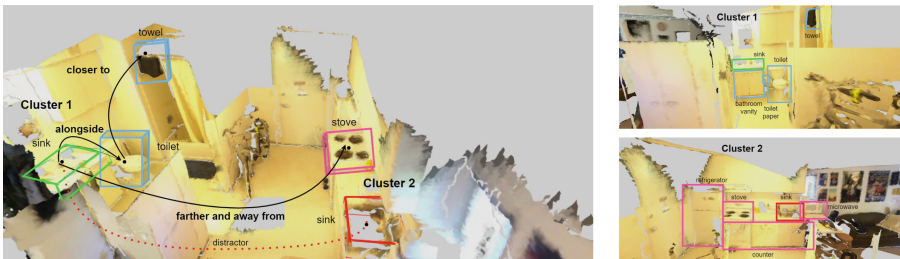


Fig. 1. The left figure depicts a 3D scene with objects annotated with their corresponding bounding boxes and the resulting spatial clusters are illustrated in the figures on the right. Based on this clustering, coarse spatial relationships such as “near” (intra-cluster) and “far” (inter-cluster) are subsequently auto-synthesized between pairs of these objects. Overall, this process automates the synthesis of high-quality 3D scene-text samples that can be used to pre-train 3D referential grounding models.

large samples of the 3D scene-text pairs (Sect. 3.1). We leverage the observation that coarse spatial relations, such as “far” and “near” are dominant among objects in the scene, and learning these fundamental spatial relations can facilitate the learning of more complex inter-object relationships, such as “to the right of”. Our algorithm can effectively obtain these spatial relations from 3D scenes at scale without the need for human supervision. This approach has the potential to overcome the limitations of current datasets and enable the development of more accurate and robust 3D referential grounding models. The benefits of our proposed approach are two-fold. (i) It **reduces the need for human supervision**, which can be time-consuming and expensive. (ii) It **enables the acquisition of more diverse and representative datasets**, which can lead to improved model performance and generalization to real-world scenarios.

The proposed strategy consists of two stages. Firstly, we employ a custom distance function based on the positional features of the bounding boxes to cluster the objects in the scene. For example, in Fig. 1, two clusters are generated: {‘towel’, ‘sink’, ‘toilet’, ‘toilet paper’, and ‘bathroom vanity’} and {‘refrigerator’, ‘counter’, ‘stove’, ‘sink’, and ‘microwave’}. It is important to note that the objects in the scene can be classified as either single-occurrence objects (e.g., ‘microwave’) or multi-occurrence objects (e.g., ‘sink’). In the second step, coarse spatial relationships are auto-generated between the object pairs by utilizing the clustering output from the first step.

Intra-class relationships among the single-occurrence objects within the same cluster are established using “near” spatial relation (e.g., “toilet closer to the towel” in Fig.1), while “far” spatial relations are set between objects located in different clusters, such as “sink farther and away from the stove”. Yet, some object pairs may have distractors in the scene, such as the two ‘sink’ objects in Fig.1. In such cases, the concept of “far” alone would be inadequate to discern between these instances, and hence the inter-cluster relationship (e.g., “sink farther and away from the stove”) is accompanied by a “near” relationship involving the distractor object instance (i.e., “sink alongside the toilet” in Fig.1). The additional relationship helps disambiguate the two ‘sink’ instances, leading to improved grounding.

The 3D referential grounding methods typically use 3D object detectors as backbones. Current methods [29] also use an initial set of object bounding boxes generated by these pre-trained object detectors to improve 3D referential grounding. However, these bounding boxes are used independently, making them inefficient for referential grounding tasks. To address this issue, this work proposes novel **L**anguage-conditioned **S**patially aware **R**elational (LASER) embeddings for each object box, encoding relationship between the bounding boxes. The proposed method utilizes the compatibility measure between the representation of the pairs of bounding boxes and the text query to encode the relationship between objects, making it a more efficient and effective approach for referential grounding (Sect. 3.2). Our contributions are summarized as follows: (I) We propose a novel algorithm that generates large datasets of paired 3D scene-text samples with coarse spatial relationships using existing 3D scenes and their bounding

box annotations. The dataset, augmented using these synthesized text annotations, is then used for performing pre-training of the 3D referential grounding models, which results in significant improvement in grounding accuracy on fine-tuning this pre-trained model on a target dataset. (II) Our pre-training method is versatile and can be seamlessly integrated with various existing 3D referential grounding methods to improve their performance and yield new SOTA for the task. While our primary experiments focus on the BUTD-DETR model [29], we have also shown its effectiveness with other methods such as EDA [13] and ViL3DRel [30]. This flexibility broadens the applicability of our approach across different models, enhancing its potential impact in the field of 3D referential grounding. (III) We also introduce a novel relationship embedding method, which we call **L**anguage-conditioned **S**patially aware **R**elational (**LASER**) embeddings, for encoding the semantic relationships between a set of object bounding boxes conditioned on the query text. By incorporating semantic information from the query text into the spatial relationships between object bounding boxes, our approach enhances the overall accuracy of the 3D referential grounding model.

2 Related Work

Pre-training in 2D Referential Grounding. Recent 2D referential grounding research shows pre-training on large-scale datasets, often combining multiple sources, is key to achieving state-of-the-art performance. For instance, MDETR [5], a popular text-query guided 2D object localization method, pre-trained their model on a curated collection of image-text pairs with bounding box annotations. They combined publicly available image-text datasets with bounding box annotations to create a dataset with 1.3 million image-text samples. DQ-DETR [28] is also a popular example of this approach, demonstrating significant improvements in 2D referential grounding through pre-training.

3D Referential Grounding. In recent years, the development of 3D scan datasets such as ScanNet [7], ScanRefer [2], SunRefer [11], and ReferIt3D [1] has led to a growing interest in the application of computer vision in 3D. 3D referential grounding, an emerging task, locates objects in 3D scenes using language references. This requires multimodal fusion of visual and language features. Recent research has explored the use of graph neural networks [15] and transformer models [3, 4, 29] to facilitate this fusion process. In the case of transformers, the task involves three major steps: first, feature extraction of the 3D scene point cloud, which can be done through 3D object proposals and detection [3, 4, 12–14, 29] or panoptic segmentation [9]; second, encoding the language reference using word token embeddings; and third, fusing these two modalities by employing cross encoders and decoders. BUTD-DETR [29] is a notable contribution in the field of 3D referential grounding, as it extends the state-of-the-art 2D model MDETR [5], which integrates end-to-end object detection.

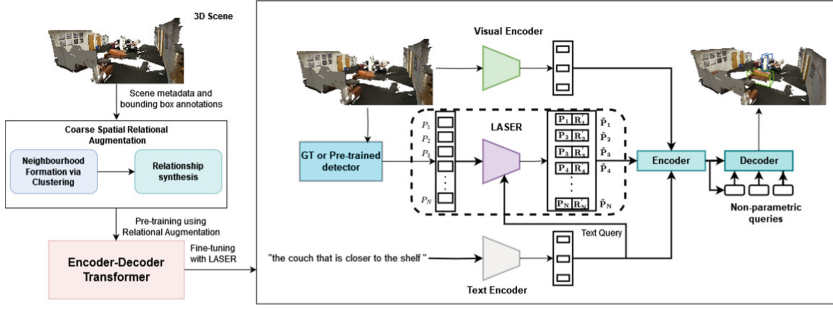


Fig. 2. We propose a novel algorithm that utilizes 3D scenes and their bounding box annotations to synthesize 3D scene-text samples with coarse spatial relationships using the concepts of “near” and “far”. Furthermore, the data, augmented using these synthesized text annotations, holds significant potential for pre-training any multimodal transformer (encoder-decoder) architecture. Additionally, we present a modular approach, that seamlessly integrates into the architecture during fine-tuning for encoding pairwise semantic relationships between the objects conditioned by language query. We call this the LAngeuage-conditioned Spatially aware RElationships method (LASER).

3 Methodology

Our proposed 3D referential grounding method is built atop the popular BUTD-DETR [29], which is based on DETR [6]. It is a transformer-based encoder-decoder architecture that inputs data from three distinct streams: the target scene representation, object bounding boxes (which are, in turn, obtained from the ground truth or using a pre-trained object detector), and query text and outputs a set of bounding boxes that correspond to the objects mentioned in the text query. We first propose a pre-training approach utilizing coarse spatial relations (refer to Sect. 3.1). For this, we create 3D scene-text datasets using publicly available 3D scenes. Furthermore, it is noteworthy that the object bounding boxes are fed independently to the model without any semantic relation information conditioned on the textual utterance, which makes it suboptimal. Therefore, we enhance the semantic relationship between the bounding boxes with a novel Language-Conditioned Spatially Aware Relationship Embedding (LASER), which is used to encode the semantic relations between the object bounding boxes (refer to Sect. 3.2), thereby improving the grounding quality. After incorporating the LASER embeddings, the model is fine-tuned on the target datasets. An overview of the proposed system is shown in Fig. 2, while an overview of LASER is presented in Fig. 3.

3.1 Pre-training with Coarse Spatial Relational Augmentation

In 2D referential grounding, pre-training a model on a large-scale dataset before fine-tuning on the target dataset has been demonstrated to improve model performance on the target datasets [5]. MDETR [5] achieves this by constructing

a large-scale 2D referential grounding dataset of 1.3 million aligned image-text pairs. However, obtaining a similarly sized aligned dataset for the 3D task is considerably more challenging. Therefore, in this work, we propose a novel algorithm for obtaining aligned 3D scene-text pairs without any human supervision.

Coarse Spatial Relational Augmentation. Existing 3D referential grounding models are trained on datasets such as ReferIt3D [1] and ScanRefer [2]. These datasets consist of pairs of 3D point clouds of indoor scenes sourced from ScanNet [7], along with corresponding referential utterances. These referential utterances are generated either synthetically (using rule-based templates) or naturally (through human free-flowing utterances). Obtaining human annotations on these 3D scenes is difficult, and the rule-based synthetic textual utterances are limited. Therefore, in this work, we introduce a novel method (Refer to Algo. 1 for the pseudo-code) for efficiently acquiring aligned 3D scene-text data to facilitate the pre-training of our 3D grounding model. Our approach leverages the observation that coarse spatial relations such as “far” and “near” are dominant among objects in the scene and that learning these coarse spatial relations facilitates the learning of more complex relations such as “to the right of”. Our proposed algorithm can effectively obtain these spatial relations from 3D scenes at scale without the need for human supervision.

A typical 3D scene contains two types of objects: single-occurrence objects (e.g., objects like TV, refrigerator, etc.) and multi-occurrence objects (e.g., chair, stool, etc.). It has been noted that grounding single-occurrence objects is easier than grounding objects with multiple occurrences in the scene where duplicate instances of the same object often act as “distractors”. We capitalize on this insight and present an algorithm. The algorithm is based on a novel neighbourhood formation strategy around an object. It automatically synthesizes textual utterances referring to objects in a 3D scene. This process specifically aids in grounding multi-occurrence objects, thereby improving model performance.

Neighbourhood Formation via Clustering. Let us consider a set S comprising of tuples (o_i, B_i) , where o_i and B_i denote the object’s class and its corresponding bounding box, respectively, representing the set of objects present in a given 3D scene. Additionally, we can divide the set S into two subsets, S_s and S_m , to represent the single-occurrence and multi-occurrence objects, respectively, with their set of corresponding bounding boxes, B_s and B_m . By utilizing the 3D positional information of these 3D bounding boxes (available from the 3D scene information), we cluster them using the K-Means algorithm [8]. Even though the Euclidean distance between the centres of two bounding boxes can serve as a distance metric, it fails to consider the size of the objects, leading to inaccurate neighbourhood clustering. To mitigate this shortcoming, we propose a novel distance function that takes into account the minimum distance between each corner of two bounding boxes.

Relationship Synthesis. Let $C = \{ C_1, C_2, \dots, C_k \}$, be the clusters obtained using the K-Means algorithm. Each cluster may have many single-occurrence and multi-occurrence objects. Among the objects in these clusters, we first define the “near” relationships between the single-occurrence objects in the same cluster. We also create “far” relationships between a multi-occurrence object in a cluster and a single-occurrence object in another cluster. For example, in Fig. 1, “the sink farther and away from the stove”. However, for some object-pairs in the scene, there could be “distractors” (e.g., the second ‘sink’ closer to the stove). In such cases, to better disambiguate such multi-occurrence object, we add another “near” relationship with a single-occurrence object within the same cluster (e.g., “sink alongside the toilet”). The inter-cluster relationship utterance is then created by merging both these relationships (e.g., “select the sink farther and away from the stove but alongside the toilet”). For each of the “far” and “near” spatial concepts, we randomly sample from the set of paraphrased textual relations representing these concepts. For example, we can use {“around the”, “near to the”, “adjacent to the”, “close to the”, “next to the”, and “alongside the”} for the “near” concept and {“away from the”, “farther from”, “farther and away from”, “distant from”, and “not close to the”} for the “far” concept.

Our relational augmentation algorithm is scalable and robust in diverse 3D scenes, as it doesn’t rely on human supervision or heuristics. Instead, it directly leverages object bounding boxes obtained from the 3D scene to estimate clusters of these objects, which are further utilized in an automated rule-based pipeline to define the coarse spatial relations between them. This strategy, therefore, can be easily adapted to different 3D scenes. Specifically, we used 565 unique scenes from the SR3D dataset [1], generating a total of 30,492 scene-text samples with coarse spatial relations, resulting in an average of 54 unique relations per scene. This approach provides an efficient and effective means of grounding natural language queries to 3D scenes without the need for manual intervention or complex heuristic design. Unlike the heuristics-based methods, i.e., SR3D [1], the proposed relational augmentation algorithm synthesizes unique and semantically richer relations without repetitions or relational rephrasing.

3.2 Language-Conditioned Spatially aware Relational (LASER) Embedding

Recent 3D referential grounding models like BUTD-DETR [29] enhance grounding by leveraging detected bounding boxes generated from a pre-trained object detector. The bounding box information includes both semantic (predicted object class) and spatial (positional encoding) information about the detected objects. The model then refines this information with several encoder and decoder layers. Though incorporating the bounding box information improves the grounding performance, it doesn’t encode the semantic relationship information between the objects represented by their bounding boxes. Thus, in this work, we propose **L**anguage-conditioned **S**patially aware **R**elational (LASER) embeddings that encode the language-conditioned semantic relational

information between the object bounding boxes enabling the model to leverage the relationship information between the bounding boxes to learn better grounding.

