

LNCS 15317

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

Pattern Recognition

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

17 Part XVII

ICPR
2024 INDIA



 Springer

MOREMEDIA 

Lecture Notes in Computer Science

15317

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 XVII

Editors

Apostolos Antonacopoulos 
University of Salford
Salford, Lancashire, UK

Rama Chellappa 
Johns Hopkins University
Baltimore, MD, USA

Saumik Bhattacharya 
IIT Kharagpur
Kharagpur, West Bengal, India

Subhasis Chaudhuri 
Indian Institute of Technology Bombay
Mumbai, Maharashtra, India

Cheng-Lin Liu 
Chinese Academy of Sciences
Beijing, China

Umapada Pal 
Indian Statistical Institute Kolkata
Kolkata, West Bengal, India

ISSN 0302-9743

ISSN 1611-3349 (electronic)

Lecture Notes in Computer Science

ISBN 978-3-031-78446-0

ISBN 978-3-031-78447-7 (eBook)

<https://doi.org/10.1007/978-3-031-78447-7>

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

Chapter “Dense Road Surface Grip Map Prediction from Multimodal Image Data” is licensed under the terms
of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>).
For further details see license information in the chapter.

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 Institute of Technology, 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	Indian Institute of Technology, 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
 Anwasha 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	Galal Binamakhshen
Emna Ghorbel	Ganesh Krishnasamy
Enrique Naredo	Gang Pan
Enyu Cai	Gangyan Zeng
Eric Patterson	Gani Rahmon
Ernest Valveny	Gaurav Harit
Eva Blanco-Mallo	Gennaro Vessio
Eva Breznik	Genoveffa Tortora
Evangelos Sartinas	George Azzopardi
Fabio Solari	Gerard Ortega
Fabiola De Marco	Gerardo E. Altamirano-Gomez
Fan Wang	Gernot A. Fink
Fangda Li	Gibran Benitez-Garcia
Fangyuan Lei	Gil Ben-Artzi
Fangzhou Lin	Gilbert Lim
Fangzhou Luo	Giorgia Minello
Fares Bougourzi	Giorgio Fumera
Farman Ali	Giovanna Castellano
Fatiha Mokdad	Giovanni Puglisi
Fei Shen	Giulia Orrù
Fei Teng	Giuliana Ramella
Fei Zhu	Gökçe Uludoğan
Feiyan Hu	Gopi Ramena
Felipe Gomes Oliveira	Gorthi Rama Krishna Sai Subrahmanyam
Feng Li	Gourav Datta
Fengbei Liu	Gowri Srinivasa
Fenghua Zhu	Gozde Sahin
Fillipe D. M. De Souza	Gregory Randall
Flavio Piccoli	Guanjie Huang
Flavio Prieto	Guanjun Li
Florian Kleber	Guanwen Zhang
Francesc Serratosa	Guanyu Xu
Francesco Bianconi	Guanyu Yang
Francesco Castro	Guanzhou Ke
Francesco Ponzio	Guhnoo Yun
Francisco Javier Hernández López	Guido Borghi
Frédéric Rayar	Guilherme Brandão Martins
Furkan Osman Kar	Guillaume Caron
Fushuo Huo	Guillaume Tochon
Fuxiao Liu	Guocai Du
Fu-Zhao Ou	Guohao Li
Gabriel Turinici	Guoqiang Zhong
Gabrielle Flood	Guorong Li
Gajjala Viswanatha Reddy	Guotao Li
Gaku Nakano	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	Suraj Kumar Pandey
Snehasis Banerjee	Surendrabikram Thapa
Snehasis Mukherjee	Suresh Sundaram
Snigdha Sen	Sushil Bhattacharjee
Sofia Casarin	Susmita Ghosh
Soheila Farokhi	Swakkhar Shatabda
Soma Bandyopadhyay	Syed Ms Islam
Son Minh Nguyen	Syed Tousiful Haque
Son Xuan Ha	Taegyeong Lee
Sonal Kumar	Taihui Li
Sonam Gupta	Takashi Shibata
Sonam Nahar	Takeshi Oishi
Song Ouyang	Talha Ahmad Siddiqui
Sotiris Kotsiantis	Tanguy Gernot
Souhaila Djaffal	Tangwen Qian
Soumen Biswas	Tanima Bhowmik
Soumen Sinha	Tanpia Tasnim
Soumitri Chattopadhyay	Tao Dai
Souvik Sengupta	Tao Hu
Spiros Kostopoulos	Tao Sun
Sreeraj Ramachandran	Taoran Yi
Sreya Banerjee	Tapan Shah
Srikanta Pal	Taveena Lotey
Srinivas Arukonda	Teng Huang
Stephane A. Guinard	Tengqi Ye
Su O. Ruan	Teresa Alarcon
Subhadip Basu	Tetsuji Ogawa
Subhajit Paul	Thanh Phuong Nguyen
Subhankar Ghosh	Thanh Tuan Nguyen
Subhankar Mishra	Thattapon Surasak
Subhankar Roy	Thibault Napol�on
Subhash Chandra Pal	Thierry Bouwmans
Subhayu Ghosh	Thinh Truong Huynh Nguyen
Sudip Das	Thomas De Min
Sudipta Banerjee	Thomas E. K. Zielke
Suhas Pillai	Thomas Swearingen
Sujit Das	Tianatahina Jimmy Francky Randrianasoa
Sukalpa Chanda	Tianheng Cheng
Sukhendu Das	Tianjiao He
Suklav Ghosh	Tianyi Wei
Suman K. Ghosh	Tianyuan Zhang
Suman Samui	Tianyue Zheng
Sumit Mishra	Tiecheng Song
Sungho Suh	Tilottama Goswami
Sunny Gupta	Tim B�chner

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 XVII

Automatic Wood Pith Detector: Local Orientation Estimation and Robust Accumulation	1
<i>Henry Marichal, Diego Passarella, and Gregory Randall</i>	
A Multi-ground Truth Approach for RGB-D Saliency Detection	16
<i>Nguyen Truong Thinh Huynh, Van Linh Pham, Xuan Toan Mai, and Tuan Anh Tran</i>	
MSSF: A Multi-scale Siamese Flow Architecture for Multi-texture Class Anomaly Detection	30
<i>Yibo Chen, Zhiyuan Hu, Le Huang, and Jianming Zhang</i>	
DSLAs: A Distance-Sensitive Label Assignment Strategy for Oriented Object Detection in Remote Sensing Images	45
<i>Minghong Wei, Yan Dong, Haobin Xiang, Guangshuai Gao, and Chunlei Li</i>	
BridgeCLIP: Automatic Bridge Inspection by Utilizing Vision-Language Model	61
<i>Powei Liao and Gaku Nakano</i>	
Writer Identification in Multiple Medieval Books: A Preliminary Study	77
<i>Tiziana D’Alessandro, Claudio De Stefano, Francesco Fontanella, and Alessandra Scotto di Freca</i>	
Cyclic Learning of a Frame Downsampler and a Recognition Model in High-Speed Camera Image Recognition	93
<i>Shigeaki Namiki, Takuya Ogawa, Keiko Yokoyama, Shoji Yachida, and Toshinori Hosoi</i>	
Early Feature Distributions Alignment in Visible-to-Thermal Unsupervised Domain Adaptation for Object Detection	109
<i>Adrien Maglo and Romaric Audigier</i>	
Estimation of Hand-Interacting Object Poses with Boundary Guidance	125
<i>Sin-Yu Fu and I.-Chen Lin</i>	
Annotation-Free Object Detection by Knowledge-Extraction Training From Visual-Language Models	141
<i>Yasuto Nagase, Yasunori Babazaki, and Takashi Shibata</i>	

An Improved YOLOF for Scale Imbalance with Dilated Attention	156
<i>Tsatsral Amarbayasgalan, Mooseop Kim, and Chi Yoon Jeong</i>	
CLIP-Based Point Cloud Classification via Point Cloud to Image Translation	173
<i>Shuvozit Ghose, Manyi Li, Yiming Qian, and Yang Wang</i>	
BarBeR: A Barcode Benchmarking Repository	187
<i>Enrico Vezzali, Federico Bolelli, Stefano Santi, and Costantino Grana</i>	
PCGAUNet: Pixel Correlation and Gaussian Attention Driven Network for Text Segmentation	204
<i>Ayush Roy, Shivakumara Palaiahnakote, Umapada Pal, Apostolos Antonacopoulos, and Raghavendra Ramachandra</i>	
DATR: Domain Agnostic Text Recognizer	220
<i>Kunal Purkayastha, Shashwat Sarkar, Shivakumara Palaiahnakote, Umapada Pal, and Palash Ghosal</i>	
DEYOLO: Dual-Feature-Enhancement YOLO for Cross-Modality Object Detection	236
<i>Yishuo Chen, Boran Wang, Xinyu Guo, Wenbin Zhu, Jiasheng He, Xiaobin Liu, and Jing Yuan</i>	
MarUCOD: Unknown but Concerned Object Detection in Maritime Environments	253
<i>Hajung Yoon, Yoonji Lee, Hwijun Lee, Daeho Um, Hong Seok Choi, and Jin Young Choi</i>	
Identifying Impurities in Liquids of Pharmaceutical Vials	269
<i>Gabriele Rosati, Kevin Marchesini, Luca Lumetti, Federica Sartori, Beatrice Balboni, Filippo Begarani, Luca Vescovi, Federico Bolelli, and Costantino Grana</i>	
RGB-T Object Detection via Group Shuffled Multi-receptive Attention and Multi-modal Supervision	284
<i>Jinzhong Wang, Xuetao Tian, Shun Dai, Tao Zhuo, Haorui Zeng, Hongjuan Liu, Jiaqi Liu, Xiuwei Zhang, and Yanning Zhang</i>	
Enhancing Object Detection by Leveraging Large Language Models for Contextual Knowledge	299
<i>Amirreza Rouhi, Diego Patiño, and David K. Han</i>	
YOLO-RSOD: Improved YOLO Remote Sensing Object Detection	315
<i>Yang Xu and Jun Lu</i>	

All-Weather Vehicle Detection and Classification with Adversarial and Semi-Supervised Learning 330
Yi-Chao Huang and Huei-Yung Lin

Who Should Have Been Focused: Transferring Attention-Based Knowledge from Future Observations for Trajectory Prediction 346
Seokha Moon, Kyuhwan Yeon, Hayoung Kim, Seong-Gyun Jeong, and Jinkyu Kim

VA-OCC : Enhancing Occupancy Dataset Based on Visible Area for Autonomous Driving 362
Yang Li, Weng Feng, Ge Gao, Jun Chang, and Ming Li

Large Models in Dialogue for Active Perception and Anomaly Detection 371
Tzoulio Chamiti, Nikolaos Passalis, and Anastasios Tefas

Dense Road Surface Grip Map Prediction from Multimodal Image Data 387
Jyri Maanpää, Julius Pesonen, Heikki Hyyti, Iaroslav Melekhov, Juho Kannala, Petri Manninen, Antero Kukko, and Juha Hyypä

Video-Based Semi-automatic Drivable Area Segmentation 405
Zhengyun Cheng, Guanwen Zhang, Changhao Wang, and Wei Zhou

CASPFFormer: Trajectory Prediction from BEV Images with Deformable Attention 420
Harsh Yadav, Maximilian Schaefer, Kun Zhao, and Tobias Meisen

LF Tracy: A Unified Single-Pipeline Paradigm for Salient Object Detection in Light Field Cameras 435
Fei Teng, Jiaming Zhang, Jiawei Liu, Kunyu Peng, Xina Cheng, Zhiyong Li, and Kailun Yang

Author Index 453



Automatic Wood Pith Detector: Local Orientation Estimation and Robust Accumulation

Henry Marichal¹(✉), Diego Passarella², and Gregory Randall¹

¹ Instituto de Ingeniería Eléctrica, Facultad de Ingeniería, Universidad de la República, Montevideo, Uruguay

hmarichal93@gmail.com

² Centro Universitario Regional Noreste, Universidad de la República, Tacuarembó, Uruguay

Abstract. A fully automated technique for wood pith detection (APD), relying on the concentric shape of the structure of wood ring slices, is introduced. The method estimates the ring's local orientations using the 2D structure tensor and finds the pith position, optimizing a cost function designed for this problem. We also present a variant (APD-PCL) using the parallel coordinate space that enhances the method's effectiveness when there are no clear tree ring patterns. Furthermore, refining Kurthongmee's work, a YoloV8 net is trained for pith detection, producing a deep learning-based approach (APD-DL). All methods were tested on seven datasets, including images captured under diverse conditions (controlled laboratory settings, sawmill, and forest) and featuring various tree species (*Pinus taeda*, *Douglas fir*, *Abies alba*, and *Gleditsia triacanthos*). All proposed approaches outperform existing state-of-the-art methods and can be used in CPU-based real-time applications. Additionally, we provide a novel dataset comprising images of gymnosperm and angiosperm species. Dataset and source code are available at <http://github.com/hmarichal93/apd>.

Keywords: Computer vision · Wood pith detection · Deep neural network object detection · Wood quality

1 Introduction

Locating the pith of tree cross-sections is essential to identify (in basal discs) the first year of growth and, therefore, the tree's age. The pith has a different type of tissue than the rest of the tree, with distinct physical-mechanical properties. Locating the pith is useful, among other reasons, to detect growing eccentricity because in the natural process of senescence of standing trees, the fungi that

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-78447-7_1.

degrade the wood enter through the pith or because the industry discards that part as it has different uses than the rest of the wood. Moreover, some tree ring delineation algorithms are sensitive to a precise pith location [2, 14, 16, 18], mainly when those algorithms are based on the ring structure, a concentric pattern similar to a spider web as illustrated in Fig. 1d. That figure shows some examples of the diversity of images of tree slices. Ideally, the intersection point between the perpendicular lines through the tree rings should be the pith (the center of the structure, located inside the tree’s medulla). The *spider web* model is only a general approximation. Real slices include ring asymmetries, cracks, knots, fungus, etc., as seen in Fig. 1b and c. Different species produce diverse patterns. Moreover, gymnosperm, as the ones illustrated in Fig. 1, and angiosperm species produce a different wood structure, as seen in Fig. 6e. Automatic pith detection must be robust to such variations and perturbations.

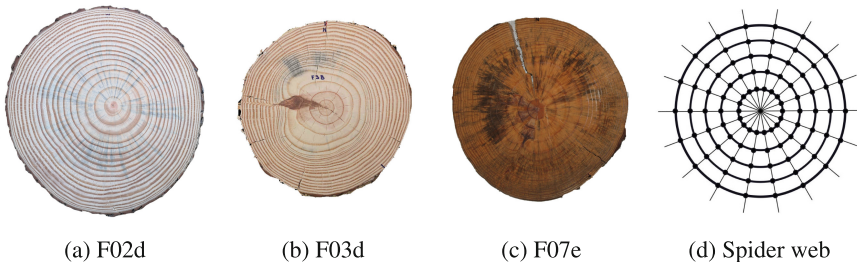


Fig. 1. Some examples from **UruDendro** dataset [15] (a to c). (d) The whole structure, called *spider web*, is formed by a *center* (the slice pith), *rays*, and the *rings* (concentric curves). In the scheme, the *rings* are circles, but in practice, they can be (strongly) deformed as long as they don’t intersect another *ring*.

This paper presents several key contributions: the release of a new challenging dataset (UruDendro2 and UruDendro3) for wood pith detection, the development of real-time automatic detection methods (APD and APD-PCL), training of a YoloV8 net (APD-DL) for the same purpose, and rigorous comparison with state-of-the-art methods on various public datasets. These contributions enhance the field of wood pith detection, offering practical solutions and insights for real-time applications.

2 Previous Work

Schraml and Uhl et al. [21] proposed a method (here called LFSA) that splits the wood cross-section into patches, estimating the patch’s orientation by 2D Fourier Transform. They accumulate the patch’s orientation using a Hough Transform approach and calculate the pith position as the maximum in the accumulation space. Kurdthongmee et al. [11] proposed the Histogram Orientation Gradient to estimate the tree ring local orientation and proceed similarly to [21].

In the same line, Norell et al. [19] proposed two ways for estimating the local orientations: quadrature filters and a Laplacian pyramids approach. Recently, Decelle et al. [3] proposed ACO, a method based on an ant colony optimization algorithm for the local orientation accumulation step.

Deep Neural Network (DNN) methods have also been applied to solve this problem. Kurdhongmeed et al. [9] compared the effectiveness of two DNN object detector models (YoloV3 and SSD MobileNet) to locate the pith. They trained the models via transfer learning over 345 wood slice RGB images captured within a sawmill environment and evaluated over a separate dataset of 215 images.

3 APD: Automatic Wood Pith Detection

We propose an automatic pith detection method based on a model of the wood slice. In a gymnosperm tree cross-section, as the ones shown in Fig. 1, two types of structures are present: the rings formed by (roughly) concentric curves and, in some cases, the presence of radial structures such as cracks and fungi. Both are fundamentally related to the pith. The former is due to the growing process of the tree, which forms the rings, and the latter is because the tree’s anatomy leads naturally to the radial characteristic of cracks and fungus growing. The principal idea of the proposed method derives from this observation: we can locate the pith at the intersection of the lines supported by radial structures and the perpendiculars to the rings.

The angiosperm tree cross-section structure is slightly different, as seen in Fig. 6e. Still, it is also formed of radially organized cells, with texture patterns appearing at different pith radii. This produces visual macrostructures that allow a similar approach to determine the pith position as depicted in the previous paragraph.

Not always do those hypotheses stand out completely. Sometimes, the ring structure can be highly (locally) deformed, as in the presence of a knot. Sometimes, there are no cracks or fungi present. However, in general, enough information is produced by the ring structure and, eventually, by the presence of cracks and fungi to estimate the pith location correctly.

Given an image of the tree cross-section, and using the *spider web* model illustrated in Fig. 1.d, the APD approach pseudocode is described at Algorithm 1. The main steps are the following (see Fig. 3 for more details):

1. *Local orientation detection (line 1 of Algorithm 1)*. To estimate the local orientation (LO), we compute the 2D-Structure Tensor [1] $ST[p]$ at each pixel p using a window of size $st_w \times st_w$. Pixels in the window are weighted by a Gaussian kernel w of parameter st_σ . The structure tensor is calculated as $ST[p] = \sum_r w[r]ST_{xy}[p - r]$ where $ST_{xy}[p]$ is defined as

$$ST_{xy}[p] = \begin{bmatrix} (I_x[p])^2 & I_x[p]I_y[p] \\ I_x[p]I_y[p] & (I_y[p])^2 \end{bmatrix}$$

Algorithm 1: APD

Input: Im_{in} , // RGB slice image;
Output: Pith location

- 1 $ST_O, ST_C \leftarrow \text{local_orientation}(Im_{in}, st_\sigma, st_w)$
- 2 $LO_f \leftarrow \text{lo_sampling}(ST_O, ST_C, lo_w, \text{percent}_{LO})$
- 3 $LO_r \leftarrow LO_f$
- 4 **for** i **in** 1 **to** max_iter **do**
- 5 **if** $i > 1$ **then**
- 6 $LO_r \leftarrow \text{filter_lo_around_c}_i(LO_f, r_f, c_i)$, // See Figure 3.e
- 7 $c_{i+1} \leftarrow \text{optimization}(LO_r)$ // Equation 4
- 8 **if** $\|c_{i+1} - c_i\|_2 < \epsilon$ **then**
- 9 $c_i \leftarrow c_{i+1}$
- 10 **break**
- 11 $c_i \leftarrow c_{i+1}$
- 12 **return** c_i

where $I_x[p]$ and $I_y[p]$ are the first derivatives of image I in point p along x and y , respectively. We can re-write the 2×2 structure tensor matrix at pixel p as:

$$ST[p] = \begin{bmatrix} J_{11} & J_{12} \\ J_{12} & J_{22} \end{bmatrix}$$

The local orientation at pixel p is:

$$ST_O[p] = \frac{1}{2} \arctan\left(\frac{2J_{12}}{J_{22} - J_{11}}\right) \quad (1)$$

The *coherence* of the LO estimation in p is given by the relative value of $ST[p]$ eigenvalues λ_1 and λ_2 (where λ_1 is the largest and λ_2 is the smallest one):

$$ST_C[p] = \left(\frac{\lambda_1 - \lambda_2}{\lambda_1 + \lambda_2}\right)^2 \quad (2)$$

The outputs of this step are two matrices: one of local orientations (ST_O) and one of coherence (ST_C).

2. *Local orientation sampling* (line 2 of Algorithm 1). The LO estimations are sampled in the following way: 1) ST_O and ST_C are divided in non-overlapping patches of size $lo_w \times lo_w$. 2) We find the pixel p^j with the highest coherence (c_{high}^j) within patch $patch_i$. A minimum patch coherence st_{th} is defined. We assign $ST_O[p^j]$ to $patch_i$ in position p^j , if $c_{high}^j > st_{th}$. To fix st_{th} , we calculate the value of ST_C such that a given percentage (parameter percent_{LO}) of the LO in the slice has $ST_C > st_{th}$. Each LO is a segment $lo_i = p_1^i p_2^i$, defined by the limits p_1^i and p_2^i . p_{LO}^i is the middle point between them (p^j). Given the local orientation $\alpha_i = ST_O[p_{LO}^i]$, points p_1^i and p_2^i are computed as $p_{1,2}^i = p_{LO}^i \pm (\cos(\alpha_i), \sin(\alpha_i))$. Suppose N patches have coherently enough

LO; the output of the step is a matrix, LO_f of size $N \times 4$. In this way, lines are supported by the LO of all meaningful structures in the cross-section, such as the rings. The pseudocode of the step is described in the supplementary material.

3. *Find the center (line 7 of Algorithm 1)*. Given the filtered local orientation matrix, LO_r , we define the following optimization problem: find c_{opt} , the geometrical position that maximizes the collinearity between the lo_i and a line passing by c_{opt} and p_{LO}^i . To this aim, we define the following cost function:

$$h(x, y) = \frac{1}{N} \sum_{i=1}^N \cos^2(\theta_i(x, y)) \quad (3)$$

Figure 2 illustrates the vectors involved in computing Eq. 3. The angle between $\overrightarrow{p_1^i p_2^i}$ and $\overrightarrow{cp_{LO}^i}$ is θ_i . The pith position c of coordinates (x, y) is the origin of a segment $\overrightarrow{cp_c^i}$. As $\cos(\theta_i(x, y)) = \frac{\langle \overrightarrow{cp_{LO}^i}, \overrightarrow{p_1^i p_2^i} \rangle}{|\overrightarrow{cp_{LO}^i}| |\overrightarrow{p_1^i p_2^i}|}$, the optimization problem to be solved becomes:

$$c_{opt} = \max_c \frac{1}{N} \sum_{i=1}^N \left(\frac{\langle \overrightarrow{cp_{LO}^i}, \overrightarrow{p_1^i p_2^i} \rangle}{|\overrightarrow{cp_{LO}^i}| |\overrightarrow{p_1^i p_2^i}|} \right)^2 \quad (4)$$

s.t. $c \in \text{Slice Region}$

4. To find the global maximum (c_{opt}) of the former optimization problem, we use the *SLSQP*¹ algorithm [8]. Problem 4 has a global maximum and is unique if it is restricted to the region of the wood cross-section. To initialize the *SLSQP* method, we use the least squares solution of finding the point c_{ini} , which minimizes the distance to all the lines in LO_r within the slice.
5. *Refinement (lines 4 to 11 of Algorithm 1)*. Once a candidate for the pith location c_{opt} is obtained, the optimization procedure (4) is repeated using only the local orientations within a squared region of size $Size_{image}/r_f$ centered in c_{opt} (see Fig. 3.e). This step is repeated until the pith location doesn't move more than a given tolerance ($\epsilon = 10^{-5}$) or the iteration counter reaches $max_iter = 5$. This approach avoids distortions introduced by an asymmetric tree ring growth pattern.

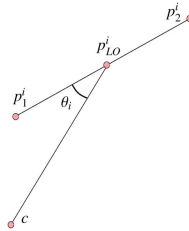


Fig. 2. Cost function definitions.

¹ Using the python implementation in `scipy.optimize.minimize` method.

4 APD-PCL: PCLines Based Automatic Wood Pith Detection

The APD method described in Sect. 3 works fine when the ring structure gives enough information. In some (rare) cases, the ring structure is not visible due to fungi or other perturbations. In those cases, it is possible to solve the same problem using the lines supported by the radial structure of those perturbations and the lines produced by the ring structure. Hence, the APD-PCL version of the method is more robust and allows for the successful treatment of cross-sections with highly degraded ring patterns. The price to pay is a slower algorithm, as it includes a RANSAC-based clustering step.

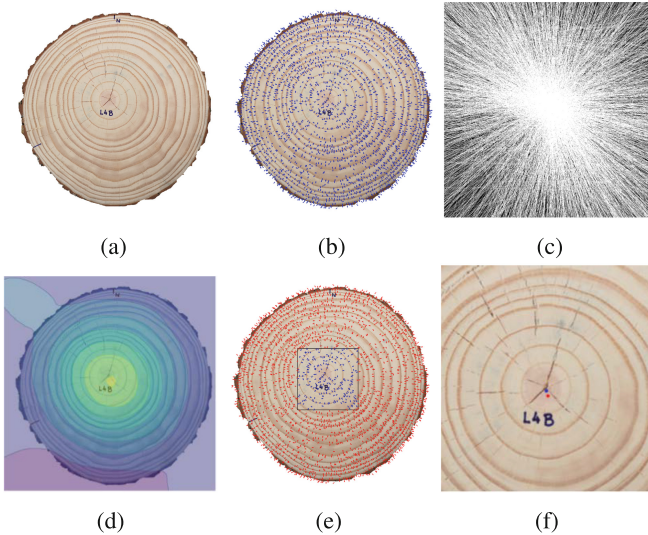


Fig. 3. Principal steps of APD method (L04d image from UruDendro2 collection). (a) Resized slice image, without background; (b) Sampled LO produced by the Structure Tensor estimation; (c) Accumulation space defined by the LO supported lines; (d) Plot of the cost function (Eq. 3), highest values in yellow; (e) Sub image built around the solution c_1 obtained after the first iteration; (f) Evolution of c_i . The final solution is in blue; previous iterations' solutions are in red. (Color figure online)

The APD-PCL method selects which local orientations to consider in the optimization problem of Eq. 4. In general, the estimation made by the structure tensor calculation step is determined by the rings. Different perturbations also produce some LO, but its number is minimal, and the lines they support don't converge to the pith. In some (rare) cases, the perturbations are so important that they overshadow the ring structure. In those cases, the number of LO produced by the perturbations is more significant than those produced by the ring structure. The set of perturbations-related LO can be of diverse origin: knots,

fungi, cracks, and noise. Some of them (as fungi and cracks) have a typical radial orientation, so the perpendicular lines to its LO converge to the pith. Considering this, we modify Algorithm 1, by including a post-processing step over matrix LO_f , between lines 2 and 3. The rest of the algorithm is the same:

1. Use the PClines transform [4] to convert each line into a point.
2. The PClines space is formed by two sub-spaces defined by a parameter d : the *straight space* includes lines with orientations $\alpha_i \in [0, \frac{\pi}{2}]$ and the *twisted space* lines with orientations $\alpha_i \in [\frac{\pi}{2}, \pi]$. As seen in Fig. 4, convergent lines in the Euclidean space correspond to aligned points in the PClines spaces. This allows the following steps:
 - (a) Lines supported by LO produced by the ring structure converge somewhere around the pith. They produce a line-shaped cluster in the PClines spaces (Figs. 4b and c). Working only in the $[-d, 0]$ and $[0, d]$ ranges for the twisted and straight sub-spaces, we select the aligned points using a RANSAC [5] approach. This avoids the use of points near the infinity. We select the converging lines in each sub-space, excluding those simultaneously selected in both.
 - (b) The previous step clusters all convergent LO_f in the image producing the set LO_{ring} . We rotate by 90° all the orientations in LO_f and repeat the previous procedure to detect the converging ones. These rotated converging lines cluster, LO_{radial} , are produced by cracks, fungi, or similar structures. Adding both gives the set:

$$LO_f^{PClines} = LO_{ring} + LO_{radial}$$

- (c) To make the line segment selection method more robust, we add a third PClines transform using the lines supported by $LO_f^{PClines}$. Most ring-related LO and rotated LO generated by radial structures are expected to converge (hence, to form a line cluster in the PCline space). Therefore, most of the outliers should be removed at this step.

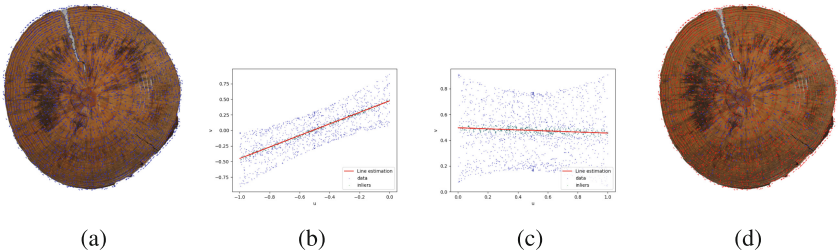


Fig. 4. Use of PClines to cluster converging local orientations for slice F07e. (a) Local orientations; (b) Selection of the converging segments in the twisted space using RANSAC to fit a line (in red). Inliers are colored in green; (c) The same procedure is applied in the straight space; (d) In blue, the converging LO (inliers from both sub-spaces) and the LO to be removed in red. (Color figure online)

Figure 5 illustrates the considered lines and the accumulation space of Eq. 3 without and with the PClines step. Note how the method filters out many non-convergent lines and regularizes the cost function.

The APD-PCL method is similar to the APD one, but the PClines-based filtering step diminishes the number of considered lines, filtering out many non-convergent ones.

5 APD-DL: Deep Learning Based Automatic Wood Pith Detection

In Sects. 3 and 4, we tackle the pith detection problem using a *spider web* model, as in the “classic” image processing approaches. Now, we present a Deep-learning approach that learns the model from the data. Inspired by Kurdthongmee et al. [10], who used a YoloV3 model, we train a YoloV8 [6] network using the datasets described in Sect. 6. This is an architecture tailored for object detection and segmentation. To train the model, we need to supply a dataset of wood cross-section images, each labeled with the ground truth pith location indicated by a bounding box, where the bounding box size is one-tenth of the image dimensions.

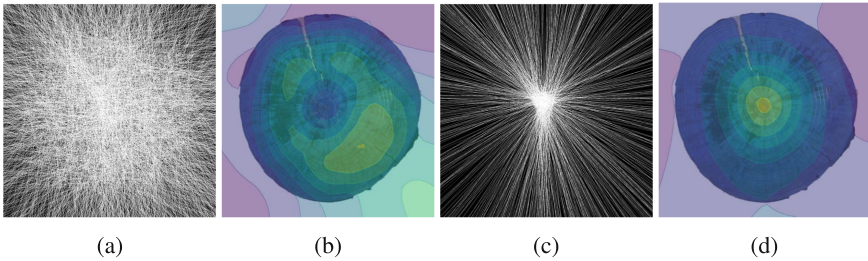


Fig. 5. LO Accumulation space and cost function for slice F07e with and without applying the PClines filtering method. (a) LO Accumulation space without filtering; (b) cost function without filtering; (c) LO Accumulation Space with PClines filtering; (d) cost function with PClines filtering.

We divide the data into five sets and use five-fold cross-validation. In each fold i , we use one set ($test_i$) for testing and the other four for training. The training process in each fold is done as usual, and we use the produced model to label the data in $test_i$. The process is repeated for all the folds. In the end, we have predictions for all the data, and in each case, the used model was generated without the influence of the $test_i$ data. With the predictions for all images produced in this manner, we can deliver the metrics to determine the method’s performance.

Table 1 shows the results using normalized errors (see Sect. 7.2). Training with such a high diversity of data produces state-of-the-art results. Results over each row (collection) are calculated using the predictions and the bounding box

Table 1. Prediction results of 5-fold cross-validation for the APD-DL. The second to fourth columns show the Mean (and standard deviation in parenthesis), Median, and Maximum normalized error (defined in Sect. 7.2) values. The last column shows the false negatives. We use all the datasets together to train the model and calculate the performance within each dataset.

Collection	Mean (Std)	Median	Maximum	FN
Uru2	0.55 (1.45)	0.18	11.32	2
Uru3	0.13 (0.06)	0.13	0.27	0
Kennel	0.14 (0.07)	0.13	0.24	0
Forest	0.45 (1.85)	0.12	13.91	0
Logyard	0.52 (1.29)	0.27	7.51	0
Logs	0.22 (0.46)	0.13	4.42	1
Discs	0.23 (0.54)	0.14	5.67	0
All	0.33 (1.01)	0.14	13.91	3

center, produced during the five-fold cross-validation with all the images in the seven datasets. In some rare cases, this approach doesn’t give a prediction (hence a false negative). In those situations, the method provides the center of the image as the pith position. This explains the relatively large value of the Maximum error and the differences between the Mean and Median errors. Besides the rare false negatives, the results are excellent. The last row depicts the results for all the collections.

Hyperparameters The algorithm was trained with 640 pixels width images (keeping the aspect ratio), using a batch size of 16, for 100 epochs, with the optimizer AdamW [13] ($lr = 0.002$, $momentum = 0.9$) and yolov8n as pre-trained weights. All the network was re-trained.

6 Datasets Description

We use the following datasets:

- UruDendro. We introduce here a new public dataset with two collections of wood cross-section samples with experts’ annotated ground truth [15]:
 - UruDendro2: 119 RGB images of *Pinus taeda* slices. This collection includes 64 images taken under different illumination conditions and cameras, published in 2022 on our website [15], now increased with 55 new images taken in laboratory conditions, with an iPhone 6S phone (12 Mpx camera) at a distance between 43 and 51 cm from the slice, under controlled illumination with a led ring of 35 W. Size images range between 1000 and 3000 pixels in width. Slice’s surface presented different conditions: some were cut by chainsaw, smoothed by a handheld planner, and polished with a rotary sander. All images are annotated by at least one expert with the position of the pith.

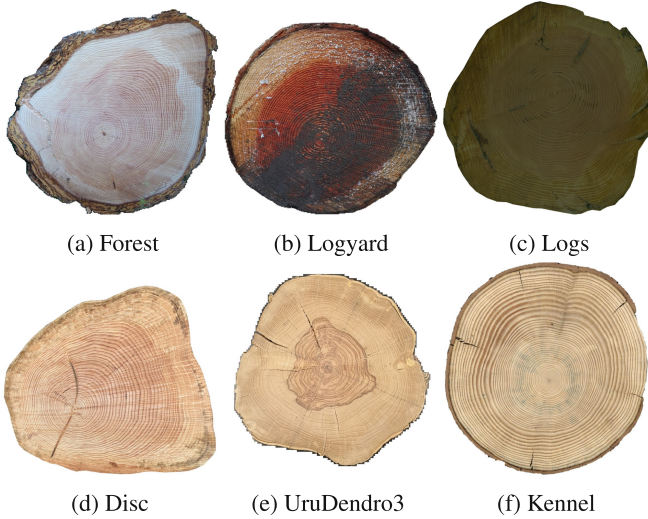


Fig. 6. Examples of the used datasets. Species are (a–d) *Douglas fir*, (e) *Gleditsia triacanthos*, (f) *Abies alba*. Acquisition conditions: (a–c) in the field, with a smartphone camera; (d–e) in the laboratory, with controlled illumination. The samples (d–e) were previously sanded and polished. Images (a–c) didn’t have any special treatment.

- UruDendro3: 9 RGB images of *Gleditsia triacanthos*, an angiosperm, acquired in a laboratory, without illumination-controlled conditions, using a Huawei P20 Pro smartphone (24 Mpx camera) at a distance of approximately 1 m from the slice. Size images range between 1000 and 2000 pixels in width. All the slices were polished. All these images are annotated by at least one expert with the position of the pith.
- Kennel [7]. A public dataset with 7 RGB 1280 pixels squared images of *Abies alba*, polished and acquired in controlled illumination laboratory conditions. The pith pixel location is provided as metadata.
- TreeTrace [12]. It is a public dataset, with samples of *Douglas fir* taken at different stages of the wood process chain, and the pith pixel location provided as metadata. Each image has several wood slices. To build the collections, we extract sub-images containing one slice each, producing images between 1000 and 3000 pixels in width. This dataset includes the following collections:
 - Forest, 57 RGB images taken from the freshly cut logs with a digital camera.
 - Logyard, 32 RGB images of the same log ends, acquired with a smartphone in the sawmill courtyard several days after the cutting.
 - Logs, 150 RGB images acquired in the sawmill with a smartphone.
 - Discs, 208 RGB images acquired with a 400 dpi scanner from sanded and polished slices after several weeks of air-drying.

Table 2 summarize the used datasets. Figure 1 show images from the UruDendro2 dataset, and Fig. 6 show examples from the other collections. These

Table 2. Datasets description.

Collection	Number of images	Specie
UruDendro2	119	<i>Pinus taeda</i>
UruDendro3	9	<i>Gleditsia triacanthos</i>
Kennel	7	<i>Abies alba</i>
Forest	57	<i>Douglas fir</i>
Logyard	32	<i>Douglas fir</i>
Logs	150	<i>Douglas fir</i>
Discs	208	<i>Douglas fir</i>

datasets convey a high degree of variability. It includes examples of gymnosperm (*Pinus taeda*, *Abies alba* and *Douglas fir*) as well as angiosperm (*Gleditsia triacanthos*). Acquisition conditions are also diverse, including images obtained with a smartphone in the forest and the sawmill. Samples were acquired in the field with dirt, sap, or saw marks, and others were obtained in controlled illumination conditions in the laboratory from polished samples. The samples include perturbations as fungi, cracks, and knots, as can be seen in Fig. 1 and sap and saw marks (Fig. 6b and Fig. 6d). All have the ground truth position of the pith. Considering all datasets, we work with 582 images.

7 Results and Discussion

7.1 Preprocessing

Sometimes, the images are acquired in the field with a smartphone camera, and one image can contain more than one cross-section. Regardless of the method used, all images are preprocessed in such a way as to standardize the image input:

1. *Background subtraction.* Produce a new image limited to one slice. To this aim, when possible, we filter out the background using the mask provided in the datasets. If the mask is not provided, we use a deep learning-based method [20], which uses an U^2Net to segment salient objects.
2. *Resize the image.* This step, which is not strictly necessary, allows us to fix the algorithm’s parameters once and for all. All images are resized to 640 pixels width, respecting the original image’s aspect ratio using Lanczos interpolation.²

² We impose this restriction due to the GPU memory limitations encountered during the training of the APD-DL method.

7.2 Normalized Errors

Given the cross-sections’ diverse dimensions, presenting the errors in pixels is not informative. Additionally, not all datasets provide millimeter pixel relations. We use the percentage of the equivalent slice radius. Given a prediction P_i and a Ground Truth GT_i , this error is calculated as follows:

$$Err_i = \frac{100 \times Dist(P_i, GT_i)}{Equivalent_radii(image_i)}$$

Where $Dist(P_i, GT_i)$ is the Euclidean distance between the prediction and the Ground Truth, in pixels, remember that the pith is modeled as a point in the image within this work. Therefore, $Dist(P_i, GT_i)$ is the distance between points. $Equivalent_radii(image_i)$ (in pixels) is half the biggest side of the rectangle that circumscribes the slice.

7.3 Experiments

The method to fine-tune the APD-DL method is explained in Sect. 5. To determine the best parameters’ values for ACO, LFSA, APD, and APD-PCL, we minimize the average of Euclidean distances between ground truth and predictions for all used datasets.

For the APD and APD-PCL methods, the parameters $r_{ansac_outlier_th}$ (0.03), st_σ (1.2) and r_f (7) were set after experiments over a few images. The fixed values are shown in parentheses. The rest of the parameters, $percent_{LO}$, st_w and lo_w were set searching the minimum over the following grid: $percent_{LO}$ in [0.3, 0.5, 0.7, 0.9], st_w in [3, 7, 9, 11] and lo_w in [3, 7, 9, 11].

Inferences were made using an Intel Core i5 10300H workstation with 16 GB and a GPU GTX1650 (when needed).

7.4 Results

In this section, a performance comparison is made between the mentioned methods. Table 3 shows the performance of the proposed methods and two state-of-the-art ones [3, 21], over the datasets presented in Sect. 6. To compare different-size wood cross-sections, we use the mean error and standard deviation using

Table 3. Results on all the datasets. Normalized errors. We show the mean error and the standard deviation between parenthesis.

	UruDendro2	UruDendro3	Kennel	Forest	Logyard	Logs	Discs
LFSA [21]	1.03 (0.85)	1.46 (0.97)	0.42 (0.18)	0.80 (0.36)	1.02 (0.62)	0.80 (0.46)	0.72 (0.43)
ACO [3]	2.23 (6.64)	4.52 (11.96)	0.2 (0.06)	0.24 (0.24)	0.60 (1.11)	0.46 (0.45)	0.24 (0.35)
APD-PCL	0.42 (0.34)	0.74 (0.54)	0.19 (0.10)	0.81 (0.98)	0.82 (0.84)	0.52 (0.47)	0.46 (0.57)
APD	1.02 (2.45)	0.55 (0.30)	0.14 (0.06)	0.22 (0.18)	0.35 (0.17)	0.29 (0.33)	0.26 (0.42)
APD-DL	0.55 (1.45)	0.13 (0.06)	0.14 (0.07)	0.45 (1.85)	0.52 (1.29)	0.22 (0.46)	0.23 (0.54)

normalized errors. The performance of the methods differs for each collection due to its specific characteristics regarding species, acquisition conditions, etc. Note that ACO was developed (and tailored) for the TraceTree collections. Its performance degrades when tried on other species (such as UruDendro collections). LFSA performance is more regular across collections. The three methods proposed in this paper outperform ACO and LFSA on all collections. APD and APD-DL perform better for almost all collections. APD outperforms APD-PCL for all collections except UruDendro2, which has some images with fungi and cracks overshadowing the ring structure. Note that in all the cases, the precision of the pith detection is very high.

Table 4. Results of all the methods over the whole set of images, i.e., merging all collections. Normalized errors, number of false negatives, and execution time in milliseconds.

Method	Mean	Median	Max	FN	Time
LFSA [21]	0.83	0.72	5.03	0	627
ACO [3]	0.79	0.21	36.39	2	918
APD-PCL	0.52	0.34	4.33	0	2339
APD	0.42	0.19	15.44	0	784
APD-DL	0.33	0.14	13.91	3	209

Table 4 compares the performance of all tested methods using the 582 images of all collections. All methods presented in this paper outperform LFSA and ACO methods. The APD performance is surpassed only by the APD-DL method but at the cost of some false negatives: images in which APD-DL didn’t find a solution. We can see that APD slightly outperforms the APD-PCL method. This is due to the RANSAC algorithm used to cluster points in the PCLines space. When there is no apparent clustering of points around a line, RANSAC tries to fit a line anyway, selecting a wrong set of LO and producing a wrong pith localization. This situation sometimes appears in the TreeTrace dataset. Considering each method’s mean processing time per image, we must stress that APD-DL and ACO methods run on GPU, while APD, APD-PCL, and LFSA run on a CPU machine. Note that APD is roughly three times faster than APD-PCL. All in all, it is remarkable that the “classic style” model-based proposed methods (APD and APD-PCL) and a Deep Learning one (APD-DL) have similar performance and execution times, allowing real-time applications with a CPU in the APD and APD-PCL cases. In the supplementary material, we add showcases illustrating how the different methods work under extreme conditions.

8 Conclusions and Future Work

This paper addresses the wood pith detection on tree slices problem using classic image processing and machine learning-based approaches. Both approaches

are determined by the characteristics of the data used to tune the algorithm. In search of a more general solution, we use a set of diverse datasets, which spans different species, acquisition conditions, and perturbations (from cracks and knots to saw marks and dirt for images acquired on the field).

We proposed three real-time methods. The first two are based on a *spider web* model in a classic image processing approach, and the third one is a Deep Learning method. The former has excellent performance and runs in real-time on a CPU-based machine, and the model allows a clear comprehension of the approach. The limited number of parameters is understandable and can be fixed once and for all. The latter has better (although similar) performance but has some false negatives and is more opaque concerning the meaning of its millions of parameters.

The UruDendro dataset, with annotated images of *Pinus taeda* (a gymnosperm) and *Gleditsia triacanthos* (an angiosperm), are presented and can be used by the community to test other approaches to this problem.

Acknowledgments. The experiments presented in this paper used ClusterUY [17] (site: URL: <https://cluster.uy>). We had valuable conversations with J. M. Morel and J. Di Martino. This work was supported by project ANII-FMV-176061.