Algorithm 1 Relational Augmentation Algorithm

```

1: procedure RELATIONALUTTERANCES( $S, B_s, B_m$ )
2:    $C = \text{K-Means}(S, B_s, B_m)$ 
3:    $U = \{\}$ 
4:   for all  $C_i \in C$  do
5:      $U_n = \text{pair-wise NEAR relations between single-occurrence objects in } C_i$ 
6:      $U.\text{extend}(U_n)$ 
7:     for all  $C_j \in C, i \neq j$  do
8:        $U_f = \{\}$ 
9:       for all  $o_{m_j} \in C_j$  do  $\triangleright o_{m_j}$  is multi-occurrence object in  $C_j$ 
10:        if  $\text{distractor}(o_{m_j}) \in C_i$  then
11:          for all  $o_{s_j} \in C_j$  do  $\triangleright o_{s_j}$  is single-occurrence object in  $C_j$ 
12:            for all  $o_{s_i} \in C_i$  do
13:               $rel = o_{m_j} - \text{FAR} - o_{s_i} - \text{BUT NEAR} - o_{s_j}$ 
14:               $U_f.\text{add}(rel)$ 
15:           $U.\text{extend}(U_f)$ 
16:   return  $U$   $\triangleright U$  is the set of synthesised utterances.

```

Let \mathbf{P} denote a set of N bounding boxes, and each of these boxes $\mathbf{P}_i \in \mathbf{P}$ is represented by concatenating its positional feature and a semantic embedding (RoBERTa [27] features of the predicted class) as $\mathbf{P}_i \in \mathbb{R}^d$ (d is the model dimensionality). For each bounding box \mathbf{P}_i , a conditional embedding vector $\mathbf{R}_i \in \mathbb{R}^N$ is computed, where the j^{th} entry in the vector \mathbf{R}_i represents the strength of the relationship between the boxes \mathbf{P}_i and \mathbf{P}_j . Given a pair of bounding boxes \mathbf{P}_i and \mathbf{P}_j and the textual query \mathbf{T} (CLS embedding of text), we first obtain the pairwise box representation as $\mathbf{P}_{i,j} = \mathbf{W}_1 \cdot [\mathbf{P}_i; \mathbf{P}_j]$, where $\mathbf{W}_1 \in \mathbb{R}^{d \times 2d}$. Then the conditional relationship information between \mathbf{P}_i and \mathbf{P}_j is encoded as $\mathbf{R}_{i[j]} = \mathbf{P}_{i,j}^T \mathbf{T}$, where $\mathbf{T} \in \mathbb{R}^d$ is text query corresponding to the natural language utterance. The score function mentioned above assigns a high score to the pair of bounding boxes consistent with the relationships in the textual query. In contrast, low scores are assigned to incompatible pairs. The same process is repeated for all the other boxes to obtain the language-conditioned relationship vector \mathbf{R}_i . The relationship scores for each box are then normalized using the softmax and projected to create box score embeddings $(\mathbf{R}_1, \mathbf{R}_2, \dots, \mathbf{R}_N)$. Further, these box score embeddings are concatenated with the original box features and projected in model dimensionality to obtain a language-conditioned spatially aware relational (LASER) embedding for the bounding box \mathbf{P}_i : $\mathbf{R}'_i = \mathbf{W}_2 \cdot \mathbf{R}_i$, where $\mathbf{W}_2 \in \mathbb{R}^{d_1 \times N}$; d_1 represents the box score embedding dimensionality; $\tilde{\mathbf{P}}_i = \mathbf{W}_3 \cdot [\mathbf{P}_i; \mathbf{R}'_i]$, where $\mathbf{W}_3 \in \mathbb{R}^{d \times (d+d_1)}$. Note that \mathbf{W}_1 , \mathbf{W}_2 , and \mathbf{W}_3 are learnable projection matrices. Sinusoidal positional encodings are added to box

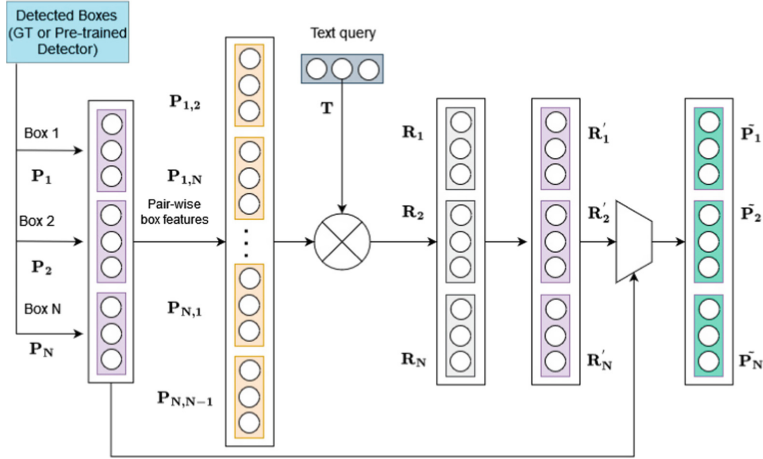


Fig. 3. The LASER receives the input of box features (obtained by concatenating positional features and semantic embeddings; $\mathbf{P}_1, \mathbf{P}_2, \dots, \mathbf{P}_N$) from the set of N bounding boxes obtained from ground truth or pre-trained detector. It then processes these features conditioned on the text query to generate relationship score vectors of bounding boxes ($\mathbf{R}_1, \mathbf{R}_2, \dots, \mathbf{R}_N$). Further, these scores are normalized using the softmax and projected to create box score embeddings ($\mathbf{R}'_1, \mathbf{R}'_2, \dots, \mathbf{R}'_N$). These embeddings are concatenated with the original box features and projected to obtain the LASER embeddings ($\tilde{\mathbf{P}}_1, \tilde{\mathbf{P}}_2, \dots, \tilde{\mathbf{P}}_N$) for the bounding boxes.

features to incorporate ordering information, as these features inherently lack it. This allows the model to differentiate between boxes. The encoded features, along with visual and textual inputs, are then processed by the grounding model to produce the final output.

4 Experiments

The performance is evaluated using three datasets derived from ScanNet [7] dataset: ReferIt3D [1] (SR3D and NR3D) and ScanRefer [2]. SR3D consists of 83,500 template-based scene-text pairs for localizing target objects using one or two reference objects as anchors. NR3D comprises 41,500 pairs of longer, natural-language scene-text samples collected through a 2-player object reference game. ScanRefer consists of about 51,500 pairs of natural, free-form scene-utterances. These datasets evaluate the model’s ability to perform 3D reference grounding tasks across various scenarios and reference expressions. Inspired by [29], we performed the experiments in the two settings: 1) The **GT** setup utilizes ground-truth 3D object boxes with the PointNet++ [24] object categories for both our model and the baseline, and 2) The **det** setup involves using a pre-trained object detector (Group-Free [10]) to obtain an initial set of bounding boxes.

Table 1. Top-1 accuracy results of the model (in the GT setting) for referential grounding on ReferIt3D (SR3D and NR3D) dataset. The model is pre-trained (using relational augmentation (PT(CR)); Sect. 3.1) and fine-tuned (FT) with LASER (Sect. 3.2).

Model	SR3D					NR3D				
	Easy	Hard	View-Dep	View-Ind	Overall (GT)	Easy	Hard	View-Dep	View-Ind	Overall (GT)
ReferIt3DNet [1]	44.7	31.5	39.2	40.8	39.8	43.6	27.9	32.5	37.1	35.6
TGNN [15]	48.5	36.9	45.8	45.0	45.0	44.2	30.6	35.8	38.0	37.3
InstanceRefer [9]	51.1	40.5	45.4	48.1	48.0	46	31.8	34.5	41.9	38.8
3DVG-Transformer [4]	54.2	44.9	44.6	51.7	51.4	48.5	34.8	34.8	43.7	40.8
TransRefer3D [16]	60.5	50.2	49.9	57.7	57.4	48.5	36.0	36.5	44.9	42.1
SAT [17]	61.2	50.0	49.2	58.3	57.9	56.3	42.4	46.9	50.4	49.2
HAM [12]	65.9	54.6	52.5	63.0	62.5	54.3	41.9	41.5	51.4	48.2
MVT [14]	66.9	58.8	58.4	64.7	64.5	61.3	49.1	54.3	55.4	55.1
BUTD-DETR [29]	69.1	59.0	50.1	66.8	66.1	64.4	48.6	47.8	60.0	56.5
BUTD-DETR+PT(CR)+FT+LASER	70.3	63.8	49.7	69.2	68.3	65.2	49.7	48.7	61.0	57.5

4.1 Results on ReferIt3D Dataset

The object bounding boxes for ReferIt3D [1] (SR3D and NR3D) dataset are generated from the ground truth. The model is evaluated using the overall accuracy metric, which calculates the percentage of correctly predicted bounding box matches with the ground truth. The empirical results presented in Table 1 clearly show the efficacy of the proposed method, which achieves significant improvements in performance across all the datasets. Specifically, the proposed approach achieves a new state-of-the-art performance, with an overall improvement of 2.2% on the SR3D dataset. The models’ performance is particularly noteworthy in the case of ‘**Hard**’ and ‘**View-Independent**’ cases in the datasets. The proposed pre-training step aids the model in learning the coarse spatial relationships that are view-independent, which later helps it to better learn the harder relationships between objects. While methods such as MVT [14] utilize scenes with multiple viewpoints as input, leading to better performance in ‘View-Dependent’ cases, their performance deteriorates for other challenging examples, resulting in lower overall performance. Also, processing 3D scenes with many viewpoints considerably increases the computation cost. Therefore, the proposed method offers a more computationally efficient solution that delivers impressive results across various challenging examples.

4.2 Impact of Pre-training on 3D Referential Grounding

We validated the impact of the pre-training with Coarse Spatial Relational Augmentation (PT(CR)) on the state-of-the-art models, namely EDA [13], BUTD-DETR [29] and ViL3DRel [30]. The EDA [13] and BUTD-DETR [29] models were pre-trained using our coarse spatial relational augmentation, then fine-tuned on SR3D. The ViL3DRel [30] employs teacher-student knowledge distillation, training identical architectures on different inputs (teacher: ground truth annotations, student: point clouds). The teacher is pre-trained before the student learns via distillation. We enhance the teacher’s pre-training by incorporating synthesized coarse spatial relations (PT(CR)) into the scene. The results in

Table 2 highlight the impact of the proposed pre-training strategy. We observe a noticeable improvement of **2%** after leveraging the pre-training for the BUTD-DETR model. Similarly, employing pre-training also brings about noticeable accuracy improvement in EDA (0.9%) and ViL3DRel (0.6%).

Table 2. Impact of pre-training strategy on BUTD-DETR [29], EDA [13] , and ViL3DRel [30] on the SR3D dataset. (Baselines retrained for fairness.)

Model	BUTD-DETR	BUTD-DETR +PT(CR)	EDA	EDA+PT(CR)	ViL3DRel	ViL3DRel +PT(CR)
Top-1 Accuracy	66.1	68.1	65.2	66.1	72.9	73.5

Table 3. The table shows pre-training benefits (Top-1 accuracy) with limited fine-tuning annotations. “% Data” means the portion of SR3D data used for fine-tuning. “w/o PT(CR)” denotes no pre-training, while “w PT(CR)” means BUTD-DETR pre-trained with our method, then fine-tuned on varying proportions of SR3D. The pre-training data is the same across all the settings.

% Data	5	10	20	30	100
w/o PT(CR)	40.4	43.5	63.0	64.1	66.1
w PT(CR)	65.6	66.2	66.4	66.7	68.1

4.3 Fine-Tuning Using Limited Data

Pre-training deep models enhances performance in scenarios with limited fine-tuning data. In an experiment, we used a pre-trained model that was initially trained on our synthesized coarse spatial relations and fine-tuned using various proportions of the SR3D dataset. Results in Table 3 demonstrate significantly improved localization performance with pre-training (65.6% w PT(CR) vs. 40.4% w/o PT(CR)), **even when only 5% of the SR3D data is used during fine-tuning**. This suggests that pre-training on coarse spatial relations aids the model in learning more complex features, even with a small amount of training data.

4.4 Impact of LASER Embeddings

We evaluate the impact of LASER embeddings on the 3D referential grounding performance using the ScanRefer dataset [2] in the `det` setting. Here, 3D object bounding boxes are generated by a pre-trained 3D object detector, specifically the Group-Free 3D Object Detector [10]. Our experiments involve incorporating the LASER embedding module into the BUTD-DETR model without pre-training with coarse relations. The evaluation is based on the $\text{Acc}@m\text{IoU}$ metric, measuring the percentage of text utterances with predicted bounding boxes

Table 4. Results for referential grounding on ScanRefer dataset (det setting).

Model	Unique		Multi		Overall	
	Acc@0.25	Acc@0.5	Acc@0.25	Acc@0.5	Acc@0.25	Acc@0.5
ScanRefer [2]	63.0	40.0	28.9	18.2	35.5	22.4
TGNN [15]	68.6	56.8	29.8	23.2	37.4	29.7
InstanceRefer [9]	77.5	66.8	31.3	24.8	40.2	32.9
MVT [14]	77.7	66.5	31.9	25.3	40.8	33.3
SAT [17]	73.2	50.8	37.6	25.2	44.5	30.1
3DVG-Transformer [4]	77.2	58.5	38.4	28.7	45.9	34.5
3DJCG [3]	78.8	61.3	40.1	30.1	47.6	36.1
HAM [12]	79.2	67.9	41.5	34.0	48.8	40.6
BUTD-DETR [29]	84.8	67.6	47.7	35.7	53.3	40.5
BUTD-DETR+LASER	86.8	68.9	47.8	36.6	53.6	41.4

having a 3D IoU overlap with ground truth boxes greater than m . Our findings, detailed in Table 4, showcase an overall accuracy enhancement of 0.3% and 0.9% for Acc@0.25 and 0.5, respectively. These results underscore the significant improvement in localization performance achieved by integrating the proposed LASER embedding with off-the-shelf 3D referential grounding methods, such as BUTD-DETR.

4.5 Ablation Study

We conducted an ablation study on the SR3D dataset to explore the impact of different factors on the model’s performance, and the results are presented in Table 5. We observed that pre-training the model with text queries containing only the class name of the object without any relationships already led to a significant improvement in performance (Refer to PT(cls) in Table 5). Furthermore, instead of just class names, incorporating coarse spatial relationships between

Table 5. Analysis of the impact of different components on the BUTD-DETR model’s performance. The Top-1 accuracy of the model (GT setting) on SR3D is considered for evaluation (Sect. 4.5). The first row shows the accuracy of the baseline model [29]. FT represents the fine-tuning of the pre-trained models.

PT(cls)	PT(CR)	FT	LASER	Easy	Hard	View-Dep	View-Ind	Overall (GT)
				69.1	59	50.1	66.8	66.1
✓		✓		70.1	61	48.1	68.3	67.4
	✓	✓		70.1	63.3	51.9	68.8	68.1
			✓	69.9	62	55.2	68.1	67.5
	✓	✓	✓	70.3	63.8	49.7	69.2	68.3

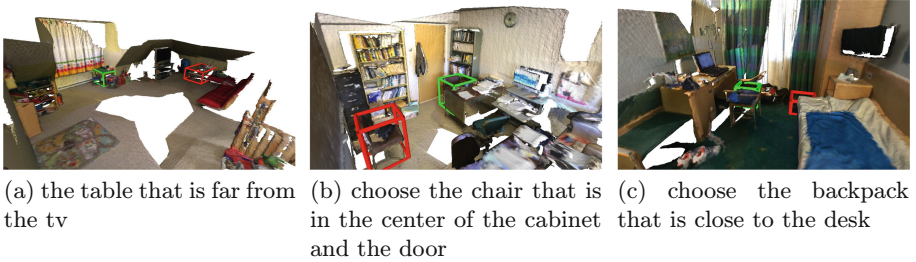


Fig. 4. Qualitative Analysis of our model on SR3D. The bounding box predicted by our model (PT(CR) + FT + LASER) is highlighted in green (which also matches with the ground-truth), and the one predicted by the baseline model [29] is shown in red. We can infer from the above subjective examples our model exhibits a greater ability to differentiate the “distractor” compared to the baseline model. (Color figure online)

the objects in the scene during pre-training helped to enhance the model’s performance (Refer to **PT(CR)** in Table 5). This improvement can be attributed to the fact that during the pre-training phase, the model develops a strong understanding of fundamental concepts such as “near” and “far”, enabling it to grasp more intricate and complex relationships during the subsequent fine-tuning phase (denoted as **FT** in Table 5). Additionally, incorporating language-conditioned relationship features into the bounding boxes’ features allowed the model to better learn the relationships between the objects in the scene, resulting in improved grounding performance (Refer to **LASER** in Table 5). Finally, combining both pre-training and the LASER embedding (Refer to **PT(CR) + FT + LASER** in Table 5) yielded state-of-the-art results with a 2.2% improvement in performance over the baseline model. These results highlight each component’s importance in the model’s overall performance. The pre-training step aids in developing a strong comprehension of fundamental concepts such as spatial relationships, while the LASER embedding method effectively captures language-conditional relationships. Combining all components results in a state-of-the-art model that outperforms the baseline significantly.

4.6 Qualitative Analysis

Figure 4 presents the qualitative analysis of our model’s performance on the SR3D dataset compared to the baseline. The analysis reveals some interesting insights. Firstly, Fig. 4b demonstrates that our proposed model is more proficient in managing longer localizing queries than the baseline model. Moreover, from Fig. 4b it is apparent that the pre-training on coarse spatial relationships helps the model generalize well on complex queries (“in the centre of”). Secondly, the results presented in Fig. 4a, Fig. 4b, and Fig. 4c, highlight our model’s superior ability to handle “distractors” in the scene and precisely localize the desired object, which is an essential aspect of referential grounding. Based on the above examples, we can infer that the baseline model struggles to handle certain chal-

lingering queries and scenes, resulting in lower localizing accuracy. In contrast, our model exhibits a more consistent and robust performance across various scenes and query types, making it a promising approach for the 3D referential grounding task.

5 Conclusion

We presented a novel and effective approach to tackle the challenge of pre-training in 3D referential grounding, where the availability of annotated 3D scene-text pairs is limited. By leveraging the large corpus of 3D scenes with object instance annotations, we design an automated strategy to synthesize scene-text data for pre-training. Our approach creates pairwise and triplet relations between objects in the 3D scene based on their coarse spatial relationships without any human supervision. Additionally, we introduced a novel **LASER** (**L**anguage-conditioned **S**patially aware **R**elational) embeddings method that encodes semantic relationships between objects conditioned on the language utterance. Our pre-training strategy and LASER can enhance the localization results of any transformer architecture. Our approach has achieved remarkable progress in state-of-the-art localization accuracy and outperformed other competitive methods on various publicly available 3D referential grounding datasets.

Acknowledgements. The authors are grateful to the Machine Intelligence and Robotics Center (MINRO) at the International Institute of Information Technology Bangalore (IIITB), for providing the computing resources, and conducive research environment that has been instrumental in completing this work.

References

1. Achlioptas, P., Abdelreheem, A., Xia, F., Elhoseiny, M., Guibas, L.: ReferIt3D: neural listeners for fine-grained 3D object identification in real-world scenes. In: 16th European Conference on Computer Vision (ECCV) (2020)
2. Chen, D., Chang, A., Nießner, M.: ScanRefer: 3D object localization in RGB-D scans using natural language. In: 16th European Conference on Computer Vision (ECCV) (2020)
3. Cai, D., Zhao, L., Zhang, J., Sheng, L., Xu, D.: 3DJCG: a unified framework for joint dense captioning and visual grounding on 3D point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 16464–16473 (2022)
4. Zhao, L., Cai, D., Sheng, L., Xu, D.: 3DVG-transformer: relation modeling for visual grounding on point clouds. In: ICCV, pp. 2928–2937 (2021)
5. Kamath, A., Singh, M., LeCun, Y., Misra, I., Synnaeve, G., Carion, N.: MDETR - modulated detection for end-to-end multi-modal understanding. In: 2021 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 1760–1770 (2021). <https://api.semanticscholar.org/CorpusID:233393962>
6. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-End object detection with transformers. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12346, pp. 213–229. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58452-8_13

7. Dai, A., Chang, A., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: ScanNet: richly-annotated 3D reconstructions of indoor scenes. In: Proceedings Computer Vision and Pattern Recognition (CVPR). IEEE (2017)
8. Jin, X., Han, J.: K-means clustering. In: Encyclopedia Of Machine Learning, pp. 563–564 (2010). https://doi.org/10.1007/978-0-387-30164-8_425
9. Yuan, Z., Yan, X., Liao, Y., Zhang, R., Li, Z., Cui, S.: InstanceRefer: cooperative holistic understanding for visual grounding on point clouds through instance multi-level contextual referring. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 1791–1800 (2021)
10. Liu, Z., Zhang, Z., Cao, Y., Hu, H., Tong, X.: Group-free 3D object detection via transformers. In: 2021 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 2929–2938 (2021)
11. Liu, H., Lin, A., Han, X., Yang, L., Yu, Y., Cui, S.: Refer-it-in-RGBD: a bottom-up approach for 3D visual grounding in RGBD images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6032–6041 (2021)
12. Chen, J., Luo, W., Wei, X., Ma, L., Zhang, W.: HAM: hierarchical attention model with high performance for 3D visual grounding (2022)
13. Wu, Y., Cheng, X., Zhang, R., Cheng, Z., Zhang, J.: EDA: explicit text-decoupling and dense alignment for 3D visual grounding. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2023)
14. Huang, S., Chen, Y., Jia, J., Wang, L.: Multi-view transformer for 3D visual grounding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 15524–15533 (2022)
15. Huang, P., Lee, H., Chen, H., Liu, T.: Text-guided graph neural networks for referring 3D instance segmentation. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp. 1610–1618 (2021). <https://ojs.aaai.org/index.php/AAAI/article/view/16253>
16. He, D., et al.: TransRefer3D: entity-and-relation aware transformer for fine-grained 3D visual grounding. In: Proceedings of the 29th ACM International Conference on Multimedia, pp. 2344–2352 (2021). <https://doi.org/10.1145/3474085.3475397>
17. Yang, Z., Zhang, S., Wang, L., Luo, J.: SAT: 2D semantics assisted training for 3D visual grounding. In: ICCV (2021)
18. Bao, H., Dong, L., Piao, S., Wei, F.: BERT pre-training of image transformers, BEiT (2022)
19. Dai, Z., Cai, B., Lin, Y., Chen, J.: UP-DETR: unsupervised pre-training for object detection with transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1601–1610 (2021)
20. Lu, Z., Pei, Y., Wang, G., Yang, Y., Wang, Z., Shen, H.: ScanERU: interactive 3D visual grounding based on embodied reference understanding (2023)
21. Rao, J., Bian, H., Xu, X., Chen, J.: Autonomous visual navigation system based on a single camera for floor-sweeping robot. Appl. Sci. **13** (2023). <https://www.mdpi.com/2076-3417/13/3/1562>
22. Xia, F., Zamir, A.R., He, Z., Sax, A., Malik, J., Savarese, S.: Gibson env: real-world perception for embodied agents. In: 2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
23. Szot, A., et al.: Habitat 2.0: training home assistants to rearrange their habitat. In: Advances in Neural Information Processing Systems (NeurIPS) (2021)
24. Qi, C., Yi, L., Su, H., Guibas, L.: PointNet++: deep hierarchical feature learning on point sets in a metric space. In: Proceedings of the 31st International Conference on Neural Information Processing Systems, pp. 5105–5114 (2017)

25. Yin, P., Xu, L., Ji, J., Scherer, S., Choset, H.: 3D segmentation learning from sparse annotations and hierarchical descriptors (2021)
26. Behley, J., et al.: SemanticKITTI: a dataset for semantic scene understanding of LiDAR sequences. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2019)
27. Liu, Y., et al.: A robustly optimized BERT pretraining approach, RoBERTa (2019)
28. Liu, S., et al.: Dual query detection transformer for phrase extraction and grounding, DQ-DETR (2022)
29. Jain, A., Gkanatsios, N., Mediratta, I., Fragkiadaki, K.: Bottom up top down detection transformers for language grounding in images and point clouds. In: Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXVI, pp. 417–433 (2022). https://doi.org/10.1007/978-3-031-20059-5_24
30. Chen, S., Guhur, P., Tapaswi, M., Schmid, C., Laptev, I.: Language conditioned spatial relation reasoning for 3D object grounding. In: NeurIPS (2022)



PointGADM: Geometry Acquainted Deep Model for 3D Point Cloud Analysis

Seema Kumari¹(✉), Samay Kalpesh Patel², Raja Muthalagu²,
and Shanmuganathan Raman¹

¹ Indian Institute of Technology, Gandhinagar, India
{seema.kumari, shanmuga}@iitgn.ac.in

² BITS-Pilani, Dubai, UAE

{f20200064, raja.m}@dubai.bits-pilani.ac.in

Abstract. Deep learning-based approaches have shown great achievement in 3D point analysis. Due to the irregular and unordered data structure, point cloud analysis is still very challenging. Most existing work uses the convolution, graph, or attention mechanism to achieve the 3D geometry of the target shape. Only a few approaches consider global and local geometry information of point clouds. However, both kinds of geometry play a significant role in analysis. This paper proposes a geometry-acquainted fusion (GAF) module that considers global-to-local geometry information by multi-step processing. Further, we consider in-plane and out-plane distances to capture the geometrical information in the raw point cloud. The modules are utilized in two different architectures, devised for classification and segmentation. The classification network is a simple feed-forward architecture, whereas the segmentation network is developed based on a U-Net-like architecture with residual connections. We show that the proposed architectures perform quite well compared to the state-of-the-art methods in classification and segmentation tasks.

Keywords: 3D point cloud · geometry · fusion · classification · segmentation

1 Introduction

Point cloud analysis is a critical task in 3D computer vision, driving advancements in augmented reality, robotics, autonomous driving, and more [21]. A point cloud is a collection of data points defined in a three-dimensional space, typically generated by 3D scanners or LiDAR systems. Each point in the cloud represents a precise location on the surface of an object or scene, often accompanied by additional attributes such as colour, intensity, or normal vectors. Point clouds'

The work is supported by Jibaben P. Chair in Artificial Intelligence, IIT Gandhinagar, India.

Seema K. and Samay K. P.—Equal Contributions.

inherent complexity and irregularity present significant challenges for analysis, such as classification, segmentation, etc. Unlike 2D images with a regular grid structure, point clouds are unordered and have varying point densities. It necessitates the development of sophisticated processing techniques that can effectively handle the unique properties of point clouds. Given the irregular structure of point clouds, extracting meaningful features is crucial.

Traditional methods have relied on handcrafted features that capture points' geometric and statistical properties to address these issues. However, recent advancements leverage deep learning techniques to automatically learn features from the raw data. Methods like PointNet [23] and its variants have revolutionized feature extraction by effectively handling the unordered and irregular nature of points. Some existing approaches transform the point clouds into irregular structures, e.g., voxelization methods [19, 45] and projection to multi-view images [3]. Most of the existing CNN-based networks are designed to preserve the structure of the target object in point space [2, 15, 31, 38, 41, 42]. Each method has strengths and trade-offs in preserving geometric details, computational efficiency, and scalability. Some limitations of these convolutional-based approaches include difficulty capturing the long-range dependencies [32]. Transformer [33] based methods propose to address these issues. Initially, the transformer-based method was designed for natural language and image processing areas [6, 17, 35]; later on, it was demonstrated in various areas. Architectures such as Dynamic Graph CNN (DGCNN) [36] and Point Transformer [44] further enhance classification accuracy by capturing local and global contextual information.

Geometry information, local or global, is crucial for point cloud analysis. However, only a few methods consider both kinds of geometry. In this paper, we propose a geometry-acquainted fusion module (GAF) that incorporates local-to-global geometry information of point clouds into the framework. The GAF module considers information from two spaces: metric and feature. In metric space, the point features adaptively transform in a local region using the geometric affine module. Self-attention is the critical component in feature space to highlight important features. Fusing metric and feature space information creates a suitable amalgamation for point cloud analysis. The GAF module not only enhances the overall accuracy of point cloud classification but also provides a flexible framework that can be tailored to specific application requirements. By systematically addressing the challenges at each stage, this approach enables more effective utilization of point cloud data, paving the way for advancements in various applications of 3D computer vision.

Further, we consider in-plane and out-plane distances of raw 3D point clouds to generate the initial features. The in-plane distance measures the closeness of points in a plane, whereas the out-plane distance estimates the distance between a plane and the points. Both of these distances incorporate geometry information at the raw level. Hence, our method considers geometry information at various levels.

We design two different architectures utilizing the GAF module and in-plane and out-plane distances for classification and segmentation tasks. The classifi-

cation network is a relatively simple feed-forward network with the mentioned modules. We design a U-Net-based architecture with GAF and distance modules for segmentation. Along with global-to-local geometry information, the residual connections in U-Net enable smoother information flow, thus capacitating spatial data preservation, which is crucial for segmentation. We demonstrate the effectiveness of our proposed GAF and distance modules in classification as well as segmentation tasks using three datasets: ModelNet40 [39], Scanobjects [32] and SN-part [37]. The first two datasets are used for classification, whereas segmentation results are shown using the third dataset. Our models perform better than the existing recent approaches.

The main contributions of our work can be summarized as follows.

- We propose a geometry acquainted fusion (GAF) module in the deep learning architectures to incorporate local to global geometry information in the learning process that aids in classification and segmentation in point space. GAF considers the geometric affine transformation of the local neighbour point features and a target point feature self-attention mechanism.
- We propose to utilize the in-plane distance between points in a plane and the out-plane distance between a point and a reference plane to include the raw geometry information, facilitating better point cloud analysis.
- Incorporating GAF and in-plane and out-plane distance modules in a U-Net-like module improves the parts segmentation results. In contrast, the combination facilitates the effectiveness of a simple network in classification tasks.
- Quantitative and qualitative analysis shows the superiority of our models in the object classification and part segmentation of point cloud.

2 Related Work

We choose classification and segmentation to demonstrate the effectiveness of the proposed method. Hence, in this section, we discuss some of the works related to those applications.