References

1. Bigun, J., Granlund, G., Wiklund, J.: Multidimensional orientation estimation with applications to texture analysis and optical flow. *IEEE Trans. Pattern Anal. Mach. Intell.* **13**(8), 775–790 (1991). <https://doi.org/10.1109/34.85668>
2. Cerda, M., Hitschfeld-Kahler, N., Mery, D.: Robust tree-ring detection. In: Mery, D., Rueda, L. (eds.) *PSIVT 2007*. LNCS, vol. 4872, pp. 575–585. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-77129-6_50
3. Decelle, R., Ngo, P., Debled-Rennesson, I., Mothe, F., Longuetaud, F.: Ant colony optimization for estimating pith position on images of tree log ends. *Image Process. On Line* **12**, 558–581 (2022). <https://doi.org/10.5201/ipol.2022.338>
4. Dubska, M., Herout, A., Havel, J.: PClines—line detection using parallel coordinates. In: *Proceedings/CVPR, IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1489–1494 (2011). <https://doi.org/10.1109/CVPR.2011.5995501>
5. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM* **24**(6), 381–395 (1981). <https://doi.org/10.1145/358669.358692>
6. Jocher, G., Chaurasia, A., Qiu, J.: Ultralytics YOLOv8 (2023). <https://github.com/ultralytics/ultralytics>
7. Kennel, P., Borianne, P., Subsol, G.: An automated method for tree-ring delineation based on active contours guided by DT-CWT complex coefficients in photographic images: application to *Abies alba* wood slice images. *Comput. Electron. Agric.* **118**, 204–214 (2015). <https://doi.org/10.1016/j.compag.2015.09.009>
8. Kraft, D.: *A Software Package for Sequential Quadratic Programming*. Deutsche Forschungs- und Versuchsanstalt für Luft- und Raumfahrt Köln: Forschungsbericht, Wiss. Berichtswesen d. DFVLR (1988). <https://books.google.com.uy/books?id=4rKaGwAACAAJ>

9. Kurdthongmee, W.: A comparative study of the effectiveness of using popular DNN object detection algorithms for pith detection in cross-sectional images of para-wood. *Heliyon* **6**(2), e03480 (2020). <https://doi.org/10.1016/j.heliyon.2020.e03480>. <https://www.sciencedirect.com/science/article/pii/S240584402030325X>
10. Kurdthongmee, W., Suwannarat, K.: Locating wood pith in a wood stem cross sectional image using yolo object detection. In: 2019 International Conference on Technologies and Applications of Artificial Intelligence (TAAI), pp. 1–6 (2019). <https://doi.org/10.1109/TAAI48200.2019.8959823>
11. Kurdthongmee, W., Suwannarat, K., Panyuen, P., Sae-Ma, N.: A fast algorithm to approximate the pith location of rubberwood timber from a normal camera image. In: 2018 15th International Joint Conference on Computer Science and Software Engineering (JCSSE), pp. 1–6 (2018). <https://doi.org/10.1109/JCSSE.2018.8457375>
12. Longuetaud, F., et al.: Traceability and quality assessment of Douglas fir (*Pseudotsuga menziesii* (Mirb.) Franco) logs: the TreeTrace_Douglas database. *Ann. For. Sci.* **79**(46) (2022). <https://doi.org/10.1186/s13595-022-01163-7>. <https://hal.science/hal-03658479>
13. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization (2019)
14. Makela, K., Ophelders, T., Quigley, M., Munch, E., Chitwood, D., Dowtin, A.: Automatic tree ring detection using Jacobi sets (2020). <https://doi.org/10.48550/ARXIV.2010.08691>. <https://arxiv.org/abs/2010.08691>
15. Marichal, H., et al.: UruDendro: An Uruguayan Disk Wood Database For Image Processing (2023). <https://iie.fing.edu.uy/proyectos/madera>
16. Marichal, H., Passarella, D., Randall, G.: CS-TRD: a cross sections tree ring detection method (2023)
17. Nesmachnow, S., Iturriaga, S.: Cluster-UY: collaborative scientific high performance computing in Uruguay. In: Torres, M., Klapp, J. (eds.) ISUM 2019. CCIS, vol. 1151, pp. 188–202. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-38043-4_16
18. Norell, K.: An automatic method for counting annual rings in noisy sawmill images. In: Foggia, P., Sansone, C., Vento, M. (eds.) ICIAP 2009. LNCS, vol. 5716, pp. 307–316. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-04146-4_34
19. Norell, K., Borgfors, G.: Estimation of pith position in untreated log ends in sawmill environments. *Comput. Electron. Agric.* **63**(2), 155–167 (2008). <https://doi.org/10.1016/j.compag.2008.02.006>. <https://www.sciencedirect.com/science/article/pii/S016816990800077X>
20. Qin, X., Zhang, Z., Huang, C., Dehghan, M., Zai'ane, O.R., Jägersand, M.: U²-net: going deeper with nested u-structure for salient object detection. *CoRR* abs/2005.09007 (2020). <https://arxiv.org/abs/2005.09007>
21. Schraml, R., Uhl, A.: Pith estimation on rough log end images using local Fourier spectrum analysis. In: Proceedings of the 14th Conference on Computer Graphics and Imaging (CGIM 2013), Innsbruck, AUT. vol. 10, pp. 2013–797. Citeseer (2013)



A Multi-ground Truth Approach for RGB-D Saliency Detection

Nguyen Trung Thinh Huynh¹, Van Linh Pham¹, Xuan Toan Mai²,
and Tuan Anh Tran²(✉)

¹ Viettel Artificial Intelligence and Data Services Center, Viettel Group, Lot D26
Cau Giay New Urban Area, Yen Hoa Ward, Cau Giay District, Hanoi, Vietnam

² Faculty of Computer Science and Engineering, Ho Chi Minh City University of
Technology (HCMUT), VNU-HCM, Ho Chi Minh City, Vietnam
{mxtoan, trtanh}@hcmut.edu.vn

Abstract. Segmenting the most prominent objects in a scene using a pair of color and depth images requires the model to learn effective multimodal fusion. Despite an explosive number of recent studies, a significant problem remains underestimated: datasets have been labeled from people’s subjectivity, thus lacking consistency in determining the most prominent objects, while one picture can contain numerous sets of salient objects. To tackle this issue, we propose a multi-ground truth approach for RGB-D Saliency Detection (dubbed S-MultiMAE) that combines multi-perspective tokens to guide the model to create various desirable predictions and a masked autoencoding pretraining task (inherits MultiMAE) to achieve a superior multi-model synthesis of color and depth images. We conducted extensive analyses on both multi- and single-ground truth benchmarks on the COME15K dataset to demonstrate the effectiveness of our proposed method. The source code is available at <https://github.com/thinh-re/s-multimae>.

Keywords: RGB-D · Saliency Detection · Multimodal

1 Introduction

The main goal of Salient Object Detection (SOD) is to find and segment the most visually prominent objects in a scene. Multi-ground truth SOD involves each scene having different sets of salient objects, hence multiple ground truths per scene. RGB-D salient object detection (RGB-D SOD) task uses two modalities: a color modality (provide texture information) and a depth modality (provide geometric structures and extra contrast cues). Additional depth maps contribute crucial supplemental information for handling complex environments, such as low-contrast salient objects with similar appearances to the background. Convolutional neural network (CNN) approaches (e.g., BBS-Net [13], CMINet [46]) achieve significant RGB-D SOD performance by fusing color (RGB) information and additional depth information. Recently, Transformer [38] was used in

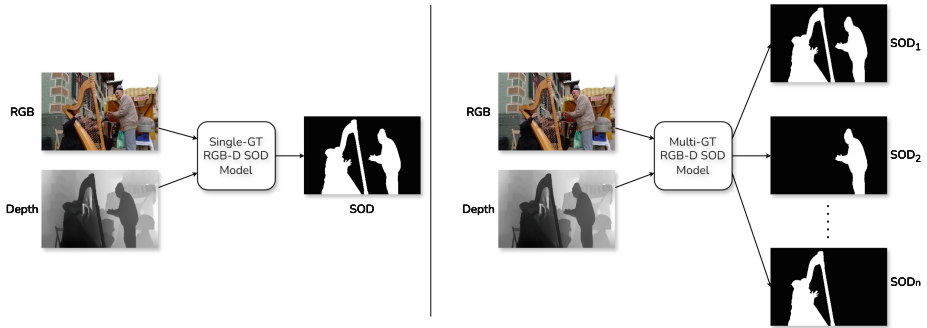


Fig. 1. Single-ground truth approach (single-GT, *left*) and Multi-ground truth approach (multi-GT, *right*)

TriTransNet [24], GroupTransNet [14], and SwinNet [23] to learn better global long-range semantic information.

Despite an enormous number of recent studies, two major issues are still underappreciated: Firstly, there has been an inconsistent convention about determining which objects are the most attractive in complex scenes. This leads to one scenario that may have multiple accepted salient objects. Inconsistent ground truth annotations can perplex the model during training, leading to undesirable outcomes like blurred regions or wrong salient objects. Most existing RGB-D SOD models [40, 41, 43, 44] are rigid, predict only one saliency map per scene. Secondly, the majority of RGB-D SOD models operated a feature extractor separately for each modality and a middle multi-level features fusion method. This design restricted the model to initializing pre-trained weights from solely single-modal pretraining, thus requiring the creation of a successful fusion model to incorporate information from color and depth branches. Thanks to a multi-modal masked auto-encoding pretraining technique (influenced by MultiMAE [2]), during pretraining, the model learns feature extraction and multimodal fusion simultaneously and generalizes satisfactorily for downstream tasks (e.g., RGB-D SOD task).

Our main contributions can be summarized as follows:

- We introduce a novel S-MultiMAE method that combines the powerful multi-modal fusion model inherited from MultiMAE [2] and effective multi-perspective signals to address the Multi-ground truth RGB-D SOD task.
- The fusion model learns cross-modal predictive coding among two modalities (color and depth images), which improves the robustness of saliency predictors. The multi-perspective signals guide the model to output desirable sets of salient objects in each scene, overcoming the limitation of inconsistent ground truth annotations in complex scenes.
- Extensive experiments demonstrate our proposed method’s effectiveness in multi-ground and single-ground truth benchmarks on the COME15K dataset.

2 Related Works

2.1 RGB-D Salient Object Detection

Combining the depth and color modalities has significantly improved salient object detection since the depth image provides more reliable spatial structure information and is insensitive to the variations of light and colors.

Different from an instance segmentation task, which aims to predict the regions of all instances belonging to predefined classes that appear in the scene, SOD focuses only on a smaller set of objects that are visually attracted and prominent in a scene. In addition, the original instance segmentation task [17] is often limited by the number of predefined classes (e.g., cats, dogs), while SOD is not bounded by predefined classes since any object can potentially become prominent in a scene, ranging from common objects (e.g., people, cats, and dogs) to rare objects (e.g., musical instruments, ancient artifacts, toys, and glass candle holders).

The limitation of CNN in learning global long-range dependencies directs methods toward Transformer-based architecture. Due to its high computational cost, TriTransNet [24], GroupTransNet [14], and CAVER [29] surrogated Transformer architecture make it suitable for RGB-D SOD tasks. TriTransNet proposes the triple transformer embedded module to learn cross-layer long-range dependencies to enhance high-level features. GroupTransNet proposed a Group Transformer Network in which energy weights outside groups pursue various features, whereas energy weights within groups pursue the consistency of features. CAVER constructed a top-down Transformer-based information propagation path by cascading several cross-modal integration units. SwinNet [23] made use of the Swin Transformer backbone, which absorbed CNN’s local advantage and the Transformer’s merit of long-range dependency.

2.2 Self-supervised Representation Learning

Over the past few years, the focus has steadily changed from pretraining models in a supervised learning manner (e.g., image classification) to self-supervised learning (SSL) by leveraging massive unlabeled datasets. Many pretraining tasks have been applied for self-supervised learning, such as image inpainting [30], clustering [5], and image colorization [19]. Practical SSL pretraining tasks can promote models to learn useful semantic features for downstream tasks.

A masked autoencoder, also known as a denoising autoencoder [39], predicts a property of masked input from unmasked input content. Masked autoencoders revived success in recent MAE [16] inspires numerous works to apply it in various applications, such as video (VideoMAE [37]), generative models (MAGE [21]), multi-model multi-task learning (MultiMAE [2]).

Original MAE learns to reconstruct missing pixels in randomly masked image patches using only input from a visible subset of patches. MultiMAE applied this technique to multiple modalities (e.g., color, depth, and semantic segmentation). Inspired by MultiMAE, our method applies pretraining to dual-modality (color and depth images).

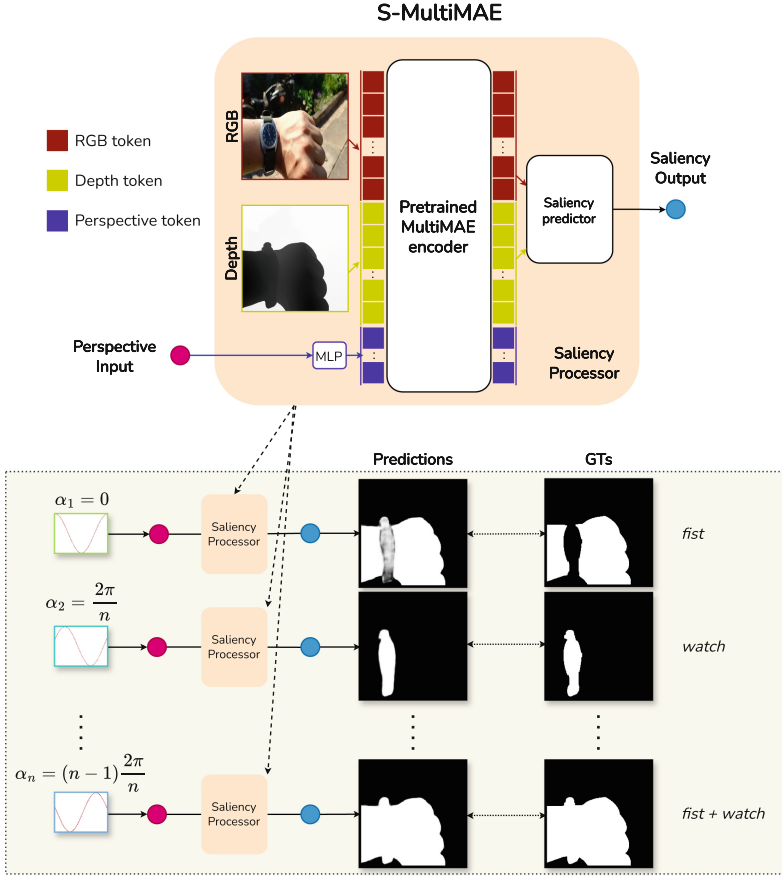


Fig. 2. Our proposed method for Multi-Ground truth RGB-D SOD with different perspective signals (generated from a cosine function) going through a shared saliency processor to produce different saliency outputs.

3 Our Proposed Method

In this section, we explain how multi-perspective signals are used to solve Multi-Ground truth RGB-D SOD tasks (Sect. 3.1), review the optimized version of MultiMAE used in the shared saliency processor (Sect. 3.2), and finally describe a pretraining technique (Sect. 3.3).

3.1 Multi-perspective Signals

Most existing single-GT RGB-D SOD models modeled classification as the probability of each output pixel given some input, $Pr_{\theta}(y|I)$ parameterized by the weights θ , where I is a series of color and depth features, and y is a binary class (i.e., $y = 1$ when this pixel belongs to salient objects, otherwise $y = 0$). To adapt

to multi-ground truth, we model saliency detection as a conditional generation by appending a perspective signal, P , such that the model learns to maximize the likelihood of the correct y per pixel, $Pr_{\theta}(y|[I, P])$.

The j -th scene has n_j perspective signals, each corresponding to one set of salient objects (i.e., one GT). The number of perspective signals varies among various scenes depending on their complexity. (e.g., in the COME15K dataset [46], a simple scene can contain only one set of prominent objects, but a highly complicated one can contain up to five sets).

Each perspective signal is a sequence of numbers sampled from a cosine function in one complete cycle. To adapt the perspective signals for different numbers of sets of salient objects (i.e., multi-GT), we distribute their starting phase evenly within a single cycle. For instance, if a scene has three GTs, three perspective signals have the starting phases at 0 , $\frac{\pi}{3}$, and $\frac{2\pi}{3}$, respectively. This constraints the maximum number of GTs the model can generate in each scene and implicitly forces the model to learn the underlying number of possible sets of salient objects that each scene can have.

To make the multi-perspective signals less rigid, during the training process of RGB-D SOD tasks, each signal is sampled from a Gaussian distribution (i.e., $\mathcal{N}(\mu, \sigma^2)$) around its phase. For instance, with the scene having 3 GTs, the second perspective signal is selected randomly from the Gaussian distribution $\mathcal{N}(\mu = \frac{\pi}{3}, \sigma^2)$.

The formula for the i -th perspective signal in the j -th scene is built as follows:

$$MLP\left(\left[\begin{array}{l} A \times \cos\left(2\pi \times \frac{0}{d} + i \times T\right), \\ A \times \cos\left(2\pi \times \frac{1}{d} + i \times T\right), \\ \dots \\ A \times \cos\left(2\pi \times \frac{d-1}{d} + i \times T\right) \end{array}\right]\right) \quad (1)$$

where A is the amplitude of a cosine function, d indicates the token dimension, i is the order of the perspective token, and $T = \frac{2\pi}{n_j}$ is the difference in the starting phase.

3.2 Saliency Processor

Our saliency processor utilizes a Vision Transformer encoder [9] for a cross-modal fusion and ConvNeXt [25] as a saliency predictor. The saliency processor’s weights and a pair of color and depth images are shared among multiple perspective signals for each scene. Firstly, color and depth images are split into fixed-size patches, which subsequently are added 2d positional embeddings. These tokens are concatenated in the following order: color tokens, depth tokens, and perspective tokens. Then, the encoding tokens, color, and depth tokens are fed to the decoder to predict the regions of salient objects, whereas all perspective tokens are disregarded.

3.3 Pretraining Multimodal Masked Autoencoding Task

Similar to MultiMAE [2], we randomly select a small portion of patches from input modalities and encode them using a ViT encoder. The objective is to reconstruct the masked-out patches using task-specific decoders. The pretraining MultiMAE helps the model learn spatial predictive coding (in-painting within RGB and depth images) and cross-modal predictive coding (reconstructing tasks from multiple input modalities).

3.4 Loss Function

Given the color image $X \in \mathbb{R}^{H \times W \times 3}$ and its corresponding depth image $Y \in \mathbb{R}^{H \times W \times 1}$, by conditioning on perspective tokens $Z_i \in \mathbb{R}^{d \times N_P}$, our model predicts a saliency map $S_i \in [0, 1]^{H \times W \times 1}$. Let $G_i \in \{0, 1\}^{H \times W \times 1}$ denote the i -th binary ground-truth saliency map. Our loss is the binary cross entropy loss (BCE) between the predicted saliency maps and their corresponding binary ground-truth saliency maps.

$$\mathcal{L} = BCE(S_i, G_i) = -[G_i \log(S_i) + (1 - G_i) \log(1 - S_i)] \quad (2)$$

where $S_i = f_\theta(X, Y, Z_i)$. The loss is performed on the pixel level as we treat all pixels equally.

4 Experiments

4.1 COME15K Benchmark

Table 1. Percentages of the number of ground truths in the COME15K dataset.

	1 GT	2 GTs	3 GTs	4 GTs	5 GTs
COME8K (8025 samples)	77.61%	1.71%	18.28%	2.24%	0.16%
COME-E (4600 samples)	70.5%	1.87%	21.15%	5.70%	0.78%
COME-H (3000 samples)	62.3%	2.00%	25.63%	8.37%	1.70%

The COME15K dataset [46] contains a total of 15,625 samples, including COME-8K (8025 samples), “normal” subdataset COME-E (4600 samples), and “difficult” subdataset COME-H (3000 samples) under multi-ground truth annotations (Table 1). Each scene contains a pair of color and depth images, and a fixed number of ground truths. This dataset allows us to explore the capacity of RGB-D Saliency Detection models when scenes become more diverse and complex. We evaluate on ten datasets (approximately 12K samples): COME-E, COME-H, DES [7], DUT-RGBD [33], LFS [20], NJU2K [18], NLPR [31], ReDWeb-S [22], SIP [12], and STEREO [27].

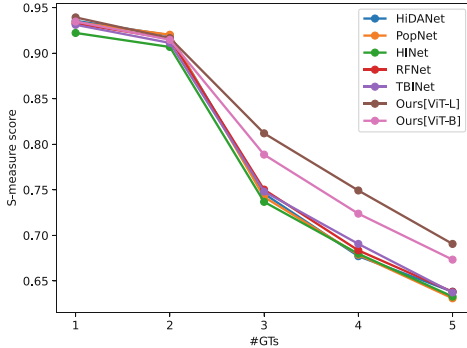


Fig. 3. S-measure by the number of GTs.

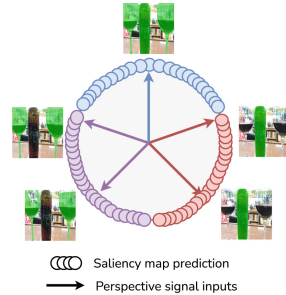


Fig. 4. An example where our model thinks there are only 3 different sets of salient objects but we input 5 multi-perspective signals.

4.2 Evaluation Metrics

We use four metrics to evaluate the effectiveness of different methods quantitatively. **MAE** [32] (M) shows the average absolute pixel inaccuracy. **S-measure** [10] (S_m) concentrates on object-aware and region-aware structural similarities between the predicted saliency map and the ground truth where α is set to 0.5 to assign equal constraints to both. **F-measure** [1] (F_β) is a region-based similarity metric with recall and precision as its foundations where β is set to 0.3 to emphasize precision more than recall as suggested in [4]. **E-measure** [11] (E_m) is defined as local pixel matching in addition to image-pixel statistics.

4.3 Implementation Detail

Our strategy consists of two sequential phases: pretraining with masked prediction task and finetuning with RGB-D saliency prediction task. Both phases use a similar image resolution 224×224 with a patch size of 16×16 pixels.

Pretraining Phase. The models were trained with masked image prediction from 1.5M pairs of color and depth images on 10 RGB-D SOD datasets, NYU-depth-v2 [8] dataset, and 1.28M images from ImageNet1K [36] dataset (with pseudo-depth images generated from DPT-Hybrid [35]). The AdamW [26] optimizer was used with base learning rate 10^{-5} and weight decay 0.05. The batch size was set to 512 when the models were trained for 100 epochs using 4 V100 GPUs with automatic mixed precision enabled. To reduce training costs, the ViT encoder was initialized by pre-trained weights from MAE [16]. The pretraining stage do not include any perspective token.

Finetuning on Multi-ground Truth RGB-D SOD Datasets. Pretrained weights acquired during the pretraining phase are used as the initialization for the models for the RGB-D SOD task. Since the designs of the decoder in the pre-trained and finetuning stages are different, only the pre-trained weights in the encoder are retained, and the decoder must be trained from scratch. The learning rate in the encoder is kept low (at 10^{-7}), while the learning rate in the decoder is set higher than 100 times (at 10^{-5}), preserving the efficiency of the multimodal fusion. We apply the linear learning rate decay, gradually decreasing the learning rate until 10^{-8} . Like the pretraining stage, the AdamW optimizer [26] is used with weight decay 0.05. We implement conventional SOTA methods as follows: each method consists of five weights-separated models (i.e., the same structures but differences in the weights), in which each model learns a distinct set of salient objects.

Evaluation Phase. Every SOTA method is expected to yield five saliency maps in each scene for evaluation. For each ground truth, we select the best one out of five based on mean-absolute-error between saliency maps and that ground truth. Afterward, with the best saliency map found, we compute the remaining scores of S_m , F_β , and E_m (Tables 2 and 4). In the case of the conventional SOTA method, five separate models are all used to generate five saliency maps in each scene. Our proposed method only needs one model (instead of five) since our model can produce five saliency maps simultaneously by passing 5 multi-perspective signals to the model. Although during the training process, our models can implicitly learn the underlying number of GTs per scene (Sect. 3.1), it would be costly to find the maximum number of distinct saliency maps per scene. Instead, Fig. 3 shows that if the model thinks there are only 3 possible sets of salient objects, passing 5 multi-perspective signals to the model would result in only 3 distinct sets. The second (at $\frac{2\pi}{5}$) and the third (at $\frac{4\pi}{5}$) predictions having the same set of salient objects (i.e., a bottle of wine in the middle), whereas the fourth (at $\frac{6\pi}{5}$) and the fifth (at $\frac{8\pi}{5}$) predictions have the same saliency map (i.e., two glasses). Finally, the first prediction (at 0) has a different saliency map (i.e., both two glasses and a bottle of wine).

4.4 Comparison with SOTA RGB-D Models

Quantitative Comparison PySODMetrics [28] calculates the quantitative results.

Under the multi-ground truth setting, Table 2 shows the potential of our approach that only one ViT-L model consisting of 328M parameters outperforms the combination of five distinct models of AFNet [6] (added up to 1.27B parameters in total). Specifically, our large model gains 15.625% M , 0.65% S_m , 0.22% F_β , and 0.21% E_m over the second-best method AFNet.

Under the single-ground truth setting, Table 3 shows that our base model (with the ViT-B backbone) achieves competitive results with the second-best method AFNet [6]. Our large model (ViT-L) outperforms the others in all four

Table 2. Top-5 Multi-ground truth benchmark on COME15K datasets. **RED**, **GREEN**, and **BLUE** are used to highlight the top three results.

METHOD		HiDANet ₂₃ [41]	PopNet ₂₃ [43]	HINet ₂₂ [3]	AFNet ₂₃ [6]	RFNet ₂₃ [42]	TBINet ₂₃ [40]	Ours* [ViT-B] [ViT-L]	
Num. distinct models		5	5	5	5	5	5	1	1
Num. params		224M × 5	131M × 5	98.9M × 5	254M × 5	90.8M × 5	1.7M × 5	108M	328M
Input size		354 × 354	384 × 384	354 × 354	354 × 354	354 × 354	354 × 354	224 × 224	224 × 224
COME-E (4600)	$M \downarrow$	0.020	0.022	0.030	0.023	0.021	0.022	0.026	0.019
	$S_m \uparrow$	0.935	0.933	0.924	0.935	0.933	0.930	0.928	0.935
	$F_\beta \uparrow$	0.935	0.929	0.903	0.925	0.926	0.926	0.901	0.927
	$E_m \uparrow$	0.963	0.961	0.939	0.962	0.960	0.961	0.943	0.960
COME-H (3000)	$M \downarrow$	0.040	0.041	0.054	0.037	0.041	0.041	0.040	0.030
	$S_m \uparrow$	0.902	0.902	0.886	0.910	0.901	0.898	0.908	0.918
	$F_\beta \uparrow$	0.904	0.901	0.871	0.904	0.895	0.897	0.882	0.914
	$E_m \uparrow$	0.935	0.932	0.905	0.941	0.931	0.932	0.922	0.945
DUT-RGBD (400)	$M \downarrow$	0.021	0.027	0.039	0.020	0.030	0.022	0.037	0.016
	$S_m \uparrow$	0.945	0.937	0.926	0.952	0.931	0.944	0.939	0.958
	$F_\beta \uparrow$	0.945	0.928	0.906	0.944	0.916	0.940	0.895	0.951
	$E_m \uparrow$	0.968	0.958	0.934	0.972	0.953	0.968	0.932	0.975
ReDWeb-S (1000)	$M \downarrow$	0.086	0.096	0.104	0.079	0.087	0.090	0.073	0.068
	$S_m \uparrow$	0.797	0.782	0.782	0.828	0.799	0.797	0.841	0.840
	$F_\beta \uparrow$	0.803	0.779	0.759	0.828	0.794	0.802	0.808	0.829
	$E_m \uparrow$	0.837	0.808	0.795	0.867	0.829	0.838	0.853	0.863
Average (11964)	$M \downarrow$	0.032	0.034	0.047	0.032	0.034	0.035	0.034	0.027
	$S_m \uparrow$	0.914	0.912	0.896	0.918	0.912	0.908	0.918	0.924
	$F_\beta \uparrow$	0.913	0.907	0.873	0.908	0.903	0.903	0.889	0.915
	$E_m \uparrow$	0.944	0.940	0.911	0.947	0.940	0.941	0.931	0.949

Table 3. Single-ground truth benchmark on COME15K datasets. **RED**, **GREEN**, and **BLUE** are used to highlight the top three results.

	A2Dele ₂₀ [34]	JLDCF ₂₀ [15]	UCNet ₂₀ [45]	BBS-Net ₂₁ [13]	CMINet ₂₂ [46]	HINet ₂₃ [3]	AFNet ₂₃ [6]	RFNet ₂₃ [42]	TBINet ₂₃ [40]	HiDANet ₂₃ [41]	PopNet [43]	Ours [ViT-B] [ViT-L]	
$M \downarrow$	0.077	0.057	0.057	0.057	0.049	0.065	0.047	0.056	0.052	0.052	0.052	0.045	0.038
$S_m \uparrow$	0.802	0.869	0.868	0.874	0.885	0.859	0.890	0.871	0.876	0.874	0.876	0.897	0.906
$F_\beta \uparrow$	0.800	0.846	0.847	0.849	0.873	0.826	0.878	0.853	0.864	0.869	0.864	0.873	0.895
$E_m \uparrow$	0.847	0.893	0.900	0.893	0.912	0.878	0.921	0.899	0.910	0.908	0.906	0.916	0.933

Table 4. Multi-ground truth benchmark (diversity test). **RED**, **GREEN**, and **BLUE** are used to highlight the top three results.

METHOD		HiDANet ₂₃ [41]	PopNet ₂₃ [43]	HINet ₂₂ [3]	AFNet ₂₃ [6]	RFNet ₂₃ [42]	TBINet ₂₃ [40]	Ours* [ViT-B] [ViT-L]	
COME-E (7562 sets in 4600 scenes)	$M \downarrow$	0.068	0.068	0.073	0.068	0.066	0.062	0.049	0.039
	$S_m \uparrow$	0.828	0.825	0.820	0.830	0.829	0.829	0.848	0.862
	$F_\beta \uparrow$	0.768	0.760	0.740	0.763	0.763	0.767	0.757	0.798
	$E_m \uparrow$	0.861	0.859	0.842	0.861	0.860	0.867	0.860	0.887
COME-H (5555 sets in 3000 scenes)	$M \downarrow$	0.091	0.089	0.098	0.087	0.088	0.083	0.065	0.050
	$S_m \uparrow$	0.793	0.792	0.783	0.801	0.796	0.794	0.825	0.844
	$F_\beta \uparrow$	0.727	0.722	0.698	0.729	0.724	0.726	0.724	0.773
	$E_m \uparrow$	0.831	0.829	0.809	0.837	0.831	0.837	0.836	0.871

criteria. In particular, our large model gains 19.15% M , 1.80% S_m , 1.94% F_β , and 1.30% E_m over the second-best method AFNet.

Moreover, Fig. 3 shows that as the number of GTs per scene increases, the overall performance of all models drops, but the S-measure scores of our ViT-L and ViT-B models remain relatively higher than other SOTA methods’.

Qualitative Comparison. Figure 5 shows three complex scenes presenting two examples in which our models perform best and one case in which our models perform worst when compared to other SOTAs. Some of our predictions have S-measure scores slightly lower than other SOTAs’, but the average S-measure scores are significantly higher than other SOTAs’ since our models predict more diverse and correct sets of salient objects (e.g., 0.966 S_m of our second prediction is slightly lower than 0.976 S_m of RFNet and 0.967 S_m of HiDANet, but our average score is substantially greater). Diversity is crucial in the multi-ground truth settings.

4.5 Ablation Studies

Contributions of Pre-Trained Self-supervised Methods in Multimodal Fusion Process. We analyze in Table 5 about weights initialization from different pre-trained paradigms, including:

- (1) *MultiMAE*: a self-supervised learning paradigm and pre-trained with an image mask prediction task but with three modalities including additional semantic segmentation modality, about two half of the training data is pseudo-generated.
- (2) *S-MultiMAE*: a modified version of MultiMAE but with two modalities including color and depth.
- (3) *MAE*: a simplified version of MultiMAE and the same as MultiMAE but with only one modality.
- (4) *Supervised ViT*: a supervised learning paradigm with an image classification task, pre-trained on Image1K.
- (5) *Non-pretrained ViT*: trained from scratch with random initialization.

As can be seen in Table 5, with a 75% mask ratio, MAE and MultiMAE have improved the saliency prediction tasks over the supervised learning paradigms. Although the results indicate that MultiMAE performs marginally better than MAE, neither of the two pretrained methods has yet reached state-of-the-art due to the following likely causes: Firstly, because MAE training only employs one modality (i.e., RGB), it can only learn the feature representation of that modality and has a restriction on the dual-modality fusion (i.e., RGB-D), which is essential for RGB-D applications; Secondly, Although MultiMAE allows for multimodal fusions (i.e., RGB-D-Semantic Segmentation), the majority of the training data is made up of images that were both generated artificially and came from a single source (color images). As a result, the models do not adequately account for the complex semantic contexts between modalities, which lowers their ability to perform RGB-D tasks.

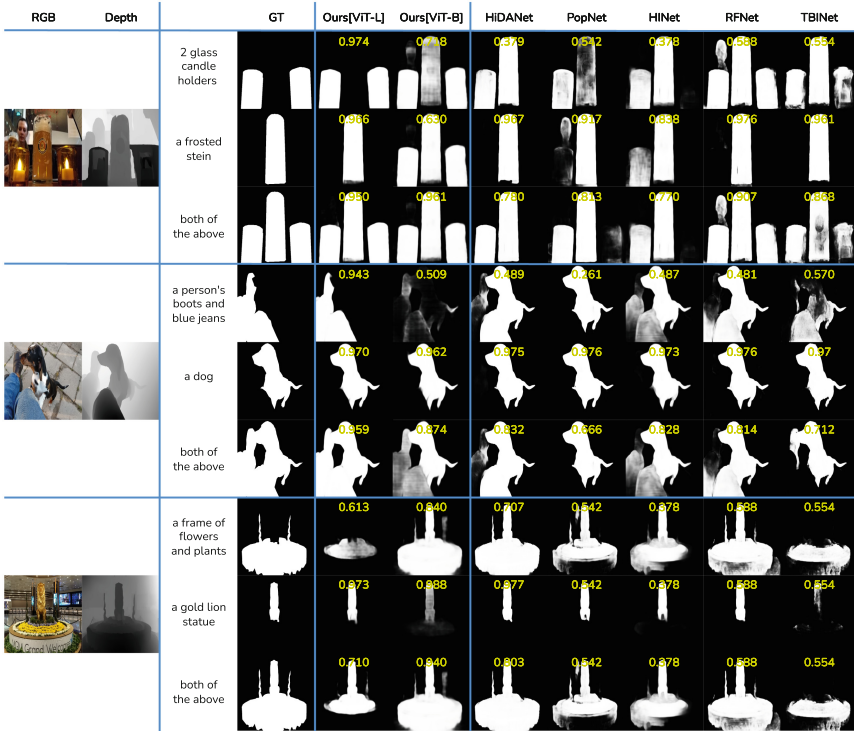


Fig. 5. Qualitative comparison between our S-MultiMAE and existing SOTA methods under the multi-ground truth setting includes our two best-case scenarios and one worst-case scenario with each having three different GTs. A number on top of each prediction indicates S_m score. The text is for interpretation only and not used as an input to the model.

Table 5. Ablation study: Weights initialization from different pre-trained paradigms.

		S-MultiMAE		MultiMAE	MAE	Supervised ViT	No pretrained
Modalities		RGB+D		RGB+D+Semseg	RGB	RGB	RGB
Mask ratio		83%	75%	75%	75%	-	-
COME-E	$M \downarrow$	0.020	0.021	0.027	0.031	0.087	0.090
	$S_m \uparrow$	0.936	0.936	0.933	0.921	0.828	0.820
	$F_\beta \uparrow$	0.931	0.927	0.912	0.893	0.756	0.747
	$E_m \uparrow$	0.964	0.960	0.946	0.935	0.827	0.818
COME-H	$M \downarrow$	0.031	0.032	0.044	0.048	0.116	0.120
	$S_m \uparrow$	0.917	0.918	0.906	0.897	0.791	0.783
	$F_\beta \uparrow$	0.915	0.911	0.885	0.873	0.729	0.721
	$E_m \uparrow$	0.947	0.944	0.918	0.913	0.792	0.783

5 Conclusion

In this research, we offer a straightforward and effective S-MultiMAE (a modified version of MultiMAE) for RGB-D saliency detection that adopts the multi-perspective signals to urge the models to think differently in complicated scenarios (where several ground truth annotations are available). Numerous tests using challenging Multi-ground truth RGB-D SOD benchmarks show that our S-MultiMAE not only enhances saliency detections but also overcomes the limitation of inconsistent ground truth annotations in complex scenes, which occurred with the traditional technique of most SOTA models.

Acknowledgments. We acknowledge Ho Chi Minh City University of Technology (HCMUT), VNU-HCM, for supporting this study.

References

1. Achanta, R., Hemami, S., Estrada, F., Susstrunk, S.: Frequency-tuned salient region detection. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1597–1604. IEEE (2009)
2. Bachmann, R., Mizrahi, D., Atanov, A., Zamir, A.: MultiMAE: multi-modal multi-task masked autoencoders. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) ECCV 2022. LNCS, vol. 13697, pp. 348–367. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-19836-6_20
3. Bi, H., Wu, R., Liu, Z., Zhu, H., Zhang, C., Xiang, T.Z.: Cross-modal hierarchical interaction network for RGB-D salient object detection. *Pattern Recognit.* **136**, 109194 (2022). <https://api.semanticscholar.org/CorpusID:253822254>
4. Borji, A., Cheng, M.M., Jiang, H., Li, J.: Salient object detection: a benchmark. *IEEE Trans. Image Process.* **24**(12), 5706–5722 (2015)
5. Caron, M., Bojanowski, P., Joulin, A., Douze, M.: Deep clustering for unsupervised learning of visual features. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) *Computer Vision – ECCV 2018*. LNCS, vol. 11218, pp. 139–156. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01264-9_9
6. Chen, T., Xiao, J., Hu, X., Zhang, G., Wang, S.: Adaptive fusion network for RGB-D salient object detection. *Neurocomputing* **522**, 152–164 (2023)
7. Cheng, Y., Fu, H., Wei, X., Xiao, J., Cao, X.: Depth enhanced saliency detection method. In: *Proceedings of International Conference on Internet Multimedia Computing and Service*, pp. 23–27 (2014)
8. Couprie, C., Farabet, C., Najman, L., LeCun, Y.: Indoor semantic segmentation using depth information. *CoRR abs/1301.3572* (2013). <https://api.semanticscholar.org/CorpusID:6681692>
9. Dosovitskiy, A., et al.: An image is worth 16x16 words: transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929* (2020)
10. Fan, D.P., Cheng, M.M., Liu, Y., Li, T., Borji, A.: Structure-measure: a new way to evaluate foreground maps. In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4548–4557 (2017)
11. Fan, D.P., Gong, C., Cao, Y., Ren, B., Cheng, M.M., Borji, A.: Enhanced-alignment measure for binary foreground map evaluation. In: *International Joint Conference on Artificial Intelligence* (2018). <https://api.semanticscholar.org/CorpusID:44072899>

12. Fan, D.P., Lin, Z., Zhang, Z., Zhu, M., Cheng, M.M.: Rethinking RGB-D salient object detection: models, data sets, and large-scale benchmarks. *IEEE Trans. Neural Netw. Learn. Syst.* **32**(5), 2075–2089 (2020)
13. Fan, D.-P., Zhai, Y., Borji, A., Yang, J., Shao, L.: BBS-net: RGB-D salient object detection with a bifurcated backbone strategy network. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) *ECCV 2020*. LNCS, vol. 12357, pp. 275–292. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58610-2_17
14. Fang, X., Zhu, J., Shao, X., Wang, H.: Grouptransnet: group transformer network for RGB-D salient object detection. *arXiv preprint arXiv:2203.10785* (2022)
15. Fu, K., Fan, D.P., Ji, G.P., Zhao, Q.: JL-DCF: joint learning and densely-cooperative fusion framework for RGB-D salient object detection, pp. 3052–3062 (2020)
16. He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick, R.: Masked autoencoders are scalable vision learners. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16000–16009 (2022)
17. 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)
18. Ju, R., Ge, L., Geng, W., Ren, T., Wu, G.: Depth saliency based on anisotropic center-surround difference. In: *2014 IEEE International Conference on Image Processing (ICIP)*, pp. 1115–1119. IEEE (2014)
19. Larsson, G., Maire, M., Shakhnarovich, G.: Colorization as a proxy task for visual understanding. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6874–6883 (2017)
20. Li, N., Ye, J., Ji, Y., Ling, H., Yu, J.: Saliency detection on light field. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2806–2813 (2014)
21. Li, T., Chang, H., Mishra, S., Zhang, H., Katabi, D., Krishnan, D.: MAGE: masked generative encoder to unify representation learning and image synthesis. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2142–2152 (2023)
22. Liu, N., Zhang, N., Han, J.: Learning selective self-mutual attention for RGB-D saliency detection, pp. 13756–13765 (2020)
23. Liu, Z., Tan, Y., He, Q., Xiao, Y.: SwinNet: swin transformer drives edge-aware RGB-D and RGB-T salient object detection. *IEEE Trans. Circuits Syst. Video Technol.* **32**(7), 4486–4497 (2021)
24. Liu, Z., Wang, Y., Tu, Z., Xiao, Y., Tang, B.: TriTransNet: RGB-D salient object detection with a triplet transformer embedding network. In: *Proceedings of the 29th ACM International Conference on Multimedia*, pp. 4481–4490 (2021)
25. Liu, Z., Mao, H., Wu, C.Y., Feichtenhofer, C., Darrell, T., Xie, S.: A convnet for the 2020s. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11976–11986 (2022)
26. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: *International Conference on Learning Representations* (2017). <https://api.semanticscholar.org/CorpusID:53592270>
27. Niu, Y., Geng, Y., Li, X., Liu, F.: Leveraging stereopsis for saliency analysis. In: *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 454–461. IEEE (2012)
28. Pang, D.J.: PySODmetrics: a simple and efficient implementation of SOD metrics. <https://github.com/lartpang/PySODMetrics>. Accessed 23 Oct 2023

29. Pang, Y., Zhao, X., Zhang, L., Lu, H.: CAVER: cross-modal view-mixed transformer for bi-modal salient object detection. *IEEE Trans. Image Process.* **32**, 892–904 (2023)
30. Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., Efros, A.A.: Context encoders: feature learning by inpainting. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2536–2544 (2016)
31. Peng, H., Li, B., Xiong, W., Hu, W., Ji, R.: RGBD salient object detection: a benchmark and algorithms. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) *ECCV 2014*. LNCS, vol. 8691, pp. 92–109. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10578-9_7
32. Perazzi, F., Krähenbühl, P., Pritch, Y., Hornung, A.: Saliency filters: contrast based filtering for salient region detection. In: *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 733–740. IEEE (2012)
33. Piao, Y., Ji, W., Li, J., Zhang, M., Lu, H.: Depth-induced multi-scale recurrent attention network for saliency detection. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7254–7263 (2019)
34. Piao, Y., Rong, Z., Zhang, M., Ren, W., Lu, H.: A2dele: adaptive and attentive depth distiller for efficient RGB-D salient object detection, pp. 9060–9069 (2020)
35. Ranftl, R., Bochkovskiy, A., Koltun, V.: Vision transformers for dense prediction. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 12179–12188 (2021)
36. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. *Int. J. Comput. Vision* **115**(3), 211–252 (2015)
37. Tong, Z., Song, Y., Wang, J., Wang, L.: Videomae: masked autoencoders are data-efficient learners for self-supervised video pre-training. *Adv. Neural. Inf. Process. Syst.* **35**, 10078–10093 (2022)
38. Vaswani, A., et al.: Attention is all you need. *Adv. Neural Inf. Process. Syst.* **30**, 6000–6010 (2017)
39. Vincent, P., Larochelle, H., Bengio, Y., Manzagol, P.A.: Extracting and composing robust features with denoising autoencoders. In: *Proceedings of the 25th International Conference on Machine Learning*, pp. 1096–1103 (2008)
40. Wang, Y., Zhang, Y.: Three-stage bidirectional interaction network for efficient RGB-D salient object detection. In: Wang, L., Gall, J., Chin, T.J., Sato, I., Chellappa, R. (eds.) *ACCV 2022*. LNCS, vol. 13845, pp. 215–233. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-26348-4_13
41. Wu, Z., Allibert, G., Meriaudeau, F., Ma, C., Demonceaux, C.: HiDAnet: RGB-D salient object detection via hierarchical depth awareness. *IEEE Trans. Image Process.* **32**, 2160–2173 (2023)
42. Wu, Z., Gobichettipalayam, S., Tamadazte, B., Allibert, G., Paudel, D.P., Demonceaux, C.: Robust RGB-D fusion for saliency detection. In: *2022 International Conference on 3D Vision (3DV)*, pp. 403–413. IEEE (2022)
43. Wu, Z., et al.: Source-free depth for object pop-out. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1032–1042 (2023)
44. Wu, Z., et al.: Object segmentation by mining cross-modal semantics. In: *Proceedings of the 31st ACM International Conference on Multimedia* (2023). <https://api.semanticscholar.org/CorpusID:258762372>
45. Zhang, J., et al.: UC-Net: uncertainty inspired RGB-D saliency detection via conditional variational autoencoders, pp. 8582–8591 (2020)
46. Zhang, J., et al.: RGB-D saliency detection via cascaded mutual information minimization (2021)



MSSF: A Multi-scale Siamese Flow Architecture for Multi-texture Class Anomaly Detection

Yibo Chen¹, Zhiyuan Hu², Le Huang³, and Jianming Zhang^{1,2}(✉)

¹ College of Control Science and Engineering, Zhejiang University, Hangzhou 310027, China

ncsl@zju.edu.cn

² Zhejiang Polytechnic Institute, Polytechnic Institute, Zhejiang University, Hangzhou 310027, China

³ College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou 310027, China

Abstract. Multi-class anomaly detection has been a promising research area. However, most methods focus on increasing backbone parameters or the depth of the network. This study uses multi-texture anomaly detection as an example to validate a lightweight flow-based pipeline called Multi-Scale Siamese Flow (MSSF) with a Multi-level Feature Fusion (MLFF) to fully use extracted shallow and deep features. Besides, a Mixed anomalies synthesis (MAS) method is incorporated into the MSSF and trains our pipeline in a self-supervised manner by designing a novel training loss combining negative log-likelihood with a changeable self-supervised hindering loss. Extensive experiments on real-world texture subsets or texture datasets, including MVTec-AD, KSDD2, MT, and AITEX, indicate the effectiveness of our MSSF. The inference speed surpasses the second fastest method, UniAD, about 2 times. Compared with other cutting-edge methods, the MSSF achieves an effective balance between performance and speed.

Keywords: Anomaly Detection · Normalizing Flow · Multiple Texture Classes · Inference Speed · Backbone Parameters

1 Introduction

Research on industrial anomaly detection has thrived [15, 17, 21] recently. Additionally, with the development of various networks, the multi-class anomaly detection task has become a popular branch of the visual anomaly detection field. The main challenge of the topic is the diverse characteristics of various

J. Zhang—This work was supported by Robotics Institute of Zhejiang University under Grant K12201.

Y. Chen and Z. Hu—Equal first authorship.

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

A. Antonacopoulos et al. (Eds.): ICPR 2024, LNCS 15317, pp. 30–44, 2025.

https://doi.org/10.1007/978-3-031-78447-7_3

industrial product styles. Most methods [19, 21, 27] focus on increasing the network parameters to improve perceptual power. To the best of our knowledge, only a few of them [7, 9] focus on analyzing computation and storage costs. Besides, due to the inner limited perception of distribution-based networks [4, 20], none of them is applied to multi-class anomaly detection and localization settings. In this paper, we aim to investigate the performance of a distribution-based structure on anomaly detection of regular industrial multi-texture class products.

As shown in Fig. 1, the main difference between the single-texture and multi-texture settings is the set number of network parameters. This indicates the utilization of a unified set of network parameters across different texture product types, as illustrated in Fig. 1b. Even though the texture classes seem regular and easy to distinguish, the network may easily tilt towards a few-class local optimum instead of a balance between various product types.

The multi-class mainstream detection ViT-based large models [5, 6, 19, 21, 22, 27] have been proven to outperform these methods on various datasets [2, 3, 10] competitively. However, most methods are mainly based on the extracted features or improve the performance by adding trainable layers. Besides, little research has looked into the lightweight flow-based method for anomaly detection.

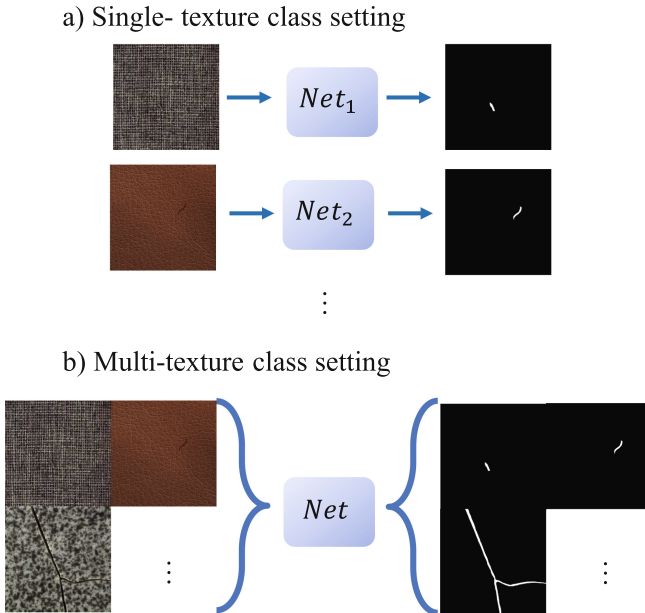


Fig. 1. Display of acceleration methods and the single-class task as well as multi-class task

Inference speed and efficacy are necessary indexes for the application of anomaly detection application. To improve the efficiency of training and testing,

a common approach is to directly use model frameworks with fewer parameters or distill large models. In the case of model distillation, it is necessary to prepare well-pretrained large models and feature extraction modules, which requires extra data preparation. However, using models with fewer parameters poses a significant challenge to the perceptual capabilities and may not achieve the desired training results. This study explores the potential application of small models based on normalizing flow in detecting defects in multiple categories.