Point cloud classification has become essential in 3D computer vision, enabling applications in various fields. Early approaches relied on handcrafted features and classical machine-learning algorithms. These methods focused on extracting the geometric and statistical properties of the points. Spin Images [13] method projects the local neighbourhood of each point into a 2D histogram, capturing surface properties to facilitate recognition. 3D Shape Contexts [9] inspired by 2D shape contexts, this method extends the idea to 3D, creating histograms that represent the spatial distribution of points around a reference point. Further, Point Feature Histograms (PFH) [30] compute geometric relationships between points in a local neighbourhood. These are then used as input features for classification algorithms like SVMs or Random Forests.

In deep learning-based approaches, Graph-based methods model point clouds as graphs, where points are nodes and edges represent relationships between

points. Dynamic Graph CNN (DGCNN) [36] constructs a k-nearest neighbour graph dynamically during the learning process, allowing the network to learn both local and global structures adaptively. EdgeConv, a core operation in DGCNN, dynamically updates graph edges based on feature space distances, which improves the model’s capacity to grasp local geometric nuances. In addition, voxel-based techniques transform point clouds into structured 3D grids known as voxels and utilize 3D Convolutional Neural Networks (CNNs). VoxNet [19] transform 3D points into a 3D occupancy grid and applies 3D CNNs for feature extraction and classification. OctNet [29] uses an octree-based representation to hierarchically partition the space, making the approach more memory-efficient and scalable. However, Multi-view approaches convert 3D points into several 2D projections and apply 2D CNNs. Multi-View CNNs [8] combine features from different 2D perspectives of a 3D object, achieving cutting-edge performance by taking advantage of 2D CNN strengths.

Further, PointNet [23] revolutionized point cloud processing by directly consuming raw point clouds without requiring handcrafted features. This approach processes each point independently using shared MLPs (Multi-Layer Perceptrons) and then aggregates these features using a symmetric function (e.g., max pooling). This approach preserves permutation invariance and captures the global context effectively. Instead of focusing on local information, it blindly processes the points. To address this issue, an extension of PointNet, PointNet++ [24], hierarchically applies PointNet to local regions of the point cloud, capturing local geometric features more effectively. To extend more understanding in local contexts, some recent approaches [2, 15, 31, 38, 41, 42] have proposed explicit convolution kernels on the point cloud. However, KPConv [31] has addressed this with deformable convolution [46] to obtain the local information from the point space. Further, the improved version of PointNet++, PointNeXt [25], has been proposed as an enhanced training and augmentation scheme.

Inspired by their success in natural language processing and image recognition, recent advances in transformers have been adapted for point cloud processing [5, 20, 28, 40, 44]. Point transformer [44], this method uses self-attention mechanisms to model dependencies between points, capturing both local and global contexts effectively. Cloud transformer [20] proposes an attention mechanism that transforms the point cloud into a voxel grid for convolutional operation. From handcrafted features to deep learning-based methods, point cloud classification has significantly improved. PointNet and its variants, graph-based methods, voxel-based approaches, multi-view strategies, and the recent incorporation of transformers and self-supervised learning techniques have collectively pushed the state-of-the-art. The continuous development in this area promises further improvements in accuracy, efficiency, and application scope. This paper proposes a geometry-acquainted fusion (GAF) module to obtain local and global geometrical information for the point cloud analysis tasks.

3 Proposed Approach

MLP-based methods [23] learn feature maps f directly from a given set of points ($P = (\{p_i | i = 1, \dots, N\}) \in R^{N \times 3}$, where N denotes the number of points in (x, y, z) Cartesian space. However, in this method, we follow the farthest point sampling method (FPS) as proposed by PointNet++ [24] to select a subset of M points from N so that the selected point has the largest distance as compared to the rest of the points. In addition to FPS, PointNet++ creates groups of sampled points with the help of neighbouring points. K -neighbors are considered for each sampled point to capture local structure, aggregated by max-pooling. The operations can be summarized through

$$z_i = A(\Phi(f_{i,j}) | j = 1, \dots, K). \quad (1)$$

Here, $A(\cdot)$ and $\phi(\cdot)$ denote the max-pooling and MLP functions. $f_{i,j}$ denotes the j -th neighbor point feature of i -th sampled point. This method can progressively enlarge the receptive fields and capture local geometric information by repeating the operation. PointNet++ represents a universal pipeline for point cloud processing in network architecture.

Further, some methods following PointNet++ mainly focus on the local feature extractors $\phi(\cdot)$ [16, 31, 41, 44]. Local feature extractors bring out local geometric information using these learning based methods. However, in the RSCNN approach, these extractors are obtained from the point relations. The point transformer method considers the similarities between pairwise points in a local region. While these existing methods perform better in obtaining promising results, their development has some limitations. These limitations are a requirement of memory access, which increases the computational cost, and the performance of these methods started saturating on popular benchmarks.

Our method uses two-step processing to get in-depth information about the target object in point space. In the first step, we compute the in-plane and out-plane distances on raw points to get the geometric cues and project to the shared MLP block to get the initial features. The second step contains the fusion block, specifically the geometry acquainted fusion (GAF) block that we apply twice to get the deeper features for multi-scale processing. The in-plane and out-plane distances and GAF blocks are the key modules of our classification and segmentation models. For classification, we consider a simple feed-forward model, whereas, for segmentation, we adopt a U-Net-like architecture with MLP-based residual blocks to incorporate the local and global information of the target object.

We first discuss our proposed geometry acquainted fusion (GAF) block, followed by in-plane and out-plane distance calculation. Later, we present their significance in point cloud classification and part-segmentation by designing different architectures.

3.1 Geometry-Acquainted Fusion (GAF) Module

We introduce a geometry-acquainted fusion (GAF) module (shown in Fig. 1) that hierarchically processes the point’s features and generates deeper representations. The fusion block processes information in two spaces: metric space and feature space.

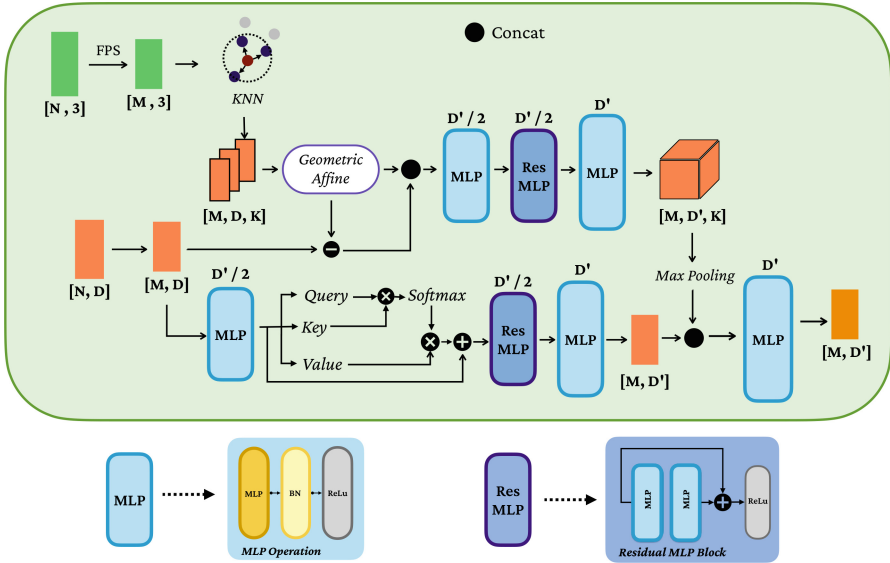


Fig. 1. Block diagram of our geometry acquainted fusion (GAF) module.

Metric Space Processing. Given points $P_i(N, 3)$ and their point features $F_i(N, D)$, the points are down-sampled to $p_i(M, 3)$ using FPS technique (stated earlier). The features are grouped based on the K nearest neighbours in the metric space, denoted as $f_{i,j}(M, D, K)$. $f_{i,j}$ is the j -th neighbor point feature of the i -th sampled point. A combination of MLP, residual MLP, and a second MLP block (Φ) learns shared weights for local regions, extracting the local features and a geometric affine (GA) module [18] handles geometric structure variations. Finally, the features are aggregated using max-pooling. The operation is defined as:

$$z_i = A(\Phi[(GA(f_i, f_{i,j})), f_i - \{f_{i,j}\}] | j = 1, \dots, K) \tag{2}$$

Here, $A(\cdot)$ represents the max pooling operation. $[\cdot]$ denotes concatenation operation. $GA(f_i, f_{i,j})$ is computed as [18]

$$GA(f_i, f_{i,j}) = \{f_{i,j}\} = \alpha \odot \frac{\{f_{i,j}\} - f_i}{\sigma + \epsilon} + \beta \tag{3}$$

where α and β are trainable parameters, and σ is defined as

$$\sigma = \sqrt{\frac{1}{K \times n \times D} \sum_{i=1}^n \sum_{j=1}^K (f_{i,j} - f_i)^2}. \quad (4)$$

Feature Space Processing. We apply a self-attention mechanism directly to the features, capturing long-range and short-range interactions between features without complex geometric operations. This process can be defined as:

$$v_i = \Phi(\text{selfattention}(f_i)) \quad (5)$$

where the self-attention is computed as [33]

$$\text{selfattention}(f_i) = \text{attention}(Q, K, V) \approx \text{softmax}(QK^T)V, \quad (6)$$

where Q , K , and V are the query, key, and value computed by multiplying the feature matrix F with learned weight matrices. The output of self-attention is further added to the input feature and passed through residual MLP and MLP block.

Information Fusion. The metric and feature space information is fused by concatenating the outputs z_i and v_i . This combined representation undergoes another non-linear transformation to generate deeper features in a high-level space. It is defined as:

$$u_i = \Phi_{\text{fusion}}([z_i, v_i]) \quad (7)$$

where, z_i is the output from the metric space processing, v_i is the output from the feature space processing. $[z_i, v_i]$ denotes the concatenation of z_i and v_i . Φ_{fusion} is a non-linear transformation function, a residual MLP.

3.2 In-Plane and Out-Plane Distance

We propose a geometric descriptor instead of applying MLP only to raw points. Traditional handcrafted 3D descriptors inspired the inclusion of the in-plane and out-plane distance to form the geometric cues. In point cloud analysis, understanding the spatial relationships between the points is essential. In-plane and out-plane distances are fundamental concepts for this task.

The in-plane distance used to compute the distance between points within the same surface is vital to maintaining surface smoothness and continuity. However, the out-plane distance computes the deviation of a point from a reference plane, which indicates how far a point is from the expected plane surface. This computation is essential in detecting surface irregularities. The in-plane distance $d_{\text{in-plane}}$ between two points p_i and p_j can be computed using the Euclidean distance constrained to a plane:

$$d_{\text{in-plane}}(p_i, p_j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (8)$$

The out-plane distance $d_{\text{out-plane}}$ between two points p_i and p_j is defined as:

$$d_{\text{out-plane}}(p_i, p_j) = |z_i - z_j| \tag{9}$$

For a given point set P_i , we first calculate the in-plane and out-plane distances concatenated with the point set followed by shared MLP. The MLP block projects the concatenated information into a higher dimensional space to produce the initial point features F_i .

3.3 Classification

Here, we discuss the proposed architecture (shown in Fig. 2) for classification using the GAF, in-plane, and out-plane distance modules. The input 3D point is concatenated with the in-plane and out-plane distances to form a 5D vector with geometrical information for each of the N points. These vectors are projected to a space of 64 dimensions using shared MLP to produce the initial feature, which is then processed through two GAF blocks. The first block reduces the number of features by factor 4 but increases the feature dimension to 256, whereas the second block narrows down the number of features by factor 2 with a surge in feature dimension to 512. Reducing the number of features enables the network to capture global information from the point cloud. Moreover, a fusion of metric space and feature space allows the architecture to learn rich details on the target point cloud. In the fusion process, $K = 20$ neighbors are aggregated for each sampled point to extract high-level geometric properties. The 512-dimensional features are further mapped to 1024-dimensional features through shared MLP. Afterward, the max-pooling helps to pluck out the significant information in the form of a global feature vector from the pool of $N/8$ point features. The global feature vector is then passed through the classification head, consisting of five fully connected (512, 256, 128, 64, c) layers to predict the classification score. A couple of drop-out layers with 0.5 probability are employed to address the over-fitting issue.

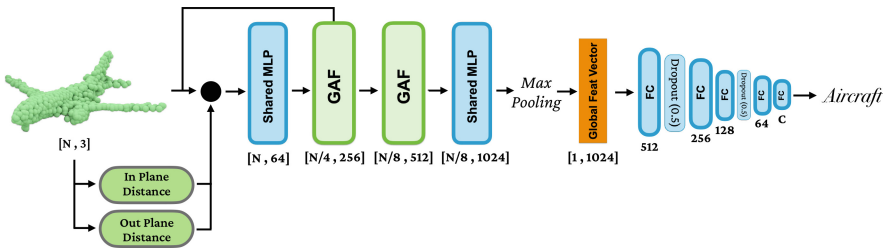


Fig. 2. Architecture of point cloud classification.

3.4 Segmentation

In part segmentation, we incorporate the GAF blocks and the in-plane, out-plane distance modules in a UNet-like architecture (See Fig. 3). The initial processing of points is similar to the classification task. The 3D points are appended with two distances to incorporate geometry information at the initial level. The 5D vector for each of the N points is passed through the shared MLP to generate 64 dimensional N features. The feature is then passed through a couple of GAF blocks to generate $N/4$ features with 128 dimensions and $N/8$ features with 256 dimensions, respectively. The features generated at each step are concatenated with the GAF modules present at the output side through skip connections. Skip connections play an important role in enabling the smoother flow of information. The number of points is interpolated to match with the input number [18]. The final interpolated feature is passed through the shared MLP block to achieve the part segmentation result.

We employ the cross-entropy loss to optimize the proposed model; it computes the loss between the ground truth and prediction for both the analysis tasks to achieve the best results.

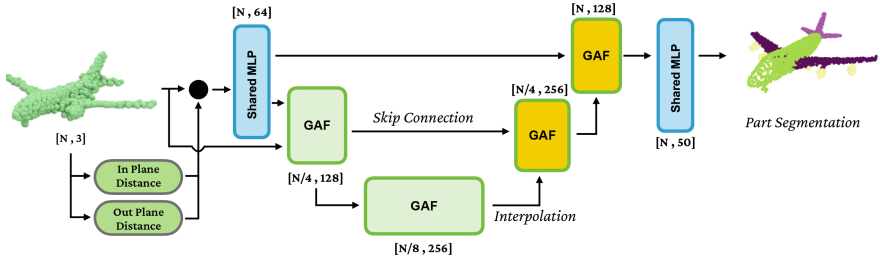


Fig. 3. Architecture of point cloud segmentation.

4 Experimentation

Here, we present the implementation details of our proposed architecture, including parameter values and hardware configuration. Additionally, we demonstrate the effectiveness of our approach for classification and segmentation by comparing our results with state-of-the-art techniques. We also conduct ablation experiments to analyze our architecture.

4.1 Implementation Details

We utilize the PyTorch framework to develop our classification and segmentation architectures. Both models are trained and tested on an NVIDIA GeForce RTX 2080 GPU. Stochastic Gradient Descent (SGD) is employed for optimization

with momentum (0.9) and weight decay (0.0001). The learning rate is initially set to 0.1 and is adjusted using a cosine annealing scheduler every 100 epochs. Our model undergoes training for 300 epochs with 32 batch size. In the classification task, we use $K = 20$ neighbours, whereas, for segmentation, we choose $K = 24$, neighbors, as this additional local information aids in accurately segmenting the object.

4.2 Object Classification

We evaluate our network performance for the classification task on a real-world dataset ModelNet40 [39] and ScanObjectNN(SONN) [32]. ModelNet40 dataset consists of 12,311 CAD models from 40 artificial scene categories, which we divide into 9,843 for training and 2,468 for testing similar to PointNet [23]. The SONN [32] dataset contains 2,902 objects that are indices into 15 classes from SceneNN [11] and ScanNet [7]. We follow the standard split for training (80%) and testing (20%) as has been used in [21]. We sample 1024 points for the training and testing to evaluate our model. Moreover, we consider the hardest perturbed variant (PB_T50_RS) to scrutinize the robustness of our model.

Table 1. Comparison of object classification results with existing method on ScanObjectNN Dataset.

Methods	Venue	mAcc	OA
PointNet [23]	CVPR	63.4	68.2
PointNet++ [24]	NIPS	75.4	77.9
SpiderCNN [42]	ECCV	69.8	73.7
PointCNN [15]	NIPS	75.1	78.5
DGCNN [36]	TOG	73.6	78.1
DRNet [26]	CVPR	78	80.3
GBNet [27]	TOM	77.8	80.5
PointNet + SageMix [14]	NIPS	-	66.1
PRA-Net [4]	TIP	77.9	81
OcCo [34]	ICCV	-	78.3
CrossPoint [1]	CVPR	-	81.7
MVTN [10]	ICCV	-	82.8
PointMLP-elite [18]	ICLR	81.8±0.8	83.8±0.6
PointGADM (ours)	-	82.5	83.8±0.5

We compare our results with the state-of-the-art methods. We use two evaluation metrics for comparison: mAcc (mean Accuracy) and OA (Overall Accuracy). The quantitative results are presented in Tables 1 and 2. In Table 1, we show object classification results on the ScanObjectNN dataset. One can note that

our method outperforms the existing baseline (CrossPoint [1] and MVTN [10]) approaches by 2.1 and 1.0 absolute percentage points. Our method performs better than the existing approaches with fewer parameters, 3.353 M and 3.4G FLOPs, compared to most existing methods (KPConv, PPointMLP, GBNet, etc.). It makes the network lighter, so the training and testing can be done quickly. The t-SNE plot reveals distinct clusters corresponding to 14 classes on the ScanobjectNN dataset, visualized in Fig. 4. The separation among the clusters suggests that the features used for classification are quite suitable.

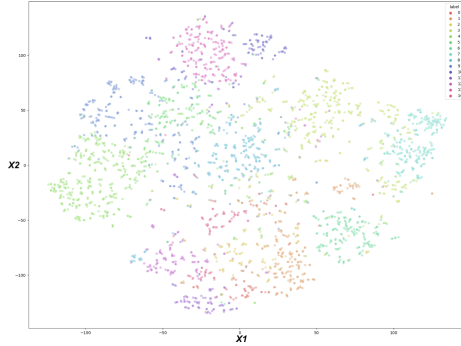


Fig. 4. t-SNE plot for classification task on ScanObjectNN Dataset.

Table 2. Comparison object Classification Results with state-of-the-art methods on ModelNet40 Dataset.

Methods	Venue	mAcc	OA
3DShapeNets [39]	CVPR	77.3	84.7
PointNet [23]	CVPR	86.0	89.2
PointNet++ [24]	NIPS	88.2	91.9
Perceiver [12]	ICML	-	85.7
PATs [43]	CVPR	88.6	91.7
PointNet + SageMix [14]	NIPS	79.5	90.3
OcCo [34]	ICCV	-	89.2
CrossPoint [1]	CVPR	-	91.2
CrossMoCo [22]	CRV	-	91.4
PointGADM (ours)	-	87.5	91.4

The results of our method and existing approaches on the ModelNet40 dataset are tabulated in Table 2 for comparison. Our model can classify the

point clouds with mAcc 87.5% and OA 91.4%. It can be noted that our model can produce competitive results compared to the existing methods such as Cross-Point [1] and CrossMoCo [22].

4.3 Part Segmentation

We evaluate our model on SN-part [37] dataset for part segmentation. This synthetic dataset contains 16,881 shapes from 16 categories with 50 part labels. For validation, we follow the same split given in [21], where 14,006 and 2,874 samples are considered for training and testing, respectively. We randomly sample each shape with 2048 points. Our lightweight segmentation model has just 0.856M parameters and 1.3G FLOPS. We use the mIoU (category-wise mean Intersection over Union) and instance-wise mIoU metrics to estimate our model and compare it with existing methods in Table 3.

Table 3. Part segmentation results compared with previously proposed approaches on ShapeNetPart dataset.

Method	Year	Cls. mIoU	Inst. mIoU	aero	bag	cap	car	chair	aerp- hone	guitar	knife	lamp	laptop	motor bike	mug	pistol	rocket	skate board	table
PointNet	2017	80.4	83.7	83.4	78.7	82.5	74.9	89.6	73	91.5	85.9	80.8	95.3	65.2	93	81.2	57.9	72.8	80.6
PointNet++	2017	81.9	85.1	82.4	79	87.7	77.3	90.8	71.8	91	85.9	83.7	95.3	71.6	94.1	81.3	58.7	76.4	82.6
PointCNN	2018	84.6	86.1	84.1	86.5	86	80.8	90.6	79.7	92.3	88.4	85.3	96.1	77.2	95.2	84.2	64.2	80	83
SpiderCNN	2018	82.4	85.3	83.5	81	87.2	77.5	90.7	76.8	91.1	87.3	83.3	95.8	70.2	93.5	82.7	59.7	75.8	82.8
DGCNN	2019	82.3	85.2	84	83.4	86.7	77.8	90.6	74.7	91.2	87.5	82.8	95.7	66.3	94.9	81.1	63.5	74.5	82.6
DRNet	2020	-	86.4	84.3	85	88.3	79.5	91.2	79.3	91.8	89	85.2	95.7	72.2	94.2	82	60.6	76.8	84.2
PointASNL	2020	-	86.1	84.1	84.7	87.9	79.7	92.2	73.7	91	87.2	84.2	95.8	74.4	95.2	81	63	76.3	83.2
CurveNet	2021	-	86.8	85.1	84.1	89.4	80.8	91.9	75.2	91.8	88.7	86.3	96.3	72.8	95.4	82.7	59.8	78.5	84.1
DeltaConv	2021	-	86.9	85.3	88.1	88.6	81.4	91.8	78.4	92	89.3	85.6	96.1	76.4	95.9	82.7	65	76.6	84.1
KPConv	2021	85.1	86.4	84.6	86.3	87.2	81.1	91.1	77.8	92.6	88.4	82.7	96.2	78.1	95.8	85.4	69.0	82.0	83.6
Paconv	2021	84.3	85.0	90.4	79.7	87.5	80.54	90.6	80.8	92.0	88.7	82.2	95.9	73.9	94.7	84.7	65.9	81.4	84.0
PointMLP	2022	84.6	86.1	83.5	83.4	87.5	80.54	90.3	78.2	92.2	88.1	82.6	96.2	77.5	95.8	85.4	64.6	83.3	84.3
PointMLP+TAP	2023	85.2	86.9	84.8	86.1	89.5	82.5	92.1	75.9	92.3	88.7	85.6	96.5	79.8	96	85.9	66.2	78.1	83.2
SPoTr	2023	85.4	87.2	85.8	86.9	89.3	82.2	92	82.4	91.8	88.6	85.7	96.2	77.6	96.3	85.3	64	78	84.1
PointGADM	-	83.56	87.3	83.81	91.5	74.68	84.3	89.3	76.56	92.6	83.46	81.6	94	73.64	96.07	89.12	50	86.41	90

Our proposed architecture performs better as compared to the baseline methods KPConv [31], Paconv [41], PointMLP [18] and SPoTr [21] with respect to category-wise mIoU and instance-wise mIoU. The qualitative results of part segmentation can be visualized in Fig 5. It can be observed that the results predicted using our proposed approach are close to the ground truth. The quantitative and qualitative results highlight the effectiveness of our model.

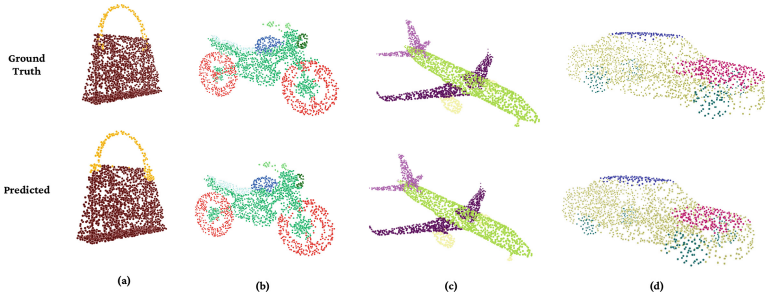


Fig. 5. Visual results of part segmentation on ShapeNetPart Dataset.

4.4 Ablation Study

Here, we demonstrate ablation experiments to show the importance of the proposed network depth, which involves systematically removing or altering components and observing the changes in model performance. Proposed architectures consist of two main components: geometry acquainted fusion (GAF) block and distance functions (in-plane and out-plane), typical for classification and segmentation networks. For the segmentation network, we study the role of the residual connection.

GAF. First, we analyze the importance of the GAF block in our network by training and testing without one of the mentioned blocks. The classification accuracy is dropped from $83.8 \pm 0.5\%$ to 79.1% . It shows the significance of hierarchical processing, which benefits from learning different scales and extracting useful information at various embedding spaces.

Distance Functions. Further, we show the importance of distance functions in our model. These distance functions apply directly to the given input without processing, as shown in the architecture of our method in Figs. 2,3. When we train our network without these distance functions, classification accuracy is degraded from $83.8 \pm 0.5\%$ to 82.9% . One can conclude that the distance functions improve the classification accuracy approximately by 1.0% . Hence, in-plane and out-plane distances are essential in our proposed model.

Residual Connection. In the segmentation architecture, residual connections are incorporated into the interpolation path to ease the gradient flow through the skip connection.

To test this hypothesis, we remove the residual connection from the baseline while providing the geometric cues and plot the loss landscape of the model in Fig. 6. One can observe an apparent decrease in loss in residual connection. The plot starts from an intermediate epoch to show the divergence clearly. Moreover, the instance mIoU accuracy decreases by 1.0% without the residual connection. It demonstrates the importance of residual connection for segmentation tasks. This block not only improves segmentation accuracy but is also helpful in making the network stable and robust.

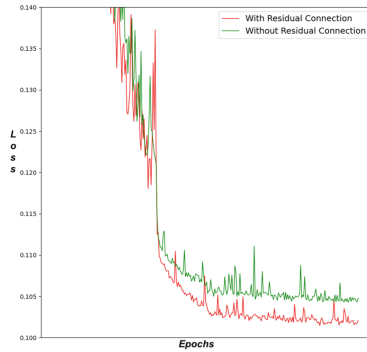


Fig. 6. Effectiveness of the residual connections in the segmentation task.

5 Summary

We propose a geometry-acquainted fusion (GAF) module that enables the network to learn local-to-global levels of geometrical information, which is quite useful for 3D point cloud analysis. The module enables metric space and feature space interaction. In metric space, geometry affine transformation of features is the key operation, whereas in the feature space, self-attention is the main activity. Further, we propose to use in-plane and out-plane distances to capture the geometry of the raw point cloud. The conjugation of both the geometry-aware modules significantly advances point cloud analysis tasks. The classification architecture is simple yet quite effective in classifying different point clouds. A U-Net-like architecture with residual connections is used for segmentation tasks. Our proposed model has fewer parameters and less time complexity. The proposed models perform quite well as compared to the state-of-the-art techniques.

References

1. Afham, M., Dissanayake, I., Dissanayake, D., Dharmasiri, A., Thilakarathna, K., Rodrigo, R.: Crosspoint: self-supervised cross-modal contrastive learning for 3D point cloud understanding. In: CVPR, pp. 9902–9912 (2022)
2. Atzmon, M., Maron, H., Lipman, Y.: Point convolutional neural networks by extension operators. arXiv preprint [arXiv:1803.10091](https://arxiv.org/abs/1803.10091) (2018)
3. Chen, X., Ma, H., Wan, J., Li, B., Xia, T.: Multi-view 3D object detection network for autonomous driving. In: CVPR, pp. 1907–1915 (2017)
4. Cheng, S., Chen, X., He, X., Liu, Z., Bai, X.: Pra-net: point relation-aware network for 3D point cloud analysis. *IEEE Trans. Image Process.* **30**, 4436–4448 (2021)
5. Choe, J., Park, C., Rameau, F., Park, J., Kweon, I.S.: PointMixer: MLP-mixer for point cloud understanding. In: European Conference on Computer Vision, pp. 620–640. Springer (2022)
6. Chu, X., et al.: Twins: revisiting the design of spatial attention in vision transformers. *Adv. Neural. Inf. Process. Syst.* **34**, 9355–9366 (2021)

7. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: ScanNet: richly-annotated 3D reconstructions of indoor scenes. In: CVPR, pp. 5828–5839 (2017)
8. Feynman, R., Vernon, F.: Multi-view convolutional neural networks for 3D shape recognition. *Ann. Phys.* **24**, 118–173 (1963)
9. Frome, A., Huber, D., Kolluri, R., Bülow, T., Malik, J.: Recognizing objects in range data using regional point descriptors. In: ECCV, pp. 224–237. Springer (2004)
10. Hamdi, A., Giancola, S., Ghanem, B.: MVTN: multi-view transformation network for 3D shape recognition. In: CVPR, pp. 1–11 (2021)
11. Hua, B.S., Pham, Q.H., Nguyen, D.T., Tran, M.K., Yu, L.F., Yeung, S.K.: SceneNN: a scene meshes dataset with annotations. In: International Conference on 3DV, pp. 92–101. IEEE (2016)
12. Jaegle, A., Gimeno, F., Brock, A., Vinyals, O., Zisserman, A., Carreira, J.: Perceiver: general perception with iterative attention. In: International Conference on Machine Learning, pp. 4651–4664. PMLR (2021)
13. Johnson, A.E., Hebert, M.: Using spin images for efficient object recognition in cluttered 3D scenes. *PAMI* **21**(5), 433–449 (1999)
14. Lee, S., Jeon, M., Kim, I., Xiong, Y., Kim, H.J.: SageMix: saliency-guided MixUp for point clouds. In: Advances in Neural Information Processing Systems (2022)
15. Li, Y., Bu, R., Sun, M., Wu, W., Di, X., Chen, B.: PointCNN: convolution on x-transformed points. *Adv. Neural Inf. Process. Syst.* **31** (2018)
16. Liu, Y., Fan, B., Xiang, S., Pan, C.: Relation-shape convolutional neural network for point cloud analysis. In: CVPR, pp. 8895–8904 (2019)
17. Liu, Z., et al.: Swin transformer: hierarchical vision transformer using shifted windows. In: CVPR, pp. 10012–10022 (2021)
18. Ma, X., Qin, C., You, H., Ran, H., Fu, Y.: Rethinking network design and local geometry in point cloud: a simple residual MLP framework. In: ICLR (2022)
19. Maturana, D., Scherer, S.: VoxNet: a 3D convolutional neural network for real-time object recognition. In: IEEE International Conference on Intelligent Robots and Systems, pp. 922–928. IEEE (2015)
20. Mazur, K., Lempitsky, V.: Cloud transformers: a universal approach to point cloud processing tasks. In: CVPR, pp. 10715–10724 (2021)
21. Park, J., Lee, S., Kim, S., Xiong, Y., Kim, H.J.: Self-positioning point-based transformer for point cloud understanding. In: CVPR, pp. 21814–21823 (2023)
22. Paul, S., Patterson, Z., Bouguila, N.: CrossMoCo: multi-modal momentum contrastive learning for point cloud. In: International Conference on Robots and Vision, pp. 273–280. IEEE (2023)
23. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: PointNet: deep learning on point sets for 3D classification and segmentation. In: CVPR, pp. 652–660 (2017)
24. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: deep hierarchical feature learning on point sets in a metric space. *Adv. Neural Inf. Process. Syst.* **30** (2017)
25. Qian, G., et al.: PointNext: revisiting PointNet++ with improved training and scaling strategies. *Adv. Neural Inf. Process. Syst.* **35**, 23192–23204 (2022)
26. Qiu, S., Anwar, S., Barnes, N.: Dense-resolution network for point cloud classification and segmentation. In: CVPR, pp. 3813–3822 (2021)
27. Qiu, S., Anwar, S., Barnes, N.: Geometric back-projection network for point cloud classification. *IEEE Trans. Multimedia* **24**, 1943–1955 (2022)
28. Ran, H., Zhuo, W., Liu, J., Lu, L.: Learning inner-group relations on point clouds. In: CVPR, pp. 15477–15487 (2021)

29. Riegler, G., Osman Ulusoy, A., Geiger, A.: OctNet: learning deep 3D representations at high resolutions. In: CVPR, pp. 3577–3586 (2017)
30. Rusu, R.B., Blodow, N., Beetz, M.: Fast point feature histograms (FPFH) for 3D registration. In: International Conference on Robotics and Automation, pp. 3212–3217 (2009)
31. Thomas, H., Qi, C.R., Deschaud, J.E., Marcotegui, B., Goulette, F., Guibas, L.J.: KPConv: flexible and deformable convolution for point clouds. In: CVPR, pp. 6411–6420 (2019)
32. Uy, M.A., Pham, Q.H., Hua, B.S., Nguyen, T., Yeung, S.K.: Revisiting point cloud classification: a new benchmark dataset and classification model on real-world data. In: CVPR, pp. 1588–1597 (2019)
33. Vaswani, A., et al.: Attention is all you need. *Adv. Neural Inf. Process. Syst.* **30** (2017)
34. Wang, H., Liu, Q., Yue, X., Lasenby, J., Kusner, M.J.: Unsupervised point cloud pre-training via occlusion completion. In: CVPR, pp. 9782–9792 (2021)
35. Wang, W., et al.: Pyramid vision transformer: a versatile backbone for dense prediction without convolutions. In: CVPR, pp. 568–578 (2021)
36. Wang, Y., Sun, Y., Liu, Z., Sarma, S.E., Bronstein, M.M., Solomon, J.M.: Dynamic graph CNN for learning on point clouds. *ACM Trans. Graph.* **38**(5), 1–12 (2019)
37. Warehouse, D.: Sketchup, pp. 2–6 (2022). <https://3dwarehouse.sketchup.com/>
38. Wu, W., Qi, Z., Fuxin, L.: PointConv: deep convolutional networks on 3D point clouds. In: CVPR, pp. 9621–9630 (2019)
39. Wu, Z., et al.: 3D ShapeNets: a deep representation for volumetric shapes. In: CVPR, pp. 1912–1920 (2015)
40. Xie, S., Liu, S., Chen, Z., Tu, Z.: Attentional ShapeContextNet for point cloud recognition. In: CVPR, pp. 4606–4615 (2018)
41. Xu, M., Ding, R., Zhao, H., Qi, X.: PACConv: position adaptive convolution with dynamic kernel assembling on point clouds. In: CVPR, pp. 3173–3182 (2021)
42. Xu, Y., Fan, T., Xu, M., Zeng, L., Qiao, Y.: SpiderCNN: deep learning on point sets with parameterized convolutional filters. In: ECCV, pp. 87–102 (2018)
43. Yang, J., et al.: Modeling point clouds with self-attention and Gumbel subset sampling. In: CVPR, pp. 3323–3332 (2019)
44. Zhao, H., Jiang, L., Jia, J., Torr, P.H., Koltun, V.: Point transformer. In: CVPR, pp. 16259–16268 (2021)
45. Zhou, Y., Tuzel, O.: VoxelNet: end-to-end learning for point cloud based 3D object detection. In: CVPR, pp. 4490–4499 (2018)
46. Zhu, X., Hu, H., Lin, S., Dai, J.: Deformable ConvNets v2: more deformable, better results. In: CVPR, pp. 9308–9316 (2019)



CroMA: Cross-Modal Attention for Visual Question Answering in Robotic Surgery

Greetta Antonio¹ , Jobin Jose¹ , Sudhish N George¹ ,
and Kiran Raja² 

¹ National Institute of Technology Calicut, Kolkata, India

{greetta.antonio,jobin.jbn}@gmail.com, sudhish@nitc.ac.in

² Norwegian University of Science and Technology, Trondheim, Norway
kiran.raja@ntnu.no

Abstract. Most of the existing Surgical Visual Question Answering (VQA) systems use naive fusion strategies for text and image modalities and there is an absence of localized answering. The limited availability of annotated medical data and the complexity of domain-specific terminology have further limited the exploration of VQA systems for surgical procedures. We propose a Cross-Modal Attention (CroMA) based VQA system which can effectively fuse multimodal features from visual and textual sources. The fused embedding will feed a standard Class-Attention in Image Transformer (CaiT) module to the parallel classifier and the detector for joint prediction. Our experimental results on two public datasets suggest that CroMA based VQA system can better comprehend the surgical scene and localize the specific areas related to it with fewer parameters compared to other state-of-the-art (SOTA) models.

Keywords: Computer Vision · Cross- Modal Attention · Surgical Visual Question Answering

1 Introduction

The lack of specialized medical knowledge leaves many individuals, including patients and junior healthcare professionals, with unanswered questions about medical diagnoses and surgical procedures [1]. Access to medical experts is limited due to their scarcity and heavy workload. A computer-assisted system that processes medical data and answers questions could benefit junior doctors and reduce the expert's workload. However, developing a generalizable algorithm for surgical Visual Question Answering (VQA) is challenging due to factors such as large dataset size, variations in surgical techniques and patient anatomy, limited lighting, and occlusion caused by surgical tools and blood in tissue visuals [2, 3].

Recently, MedFuseNet [4] was introduced to perform medical visual question answering, demonstrating the potential to develop a reliable VQA model that could assist medical professionals in answering the queries from students and patients. Additionally, Surgical-VQA [1] has been developed to answer questions

about surgical instruments, their interactions, and surgical phases based on the given visual input. Although Surgical-VQA [1] attempts to address the “why?” question using a sentence-based VQA model, it becomes challenging and time-consuming due to the lack of annotated datasets in the medical field [3]. To facilitate easier inference for the “why?” aspect, our proposed model involves addressing the “what?” and the “where” through a VQA system tailored to the surgical domain. Figure 1 represents the overall pipeline of the proposed CroMA, which does not require object proposals, making it more efficient and less depend on preliminary region identification, and it is possible to output bounding box prediction along with the classification results.

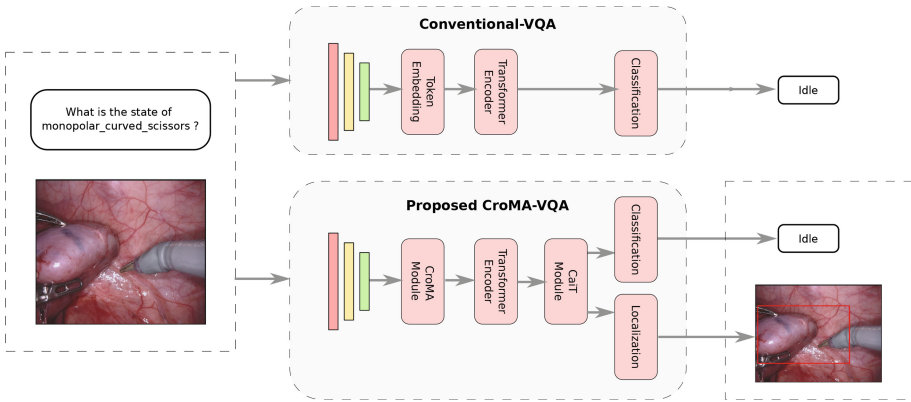


Fig. 1. Overview of the proposed CroMA framework, against the conventional VQA methods. Object proposals are not necessary, and the proposed model can output bounding box predictions along with the classification results.

1.1 Related Works

The integration of visual and language data for multimodal tasks have gained a significant attention in recent years, particularly in the VQA [1]. Sharma *et al.* [4] introduced MedFuseNet, an attention-based multimodal deep learning model for VQA [1] in the medical domain. MedFuseNet [4] learns representations by optimally fusing multimodal inputs using an attention mechanism [11]. The model consists of an answer prediction module, a feature fusion, and a feature extractor for questions and images. Attention modules within the model enhances interpretability. MedFuseNet [4] excels in both answer categorization and generation tasks by selecting appropriate answers from a predefined set. However, despite its superior performance when compared to various state-of-the-art attention-based VQA models, it faces challenges in handling complex multimodal interactions and understanding detailed medical queries, which limits its applicability in certain contexts.

Building on the idea of multimodal [12] fusion, Himanshu *et al.* [5] proposed a VQA model that leverages information from the image captioning task. VQA model integrates an image-captioning module with the VQA system by combining the semantic visual features derived from the image-captioning [13] dataset with visual features attended to based on the questions. Using ResNet101 for image encoding and a graph neural network (GNN) [14] to model the contextual relationships between detected objects, the model employs an attention mechanism over these object relationships and image regions to derive question-relevant visual features. This approach significantly boosts accuracy by capturing the fine-grained correspondence between images and questions. However, the model’s reliance on captioning quality for effective VQA remains as a notable limitation.

Florence-2 [34] advances vision-language integration by offering a unified, prompt-based architecture for diverse tasks such as object detection and image captioning. Its sequence-to-sequence framework processes images and textual prompts together, leveraging extensive pretraining on the FLD-5B [34] dataset with 5.4 billion annotations. While Florence-2 excels in versatility and achieves state-of-the-art performance, its dependence on the quality of the FLD-5B [34] dataset and high computational demands pose notable limitations, potentially restricting its use in resource-constrained settings.

Wang *et al.* [15] presented a simple yet efficient vision-language pretraining framework in which they process images as patches and is trained end-to-end with a unified prefix language modeling goal. By separating bidirectional encoding from unidirectional decoding, they achieved better joint vision-language representation learning. The use of both weakly aligned image-text data and text-only dataset helped to bridge the gap between visual and textual representations. In a related effort to enhance image understanding, Cornia *et al.* [16] developed a fully-attentive image captioning [13] algorithm with a multi-layer encoder for image regions and a decoder for generating output sentences. By combining the encoder and the decoder layers through a mesh-like structure with a learnable gating mechanism, the model exploits both low-level and high-level features. Despite its robust performance, the model is computationally expensive and requires substantial resources for training and inference.

Focusing on revamping feature extraction, Islam *et al.* [17] demonstrated enhanced feature extraction methods using label smoothing weighted loss, a regularization technique that smoothens the target labels during training to prevent overfitting in a GNN [14] model for tool-tissue interaction in surgical scenes. Their model predicts relationships between the defective tissue and surgical tools with improved accuracy over baseline models. Label smoothing significantly increases the performance, but the model’s complexity and additional attention mechanisms add to its computational overhead, posing a drawback in resource-limited settings.

Addressing the need for rich semantic understanding, Li *et al.* [7] proposed VisualBERT, a framework that captures rich semantics from images and associated text. Integrating BERT [8] with pretrained object proposal systems like

Faster-RCNN [18], VisualBERT [7], etc. aligns words and image regions using attention weights. It improves alignments by using successive transformer layers, and helps to better understand the detailed semantics of images. VisualBERT performs well on various vision-and-language tasks but can be limited by its computational complexity, dependence on large annotated datasets, and slower inference speeds.

The computer vision field has recently experienced a surge in models that combine image understanding and language processing to tackle VQA challenges. These models are designed to extract detailed and context-specific information from visual data by tailoring the question to the task, frequently employing long short-term memory (LSTM) [5] networks or attention mechanisms [4]. To extract the visual feature, they require object detection models which identify the key objects and primary regions in an image. During the initial training phase of the object detection model, both question and answer annotations and bounding box annotations are required. However, these methods face limitations in effectively capturing the complex interactions between visual and textual features, particularly in specialized domains like surgical scene understanding. To address these limitations and enhance feature fusion, this work introduces:

- A detection-free **Cross-Modal Attention (CroMA)** based Surgical VQA model which enables comprehensive training for localised answering, leveraging both visual and language inputs is designed.
- A unified cross-modal attention module which utilises attention mechanisms to fuse the visual and textual features efficiently is used. It provides more fine-grained control by handling the interactions between modalities separately before combining them.
- The performance of the proposed CroMA model with other SOTA models is compared and it is observed that the CroMA achieves better performance with fewer parameters. Furthermore, the ablation study confirmed the superior fusion strategy compared to other fusion techniques.

2 Cross-Modal Attention (CroMA) Model

The proposed CroMA model can handle and integrate data from various modalities and perform the VQA task in surgical context. Figure 2 represents the detailed architecture of the proposed CroMA model which consists of a visual feature extractor, tokenizer, a unified cross modal attention based module, and a standard CaiT module followed by prediction heads.

Feature Extraction: Conventional VQA models typically extract visual features using object proposals [7]. But here, we utilize ResNet18 [19], pre-trained on ImageNet [20] for visual feature extraction. This approach allows to have a broader understanding of the surgical context along with better inference speeds. Unlike VisualBERT [7], we found that pre-trained ResNet18 [19] achieves superior performance in our task with low computational overhead. Language embeddings are obtained through a pre-trained BERT [1] tokenizer. For visual data,

the input features are processed through a non-linear visual projection layer, which consists of a multi-layer perceptron (MLP) with Leaky ReLU activation. This maps the raw visual features into a compatible embedding space matching the text embedding dimensions.