The aforementioned multi-class anomaly detection methods have their limitations. Some pipelines have restricted parameters but require performance enhancement, while others deliver excellent results but involve an extensive number of parameters, which hinders real-time application.

To explore the potential of parameter limitation with competitive detection ability for practical application, a Multi-scale Siamese Flow (MSSF) architecture is introduced with a Multi-level Feature Fusion (MLFF) block to combine multi-level information and optimize inference speed in a rarely precedent self-supervised flow manner. Besides, a specified anomaly detection loss for a Mixed anomalies synthesis (MAS) process is proposed to take advantage of multi-source self-supervised anomaly synthesis and to benefit overall and detailed normal feature learning. Thus, by assessing the distribution of normal samples and multi-style artificial anomalies, the training processes optimize parameters of distribution estimating pipeline and the fusion modules MLFF. To validate the efficiency of our methods, we compare its performance with mainstream methods on various texture datasets [2, 3, 10, 16].

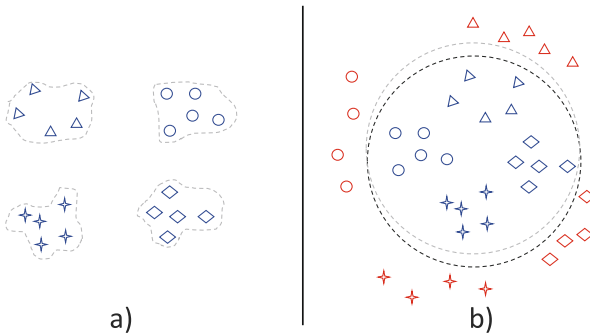


Fig. 2. The demonstration of the improvement and novelty of flow-based architecture

In Fig. 2, each enclosed dashed line delineates the decision boundary between normal and anomalous samples as trained by individual flow models. The samples residing within these boundaries are the mapped normal-sample patterns, and vice versa. The different shape representations correspond to various types of test samples.

The characteristics of traditional flow models are depicted in Fig. 2 a). They estimate the distribution exclusively using normal samples, with different cate-

gories estimated separately using a set of parameters. Figure 2 b) showcases the distinct novelties and technical strategies of the MSSF. Multiple categories are trained simultaneously to enhance inference efficiency and speed. Even under the constraints of finite samples and parameter volumes, area boundaries can be defined with the assistance of self-supervision, which further boosts the precision of the defect detection process. Moreover, even when normal samples are limited, the network can still predict a robust and well-generalized boundary employing the MAS and Siamese flow architecture.

The main contributions of our method are as follows:

- (1) To explore the potential of parameter limitation with competitive detection ability for practical application, we introduce a Multi-Scale Siamese Flow (MSSF) architecture that utilizes Siamese flows to enhance robustness while preserving limited parameters and ensuring rapid inference speed. To further exploit the usage of cross-level information and improve performance, the Multi-Level Feature Fusion (MLFF) facilitates the integration of multi-level features.
- (2) To train the MSSF in a self-supervised manner, the Mixed Anomalies Synthesis (MAS) process is introduced to produce realistic anomalies for global normal template and extra-source defects for detailed usual pattern. Thus, by assessing the distribution of normal samples and multi-style artificial anomalies, the training processes optimize parameters of distribution estimating pipeline and the fusion modules MLFF. A dynamic novel loss function is designed for the self-supervised learning context which can change the ratio of different parts gradually during training.
- (3) We compare the MSSF with other comparative pipelines on various texture categories of different datasets. The index results indicate the potential of multi-texture class anomaly detection with flow-based models.

2 Related Work

2.1 Distribution-Based Anomaly Detection Pipelines

Distribution-based models can be divided into distribution-estimation pipelines [4,20], diffusion-based models [11,26], and Normalizing-flow-based types. Distribution-assessment methods are mainly focused on directly assessing the characteristic distribution of target samples. Diffusion-based model pipelines aim to add noise to corrupt target images and reconstruct them for non-defect samples.

Normalizing-flow-based frameworks [7,9,23,28] aim to find reversible flows to project images into individual normal distributions. MSFlow [28] combines cross-scale features to estimate the proper distribution of normal samples precisely. The training and the testing speed of the above methods still have space to improve compared with Fastflow [23]. However, they all lack generalization capacity.

2.2 Multi-class Anomaly Detection Methods

Multi-class anomaly detection is a promising research topic. Zhao et al. [27] adopt a unified CNN-based network to locate anomaly areas. PMAD [21] and UniAD [22] apply an attention mechanism to increase the perception under multi-class situations.

Various studies [1, 17, 18] focus on texture anomaly detection. MCDEN [19] and CKT [6] study the multi-texture class anomaly detection task. The study of MSSF helps bridge the gap in pipeline efficiency research.

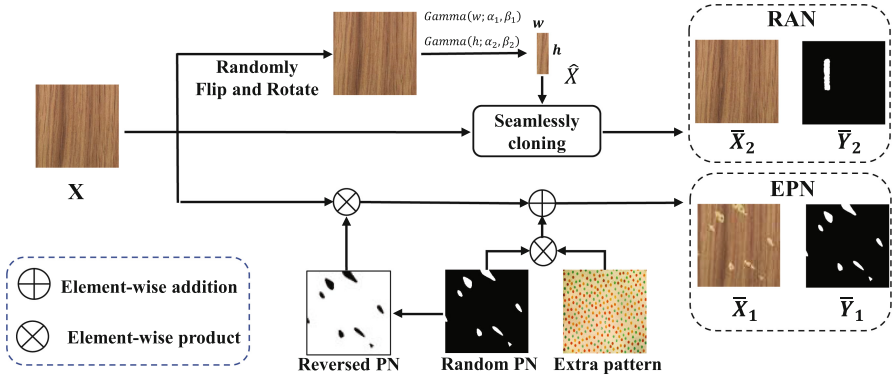


Fig. 3. The figure of the Mixed Anomalies Synthesis (MAS)

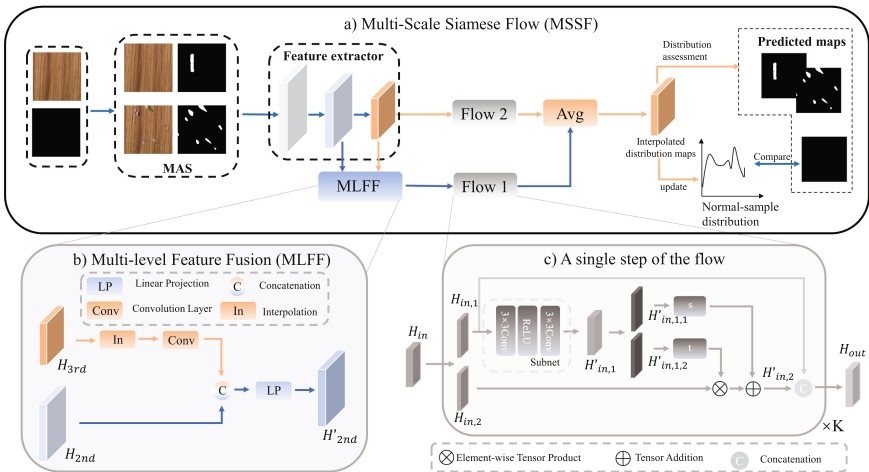


Fig. 4. The pipeline and inner structure of Multi-Scale Siamese Flow (MSSF)

3 Method

This section introduces a novel flow-based pipeline, Multi-Scale Siamese Flow (MSSF), as shown in Fig. 4. To start with, as shown in Fig. 3, the Mixed Anomalies Synthesis (MAS) is applied to inject obvious Extra-source Perlin-Noise (EPN) anomalies from extra-source patterns like the Draem [25] and Realistic Anomaly Noise (RAN) seamlessly from inner-class patches [12]. Subsequently, the input of the MSSF is non-defect images and synthetic anomaly images, which novel flow-based anomaly detection loss are used to update the projected Gaussian distribution and push away abnormal projections individually. With the help of MLFF, the MSSF can make full use of multi-level features. The details of the proposed pipelines are demonstrated as follows.

3.1 The Mixed Anomalies Synthesis (MAS)

As shown in Fig. 3, according to the given normal training set $X = \{x_i \mid i \in 1, \dots, n\}$, the MSSF periodically synthesizes the EPN \bar{X}_1 and the RAN \bar{X}_2 by randomly selecting k samples from X .

During the creation of the EPN $\bar{X}_1 = \{\bar{x}_{1,i} \mid i \in 1, \dots, k\}$, a mask set containing rectangular shapes $\bar{Y}_1 = \{\bar{y}_{1,i} \mid i \in \{1, \dots, k\}\}$ is randomly generated in each epoch, where the width (w_i) and height (h_i) of each rectangle follow the Gamma distributions, i.e., $w_i \sim \Gamma(\alpha_w, \beta_w)$ and $h_i \sim \Gamma(\alpha_h, \beta_h)$ for $i = 1, \dots, k$. \bar{Y}_1 dot product the randomly rotated and flipped \bar{X} to output $\hat{X} = \{\hat{x}_{1,i} \mid i \in 1, \dots, k\}$. Finally, the X and the \hat{X} are seamlessly combined [12] to obtain $\bar{X}_1 = \{\bar{x}_{1,i} \mid i \in 1, \dots, k\}$.

During the fusion of the RAN \bar{X}_2 , a set of Normalized Perlin Noise maps $\bar{Y}_2 = \{\bar{y}_{2,i} \mid i \in 1, \dots, k\}$ is randomly generated like the Draem [25]. Nevertheless, to preserve the alien features, we directly remove original parts and paste frequent anomaly regions in a manner different from the Draem. Meanwhile, k extra-source texture images, $X_{ex} = \{x_{ex,i} \mid i \in 1, \dots, k\}$, are chosen randomly.

$$\begin{aligned} \bar{x}_{2,i} &= (1 - \bar{y}_{2,i}) \odot \bar{x}_{2,i} + \\ &\bar{y}_{2,i} \odot [T_i x_{ex,i} + (1 - T_i) x_i], \quad i = 1, \dots, k \end{aligned} \quad (1)$$

In formula (1), the T is a transparent parameter that balances the injected extra-source and original images. In conclusion, the input of the pipeline is X and \bar{X} , where \bar{X} contains $\{\bar{X}_1, \bar{X}_2\}$.

3.2 The Multi-Scale Siamese Flow (MSSF)

Initially, a pretrained feature extractor, denoted as f_{ea} , is deployed to condense the information. Let H denotes the features in the Flow1, while \bar{H} denotes the features in the Flow2. The notations without $\bar{\cdot}$ indicates the first flow, while ones with $\bar{\cdot}$ refers to the second flow. The MLFF to obtain the fused 2nd-layer and 3rd-layer features, namely H_{2nd} , \bar{H}_{2nd} and H_{3rd} , \bar{H}_{3rd} , as shown in formula (2),

where $f_{ea,12}$ and $f_{ea,123}$ indicates utilizing 2 and 3 layers of extraction blocks individually. $LP(\cdot)$, $C(\cdot)$, $Intp(\cdot)$, and $Conv(\cdot)$ indicate linear projection, concatenation, interpolation, and convolution layers individually. The more intuitive representation of this process is illustrated in Fig. 4(b).

$$\begin{cases} H_{3rd} = f_{ea,123}(X), \\ \bar{H}_{in} = f_{ea,123}(X); \\ H_{in} = LP(C(Conv(Intp(H_{3rd})), f_{ea,12}(X))), \end{cases} \quad (2)$$

As shown in Fig. 4(a), the Multi-Scale Siamese Flow (MSSF) is introduced to estimate the likelihood and obtain the reversible multi-step Siamese flow $f_{sia} = \{f^1 : H_{in} \rightarrow H_{out}; f^2 : \bar{H}_{in} \rightarrow \bar{H}_{out}\}$. Both the first flow f^1 and the second flow f^2 are each composed of K flow blocks. On the one hand, the first flow utilizes the multi-layer combination to improve understanding of detailed information. On the other hand, the second flow is aimed at focusing on high-level semantic information to concentrate on global representations. As a result, the innovative twin pipelines avoid the predilection to local details or simply global expressions.

As shown in Fig. 4(c), every single step of each flow has the same structure whose expression can be represented as follows:

$$\begin{cases} H_{in} = \{H_{in,1}, H_{in,2}\}, \\ H'_{in,1} = Subnet(H_{in,1}), \\ H'_{in,1} = \{H'_{in,1,1}, H'_{in,1,2}\}, \\ H'_{in,2} = s(H'_{in,1,1}) + H_{in,2} \odot t(H'_{in,1,2}), \\ H_{out} = C(H_{in,1}, H'_{in,2}) \end{cases} \quad (3)$$

If it is the 1st step, the input H_{in} is the concatenation of $\{H_{3rd}, H_{2nd}\}$ and \bar{H} is the concatenation of $\{\bar{H}_{3rd}, \bar{H}_{2nd}\}$. In formula (3), the *Subnet* indicates a subnet consisting of a 3×3 convolution block $Conv_{3 \times 3}$ and an activation function *ReLU*. As shown in Fig. 4(c), $\{H_{in,1}, H_{in,2}\}$ and $\{H'_{in,1,1}, H'_{in,1,2}\}$ are the split of H_{in} and $H'_{in,1}$ individually. $H_{in,1}$ corresponds to the first half of the features in H_{in} . $H_{in,2}$ corresponds to the remaining portion. $H'_{in,1,1}$ and $H'_{in,1,2}$ represent the similar definitions. $C(\cdot)$, $s(\cdot)$, and $t(\cdot)$ represent concatenation, scale, and exponential function, respectively.

3.3 Training Loss

MSSF aims to acquire the distribution of original normal images. The foundation of flow-based methods is to optimize the assessment of the distribution $p_H(h)$ of the original flawless image features $H = \{H_{2nd}, H_{3rd}\}$ to closely approximate non-defect maps as much as possible. So, the ideal result is to make the $p_H(h)$ close to zero. As a result, the loss of the Siamese-flow structure $f_{sia} = \{f^1, f^2\}$ can be measured by punitive negative log-likelihood $-\log(p_H(h))$ as follows:

$$L_{neg} = \sum_{i=1}^2 \left[\frac{\|z_i\|_2^2}{2} - \log \left| \frac{\partial f^i(h)}{\partial h} \right| \right] \quad (4)$$

Additionally, we suppose the mapped distribution is the standard Gaussian distribution. Thus, the L_{neg} has the property shown in formula (4).

The mapped results in formula (4) are $Z \sim \mathcal{N}(0, 1)$. The Jacobian matrix $\frac{\partial f(h)}{\partial h}$ is estimated by clamped $\tanh(s(H'_{in,1,1}))$, where \tanh is the hyperbolic tangent.

With the averaged results \hat{H} from H and \bar{H} , the anomaly map \hat{Y} are calculated as follows:

$$\hat{Y} = Intp(\exp(\hat{H})) \quad (5)$$

where $Intp(\cdot)$ and $exp(\cdot)$ represent interpolation and exponential functions individually.

The artificial anomalies seem redundant and hinder the proper inference of the distribution evaluation. Indeed, this part is designed as an obstacle for flow updates. The anomaly-localization pixel-wise average $L1$ loss L_{mask} is listed as follows:

$$\begin{cases} Y_{all} = \{Y_{norm}, \hat{Y}\} \\ L_{mask} = L_1(Y, Y_{all}) \end{cases} \quad (6)$$

where Y_{norm} and \hat{Y} are the zero maps of the input normal samples and synthetic defect maps separately. Y indicates the ground-truth maps. The formula (7) is the calculation of anomaly score S_{ano} . $\{\cdot\}$ indicates the concatenation operation.

$$S_{ano} = max(Y_{all} * f_{m \times m}) \quad (7)$$

where the $f_{m \times m}$ indicates the average pooling kernel size is $m \times m$ and $*$ means the convolution operator. Y_{all} is the composite set of Y_{norm} and \hat{Y} .

The eventual training loss L_{tr} , taking advantage of the negative log-likelihood and the segmentation loss in an adversarial way, is shown as formula (8).

$$L_{tr} = L_{neg} - \lambda \cdot epoch \cdot L_{mask} \quad (8)$$

where λ is a parameter that influences the constraint capability of L_{mask} . Larger λ indicates more intensive adversarial behavior. The $epoch$ represents the current training epoch. The $epoch$ is a dynamic factor that gradually balances the log-likelihood component with the pixel-level part.

4 Experiments

4.1 Datasets, Indexes and Experiment Setups

MVTec-AD [2] dataset contains five texture classes with over 200 normal images for training. **KSDD2** [3] dataset includes 356 defective images and 2979 normal images. **MT** (the magnetic tiles) [10] dataset contains 784 defective images and 1904 non-defective images. **AITEX** [16] dataset contains seven texture classes with 140 non-defect images and over 100 pixel-wise labeled defect images.

Table 1. Image-level/pixel-level AUROC(%) on texture classes of MVTec-AD dataset (Multi-class)

Category	Non-flow-based			Flow-based		
	MKD [14]	UniAD [22]	PMAD [21]	CSFlow [13]	CFlow [9]	MSSF
Carpet	69.8/95.5	99.5/97.4	99.9/98.8	100.0/98.1	99.3/99.5	99.3/98.3
Grid	83.8/82.3	98.4/94.8	98.2/96.2	99.5/97.3	98.5/99.3	98.6/97.1
Leather	93.6/96.7	100.0/97.3	100.0/99.0	100.0/98.2	100.0/99.2	99.2/99.3
Tile	89.5/85.3	99.3/98.7	100.0/95.6	100.0/96.7	97.6/99.1	99.6/98.5
Wood	93.4/80.5	98.6/91.8	98.5/90.8	98.6/92.5	98.2/96.7	99.3/99.2
Average	86.0/88.1	99.2/96.0	99.3/96.1	99.6 /96.6	98.9/ 98.7	99.2/98.3

Table 2. Pixel-level AUROC(%) on KolektorSDD2, MT and AITEX datasets

Methods	KSDD2	MT	AITEX	Average
MKD [14]	94.4	76.6	81.2	81.1
CKT [6]	94.7	78.9	81.3	85.2
CFlow [9]	96.2	94.7	89.1	93.3
MSSF	97.1	93.0	91.6	93.9

The Area Under the Receiver Operating Characteristic Curve (AUROC) is a reliable index to compare the image-level and pixel-level performance.

All the training stages are completed on a single GPU (GeForce RTX 2080 Ti). We resize all input images and masks to a consistent size of 256×256 pixels. The Wide-Resnet50 [24] is selected as the pretrained feature extractor. The defective parts originate from the Describable Textures Dataset (DTD) [8]. The flow step K is 8. Transparency T is 0.8. Both the (α_w, β_w) and (α_h, β_h) are set as $(2, 0.1)$. The numbers of normal samples n and noise-injected samples k are 50 and 100 individually, with 50 RAN abnormal samples and 50 EPN anomaly samples. The learning rate lr is initialized as $1e-3$ with the weight decay as $1e-5$, while the batch size is set to 128. The pipeline is trained for 500 epochs. The λ is set as 100.

**Fig. 5.** Visualization of segmentation results compared with other flow-based methods

4.2 Results on Multiple Datasets

As visualized in Fig. 5, our method achieves relatively good segmentation results on some representative samples. The image-level texture anomaly detection ability has been proven effective, especially for several outstanding types: carpet, leather, and tile. Compared to large CFlow [9] and CSFlow [13], the MSSF achieves more considerable segmentation certainty with less computational cost and space.

As shown in Table 1, the image-level texture anomaly detection ability has been proven effective, especially for types like tile (99.6%/98.5%) and wood (99.3%/99.2%). Although the image-level result of 99.2% falls behind CSFlow [13], our method still remains competitive compared with other methods like UniAD [22] and PMAD [21] in pixel-level results. The averaged image-level and pixel-level results represent a balance between inference efficiency and performance.

According to Table 2, the MSSF has achieved considerable averaged pixel-level AUROC results (93.9%) compared with CKT (85.2%) [6] and CFlow (93.3%) [9], especially on the KSDD2 [3] and AITEX [16] dataset. The validation results remain . Generally speaking, even executed on low-resolution and challenging real-life datasets, MSSF is still competitive in results and running speed with finite resources.

Table 3. Training/testing speed (images/sec) and backbone parameters (M) of the models

Methods	Training	Testing	Param
UniAD [22]	38.6	64.7	27.4
PMAD [21]	31.9	31.1	92.0
CSFlow [13]	26.9	19.4	275.2
CFlow [9]	27.4	38.2	81.6
MSSF	74.3	124.7	45.6

Table 4. Ablation results of image-level/pixel-level AUROC(%) on texture classes of MVTec-AD dataset

8-Steps	EPN	RAN	MLFF	Flow ₁	Flow ₂	results
				✓		80.2/76.3
					✓	89.3/84.5
				✓	✓	92.2/89.1
			✓	✓	✓	93.7/91.4
	✓		✓	✓	✓	96.9/94.1
		✓	✓	✓	✓	95.8/95.2
	✓	✓	✓	✓	✓	97.6/96.1
✓	✓	✓	✓	✓	✓	99.2/98.4

4.3 Computation Cost Analysis

According to Table 3, with respect to MSSF, the backbone parameters are outstanding and lead to relatively considerable results in contrast to other public codes^{1,2,3}. Besides, compared with the open-resource codes⁴ of the smallest and second fastest method, UniAD [22], the training and inference speeds of the MSSF are approximately 2 times faster. As analyzed in the previous discussion, the extraordinary efficiency of the MSSF remains competitive in various texture categories. Compared with other decoder-encoder structures, MSSF applies an end-to-end structure to acquire distribution and segmentation results directly. In summary, the MSSF sacrifices tolerable performance for the superior inference and parameter quantity.

Compared with other flows like CS-Flow [13] and CFlow [9], MSSF distinguishes itself as it doesn't require the interaction of multiple scales or deep structures during propagation, yet it efficiently handles smaller input images and features. The MSFF combines the multi-level features both initially and ultimately. Instead of enhancing performance with numerous parameters, blocks, twin structures and self-supervised manifold and distinct loss are designed to improve performance based on the shallower architecture. Consequently, the MSSF is more efficient.

4.4 Ablation Studies

The ablation processes are based on the five texture classes of MVTec-AD dataset.

As visualized in Table 4, the ablation studies reflect the rational choice of our Siamese pipeline. The initial ablation studies are conducted when the MSFF contains four steps. Simply remaining one flow channel limits the performance of the pipeline. The 3rd-layer features retain semantic-level information, and the 2nd-layer features keep low-level details. To be more specific, the first flow utilizes the multi-layer combination to improve understanding of detailed information, while the second flow is aimed at focusing on high-level semantic information to concentrate on global representations. To further balance the performance (99.2%/98.4%) and the speed, we chose the 8-step structure. The results of varying parameters are listed in Table 6.

In addition, the self-supervised anomaly detection mode is an irreplaceable approach. After mixing the EPN and RAN, the AUROC reaches 97.6%/96.1%, surpassing the results obtained by solely inputting the EPN (96.9%/94.1%) or the RAN (95.8%/95.2%). Indeed, the prompt of EPN facilitates distinguishing the general character of normal regions with the introduction of obvious differences, while RAN pushes the pipeline to focus on minor details of the authentic

¹ <https://github.com/gudovskiy/cflow-ad>.

² <https://github.com/xcyao00/PMAD>.

³ <https://github.com/marco-rudolph/cs-flow>.

⁴ <https://github.com/zhiyuanyou/UniAD>.

features by distinguishing the nearly seamless minor differences between normal and abnormal patterns.

During training, we found that the magnitude of L_{neg} in formula (8) can vary significantly. Our experiments showed that maintaining a fixed loss, whether large or small, often results in pixel-level AUROC scores below 90.0%, similar to scenarios involving only normal samples. This issue stems from the dynamic nature of L_{neg} . If the coefficient is fixed and large, then the L_{neg} part will lead to overfitting on generated abnormal patterns when the normal patterns have converged as the number of epochs increases. However, if the coefficient is fixed and small, L_{neg} has little impact, and the self-supervised architecture becomes useless. To address this, we incorporated *epoch* as an adaptive adjustment factor and λ as a scaling factor. The effects of λ are detailed in Table 5.

Table 5. Ablation results of λ based on image-level/pixel-level AUROC(%)

λ	1	10	100	1000
AUROC	93.5/92.4	97.1/96.8	99.2/98.4	96.2/95.9

If the factor of normalized L_{mask} becomes too large, such as a thousand, the influence of the normal distribution mapping loss, L_{neg} , might become negligible, which could impair the mapping capabilities of the flow-based MSFF. Conversely, a small factor might make the self-supervised component ineffective. Based on our experiments, we select 100 as the optimal choice, making it the value adopted in our other ablation studies.

Table 6. Ablation results of the backbone parameters(M) based on image-level/pixel-level AUROC(%) and the Training/testing speed (images/sec)

Steps	4	8	16
Param	22.1	45.6	86.3
AUROC	97.6/96.1	99.2/98.4	99.4/98.9
Training	316.5	124.7	34.4
Testing	142.1	74.3	42.5

The number of steps indicates the number of parameters. As shown in Table 6, with increasing in steps, MSSF achieves better anomaly detection results on the grid and across categories. Thus, compared to CFlow (98.9%/98.7%) and CSFlow (99.6%/96.6%), MSSF has more competitive anomaly detection ability (99.4%/98.9%). However, the training and testing speeds are not competitive. If the flow steps are limited to four, the training and testing speeds increase dramatically. The parameter amount is optimal, even when compared with UniAD

[22]. Additionally, the 4-step changes in pixel values and AUROC are also tolerable for classes like Leather (99.6%/98.5%), Carpet (99.2%/98.1%) and Tile (99.5%/95.4%) when the 8-step MSFF achieves Leather (99.2%/99.3%), Carpet (98.6%/97.1%) and Tile (99.6%/98.5%).

From the ablation experiments, MSSF demonstrates a significant balance between metrics and efficiency when parameters are limited. The results for 8-step configurations in our study indicate that MSSF significantly improves inference speed and efficiency with limited loss in metric performance.

5 Discussion and Conclusion

With balanced excellent inference and inference efficiency, a novel self-supervised flow-based pipeline is proposed to detect multi-texture defects. Besides, the proposed MSSF with the MLFF combines the self-supervised anomaly synthesis MAS, assisting efficient assessment of the distribution of normal samples and multi-style artificial anomalies. The performance on various datasets reflects the superiority in training and inference speed of MSFF while sacrificing little performance indices on specific categories like *Grid*. However, the training results are uncertain sometimes, probably because of the extremely limited backbone parameters for thorough perception of multi-texture categories.

References

1. Aota, T., Tong, L.T.T., Okatani, T.: Zero-shot versus many-shot: unsupervised texture anomaly detection. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 5564–5572 (2023)
2. Bergmann, P., Fauser, M., Sattlegger, D., Steger, C.: Mvtec ad—a comprehensive real-world dataset for unsupervised anomaly detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9592–9600 (2019)
3. Božič, J., Tabernik, D., Skočaj, D.: Mixed supervision for surface-defect detection: from weakly to fully supervised learning. *Comput. Ind.* **129**, 103459 (2021)
4. Cao, Y., Xu, X., Liu, Z., Shen, W.: Collaborative discrepancy optimization for reliable image anomaly localization. *IEEE Trans. Ind. Inf.* (2023)
5. Chen, Y., Peng, H., Huang, L., Zhang, J., Jiang, W.: A novel mae-based self-supervised anomaly detection and localization method. *IEEE Access* **11**, 127526–127538 (2023). <https://doi.org/10.1109/ACCESS.2023.3332475>
6. Chen, Z., Yao, X., Liu, Z., Zhang, B., Zhang, C.: Ckt: cross-image knowledge transfer for texture anomaly detection. In: 2023 IEEE International Conference on Image Processing (ICIP), pp. 266–270. IEEE (2023)
7. Chiu, L.L., Lai, S.H.: Self-supervised normalizing flows for image anomaly detection and localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2926–2935 (2023)
8. Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., Vedaldi, A.: Describing textures in the wild. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3606–3613 (2014)

9. Gudovskiy, D., Ishizaka, S., Kozuka, K.: Cflow-ad: real-time unsupervised anomaly detection with localization via conditional normalizing flows. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 98–107 (2022)
10. Huang, Y., Qiu, C., Yuan, K.: Surface defect saliency of magnetic tile. *Vis. Comput.* **36**, 85–96 (2020)
11. Mousakhan, A., Brox, T., Tayyub, J.: Anomaly detection with conditioned denoising diffusion models. arXiv preprint [arXiv:2305.15956](https://arxiv.org/abs/2305.15956) (2023)
12. Pérez, P., Gangnet, M., Blake, A.: Poisson image editing. *ACM Trans. Graph.* **22**(3), 313–318 (2003)
13. Rudolph, M., Wehrbein, T., Rosenhahn, B., Wandt, B.: Fully convolutional cross-scale-flows for image-based defect detection. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 1088–1097 (2022)
14. Salehi, M., Sadjadi, N., Baselizadeh, S., Rohban, M.H., Rabiee, H.R.: Multiresolution knowledge distillation for anomaly detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 14902–14912 (2021)
15. Shin, W., Lee, J., Lee, T., Lee, S., Yun, J.P.: Anomaly detection using score-based perturbation resilience. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 23372–23382 (2023)
16. Silvestre-Blanes, J., Albero-Albero, T., Miralles, I., Pérez-Llorens, R., Moreno, J.: A public fabric database for defect detection methods and results. *Autex Res. J.* **19**(4), 363–374 (2019)
17. Tao, X., Adak, C., Chun, P.J., Yan, S., Liu, H.: Vitalnet: anomaly on industrial textured surfaces with hybrid transformer. *IEEE Trans. Instrum. Meas.* **72**, 1–13 (2023)
18. Tao, X., Yan, S., Gong, X., Adak, C.: Learning multi-resolution features for unsupervised anomaly localization on industrial textured surfaces. *IEEE Trans. Artif. Intell.* (2022)
19. Yang, H., Zhu, H., Li, J., Chen, J., Yin, Z.: Multi-category decomposition editing network for the accurate visual inspection of texture defects. *IEEE Trans. Autom. Sci. Eng.* (2023)
20. Yang, Y., Mao, J., Wang, Y., Zhang, H., Zhou, X., Chen, Y.: Patch variational autoencoder-based industrial defect detection. In: 2022 13th Asian Control Conference (ASCC), pp. 674–677. IEEE (2022)
21. Yao, X., Zhang, C., Li, R., Sun, J., Liu, Z.: One-for-all: proposal masked cross-class anomaly detection. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 37, pp. 4792–4800 (2023)
22. You, Z., et al.: A unified model for multi-class anomaly detection. *Adv. Neural. Inf. Process. Syst.* **35**, 4571–4584 (2022)
23. Yu, J., et al.: Fastflow: unsupervised anomaly detection and localization via 2d normalizing flows. arXiv preprint [arXiv:2111.07677](https://arxiv.org/abs/2111.07677) (2021)
24. Zagoruyko, S., Komodakis, N.: Wide residual networks. arXiv preprint [arXiv:1605.07146](https://arxiv.org/abs/1605.07146) (2016)
25. Zavrtnik, V., Kristan, M., Skočaj, D.: Draem-a discriminatively trained reconstruction embedding for surface anomaly detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 8330–8339 (2021)
26. Zhang, H., Wang, Z., Wu, Z., Jiang, Y.G.: Diffusionad: denoising diffusion for anomaly detection. arXiv preprint [arXiv:2303.08730](https://arxiv.org/abs/2303.08730) (2023)

27. Zhao, Y.: Omnia: a unified CNN framework for unsupervised anomaly localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3924–3933 (2023)
28. Zhou, Y., Xu, X., Song, J., Shen, F., Shen, H.T.: Msflow: multi-scale flow-based framework for unsupervised anomaly detection. arXiv preprint [arXiv:2308.15300](https://arxiv.org/abs/2308.15300) (2023)



DSLA: A Distance-Sensitive Label Assignment Strategy for Oriented Object Detection in Remote Sensing Images

Minghong Wei¹, Yan Dong^{1,2}, Haobin Xiang¹, Guangshuai Gao¹,
and Chunlei Li¹(✉)

¹ School of Information and Communication Engineering,
Zhongyuan University of Technology, ZhengZhou, China
lichunlei1979@zut.edu.cn

² School of Automation Engineering, University of Electronic Science
and Technology of China, ChengDu, China

Abstract. Detectors based on convolutional neural networks (CNN) commonly employ label assignment to distinguish positive and negative samples during training. However, existing label assignment strategies overlook the diverse characteristics of objects in remote sensing images (RSI), such as arbitrary directions, large aspect ratios, and varying scales, which leads to inadequate and low-quality sample issues. To tackle these challenges, we propose a novel distance-sensitive label assignment (DSLA) strategy to effectively select both adequate and high-quality positive samples. Specifically, we design an elliptical region sampling (ERS) strategy to carefully screen candidate positive samples by utilizing elliptical regions, thereby mitigating background interference that hampers the model's performance. Furthermore, we propose a distance-controlled compensation loss (DC-Loss) to further enhance the effectiveness of ERS by reducing the impact of low-quality samples. Extensive experiments are conducted on two challenging datasets for rotated object detection, namely DIOR-R and HRSC2016, validate the superiority of our proposed method.

Keywords: remote sensing images · label assignment · objects with huge diversity · elliptical region sampling

1 Introduction

Oriented object detection (OOD) uses rotated bounding boxes to locate and identify objects of interest. In comparison to horizontal bounding boxes, rotated boxes offer greater accuracy and retain directional information about the objects, which are widely used in remote sensing object detection, facial recognition, scene text detection, and natural scenes [7, 15, 17, 26]. However, objects in remote sensing images often exhibit intricate distribution patterns, characterized by dense arrangements, varied orientations, and significant aspect ratios, which leads to

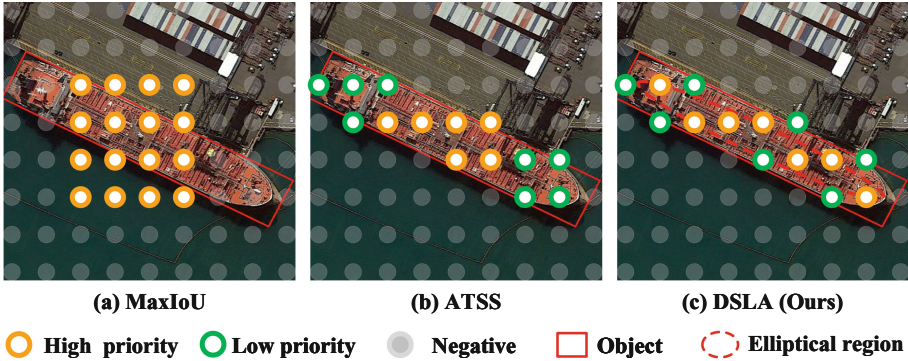


Fig. 1. Different strategies for the priority and selection range of positive samples. (a) MaxIoU uses a fixed IoU threshold for sampling. (b) ATSS assigns a dynamic IoU threshold based on simple distance. (c) DSLA strategy uses a dynamic threshold, and then prioritizes selecting high-quality samples from elliptical regions

inadequate sampling quantities and low quality. Oriented object detection still faces challenges in the field of remote sensing.

Numerous label assignment methods [9, 10, 20, 21, 30] have been developed for object detection, as they play a crucial role in determining positive or negative samples, which directly and significantly influences performance. However, these methods often overlook the actual shape and content of object intersection areas, which brings tough problems for remote sensing object detection. As shown in Fig. 1(a), [7, 16, 18, 26, 28] rely on the maximum union intersection (IoU) value between proposals and objects (MaxIoU for simplicity). Meanwhile, ATSS [30] introduced a sample selection strategy utilizing dynamic IoU thresholds, as illustrated in Fig. 1(b). Nonetheless, while increasing the number of samples, this method introduces a significant amount of complex background noise. Although these strategies are more effective than fixed assignment strategies, they exhibit the following issues: (1) disregarding the shape information of oriented objects, leading to insufficient sampling; (2) uniformly processing the selected positive samples without considering their quality leads to the introduction of background noise.

To address the aforementioned limitations in scale and spatial assignment, in this paper, we propose a distance-sensitive label assignment (DSLAs) strategy to dynamically select higher quality positive samples on multi-level feature maps, thereby improving detection performance. Specifically, we designed a novel and simple strategy, namely the elliptical region sampling (ERS) strategy, as shown in Fig. 1(c), to avoid insufficient sampling and low sample quality. In addition, a distance-controlled compensation loss (DC-loss) is proposed, which mitigates the impact of low-quality samples through the direction and shape characteristics of the object. Extensive experiments conducted on public remote sensing datasets such as DIOR-R and HRSC2016 have demonstrated the effectiveness and superiority of our proposed DSLAs.

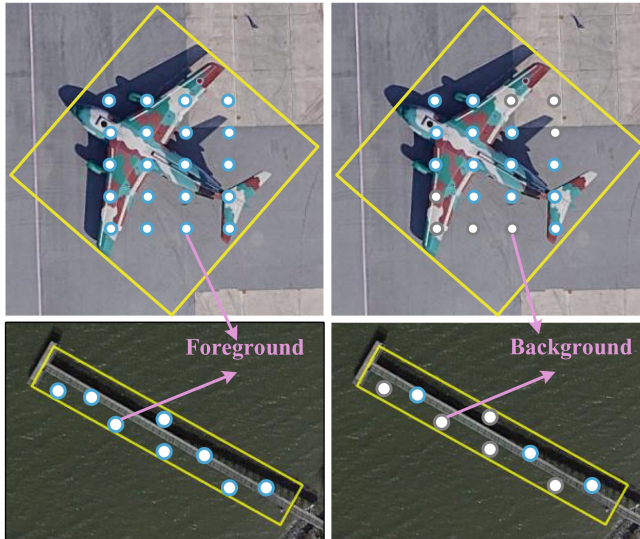


Fig. 2. Illustration of the difference between two sampling strategies. (Left column) ATSS. (Right column) DSLA

The contributions of this work are summarized as follows:

- 1) To address the issue of insufficient and low-quality sampling, we introduce a distance-sensitive label assignment (DSLA) strategy. This strategy dynamically selects positive samples across all feature levels.
- 2) We propose an elliptical region sampling (ERS) strategy that leverages the orientation and shape properties of objects to effectively select high-quality samples.
- 3) We design a distance-controlled compensation loss (DC-Loss) to further improve the quality of positive samples and mitigate the impact of low-quality ones.

The remaining sections of this article are as follows: the related work is reviewed in Sect. 2. Section 3 provides a detailed introduction to the method proposed in this article. In the Sect. 4, ablation experiments and comparative experiments were conducted on two publicly remote sensing datasets: DIOR-R and HRSC2016. Finally, the conclusion is drawn in the Sect. 5.

2 Related Work

2.1 Oriented Object Detection in Remote Sensing Images

The traditional oriented object detection method [11, 13] solves the problem of anchor box angle regression by preset rotating anchor boxes with different

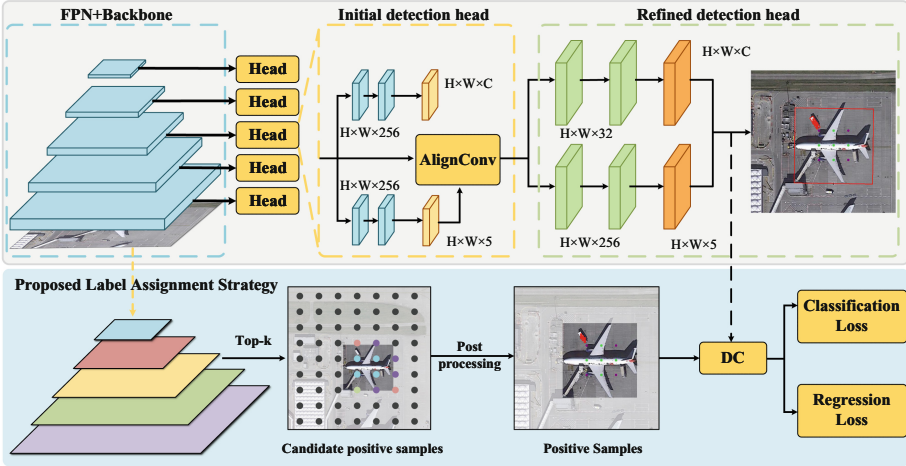


Fig. 3. Overview of the proposed DSLA. The upper and lower block represent baseline architecture of S2A-NET and our proposed DSLA strategy, respectively. The ERS strategy selects candidate samples based on the elliptical region of the object on each level of feature map. The candidate positive samples will be sorted by the distance from the center point of the ground truth bounding box. The top-k samples are selected as positive samples

widths, heights, and angles. However, presetting massive anchor boxes leads to the imbalance of positive and negative samples and redundant calculations. To solve the above problems, the method of horizontal anchor [2, 7] achieves regression by converting horizontal anchors to rotated anchors. Ding et al. [2] proposed a RoI transformer to achieve the transformation from horizontal anchors to rotated anchors. S2A-Net [7] proposes a feature alignment module that generates high-quality rotated anchors while only presetting single-scale horizontal anchors. Abandoning the traditional anchor-based method [6, 24] also achieved good detection results, Guo et al. [6] proposed a convex hull representation method to optimize the prediction box regression. Xu et al. [24] designed quadruple sliding vertices to represent objects and achieved point localization. Although the above methods have achieved considerable progress, there still suffers heavily from several drawbacks caused by objects with diverse distribution characteristics.

2.2 Label Assignment in Object Detection

Existing coarse-grained label assignment strategies (such as MAXIOU) have limitations when matching objects with large aspect ratios and angle variations, as they cannot guarantee that all objects can be matched with a sufficient number of positive samples during the sample assignment stage, which affects the detection performance of the model. Therefore, dynamic sample assignment strategies [14, 21, 30] are proposed, which used dynamic matching metrics for sample

assignment tasks. Zhang et al. [30] studied the impact of anchor-based sample assignment on model performance and constructed a sample assignment strategy based on the statistical characteristics of the object. Ming et al. [14] adaptively selected high-quality anchor boxes based on their ability to capture key features. For stable optimization, Sun et al. [21] explored how sample distribution influences the task of sample assignment. While the dynamic measurement based on IoU is simple and intuitive, it may not adequately address the requirements of object diversity distribution in remote sensing images. Huang et al. [10] introduced a general approach for representing positive samples using a two-dimensional Gaussian distribution. Although the above strategy effectively alleviates the problem of imbalanced samples, it requires prior setting of parameters and complex functions, as well as introducing low-quality samples.

3 The Proposed Method

This section provides a detailed introduction to the two key components of the proposed DSLA strategy: ERS strategy and DC-Loss. Implemented on the S2A-Net, the method’s pipeline is shown in Fig. 3. It consists of a backbone network, feature pyramid network (FPN), initial detection head, and refined detection head. The proposed DSLA strategy is implemented in the initial detection stage to select high-quality samples for objects with different shapes and arbitrary orientations. In Algorithm 1, we detail the sampling process incorporated in our DSLA strategy. Meanwhile, a DC-Loss is designed for the anchor-based bounding box regression, which automatically changes the form of regression loss function based on the center distance and angle deviation during training.

3.1 Elliptical Region Sampling Strategy

Assume the given ground truth $g_i(x_i, y_i, w_i, h_i, \theta_i)$ and the center point of an anchor box $a_j(x_j, y_j, w_j, h_j, \theta_j)$, which is already mapped back to the input image, elliptical region can be formulated as:

$$F(\cdot) = \begin{cases} \text{true,} & \frac{a^2}{(0.5w_i)^2} + \frac{b^2}{(0.5h_i)^2} < \eta \\ \text{false,} & \text{otherwise} \end{cases} \quad (1)$$

where the ratio factor η acts as an adaptive threshold based on the object’s shape. It controls the extent of the elliptical distribution, ensuring the selection of high-quality samples with minimal background noise. We will provide a detailed explanation of η in Sect. 4. Parameters a and b are calculated by the offset of the center point coordinates between g_i and a_j and the angle of g_i , which can be formulated respectively as:

$$\begin{aligned} a &= x_j \cos \theta_i + y_j \sin \theta_i \\ b &= x_j \sin \theta_i + y_j \cos \theta_i \end{aligned} \quad (2)$$

Algorithm 1. Position sensitive label assignment strategy.

Require:

- The set of ground truth bboxes for current batch, \mathcal{G} ;
- The set of preset anchor boxes for current batch, \mathcal{A} ;
- The set of each level in the pyramid layers, \mathcal{L} ;
- The sample number, $topk$;

Ensure:

- The set of positive samples \mathcal{P} and negative samples \mathcal{N} ;
 - 1: Compute the center points of the anchor box, $Points$;
 - 2: Compute the set of labels for the elliptical region with \mathcal{G} :
 $Flag = CheckPointsInEllipse(\mathcal{G}, Points)$ (1);
 - 3: **for** each level $l \in \mathcal{L}$ **do**
 - 4: Build an empty set for candidate samples: $\mathcal{C} \leftarrow \emptyset$
 - 5: $S_i \leftarrow$ Select k anchors from \mathcal{A} whose center are closest to the center of \mathcal{G}_i when $Flag_i = True$;
 - 6: **if** $k < topk$ **then**
 - 7: $E_i \leftarrow$ Select $topk - k$ anchors from \mathcal{A} whose center are closest to the center of \mathcal{G}_i when $Flag_i = False$;
 - 8: $S_i = E_i \cup S_i$
 - 9: **end if**
 - 10: $C_i = C_i \cup S_i$
 - 11: **end for**
 - 12: Compute threshold for each ground truth bboxes:
 $T = ComputThreshold(Cg, g)$ (3);
 - 13: **for** each candidate $c \in C_i$ **do**
 - 14: **if** the intersection union ratio between the candidate box and \mathcal{G} , $\mathcal{G} > T_i$ and center of c in \mathcal{G} **then**
 - 15: $\mathcal{P} = \mathcal{P} \cup c$
 - 16: **end if**
 - 17: **end for**
 - 18: $\mathcal{N} = \mathcal{A} - \mathcal{P}$
 - 19: **return** \mathcal{P}, \mathcal{N} ;
-

In this way, the sampling distribution can be adjusted dynamically according to the shapes of objects. As shown in Fig. 4, the sampling range tends to be a circular distribution when the shape of ground truth is close to a square. When the ground truths with an extremely large aspect ratio, the sampling range will approximate that of an inner tangent ellipse, which is better suited for such object shapes.

3.2 Dynamic Positive Sample Threshold

Based on the properties analyzed by the affine transformation, we propose a distance-based dynamic label assignment strategy to allocate sufficient samples for the hard ground truth. In detail, we set a monotonic decreasing function as

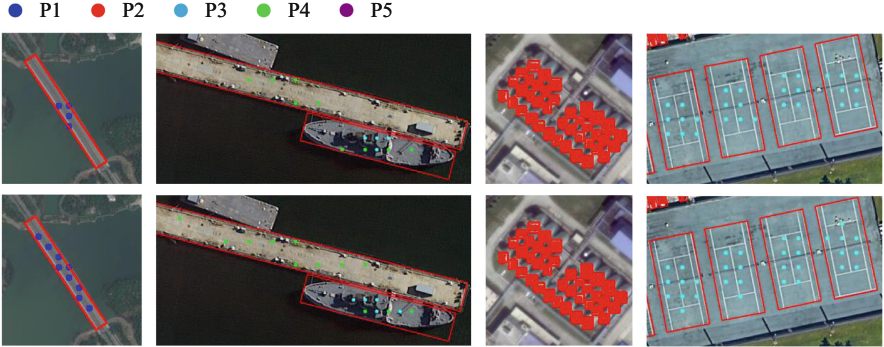


Fig. 4. Illustration of ATSS strategy (top row) and DSLA strategy (bottom row) for selecting positive samples of objects with different scales

the weighting factor for the IoU threshold. For a given ground truth g , the IoU threshold \mathcal{T}_i can be defined as:

$$\mathcal{T}_i = \mu + \sigma \quad (3)$$

where μ and σ represent the mean and standard deviation of the IoU between samples and the ground truth, which are defined as:

$$\mu = \frac{1}{N} \sum_{j=1}^N I_{i,j}, \sigma = \sqrt{\frac{1}{N} \sum_{j=1}^N (I_{i,j} - \mu)^2} \quad (4)$$

where N is the number of candidate samples, and $I_{i,j}$ is the IoU value between the i -th ground-truth box and the j -th prediction it matches.

3.3 Distance-Controlled Compensation Loss

In line with the baseline (S2A-Net), long edge definition is adopted to represent an arbitrary-oriented rectangle by five parameters (x, y, w, h, θ) , and the angle $\theta \in [-\pi/4, 3\pi/4]$. Therefore, the regression process of the sample can be formulated as:

$$A = \begin{pmatrix} S_x/S_y & 0 \\ 0 & (S_x/S_y) \cdot (a_x/a_y) \end{pmatrix} \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \quad (5)$$

where S_x and S_y are the scale differences of the x-axis and y-axis, a_x and a_y denote the aspect ratio differences of the x-axis and y-axis.

As illustrated in Fig. 4, the ERS strategy results in anchors' center points located in low-quality regions far from the center of the ground truth. As shown in Fig. 5, where the difficulty of sample regression is related to angle difference and center distance. Due to the geometric symmetry of the sample, when the

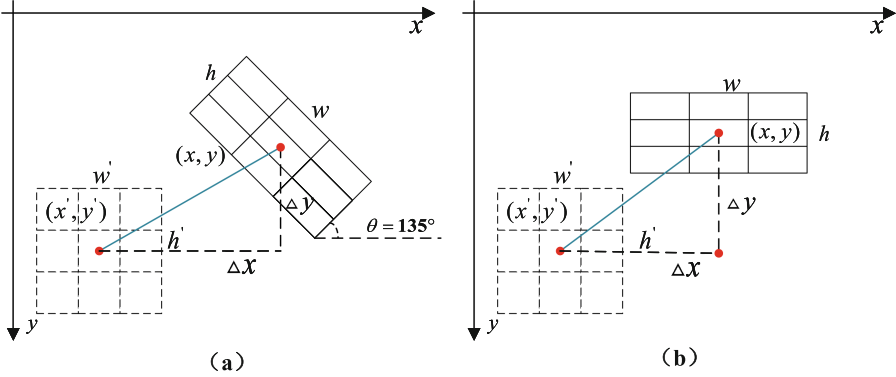


Fig. 5. Illustration of the mapping relationship between ground truth and label. (x, y, w, h, θ) and $(x', y', w', h', \theta')$ are the center, width, height and angle of ground truth and label. $(\Delta x, \Delta y)$ denotes the offset between them

angle is 0 or $-\pi/4$, there is no angle difference between the sample and the ground truth, and the difficulty of sample regression is the smallest. Angles of $-\pi/4$, $\pi/4$ or $3\pi/4$ represent the greatest deviations, making these samples the most challenging for regression. To mitigate above impact, we introduced DC-Loss to weight the regression loss and classification probability during the training process.

Specifically, for each positive sample a_i , the DC-Loss module calculates weight \mathcal{W} based on the L_2 distance and the deviation of angle between g_j and a_i , which is represented as L_{ij} and $\Delta\theta_{ij}$. Here, \mathcal{W} is formulated as:

$$\mathcal{W}(L_{ij}; \Delta\theta_{ij}) = e^{-L_{ij} \cdot \Delta\theta_{ij}} \quad (6)$$

where L_{ij} represents the center distance after normalization, which can be formulated as:

$$L_{ij} = \frac{\|g_c - a_c\|_2}{0.5 \cdot \|d_j\|_2} \quad (7)$$

where g_c and a_c represent the center point coordinates of g_j and a_i , respectively. d_j is the diagonal length of the g_j box. $\Delta\theta_{ij}$ represents the deviation between angles, which can be formulated as:

$$\Delta\theta_{ij} = \begin{cases} 0.5 + 0.5\sin^2\theta_{ij}, & -\frac{\pi}{4} \leq \theta_{ij} < \frac{\pi}{4} \\ 0.5 + 0.5\sin^2\left(\theta_{ij} - \frac{\pi}{2}\right), & -\frac{\pi}{4} \leq \theta_{ij} < \frac{3\pi}{4} \end{cases} \quad (8)$$

Finally, \mathcal{W} is used to measure the weight of candidate boxes participating in the loss function calculation. The total loss \mathcal{L}_{total} is denoted as:

$$\mathcal{L}_{total} = \mathcal{L}_{reg}^{init} + \mathcal{L}_{cls}^{ref} + \mathcal{L}_{reg}^{ref} + \mathcal{L}_{cls}^{ref} \quad (9)$$

\mathcal{L}_*^{init} and \mathcal{L}_*^{ref} representing the losses in the initial and refinement stages, respectively. The box regression loss is formulated as:

$$\mathcal{L}_{reg}^* = \frac{1}{N_i} \sum_{n=1}^N t'_n \sum_{j \in \{x,y,w,h,\theta\}} L_{reg}(v'_{nj}, v_{nj}, \mathcal{W}_j) \quad (10)$$

where N indicates the number of proposals, t_n represents the label of object, t'_n is a binary value ($t'_n = 1$ for foreground and $t'_n = 0$ for background, no regression for background). v_{*j} represents the predicted offset vectors, v_{*j} represents the objects vector of ground truth. The regression loss \mathcal{L}_{reg} adopts smooth L1 loss, which as defined in [4]. The classification loss is formulated as:

$$\mathcal{L}_{cls}^* = \begin{cases} -\delta(\mathcal{W}_j - p_n)^\gamma \log(p_n), & t_n = 1 \\ -\delta(\mathcal{W}_j)^\gamma \log(1 - p_n), & otherwise \end{cases} \quad (11)$$

where $p_n \in \{0, 1\}$ is the probability distribution of various classes calculated by Softmax function. δ and γ are hyperparameters of the focal loss [19].

4 Experiments

4.1 Datasets

DIOR-R [1] is an aerial image dataset annotated by oriented bounding boxes from the DIOR-R dataset. There are 23,463 images and 192,518 instances in this dataset, containing 20 common categories. The categories of objects in DIOR-R include Airplane (APL), Airport (APO), Baseball Field (BF), Basketball Court (BC), Bridge (BR), Chimney (CH), Expressway Service Area (ESA), Expressway Toll Station (ETS), Dam (DAM), Golf Field (GF), Ground Track Field (GTF), Harbor (HA), Overpass (OP), Ship (SH), Stadium (STA), Storage Tank (STO), Tennis Court (TC), Train Station (TS), Vehicle (VE) and Windmill (WM). **HRSC2016** [12] is a high-resolution ship remote sensing dataset collected from six famous ports. It includes 1061 images, all annotated with rotation boxes. The dataset is divided into training set, validation set, and test set, which contain 436, 181, and 444 images, respectively.

4.2 Experimental Details

In our experiments, for simplicity and efficiency, ResNet-50 pretrained on ImageNet is used as the backbone network, and FPN is employed as the neck network unless specified. The hyperparameters of SGD, i.e., weight decay, momentum and gamma, are set to be 1.0×10^{-2} , 0.9 and 0.1, respectively. For experiments on DIOR-R dataset, our network is trained with four NVIDIA V100 GPUs with 8 images per batch. We train the models on the HRSC2016 and DIOR-R datasets for 48 and 36 epochs, respectively, with a learning rate initialized to 6.25×10^{-2} . It was reduced by 10 times in iterations of 24, 32, and 38 epochs.

4.3 Ablation Study

In this section, a series of ablative experiments are conducted with DIOR-R dataset to illustrate the advantages of each proposed component in DSLA. Here, the components of proposed DSLA are indicated in abbreviated form, i.e., ‘-E’ indicates ERS strategy and ‘-D’ means DC-Loss. The second row shows the results using only the dynamic positive sampling threshold (DPST) strategy, which is strongly associated with the elliptical region sampling (ERS) strategy. The overall results of the ablative experiments are presented in Table 1. Specifically, the first row represents the results of our baseline detector, followed by the ablative results of 65.40 and 38.80 in AP50 and AP75, respectively, obtained by replacing the scale assignment with our ERS strategy. In addition, by adopting DC-Loss strategy, we achieve an improvement of 0.30 and 0.60 in AP50 and AP75, respectively, as compared with baseline. When combining all the components of DSLA together, we can achieve the AP50 and AP75 performance of 65.70 and 38.90, as shown in the last row of Table 1.

Moreover, in order to introduce the improvement for different categories, a more detailed experimental results for each category are provided in Table 2, and the contribution of improvement from each component is discussed in detail as follows.

Table 1. Ablative experiments and evaluations of the proposed method on the DIOR-R dataset. **Red** and **blue**: top two performances

DPST	ERS	DC-Loss	AP50	AP75
-	-	-	64.40	37.40
✓	-	-	64.60 (+0.20)	38.02(+0.62)
✓	✓	-	65.40 (+1.00)	38.80 (+1.40)
-	-	✓	64.70 (+0.30)	38.00 (+0.60)
✓	✓	✓	65.70 (+1.30)	38.90 (+1.50)

Table 2. Ablative experiments and evaluations of the proposed method on the DIOR-R dataset. The best result is highlighted in bold. All methods adopt ‘3x’ training schedule and use R-101 as backbone.

Method	APL	APO	BF	BC	BR	CH	ESA	ETS	DAM	GF	-
Baseline	62.80	43.80	74.10	81.50	41.10	72.60	80.40	70.70	27.30	77.90	-
DSLA(w/ -E)	62.80	52.20	75.50	81.50	42.50	78.30	80.00	70.10	31.50	78.20	-
DSLA(w/ -D)	67.70	50.10	75.30	81.50	41.30	72.70	79.20	70.20	29.10	76.70	-
DSLA(w/ -E, -D)	62.90	52.30	75.80	81.50	43.70	75.40	80.10	70.50	31.70	78.80	-
Method	GTF	HA	OP	SH	STA	STO	TC	TS	VE	WM	AP50
Baseline	80.90	44.70	57.10	80.80	67.90	69.30	81.50	59.20	48.90	65.70	64.40
DSLA(w/ -E)	80.30	46.00	57.00	80.90	70.40	68.60	81.50	57.80	48.10	65.00	65.40
DSLA(w/ -D)	80.30	44.60	56.70	80.80	68.30	68.90	81.60	55.70	49.00	63.80	64.70
DSLA(w/ -E, -D)	80.60	46.00	57.80	80.90	71.60	69.50	81.60	59.40	48.10	64.80	65.70

Effect of Elliptical Region Sampling Strategy. At the heart of this method lies the ERS strategy, which effectively addresses the issue of inadequate object sampling at extreme scales. For instance, quantitatively speaking, the proposed ERS strategy not only yielded a 1.00 increase in AP50 but also notably enhanced the AP performance for challenging-to-identify objects (such as APO, BR, and HA) by 8.40, 1.40, and 1.30, respectively, when compared to the baseline method, as illustrated in the first and second rows of Table 1. These results underscore the efficacy of the ERS strategy in improving detection accuracy across various object types.

Effect of Distance-Controlled Compensation Loss. Based on ERS, a DC-Loss module was proposed to mitigate the impact of low-quality samples on detection performance. Compared to the baseline method, this strategy improves the accuracy of AP50 and AP75 by 0.30 and 0.60, respectively. By compensating anchors based on the deviation of center distance and angle, samples with higher quality are given higher weights, thus obtaining samples with more feature information and avoiding the influence of anchors at the edge of the elliptical sampling area.

Table 3. Evaluation of using various ratios factor η in our strategy. **Red** and **blue**: top two performances

η	1	0.75	0.5	0.25
AP50	65.10	65.40	65.10	65.40
AP75	39.10	38.80	38.60	38.20

Table 4. Comparisons with the advanced oriented detectors on DIOR-R dataset. All methods rely on ‘3x’ training schedule and use R-50 as backbone. * indicates R-101 as backbone. † indicates random rotate data enhancement. **Red** and **blue**: top two performances

Method	APL	APO	BF	BC	BR	CH	ESA	ETS	DAM	GF	GTF	HA	OP	SH	STA	STO	TC	TS	VE	WM	mAP
Rotated RetinaNet [19]	59.54	25.03	70.08	81.01	28.26	72.02	55.35	56.77	21.26	65.70	70.28	30.52	44.37	77.02	59.01	59.39	81.18	38.43	39.10	61.58	54.83
SASM [9]	61.41	46.03	73.22	82.04	29.41	71.03	69.22	53.91	30.63	70.04	77.02	39.33	47.51	78.62	66.14	62.92	79.93	54.41	40.62	63.01	59.81
S2A-Net [7]	67.98	44.44	71.63	81.39	42.66	72.72	79.03	70.40	27.08	75.56	81.02	43.41	56.45	81.12	68.00	70.03	87.07	53.88	51.12	65.31	64.50
R3Det [26]	62.55	43.44	71.72	81.48	36.49	72.63	79.50	64.41	27.02	77.36	77.17	40.53	53.33	79.66	69.22	61.10	81.54	52.18	43.57	64.13	61.91
Gliding Vertex [24]	62.67	38.56	71.94	81.20	37.73	72.48	78.62	69.04	22.81	77.89	82.13	46.22	54.76	81.03	74.88	62.54	81.41	54.25	43.22	65.13	62.91
GWD [27]	66.52	46.80	71.76	81.43	40.81	78.25	79.23	66.63	29.01	78.68	80.19	44.88	57.23	80.91	74.17	68.02	81.48	54.63	47.80	64.41	64.63
KLD [29]	69.68	28.83	74.32	81.49	29.62	72.67	76.45	63.14	27.13	77.19	78.94	39.11	42.18	79.10	70.41	58.69	81.52	47.78	44.47	62.63	60.31
Rotated Faster RCNN [3]	66.52	46.80	71.76	81.43	40.81	78.25	79.23	66.63	29.01	78.68	80.19	44.88	57.23	80.91	74.17	68.02	81.48	54.63	47.80	64.41	64.63
Rotated FCOS [22]	62.31	42.18	75.34	81.32	39.26	74.89	77.42	68.67	26.00	73.94	78.73	41.28	54.19	80.61	66.92	69.17	87.20	52.31	47.08	65.21	63.21
Rotated ATSS [30]	62.19	44.63	71.55	81.42	41.08	72.37	78.54	67.50	30.56	75.69	79.11	42.77	56.31	80.92	67.78	69.24	81.62	55.45	47.79	64.10	63.52
RoI Trans. [2]	63.18	44.33	71.91	81.26	42.19	72.64	79.30	69.67	29.42	77.33	82.88	48.09	57.03	81.18	77.32	62.45	81.38	54.34	43.91	66.30	64.31
CFA [5]	61.10	44.93	77.62	84.67	37.69	75.71	82.68	72.03	33.41	77.25	79.94	46.20	54.27	87.01	70.43	69.58	81.55	55.51	49.53	64.92	65.25
ReDet [8]	63.22	44.18	72.11	81.26	43.83	72.72	79.10	69.78	28.45	78.69	77.18	48.24	56.81	81.17	69.17	62.73	81.42	54.90	44.04	66.37	63.81
Oriented RCNN [23]	63.31	43.10	71.89	81.17	44.78	72.64	80.12	69.67	33.78	77.92	83.11	46.29	58.31	81.17	74.54	62.32	81.29	56.30	43.78	65.26	64.53
DLSA(Ours)	68.60	49.70	71.70	81.50	42.50	76.80	79.70	68.60	31.80	77.90	80.60	44.80	56.40	80.90	70.70	69.40	81.60	58.10	48.30	64.70	65.20
DLSA(Ours)*	62.90	52.30	75.80	81.50	43.70	75.40	80.10	70.50	31.70	78.80	80.60	46.00	57.80	80.90	71.60	69.50	81.60	59.40	48.10	64.80	65.70
DLSA(Ours)†	71.30	53.90	77.40	89.40	44.60	78.20	86.60	72.10	37.30	78.10	82.50	46.90	58.60	81.00	75.00	69.00	88.90	62.60	49.00	65.40	68.40

4.4 Evaluation of Hyperparameters

Sample Number η . By incorporating a ratio factor into the ERS strategy, we enhance the quality of sample selection, thereby controlling the sample distribution. As illustrated in the Table 3, various tests were conducted to determine the optimal value of η , and the impact of different ratio factors was analyzed. Notably, AP75 serves as a high-precision detection benchmark, and the gradual decrease in AP75 values with varying η underscores the efficacy of our ERS strategy. When $\eta = 0.75$, we observed AP50 and AP75 values of 65.40 and 38.80, respectively, leading us to adopt a compromise in the detection results and select a ratio factor of 0.75.

4.5 Comparison with State-of-the-Art

Results on DIOR-R. Our DSLA strategy achieved an impressive mAP of 65.70, surpassing the baseline by 1.30, and the experimental results are presented in Table 4. Furthermore, when incorporating random rotation enhancement, our method achieved the best mAP results, particularly in detecting challenging types such as APO, BR and DAM. ATSS [30] is a well-established method for object detection in natural scenes. We integrated the ERS strategy to adapt it for oriented object detection in remote sensing, resulting in a 2.18% increase in mAP. Specifically, SASM [9] enhances the learning of objects with extreme aspect ratios by setting a dynamic threshold, but it barely considers the edge feature extraction of such objects and its mAP is 5.89% lower than ours. For four large aspect ratio objects, BR, DAM, HA and SH, our method improves mAP by 14.29%, 1.07%, 6.67%, and 2.28%, respectively. Compared with S2A-Net [7], our elliptical region sampling strategy is more friendly to large-scale and large aspect ratio objects, for example, APO, BF, and DAM are improved by 7.86%, 4.17%, and 4.62%, respectively. While slightly harming the detection performance of some objects such as STO and VE, the mAP is improved by 1.2%. Our detection results are visualized in Fig. 6, showcasing the method’s effectiveness in handling objects of diverse distributions, as further demonstrated in Fig. 7.

Results on HRSC2016. To further evaluate the proposed method’s robustness, we conducted experiments using the HRSC2016 dataset, and the results are detailed in Table 5. Notably, our DSLA strategy achieves a remarkable mAP of 90.30. The detection results on the HRSC2016 dataset, depicted in Fig. 8.

Table 5. Performance comparison with different state-of-the-art methods on the HRSC2016 dataset. **Red** and **blue**: top two performances

Method	Backbone	mAP
RoI-Transformer [2]	R-101	86.20
Gliding Vertex [24]	R-101	88.20
R3Det [26]	R-101	89.26
CSL [25]	R-101	89.62
DAL [14]	R-101	89.77
S2A-Net [7]	R-101	90.17
DSLA (Ours)	R-101	90.30

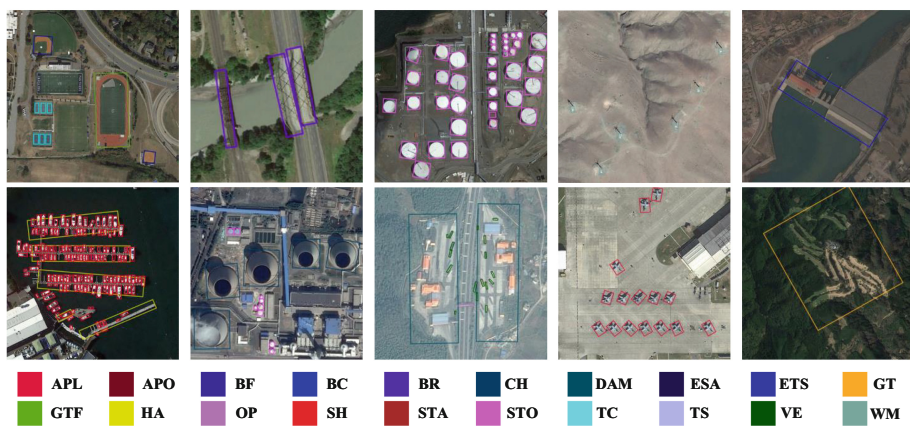


Fig. 6. Visualization of detection results on the DIOR-R dataset

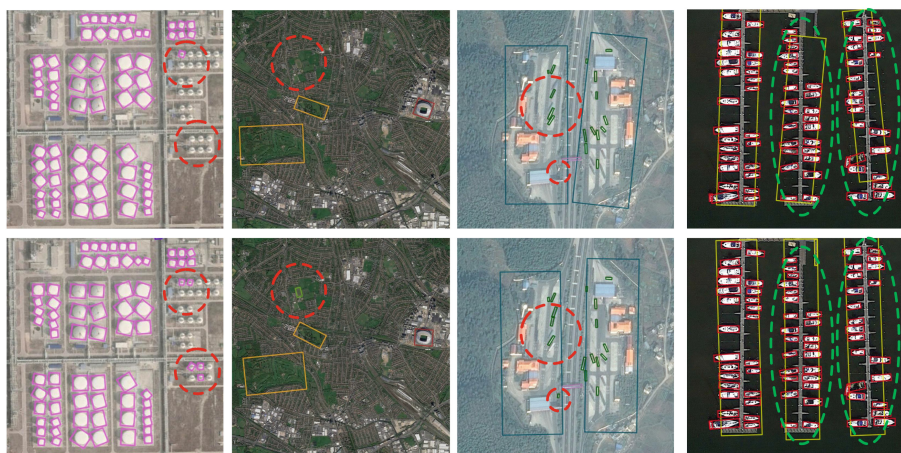


Fig. 7. Comparison of detection results for diverse distribution objects on the DIOR-R dataset. Baseline (top row) and ours method (bottom row)

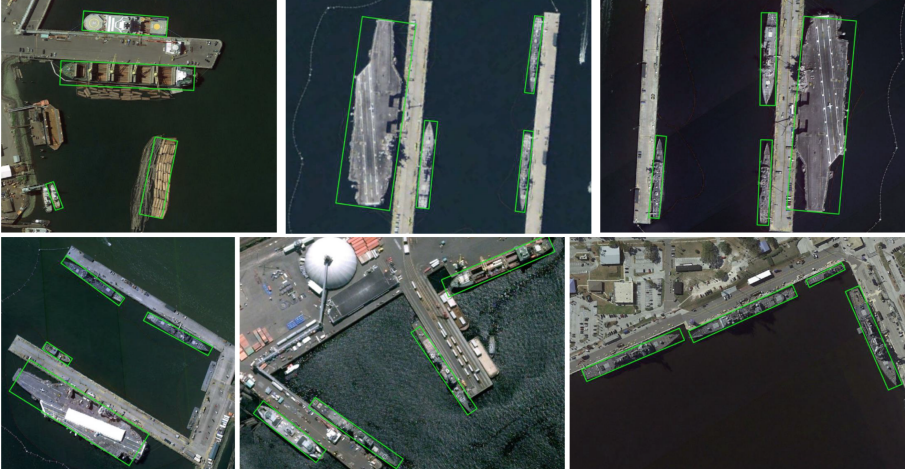


Fig. 8. Visualization of detection results on the HRSC2016 dataset

5 Conclusion

In this paper, we propose a novel and efficient strategy for label assignment, namely DSLA. A distance-sensitive label assignment strategy has been proposed for handling objects with significant distribution diversity, especially in remote sensing images, by setting dynamic thresholds from appropriate and continuous multi-level feature maps. To mitigate the impact of background noise, we design an elliptical region assignment method, which was adaptively controlled by the spatial shape of the objects. To learn high-quality information from selected training samples, a distance-controlled compensation loss was developed. Extensive experimental results demonstrated the effectiveness of our proposed.

References




1. Cheng, G., Wang, J., Li, K., Xie, X., Lang, C., Yao, Y., Han, J.: Anchor-free oriented proposal generator for object detection. *IEEE Trans. Geosci. Remote Sens.* **60**, 1–11 (2022)
2. Ding, J., Xue, N., Long, Y., Xia, G.S., Lu, Q.: Learning RoI transformer for oriented object detection in aerial images. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2849–2858 (2019)
3. Faster, R.: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* **9199**(10.5555), 2969239–2969250 (2015)
4. Girshick, R.: Fast R-CNN. In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1440–1448 (2015)
5. Guo, Z., Liu, C., Zhang, X., Jiao, J., Ji, X., Ye, Q.: Beyond bounding-box: convex-hull feature adaptation for oriented and densely packed object detection. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8792–8801 (2021)

6. Guo, Z., Zhang, X., Liu, C., Ji, X., Jiao, J., Ye, Q.: Convex-hull feature adaptation for oriented and densely packed object detection. *IEEE Trans. Circuits Syst. Video Technol.* **32**(8), 5252–5265 (2022)
7. Han, J., Ding, J., Li, J., Xia, G.S.: Align deep features for oriented object detection. *IEEE Trans. Geosci. Remote Sens.* **60**, 1–11 (2021)
8. Han, J., Ding, J., Xue, N., Xia, G.S.: ReDet: a rotation-equivariant detector for aerial object detection. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2786–2795 (2021)
9. Hou, L., Lu, K., Xue, J., Li, Y.: Shape-adaptive selection and measurement for oriented object detection. In: *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, pp. 923–932 (2022)
10. Huang, Z., Li, W., Xia, X.G., Tao, R.: A general Gaussian heatmap label assignment for arbitrary-oriented object detection. *IEEE Trans. Image Process.* **31**, 1895–1910 (2022)
11. Liu, Z., Wang, H., Weng, L., Yang, Y.: Ship rotated bounding box space for ship extraction from high-resolution optical satellite images with complex backgrounds. *IEEE Geosci. Remote Sens. Lett.* **13**(8), 1074–1078 (2016)
12. Liu, Z., Yuan, L., Weng, L., Yang, Y.: A high resolution optical satellite image dataset for ship recognition and some new baselines. In: *International Conference on Pattern Recognition Applications and Methods*, vol. 2, pp. 324–331. *SciTePress* (2017)
13. Ma, J., Shao, W., Ye, H., Wang, L., Wang, H., Zheng, Y., Xue, X.: Arbitrary-oriented scene text detection via rotation proposals. *IEEE Trans. Multimedia* **20**(11), 3111–3122 (2018)
14. Ming, Q., Zhou, Z., Miao, L., Zhang, H., Li, L.: Dynamic anchor learning for arbitrary-oriented object detection. In: *Proceedings of the AAAI conference on artificial intelligence*. vol. 35, pp. 2355–2363 (2021)
15. Qian, W., Yang, X., Peng, S., Zhang, X., Yan, J.: RSDet++: point-based modulated loss for more accurate rotated object detection. *IEEE Trans. Circuits Syst. Video Technol.* **32**(11), 7869–7879 (2022)
16. Qin, R., Liu, Q., Gao, G., Huang, D., Wang, Y.: MRDet: a multihead network for accurate rotated object detection in aerial images. *IEEE Trans. Geosci. Remote Sens.* **60**, 1–12 (2021)
17. Raghunandan, K., Shivakumara, P., Roy, S., Kumar, G.H., Pal, U., Lu, T.: Multi-script-oriented text detection and recognition in video/scene/born digital images. *IEEE Trans. Circuits Syst. Video Technol.* **29**(4), 1145–1162 (2018)
18. 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)
19. Ross, T.Y., Dollár, G.: Focal loss for dense object detection. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2980–2988 (2017)
20. Shi, L., Kuang, L., Xu, X., Pan, B., Shi, Z.: CANet: centerness-aware network for object detection in remote sensing images. *IEEE Trans. Geosci. Remote Sens.* **60**, 1–13 (2021)
21. Sun, P., Zheng, Y., Wu, W., Xu, W., Bai, S.: Metric-aligned sample selection and critical feature sampling for oriented object detection. *arXiv preprint [arXiv:2306.16718](https://arxiv.org/abs/2306.16718)* (2023)
22. Tian, Z., Shen, C., Chen, H., He, T.: FCOS: fully convolutional one-stage object detection. *arxiv 2019. arXiv preprint [arXiv:1904.01355](https://arxiv.org/abs/1904.01355)* (1904)

23. Xie, X., Cheng, G., Wang, J., Yao, X., Han, J.: Oriented R-CNN for object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 3520–3529 (2021)
24. Xu, Y., et al.: Gliding vertex on the horizontal bounding box for multi-oriented object detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **43**(4), 1452–1459 (2020)
25. Yang, X., Yan, J.: Arbitrary-oriented object detection with circular smooth label. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) *ECCV 2020. LNCS*, vol. 12353, pp. 677–694. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58598-3_40
26. Yang, X., Yan, J., Feng, Z., He, T.: R3Det: refined single-stage detector with feature refinement for rotating object. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp. 3163–3171 (2021)
27. Yang, X., Yan, J., Ming, Q., Wang, W., Zhang, X., Tian, Q.: Rethinking rotated object detection with Gaussian Wasserstein distance loss. In: International Conference on Machine Learning, pp. 11830–11841. PMLR (2021)
28. Yang, X., et al.: SCRDet: towards more robust detection for small, cluttered and rotated objects. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 8232–8241 (2019)
29. Yang, X., et al.: Learning high-precision bounding box for rotated object detection via Kullback-Leibler Divergence. *Adv. Neural. Inf. Process. Syst.* **34**, 18381–18394 (2021)
30. Zhang, S., Chi, C., Yao, Y., Lei, Z., Li, S.Z.: Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9759–9768 (2020)



BridgeCLIP: Automatic Bridge Inspection by Utilizing Vision-Language Model

Poweï Liao¹ and Gaku Nakano²

¹ Waseda University, Tokyo, Japan
presley_liao@fuji.waseda.jp

² NEC Corporation, Tokyo, Japan
g-nakano@nec.com

Abstract. We propose BridgeCLIP, an innovative framework designed to harness the power of vision-language models for bridge inspection from images. BridgeCLIP is a CLIP-based multi-label classifier that finds multiple damages in a single bridge image. Pre-trained vision-language models learn the relationships between general objects by millions of text-image pairs, of which descriptions are not precise enough for domain-specific problems. Following the concept that humans normally learn the visual appearance of bridge damage by reading a manual, we introduce a novel Description Attention Module (DAM) to incorporate the domain-specific knowledge extracted from the professional descriptions in bridge inspection manuals. By utilizing both general knowledge of the pre-trained CLIP and professional knowledge of bridge inspection, BridgeCLIP comprehensively learns the inter-class relationships of different damages. Experimental results on bridge inspection datasets show that BridgeCLIP outperforms the state-of-the-art multi-label classifiers.

Keywords: Bridge Inspection · Vision-Language Models · Multi-label Classification

1 Introduction

As a crucial part of city infrastructure, bridges facilitate transportation and connect communities, making them indispensable in our daily lives. Tokyo in Japan has over one thousand bridges that support the city's bustling life and commerce. However, 43% of the bridges have been built for more than 50 years since their construction before the period of rapid economic growth, and the aging situation of these bridges is becoming more severe [41]. This social issue is not unique to Japan; now, it is a global concern, with countries worldwide facing similar issues [2, 42]. In addition to the elapsed years, the daily conditions of bridges are very severe. Wet conditions, strong winds, vehicle vibrations, and even earthquakes or natural disasters accelerate the deterioration. Without

*P. Liao's contribution was made when he was an intern at NEC Corporation.

regular maintenance, minor damages can escalate into significant deterioration, eventually resulting in disasters with profound economic and social losses. For instance, the I-40 Bridge collapse in Oklahoma in 2002 profoundly impacted freight flow movement in the U.S. highway network [1]. The collapse affected nearby highway network links and those further away, indicating the extensive economic implications of such events on local, regional, and national economies.

The importance of bridge inspection cannot be overstated to avert such disasters. By bridge inspection, damages can be found before they cause irretrievable consequences. However, conducting these inspections requires specialized knowledge in civil engineering and often involves working in dangerous locations. With the global labor force in decline, the need for more professionals is becoming increasingly critical for bridge inspection.

Addressing challenges like the laborious unsafe conditions and high costs of traditional bridge inspections, the interest in using advanced technologies has remarkably increased over the last decade. Laser scanners have been widely applied for bridge engineering [35, 43]; however, the cost of laser scanning technology remains relatively high. Consequently, bridge inspection methodologies utilizing vision-based techniques coupled with deep learning algorithms have gained increasing research interest [7, 8, 24]. In vision-based bridge inspection, accurately detecting all damages within the images is crucial. Multi-label bridge inspection, which aims at detecting multiple types of damages from a single image, remains challenging due to the following reasons:

- The relationships between different damage types (inter-class relations) is unclear.
- A substantial amount of data is required for training.
- Precise annotation for multi-label classification is difficult and costly.

To facilitate the comprehension of the inter-class relationships, we incorporate a pre-trained vision-language model into our framework.

In recent years, models trained under the supervision of natural language have achieved remarkable success in the computer vision community. A notable example is CLIP [34], which was pre-trained on 400 million web-scraped image-text pairs, demonstrating promising capabilities across various datasets. By incorporating the pre-trained model of CLIP, some traditional vision tasks such as object detection [17, 46, 54] and semantic segmentation [27, 30, 49] can be interfaced with text. Through the pre-trained text-feature space CLIP has learned, the vision-language model can understand the inter-class relationships and facilitate multi-label classification [38]. However, despite CLIP’s impressive performance on general datasets, e.g., ImageNet [11], the performance will greatly decrease on tasks that demand highly specialized knowledge, e.g., EuroSAT [19] and DTD [10]. Since the knowledge that CLIP learned is general, researchers have explored strategies to adapt CLIP for domain-specific tasks such as transforming prompts into learnable vectors [53] or appending a Multilayer Perceptron (MLP) to the CLIP encoders [16].

In this paper, we aim to investigate the capabilities of CLIP for a task that highly requires professional knowledge, such as bridge inspection. Typically,

reading a manual proves helpful when humans with general knowledge want to learn how to do bridge inspection. In the manual, instead of only providing the name of every damage, it will also provide detailed descriptions of each type of damage, enabling inspectors to combine this specific information with their pre-existing general knowledge to learn bridge inspection skills. Inspired by this learning process, we introduce a novel Description Attention Module (DAM) to enable CLIP to “read” and “learn” from such manuals. We design this module to enhance the ability of the proposed framework, BridgeCLIP, to perform bridge inspection tasks by combining the knowledge from pre-trained CLIP and professional descriptions. In summary, our contribution is two-fold:

- We propose BridgeCLIP for multi-label bridge inspection. The Description Attention Module (DAM) effectively adapts a pre-trained vision-language model with professional descriptions.
- We conduct experiments on bridge inspection datasets (dacl10k [15] and CODEBRIM [32]) and demonstrate that our BridgeCLIP improves mAP by 3.02% on dacl10k and 3.77% on CODEBRIM over DualCoOp [38].

2 Related Work

2.1 Image-Based Bridge Inspection

Vision plays a pivotal role in bridge inspection, offering abundant information to detect various damage types. Over the past decade, many researches have been focusing on crack detection by various computer vision methods like histograms of oriented gradients (HOG) [23], Hough transform [33], object detection [12, 14], motion analysis [3], and semantic segmentation [28, 29, 50]. However, while crack detection is vital, comprehensive bridge inspection demands attention to a broader spectrum of damages, such as corrosion, efflorescence, and so on. To deal with various damage types in bridge inspection, images with different damage labels are collected. They are used for different algorithms, e.g., multi-classes classification [21], multi-label classification [32], and semantic segmentation [15]. Recent studies have also explored the use of vision-language models for bridge inspection. Chun et al. [9] trained a vision-language model to do image captioning for bridge inspection. Kunlamai et al. [25] utilized a vision-language model to conduct visual question answering (VQA) in bridge inspection.

2.2 Multi-label Classification

Single-label classification, identifying the primary object in an image, is one of the most popular computer vision tasks [11]. Meanwhile, the more complex challenge of multi-label classification [40], which aims to identify multiple objects in an image, is more related to real-world usage and gaining increasing research interest in recent years. Although multi-label classification can be performed by simply transferring to multiple single-label classifications, this approach often fails to consider the relationship between labels. Recent developments

have introduced methodologies employing Graph Neural Networks (GNN) [5, 6], RNN/LSTM [4, 45], and vision-language models [38], which have proven effective in learning and leveraging the relationships between labels, thereby facilitating significant progress in multi-label classification tasks.

2.3 Vision-Language Models Adaption

Vision-language models such as CLIP [34] and ALIGN [22] have demonstrated that they can understand the relationship between vision and nature language in everyday contexts, a capability acquired through extensive training on numerous images with corresponding text descriptions. Leveraging this capability, these pre-trained models can be adapted for various downstream tasks, including but not limited to object detection [17, 46, 54], image segmentation [13, 27, 30, 49], image editing [26, 47, 48], and image captioning [20, 31].

However, in focusing on classification, the original CLIP model struggles with domain-specific datasets such as EuroSAT [19] and DTD [10]. A promising solution to this issue is “prompt learning”, where prompt significantly influences the model’s performance. To avoid tedious prompt engineering, CoOp [53] introduces learnable vectors as prompts, allowing the model to train and optimize these prompts. CoCoOp [52] employs a lightweight network to dynamically adapt prompts. Other than prompt learning, Clip-adapter [16] adds a bottleneck layer to learn new features, utilizing residual connections to preserve the model’s original pre-trained knowledge. DualCoOp [38] encodes positive and negative contexts with class names, by binary classification for every class individually, can be adapted for multi-label classification.

3 Methodology

3.1 Approach Overview

Figure 1 illustrates the overview of the proposed method. Our approach addresses the challenge of identifying multiple damage types of bridges in a single image through multi-label classification. Since identifying inter-class relationships is vital in multi-label classification [5, 6, 40], we employ a powerful pre-trained vision-language model (CLIP [34]), which is adept at understanding complex class relationships through extensive image-text pair learning. To adapt the pre-trained model to our task, we train the Description Attention Module (DAM) while freezing the parameters of the image encoders and the text encoders of CLIP. Within DAM, a pair of learnable prompts and an attention module are designed to associate the general knowledge from the pre-trained CLIP text encoder with the specialized knowledge from the description. This fusion enables our model to adjust to the demands of bridge inspection applications, even in the absence of extensive data sets. Additionally, to recognize all the bridge damage in different regions of the image, we apply Class-Specific Region Feature Aggregation [38] in our method. During the training, we optimize the network by minimizing Asymmetric Loss (ASL) [36].

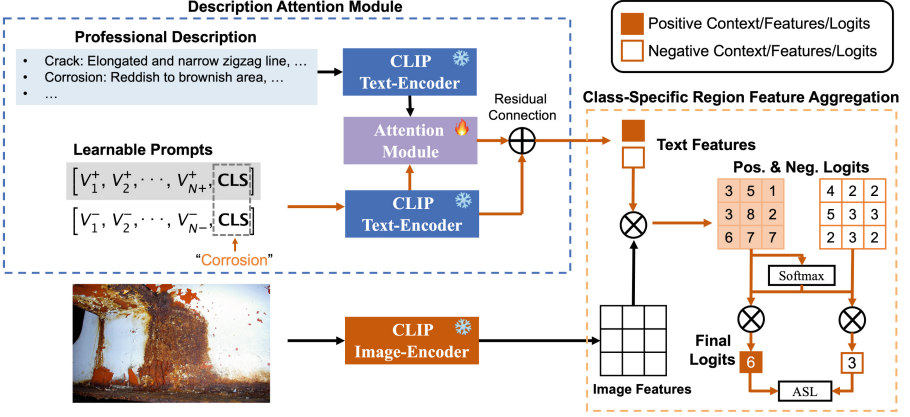


Fig. 1. Overview of the proposed approach. BridgeCLIP learns a pair of prompts and an Attention Module to adapt powerful pre-trained Vision-Language encoders for multi-label bridge inspection.

3.2 Description Attention Module

We present the Description Attention Module (DAM) as an innovative adaptation of the CLIP framework for bridge inspection applications, enhancing its multi-label classification capabilities. Unlike previous approaches that utilize a single learnable prompt [53], we set a pair of learnable prompts [38] for multi-label classification. Each prompt pair consists of a positive and a negative context. A series of learnable vectors act as the learnable prompts and follow the name of the targeted bridge damage (Fig. 2).

There are N learnable vectors V_1, \dots, V_N in each prompt with a class name (damage type) CLS. A pair of a positive and negative prompt can be written by

$$\text{Prompt}^+ = [V_1^+, V_2^+, \dots, V_{N^+}^+, \text{CLS}], \quad (1)$$

$$\text{Prompt}^- = [V_1^-, V_2^-, \dots, V_{N^-}^-, \text{CLS}]. \quad (2)$$

These vectors are input alongside the damage’s class name into the text encoder E_t , yielding the Classes Name Text-Feature F_c and the Descriptions Text-Feature F_d generated from professional descriptions:

$$F_c = E_t(\text{Prompt}^+, \text{Prompt}^-), \quad (3)$$

$$F_d = E_t(\text{Descriptions}). \quad (4)$$

To let our model “read” and “learn” from professional descriptions, we concatenate F_c and F_d , then input it into an Attention Module. We get the output of a pair of features F_{Att}^+ and F_{Att}^- by

$$F_{Att}^+, F_{Att}^- = \text{ReLU}(w(\text{softmax}(q(F_c)k(F_c, F_d)^T)v(F_c, F_d))), \quad (5)$$

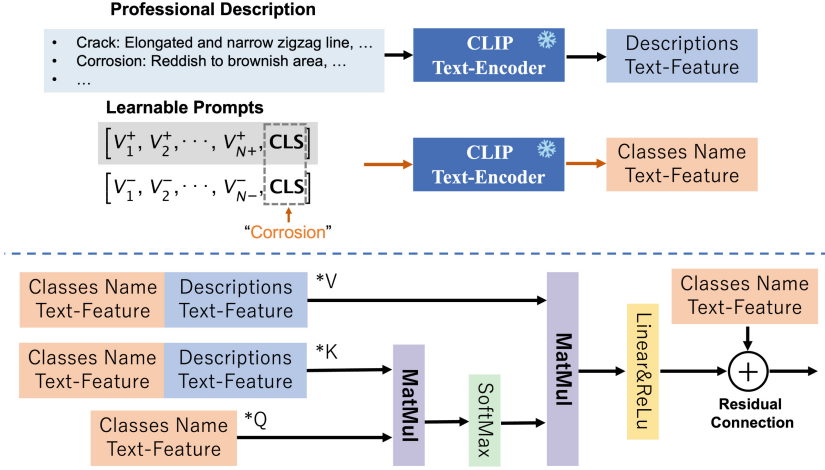


Fig. 2. Illustration of Description Attention Module (DAM). Descriptions Text-Feature and Classes Name Text-Feature, generated from profession description and a pair of learnable prompts, are mixed by an attention module.

where w , q , k , and v represent independent linear embedding layers. To keep the original knowledge of CLIP, we set a residual connection after the Attention Module output as

$$F_{DAM}^+ = \alpha F_{Att}^+ + (1 - \alpha) E_t(\text{Prompt}^+), \tag{6}$$

$$F_{DAM}^- = \alpha F_{Att}^- + (1 - \alpha) E_t(\text{Prompt}^-). \tag{7}$$

The ratio α can be manually adjusted during the training. The features F_{DAM}^+ and F_{DAM}^- will be used to calculate the cosine similarity in Class-Specific Region Feature Aggregation [38].

3.3 Class-Specific Region Feature Aggregation

During bridge inspections, images often contain multiple damages, which may be spread across various regions. To accurately recognize all damages in disparate regions, we incorporate Class-Specific Region Feature Aggregation [38] in our approach. In the original CLIP [34], the last attention-pooling layer can be presented as follows:

$$\begin{aligned} \text{AttnPool}(x) &= \text{Proj}_{v \rightarrow t} \left(\sum_i \text{softmax} \left(\frac{q(\bar{x})k(x_i)^T}{C} \right) \cdot v(x_i) \right) \\ &= \sum_i \text{softmax} \left(\frac{q(\bar{x})k(x_i)^T}{C} \right) \cdot \text{Proj}_{v \rightarrow t}(v(x_i)) \\ &= \text{Pool}(\text{Proj}_{v \rightarrow t}(v(x_i))), \end{aligned} \tag{8}$$

which pools the visual feature map initially, followed by the projection of the global feature vector into the text space. The q , v , and k are independent linear layers, and x is the output feature map of the visual encoder. x_i denotes the input feature at spatial location i , and \bar{x} is the average of all x_i . By removing the pooling operation in the last multi-headed attention pooling layer of the visual encoder in CLIP [34], we can project the vision feature x_i of each region i to textual feature space as follows [51]:

$$F_v^i = \text{Proj}_{v \rightarrow t}(v(x_i)) \quad (9)$$

Subsequently, each image feature F_v^i is compared against both positive and negative features F_{DAM}^+, F_{DAM}^- from the DAM, using cosine similarity S_{cos} to produce positive and negative logits:

$$S_{i,m}^+ = S_{\text{cos}}(F_v^i, F_{DAM}^+), S_{i,m}^- = S_{\text{cos}}(F_v^i, F_{DAM}^-). \quad (10)$$

After aggregating the positive and negative logits $S_{i,m}^+, S_{i,m}^-$, we obtain a final positive and negative logit for every class by

$$S_m^+ = \sum_i (\text{softmax}(S_{i,m}^+) \cdot S_{i,m}^+), \quad (11)$$

$$S_m^- = \sum_i (\text{softmax}(S_{i,m}^-) \cdot S_{i,m}^-). \quad (12)$$

Finally, with a pair of the final logits, the binary classification output p can be given by

$$p = \frac{\exp(S_m^+/\tau)}{\exp(S_m^+/\tau) + \exp(S_m^-/\tau)}. \quad (13)$$

3.4 Optimization

During the optimization, we apply ASL [36] to address the imbalance between positive and negative labels in multi-label classification. Specifically, we calculate the losses for positive {image, label} pairs L_+ , and negative {image, label} pairs L_- , using the following formulas:

$$L_+ = (1 - p)^{\gamma^+} \log(p), \quad (14)$$

$$L_- = p_m^{\gamma^-} \log(1 - p_m), \quad (15)$$

where p_m denotes the *shifted probability*, which fully discards negative pairs when the possibility is very low, and is defined as

$$p_m = \max(p - m, 0), \quad (16)$$

where m is a margin for hard thresholds. The loss-weight for L_- is greater than or equal to L_+ to ensure that ASL minimizes the influence of hard thresholds on these easy negatives. The learnable prompts and the attention module in DAM are then refined through back-propagation of ASL across the frozen text encoder.

4 Experiments




Image	Damage: Description
	<p>Crack: Elongated and narrow zigzag line. Clearly darker compared to the surrounding area or black.</p> <p>Rust: Reddish to brownish area. Often appears on concrete surfaces and metallic objects.</p>
	<p>Crack: Elongated and narrow zigzag line. . .</p> <p>Rust: Reddish to brownish area. . .</p> <p>Graffiti: All kinds of paintings on concrete and objects apart from defect markings.</p> <p>Weathering: Summarizes all kinds of weathering on the structure (e.g., smut, dirt, debris) and . . .</p>
	<p>Efflorescence: Mostly roundish areas of white to yellowish or reddish color. Strong efflorescence can . . .</p> <p>Hollowareas: Hollowareas are not visually recognizable but their markings made with crayons . . .</p> <p>Cavity: Small air voids. Mostly on vertical surfaces.</p> <p>Spalling: Spalled concrete area revealing the coarse aggregate. Significantly rougher surface (texture) . . .</p> <p>Weathering: Summarizes all kinds of weathering . . .</p>

Fig. 3. Examples from dacl10k datasets. Examples of bridge damage images and corresponding damages and descriptions from dacl10k [15].