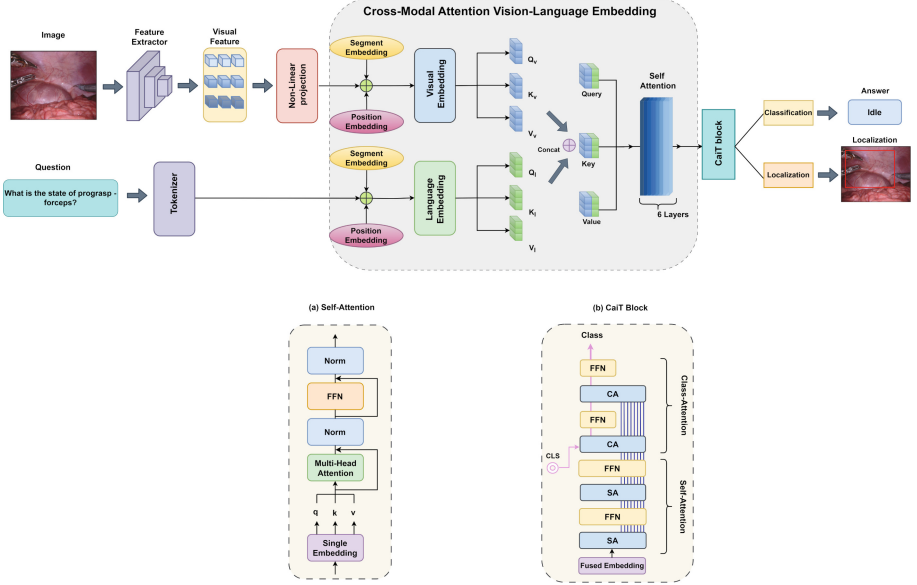


Fig. 2. The proposed CroMA architecture. The components include a feature extractor for visual inputs, tokenizer, CroMA embedding module, pre-trained CaiT block, and a prediction head for classification and localization tasks. (a) Self-Attention block [32], (b) CaiT block [9]

VisualBERT Embedding: The extracted features are then transformed into embeddings similar to VisualBERT [7]. It is an advanced model that enhances the BERT [8] framework by incorporating visual data along with textual information. The BERT model [8] processes an input sentence by breaking it down into a series of tokens for language analysis. Each word token is then associated with a set of embeddings E , where each embedding ($e \in E$) is derived from a combination of the token e_t , the segment e_s , and the position e_p embeddings. VisualBERT [7] extends the functionality of BERT by breaking down the extracted image features into tokens, similar to how words are broken down in BERT [8]. Each visual token embedding ($f \in F$) is constructed using visual f_o features, segment f_s and position f_p embeddings. Both the language and visual embeddings are then passed through the subsequent layers in VisualBERT [7] model, facilitating intricate interactions and forming a joint representation of both modalities.

CroMA Embedding: The proposed method introduces a unified cross-modal attention (CroMA) model which transforms the extracted features into embeddings similar to VisualBERT [7] that will combine both modalities. However, CroMA distinguishes itself from VisualBERT [7], which directly concatenates the embeddings before feeding them to the transformer, by implementing explicit cross-modal attention layers that independently process textual and visual data before merging them. This approach offers more fine-grained control over how the textual and visual data is integrated before merging them. Additionally, separate projection layers for language and visual queries, keys, and values allow for more specialized transformations, optimizing the interaction between the heterogeneous embeddings.

CroMA module consists of multiple attention layers designed to align and integrate the language and visual features. Queries (\mathbf{Q}), keys (\mathbf{K}), and values (\mathbf{V}) play crucial roles in each attention layer, facilitating the integration of multimodal information. Queries identify elements of interest, keys represent all possible pieces of information to attend to, and values are the information to be combined based on relevance. Attention mechanism enables effective integration of multimodal information by computing relevance scores between queries and keys.

For each attention layer i , the model creates separate projections for queries, keys, and values for both language and visual inputs. Language queries, keys and values ($\mathbf{Q}_l^i, \mathbf{K}_l^i, \mathbf{V}_l^i$) are derived from the language embedding \mathbf{E}_l^i using respective weight matrices $\mathbf{W}_{\mathbf{Q}_l}^i, \mathbf{W}_{\mathbf{K}_l}^i, \mathbf{W}_{\mathbf{V}_l}^i$ as:

$$\mathbf{X}_l^i = W_{\mathbf{X}_l}^i \cdot \mathbf{E}_l^i + \mathbf{b}_{\mathbf{X}_l}^i \quad (1)$$

where \mathbf{X}_l^i can represent the queries \mathbf{Q}_l^i , keys \mathbf{K}_l^i or values \mathbf{V}_l^i , with corresponding weight matrices $\mathbf{W}_{\mathbf{X}_l}^i$ and bias terms $\mathbf{b}_{\mathbf{X}_l}^i$. Similarly visual queries, keys and values ($\mathbf{Q}_v^i, \mathbf{K}_v^i, \mathbf{V}_v^i$) are generated from the visual embedding \mathbf{E}_v^i using weight matrices $\mathbf{W}_{\mathbf{Q}_v}^i, \mathbf{W}_{\mathbf{K}_v}^i, \mathbf{W}_{\mathbf{V}_v}^i$ as:

$$\mathbf{X}_v^i = W_{\mathbf{X}_v}^i \cdot \mathbf{E}_v^i + \mathbf{b}_{\mathbf{X}_v}^i \quad (2)$$

These projections are then concatenated to form unified queries, keys, and values ($\mathbf{Q}^i, \mathbf{K}^i, \mathbf{V}^i$) as shown in Eq. (3) which enables the attention mechanism to compute relevance scores across both modalities.

$$\mathbf{Q}^i = \begin{bmatrix} \mathbf{Q}_l^i \\ \mathbf{Q}_v^i \end{bmatrix}, \quad \mathbf{K}^i = \begin{bmatrix} \mathbf{K}_l^i \\ \mathbf{K}_v^i \end{bmatrix}, \quad \mathbf{V}^i = \begin{bmatrix} \mathbf{V}_l^i \\ \mathbf{V}_v^i \end{bmatrix} \quad (3)$$

$$\text{Attention}(\mathbf{Q}^i, \mathbf{K}^i, \mathbf{V}^i) = \text{softmax} \left(\frac{\mathbf{Q}^i (\mathbf{K}^i)^T}{\sqrt{d_k}} \right) \mathbf{V}^i \quad (4)$$

Equation (4) represents the final attention output for each layer i where d_k represents the dimension of the key. These attention outputs, representing the fused

embeddings are then fed into the pre-trained CaiT [9] module before reaching the prediction head.

Algorithm 1: Computational flow of the CroMA model.

Input : Text and visual features
Output : Classification labels, Bounding box predictions
Initialization: VisualBert and CaiT models, embedding projection, classifier, MLP, no_improvement_count

best_val_loss $\leftarrow \infty$;
no_improvement_count $\leftarrow 0$;

- 1 Transform and project text and visual inputs into embeddings;
- 2 **repeat**
- 3 $(\mathbf{Q}_l^i, \mathbf{K}_l^i, \mathbf{V}_l^i) \leftarrow$ Compute using Eq. (1);
- 4 $(\mathbf{Q}_v^i, \mathbf{K}_v^i, \mathbf{V}_v^i) \leftarrow$ Compute using Eq. (2);
- 5 $\mathbf{Q}^i \leftarrow$ concatenate($\mathbf{Q}_l^i, \mathbf{Q}_v^i$);
- 6 $\mathbf{K}^i \leftarrow$ concatenate($\mathbf{K}_l^i, \mathbf{K}_v^i$);
- 7 $\mathbf{V}^i \leftarrow$ concatenate($\mathbf{V}_l^i, \mathbf{V}_v^i$);
- 8 attention output \leftarrow Compute using Eq. (4);
- 9 combined embeddings \leftarrow attention output;
- 10 Process combined embeddings through CaiT.;
- 11 classification outputs \leftarrow classifier(CaiT outputs);
- 12 boundingbox outputs \leftarrow sigmoid(boundingbox(CaiT outputs));
- 13 validation loss \leftarrow Compute on validation dataset;
- 14 **if** validation loss $<$ best_val_loss **then**
- 15 | best_val_loss \leftarrow validation loss;
- 16 | no_improvement_count $\leftarrow 0$;
- 17 **else**
- 18 | no_improvement_count \leftarrow no_improvement_count + 1;
- 19 **until** validation loss not reducing;
- 20 **return** classification outputs, bounding box outputs;

Class-Attention in Image Transformers (CaiT): [9] It is an advanced variant of the Vision Transformer [10] architecture, specifically designed to enhance image classification tasks by introducing a class-specific attention mechanism. Unlike traditional ViT [10] models that process images by dividing them into a sequence of patches and embedding them into high-dimensional vector spaces, CaiT [9] incorporates an additional class-attention stage. This stage focuses on a classification token (CLS token), refining its representation through multiple class-attention layers.

The architecture consists of two distinct stages: a self-attention [11] stage identical to ViT [10] but without the class embedding, and a class-attention stage where only the class embedding is updated. This approach separates the objectives of self-attention and classification, allowing more effective feature extraction and representation. The outputs from these stages are subsequently fed into the prediction head for the classification and the localization tasks.

Prediction Head: The output features are processed through the prediction heads which consist of a classification and a localization head. In the classification stage, the features from the CaiT [9] block’s output is given to a linear layer followed by a Softmax activation function to generate classification predictions. Whereas, the localization head utilizes a feed-forward network (FFN) architecture which consists of a three-layer perceptron along with ReLU activation function preceding a linear projection layer. The overall computational flow of the proposed CroMA model is outlined in the Algorithm 1.

Loss Function: The proposed model utilizes a combined loss function, where cross-entropy loss (LCE) is used for classification tasks. For the detection task, \mathcal{L}_1 norm is combined with the Generalized Intersection over Union (GIoU) [33] loss and is used for bounding box prediction. The GIoU [33] enhances the model’s accuracy in predicting bounding box locations by considering both the overlap and the distance between predicted and actual boxes. The final loss function is given by Eq. (5) which is the sum of detection and classification losses.

$$\mathcal{L} = \mathcal{L}_{CE} + (\mathcal{L}_{GIoU} + \mathcal{L}_1) \quad (5)$$

3 Results and Discussions

3.1 Datasets

EndoVis 18 Dataset: EndoVis 18 Dataset is derived from the MICCAI Endoscopic Vision Challenge 2018 [21] which is a publicly available dataset consisting of 14 video sequences of robotic surgery procedures. This dataset uniquely integrates bounding box annotations for tissue-instrument interaction [17] and question-answer pairs from VQA classification tasks. The resultant EndoVis-18-VQLA dataset has extensive annotations [1] that incorporate matching bounding box data with question-answer pairings which cover organs, tool interactions, and their locations. The training set comprises of 11 sequences which include 1560 images with 9014 question-answer (QA) pairs, while the test set consists of 3 sequences containing 447 frames and 2769 question-answer pairs.

EndoVis 17 Dataset: EndoVis 17 Dataset is a dataset from the MICCAI Endoscopic Vision Challenge 2017 [22] which is also publicly accessible. Using standard tools and interactions from EndoVis-2017, 97 frames are manually chosen in order to examine the model’s generalisation capabilities. The frames are then annotated with question-answer bounding box labels. This external validation dataset includes 472 question-answer pairs with 97 frames.

3.2 Implementation Details

The models are trained by using Adam optimizer and the batch size and epoch are set to 64 and 80, along with a learning rate of 1×10^{-5} . Python PyTorch framework is used for all experiments, which are run on a server equipped with an Intel® Core™ i7-10700 CPU and an NVIDIA A100-SXM GPU. Training is conducted on the EndoVis-18-VQLA training set, while validation is performed on both the EndoVis-18 validation set and EndoVis-17 external dataset. We conduct the quantitative comparison experiments against the models VisualBERT [7], VisualBERT ResMLP [1], MCAN [23], VQA-DeiT [24], MUTAN [25], MFH [26], and Block-Tucker [27]. In CroMA, we use CaiT block [9] in place of the multilayer transformer module in VisualBERT [7].

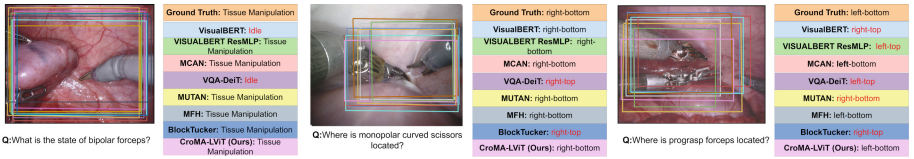


Fig. 3. Qualitative comparison of our proposed CroMA model against the SOTA models on generation of answers and bounding boxes. The colours of the bounding box are denoted as follows: orange: Ground-truth, light blue: VisualBERT [7] (Color figure online), light green: VisualBERT ResMLP [1], red: MCAN [23], purple: VQA-DeiT [24], yellow: MUTAN [25], gray: MFH [26], dark blue: Block Tucker [27], pink: CroMA (Ours)

3.3 Results

The performance of proposed CroMA model is shown both quantitatively (Table 1) and qualitatively in Fig. 3 against the SOTA models for VQA task on EndoVis-18 and EndoVis-17 datasets. Figure 3 shows examples of classification and bounding box generated by the SOTA models VisualBERT [7], VisualBERT ResMLP [1], MCAN [24], VQA-DeiT [25], MUTAN [26], MFH [27], Block-Tucker [28] and our CroMA model. This qualitative comparison shows that our model gives more accurate localization and classification prediction results when compared with the baseline models. Furthermore, our comparison of the outputs using object detection model to extract the features with these models using features from the whole image reveals that the latter consistently outperforms the former.

From Table 1 it is observed on EndoVis-18 dataset, CroMA achieves the highest accuracy (0.641) and F-Score (0.762), along with a competitive mIoU (0.408).

Table 1. Comparison results of CroMA model against other state-of-the-art models

Model	Detection	Feature Extraction	EndoVis-18 dataset			EndoVis-17 dataset		
			Acc	F-Score	mIoU	Acc	F-Score	mIoU
VisualBERT [7]	FRCNN	ResNet18	0.597	0.323	0.734	0.438	0.374	0.682
VisualBERT ResMLP [1]			0.606	0.323	0.730	0.427	0.351	0.695
MCAN [23]			0.608	0.343	0.726	0.426	0.304	0.683
VQA-DeiT [24]			0.609	0.322	0.734	0.449	0.321	0.713
MUTAN [25]			0.605	0.324	0.722	0.437	0.321	0.687
MFH [26]			0.618	0.316	0.723	0.373	0.205	0.718
BlockTucker [27]			0.607	0.341	0.731	0.437	0.321	0.683
CroMA(Ours)					0.612	0.355	0.753	0.441
VisualBERT [7]	X	ResNet18	0.621	0.332	0.736	0.389	0.316	0.711
VisualBERT ResMLP [1]			0.632	0.331	0.751	0.419	0.332	0.703
MCAN [23]			0.628	0.334	0.753	0.414	0.293	0.703
VQA-DeiT [24]			0.610	0.316	0.734	0.379	0.286	0.690
MUTAN [25]			0.628	0.339	0.763	0.424	0.348	0.722
MFH [26]			0.628	0.325	0.759	0.410	0.350	0.722
BlockTucker [27]			0.620	0.329	0.765	0.422	0.351	0.729
CroMA(Ours)					0.641	0.408	0.762	0.439

Similarly on the external test dataset it gives highest values for all evaluation metrics clearly indicating the superior performance of the model. Specifically, the results on the EndoVis-18 dataset show that bypassing the object proposal model (Faster RCNN [18]) in favor of using whole-image features (ResNet18 [19]) significantly enhances the performance in both classification and localization tasks, demonstrating the effectiveness of this approach in reducing false positives.

Table 2. Ablation study on various fusion strategies

Fusion Strategies	EndoVis-18-VQLA			EndoVis-17-VQLA		
	Acc	F-Score	mIoU	Acc	F-Score	mIoU
Concatenation [7]	0.610	0.316	0.734	0.380	0.286	0.691
JCA [28]	0.602	0.301	0.753	0.375	0.284	0.715
MMHCA [29]	0.609	0.312	0.745	0.358	0.300	0.708
MAT [30]	0.619	0.318	0.742	0.337	0.285	0.696
Gated Fusion [31]	0.607	0.379	0.768	0.403	0.282	0.738
Self-Attention [32]	0.592	0.309	0.727	0.369	0.267	0.672
Guided Attention [23]	0.619	0.313	0.731	0.352	0.229	0.719
CroMA(Ours)	0.641	0.408	0.762	0.439	0.363	0.738

An ablation study on different techniques of vision-language fusion is conducted in Table 2 where they all use the same feature extractor. We compare with Concatenation [7], Joint Cross-Attention (JCA) [28], Multimodal Multi-Head Convolutional Attention (MMHCA) [29], Multimodal Attention Transformers (MAT) [30], Gated Fusion [31], Self-Attention Fusion [32], and with Guided-Attention Fusion [23]. The study proves that our CroMA model has a better embedding fusion strategy when compared against these methods.

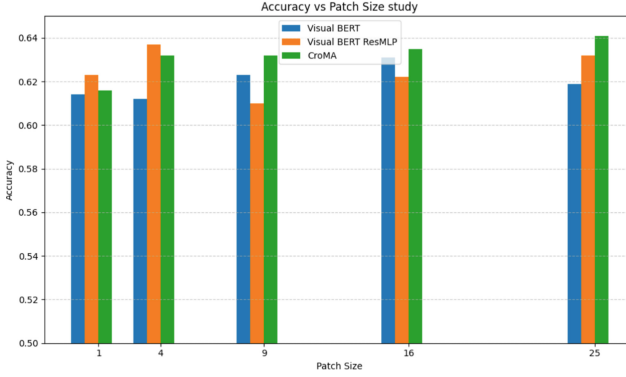


Fig. 4. Accuracy vs patch size study.

The performance of CroMA model is studied by changing the number of input image patches to 1, 4, 9, 16, and 25, respectively. It is observed from Fig. 4 that CroMA-based model generally performs better than VisualBERT [7] and VisualBERT ResMLP [1] based models, even with varied number of input patches. In general, it is also observed that there is an improvement in the performances with an increase in the number of patches. It is also worth noting that the proposed CroMA model (55.5M) requires 69.5% fewer parameters compared to VisualBERT ResMLP encoder and transformer decoder model (184.7M) [1] while maintaining similar performances.

4 Conclusion

This paper presents a transformer model incorporating Cross-Modal Attention Vision Language embeddings for surgical VQLA tasks, enabling it to provide localized answers based on specific surgical scenes and corresponding questions. The proposed CroMA embedding module effectively enhances the integration and fusion of heterogeneous features. Extensive comparative and quantitative experiments demonstrate the superior performance and robustness of the proposed CroMA compared to all other SOTA models in both classification and localization tasks, highlighting its potential for real-time applications. The model attends to incorrect regions of the image due to its overlapping features and when the target is far-off from the bounding box affecting its performance. Future work and investigations could focus on utilizing localization information to improve prediction reliability and exploring additional VQA-related challenges within the medical domain. Additionally, the integration of multimodal embeddings like Contrastive Language-Image Pre-training (CLIP) could be explored to enhance the detection-free capabilities of the CroMA model, potentially offering new insights into improving multimodal understanding and performance.

References

1. Seenivasan, L., Islam, M., Krishna, A.K., Ren, H.: Surgical-VQA: visual question answering in surgical scenes using transformer. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 33–43. Springer, Cham (2022)
2. Zhang, Y., Fan, W., Peng, P., Yang, X., Zhou, D., Wei, X.: Dual modality prompt learning for visual question-grounded answering in robotic surgery. *Vis. Comput. Ind. Biomed. Art* **7**(1), 9 (2024)
3. Spasic, I., Nenadic, G.: Clinical text data in machine learning: systematic review. *JMIR Med. Inf.* **8**(3) (2020)
4. Sharma, D., Purushotham, S., Reddy, C.K.: MedFuseNet: an attention-based multimodal deep learning model for visual question answering in the medical domain. *Sci. Rep.* **11**(1), 19826 (2021)
5. Sharma, H., Jalal, A.S.: Image captioning improved visual question answering. *Multimedia Tool Appl.* **81**(24), 34775–34796 (2022)
6. Bai, L., Islam, M., Seenivasan, L., Ren, H.: Surgical-VQLA: transformer with gated vision-language embedding for visual question localized-answering in robotic surgery. arXiv preprint [arXiv:2305.11692](https://arxiv.org/abs/2305.11692) (2023)
7. Li, L.H., Yatskar, M., Yin, D., Hsieh, C.J., Chang, K.W.: VisualBERT: a simple and performant baseline for vision and language. arXiv preprint [arXiv:1908.03557](https://arxiv.org/abs/1908.03557) (2019)
8. Kenton, J.D.M.W.C., Toutanova, L.K.: BERT: pre-training of deep bidirectional transformers for language understanding. arXiv preprint [arXiv:1810.04805](https://arxiv.org/abs/1810.04805) (2018)
9. Touvron, H., Cord, M., Sablayrolles, A., Synnaeve, G., Jégou, H.: Going deeper with image transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 32–42 (2021)
10. Dosovitskiy, A., et al.: An image is worth 16x16 words: transformers for image recognition at scale. arXiv preprint [arXiv:2010.11929](https://arxiv.org/abs/2010.11929) (2020)
11. Vaswani, A., et al.: Attention is all you need. *Adv. Neural Inf. Process. Syst.* **30** (2017)
12. Baltrušaitis, T., Ahuja, C., Morency, L.P.: Multimodal machine learning: a survey and taxonomy. *IEEE Trans. Pattern Anal. Mach. Intell.* **41**(2), 423–443 (2018)
13. Vinyals, O., Toshev, A., Bengio, S., Erhan, D.: Show and tell: a neural image caption generator. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3156–3164 (2015)
14. Cho, K., et al.: Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint [arXiv:1406.1078](https://arxiv.org/abs/1406.1078) (2014)
15. Wang, Z., Yu, J., Yu, A.W., Dai, Z., Tsvetkov, Y., Cao, Y.: SimVLM: simple visual language model pretraining with weak supervision. arXiv preprint [arXiv:2108.10904](https://arxiv.org/abs/2108.10904) (2021)
16. Cornia, M., Stefanini, M., Baraldi, L., Cucchiara, R.: Meshed-memory transformer for image captioning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10578–10587 (2020)
17. Islam, M., Seenivasan, L., Ming, L.C., Ren, H.: Learning and reasoning with the graph structure representation in robotic surgery. In: Medical Image Computing and Computer Assisted Intervention–MICCAI 2020: 23rd International Conference, Lima, Peru, October 4–8, 2020, Proceedings, Part III, 23, pp. 627–636. Springer (2020)

18. Girshick, R.: Fast r-CNN. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1440–1448 (2015)
19. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
20. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: a large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255. IEEE (2009)
21. Allan, M., et al.: 2018 robotic scene segmentation challenge. arXiv preprint [arXiv:2001.11190](https://arxiv.org/abs/2001.11190) (2020)
22. Allan, M., et al.: 2017 robotic instrument segmentation challenge. arXiv preprint [arXiv:1902.06426](https://arxiv.org/abs/1902.06426) (2019)
23. Yu, Z., Yu, J., Cui, Y., Tao, D., Tian, Q.: Deep modular co-attention networks for visual question answering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6281–6290 (2019)
24. Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jégou, H.: Training data-efficient image transformers and distillation through attention. In: International Conference on Machine Learning, pp. 10347–10357. PMLR (2021)
25. Ben-Younes, H., Cadene, R., Cord, M., Thome, N.: MUTAN: multimodal tucker fusion for visual question answering. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2612–2620 (2017)
26. Xiang, S., Chen, Q., Fang, X., Guo, M.: Improving visual question answering by multimodal gate fusion network. In: 2023 International Joint Conference on Neural Networks (IJCNN), pp. 1–8. IEEE (2023)
27. Ben-Younes, H., Cadene, R., Thome, N., Cord, M.: BLOCK: Bilinear superdiagonal fusion for visual question answering and visual relationship detection. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, no. 1, pp. 8102–8109 (2019)
28. Praveen, R.G., et al.: A joint cross-attention model for audio-visual fusion in dimensional emotion recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2486–2495 (2022)
29. Georgescu, M.I., et al.: Multimodal multi-head convolutional attention with various kernel sizes for medical image super-resolution. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 2195–2205 (2023)
30. Wu, Z., Liu, L., Zhang, Y., Mao, M., Lin, L., Li, G.: Multimodal crowd counting with mutual attention transformers. In: 2022 IEEE International Conference on Multimedia and Expo (ICME), pp. 1–6. IEEE (2022)
31. Arevalo, J., Solorio, T., Montes-y-Gómez, M., González, F.A.: Gated multimodal units for information fusion. arXiv preprint [arXiv:1702.01992](https://arxiv.org/abs/1702.01992) (2017)
32. Vaswani, A., et al.: Attention is all you need. *Adv. Neural Inf. Process. Syst.* **30** (2017)
33. Rezafooghi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I., Savarese, S.: Generalized intersection over union: a metric and a loss for bounding box regression. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 658–666 (2019)
34. Xiao, B., et al.: Florence-2: advancing a unified representation for a variety of vision tasks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4818–4829 (2024)
35. Li, J., et al.: LLaVA-Surg: towards multimodal surgical assistant via structured surgical video learning. arXiv preprint [arXiv:2408.07981](https://arxiv.org/abs/2408.07981) (2024)

Author Index

A

Antonio, Greetta 459
Anwar, Saeed 151
Azarmi, Mohsen 151

B

B. Eisma, Yke 265
Biswas, Soumen 45
Bonnet, Pierre 296
Burghouts, Gertjan 265

C

Chai, Abel Yu Hao 296
Chakraborty, Anirban 427
Chen, Li 16
Chen, Xucan 377
Chen, Yaojie 16
Chen, Yongxin 61
Chen, Zeyuan 393
Chen, Zibo 329
Chib, Pranav Singh 29

D

Ding, Errui 167
Ding, Shuaipeng 183
Du, Zetao 280

F

Fang, Yi 119, 135, 346
Feng, Zhida 16
Fu, Ruigang 411
Fujisawa, Makoto 393

G

Gabr, Mohamed 235
Ganesh, Ananth 45
Gao, Qiang 393
Gatti, Prajwal 312
Geng, Wanpeng 198
Gopalakrishnan, Viswanath 427

Graser, Rainer 105
Guan, Yong 61
Guo, Baizhang 360
Guo, Shuai 360

H

Hao, Yu 346
Hu, Qingyong 411
Hu, Quan 16
Hu, Wei 249
Huang, Hao 119, 135, 346
Huang, Yan 280
Hussain, Tanveer 151

J

Jiang, Jian-Jian 329
Joly, Alexis 296
Jose, Jobin 459
Joshi, Soham 427

K

Kalpesh Patel, Samay 443
Kancharla, Damodar Datta 74
Kandath, Harikumar 74
Kiefer, Benjamin 105
Kumar, Nikhil 29
Kumari, Seema 443

L

Lai, Zhihui 167
Lee, Sue Han 296
Li, Biao 411
Li, Jiamao 219
Li, Mingyong 183
Li, Ruihao 377
Li, Wenjing 151
Li, Xin 183
Li, Yang 90
Li, Zhanli 360
Liang, Jiazhao 346

Liaw, Jerad Zherui 296
 Lin, Hui 346
 Liu, Daizong 249
 Liu, Jing 198
 Liu, Yang 249
 Liu, Yu-Shen 346
 Liu, Zhe 377
 Lu, Jinchen 167
 Lu, Jun 1

M

Messmer, Martin 105
 Mikawa, Masahiko 393
 Ming, Wenwen 377
 Mishra, Anand 312
 Mu, Qi 360
 Muthalagu, Raja 443

N

N George, Sudhish 459

Q

Qian, Chenghao 151
 Quan, Yitong 105

R

Raja, Kiran 459
 Ram Akupati, Charan 105
 Raman, Shanmuganathan 443
 Ramancharla, Pradeep Kumar 74
 Rezaei, Mahdi 151

S

Sarvadevabhatla, Ravi Kiran 74
 Shao, Zhenzhou 61
 Sharma, Arvind Kumar 312
 Shi, Wenjun 219
 Shui, Jianan 183
 Singh, Pravendra 29
 Song, Yu 411
 Srivastava, Kushagra 74

T

Tahereen, Rizvi 74
 Tan, Xiao 167

Telib, Abdelrahman 235
 Tripathi, Aditay 427
 Tzes, Anthony 119, 135

U

Unlu, Halil Utku 119, 135

V

Vaidya, Shreyas 312

W

Wang, Jingdong 167
 Wang, Junbo 280
 Wang, Lei 219
 Wang, Liang 280
 Wang, Mengfei 219
 Wang, Wei 151
 Wang, Xuekuan 167
 Wang, Ziquan 393
 Wei, Yi-Lin 329
 Wen, Congcong 119, 135, 346
 Wu, Xiao-Ming 329

X

Xiao, Jianli 90
 Xu, Yang 1

Y

Yang, Jianhua 280
 Yang, Zhiyuan 167
 Yi, Wei 377
 Yuan, Shuaihang 119, 135, 346

Z

Zahran, Youssef 265
 Zell, Andreas 105
 Zhang, Dexin 198
 Zhang, Hui 198
 Zhang, Jie 61
 Zhang, Wei 167
 Zhao, Cairong 167
 Zheng, Wei-Shi 329
 Zhong, Ping 411
 Zhu, Dongchen 219
 Zou, Yibo 61