4.1 Multi-label Bridge Inspection

Dataset. To show our method can recognize multiple damage types from a single bridge image, we tested our model on two publicly available datasets, CODEBRIM [32] and dacl10k [15]. In CODEBRIM [32], there are five different types of bridge damage and 1590 images in the dataset. Dacl10k [15] is a dataset for semantic segmentation, and their annotation can be easily transferred for multi-label classification. There are 9,920 annotated images and 18 classes in the dacl10k dataset.



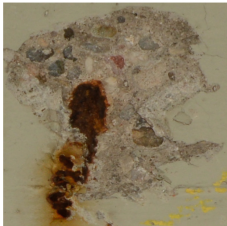
Image	Damage: Description
	<p>Efflorescence: Mostly roundish areas of white to yellowish or reddish color. Strong efflorescence can look similar to stalactites. Often appears in weathered (Weathering) or wet areas (WetSpot) of the building and in combination with Crack and/or Rust.</p> <p>Corrosion Stain: Reddish to brownish area. Often appears on concrete surfaces and metallic objects.</p>
	<p>Crack: Elongated and narrow zigzag line. Clearly darker compared to the surrounding area or black.</p>
	<p>Spallation: Spalled concrete area revealing the coarse aggregate. Significantly rougher surface (texture) inside the Spalling than in the surrounding surface.</p> <p>ExposedBars: Exposed Reinforcement (nonprestressed and prestressed) and cladding tubes of tendons. Often appears in combination with Spalling or Rockpocket, and Rust.</p> <p>Corrosion Stain: Reddish to brownish area ...</p>

Fig. 4. Examples from CODEBRIM datasets. Examples of bridge damage images and corresponding damages and descriptions from CODEBRIM [32].

Table 1. Multi-label bridge inspection dacl10k [15]

Method	Val Set				Test Set			
	Precision	Recall	F1	mAP	Precision	Recall	F1	mAP
VGG16 [37]	58.45	23.78	31.92	49.94	58.63	22.24	30.29	45.83
ResNet50 [18]	54.85	50.40	51.16	52.49	53.29	48.15	48.51	50.47
DualCoOp [38]	50.38	59.61	53.90	56.79	50.51	59.84	53.87	56.32
BridgeCLIP (ours)	51.75	62.33	55.94	60.59	50.52	62.72	55.65	59.34

Professional Descriptions. For each damage type, we defined a prompt with a professional description as follows:

$$\{\text{damage class name}\} : \{\text{description}\} \quad (17)$$

The descriptions CODEBRIM and dacl10k are named the same as the corresponding dataset because the descriptions contain the same number of descriptions as damage types in the dataset. Some examples of image and description pairs are shown in Figs. 3 and 4. There are 13 descriptions in dacl10k and 6 in CODEBRIM, respectively.

Table 2. Multi-label bridge inspection CODEBRIM [32].

Method	Val Set				Test Set			
	Precision	Recall	F1	mAP	Precision	Recall	F1	mAP
VGG16 [37]	62.12	69.72	65.39	71.70	63.50	73.09	67.65	73.84
ResNet50 [18]	62.77	69.81	65.79	72.92	66.47	72.20	68.70	73.69
DualCoOp [38]	62.00	77.90	67.41	78.64	67.31	81.83	71.90	83.75
BridgeCLIP (ours)	66.03	81.38	71.93	83.24	69.23	85.18	75.26	87.52

Evaluation. On the dacl10k and CODEBRIM datasets, we reported the average overall precision, recall, F1, and mean average precision (mAP) for both the valuation set and the test set.

Implementation. In our implementation, ResNet-50 [18] serves as the visual encoder across all baselines, and the input resolution is 448×448 pixels. The text encoding component utilizes the same Transformer architecture [44] in CLIP [34]. Both the visual and text encoders are initialized using weights from the pre-trained CLIP model and are maintained without alterations (frozen) during the optimization process. Optimization is conducted using the Stochastic Gradient Descent (SGD) optimizer, starting with an initial learning rate of 0.02. This rate undergoes adjustment according to the cosine annealing rule throughout training.

Baselines. To evaluate the effectiveness of BridgeCLIP for multi-label bridge inspection, we compared our method with the baseline DualCoOp. We also trained two prevalent models VGG16 [37] and ResNet50 [18] with cross-entropy loss.

Result. Tables 1 and 2 show that our method BridgeCLIP consistently shows superior performance in recall, F1, and mAP across both datasets, indicating its robustness and effectiveness in the multi-label bridge inspection task. While VGG16 shows high precision, it falls short in recall, suggesting a need for a better balance. ResNet50 and DualCoOp offer competitive but not leading performances. The result suggests BridgeCLIP offers a significant advancement in multi-label bridge inspection, by effectively balancing precision and recall and achieving high mAP scores.

4.2 Ablation Study

Residual Ratio α . DAM incorporates a residual connection after the attention module to preserve the inherent knowledge gained by the CLIP text encoder through extensive training on millions of text-image pairs. We varied the residual ratio α to elucidate its effect on the model’s efficiency. As Fig. 5 shows, when

setting the ratio to 0, the model will not use any knowledge from professional descriptions, significantly reducing the performance of models. Our experiment also indicates that an α of 0.8 for dacl10k and 0.6 for CODEBRIM datasets optimizes performance. The performance remains relatively stable for α adjustments between 0.2 and 0.8 but noticeably declined at $\alpha = 1.0$. This trend highlights the importance of keeping both original CLIP knowledge and professional knowledge.

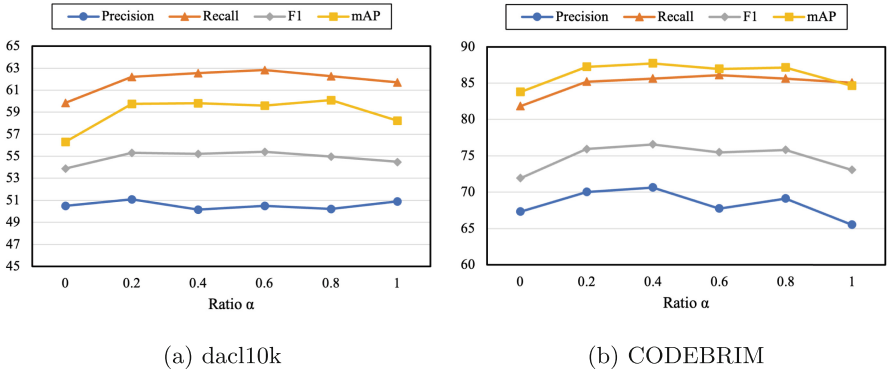


Fig. 5. Varying the residual ratio α . The Precision, Recall, F1, and mAP of BridgeCLIP (ours) trained on dacl10k and CODEBRIM dataset when the residual ratio α in DAM is varied.

Professional Descriptions. To understand how the content of professional descriptions would affect our model, we trained our model on two datasets with three different descriptions. In addition to dacl10k and CODEBRIM, we also collected the damage descriptions from NILIM Japan [39], containing 26 diverse bridge damage categories in Japan standard bridge inspection. The descriptions were written in Japanese and translated into English using ChatGPT. We chose 23 descriptions of common damages as in dacl10k and CODEBRIM.

Table 3. Different descriptions. The mAP of our BridgeCLIP with various descriptions. * indicates using a prompt for descriptions

Test data	Descriptions for training					
	CODEBRIM	CODEBRIM*	dacl10k	dacl10k*	NILIM	NILIM*
dacl10k	59.62	60.06	60.10	59.34	59.78	59.38
CODEBRIM	87.38	87.52	87.69	87.32	87.31	87.29

Table 4. Examples of different descriptions. Three examples of descriptions from dacl10k and NILIM.

Damage Class	Descriptions examples
Cracking (dacl10k)	Elongated and narrow zigzag line. Clearly darker compared to the surrounding area or black.
Cracking (NILIM)	Surface crack(s) on concrete components.
Rust (dacl10k)	Reddish to brownish area. Often appears on concrete surfaces and metallic objects
Rust (NILIM)	Rusting in ordinary steel materials, leading to reduced thickness. In weathering steel, it's when protective rust fails to form, causing abnormal rust. Common in parts with water accumulation or poor ventilation.
Spalling (dacl10k)	Spalled concrete area revealing the coarse aggregate. Significantly rougher surface (texture) inside the Spalling than in the surrounding surface.
Spalling (NILIM)	Peeling or flaking concrete layers, possibly revealing the steel bars underneath, which may show signs of rust.

As shown in the Table 3, the variation in descriptions suggests a nuanced impact on the model’s performance compared to the significant differences between them. Table 4 provides examples demonstrating how the same damage class can have vastly different descriptions from different sources. The introduction of prompts (denoted by *) for descriptions appears to offer marginal improvements in some cases (e.g., CODEBRIM from 87.38 to 87.52) but shows a slight decrease in others (e.g., dacl10k from 60.10 to 59.34). This suggests that while prompts can help model learn more when the number of descriptions is limited, when the number of descriptions is enough, prompts will not help the model and even decrease the performance. Although descriptions from NILIM have the most different kinds of bridge types, our method performs best with dacl10k descriptions. This suggests that the descriptions related to the dataset could help our model better.

5 Conclusion and Discussion

This study introduced BridgeCLIP, a novel approach for multi-label bridge inspection leveraging a vision-language model enhanced with a Description Attention Module (DAM). Our findings demonstrate that integrating DAM with the pre-trained CLIP model significantly improves the model’s ability to interpret professional descriptions, thereby enhancing its performance on specialized tasks such as bridge damage classification. Experimental results on the dacl10k

and CODEBRIM datasets underscore our method’s superiority over existing state-of-the-art method, with improvements in precision, recall, F1 scores, and mean Average Precision (mAP).

The incorporation of DAM enables BridgeCLIP to effectively utilize text features extracted from professional descriptions, a key advancement over previous methods. This innovation not only bolsters the model’s accuracy but also its applicability in real-world scenarios where expertise in bridge inspection is crucial.

Future research could explore further optimizations to the DAM and the integration of additional descriptions to enhance the model’s robustness and accuracy. By continuing to refine BridgeCLIP, we aim to contribute to the development of automated inspection systems that can aid in maintaining and ensuring the safety of critical infrastructure globally.

References

1. Aydin, S.G., Shen, G., Pulat, P.: A retro-analysis of I-40 bridge collapse on freight movement in the us highway network using GIS and assignment models. *Int. J. Transp. Sci. Technol.* **1**(4), 379–397 (2012)
2. Boller, C., Starke, P., Dobmann, G., Kuo, C.M., Kuo, C.H.: Approaching the assessment of ageing bridge infrastructure. *Smart Struct. Syst.* **15**(3), 593–608 (2015)
3. Chaudhury, S., Nakano, G., Takada, J., Iketani, A.: Spatial-temporal motion field analysis for pixelwise crack detection on concrete surfaces. In: 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 336–344. IEEE (2017)
4. Chen, T., Wang, Z., Li, G., Lin, L.: Recurrent attentional reinforcement learning for multi-label image recognition. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 32 (2018)
5. Chen, T., Xu, M., Hui, X., Wu, H., Lin, L.: Learning semantic-specific graph representation for multi-label image recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 522–531 (2019)
6. Chen, Z.M., Wei, X.S., Wang, P., Guo, Y.: Multi-label image recognition with graph convolutional networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5177–5186 (2019)
7. Choi, Y., Choi, Y., Cho, J., Kim, D., Kong, J.: Utilization and verification of imaging technology in smart bridge inspection system: an application study. *Sustainability* **15**(2), 1509 (2023)
8. Chun, P., et al.: Utilization of unmanned aerial vehicle, artificial intelligence, and remote measurement technology for bridge inspections. *J. Robot. Mechatron.* **32**(6), 1244–1258 (2020)
9. Chun, P.J., Yamane, T., Maemura, Y.: A deep learning-based image captioning method to automatically generate comprehensive explanations of bridge damage. *Comput.-Aided Civil Infrastruct. Eng.* **37**(11), 1387–1401 (2022)
10. Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., Vedaldi, A.: Describing textures in the wild. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3606–3613 (2014)
11. 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)

12. Deng, J., Lu, Y., Lee, V.C.S.: Concrete crack detection with handwriting script interferences using faster region-based convolutional neural network. *Comput.-Aided Civil Infrastruct. Eng.* **35**(4), 373–388 (2020)
13. Dong, X., et al.: MaskCLIP: masked self-distillation advances contrastive language-image pretraining. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10995–11005 (2023)
14. Fang, F., Li, L., Gu, Y., Zhu, H., Lim, J.H.: A novel hybrid approach for crack detection. *Pattern Recogn.* **107**, 107474 (2020)
15. Flotzinger, J., Rösch, P.J., Braml, T.: dacl10k: benchmark for semantic bridge damage segmentation. In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 8626–8635 (2024)
16. Gao, P., et al.: Clip-adapter: Better vision-language models with feature adapters. *Int. J. Comput. Vis.* **132**(2), 581–595 (2024)
17. Gu, X., Lin, T.Y., Kuo, W., Cui, Y.: Open-vocabulary object detection via vision and language knowledge distillation. *arXiv preprint [arXiv:2104.13921](https://arxiv.org/abs/2104.13921)* (2021)
18. 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)
19. Helber, P., Bischke, B., Dengel, A., Borth, D.: EuroSAT: a novel dataset and deep learning benchmark for land use and land cover classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **12**(7), 2217–2226 (2019)
20. Hessel, J., Holtzman, A., Forbes, M., Bras, R.L., Choi, Y.: CLIPscore: a reference-free evaluation metric for image captioning. *arXiv preprint [arXiv:2104.08718](https://arxiv.org/abs/2104.08718)* (2021)
21. Hühthwohl, P., Lu, R., Brilakis, I.: Multi-classifier for reinforced concrete bridge defects. *Autom. Constr.* **105**, 102824 (2019)
22. 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)
23. Kapela, R., et al.: Asphalt surfaced pavement cracks detection based on histograms of oriented gradients. In: *2015 22nd International Conference Mixed Design of Integrated Circuits & Systems (MIXDES)*, pp. 579–584. IEEE (2015)
24. Karim, M.M., Qin, R., Chen, G., Yin, Z.: A semi-supervised self-training method to develop assistive intelligence for segmenting multiclass bridge elements from inspection videos. *Struct. Health Monit.* **21**(3), 835–852 (2022)
25. Kunlamai, T., Yamane, T., Suganuma, M., Chun, P.J., Okatani, T.: Improving visual question answering for bridge inspection by pre-training with external data of image-text pairs. *Comput.-Aided Civil Infrastruct. Eng.* **39**(3), 345–361 (2024)
26. Kwon, G., Ye, J.C.: CLIPstyler: image style transfer with a single text condition. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18062–18071 (2022)
27. Liang, F., et al.: Open-vocabulary semantic segmentation with mask-adapted clip. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7061–7070 (2023)
28. 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)
29. 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)
30. Lüddecke, T., Ecker, A.: Image segmentation using text and image prompts. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7086–7096 (2022)

31. Mokady, R., Hertz, A., Bermano, A.H.: ClipCap: CLIP prefix for image captioning. arXiv preprint [arXiv:2111.09734](https://arxiv.org/abs/2111.09734) (2021)
32. Mundt, M., Majumder, S., Murali, S., Panetsos, P., Ramesh, V.: Meta-learning convolutional neural architectures for multi-target concrete defect classification with the concrete defect bridge image dataset. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11196–11205 (2019)
33. Quintana, M., Torres, J., Menéndez, J.M.: A simplified computer vision system for road surface inspection and maintenance. *IEEE Trans. Intell. Transp. Syst.* **17**(3), 608–619 (2015)
34. Radford, A., et al.: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning, pp. 8748–8763. PMLR (2021)
35. Rashidi, M., Mohammadi, M., Sadeghlou Kivi, S., Abdolvand, M.M., Truong-Hong, L., Samali, B.: A decade of modern bridge monitoring using terrestrial laser scanning: review and future directions. *Remote Sens.* **12**(22), 3796 (2020)
36. Ridnik, T., et al.: Asymmetric loss for multi-label classification. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 82–91 (2021)
37. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint [arXiv:1409.1556](https://arxiv.org/abs/1409.1556) (2014)
38. Sun, X., Hu, P., Saenko, K.: DualCoOp: fast adaptation to multi-label recognition with limited annotations. *Adv. Neural. Inf. Process. Syst.* **35**, 30569–30582 (2022)
39. Tamakoshi, T., Ookubo, M., Hoshino, M., Yokoi, Y., Kowase, Y.: Reference to MLIT’s bridge inspection manual (2013) – photographs related to damage rating and maintenance urgency ratings. Technical Note 748, National Institute for Land and Infrastructure Management, Ministry of Land Infrastructure Transport and Tourism Japan (2013)
40. Tarekegn, A.N., Giacobini, M., Michalak, K.: A review of methods for imbalanced multi-label classification. *Pattern Recogn.* **118**, 107965 (2021)
41. Tokyo Bureau of Construction: Current status of bridges (2020). https://www.kensetsu.metro.tokyo.lg.jp/jigyo/road/kanri/gaiyo/yobouhozen/kyouryou_genjou.html. Accessed 19 Jan 2024
42. Torti, M., Venanzi, I., Ubertini, F., et al.: Seismic structural health monitoring for reducing life cycle cost of road bridges. In: EURO DYN 2020 XI International Conference on Structural Dynamics PROCEEDINGS Volume I, vol. 1, pp. 1063–1074. Institute of Structural Analysis and Antiseismic Research School of Civil (2020)
43. Truong-Hong, L., Falter, H., Lennon, D., Laefer, D.F.: Framework for bridge inspection with laser scanning. In: EASEC-14 Structural Engineering and Construction, Ho Chi Minh City, Vietnam, 6-8 January 2016 (2016)
44. Vaswani, A.: Attention is all you need. *Adv. Neural Inf. Process. Syst.* 5998–6008 (2017)
45. Wang, J., Yang, Y., Mao, J., Huang, Z., Huang, C., Xu, W.: CNN-RNN: a unified framework for multi-label image classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2285–2294 (2016)
46. Wang, Z., et al.: CLIP-TD: CLIP targeted distillation for vision-language tasks. arXiv preprint [arXiv:2201.05729](https://arxiv.org/abs/2201.05729) (2022)
47. Wei, T., et al.: HairCLIP: design your hair by text and reference image. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 18072–18081 (2022)

48. Wei, T., et al.: HairCLIPv2: unifying hair editing via proxy feature blending. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 23589–23599 (2023)
49. Xie, J., Hou, X., Ye, K., Shen, L.: CLIMS: cross language image matching for weakly supervised semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4483–4492 (2022)
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. Zhou, C., Loy, C.C., Dai, B.: Extract free dense labels from CLIP. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) *ECCV 2022*. LNCS, vol. 13688, pp. 696–712. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-19815-1_40
52. 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. pp. 16816–16825 (2022)
53. 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)
54. Zhou, X., Girdhar, R., Joulin, A., Krähenbühl, P., Misra, I.: Detecting twenty-thousand classes using image-level supervision. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) *ECCV 2022*. LNCS, vol. 13669, pp. 350–368. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-20077-9_21



Writer Identification in Multiple Medieval Books: A Preliminary Study

Tiziana D'Alessandro , Claudio De Stefano , Francesco Fontanella ,
and Alessandra Scotto di Freca 

Department of Electrical and Information Engineering (DIEI), University of Cassino
and Southern Lazio, Via G. Di Biasio 43, 03043 Cassino, FR, Italy
{tiziana.dalessandro,destefano,fontanella,a.scotto}@unicas.it

Abstract. One of the key areas of study in palaeography involves identifying the various scribes collaborating on a medieval book. Although digital technologies have allowed significant improvements in this field, it is far from being solved in the general case and is still an open issue. Very interesting results were obtained in the case of highly standardized handwriting and book typologies, where the analysis of some basic layout features regarding the organization of the page and its exploitation by the scribe allowed a high recognition rate. The main drawback of approaches based on layout features is that the results obtained from an ancient text are difficult to use in other texts, produced following different standards. Based on these considerations, we have developed a new approach that attempts to overcome the above-mentioned limitations. The basic idea is to exploit the knowledge of palaeographers who have identified, for each scribe, some letters or abbreviations that characterize them. In this preliminary study, we used two ancient manuscripts, the *Avila Bible* and the *Trento Bible*, and we considered the letter “a” as a reference symbol: such letter, according to the indications of the palaeographers, is one of the distinctive symbols able to characterize individual scribes and it is also widely present in all pages of text. A template matching technique was used to identify the occurrences of the character “a” on each page, and a Convolutional Neural Network (CNN) was used to train a classification system capable of attributing each occurrence of the character “a” to the corresponding scribe. Finally, we used a majority voting technique to assign the entire manuscript page to the scribe with the highest number of occurrences of the character “a” on that page. The experimental results obtained on both Bibles confirmed the effectiveness of our method, allowing us to correctly attribute to each scribe over 80% of the pages processed.

1 Introduction

Palaeography is the study of ancient writing, particularly concerning deciphering, dating, and interpreting manuscripts and documents from various historical periods. This field includes examining writing systems, styles, letterforms,

abbreviations, punctuation, and other aspects of written communication, often in languages no longer in common use [3, 4, 18, 22, 27, 28].

A very important aspect of palaeographic studies is the identification of the different scribes who contributed to the production of an ancient text. Scribes, in fact, often had distinct writing styles, whose knowledge can help scholars attribute specific parts of a text to a specific author or period. Furthermore, identifying different scribes can help trace how a text was completed over time and understand the different places scribes worked. Finally, examining how copying and annotation work was carried out can provide insight into the culture and other aspects of the social and intellectual life of the time [9, 14–16, 23].

In this area, over the years, there has been an increasingly intense use of digital technologies, which have made it possible to integrate traditional paleographic methods with a vast range of techniques such as image processing, machine learning, recognition of writing patterns and styles, which helped to decode and interpret ancient handwritten texts more efficiently and accurately [17, 25]. It is useful to highlight, however, that the problem of identifying the different hands that produced an ancient text is still far from being solved in the general case and still represents one of the most difficult challenges to face.

As discussed in [6], techniques for identifying scribes in ancient manuscripts can be divided into two main categories. In the first category, we can consider approaches based on the analysis of single letters, signs, or abbreviations obtained by examining single lines of text or the entire page [19, 24]. All these approaches are based on the possibility of effectively segmenting the manuscript text into letters or graphemes: a condition which, in general, is very difficult to guarantee and often produces unsatisfactory results [5].

The second category includes techniques for extracting features from the entire manuscript page. These techniques use texture or layout features and have produced particularly interesting results in the case of highly standardized handwriting and book typologies, for which the analysis of some basic layout features, regarding the organization of the page and its exploitation by the scribe, may give precious information for distinguishing very similar hands even without recourse to paleographical analysis [2, 20].

In previous studies [7, 10–12], we proposed some pattern recognition systems for distinguishing the scribes who worked together to transcribe a single medieval Latin book. We used a specifically devised set of features directly derived from page layout analysis according to the suggestions of palaeographic and codicological researchers and performed classification using standard machine learning systems. We have also developed deep neural network approaches, in which we proposed a deep transfer learning solution for row detection and page classification, obtaining very encouraging results [6, 8]. Obviously, the main problem of approaches based on layout features is that the results obtained from an ancient text are difficult to use in other texts, produced following different standards. The effect is that it is difficult, for example, to recognize the hand of a scribe who worked on multiple ancient texts produced following different standards.

Based on these considerations, we have developed a new approach that attempts to overcome the above-mentioned limitations. The basic idea is to exploit the knowledge of palaeographers who have identified, for each scribe, some letters or abbreviations that characterize them. In this preliminary study, we used two ancient manuscripts, the Avila Bible and the Trento Bible, and we considered the letter “a” as a reference symbol: such letter, according to the indications of the palaeographers, is one of the distinctive symbols able to characterize individual scribes and it is also widely present in all pages of text. For each scribe, a template image of the character “a” was obtained with the help of paleographers, and a template matching technique was used to identify in the image of the entire page the regions (sub-images) where with higher probability an occurrence of the character “a” is present. In this way, we have overcome the problem of segmenting lines of text into individual characters. These images were then processed using Convolutional Neural Networks (CNNs) to train a classification system capable of attributing each occurrence of the character “a” to the corresponding scribe. Finally, we used a majority voting technique to assign the entire manuscript page to the scribe with the highest number of occurrences of the character “a” on that page.

The results obtained on the Avila Bible were very interesting, showing performance higher than those obtained previously using layout information. It is useful to highlight that the proposed approach is independent of all the style rules defined for producing an ancient volume and attempts to characterize the distinctive aspects of each scribe. We, therefore, tried to apply our system, trained on the Avila Bible, on the images relating to the pages of the Trento Bible. The studies carried out by paleographers on the Trent Bible have, in fact, highlighted that at least one of the scribes who worked on the Avila Bible also contributed to the writing of the Trent Bible. Also, in this case, the results were very convincing and allowed us to obtain good percentages of correct identification of this scribe in the Trento Bible.

The remainder of the paper is organized as follows: Sect. 2 illustrates the datasets derived from both the Avila and the Trento bibles, Sect. 3 describes the proposed method discussing the different parts in which it is articulated, while the experimental results are presented in detail in Sect. 4. Discussion and future works are eventually left to Sect. 5.

2 Data Description

As previously discussed, we aim to perform the identification of a scribe who participated in the handwriting of two Medieval Bibles, namely the Avila Bible and the Trento Bible. The Avila Bible was penned in Italy by a minimum of nine scribes during the third decade of the 12th century [21]. Subsequently, it was transported to Spain, where local scribes completed both the text and decoration. In a third phase, occurring in the 15th century, additional content was

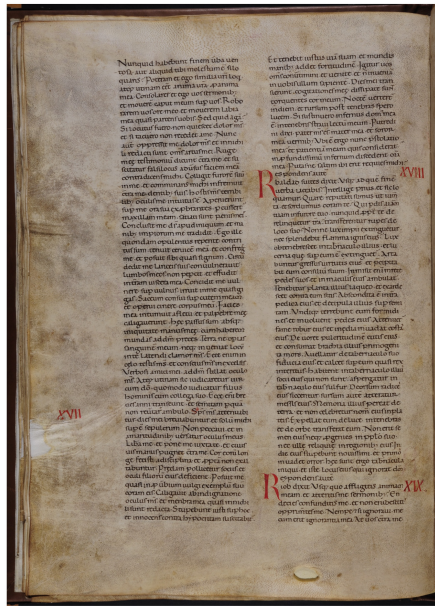
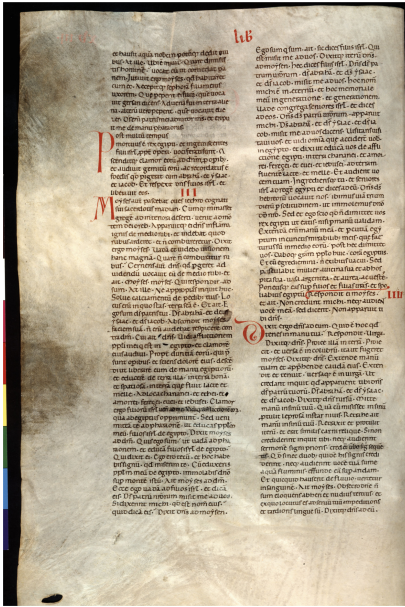


Fig. 1. Examples of pages from the Avila (left) and Trento (right) Bibles

incorporated by yet another writer. Due to the involvement of multiple scribal hands, both contemporary and non-contemporary, this manuscript serves as a rigorous testbed for assessing the efficacy of automatic scribal identification systems. As far as we know, there is currently no other standard database offering the same characteristics, such as complete high-quality reproductions and a limited number of recurring and identifiable hands. High-quality images of the Bible pages are available online [1]. The scarcity of comparable databases can be attributed primarily to the rarity of these immense Bibles, their substantial size, and the high costs associated with digitization. Consequently, they have not been frequently digitized, and the available microfilms are often inadequate for conducting automatic pattern recognition analyses due to low image quality and compromised page margins. The Avila Bible comprises 870 two-column pages, although, for this study, several pages of very poor quality were excluded. Palaeographers analyzing this manuscript have identified at least 13 distinct scribal hands. One of these hands may work only for rubricated letters that we opted to remove during preprocessing; thereby, we exclude this writer from the identification. Furthermore, paleographers furnished us with guidance on the distinctive identifying letters used solely by 12th-century contemporaneous Italian writers. Consequently, we chose to omit pages authored by scribes from the 15th century and Spanish writers. Scribes contributed to the Bible in varying capacities; some penned only a few pages (with one page being the minimum), while others were responsible for a significant portion of the text (with 143 pages being the maximum). Consequently, the classification task at hand

is marked by a notably imbalanced distribution of samples per class. Thus, we opted to exclude pages of scribes penned only a limited number of pages due to the extensive dataset requirement for training deep-learning algorithms. In this study, we focused on 705 pages featuring identifiable handwriting from 8 different writers to ensure an ample dataset for DL applications. Each page was digitized at a resolution of 4100×6110 pixels and meticulously labeled by an expert paleographer, assigning the letters *A, B, D, E, F, G, H, I* to denote individual writers.

The Trento Bible is an Atlantic volume dating back to the first half of the 12th century. The history of the volume is less complex than the Avila Bible and centered on a more limited period, without subsequent additions and modifications. The existing part of the Bible, which has come down to the present, has been analyzed by palaeographic experts. Palaeographers widely attribute the decorative elements within the bible to a singular hand, referred to as the Master of the Avila Bible and his atelier. Additionally, paleographers identified three scribes who collaborated in writing the Trento Bible, and among them, they identified a scribe already involved in the composition of the Avila Bible (the *F* scribe). While the entire Trento Bible has been digitized, the availability of digital images of satisfactory quality is limited. Trento Bible consists of 394 pages. Each page was digitized at a resolution of 2832×4256 pixels. Figure 1 compares pages from the Avila and the Trento Bible.

3 The Proposed Method

This section outlines the method proposed in this study, which unfolds in two main stages: image processing and DL approach. The workflow overview is depicted in Fig. 2. The input comprises image datasets of ancient Bibles (Avila and Trento Bibles as detailed in Sect. 2). Section 3.1 delineates the first step of the method, where many image data processing techniques were employed. Subsequently, Sect. 3.2 describes the DL architectures and techniques adopted to identify the writer who contributed to both the ancient books. Additionally, we have included Figs. 3, 4 and 5 to describe the following sections better, providing a more detailed visual explanation of each step involved in the processes. To highlight the generality of the proposed approach, in the following figures, we will indicate the Avila Bible, from which the reference patterns have been extracted, as Bible1, while the Trento Bible will be denoted as Bible2.

3.1 Image Processing

The initial phase of image processing takes as input a reference letter image for each author of the Avila Bible, along with the images of the pages from both the Avila and Trento Bibles. This phase resulted in the creation of two datasets comprising extracted instances of the letter “a” from the pages of each Bible. The details of this procedure are illustrated in Fig. 3, which delineates the process into distinct modules, each described in the following subsections.

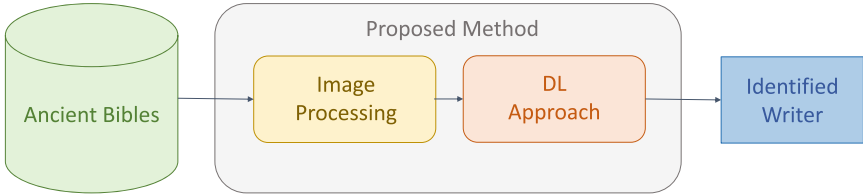


Fig. 2. The architecture of the proposed method

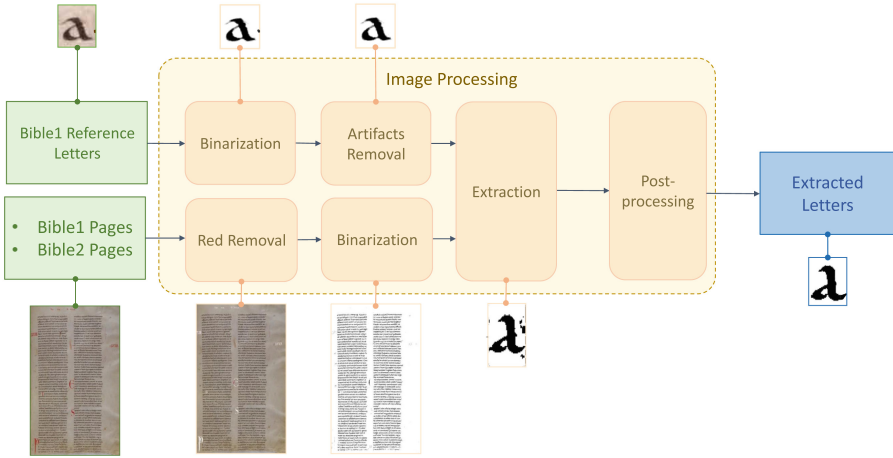


Fig. 3. The image processing scheme

Avila Reference Letters Enhancement. To build a dataset for each bible, we opted to utilize the letter “a” as recommended by the paleographer, given its prevalence as the most common letter, ensuring the extraction of an adequate number of samples. Furthermore, the paleographer selects this letter because it is particularly instrumental in identifying the writer. As discussed in Sect. 2, we considered eight distinct scribes in the Avila manuscript, each attributing a reference sample of the letter “a”. Notably, these reference letters differ among authors, showcasing their unique styles and idiosyncrasies in handwriting.

The reference samples provided by the paleographers required preprocessing to prepare them for the subsequent steps. Each sample underwent binarization to facilitate analysis. Moreover, given that each sample contained the letter itself and adjacent letter fragments, a cleaning operation was necessary to eliminate artefacts. To address the cleaning operation, we first applied a thinning algorithm to reduce the letters to their essential structure, making the contours clearer and easier to detect. Then, we considered the findContours function [29] from the

OpenCV library used for image contour detection, where a contour is a curve joining all the continuous points along a boundary with the same colour or intensity. Using this function, we analyzed the ratio of black to white pixels at the image edges to determine if the area contained part of the main letter (high black presence) or just noise from adjacent characters (high white presence). This process significantly refined our letter images by removing unwanted artefacts. Additionally, resizing the images was essential to standardize their height to match the average row height of the Bible pages and adjust the base without altering the aspect ratio. At this point, the prepared images were ready to extract letter occurrences from the pages.

Page Processing. The pages of the Bibles considered for this study were digitized with high resolution, ensuring clarity and legibility of the text while minimizing any signs of degradation. Despite their overall quality, it became imperative to perform binarization to facilitate the application of identification methods and, subsequently, DL techniques. Given that the images were in RGB format and typically consisted of three distinct colors - background texture, text, and initial characters or symbols often marked in red - the binarization process required a preliminary operation. A strategic decision was made to convert all red pixels to white. This was crucial for enhancing the clarity of the text and simplifying subsequent processing steps.

After removing the red pixels, additional transformations were applied to prepare the images for further analysis. Firstly, a conversion to grayscale was performed to standardize the representation of intensity levels across the image. Then binarization was carried out utilizing the Otsu algorithm.

By employing these preprocessing steps, the digitized images were effectively transformed into a format adequate to advanced analytical techniques, enabling the identification of occurrences and subsequent DL-based text analysis.

Letters Extraction and Postprocessing. As a result, the steps previously described produced enhanced reference letters of the Avila Bible and binarized pages for both the Avila and Trento Bibles. Once those data were obtained, it was possible to extract letter occurrences from the Avila text to arrange a dataset. We used the reference letter of each author from the Avila manuscript to systematically extract occurrences of the letter “a” from the pages of the respective writer in the Avila Bible. Regarding extracting the letters from the Trento Bible, we didn’t have references, so our focus was solely on the reference letter of the scribe who contributed to the writing process of both bibles, namely scribe F. This strategic choice was informed by the guidance of paleographers, who noted that author *F* was the only common writer across both ancient texts (see Sect. 2).

Hence, we tested various template-matching algorithms to identify every instance of the reference letter “a” attributed to each author within their respective pages. For this purpose, we opted for a method provided by the OpenCV library: the normalized cross-correlation template matching function [26]. This function is specifically designed to detect instances of a template image, in this

case, the reference letter, within a larger image representing the page. It operates by sliding the template image over the larger image and computing a similarity measure at each position. This similarity measure indicates how closely the template matches the corresponding region of the larger image. We considered a threshold for the similarity scores of the detected occurrences, which enabled us to extract reliable samples and avoid misleading occurrences. Once regions of interest (ROIs) containing the letter occurrences with similarity scores surpassing the threshold were identified, we extracted and saved them in PNG format.

The next step involved image cleaning of the extracted occurrences, as they were taken from text pages, and traces of adjacent letters were present. Therefore, we performed the same cleaning process described in Sect. 3.1, considering thinning and findContours operations. The extracted images had different sizes, so we uniformed them in a standard 224×224 shape. This process guaranteed the precise extraction of the desired letters and convenient postprocessing, thus laying a solid foundation for subsequent analysis and study.

3.2 Deep Learning Approach

This section outlines the DL approaches utilized in this study. In the first approach, we conducted a multiclass classification on the Avila dataset, each class representing a different writer. Conversely, the second approach involved training a model on the Avila dataset for binary classification, distinguishing *F* from all other classes. Subsequently, we applied this model to the Trento dataset to assess its ability to identify shared writers across the two bibles.

Multiclass Classification on Avila Bible. The initial DL approach involved a multiclass classification, intending to identify the writers of the Avila Bible based on their handwriting styles. Figure 4 depicts each implemented step. The Avila dataset consists of images depicting occurrences of the letter “a” from eight distinct writers. These images were generated through the methodology outlined in Sect. 3.1 and subsequently categorized according to the respective writers. The Avila dataset was used to feed a CNN; further details about the training and testing processes are described in Sect. 4. The output of the CNN was a prediction for every occurrence sample in the dataset. Ultimately, the outcomes were synthesized by utilizing a majority voting rule to consolidate the results and obtain a classification at the page level. This comprehensive approach facilitated the accurate identification of writers and underscored the significance of model selection, cross-validation, and metric evaluation in enhancing classification outcomes.

Inference Approach to Trace the Writer. This section describes the developed DL approach to trace the writer across the two Bibles. The process is shown in Fig. 5, which takes as input the images generated through the image processing step outlined in Sect. 3.1.

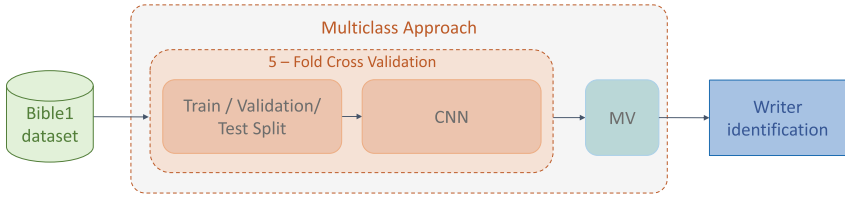


Fig. 4. The proposed Multiclass approach on Avila Bible

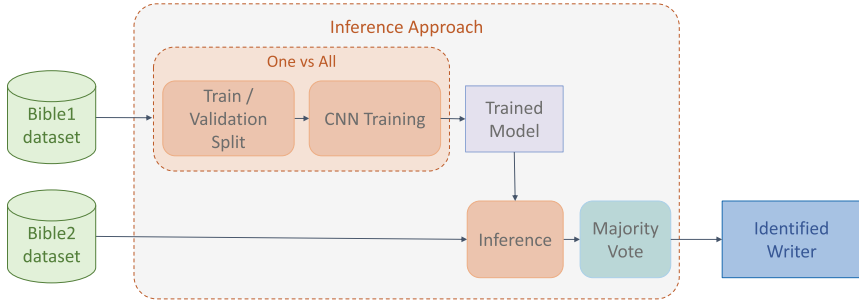


Fig. 5. The proposed Inference approach to identify the common writer

The Avila dataset consisted of images depicting occurrences of the letter “a” attributed to each of the eight writers. Since we had prior knowledge of the common writer between the two bibles, for this experiment, we structured the dataset into two distinct classes: *F*, comprising solely letters from the *F* writer, and *All*, encompassing letters from the remaining writers in the collection. Using the One vs. All approach, we trained two CNN models specifically tailored to differentiate the writing style of the *F* writer from the others. Subsequently, these models were applied to infer insights from the Trento dataset. The Trento dataset, comprising two classes—*F* and *All*—served as the test set, resulting in predictions for each sample indicating whether it belonged to the *F* writer. Recognizing that a single individual wrote each page of the Trento Bible, we enhanced our evaluation by employing a majority voting rule to aggregate the predictions of occurrences associated with each page.

The culmination of the entire system provided an answer for each page of the Trento Bible, providing insight into whether the writer *F* of the Avila Bible had authored any pages within the Trento Bible or not.

4 Experimental Results

This section focuses on deploying and analyzing the outcomes derived from the image processing procedure described in Sect. 3.1 and the experimental workflow outlined in Sect. 3.2.

Table 1. Details about the image extraction process of the letter “a” from Avila Bible.

	A	B	D	E	F	G	H	I	Total
#occurrences	2933	166	378	565	1317	269	425	52	6105
#total pages	309	31	18	76	150	31	37	53	705
#rejected pages	65	7	2	8	11	24	2	25	144
#occurrences/page	12.02	6.91	23.62	7.43	8.78	38.42	12.14	1.85	13.89

Table 1 presents comprehensive details regarding the extraction process of occurrences of the letter “a” from the Avila Bible. Each column corresponds to a writer, and the last column shows the sum of the previous columns. The first row displays each writer’s extracted occurrences and the final aggregate measure. Following this, the second row provides the number of pages attributed to each writer, while the third row delineates the count of pages rejected from our evaluation due to deterioration, impeding letter extraction. Finally, the last row presents the average number of extracted occurrences per page. This table highlights a significant imbalance among the various classes, stemming from several factors. Firstly, out of the eight authors, only two have written more than a hundred pages, resulting in higher occurrences of the chosen character. Furthermore, this imbalance is influenced by the fact that many classes exhibit a very high percentage of rejected pages and, simultaneously, a low number of occurrences per page. This suggests that certain authors may have contributed disproportionately to the dataset, potentially skewing the distribution of occurrences across classes.

The extraction process of occurrences of the letter “a” from the Trento manuscript, using the reference letter attributed to the writer *F* from the Avila Bible, is illustrated in Table 2. The row organization mirrors that of the previous table. Regarding the Trento Bible, our dataset comprises two classes: one corresponding to the *F* writer and the other labelled *All*, encompassing the remaining authors. Each column in this table pertains to a specific class within the problem, with the final column representing the aggregation of the first two classes. This table is very interesting as it shows the numeric imbalance of the problem since *F* writer wrote most of the pages of the Trento Bible. The reason for this imbalance is not only due to the higher number of pages written by writer *F* but also related to the fact that we used the reference letter of writer *F* to extract occurrences from pages not belonging to that writer. The poor or even the absence of occurrences of the character among the pages belonging to the *All* class, resulting in a small number of occurrences per page and the rejection of 48 pages, is a positive aspect. This means that the method used for template matching works effectively and serves as a first-level filter for identifying the shared writer. Indeed, only pages belonging to class *All* have been rejected, as no occurrence was found, while no page related to class *F* has been rejected.

After completing the image processing step, we utilized the DL methodologies detailed in Sect. 3.2 to carry out two experiments: a multiclass experiment using the Avila dataset, and an experiment focusing on the Trento dataset aimed

at identifying the shared writer. Subsequently, various metrics, expressed as percentages, were calculated to assess the performance of the experiments relative to their respective problem domains.

Concerning the multiclass evaluation, we adopted a 5-fold cross-validation technique to enhance our analysis’s robustness and mitigate overfitting. Each fold encompassed 20% of the dataset and served as the test set, while the remaining 80% was split between the training set (70%) and the validation set (10%). We ensured that occurrences from the same page remained within the same set during the partitioning process. Following dataset preparation, we employed deep neural networks for training and testing. Several CNN models were assessed, with InceptionV3 [30] emerging as the optimal choice based on performance. The number of parameters, the depth and input/output size of the CNN are shown in Table 3. Once the CNN architecture was chosen, its evaluation through an experimental phase was necessary to maximize the mean accuracy. This entails selecting specific hyper-parameters and settings, such as employing Stochastic Gradient Descent (SGD) with a learning rate of 0.001 and momentum of 0.9 as the optimization method to minimize the loss function. Additionally, we adopted the categorical cross-entropy as the loss function, defining a maximum of 100 epochs and setting a patience level of 2 epochs wherein training halts if validation accuracy fails to improve, and finally, utilizing accuracy as the performance metric. The training of the model was conducted through a two-step procedure. First, transfer learning was applied using a pre-trained model with the ImageNet dataset [13], and then we fine-tuned the model with the Avila dataset.

The experiment’s efficacy was gauged using diverse metrics averaged across the five folds to provide a comprehensive evaluation.

Table 4 presents the outcomes of the multiclass experiment conducted using the Avila dataset from the point of view of extracted occurrences, namely assuming as pattern to be classified the single letter occurrences. Here, the results are expressed in terms of recognition rate per class and displayed in the first eight columns. The ninth column illustrates the overall accuracy achieved for the multiclass problem. All the metrics reported were averaged across the five folds. The results shown in the table are exceptionally high. Each writer’s accuracy in letter occurrences, depicted in the first eight columns, demonstrates near-perfect performance, with values ranging from 99.75% to 100%. The ninth column shows the overall accuracy for the multiclass problem, averaging an impressive 99.96%

Table 2. Details about the image extraction process from Trento Bible, using reference letters of *F* writer from Avila.

	F	All	Total
#occurrences	43038	4554	47592
#total pages	249	145	394
#rejected pages	0	48	48
#occurrences/page	172.84	46.94	137.54

Table 3. Number of parameters, depth and input/output size of the CNNs exploited in the experiment.

Model	#Parameters	Depth	Input Size	Output Size
InceptionV3	23.9M	189	299 × 299	2048
EfficientNetB2	9.2M	186	260 × 260	1408

Table 4. Accuracy Results for the Multiclass Experiment on the Avila extracted letters.

A	B	D	E	F	G	H	I	ACC
99.96	100	100	100	100	100	99.75	100	99.96

across the five folds. Obviously, these results don’t consider all the letter occurrences possibly present in the rejected pages.

Table 5 reports the same metrics and problem as the previous table, but in this case, everything was computed on the page level, so after the application of the majority voting. Additionally, rejected pages were considered in this evaluation, which caused a decrement in the performance. The first row reports the recognition rate for each class, and the final column shows the overall accuracy. The second row provides information on the rejection rate of pages. The table shows how the recognition rate improves when the rejection rate decreases, independently of the total number of occurrences extracted for each writer. This result is the direct consequence that concerning the accepted pages, where at least one occurrence was found, the letter recognition rate is close to 100%. Considering the complexity of the problem and the number of rejected pages, these results are indeed very encouraging, showing a majority vote accuracy of 79.57%.

Concerning the second experiment, the Avila dataset was partitioned into two classes, *F* and *All*, and it was split into training and validation sets to train a CNN using the One vs All approach. As previously described, we selected the same model, InceptionV3, and hyper-parameters setting of the multiclass experiment. Moreover for this experiment, we tested a second CNN model, EfficientNetB2 [31]. Table 3 reports details about the two models. InceptionV3 has more parameters and a larger input size, whereas EfficientNetB2 is more compact with fewer parameters and a smaller input size. EfficientNetB2 is also the more

Table 5. Majority Vote Results for the Multiclass Experiment on the Avila dataset considering rejected pages.

	A	B	D	E	F	G	H	I	ACC
Recognition Rate	78.96	77.41	88.88	89.47	92.66	22.58	94.59	52.83	79.57
Rejection Rate	21.04	22.59	11.12	10.53	7.34	77.42	5.41	47.17	20.43

Table 6. Results for the F vs All experiment on the Trento Bible.

Model	Level	ACC	TPR	PPV
InceptionV3	Occurrence	68.70	70.45	93.29
	Page (MV)	84.39	93.17	86.24
EfficientNetB2	Occurrence	69.70	72.74	92.14
	Page (MV)	84.12	97.43	84.77

recent model, designed to achieve better performance with optimized efficiency compared to earlier models like InceptionV3. The output of this procedure was the InceptionV3 and the EfficientNetB2 models trained on the Avila dataset, ready to receive new inputs and recognize if a sample belongs to the F writer. After the models underwent training using the Avila dataset, they were preserved and deployed for inference on the Trento dataset. In this scenario, the Trento dataset, consisting of occurrences of the letter “a”, was assumed as the test set and used to evaluate the method’s performance. This decision was motivated by our interest in assessing whether the model trained on the Avila dataset could accurately discern samples from the Trento Bible attributed to writer F .

The outcome of this phase yielded a prediction for every letter sample of the Trento Bible. After acquiring these predictions, we computed various metrics to assess the system’s performance at the letter occurrence and page level, as shown in Table 6. For every model (first column) and classification level (second column), the third column reports the overall accuracy metric, while the following columns refer solely to the positive class, F , showing the sensitivity - True Positive Rate (TPR) - and the precision - Positive Predict Value (PPV). Sensitivity measures the proportion of actual positive cases correctly identified by the model. Precision quantifies the proportion of positive cases identified by the model that are truly positive. In essence, sensitivity assesses the model’s ability to capture all positive instances, while precision evaluates its accuracy in labelling instances as positive.

These results are notable for several reasons. For every CNN model, the first row shows that it performs very well for the F class, correctly identifying approximately 70% of occurrences with high sensitivity and precision. The second row presents the metrics computed for the same problem, this time applying the majority vote rule at the page level. Notably, the results demonstrate a significant improvement over the previous case, showcasing enhanced classification performance with the aggregation of predictions at the page level. Table 6 allows for a comparison between the two models tested: InceptionV3 and EfficientNetB2. Although the overall accuracy of both models is quite similar, other metrics reveal distinct differences in performance. EfficientNetB2 performs superior in recognizing samples from the F writer, as evidenced by a higher TPR. Conversely, it performs less in recognizing samples from the All class, as indicated by a lower PPV.

5 Conclusions and Future Work

Palaeographers study ancient documents from different historical periods to decipher, date, and interpret their content. In this framework, identifying the scribes who collaborated to transcribe a single medieval book is of great interest.

In previous studies, we proposed some pattern recognition systems for scribe distinction, using basic layout features regarding the organization of the page and its exploitation by the scribe. We considered the case of highly standardized handwriting and book typologies, where such features are highly distinctive. Even if the results were very interesting, the above features didn’t allow the system to characterize the peculiarities of the writing style of each scribe, making it very difficult to recognize the hand of a scribe who worked on multiple ancient texts produced following different standards.

This paper presents preliminary results from a novel approach to overcoming the abovementioned limitations. The rationale is to exploit the knowledge of palaeographers who have identified, for each scribe, some letters or abbreviations that characterize them. According to their suggestions, we considered the letter “a” and used a template matching technique to identify the occurrences of this letter on each page of the considered manuscripts. Then, a Convolutional Neural Network (CNN) was used to train a classification system capable of attributing each character “a” occurrence to the corresponding scribe. Finally, we used a majority voting technique to assign the entire manuscript page to the scribe with the highest number of occurrences of the character “a” on that page.

The experimental results obtained by applying our approach to the Avila Bible were very interesting, improving the performance previously obtained with layout features. Furthermore, using data relating to the Avila Bible, we trained two CNN models to recognize the writing of a scribe whose hand was also identified by paleographers in the Trento Bible: the CNNs obtained in this way were able to identify with good reliability the pages written by this scribe in the Trento Bible without using any prior knowledge of this bible.

Based on the preliminary but encouraging results presented here, we will focus our future work on investigating two aspects. First, we will explore improving our convolution-based approach to extract the reference images. To this aim, we will test several techniques to merge the single letters extracted by palaeographers and test other convolutional filters. Second, we will validate the effectiveness of the proposed approach by including more ancient Bibles.

References


1. BIBLIOTECA DIGITAL HISPÁNICA, BIBLIOTECA NACIONAL DE ESPAÑA. <https://bdh.bne.es/bnearch/detalle/bdh0000014221>
2. Afzal, M.Z., et al.: Deepdocclassifier: document classification with deep convolutional neural network. In: 2015 13th International Conference on Document Analysis and Recognition (ICDAR), pp. 1111–1115 (2015)
3. Aguilar, S.T., Jolivet, V.: Handwritten text recognition for documentary medieval manuscripts. *J. Data Min. Digit. Humanit.* (2023). <https://api.semanticscholar.org/CorpusID:266514075>

4. Antonacopoulos, A., Downton, A.C.: Special issue on the analysis of historical documents. *IJDAR* **9**(2–4), 75–77 (2007)
5. Bulacu, M., Schomaker, L.: Text-independent writer identification and verification using textural and allographic features. *IEEE Trans. Pattern Anal. Mach. Intell.* **29**(4), 701–717 (2007)
6. Cilia, N., De Stefano, C., Fontanella, F., Marrocco, C., Molinara, M., Scotto di Freca, A.: An end-to-end deep learning system for medieval writer identification. *Pattern Recognit. Lett.* **129**, 137–143 (2020)
7. Cilia, N.D., De Stefano, C., Fontanella, F., Molinara, M., Scotto di Freca, A.: What is the minimum training data size to reliably identify writers in medieval manuscripts? *Pattern Recognit. Lett.* **129**, 198–204 (2020)
8. Cilia, N.D., De Stefano, C., Fontanella, F., Marrocco, C., Molinara, M., Scotto di Freca, A.: An experimental comparison between deep learning and classical machine learning approaches for writer identification in medieval documents. *J. Imaging* **6**(9) (2020)
9. Dahllorf, M.: Scribe attribution for early medieval handwriting by means of letter extraction and classification and a voting procedure for larger pieces. In: *Proceedings of the 22nd International Conference on Pattern Recognition*, pp. 1910–1915. IEEE Computer Society (2014)
10. De Stefano, C., Maniaci, M., Fontanella, F., Scotto di Freca, A.: Layout measures for writer identification in mediaeval documents. *Measurement* **127**, 443–452 (2018)
11. De Stefano, C., Maniaci, M., Fontanella, F., Scotto di Freca, A.: Reliable writer identification in medieval manuscripts through page layout features: the avila bible case. *Eng. Appl. Artif. Intell.* **72**, 99–110 (2018)
12. De Stefano, C., Fontanella, F., Maniaci, M., Scotto di Freca, A.: A method for scribe distinction in medieval manuscripts using page layout features. In: Maino, G., Foresti, G.L. (eds.) *ICIAP 2011. LNCS*, vol. 6978, pp. 393–402. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-24085-0_41
13. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: a large-scale hierarchical image database. In: *CVPR*, pp. 248–255. IEEE Computer Society (2009)
14. Dhali, M.A., He, S., Popovic, M., Tigchelaar, E., Schomaker, L.: A digital palaeographic approach towards writer identification in the dead sea scrolls. In: *Proceedings of the 6th International Conference on Pattern Recognition Applications and Methods, ICPRAM*, pp. 693–702 (2017)
15. Fagioli, A., Avola, D., Cinque, L., Colombi, E., Foresti, G.L.: Writer identification in historical handwritten documents: a Latin dataset and a benchmark. In: Foresti, G.L., Fusiello, A., Hancock, E. (eds.) *Image Analysis and Processing - ICIAP 2023 Workshops. ICIAP 2023. LNCS*, vol. 14366, pp. 465–476. Springer, Cham (2024). https://doi.org/10.1007/978-3-031-51026-7_3
16. Gattal, A., Djeddi, C., Abbas, F., Siddiqi, I., Bouderah, B.: A new method for writer identification based on historical documents. *J. Intell. Syst.* **32**(1), 20220244 (2023)
17. Grieggs, S., Henderson, C.E.M., Sobecki, S., Gillespie, A., Scheirer, W.: The paleographer’s eye ex machina: using computer vision to assist humanists in scribal hand identification. In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)* pp. 7177–7186, January 2024
18. He, S., Samara, P., Burgers, J., Schomaker, L.: Image-based historical manuscript dating using contour and stroke fragments. *Pattern Recogn.* **58**, 159–171 (2016)

19. Lastilla, L., Ammirati, S., Firmani, D., Komodakis, N., Merialdo, P., Scardapane, S.: Self-supervised learning for medieval handwriting identification: a case study from the vatican apostolic library. *Inf. Process. Manag.* **59**(3), 102875 (2022). <https://doi.org/10.1016/J.IPM.2022.102875>
20. Lombardi, F., Marinai, S.: Deep learning for historical document analysis and recognition—a survey. *J. Imaging* **6**(10) (2020)
21. Maniaci, M., Ornato, G.: Prime considerazioni sulla genesi e la storia della bibbia di avila. In: *Miscellanea F. Magistrale* (2010)
22. Omayio, E.O., Indu, S., Panda, J.: Historical manuscript dating: traditional and current trends. *Multimed. Tools Appl.* **81**(22), 31573–31602 (2022). <https://doi.org/10.1007/s11042-022-12927-8>
23. Papaodysseus, C., et al.: Identifying the writer of ancient inscriptions and byzantine codices. a novel approach. *Comput. Vis. Image Underst.* **121**, 57–73 (2014)
24. Peer, M., Kleber, F., Sablatnig, R.: Towards writer retrieval for historical datasets (2023). <https://arxiv.org/abs/2305.05358>
25. Rahal, N., Vögtlin, L., Ingold, R.: Historical document image analysis using controlled data for pre-training. *Int. J. Doc. Anal. Recognit.* **26**(3), 241–254 (2023). <https://doi.org/10.1007/s10032-023-00437-8>
26. Sarvaiya, J., Patnaik, S., Bombaywala, S.: Image registration by template matching using normalized cross-correlation. In: *2009 International Conference on Advances in Computing, Control, and Telecommunication Technologies*, pp. 819–822 (2009). <https://doi.org/10.1109/ACT.2009.207>
27. Stokes, P.: *Computer-Aided Palaeography*, Present and Future, pp. 309–338. Institut für Dokumentologie und Editorik (2009)
28. Stokes, P.A.: Digital approaches to paleography and book history: some challenges, present and future. *Front. Digit. Humanit.* **2**, 5 (2015)
29. Suzuki, S., be, K.: Topological structural analysis of digitized binary images by border following. *Comput. Vis. Graph. Image Process.* **30**(1), 32–46 (1985). [https://doi.org/10.1016/0734-189X\(85\)90016-7](https://doi.org/10.1016/0734-189X(85)90016-7)
30. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2818–2826 (2016)
31. Tan, M., Le, Q.: Efficientnet: rethinking model scaling for convolutional neural networks. In: *International Conference on Machine Learning*, pp. 6105–6114. PMLR (2019)



Cyclic Learning of a Frame Downsampler and a Recognition Model in High-Speed Camera Image Recognition

Shigeaki Namiki^(✉), Takuya Ogawa, Keiko Yokoyama, Shoji Yachida, and Toshinori Hosoi

NEC Corporation, Kanagawa, Japan

{s-namiki, takuya_ogawa, k.yokoyama, s-yachida, t.hosoi}@nec.com

Abstract. High-speed cameras capture instantaneous changes in dynamic phenomena, enabling new image classification applications such as nonstop visual inspection of dynamically moving objects. However, due to high-speed cameras' high frame rate, downsampling excess frames or limiting the size of the recognition model is necessary for real-time processing. Previous work has introduced a frame downsampler (a binary classifier optimized to predict in advance whether the output of a recognition model is true or false) and applied the downsampler to retain high-scoring frames for the recognition model. However, further optimization of the recognition model for the sampled data distribution is unexplored and remains sub-optimal. In this study, we propose “cyclic learning” for high-speed camera image recognition. It optimizes the recognition model for the data distribution left by the downsampler and retrains the downsampler based on the updated recognition model. We constructed a dataset of fast-moving objects captured by a high-speed camera to classify object types, and experimental results on this dataset proved that the proposed method outperformed previous studies regarding overall classification performance with the same number of samples and the number of samples required for comparable classification performance.

Keywords: High-speed camera · Image classification · Optimizations

1 Introduction

High-speed cameras can capture time-dense images at very high frame rates, allowing them to capture the appearance of instantaneous features and their changes, such as tracking fast-moving objects or classifying differences in the surface textures of such objects. It is, therefore, used in fields where high-speed imaging of events is required. For example, it is beginning to be used for industrial product manufacturing sites [11].

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-78447-7_7.

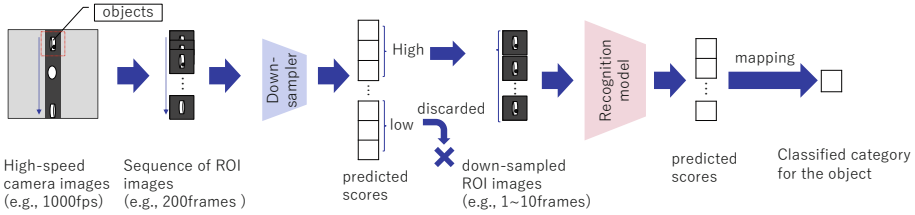
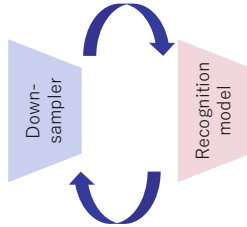


Fig. 1. A inference flow of the high-speed camera image recognition

WCE with predicted Recognition model's correctness



KD with Recognition model's correctness labels

Fig. 2. A training flow of the proposed cyclic learning for the high-speed camera image recognition. WCE means Weighted Cross Entropy, and KD means Knowledge Distillation.

For image processing of high-speed camera images, the characteristic of low frame-to-frame variation of the image makes it possible to perform object tracking with simple and lightweight processing. Based on this, various applications are being researched, for example, controlling high-speed robot hands and manipulating high-speed moving objects [14]. They use lightweight image processing to acquire an object’s position and orientation motion.

On the other hand, for object recognition, there are examples of gesture or posture recognition based on motion information [7, 12] as extensions of object tracking. However, only a few studies have explored the classification of object appearance.

Due to the high image frame rate, the problem of object classification from high-speed camera images requires limiting the size of the recognition model or downsampling the extra frames for real-time processing.

Especially in order to apply a deep model as a recognition model, there are two directions of existing research: direction to lighten the deep model [1, 2, 6] and frame selection direction [9].

Although various architectures and model weight reduction methods have been proposed for the former, the size of recent large-scale models such as ViT [3]

is increasing due to their characteristic of scaling for performance. Therefore, the latter direction of choosing frames remains useful.

The previous study [9] is the only example in this direction. It assumes high-speed camera frames contain features either visible for object classification or too ambiguous. They propose a downsampler to retain clear frames and exclude ambiguous ones, implemented with a binary classifier predicting the recognition model’s output accuracy. However, the recognition model remains sub-optimal due to the downsampler’s data distribution differing from the training set. Further optimizing the recognition model for this sampled data distribution is conceivable but remains unexplored.

We propose a ‘cyclic learning’ method for fast camera image recognition. It optimises the recognition model for the data distribution left by the down sampler and re-trains the down sampler based on the updated recognition model.

We also created a dataset of fast-moving objects captured by a high-speed camera and set up a task to classify object types (normal vs. anomalous). In this task, the features representing anomalies appear only fleetingly so that the downsampler cannot miss these instantaneous frames. In this sense, this task is more complicated than the classification task in previous studies (object type unchanged in any frame). This task allows a better assessment of the performance differences of the downsampler.

Experimental results on this dataset prove that the proposed method outperforms previous studies concerning the overall classification performance at the same number of left samples and the number of samples required for comparable classification performance.

Our main contributions are:

- Proposed to optimise the recognition model not for all data but for the sampled data of a high-speed camera.
- Proposed to train the downsampler and recognition model alternately. Improves performance compared to training each only once.
- Experiments show the effectiveness of cyclic learning on multiple types of data and neural network architectures.

2 Related Works

2.1 High-Speed Camera Image Recognition

[9] is the only previous study with a high-speed camera image classification and introduces a downsampler. Assuming that there are frames in a high-speed camera image of an object in which the features necessary for object classification are visible and frames in which the features are ambiguous and difficult to classify, they propose a downsampler that retains the former and excludes the latter. The downsampler was implemented with a binary classifier optimised to predict in advance whether the output of the recognition model would be true or false. However, the recognition model was sub-optimal because the data distribution output by the downsampler varied from the distribution on which the recognition

model was trained. Therefore, it is conceivable to optimise the recognition model further for this sampled data distribution for further performance improvement, but this has yet to be explored in existing research. We are doing this for the first time.

2.2 Easy Example Mining

[8] proposed online easy example mining to learn credible supervision signals instead of noisy pseudo labels from the weakly supervised learning setting. They assumed the confidence score as a metric to indicate the learning difficulty and proposed the cross entropy loss weighted with the metric calculated using the confidence score. Inspired by this, we assume the frames with high confidence scores output by the recognition model are easy examples in high-speed camera image recognition settings. As mentioned before, we are thinking about optimising the recognition model to fit the data distribution sampled by the downsampler, so the easy example of a high confidence score from the recognition model will be changed after the re-training. This relationship means that the downsampler can be further optimised for the output data distribution from the recognition model. We focus on this point. Note that [8] could perform easy example mining online because they handled loss-weighted easy examples based on the confidence of the inference results. However, in our problem set-up, the easy example is calculated from the confidence of the inference results of the recognition model, and the downsampler learns the easy example, so it is more like alternating learning than online learning. Cycling through this alternating optimisation should improve performance.

2.3 Cyclic Learning

There are some examples of cyclically optimising multiple neural networks. Cycle GAN [16] is a method for training neural networks that transform images from one domain to another. They train an image-transforming neural network from the source domain to the target domain and another image-transforming neural network from the target domain to the source domain. This method minimises the loss between the source images and transformed source-to-target-to-source images via the two networks. The information shared between the two neural networks in this example is the image data. Also [15] propose “cyclic learning” in the context of weakly supervised learning. Instead of using costly segmentation masks, they propose to utilise relatively low-cost whole-image labels. After supervised neural network training to infer labels for the whole image, a pseudo-segmentation mask is created with CAM visualising the neural network’s points of interest during inference. The encoder weights are copied and used as a backbone of another neural network that infers the segmentation mask, which is trained using that pseudo-segmentation mask. In this example, the information shared between the two neural networks is the pseudo masks and the encoder weights.

We first introduce a cyclic learning framework for high-speed camera image recognition in different information-sharing manner from the above two studies. In our framework, the information shared between the two neural networks is the confidence score, which can be seen as the degree of the easy examples. The downsampler is trained with the confidence score of the recognition model’s prediction correctness in a knowledge distillation manner instead of directly sharing weights between the downsampler and the recognition model. This is because we assume the downsampler is more lightweight, and thus, its architecture is different from the recognition model, so we cannot share the weights directly. Next, the trained downsampler infers the confidence score of the recognition model’s prediction correctness and weighs the samples with its score in the training loss of the recognition model. After that, the downsampler is trained again with the altered confidence score of the recognition model. In this order, the downsampler and the recognition model are cyclically trained until the validation accuracy peaks.

Experimental results on this dataset prove that the proposed method outperforms previous studies concerning the overall classification performance at the same number of left samples and the number of samples required for comparable classification performance.

3 Our Framework

3.1 Pipeline of the Framework

We follow the inference framework of high-speed camera image recognition in [9] as shown in Fig. 1. It can efficiently handle massive amounts of incoming temporally dense images captured by a high-speed camera.

The input to the system is a sequence of images of an object captured by a high-speed camera, and the output is the classification result of the object with C classes. The system has two main components: a downsampler and a recognition model. The downsampler is a lightweight binary classifier that predicts whether the output prediction of the recognition model is true or false. The recognition model is a much deeper neural network than the downsampler that classifies the input image into one of C classes.

Following [9], we detect and track objects in the camera’s field of view in real-time using the same lightweight processing as [9] and obtain a sequence of ROI images of the object in real-time. We denote the i -th sequence of ROI images as S_i :

$$S_i = \{x_{i,1}, x_{i,2}, \dots, x_{i,N_i}\}, \quad (1)$$

where $x_{i,j}$ is the j -th image in the i -th sequence and N_i is the number of images in the i -th sequence.

The sequence of image-wise ground-truth labels for S_i is denoted as S_i^y :

$$S_i^y = \{y_{i,1}, y_{i,2}, \dots, y_{i,N_i}\}, \quad (2)$$

where $y_{i,j}$ is the true value of the j -th image in the i -th sequence. If the image recognition task is to classify categories that remain the same regardless of the frame-by-frame visibility, the labels do not change within the sequence. In other words, the ground truth label $y_{i,j}$ is the same for all j . In contrast, the labels change within the sequence if the task is to classify categories that vary according to how each frame looks. In other words, the frame-wise ground-truth label $y_{i,j}$ differs for each j . An example of the former is a task that classifies the type of object itself, while an example of the latter is a task that classifies whether an object has an abnormality. Depending on the type of task, one sequence-wise label Y_i is calculated from S_i^y and assigned to the input sequence.

We then input each ROI image in the sequence S_i to the downsampler $f(x; \theta)$, which is a binary classifier learned to predict whether the output of the recognition model $g(x; \phi)$ is true or false and output the probability of true. We then get the output sequence $f(S_i)$:

$$f(S_i) = \{f(x_{i,j}; \theta) | j = 1, \dots, N_i\}. \quad (3)$$

In the following, θ and ϕ are omitted for simplicity.

The output of f is taken as the score, and the top K images in the score are selected to form a new sequence S_i^f :

$$S_i^f = \{x_{i,\tilde{j}} | \tilde{j} = 1, \dots, K; f(x_{i,\tilde{j}}) > f(x_{i,\tilde{j}-1})\} \subset S_i, \quad (4)$$

where \tilde{j} is the index of the selected image in the sequence S_i .

We then pass S_i^f to the recognition model g and get the output sequence $g(S_i^f)$.

$$g(S_i^f) = \{g(x_{i,\tilde{j}}) | \tilde{j} = 1, \dots, K\}. \quad (5)$$

In order to get the final classification result for the i -th sequence S_i , a single category needs to be assigned depending on the task type; we map the output sequence $g(S_i^f)$ to the final class-wise score H_i . This mapping function can be arbitrary, depending on the task.

For example, we can use the average of $g(S_i^f)$:

$$H_i = \frac{1}{K} \sum_{x_{i,\tilde{j}} \in S_i^f} g(x_{i,\tilde{j}}). \quad (6)$$

or, we can use the maximum of $g(S_i^f)$:

$$H_i = \max_{x_{i,\tilde{j}} \in S_i^f} g(x_{i,\tilde{j}}). \quad (7)$$

Finally, we classify the i -th sequence as the category of interest Y_i :

$$\hat{Y}_i = \underset{c}{\operatorname{argmax}} H_i. \quad (8)$$

where \hat{Y}_i is the final classification result of the i -th sequence.

The training flow are shown in Fig. 2 and the details are described in the following subsections.

3.2 Learning the Downsampler

We use some neural network models for the downsampler f , such as ResNet or MobileNet. Previous work [9] used primitive image features and SVM or LDA as a classifier. We relax this and use a deep neural network where the feature extraction and classifier can be trained arbitrarily. While previous studies did not require perfect inference performance for f since the purpose was cleansing the data, we need high inference performance to avoid learning stalls as we alternate between optimising the downsampler and the recognition model.

We first assign the training data with the new labels to train the downsampler, indicating whether the recognition model is correct or incorrect for the given ROI image. The training of the recognition model is described in the following subsection, but let us assume that the base pre-trained recognition model is already obtained and that the true value $y_{i,j}$ and the output of the recognition model is obtained for the input ROI image $x_{i,j}$. If the true value $y_{i,j}$ and the output of the recognition model $\hat{y}_{i,j}$ match, then the new label $y_{T/F,i,j}$ is assigned to the input ROI image $x_{i,j}$ as 1; otherwise it is assigned as 0. So the new sequence of labels $S_i^{T/F}$ for

$$S_i^{T/F} = \{y_{T/F,i,1}, y_{T/F,i,2}, \dots, y_{T/F,i,N_i}\}, \quad (9)$$

where $y_{T/F,i,j}$ is the new label of the j -th image in the i -th sequence.

Next, We assume a binary classifier for the downsampler, so we adopt the binary cross entropy loss (L_{BCE}):

$$L_{BCE} = -\frac{1}{N_{total}} \sum_{i,j}^{N_{total}} y_{T/F,i,j} \log(p_{y_{T/F}}(x_{i,j})), \quad (10)$$

where $p_{y_{T/F}}(x_{i,j})$ is the probability, the softmax of the output of the downsampler $f(x_{i,j})$ for the true value $y_{T/F,i,j} = 1$, and N_{total} is the number of images in training data.

Furthermore, we also perform knowledge distillation learning [4], considering that the correct or incorrect inference results of a recognition model of a larger model size are inferred by a downsampler of a smaller model size. Note that the recognition model g outputs are the logits of the C -class classification problem. So we define the softmax with temperature proposed in [4] of the recognition model $q_{i,j}$ as:

$$q_{i,j,c=c_{y_T}} = \frac{\exp(\frac{g(x_{i,j})_c}{T})}{\sum_{c'=1}^C \exp(\frac{g(x_{i,j})_{c'}}{T})}, \quad (11)$$

$$q_{i,j,c=c_{y_F}} = 1 - q_{i,j,c=c_{y_T}},$$

where c_{y_T} is the class of the true value $y_{T/F}$, C is the number of classes, c_{y_F} is the class of the false value $y_{T/F}$, $g(x_{i,j})_c$ is the c -th element of the output of the recognition model $g(x_{i,j})$, and T is a hyperparameter denoting temperature.

the softmax with temperature of the downsampler $p_{i,j}$ is:

$$p_{i,j,c=c_{y_T}} = \frac{\exp(\frac{f(x_{i,j})_c}{T})}{\sum_{c'=1}^C \exp(\frac{f(x_{i,j})_{c'}}{T})}, \quad (12)$$

$$p_{i,j,c=c_{y_F}} = 1 - p_{i,j,c=c_{y_T}}.$$

We then introduce knowledge distillation loss L_{KD} with KL-Divergence:

$$L_{KL} = \frac{1}{N_{total}} \sum_{i=1}^{N_{total}} T^2 \sum_{c=1}^C q_{i,j,c=c_{y_T}} \log\left(\frac{q_{i,j,c=c_{y_T}}}{p_{i,j,c=c_{y_T}}}\right), \quad (13)$$

where C is the number of classes, $g(x_{i,j})_c$ is the c -th element of the output of the recognition model $g(x_{i,j})$, and T is a hyperparameter denoting temperature.

we add the two losses to obtain the total loss L_{DS_KD} with a weight hyperparameter α :

$$L_{DS_KD} = \alpha \cdot L_{BCE} + (1 - \alpha) \cdot L_{KL}. \quad (14)$$

We can train the downsampler by minimising this loss.

The trained downsampler will produce sample weights $w_{i,j}$ for the recognition model training in the next step:

$$w_{i,j} = Sigmoid(f(x_{i,j})). \quad (15)$$

So we can get S_i^w :

$$S_i^w = \{w_{i,1}, w_{i,2}, \dots, w_{i,N_i}\}. \quad (16)$$

3.3 Learning the Recognition Model

We also use neural network models for the recognition model g , such as ResNet or MobileNet, like in the downsampler, but the difference is the model size. The downsampler solves the binary classification problem of correct/incorrect. In contrast, the recognition model generally solves the more difficult classification problem of inferring the categories of a given ROI image. Therefore, a larger model size, such as ResNet or MobileNet models with more layers, is required.

To train the recognition model, we use the sample weights $w_{i,j}$ output by the downsampler and use them as the weight for the weighted cross entropy loss denoted as L_{Rec_WCE} :

$$L_{Rec_WCE} = -\frac{1}{N_{total}} \sum_{i=1}^{N_{total}} \sum_{j=1}^{N_i} w_{i,j} \cdot y_{i,j} \log(g(x_{i,j})). \quad (17)$$

For training the recognition model as the base pretrain model used in the initial training of the downsampler described in the previous subsection, the initial sample weights $w_{i,j}$ are set to 1.0 for all i and j . On the other hand, to train the recognition model to fit the data distribution produced with outputs by the downsampler, the sample weights $w_{i,j}$ are set to weights output by the downsampler for all i and j .

3.4 Cyclic Learning of the Downsampler and the Recognition Model

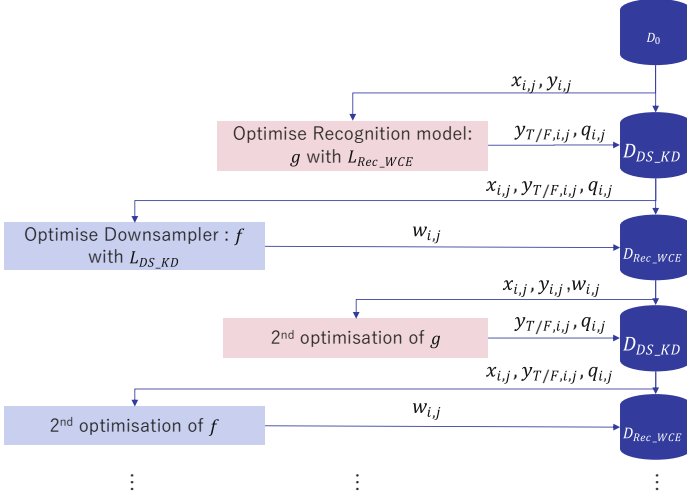


Fig. 3. Flow chart of the cyclic learning

We propose a cyclic learning framework for high-speed camera image recognition. It optimises the downsampler and the recognition model alternately. At first, It optimises the downsampler with the new labels produced by the recognition model’s inference correctness. Then, it retraines the recognition model based on the sample weights updated by the downsampler. This process is repeated until the data convergence or early stopping with validation data.

This cyclic learning framework is shown in a flow chart Fig. 3 and Algorithm 1. The initial data D_0 is set as the training and validation data. The initial sample weights S_i^w are set to 1.0 for all i and j . The initial recognition model g is learned by optimizing L_{Rec_WCE} with D_0 . Then, the new labels $S_i^{T/F}$ are obtained by the recognition model g . The downsampler f is learned by optimizing L_{DS_KD} with D_{DS_KD} . The sample weights S_i^w are obtained by the downsampler f . The recognition model g is learned by optimizing L_{Rec_WCE} with D_{Rec_WCE} . This process is repeated until the data convergence or early stopping with validation data.

Algorithm 1 Cyclic learning of the downsampler and the recognition model

Require: initial data: $D_0 = \{(S_i, S_i^y)\}, f, g, S_i^w$
1: set initial sample weights: $S_i^w = \{1.0\}$ for all i and j
2: learn the initial recognition model g : optimize $L_{Rec_WCE}^S$ with D_0
3: **repeat**
4: get new labels: $S_i^{T/F}$ with the recognition model g
5: set data: $D_{DS_KD} = \{(S_i, S_i^{T/F})\}$
6: learn the downsampler f : optimize $L_{DS_KD}^f$ with D_{DS_KD}
7: get data: S_i^w with the downsampler f
8: set data: $D_{Rec_WCE} = \{(S_i, S_i^y, S_i^w)\}$
9: learn the recognition model: optimize $L_{Rec_WCE}^S$ with D_{Rec_WCE}
10: **until** Convergence of D or EarlyStopping with validation data.

4 Evaluation

Datasets

To evaluate the effectiveness of our approach on temporally dense images produced by a high-speed camera, we created a new dataset because there was no practical high-speed camera dataset with industrial applications such as anomaly inspection in mind. In our dataset, tablets speedily rolling over a curved lane on a slope are captured by a high-speed camera. These tablets invert at the middle of the road so that we can capture with multiple frames images the instant appearance of an anomaly anywhere on the tablets' surface (the figure of the lane and the field of view(FOV) can be viewed in the supplemental material). Figure 4 shows an example sequence of temporally dense images of an anomaly tablet from our dataset. The same setting is popular at production inspection lines, but human visual inspectors inspect them instead of a high-speed camera system, and tablets' speed is relatively slow. These tablets have minimal dotted black ink somewhere randomly on their surfaces as a simulated anomaly, and it is captured with a few pixels with an unclear edge. Tablets roll over and invert, making conventional image processing-based inspection algorithms developed for constant object pose and illumination hard to distinguish the anomaly. We carefully cleaned the environment around the lane so that something else did not contaminate the normal tablets. Moreover, human annotators watched the resulting sequences frame by frame and labelled anomalies to avoid each image while checking contaminations. We have captured 900 sequences of images for anomaly tablets and 900 for normal tablets. We divided the sequences into three parts equally for training, validation, and test data. Each image in the sequence has a label of anomaly or not. Each sequence consists of around 200 images, though it varies depending on the rolling speed of the tablet. For a sequence of anomaly tablets, the images labelled as anomaly occupy about 10% of the total images in the sequence. More detailed information on the images in the dataset is described in the supplemental material.

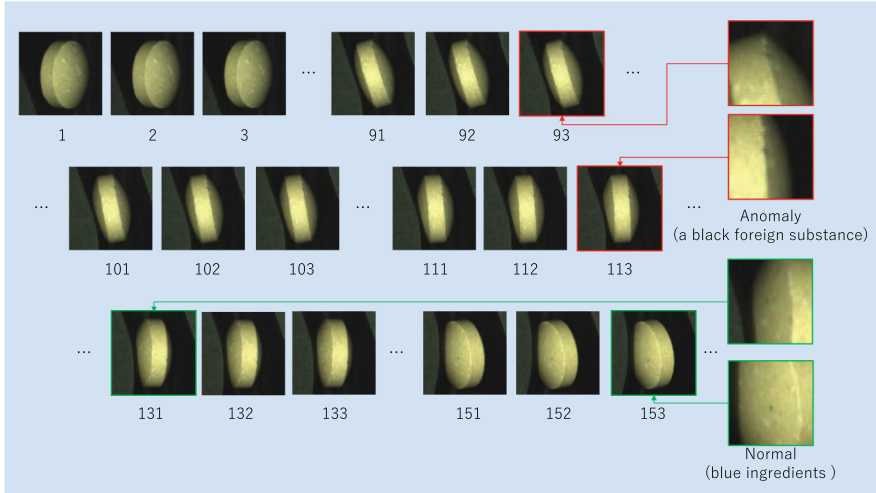


Fig. 4. An example sequence of temporally dense images of an anomaly tablet from our dataset. The number under each image is an index of the image. It is best viewed in colour.

Evaluation Metrics

To evaluate the effectiveness of our methods, we have chosen the Area Under Curve of Precision-Recall Curve (PR-AUC) as the basis of our evaluation metric. This metric is usually applied when the positives and negatives are imbalanced. In our case, the positives are the anomaly images in a sequence, and the negatives are the normal images. Usually, a high recall rate is more important than a high precision since missing an anomaly is to be avoided as much as possible. So we are interested in higher recall than lower. So, we calculate partial PR-AUC at a higher recall than a threshold. It can be viewed as partial AP because PR-AUC equals the average precision (AP). We apply this partial AP for our evaluation metric and set the threshold to 90%.

Prior Researches to be Compared with

We compared the proposed method with the prior research [9], which has proposed high-speed camera image recognition for the first time. Our method is based on [9] and adds novel cyclic learning approaches, so [9] should be compared as a baseline. We also add the naive method, which is the recognition model trained with the whole training data without the downsampler, and it takes randomly sampled images from a sequence as the input. This naive method is the simplest and most straightforward, but it is unsuitable for high-speed camera image recognition because it does not care which frames should be selected and might be unstable. Unfortunately, we could not find any other research that proposed high-speed camera image recognition so that we could compare our method with only [9] and the abovementioned naive method.

Implementation Details

The neural network architectures for a downsampler and the recognition model are arbitrary. However, the former should be much smaller than the latter, considering that the downsampler should process all images frame by frame and needs to run as fast as possible to capture the instant appearance of positives (a class of interest, i.e., an anomaly in our dataset) by multiple frame images. We explored such architecture by attaching auxiliary classifiers [13] to every convolutional layer in a base neural network model, learning with training data, and choosing the shallowest layer with the validation accuracy over a threshold. In our experiment, we choose MobileNetV2 [10] for the base model and 70% for the accuracy threshold. After training, the second convolutional layer was selected, and we extracted from the base model all the layers from the input layer to the corresponding auxiliary classifier as a base downsampler.

Note that this base downsampler was not the same one from [9], in which the downsampler was a linear model with hand-made features. Because we need to fairly evaluate the effectiveness of the retraining of the recognition model and cyclic learning, we need to apply the same base downsampler to [9] and ours.

For the base model of the recognition model, we also chose a MobileNetV2 [10] pre-trained with ImageNet without truncating any layers.

For the mapping function, we used the average of the output sequence $g(S_i^f)$.

We cycled up to 10 times for the cycle condition until the validation accuracy peaked. We set the batch size to 64 and the learning rate to 0.01. We set step per epoch to 2000 and epochs to 100. We used the Adam optimizer [5] for both the downsampler and the recognition model. We set the temperature T to 1.0 and the weight hyperparameter α to 0.5. We used the same training data for the downsampler and the recognition model. We used the same validation data for the early stopping of the cyclic learning. We used the same test data to evaluate the effectiveness of our approach.

The architecture of the models under comparison is organised as follows:

- Naive method: Random sampler + Recognition model trained with whole training dataset once.
- [9]: Downsampler trained with Recognition model’s output once + Recognition model trained with whole training dataset once.
- Ours: Downsampler trained with Recognition model’s output cyclically + Recognition model trained with whole training dataset cyclically.

Results

Table 1 shows partial AP(pAP) with the threshold 90% for the proposed and prior methods with various K . The proposed method outperformed the naive method and [9], and especially, at $K = 1$, 5.1% better pAP than the naive method and 3.6% better pAP than [9]. Table 2 shows the minimum K taken to achieve each AP level. Smaller K means faster inference. It is preferable for practical applications in the industry, such as visual inspection systems, because it

enables speeding up production lines. The proposed method outperformed the naive method and [9], 2.5 times faster than the naive method and two times faster than [9] on average on the AP range. These results demonstrated the effectiveness of our approach.

Table 1. Partial Average Performance (pAP) at higher recall area than 90% on our dataset compared to naive and existing methods.

Methods	K (number of samples)									
	1	2	3	4	5	6	7	8	9	10
Naive	69.9	72.8	74.5	75.7	76.7	77.3	77.8	78.2	78.2	78.7
[9]	71.3	73.7	75.5	76.4	77.1	77.6	78.2	78.4	79.1	79.2
Ours	74.9	77.4	78.7	79.8	80.5	80.7	80.9	80.6	81.0	81.1

Table 2. The minimum K taken to achieve each AP level. Smaller K means lower computation.

Methods	pAP levels										
	0.7	0.71	0.72	0.73	0.74	0.75	0.76	0.77	0.78	0.79	0.8
Naive	2	2	2	3	3	4	5	6	8	N/A	N/A
[9]	1	1	2	2	3	3	4	5	7	9	N/A
Ours	1	1	1	1	1	2	2	2	3	4	5

Analysis

Cycle Steps

We also analyze the effect of increasing cycle steps. We show the proposed method’s performance on cycle steps vs AP and 2D plots of AP in Fig. 5. Figure 5 shows that at an early stage of cycle steps (until the 3rd step), AP at the same K increased, and K to achieve the same level of AP decreased, but after that, they seemed to deteriorate and fluctuate. These results indicate that the downsampler effectively learns easy examples at an early stage, and after reaching the top level, the downsampler starts overfitting and it seems like perturbed. We also show the F-value of the downsampler at each number of cycle steps in Fig. 6. Comparing Fig. 5 and Fig. 6 shows that the downsampler’s F-value seems correlated with the AP. This is because the downsampler’s F-value is high when the recognition model is easy to fit, but it is not always high when the recognition model is hard to fit. These results indicate that the proposed cyclic learning strategy successfully found easy examples in the temporally dense images and fitted to them, enabling higher performance.

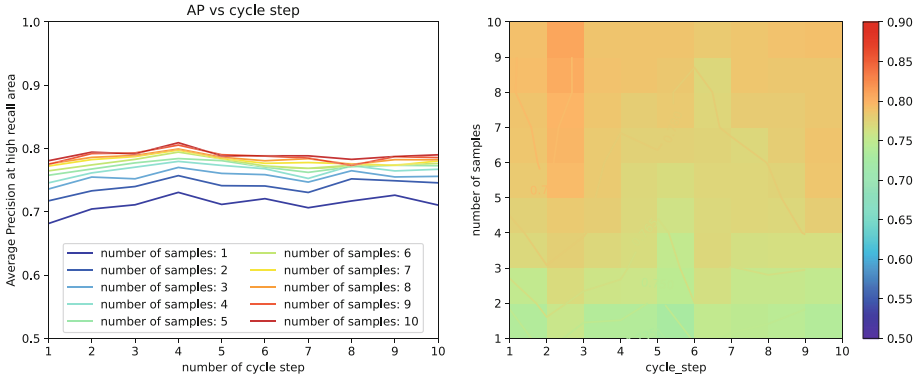


Fig. 5. Left: AP at higher recall rate area vs cycle steps. The higher number of samples: K is, the higher the AP is. The number of cycle steps where peaks of AP exists varies for different K , though they seem to be correlated. Right: 2D plots of AP at higher recall rate area conditioned by cycle steps and number of samples: K .

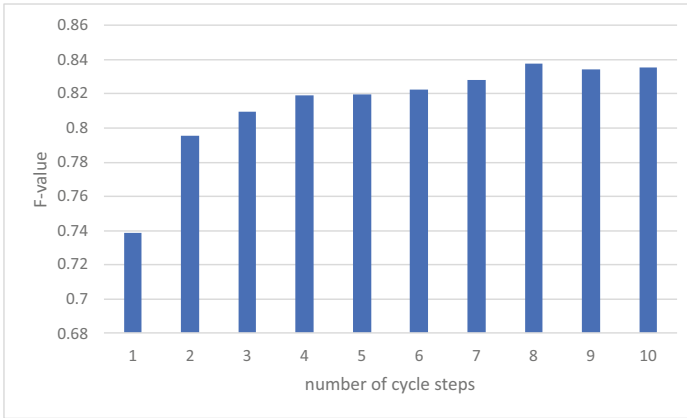


Fig. 6. F-value of the downsampler at each number of cycle steps.

5 Limitations

Our approach has some limitations. First, cyclic learning has a large computational cost. The computational cost is linearly proportional to the number of cycle steps. In our experiment, we set the number of cycle steps to 10, but it is not always necessary to set it to 10. We need to find the optimal number of cycle steps for each dataset. Second, generalization to other datasets is not guaranteed. While we have datasets the same as [9], we observed that the accuracy is so high that we cannot obtain any performance gain thanks to cyclic learning. Our method would not be available for such tasks. Future directions include reducing the computational cost and generalizing the method to other datasets.

6 Conclusions

This paper proposed a novel learning strategy to improve prior work on high-speed camera image recognition. In addition to the downsampler proposed in the prior work, we proposed to retrain the recognition model with the sample weights output by the downsampler and to cycle the learning of the downsampler and the recognition model. We evaluated the effectiveness of our approach on a newly constructed dataset with a high-speed camera. The results showed that our approach significantly improved the average precision under high recall conditions. We also analyzed the effect of increasing cycle steps and showed that the performance increased as cycle steps increased until they reached their peak. These results indicate that the proposed cycling strategy successfully found easy examples in the temporary dense images and fitted to them, enabling higher performance. Our approach is expected to be useful for practical applications in the industry, such as sorting or visual inspection systems because it speeds up production lines.

References

1. Benmeziane, H., Maghraoui, K.E., Ouarnoughi, H., Niar, S., Wistuba, M., Wang, N.: A comprehensive survey on hardware-aware neural architecture search. *ArXiv abs/2101.09336* (2021). <https://api.semanticscholar.org/CorpusID:231699126>
2. Cheng, Y., Wang, D., Zhou, P., Tao, Z.: A survey of model compression and acceleration for deep neural networks. *ArXiv abs/1710.09282* (2017). <https://api.semanticscholar.org/CorpusID:22163846>
3. Dosovitskiy, A., et al.: An image is worth 16×16 words: transformers for image recognition at scale. *ArXiv abs/2010.11929* (2020). <https://api.semanticscholar.org/CorpusID:225039882>
4. Hinton, G.E., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. *ArXiv abs/1503.02531* (2015). <https://api.semanticscholar.org/CorpusID:7200347>
5. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. *CoRR abs/1412.6980* (2014). <https://api.semanticscholar.org/CorpusID:6628106>
6. Lee, J., et al.: Resource-efficient deep learning: a survey on model-, arithmetic-, and implementation-level techniques (2021)
7. Lee, S., Kim, H., Ishikawa, M.: Deep learning approach to face pose estimation for high-speed camera network system. In: 2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIIC), pp. 084–088 (2020). <https://api.semanticscholar.org/CorpusID:215816283>
8. Li, Y., Yu, Y.Z., Zou, Y.X., Xiang, T., Li, X.: Online easy example mining for weakly-supervised gland segmentation from histology images. *ArXiv abs/2206.06665* (2022). <https://api.semanticscholar.org/CorpusID:249642594>
9. Namiki, S., Yokoyama, K., Yachida, S., Shibata, T., Miyano, H., Ishikawa, M.: Online object recognition using CNN-based algorithm on high-speed camera imaging: framework for fast and robust high-speed camera object recognition based on population data cleansing and data ensemble. In: 2020 25th International Conference on Pattern Recognition (ICPR), pp. 2025–2032 (2021). <https://api.semanticscholar.org/CorpusID:233877508>

10. Sandler, M., Howard, A.G., Zhu, M., Zhmoginov, A., Chen, L.C.: Mobilenetv2: inverted residuals and linear bottlenecks. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4510–4520 (2018). <https://api.semanticscholar.org/CorpusID:4555207>
11. Schleier, M., Adelmann, B., Esen, C., Hellmann, R.: Image processing algorithm for in situ monitoring fiber laser remote cutting by a high-speed camera. *Sensors (Basel, Switzerland)* **22** (2022). <https://api.semanticscholar.org/CorpusID:248073023>
12. Song, Q.B., Kubota, N., Zhang, Y.: Posture recognition for human-robot interaction based on high speed camera. In: 2022 World Automation Congress (WAC), pp. 419–423 (2022). <https://api.semanticscholar.org/CorpusID:253423833>
13. Szegedy, C., et al.: Going deeper with convolutions. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–9 (2014). <https://api.semanticscholar.org/CorpusID:206592484>
14. Yamakawa, Y., Matsui, Y., Ishikawa, M.: Human-robot collaborative manipulation using a high-speed robot hand and a high-speed camera. In: 2018 IEEE International Conference on Cyborg and Bionic Systems (CBS), pp. 426–429 (2018). <https://api.semanticscholar.org/CorpusID:58673585>
15. Zhou, Y., et al.: Cyclic learning: bridging image-level labels and nuclei instance segmentation. *IEEE Trans. Med. Imaging* **42**, 3104–3116 (2023). <https://api.semanticscholar.org/CorpusID:258659613>
16. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: 2017 IEEE International Conference on Computer Vision (ICCV), pp. 2242–2251 (2017). <https://api.semanticscholar.org/CorpusID:206770979>



Early Feature Distributions Alignment in Visible-to-Thermal Unsupervised Domain Adaptation for Object Detection

Adrien Maglo^(✉) and Romaric Audigier

Université Paris-Saclay, CEA, List, 91120 Palaiseau, France
{adrien.maglo,romaric.audigier}@cea.fr

Abstract. Infrared or thermal images are used in many civilian and military applications to detect objects due to the heat they emit, especially when environmental conditions such as nighttime or adverse weather prevent the use of visible images. To train an object detector based on a deep neural network, a significant amount of annotated data is required to achieve good detection performance. However, annotations for infrared images are often unavailable and costly to obtain. Besides, the trained model may show poor robustness against the change of thermal sensor. Therefore, unsupervised domain adaptation (UDA) methods have been proposed to train an object detector with annotated visible images, which are easily available, and unannotated infrared images. This paper presents a new visible-to-thermal UDA approach for object detection based on Deformable-DETR with hybrid matching. Our approach aims to establish common features between visible and thermal images at the earliest stages of the backbone network. The feature distributions extracted from visible and thermal images are aligned thanks to discriminator networks and adversarial learning. Gradient images are also used as a domain translation of input images to ease the alignment. Detection performance is further improved by randomly masking tokens at the input of the transformer. Experiments on public datasets demonstrate that our method consistently outperforms previous works.

Keywords: unsupervised domain adaptation · object detection with transformers · thermal imaging · feature alignment

1 Introduction

Thermal infrared images are used in many applications in both military and civilian domains. They enable the detection of people and objects by capturing the heat they emit, which is particularly useful in nighttime or adverse weather conditions. The detection task involves generating bounding boxes around objects and classifying them. Many detectors have been proposed in the literature based on deep learning approaches, achieving good performance across a wide range of objects. However, they require a significant amount of training data. Large detection datasets in the visible domain, such as MS-COCO [30], have been proposed to

train these models. Despite efforts to release large-scale infrared datasets [11, 21], annotated datasets in the infrared domain are much less common than in the visible domain. Additionally, using different sensors (response, quality, sensitivity, etc.) under varying weather conditions may result in thermal images with different distributions.

The challenges related to thermal data collection and annotation have led to the development of unsupervised domain adaptation (UDA) methods from the visible to the thermal domain, allowing the utilization of knowledge from the more readily available visible domain. The model is trained using both visible and thermal images. However, annotations are only provided in the visible domain. Consequently, the model learns the task in the thermal domain through pseudo-labeling or feature distribution alignment. Some previous works have focused on visible-to-thermal UDA for classification and segmentation tasks, while only Marnissi et al. proposed an approach for the object detection task [31]. The UDA for the detection task from one visible domain to another visible domain has been widely studied in the literature [32]. However, UDA from the visible domain to the thermal domain presents a different challenge. Approaches must contend with two distinct domains that possess very distinctive features. Additionally, thermal images have a single component, while RGB images have three.

We therefore propose a new visible-to-thermal UDA detection framework that aims to early align the distribution of the features extracted from both domains. Our framework is based on the Deformable-DETR detector with hybrid matching (H-Deformable-DETR) [22]. Many previous works align the distribution of features after the backbone, at the detection stage of the model. We demonstrate that early alignment of the features within the backbone can be beneficial for the visible-to-thermal domain adaptation task. Our detection model takes multi-scale backbone features as input. We propose to align the distribution of these features from the two domains using discriminator networks and adversarial training. Furthermore, we align the visible and thermal images by using gradient images as a common translated input modality for the model. Gradient images extracted from visible and thermal images are indeed much more similar than the original images. Finally, we apply token masking to the input of the detector transformer to improve its robustness.

The remainder of the paper is organized as follows. In the second section, we introduce previous work about UDA for detection and visible-to-thermal domain adaptation. We describe our method in the third section. Experimental results on two public datasets are provided in the fourth section.

2 Related Work

Unsupervised domain adaptation (UDA) involves training a model with annotated data in the source domain and unannotated data in the target domain. This technique enables the training of a model adapted to a target domain without requiring annotation for the target data. In the literature, various UDA methods have been proposed for classification tasks, segmentation tasks, and

detection tasks. In this section, we will first review previous work related to UDA for object detection. Subsequently, we will delve into the specific case of visible-to-thermal UDA.

2.1 Unsupervised Domain Adaptation for Object Detection

The UDA methods for object detection can be classified into three main categories [32]: pseudo-labeling, domain invariant feature learning and image-to-image translation.

Pseudo-labeling frameworks generate annotations for the target images using confident detections obtained by a model trained on the source data. Soft labeling is employed in the framework proposed by RoyChowdhury et al. [35] to mitigate the risk of incorrect pseudo-labels. In the approach by Khodabandeh et al. [24], bounding boxes are generated by the detection model trained on labeled source data while pseudo-label classes are provided by an additional image classifier. Kim et al. [25] proposed an algorithm that mines positive samples and weak-negative samples for each class of pseudo-labels. Zhao et al. [47] introduced a method that aligns features by minimizing the discrepancy between the Faster R-CNN region proposal network and the region proposal classifier. Other approaches utilize a mean-teacher architecture where the teacher model generates pseudo-labels to train the student. The teacher weights are then updated from the student model using exponential moving average (EMA). Cai et al. [2] perform random augmentations on a target image to obtain two images, ensuring the consistency of student predictions between them. In the recent MIC approach [18], the student network is trained by matching the pseudo-labels it generates on masked target images with those generated by the teacher. The Harmonious Teacher method [10] focuses on improving the consistency between classification scores and the Intersection-Over-Union between predicted and real object bounding boxes.

Domain invariant feature learning methods focus on aligning the features extracted by the model between the source and target domains. This is often achieved by adding discriminator modules to the original detector, which learn to classify whether the images come from the source domain or the target domain. Thus, the objective for the detector is to generate common features between the two domains. Therefore, a gradient reversal layer [13] is often added between the feature outputs and the discriminator in order to achieve this contradictory goal. The approach of Chen et al. [6] aligns the features produced by a Faster R-CNN model [34] at both the instance level and global image level. Its extension [7] integrates a feature pyramid network to independently align the image and object features of each scale. In the framework of Saito et al. [36], local image-based features and global instance-based features are extracted and aligned at two different levels of the network. Hsu et al. [19] align the instance features at the center of the object proposals. MeGA-CDA [39] aligns the features with a discriminator at the global level and a discriminator for each category. Since the object categories are unknown for the target images, memory-guided attention maps redirect the features to each discriminator. Li et al. [29] use a mean-teacher

architecture and integrate a discriminator to align the distribution of features generated by the student network.

Graph reasoning techniques model the relations between objects and categories in the source and target images as graphs. The framework proposed by Xu et al. [42] aligns the detected object proposals by merging them. It also aligns the object classes between domains by improving the compactness of each class and its separability from others. Similarly, I3Net [5] follows the same alignment objectives. It weights the target samples based on adaptation difficulty, boosts foreground objects, suppresses redundant background information, and aligns category features between domains using consistency regularization. SIGMA [28] transforms model features into graphs and employs graph matching theory to align class feature distributions.

Image-to-image translation methods involve using a model to convert images from one domain to another. Chen et al. [4] utilize CycleGAN [48] for generating synthetic samples and enhancing the training of the adversarial domain discriminator. Similarly, CycleGAN is employed by Hsu et al. [20] to create synthetic annotated images. Subsequently, their features are aligned with target image features using an adversarial discriminator. Deng et al. [9] use images translated from the source to the target domain with CycleGAN to mitigate the bias of the teacher and student networks towards their trained domain.

The methods listed above are based on convolutional detectors, with the most frequent one being Faster R-CNN. However, recent approaches have also been proposed for Deformable-DETR detectors [49] based on transformers [38]. MTTrans [43] utilizes a mean teacher approach for pseudo-label generation. The method proposed by Wang et al. employs a feature alignment strategy [40]. DA-DETR [46] adds feature fusion modules to enable information communication across channels. These approaches do not directly align features at multiple output levels of the backbone. While multiresolution feature alignment has been studied for Faster R-CNN detectors [17, 41], it has never been used with DETR architectures. Visible and thermal images have very different characteristics. We believe that early feature alignment is important for efficient visible-to-thermal UDA. Multiresolution feature alignment in the backbone network can achieve this objective.

2.2 Visible-to-Thermal Domain Adaptation

The unsupervised domain adaptation from visible-to-thermal images has received less attention in the literature. For the semantic segmentation task, MS-UDA [26] performs UDA from a large visible dataset to a smaller unlabeled visible and thermal paired dataset using pseudo-label generation. Gan et al. [12] employ domain-specific attention maps for segmentation and classification tasks. The network is trained with adversarial learning and fine-tuned with pseudo-labels. Akkaya et al. [1] select high-confidence pseudo-labels that fool a trained domain discriminator. Regarding the detection task, Lee et al. proposed a GAN-based visible to thermal image translation method [27] that focuses on

preserving the edges. It is trained on a combination of large visible and thermal datasets. They conducted thermal detection experiments by training a VFNet detector [44] on a synthetic dataset [23] translated using their method. To our knowledge, only Marnissi et al. [31] have attempted visible-to-thermal UDA for detection. Their UDAT framework, based on Faster R-CNN, requires annotations only in the visible domain. It also aligns features at four different feature map levels and instance levels. Given the distinct characteristics of visible and thermal images, our approach proposes to align thermal and visible features at shallower levels of the network so that deeper levels can benefit from features with common distributions. It also aligns visible and thermal images by using gradient images, as a common translation domain, at the input of the model.

3 Our Method

3.1 Overview

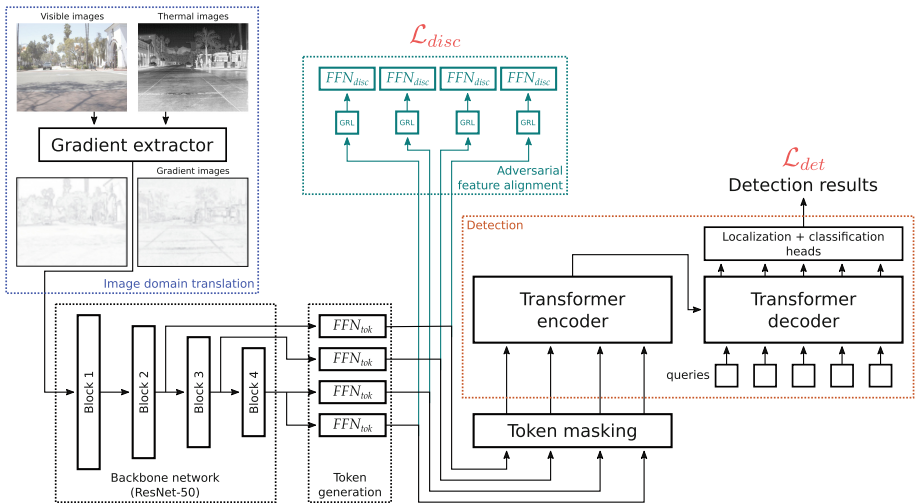


Fig. 1. Illustration of the training of our visible-to-thermal unsupervised domain adaptation method for object detection based on H-Deformable-DETR. The model is trained with a supervised detection loss \mathcal{L}_{det} on visible images. The distribution of features extracted from visible and thermal images are aligned with adversarial learning. The discriminative loss \mathcal{L}_{disc} trains the discrimination networks FFN_{disc} connected to the token generators FFN_{tok} of the backbone output levels through gradient reversal layers GRL (details in Sect. 3.3). Gradient images are also used as a common modality for backbone inputs (details in Sect. 3.4). Token masking improves the detection robustness (details in Sect. 3.5).

Our framework is based on the H-Deformable-DETR detector [22]. In order to detect objects in the thermal domain, for which we do not use annotated training data, we simultaneously train our detector in a supervised manner with annotated training data from the visible domain and align the distribution of features extracted at several levels of the backbone network. We refer to this strategy as “early alignment” (EA) because the prioritized feature distribution alignments occur at the shallowest output levels of the backbone network. This alignment is performed using discriminator networks connected to the model through gradient reversal layers. We use the image gradient operation as an image domain translation method. The gradient images, extracted from visible and thermal images, serve as inputs to the model. They reduce the domain gap between the visible and thermal images. Finally, we apply token masking at the output of the backbone to increase the model’s generalization (See Fig. 1).

3.2 Baseline Detector

The H-Deformable-DETR detector belongs to the *DETR* family of detectors [3] based on transformers [38]. With the *Deformable-DETR* detector [49], the input images are first processed by a ResNet-50 backbone [16] that extracts features at a single resolution. These features are then transformed into tokens by the feedforward networks FFN_{tok} . The shallowest output layer of the ResNet-50 backbone is discarded, and an additional output layer is artificially added by applying another FFN_{tok} to the last output layer. Positional embedding is then added to the tokens. They are processed by a transformer encoder and then by a transformer decoder. The decoder takes as additional inputs object queries that are learned parameters. The model predicts for each decoder output token a class score and a bounding box. During training, each ground-truth object is associated with a decoder query using Hungarian matching based on class scores and bounding box IoU criteria. Deformable-DETR replaces the transformer attention modules with multi-scale deformable attention modules. Instead of computing an attention map for all input feature locations, the deformable attention module is trained to sample only a few significant points around the reference point. This sampling is done at different feature-map scales. It speeds up the model training and improves the detection of small objects. The hybrid matching of *H-Deformable-DETR* increases performance by employing a second round of ground truth and object query matching. In this round, each ground truth can be assigned to multiple decoder queries from a second set of queries. We utilize the two-stage variant of H-Deformable-DETR [49]. The encoder generates object proposals, and the proposals with the highest scores are selected to be refined by the transformer decoder. Their bounding boxes are fed to the transformer decoder as positional embeddings of the decoder object queries. The model is trained in a supervised way with the images and the annotations of the visible domain. The supervised detection losses are the same as with the original H-Deformable-DETR. We call their sum \mathcal{L}_{det} .

3.3 Early Feature Distribution Alignment

Our main objective is to build a detector that has a high performance in the thermal domain. Consequently, we want our backbone to generate domain agnostic tokens. Our model should produce features with the same distribution for the thermal or visible images. The features learnt with annotated visible images should also be a good representation of thermal images. To this end, we add at each output of the backbone, gradient reversal layers *GRL* followed by discriminator networks FFN_{disc} . The role of the discriminators is to classify the tokens: they determine whether tokens come from a thermal image or a visible image. Each discriminator is composed of 5 linear layers with the same dimension as the transformer. The first 4 layers are followed by a ReLU activation function. The output dimension of the last layer is 1. Our discriminator learns to classify tokens coming from either thermal or visible images. However, we aim for backbone features from both domains to be indistinguishable. Therefore, the *GRL* inverts the signs of the gradients to enable the adversarial learning between the backbone and the discriminators. We use a cross entropy loss to train the discriminator networks:

$$\mathcal{L}_{disc} = - \sum_l w_l \sum_t y_{l,t} \times \log(x_{l,t}) + (1 - y_{l,t}) \times \log(1 - x_{l,t})$$

where l is the output layer of the backbone network, w_l is a weight for the layer l , t is the token, $x_{l,t}$ is the output of the discriminator network for the layer l and token t and $y_{l,t}$ is its target value. Features extracted by the backbone network at the shallowest layers are more of spatial nature while features extracted at the deepest levels of the network are more semantic, so less dependent from the input domain. As we want an early alignment of the features, we set much higher weight w_l to the shallower layer outputs than to the deeper ones.

During training, we build mini-batches with one half of the images coming from the visible domain and the other from the thermal domain. The adversarial discriminative loss \mathcal{L}_{disc} applies to both thermal and visible images. The feature alignment task and the detection tasks have two objectives that may disturb each other. The feature alignment task may want to generate completely uniform distribution of features so the discriminator is unable to determine whether they come from visible or thermal images. To balance the importance of the feature alignment task relative to the detection task, we dynamically weight \mathcal{L}_{disc} with the coefficient α based on the value of \mathcal{L}_{det} , ensuring that a constant ratio r_{loss} between the two losses is maintained:

$$\frac{\alpha \mathcal{L}_{disc}}{\mathcal{L}_{det}} = r_{loss}$$

where r_{loss} is a constant positive parameter set for the entire training. At each iteration, \mathcal{L}_{det} and \mathcal{L}_{disc} are first computed on the total of the mini-batch of images. Then α is determined with the formula:

$$\alpha = \frac{r_{loss} \mathcal{L}_{det}}{\mathcal{L}_{disc}}.$$

No gradient is backpropagated before the determination of α . It becomes a scaling constant for the computation of the total loss:

$$\mathcal{L}_{tot} = \mathcal{L}_{det} + \alpha \mathcal{L}_{disc}.$$

Notice that α is forced to zero for the first epoch in order to bootstrap the detector without the discriminative loss. This mechanism improves the stability of the training.

3.4 Input Gradient Images



Fig. 2. Visible and thermal images from the Free FLIR dataset [11] (top) and their respective Sobel gradient intensity images (bottom). Despite the fact that the visible and thermal gradient images do not outline the same visual features, the domain gap appears to be narrower than with the original images.

We use the gradient images as a common modality to reduce the domain gap between the visible and thermal. The Sobel [37] and Prewitt [33] image gradients have the advantage of being quick to compute. Their intensity images have a similar appearance between visible and thermal images, as depicted in Fig. 2. The gradient outlines edges in the input images, which is crucial for detecting objects in both domains. We compute the gradients for each axis with the following convolutions:

$$\mathbf{G}_{\text{Prewitt}_x} = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} * \mathbf{I} \quad ; \quad \mathbf{G}_{\text{Prewitt}_y} = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ +1 & +1 & +1 \end{bmatrix} * \mathbf{I}$$

$$\mathbf{G}_{\text{Sobel}_x} = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} * \mathbf{I} \quad ; \quad \mathbf{G}_{\text{Sobel}_y} = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} * \mathbf{I}$$

Gradient images are then obtained by taking the L^2 norms of the gradients:

$$\mathbf{G}_{\text{Sobel}} = \sqrt{\mathbf{G}_{\text{Sobel}_x}^2 + \mathbf{G}_{\text{Sobel}_y}^2} \quad ; \quad \mathbf{G}_{\text{Prewitt}} = \sqrt{\mathbf{G}_{\text{Prewitt}_x}^2 + \mathbf{G}_{\text{Prewitt}_y}^2}$$

The gradient images are then normalized between 0 and 1. During training, we randomly switch between the Sobel and Prewitt gradients to artificially augment the amount of training data. During inference, only the Sobel gradient is used.

3.5 Detector Token Masking

Image masking in the pixel space has shown its effectiveness for UDA [18]. Instead, we propose to leverage this mechanism by randomly masking the transformer tokens from all input levels. Token masking has also been shown to be beneficial for pretraining Vision Transformer models [15]. In our case, it aims at forcing the model to rely on features from all the input levels of the transformer by reducing the overfitting. Token masking is performed by randomly selecting a random ratio α_m of token at the input of the transformer encoder and setting their value to 0. Gradient backpropagation is halted for the masked tokens.

4 Experiments

4.1 Dataset

To run our experiments, we use the Free FLIR “aligned” dataset [45], version derived from the original version 1.3 [11]. It provides annotations for 4,129 well-aligned thermal and visible image pairs for training and 1,013 image pairs for testing. However, in this UDA work no alignment is used: the visible and thermal images from the training set are used in an unpaired way during training. The testing is performed on the thermal images from the test set. In addition, only the person, bicycle and car classes are considered.

Experiments are also performed on the KAIST dataset [21]. We use the “sanitized” annotations and the image sets provided in the latest release of the dataset, selecting one out of every four images for the train set and one out of every twenty images for the test set. In line with previous evaluation protocols on this dataset [31, 45], only instances annotated with the class “person”, “person?” or “people” are kept and grouped in a common “pedestrian” class.

Bounding boxes with the minimum of the width and height inferior to 50 pixels or flagged as occluded are discarded. At the end, only images with at least a valid bounding box are used for training and testing. This resulted in a dataset of 4,110 image pairs with 7,908 instances in visible images for training and 859 thermal images with 1,846 instances for testing.

4.2 Implementation Details

We built our framework on top of the implementation of H-Deformable-DETR [14] based on the Pytorch framework. Our model utilizes a ResNet-50 backbone [16] pretrained on ImageNet [8], with the remaining parts of the model initialized with random weights. As base configuration for the H-Deformable-DETR, we chose the two-stage configuration from the official implementation that performs the best on the MS-COCO dataset [30]. Thus, the number of queries for the one-to-one matching is set to 300. For the one-to-many matching, each ground truth is set to one of 1500 queries. The weight of the one-to-many matching loss is set to 1. The mixed selection is used. The dimension inside the transformer is set to 256 and its feed-forward network dimension is set to 2048. The data augmentation techniques of Deformable-DETR are used: random horizontal flip, crop and resize. In our experiments, all the model layers are trained during 12 epochs with the AdamW optimizer on two NVIDIA RTX A5500 GPUs with 24 GB or RAM. The learning rate is set to 2×10^{-5} for the backbone network and 2×10^{-4} for the rest of the network. It is divided by 10 after 11 epochs. The weight decay is set to 10^{-4} . The batch size is set to 4: two random visible images and two random thermal images. The w_l feature distribution alignment weights are set to 10, 1, 10^{-4} and 10^{-5} for shallower to deeper layers, respectively. The token masking ratio α_m is set to 0.2. The ratio between the discrimination and the detection loss r_{loss} was fine-tuned to 0.32 after a grid-search on the Free FLIR dataset. The same value of r_{loss} is used for the experiments on the KAIST dataset. We observed that the concurrency between the supervised detection loss \mathcal{L}_{det} and the discrimination loss \mathcal{L}_{disc} can lead to training instabilities and catastrophic detection performance. Disabling \mathcal{L}_{disc} for the first training epoch removed this issue in our experiments.

4.3 Results

Experimental results on the Free FLIR dataset are reported in Table 1. We use the mAP metric with an IoU of 0.5. We compare our method to existing UDA state-of-the-art approaches generally evaluated on visible-to-visible benchmarks [4, 6, 7, 10, 18, 36, 40]. Only UDAT [31] is specialized in visible-to-thermal UDA. Some experimental results of previous work have been originally reported by Marnissi et al. [31]. For methods we trained and evaluated, we provide mean and standard deviation values over four different runs. Our method outperforms all previous works in terms of mAP. It reaches an average mAP of 68.4% on the Free FLIR dataset, about 4.9% points (pp) higher than the SOTA method

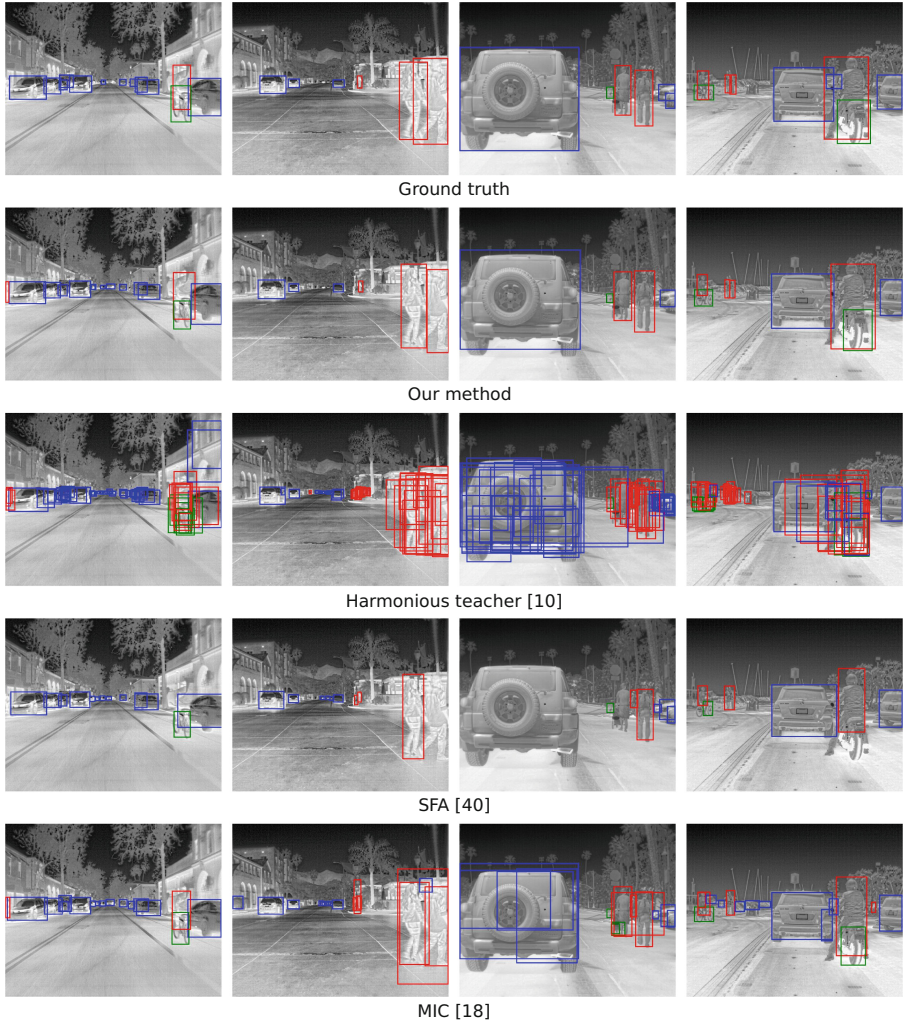


Fig. 3. Qualitative detection results on the thermal images of Free FLIR dataset. The red, blue and green bounding boxes correspond respectively to the person, car and bicycle classes. The score threshold is set to 0.4 for all the methods. No Non-Maximum Suppression is used. (Color figure online)

Harmonious teacher [10]. In order to demonstrate the performance improvements brought by each of our components, we conducted an ablation study. The results are provided in Table 1. The “Baseline” method corresponds to the H-Deformable-DETR detector trained on visible images and tested on thermal images without any adaptation. The early alignment of feature distributions improves the mAP by approximately 5.3 pp. The use of gradient images results in an additional improvement of around 7.5 pp. Finally, random token masking

Table 1. Performance in mAP (%) on the Free FLIR aligned dataset without using alignment. Results marked by * have been originally reported by Marnissi et al. [31]. The others show mean and standard deviation of the mAP we obtained by training on 4 runs each. EA stands for the early alignment of features distributions; grad. img. for the use of gradient images; mask. token. for the random masking of the transformer encoder input tokens.

Method	Car	Bicycle	Person	Average mAP
DA-faster [6]*	59.90	24.30	26.60	36.93
SWDA [36]*	58.96	32.02	32.32	41.40
HTCN [4]*	56.37	37.95	33.17	42.49
SA-DA-faster [7]*	70.38	33.30	47.27	50.30
UDAT [31]*	66.83	49.34	43.41	53.19
MIC [18]	67.89 ± 5.81	48.45 ± 5.53	57.20 ± 6.85	57.85 ± 5.96
SFA [40]	77.33 ± 1.37	45.14 ± 3.66	55.58 ± 2.23	59.35 ± 1.44
Harmonious teacher [10]	78.36 ± 1.25	45.67 ± 0.71	66.46 ± 1.43	63.50 ± 0.79
Baseline	68.11 ± 2.09	49.91 ± 2.01	45.15 ± 2.90	54.38 ± 2.25
EA	75.29 ± 1.16	53.35 ± 0.94	50.46 ± 1.56	59.70 ± 1.04
EA + grad. img.	82.81 ± 0.39	51.27 ± 2.14	67.47 ± 1.25	67.18 ± 1.00
EA + grad. img. + mask. tok.	82.76 ± 0.47	54.55 ± 1.31	67.92 ± 0.69	68.42 ± 0.32

enhances the mAP by 1.2 pp. Some qualitative detection results on the Free FLIR dataset are provided in Fig. 3.

Experimental results on the KAIST dataset are reported in Table 2. We compare our method with the previous works that performed best on the Free FLIR dataset. Surprisingly, the Harmonious teacher did not perform so well in this benchmark, whereas our approach outperforms the best SOTA method, MIC [18], by about 4.1 pp. We also conducted an ablation study on this dataset, which shows performance increase for each of the components of our approach.

4.4 Discussion

Our approach consistently outperforms previous works on the Free FLIR and KAIST datasets. It uses less computational resources during training than mean teacher approaches [10, 18] that must store, at least, two versions of the model in memory. Our method is easily implemented on top of the efficient H-Deformable-DETR detector that has available source code [14]. It uses the same initial 48 million parameters with only 1 million extra parameters for the domain discriminators FFN_{disc} during training.

Table 2. Performance in mAP (%) on the KAIST dataset.

Method	mAP
Harmonious teacher [10]	32.79 \pm 2.08
SFA [40]	37.86 \pm 0.80
MIC [18]	41.95 \pm 3.38
Baseline	26.45 \pm 3.69
Early alignment	34.04 \pm 3.35
Early alignment + gradient images	42.65 \pm 2.62
Early alignment + gradient images + masked tokens	46.04 \pm 1.94

5 Conclusion

We present in this paper a new visible-to-thermal unsupervised domain adaptation method based on an efficient H-Deformable-DETR detector. We demonstrate that early feature distribution alignment combined with image domain translation through gradient images is key to achieving good detection performance in the thermal domain. For future work, we aim to study the performance of our method on thermal images captured by various sensors under different weather and temperature conditions. Additionally, we plan to explore the applicability of the early alignment and gradient translation principles to segmentation approaches.

Acknowledgements. This work benefited from a government grant managed by the French National Research Agency (ANR-22-ASTR-0010-02) and the FactoryIA supercomputer financially supported by the Ile-de-France Regional Council.

References

1. Akkaya, I.B., Altinel, F., Halici, U.: Self-training guided adversarial domain adaptation for thermal imagery. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4322–4331 (2021)
2. Cai, Q., Pan, Y., Ngo, C.W., Tian, X., Duan, L., Yao, T.: Exploring object relation in mean teacher for cross-domain detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11457–11466 (2019)
3. 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
4. Chen, C., Zheng, Z., Ding, X., Huang, Y., Dou, Q.: Harmonizing transferability and discriminability for adapting object detectors. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8869–8878 (2020)
5. Chen, C., Zheng, Z., Huang, Y., Ding, X., Yu, Y.: I3net: implicit instance-invariant network for adapting one-stage object detectors. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12576–12585 (2021)


6. 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)
7. Chen, Y., Wang, H., Li, W., Sakaridis, C., Dai, D., Van Gool, L.: Scale-aware domain adaptive faster R-CNN. *Int. J. Comput. Vis.* **129**(7), 2223–2243 (2021)
8. 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 (2009)
9. Deng, J., Li, W., Chen, Y., Duan, L.: Unbiased mean teacher for cross-domain object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4091–4101 (2021)
10. Deng, J., Xu, D., Li, W., Duan, L.: Harmonious teacher for cross-domain object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 23829–23838 (2023)
11. Free teledyne flir thermal dataset for algorithm training. <https://www.flir.com/oem/adas/adas-dataset-form/>. Accessed 08 Mar 2024
12. Gan, L., Lee, C., Chung, S.J.: Unsupervised RGB-to-thermal domain adaptation via multi-domain attention network. In: 2023 IEEE International Conference on Robotics and Automation (ICRA), pp. 6014–6020 (2023)
13. Ganin, Y., et al.: Domain-adversarial training of neural networks. *J. Mach. Learn. Res.* **17**(59), 1–35 (2016)
14. Official implementation of the paper “DETRs with hybrid matching”. <https://github.com/HDETR/H-Deformable-DETR>. Accessed 05 Apr 2024
15. He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick, R.: Masked autoencoders are scalable vision learners. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 16000–16009 (2022)
16. 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)
17. He, Z., Zhang, L.: Multi-adversarial faster-rcnn for unrestricted object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 6668–6677 (2019)
18. Hoyer, L., Dai, D., Wang, H., Van Gool, L.: MIC: masked image consistency for context-enhanced domain adaptation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11721–11732 (2023)
19. Hsu, C.-C., Tsai, Y.-H., Lin, Y.-Y., Yang, M.-H.: Every pixel matters: center-aware feature alignment for domain adaptive object detector. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12354, pp. 733–748. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58545-7_42
20. Hsu, H.K., et al.: Progressive domain adaptation for object detection. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 749–757 (2020)
21. Hwang, S., Park, J., Kim, N., Choi, Y., So Kweon, I.: Multispectral pedestrian detection: benchmark dataset and baseline. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1037–1045 (2015)
22. Jia, D., et al.: DETRs with hybrid matching. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 19702–19712 (2023)
23. Johnson-Roberson, M., Barto, C., Mehta, R., Sridhar, S.N., Rosaen, K., Vasudevan, R.: Driving in the matrix: Can virtual worlds replace human-generated annotations for real world tasks? arXiv preprint [arXiv:1610.01983](https://arxiv.org/abs/1610.01983) (2016)

24. Khodabandeh, M., Vahdat, A., Ranjbar, M., Macready, W.G.: A robust learning approach to domain adaptive object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 480–490 (2019)
25. Kim, S., Choi, J., Kim, T., Kim, C.: Self-training and adversarial background regularization for unsupervised domain adaptive one-stage object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 6092–6101 (2019)
26. Kim, Y.H., Shin, U., Park, J., Kweon, I.S.: MS-UDA: multi-spectral unsupervised domain adaptation for thermal image semantic segmentation. *IEEE Robot. Autom. Lett.* **6**(4), 6497–6504 (2021)
27. Lee, D.G., Jeon, M.H., Cho, Y., Kim, A.: Edge-guided multi-domain RGB-to-TIR image translation for training vision tasks with challenging labels. In: 2023 IEEE International Conference on Robotics and Automation (ICRA), pp. 8291–8298 (2023)
28. Li, W., Liu, X., Yuan, Y.: Sigma: semantic-complete graph matching for domain adaptive object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5291–5300 (2022)
29. Li, Y.J., et al.: Cross-domain adaptive teacher for object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7581–7590 (2022)
30. 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
31. Marnissi, M.A., Fradi, H., Sahbani, A., Essoukri Ben Amara, N.: Feature distribution alignments for object detection in the thermal domain. *Vis. Comput.* **39**(3), 1081–1093 (2023)
32. Oza, P., Sindagi, V.A., Sharmini, V.V., Patel, V.M.: Unsupervised domain adaptation of object detectors: a survey. *IEEE Trans. Pattern Anal. Mach. Intell.* (2023)
33. Prewitt, J.M., et al.: Object enhancement and extraction. *Pict. Process. Psychopictorics* **10**(1), 15–19 (1970)
34. 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)
35. RoyChowdhury, A., et al.: Automatic adaptation of object detectors to new domains using self-training. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 780–790 (2019)
36. Saito, K., Ushiku, Y., Harada, T., Saenko, K.: Strong-weak distribution alignment for adaptive object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6956–6965 (2019)
37. Sobel, I.: An isotropic 3×3 image gradient operator. Presentation at Stanford A.I. Project 1968 (02 2014)
38. Vaswani, A., et al.: Attention is all you need. *Adv. Neural Inf. Process. Syst.* **30** (2017)
39. Vs, V., Gupta, V., Oza, P., Sindagi, V.A., Patel, V.M.: MeGA-CDA: memory guided attention for category-aware unsupervised domain adaptive object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4516–4526 (2021)
40. Wang, W., et al.: Exploring sequence feature alignment for domain adaptive detection transformers. In: Proceedings of the 29th ACM International Conference on Multimedia, pp. 1730–1738 (2021)

41. Xie, R., Yu, F., Wang, J., Wang, Y., Zhang, L.: Multi-level domain adaptive learning for cross-domain detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (2019)
42. Xu, M., Wang, H., Ni, B., Tian, Q., Zhang, W.: Cross-domain detection via graph-induced prototype alignment. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12355–12364 (2020)
43. Yu, J., et al.: MTTrans: cross-domain object detection with mean teacher transformer. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) Computer Vision – ECCV 2022. ECCV 2022. LNCS, vol. 13669, pp. 629–645. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-20077-9_37
44. Zhang, H., Wang, Y., Dayoub, F., Sunderhauf, N.: Varifocalnet: an IoU-aware dense object detector. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8514–8523 (2021)
45. Zhang, H., Fromont, E., Lefevre, S., Avignon, B.: Multispectral fusion for object detection with cyclic fuse-and-refine blocks. In: 2020 IEEE International Conference on Image Processing (ICIP), pp. 276–280 (2020)
46. Zhang, J., Huang, J., Luo, Z., Zhang, G., Zhang, X., Lu, S.: DA-DETR: domain adaptive detection transformer with information fusion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 23787–23798 (2023)
47. Zhao, G., Li, G., Xu, R., Lin, L.: Collaborative training between region proposal localization and classification for domain adaptive object detection. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12363, pp. 86–102. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58523-5_6
48. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2223–2232 (2017)
49. 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)



Estimation of Hand-Interacting Object Poses with Boundary Guidance

Sin-Yu Fu and I.-Chen Lin^(✉) 

College of Computer Science, National Yang Ming Chiao Tung University,
Hsinchu, Taiwan
ichenlin@cs.nycu.edu.tw

Abstract. Estimating poses of objects that interact with hands is a key task for tangible user interface. It is highly challenging due to its inheritance of self- and mutual occlusion. Previous approaches often predict 2D object keypoints from features to establish 2D-3D correspondence during object pose estimation. However, the features for the object and hand are usually intermixed and lead to unreliable output keypoints and inaccurate object pose estimation. To address this issue, we propose a novel Boundary-guided Network (BG-Net). This network takes two cooperative branches for the object and hand. It can effectively capture the object region and utilizes the region as guidance to narrow down the area for keypoint searching. Additionally, we introduce an efficient and effective loss function, min-max boundary distance (MMBD) loss, which restricts the range of estimated keypoint locations. This further benefits the 2D-3D mapping. Experiments demonstrate that the proposed model outperforms related state of the arts for object pose estimation in multiple interactive hand-object benchmarks.

Keywords: Object 6D pose estimation · Hand posture · Region-aware framework

1 Introduction

The interplay between hands and objects is one of the most frequent actions conducted by human beings, wherein the interactions are affected not only by the hand postures but also by those of target objects. Hence, estimating hand-object poses can help understanding human actions. For the emerging tangible interface and augmented reality, accurately estimating poses of objects that interact with hands is the key issue since such systems generate visual feedback according to estimated 6D object poses (three dimensional rotations and translations, respectively) [8, 19, 21].

To estimate 6D object posture from a single image, several approaches adopt fusing features from a RGB-D image [2, 3]. Since it is easier to access RGB images,

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-78447-7_9.

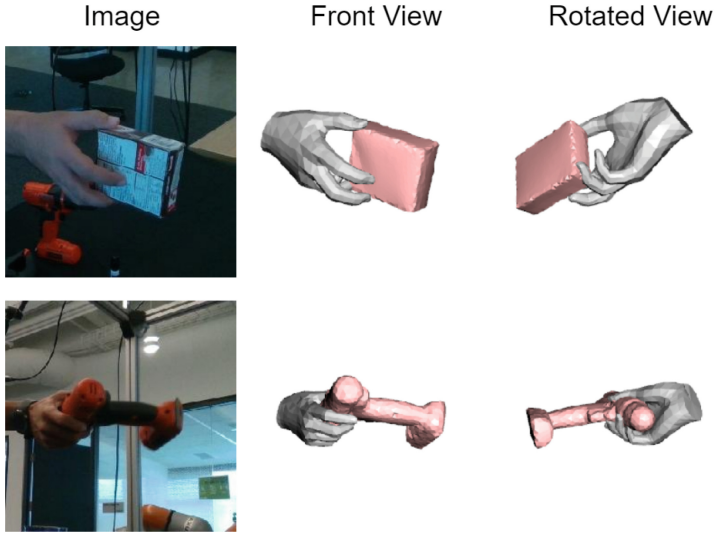


Fig. 1. Object and hand poses estimated by the proposed method from monocular RGB images, respectively.

recent work pays attention on object pose estimation from a RGB image, and the 3D models of target objects are usually available. One strategy [14, 20] is to directly regress object poses, for instance, to learn a prediction model that can map an input image to the corresponding 6D poses. Although such methods are quite effective, they do not fully leverage the projective geometry of known 3D object models.

Another strategy [9, 10, 15, 17, 18] makes use of keypoints of objects. During the inference time, these methods predict the object keypoint locations within the input image. After the 2D keypoint locations are associated with 3D ones, a Perspective-n-Point (PnP) algorithm can be employed to estimate the 6D object pose from these 2D-to-3D correspondences. While recent keypoint-based methods moderately tolerate partial occlusion, their performance usually becomes unstable when the target object is grabbed. When a user holds an object with her (or his) hand, features in the occluded regions often deviate significantly from the object characteristics. Under such circumstances, it is challenging to determine the object boundary and geometric shape, and thereby the accuracy of the output poses degrades. Several recent methods [12, 13] notice the challenge of estimating hand-interacting object poses and take hand-object correlations into account. We found that there is competition between hand and object features when they utilize a single backbone to extract features and keep them in the same space.

To address the aforementioned issues, we propose a novel Boundary-Guided Network (BG-Net) to estimate 6D poses of an object that is interacting with a human hand (Fig. 1). BG-Net capitalizes on features of the object and hand and learns their correlations. Our network is designed to comprise two branches: one focusing on the object and the other dedicated to the hand. This dual-stream design can avoid feature competition and extract distinctive object and hand

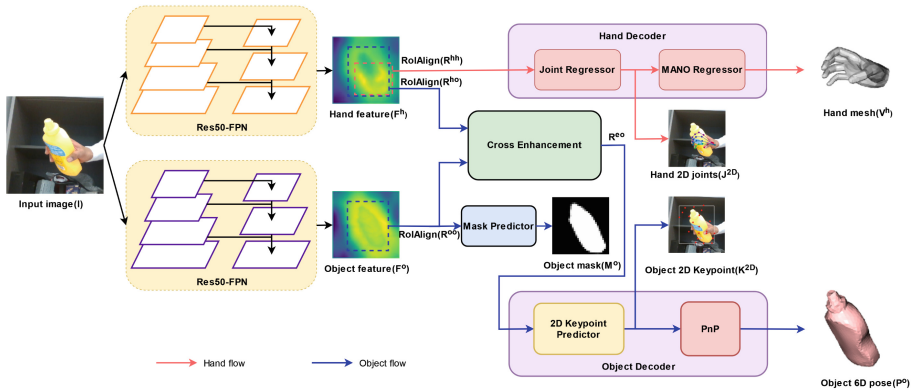


Fig. 2. Overview of the proposed BG-Net.

features. To mitigate the uncertainties in prediction, we predict the object mask as guidance and enable the network to regress object keypoints from promising regions.

By adopting attention mechanism [16] with the guided object region, our model learns the correlation of interactions between each pixel of the hand and object with less ambiguity. Even when object areas are partially occluded by the hand during interaction, our framework can still gain additional cues from joint features. Consequently, the posture of the hand can assist in inferring the distribution of object keypoints during occlusion.

Furthermore, given the potential interference from the image background, we observed that when the object features lack clarity, object keypoints tended to gather within the interior of the object and make the following PnP algorithm difficult to estimate adequate poses. Thus, we introduce a novel loss function, *min-max boundary distance* (MMBD) loss. This loss function compels the outermost 2D keypoints to align with the object bounding box, and therefore enhances the reliability of 2D keypoints, even in scenarios where the object is seriously occluded. To verify the effectiveness of our proposed method, we conducted multiple experiments on two popular hand-object interaction datasets: HO3D [4] and Dex-YCB [1]. Experiments demonstrate that our proposed framework reach state-of-the-art performance for pose estimation of hand-interacting objects.

In summary, our contributions include:

- A new framework BG-Net for hand-interacting object pose estimation is proposed. It utilizes object amodal masks as guidance and directs the network attention toward crucial regions. This approach enables better delineation of the geometric shape of the object and leads to precise keypoint prediction.
- Our proposed MMBD loss, aligning the outermost keypoint 2D coordinates to the projected object keypoint bounding box, can effectively reduce the prediction errors caused by occlusions and enhance the accuracy of poses.

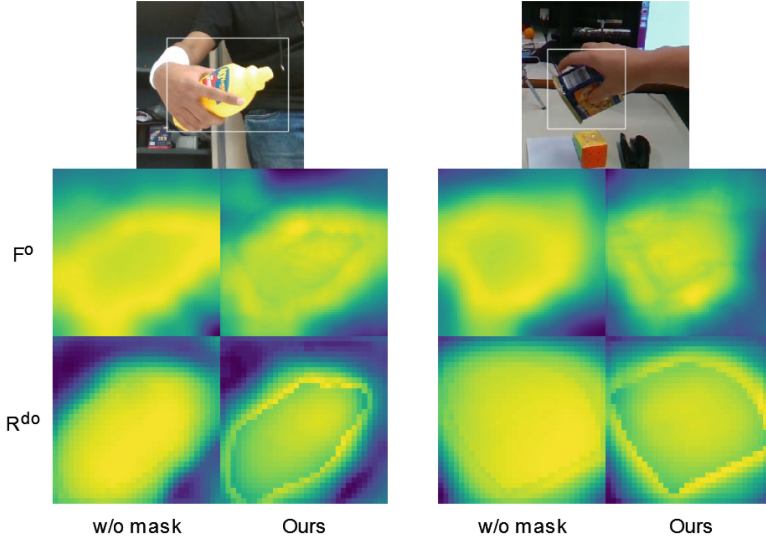


Fig. 3. Visualization of the feature maps F^o , the object feature maps from our backbone, and R^{do} , the intermediate feature maps within the object decoder. R^{do} is the last feature map before the 2D object keypoint regression. With amodal mask prediction (right columns), our network improves its capability of extracting the boundary of the object.

2 Methodology

As illustrated in Fig. 2, our BG-Net consists of two branches to predict hand and object pose respectively. Each branch utilizes its own backbone to extract features, and the object branch includes an additional mask predictor to learn the object contour. Subsequently, the cross enhancement module leverages hand information to provide more cues for occluded object regions. Afterward, the object 2D keypoints K^{2D} is estimated by 2D keypoint predictor, in which the proposed MMBD loss and other loss functions benefit the 2D keypoint alignment during training. In parallel, the 2D joint locations J^{2D} are estimated by the joint regressor. Finally, the hand and object decoders output the 3D hand mesh V^h and the 6D object pose P^o according to 2D joints and keypoints, respectively. The following sections explain each component and the loss functions we used. To ease the explanation, we use F to denote feature maps that contain the same region as the input image, and the superscript h and o represents hand and object, respectively. R denotes feature maps that are cropped and resized after RoIAlign [6]. The first and second superscripts of R denote the source and cropped region, respectively. For example, R^{ho} denotes the feature map cropped from hand feature map F^h and its cropped region is aligned with the predicted object region.

2.1 Backbone and Mask Predictor

As mentioned in the introduction, previous methods [12, 13] used a single-stream backbone to extract both hand and object features. We found that they might compete with each other during feature learning, and it lessens the distinction of these features. As a result, given an RGB image $I \in \mathbb{R}^{256 \times 256 \times 3}$, we employ two separate ResNet-50 models [7] to extract hand and object features. Two distinct Feature Pyramid Networks (FPN) [11] are utilized to fuse the output features from multiple levels within each branch individually. The extracted features for hand and object are denoted as $F^h \in \mathbb{R}^{64 \times 64 \times 256}$ and $F^o \in \mathbb{R}^{64 \times 64 \times 256}$. With our dual-branch design, when the hand and object are partially occluded by each other, the respective network can still correctly acquire information from the regions relevant to their estimation target.

As shown in the middle of Fig. 2, after FPN, we obtain $R^{hh} \in \mathbb{R}^{32 \times 32 \times 256}$ from F^h by RoIAlign according to the hand bounding box. We apply a similar operation to obtain R^{oo} and R^{ho} from F^o and F^h according to the object bounding box. R^{ho} , hand-to-object feature, is used as auxiliary information for object pose estimation in our cross enhancement module in Sect. 2.2.

Although we have obtained a rough region of the object by the object bounding box, there is still a portion of area, between the object and the boundary of the given bounding box, belongs to the background. This background area can still disturb the pose prediction. Hence, we introduce a mask predictor that not only outputs the visible part of the object but also predicts occluded areas caused by interactions between hand and object.

Specifically, we utilize R^{oo} as input for the mask predictor, which includes four convolutional layers and a sigmoid function to output the object amodal mask M^o . This mask serves a dual purpose: it aids the backbone in focusing on the object and guides subsequent modules to prioritize the object. In other words, such additional prediction compels the feature extractor to acquire adequate features that benefit the visible and occluded area estimation, and that helps our system predict more accurate keypoints around the object boundary. We illustrate how the amodal mask affects the learned features of the object branch in Fig. 3.

2.2 Cross Enhancement

The interaction between hands and objects is highly correlated, allowing visible parts within the image to contribute information to analysis of occluded regions. Previous research [12, 13], has yielded promising results by applying attention mechanisms to enhance object features. However, they employed features extracted from the object bounding box (blue box in Fig. 5.a) as query and intersecting areas between the hand and object (green box in Fig. 5.a) as key and value for the attention module.

Such a design has two limitations. Firstly, when hands and objects do not overlap, this module fails to produce meaningful learning outcomes. Secondly, due to the permutation-invariance property of Transformer [16], typical

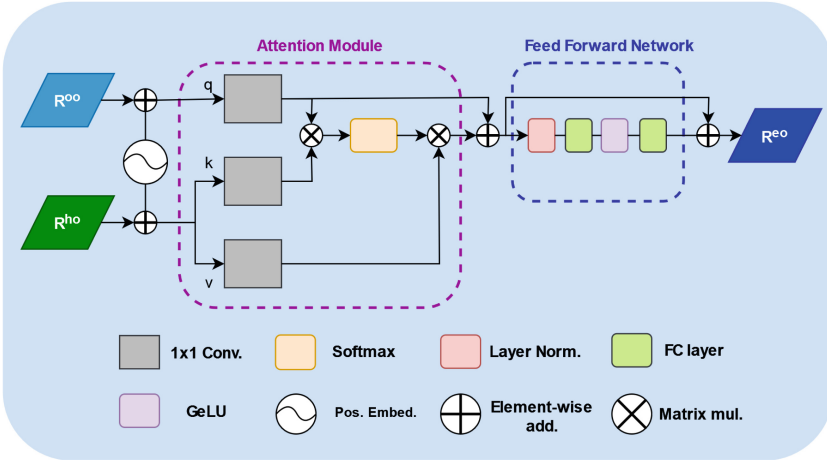


Fig. 4. The structure of our cross enhancement module. This module utilizes the object features R^{oo} (query) to identify its correlation with hand features over the object region R^{ho} (key and value) and outputs the enhanced features R^{eo} accordingly.

approaches often incorporate positional embeddings to retain spatial information. Nonetheless, the aforementioned methods [12,13] treated the overlapped regions as key and value (Fig. 5.f), which mostly do not align with the query (Fig. 5.d) size generated by the object bounding box. Such spatial misalignment hindered the use of positional embeddings.

In our paper, according to an identical object bounding box, we extract and align features regarding objects, R^{ho} (Fig. 5.e) and R^{oo} (Fig. 5.d) from hand F^h (Fig. 5.b) and object F^o (Fig. 5.c) features, respectively. This enables our module to persistently serve as a self-attention module even when there are no interactions between hands and objects. Additionally, our query, key, and value are situated in the same spatial domain by this strategy, and it allows us to add positional embeddings and ensures that the process of computing attention scores maintains spatial relationships. The illustration of cross enhancement is shown in Fig. 4.

Specifically, we add learnable positional embeddings to R^{ho} and R^{oo} and employ three separate 1×1 convolutions to derive query q , key k , and value v from R^{oo} and R^{ho} . They are then fed into a multi-head attention module following a feed-forward network, and finally we can output the enhanced object features $R^{eo} \in \mathbb{R}^{32 \times 32 \times 256}$.

2.3 Min-Max Boundary Distance Loss

Based on the cross-enhanced features, our object decoder then predicts projected 2D keypoints of an object. Afterward, the 6D object pose can be estimated from 2D keypoints by a PnP algorithm. Figure 6 show the defined keypoints on the 3D bounding box of an object (object keypoint amount, $N^o = 21$ in our case).

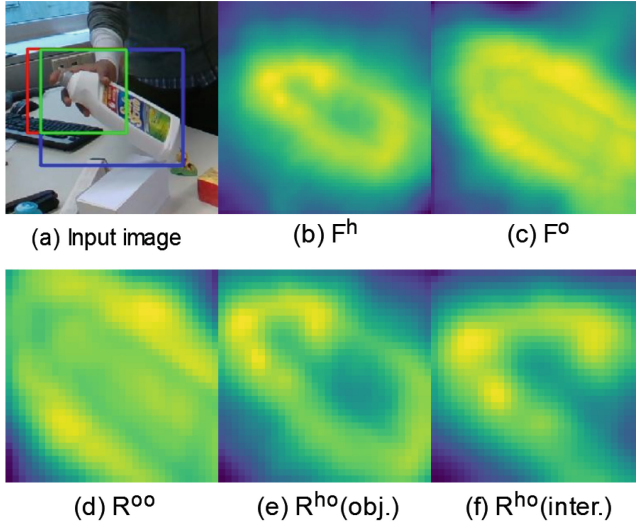


Fig. 5. Illustration of features. (a) Bounding boxes for the hand, hand-object overlap and object are in red, green, and blue, respectively. (b) and (c) Feature maps for hand and object branches after FPN. (d) Features from F^o after RoIAlign according to the blue box in (a). (e) and (f) Features from F^h after RoIAlign according to the blue and green boxes in (a). (Color figure online)

During early trials, we observed that the estimated locations of prominent, especially outermost, keypoints tend to shrink toward the object center, as shown in Fig. 7(c). Even with L2 keypoint distance loss, the model took a conservative way to fit in with various cases, including occlusion. Based on these gathered keypoints, the following PnP method then predicts a farther 3D location for the object. If we directly take object depth as a depth loss, we have to incorporate PnP computation into the network and lose the flexibility of our framework.

Hence, we propose a Min-Max Boundary Distance (MMBD) loss based on projected keypoints to effectively correct the shrunk keypoint problem. This novel loss compares the bounding boxes of projected keypoints. The objective of this loss function is to encourage the outermost predicted 2D keypoints to align with the ground-truth bounding box of projected keypoints. The MMBD loss \mathcal{L}_{MMBD} is formulated as:

$$\mathcal{L}_{MMBD} = \sum_{s \in S} (\min_{k \in K} \|k_x - s_x\|_1 + \min_{k \in K} \|k_y - s_y\|_1), \quad (1)$$

where S includes the coordinate of the top-left corner and bottom-right corner of the 2D object bounding box, K indicated the N^o keypoints. The subscript x and y denote the x coordinate and y coordinate respectively. The loss sums the distances between the four edges of the ground-truth bounding box projection and their closest predicted 2D keypoints.

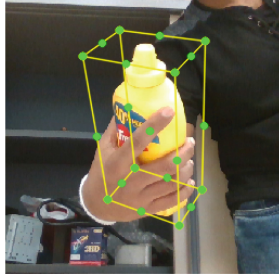


Fig. 6. Visualization of keypoints of an object, including eight corners, twelve midpoint on edges, and one central point of the 3D object bounding box.

As shown in Fig. 7, in the images without using MMBD loss, the outermost keypoints are not aligned with the bounding box and the error of estimated depth is large. By contrast, with the MMBD loss, it can be observed that the outermost keypoints are pulled toward the bounding box. This, in turn, enhances our object pose estimation and reduces the error of the output object depth.

2.4 Decoder and Overall Loss Functions

Our hand and object decoder share the same architecture as previous works [12, 13], except that we employ three residual blocks instead of six convolutional blocks in the object decoder to better retain features learned from preceding boundary-guided processes and preserve the contours of the object as shown in Fig. 3.

Besides the MMBD loss mentioned above, multiple loss functions are applied in our framework during training. We apply the binary cross entropy loss \mathcal{L}_{BCE} for our object mask M^o :

$$\mathcal{L}_{mask} = \mathcal{L}_{BCE}(M^o, \hat{M}^o), \quad (2)$$

where \hat{M}^o is the corresponding ground-truth amodal mask. We briefly describe the remaining loss used for hand and object supervision since they are the same as [12, 13]. The overall hand and object loss are as below:

$$\begin{aligned} \mathcal{L}_{hand} = & \alpha_{mano} \mathcal{L}_{mano} + \alpha_{J2D} \mathcal{L}_{J2D} + \\ & \alpha_{J3D} \mathcal{L}_{J3D} + \alpha_{V^h} \mathcal{L}_{V^h}, \end{aligned} \quad (3)$$

$$\begin{aligned} \mathcal{L}_{obj} = & \alpha_{MMBD} \mathcal{L}_{MMBD} + \alpha_{p2d} \mathcal{L}_{p2d} + \\ & \alpha_{conf} \mathcal{L}_{conf} + \alpha_{mask} \mathcal{L}_{mask}, \end{aligned} \quad (4)$$

where \mathcal{L}_{mano} denotes the L2 loss for MANO parameters θ and β . \mathcal{L}_{J2D} is the L2 loss for 2D joint predictions. \mathcal{L}_{J3D} and \mathcal{L}_{V^h} are the L2 loss for 3D joints and 3D hand mesh. \mathcal{L}_{p2d} and \mathcal{L}_{conf} are the L1 loss for 2D object keypoints and their confidence scores.

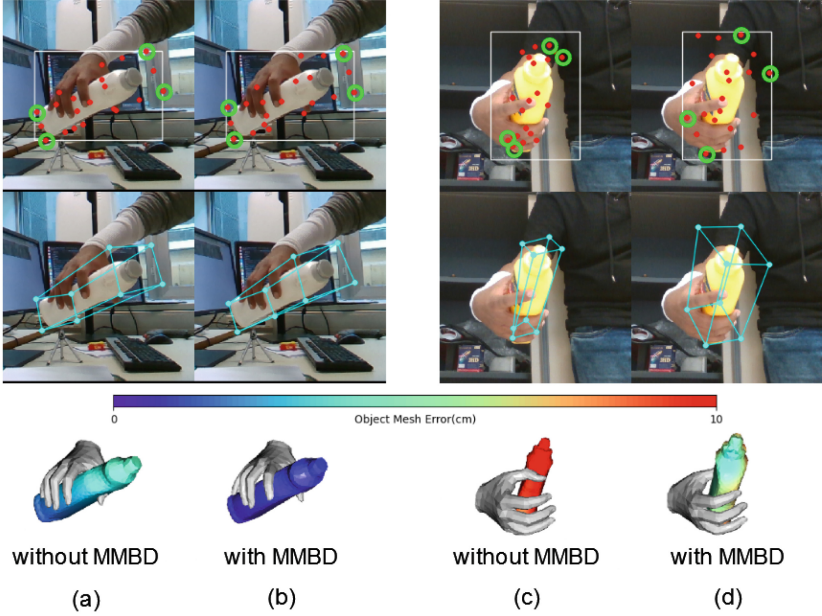


Fig. 7. Visualization of the effect of MMBD loss. (a)(c) are outputs without MMBD; (b)(d) are the corresponding outputs with MMBD loss. Red dots and green circles indicate the predicted object keypoints and the outermost ones. Our MMBD loss significantly assists in aligning the outermost keypoints along the boundaries. (Color figure online)

α_{mano} , $\alpha_{J^{2D}}$, $\alpha_{J^{3D}}$, α_{V^h} , α_{MMBD} , α_{p2d} , α_{conf} and α_{mask} are hyper-parameters for balancing each loss. (In our case, two terms of weights for MANO pose and shape are 10 and 10^{-1} . The others are 10^2 , 10^4 , 10^4 , 20, 500, 10^2 , 10^2 , respectively.) Finally, our total loss function is defined as:

$$\mathcal{L}_{total} = \mathcal{L}_{hand} + \mathcal{L}_{obj}. \quad (5)$$

3 Experiments

3.1 Datasets and Evaluation Metrics

We adopted two popularly used hand-object datasets, HO3D [4] and DexYCB [1] for our experiments. HO3D consists of 66,000 training images and 11,000 testing images, covering 10 different objects. DexYCB is a more challenging dataset, encompassing 582,000 images and featuring interactions with 21 distinct objects. This dataset presents a greater diversity of interactions between hands and objects. We employed the official **s0** split to partition the dataset into training and testing sets. We followed the evaluation metrics applied in HFL-Net [12] for fair comparisons. For our primary task, 6D object pose estimation, we apply

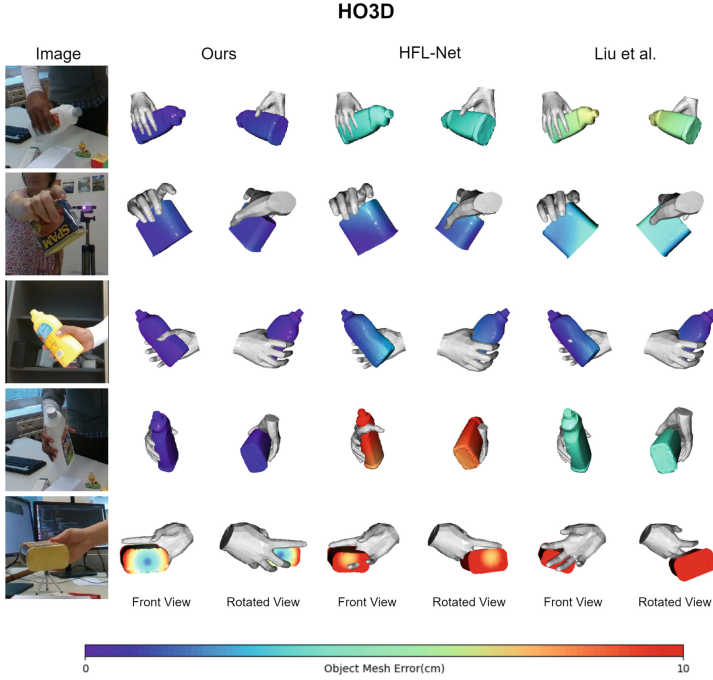


Fig. 8. Qualitative comparison of the proposed BG-Net and state-of-the-art hand-object pose estimation methods [12, 13] on HO3D [4] dataset.

the popular ADD-0.1D. It evaluates the percentage of object 3D vertices error within 10% of the object diameter of the dataset. For the hand pose estimation, besides evaluating the average joint error, joint error with procrustes alignment (PA) is another popular metric. It first aligns the centroids, scales and orientations of two shapes and evaluates the differences.

3.2 Implementation Details

We cropped and resized the input images from the dataset to 256×256 pixels, centered around the midpoint of the hand and object. During training on the HO3D dataset, we employed the Adam optimizer with an initial learning rate of $1e-4$ and a weight decay rate of 0.7 every 10 epochs. We set the batch size as 32 and trained the model with 60 epochs on a single NVIDIA RTX4090 GPU. To augment the data, we utilized techniques such as color jittering, random rotation, translation, and scaling. Please refer to the supplementary document for other details. The codes will be available from the project page of the authors.

Table 1. Comparison with state-of-the-art methods on object pose estimation on HO3D [4] dataset. “avg” denotes the average among all object categories. Our method achieves the best performance on average.

Methods	ADD-0.1D \uparrow			
	cleanser	bottle	can	avg
Liu et al. [13]	<u>88.1</u>	61.9	<u>53.0</u>	67.7
HFL-Net [12]	81.4	87.5	52.2	<u>73.3</u>
Ours	94.7	<u>80.2</u>	65.8	80.2

Table 2. Comparison with state-of-the-art methods on hand pose estimation on HO3D [4] dataset. Even though our goal is object pose estimation, our estimated hand poses are comparable to those of related methods.

Methods	Error(PA) \downarrow		F-score \uparrow	
	Joint	Mesh	F@5	F@15
Liu et al. [13]	10.1	9.7	53.2	95.2
ArtiBoost [22]	11.4	10.9	48.8	94.4
Keypoint Trans. [5]	10.8	–	–	–
HFL-Net [12]	8.9	8.7	57.5	96.5
Ours	9.7	9.7	53.1	95.3

3.3 Comparisons with State-of-the-Art Methods

HO3D. Our work emphasizes 6D object pose estimation in an interactive scenario, and the comparison with state of the arts is shown in Table 1. Our results achieved 80.2% accuracy on ADD-0.1D, surpassing the second-best method by 6.9%. It demonstrates the effectiveness of object pose estimation through our boundary-guided network. Qualitative comparisons are shown in Fig. 8. Even in cases where a large portion of hands or objects are occluded, or when object features are ambiguous, our model generates a more precise object pose compared to that of [12, 13].

Even though the proposed work focuses on hand-interacting object pose estimation, our BG-Net can still estimate accurate hand poses comparable to recent methods as shown in Table 2. Although our approach did not achieve the best performance on hand posture, our method still outperforms Liu et al. [13], which has a similar hand pose estimation structure to ours.

Table 3. Comparison with state-of-the-art methods on Dex-YCB [1] dataset. Our method achieves competitive results with the best approach [12] on hand pose estimation and outperforms the others on object pose estimation by a large margin.

Methods	ADD-0.1D(s) \uparrow	Joint \downarrow	Joint(PA) \downarrow
Liu et al. [13]	29.8	15.27	6.58
HFL-Net [12]	<u>30.2</u>	12.56	5.47
Ours	46.2	<u>12.7</u>	<u>5.53</u>

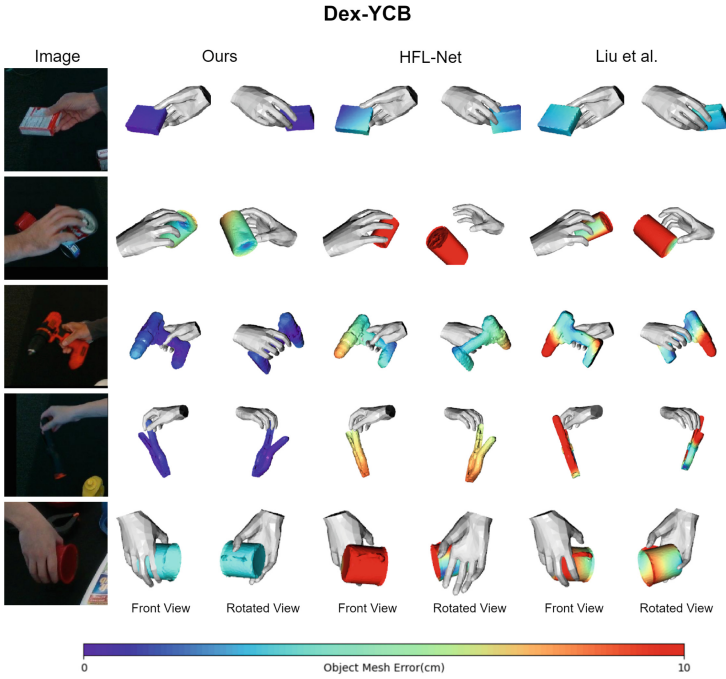


Fig. 9. Qualitative comparison of the proposed BG-Net and state-of-the-art hand-object pose estimation methods [12, 13] on DexYCB [1] dataset.

Dex-YCB. Table 3 summarize results of object and hand pose estimation on Dex-YCB dataset. The errors of joint estimation by our method with and without Procrustes Alignment are 12.7 mm and 5.53 mm, respectively. They are on a par with state-of-the-art approaches. For object pose estimation, our results reach 46.2% on ADD-0.1D(s), substantially outperforming HFL-Net [12] by 16%. We attribute this advance to our double-stream architecture and amodal mask in tackling the challenges of learning from such a diverse object dataset, where twenty one objects are included. Our approach allows the object backbone to concentrate solely on extracting object-specific features, while the mask aids in learning object boundaries, and our MMBD loss helps correct improperly estimated depth of an object. Qualitative comparisons are shown in Fig. 9.

Our framework has shown its advantage of estimating hand-interacting object poses and it employs 59,073,760 trainable parameters, while there are 46,080,659 and 34,480,019 trainable parameters in HFL-Net [12] and Liu et al. [13], respectively.

3.4 Ablation Study

To verify the effectiveness of our proposed methods, we conducted ablation study on the HO3D [4] dataset.

Table 4. Ablation study on the major components and MMBD loss.

Methods	ADD-0.1D \uparrow			
	cleanser	bottle	can	avg
w/o mask	93.3	80.7	59.7	77.6
w/o cross enhance.	93.6	77.0	57.9	76.3
w/o residual blocks	93.2	73.3	60.4	75.7
w/o MMBD loss	92.6	72.9	60.2	75.2
Ours	94.7	80.2	65.8	80.2

Effectiveness of the Major Components and MMBD Loss. As our designed approach mainly focuses on enhancing object pose estimation, we report the ADD-0.1D in Table 4. In the first experiment, we removed the mask predictor. The result indicates that the absence of the mask decreases the accuracy in pose estimation. The visualization in Fig. 3 shows that prediction with the amodal mask accentuates the object boundaries in feature maps. In the second experiment, we removed the cross enhancement module, and no additional information from hand features is provided. It results in a 2.6% performance drop. It manifests that the hand poses can provide useful features for hand-interacting object pose estimation.

For the third experiment, we replaced the three residual blocks in the object decoder with six convolutional layers, similar to [12, 13]. The result reveals that residual blocks play a significant role in preserving previously learned features. They prevent losing the cues provided by the contours of the object mask and clues from hand features. The fourth experiment and Fig. 7 validate the proposed MMBD loss. They show that without MMBD loss, the performance substantially degrades. These experiments demonstrate that the employed components and MMBD loss indeed benefit the pose estimation performance for objects that are partially occluded by a hand.

Effectiveness of Double-Stream Backbone. While related methods [12, 13] took a single-stream backbone, we adapted a double-stream backbone. To verify the effectiveness of our double-stream backbone, we replaced the architecture of our model with a shared ResNet-50 with FPN for both hand branch and object branch while keeping other components unchanged. Table 5 shows that applying our double-stream backbone, combined with the proposed modules and loss functions, provides a 7.8% improvement in object pose estimation compared to a framework adopting the single-stream backbone.

Additionally, there is a 0.3 mm enhancement in average hand joint and mesh errors. This outcome demonstrates that using two separate backbones to learn hand and object features enables an easier learning process for respective targets without interference. The mask predictor also better guides the object backbone in learning object boundaries.

Table 5. Ablation study on single-stream and double-stream architectures.

Methods	ADD-0.1D \uparrow	Joint \downarrow	Mesh \downarrow
Single-stream	72.4	10.0	10.0
Ours	80.2	9.7	9.7

Table 6. Ablation study on different settings for R^{ho} in cross enhancement. “Intersect.” and “object bbox.” denote that we use the hand-object overlapped region (green box in Fig. 5.a) or the object bounding box (blue box in Fig. 5.a) to produce R^{ho} . “Pos.” indicates that positional embeddings are appended.

Methods	ADD-0.1D \uparrow			
	cleanser	bottle	can	avg
intersect.	93.8	74.3	56.5	74.9
intersect. + pos.	93.3	71.0	62.3	75.5
object bbox.	91.8	75.2	62.3	76.4
Ours	94.7	80.2	65.8	80.2

Different Settings for R^{ho} in Cross Enhancement. Table 6 compares the results of using different bounding boxes to produce R^{ho} (Fig. 5.e & Fig. 5.f) in cross enhancement, along with the incorporation of positional embeddings. In the first and second settings, R^{ho} is extracted from the overlapping region of the hand and the object (green box in Fig. 5.a), while the second setting additionally integrates positional embeddings. It can be observed that the incorporation of positional embeddings in such settings merely gains 0.6% improvement on ADD-0.1D. It is worth noting that the performance of the first and second settings is not as good as when we do not employ cross enhancement in our model (the second row in Table 4). This suggests that when there is a spatial inconsistency among the key, value, and query in transformer, attention mechanism does not successfully benefit the model.

By contrast, in the third and fourth settings, R^{ho} is extracted based on the object bounding box (blue box in Fig. 5.a). Compared to the third setting, the fourth setting includes positional embeddings and exhibits a 3.8% enhancement on ADD-0.1D. This underscores the significance of positional embeddings for preserving spatial information, when the key, value, and query share identical space on feature maps.

4 Conclusion

This paper presents the Boundary-Guided Network (BG-Net) for 6D pose estimation of objects interacting with a hand. This framework adapts a double-stream framework to enhance the object and hand feature distinction respectively. In this framework, we estimate and utilize the object amodal mask to

guide the object branch in learning object-specific features and identifying object boundaries for accurate prediction of 2D object keypoints. Moreover, we propose a novel min-max boundary distance (MMBD) loss. It tackles the gathering issue of predicted keypoints and therefore reduces the depth error of the output object pose. Experiments demonstrate that our method surpasses state-of-the-art methods on hand-interacting object pose estimation, and it also achieves comparable performance in hand pose estimation.

References

1. Chao, Y.W., et al.: Dexycb: a benchmark for capturing hand grasping of objects. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9044–9053 (2021)
2. Cheng, Y., et al.: 6D pose estimation with correlation fusion. In: International Conference on Pattern Recognition (ICPR), pp. 2988–2994 (2021)
3. Feng, H., Zhang, L., Yang, X., Liu, Z.: Mixedfusion: 6d object pose estimation from decoupled RGB-depth features. In: International Conference on Pattern Recognition (ICPR), pp. 685–691 (2021)
4. Hampali, S., Rad, M., Oberweger, M., Lepetit, V.: Honnotate: a method for 3d annotation of hand and object poses. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3196–3206 (2020)
5. Hampali, S., Sarkar, S.D., Rad, M., Lepetit, V.: Keypoint transformer: solving joint identification in challenging hands and object interactions for accurate 3D pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11090–11100 (2022)
6. 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)
7. 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)
8. Hsu, W.T., Lin, I.C.: Associating real objects with virtual models for VR interaction. In: SIGGRAPH Asia 2021 Posters, pp. 24:1–2 (2021)
9. Hu, Y., Hugonot, J., Fua, P., Salzmann, M.: Segmentation-driven 6d object pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3385–3394 (2019)
10. Huang, W.L., Hung, C.Y., Lin, I.C.: Confidence-based 6d object pose estimation. *IEEE Trans. Multimed.* **24**, 3025–3035 (2022)
11. 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)
12. Lin, Z., Ding, C., Yao, H., Kuang, Z., Huang, S.: Harmonious feature learning for interactive hand-object pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12989–12998 (2023)
13. Liu, S., Jiang, H., Xu, J., Liu, S., Wang, X.: Semi-supervised 3d hand-object poses estimation with interactions in time. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 14687–14697 (2021)
14. Mo, N., Gan, W., Yokoya, N., Chen, S.: ES6D: a computation efficient and symmetry-aware 6d pose regression framework. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6718–6727 (2022)

15. Peng, S., Liu, Y., Huang, Q., Zhou, X., Bao, H.: PVNet: pixel-wise voting network for 6dof pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4561–4570 (2019)
16. Vaswani, A., et al.: Attention is all you need. *Adv. Neural Inf. Process. Syst.* **30** (2017)
17. Wang, D., Zhou, G., Yan, Y., Chen, H., Chen, Q.: Geopose: dense reconstruction guided 6d object pose estimation with geometric consistency. *IEEE Trans. Multimed.* **24**, 4394–4408 (2021)
18. Wang, G., Manhardt, F., Tombari, F., Ji, X.: GDR-Net: geometry-guided direct regression network for monocular 6d object pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 16611–16621 (2021)
19. Wu, L.C., Lin, I.C., Tsai, M.H.: Augmented reality instruction for object assembly based on markerless tracking. In: Proceedings of the 20th ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games, pp. 95–102. I3D '16, Association for Computing Machinery (2016)
20. Xiang, Y., Schmidt, T., Narayanan, V., Fox, D.: Posecnn: a convolutional neural network for 6d object pose estimation in cluttered scenes. arXiv preprint [arXiv:1711.00199](https://arxiv.org/abs/1711.00199) (2017)
21. Yamaguchi, M., et al.: Video-annotated augmented reality assembly tutorials. In: Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology, pp. 1010–1022. UIST '20, Association for Computing Machinery (2020)
22. Yang, L., et al.: Artiboost: boosting articulated 3d hand-object pose estimation via online exploration and synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2750–2760 (2022)



Annotation-Free Object Detection by Knowledge-Extraction Training From Visual-Language Models

Yasuto Nagase^(✉), Yasunori Babazaki, and Takashi Shibata

NEC Corporation, Kawasaki, Kanagawa, Japan
{yasuto-nagase,y_babazaki,takashi-shibata}@nec.com

Abstract. Modern object detection models often require enormous training images with accurate annotations for each scenario; it is a significant obstacle for actual applications for computer vision. In this paper, we propose a novel framework for training lightweight object detection models without additional manual annotations by inheriting the rich expression power of multiple pre-trained visual-language(VL) models. The key is to obtain elaborate pseudo labels for lightweight model by knowledge-extraction training from multiple VL models, the biases of which are corrected by score correction. We can obtain accurate detection labels without using any prior manual annotations for each image by using novel data augmentation to enhance knowledge extraction from the VL models and pseudo-label integration. In contrast to current semi-supervised and unsupervised approaches for object detection, our proposed framework is immediately applicable to state-of-the-art object detection models and training protocols. Comprehensive experiments on two public datasets demonstrated that our framework is fast and lightweight while maintaining accuracy, surpass supervised models.

Keywords: Visual and Language model · Foundation model · Annotation-free · Object detection

1 Introduction

Modern deep network architectures, large public datasets with accurate manual annotations, and open sources have led to remarkable progress in object detection [4, 9, 27, 32, 48]. However, modern object detection models (hereafter, object detectors) are data-hungry; thus, incurring enormous annotation costs. Each application scenario requires dull data collection and laborious manual annotation (e.g., hundreds to thousands) for each rare object not included in the public dataset. Although unsupervised and semi-supervised learning algorithms for object detection have also been proposed [13, 33, 34, 37, 41, 45], these

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-78447-7_10.

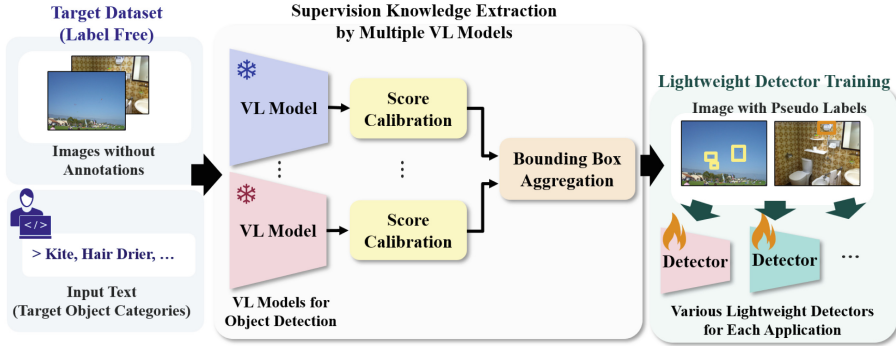


Fig. 1. Overview of proposed framework. Our framework enable to training accurate object detector without manual annotations using VL models for object detection.

algorithms have limitations in architectural scalability and accuracy. The high cost of manual annotation continues to be a significant issue for object detection in practical applications.

In parallel with the progress of object detection, pre-trained visual-language (VL) models are causing a new paradigm shift in various vision tasks such as classification [14, 26], object detection [18, 21, 39], and captioning [16, 17]. These VL models designed for object detection have high detection accuracy and zero-shot capability thanks to their high expressive power by simultaneously training large image-language pairs. In particular, GLIP [18] and Grounding DINO [21] outperform closed-set object detectors and have become standard models for highly accurate open vocabulary object detection.

In compensation for their high expressiveness power, however, these VL models require significant computer resources. The processing speed is also very slow for both training and inference. Therefore, it is infeasible to naively apply these pre-trained VL models to practical applications require real-time processing or application scenarios with massive data processing. The computational cost and low speed also make it difficult to use them for detection-based tasks such as tracking [24, 42]. For such applications, there is a strong demand for a framework to train lightweight object detectors with infinitely low supervision costs while inheriting the strengths of these pre-trained VL models with their high expressive power and detection accuracy.

In this paper, we propose an annotation-free framework for object detector using knowledge extraction training from VL models. Our framework can simultaneously leverage high detection accuracy in rare target objects using text-prompt-driven VL models and lightweight capability thanks to modern closed-set object detectors such as YOLOX [9]. An overview of our framework is shown in Fig. 1. Pseudo-annotation is generated from VL models and the set of unlabeled images. A lightweight object detector is then trained in the existing closed-set object detection manner. The key is to improve the pseudo-annotation quality using score-calibrated VL models for each instance obtained by test-time

augmentation (TTA). We can generate accurate pseudo-annotations without do so manually for arbitrary objects because we can detect the target object using the VL guided by the text prompts. Comprehensive experiments on two public datasets showed that our framework can train an object detector capable of real-time inference with low supervision cost. We also demonstrated that effective training algorithms and data augmentation can achieve detection accuracy comparable to supervised learning.

The contributions of this paper are the followings;

- A simple-yet-effective framework for training lightweight object detectors using VL models.
- Supervised-label generation using multiple score-calibrated VL models and TTA.
- Comprehensive experiments showing that our framework can train an object detector capable of real-time inference with low supervision cost.

2 Related Work

2.1 Object Detection

Object detection, used for localizing objects and recognizing their categories in images, is an essential task in computer vision. It has significantly improved in both accuracy and processing speed since the introduction of deep learning approaches and large-scale datasets.

Convolutional neural networks (CNNs) have led to significant strides in object recognition tasks [9,27]. Different CNN-based architectures for object detection, such as one-stage [9,32] and two-stage detection [19,27], were proposed to tackle such tasks. The availability of large-scale annotated datasets such as Pascal VOC [8], MS COCO [20], and OpenImage [15] has played a pivotal role in effectively training deep neural networks. In recent years, Transformer-based approaches, inspired by their success in natural-language-processing tasks, have gained traction in object detection research [4,48]. Against the backdrop of increasing demand for object detection in industrial settings, such as vehicle and road-sign detection for autonomous driving [46] and human detection for safety monitoring, real-time processing of object detection has been achieved [9,22]. Object detectors that meet both real-time processing and detection performance are thus applied to tasks such as tracking [24,42] and action detection [25,31], becoming essential in computer-vision applications.

Training object detectors for new objects, however, requires annotating large datasets with fine-grained object bounding boxes, and the annotation work is time-consuming. Our framework, unlike previous frameworks, can train object detectors capable of detecting objects from arbitrary classes without annotations and without affecting existing architectures, while maintaining real-time processing capability. We verified the effectiveness of our framework for several object detectors that have different architectures.

2.2 Visual-Language Models for Object Detection

VL models, including CLIP [26] and ALIGN [14], trained on a large-scale dataset of image-text pairs, have achieved remarkable zero-shot performance in image classification, paving the way for advancements in image recognition. The paradigm of achieving zero-shot recognition by grounding images and text has been extended to object detection, enabling the detection of arbitrary objects on the basis of input text without the need for training [10, 18, 21, 39, 40]. Despite being in a zero-shot setting, VL models for object detection demonstrate comparable performance to fully supervised models on well-established object detection benchmarks [18, 21, 39, 40]. The representative VL models for object detection, GLIP [18, 40] and Grounding DINO [21] demonstrate the effectiveness of vision-language modality fusion at middle layers, highlighting the importance of simultaneous processing of images and language.

These object detection VL models can recognize arbitrary objects guided by text even if it is not in the training data; however, due to the large size of both image and language models, they incur higher processing costs compared to conventional object detectors and lack real-time processing capability. Our framework aims to construct light-weight object detectors that can detect objects from any class with the help of VL models' high-quality labeling.

2.3 Reducing Annotation Costs in Object Detection Training

There has been growing interest in developing methods for achieving high-performance object detection while simultaneously reducing annotation costs. In this context, unsupervised, weakly-supervised, and semi-supervised learning make significant contributions.

Unsupervised object detection aims to identify objects in images without any manually labeled data, showcasing remarkable improvements [13, 33, 34]. While learning without labeled data in unsupervised object detection is similar to the setting in our study, previous methods fail to identify the categories of objects. They also still face challenges in capturing fine-grained object details and achieving high detection accuracy due to the absence of labeled data.

Weakly supervised object detection uses only coarse annotations, such as image-level labels or bounding boxes, instead of precise object annotations for each instance, to train object detectors [2, 12, 28, 29, 35, 38, 43]. Although these methods can significantly reduce annotation costs, the accuracy gap with full supervised methods is significant when the amount of available data is limited. These approaches cannot be adapted to current object detectors due to their reliance on specialized designs.

Semi-supervised object detection involves a combination of labeled and unlabeled data for training [5, 11, 37, 41, 44, 45]. In this field of research, special designs are being explored to enhance the performance of object detectors and investigate improved learning strategies. However, these designs may not be adaptable to other object detectors, potentially limiting their applicability. While model-agnostic semi-supervised learning methods have also been proposed [7], there is

Table 1. Comparison between our framework and previous method for training object detectors.

	Annotation free	Accuracy in practical scenarios	Light weight (real-time flops)	Category identification
Unsupervised	✓	×	✓	×
Semi-supervised	×	✓	✓	✓
Weakly-supervised	×	×	✓	✓
VL model	✓	✓	×	✓
Proposed	✓	✓	✓	✓

a significant gap from fully supervised performance under conditions of limited labeled data.

Table 1 shows the main comparisons between our framework and those described above. Our framework, designed to be independent of model architectures, is applicable to lightweight object detectors used in various applications, thus offering versatility. Our framework also enables the training of object detectors capable of identifying arbitrary categories without manually annotations, thus demonstrating a very high adaptability to practical applications.

3 Method

Our goal is to quickly obtain an accurate-and-lightweight object detector without incurring additional annotation costs for each scenario. We propose a lightweight detector training framework that can leverages the rich knowledge of VL models. Our framework consists of the following two phases: 1) **supervision knowledge extraction from multiple VL models**: Inputting unlabeled data into the VL models for object detection, and creating teacher data by utilizing the inferred results as pseudo-labels. 2) **lightweight detector training**: Training a lightweight object detector on the basis of the obtained training data.

The key is to extract and integrate the knowledge from multiple VL models while inheriting their strengths by using score calibration. With the proposed framework, we can exploit the rich knowledge extracted from multiple VL models to various modern object detectors by clearly separating the steps of the framework, i.e., knowledge extraction and lightweight model training. In the following sections, we provide detailed explanations for each phase.

3.1 Supervision Knowledge Extraction From Multiple VL Models

In general, VL models are capable of detecting arbitrary objects by specifying them through prompts. In our framework, we first feed a set of unlabeled images into the VL model along with prompts specifying the category names of the objects to be detected. The VL model outputs the bounding boxes and confidence scores for each object corresponding to these prompts' category names.

Let \mathbf{x} , $F_j(\cdot)$, and $\mathbf{T} = \{T^l\}_{l=1}^L$ be an input image, the j -th VL model, and a set of input texts for the prompt, respectively. Here, j , l , and L are the index for

Table 2. Examples of differences in precision and recall by classes in two foundation models: GLIP-Large (GLIP) and Grounding DINO swin-T (GDINO). We used detection results from MS COCO dataset and evaluated COCO metrics introduced in Sect. 4.1.

Category	AP		AR	
	GLIP [18]	GDINO [21]	GLIP [18]	GDINO [21]
Kite	46.8	51.6	66.3	68.7
Vase	33.0	37.5	66.4	67.4
Hair drier	43.0	17.6	54.5	57.3
Toothbrush	47.9	46.4	67.0	64.4

the VL model, that of the category, and the number of categories, respectively. The output of the j -th VL model F_j is obtained from the input image \mathbf{x} as follows:

$$\{\mathbf{y}_i^j\}_{i=1}^{N_j} = F_j(\mathbf{x}), \quad (1)$$

where i and N_j are the index for the object detected with the j -th VL model F_j and the number of detected objects. Here, $\mathbf{y}_i^j = (c_i^j, p_i^j, \mathbf{b}_i^j)$ is the i -th detected object, where c_i^j , p_i^j , and \mathbf{b}_i^j are the category label, confidence score, and bounding-box position, respectively.

The generation of training data with reliable annotations is vital for obtaining highly accurate object detection. To this end, we introduce the following two approaches, i.e., multiple VL model ensemble and TTA, into our framework to generate more accurate teacher data.

Multiple VL Model Ensemble with Score Calibration. The proposed framework uses multiple pre-trained VL models in parallel to generate annotation data. The proposed method can improve the robustness of the output results by ensembling multiple VL models. As an example of the sensitivity of VL models against categories, we show the differences in detection rates of some categories for GLIP and Grounding DINO¹ in Table 2. We can see that there are variations in detection accuracy across different categories. This suggests that ensemble fusion can combine these detection results to generate even more accurate training data. A more detailed discussion is provided in the supplementary material.

A naive approach to improve the accuracy of output results is ensemble fusion based on non-maximum suppression (NMS) [3, 23, 30]. However, applying integration algorithms such as NMS to the outputs of multiple VL models using their inherent confidence scores. This naive approach may unfairly favor VL models, which produces high confidence scores. This selection bias often arises because multiple VL models, trained on different data or architectures, define confidence

¹ Unless otherwise specified, we utilize the GLIP-Large and Grounding DINO Swin-T models, respectively.

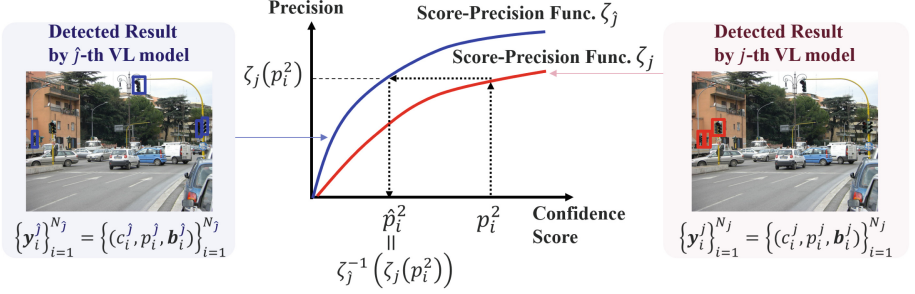


Fig. 2. Overview of the proposed calibration method in our framework.

score scales differently. Figure 3 illustrates an example of the difference in score scales among VL models when reflecting them in precision curves. We can see that GLIP tends to output higher scores than Grounding DINO.

To align the score scale between VL models in the proposed framework, we introduce the following calibration algorithm. We first statistically obtain a score-precision function $\zeta_j(p)$ on the basis of a small dataset separated from the target training dataset in the j -th VL model. For simplicity, we represent this function using a look-up table approach, and the interpolated values are linearly interpolated. As shown in Fig. 2, the score calibration from the j -th VL model to the \tilde{j} -th VL model is formally expressed using the function ζ_j and inverse function $\zeta_{\tilde{j}}^{-1}$, which is given by

$$\tilde{p}_i^j = \phi_{j \rightarrow \tilde{j}}(p_i^j) = \zeta_{\tilde{j}}^{-1}(\zeta_j(p_i^j)), \quad (2)$$

where $\phi_{j \rightarrow \tilde{j}} = \zeta_{\tilde{j}}^{-1}(\zeta_j(\cdot))$ is the calibration function for the j -th VL model². After the calibration, the output of the j -th VL model is given as $\tilde{\mathbf{y}}_i^j = (c_i^j, \tilde{p}_i^j, \mathbf{b}_i^j)$. Note that the score-precision function is robust to data-domain changes, as described in Sect. 4.3, because we use only the average relationship between score and precision.

The results before and after score calibration are shown in Fig. 3. First, we set GLIP as the target VL model and create a score-conversion look-up table using the MS COCO val subset. The results of the Grounding DINO, as shown in Fig. 3 (a), are calibrated using the created table, resulting in Fig. 3 (b). By comparing the results, we can see that the score scales of the two models are calibrated and transformed to exhibit the same score-output tendency. This enables us to suppress the effect of score-scale differences during the subsequent integration. Note that, even though the calibration table was created using the MS COCO dataset, similar effectiveness can be obtained using the PASCAL VOC dataset. The detailed discussion for the effectiveness of the proposed score calibration is discussed in Sect. 4.4.

² Note that, if \tilde{j} and j are the same, the uncalibrated score (i.e., raw score) is used because it is an identity function, i.e., $\phi_{j \rightarrow j} = \zeta_j^{-1}(\zeta_j(\cdot)) = I(\cdot)$.

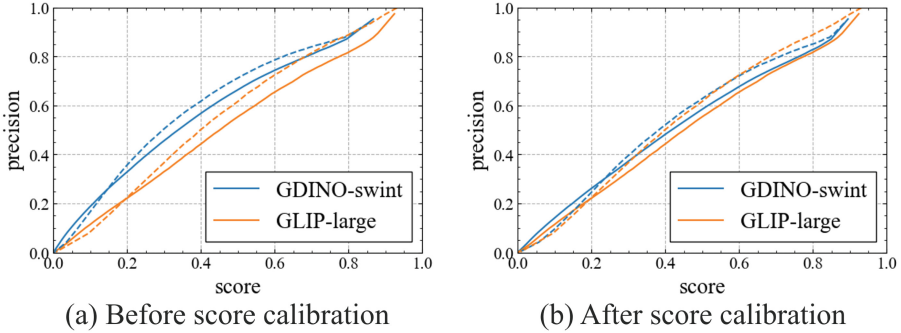


Fig. 3. Comparison of before and after score calibration. In both graphs, solid lines and dashed lines are training subset of MS COCO and PASCAL VOC results, respectively. GDINO stands for Grounding DINO. Results (b) indicate that calibration performed well on both dataset.

By using the calibrated scores obtained above and aggregating the bounding-boxes using NMS manner, we can deal with both confidence score without score scale bias; therefore higher quality pseudo-labels can be obtained. The bounding-boxes aggregation function ψ is formally given by

$$\{\hat{\mathbf{y}}_i\}_{i=1}^N = \psi\left(\left\{\left\{\tilde{\mathbf{y}}_i^j\right\}_{i=1}^{N_j}\right\}_{j=1}^J\right) = \psi\left(\left\{\tilde{\mathbf{y}}_i^1\right\}_{i=1}^{N_1}, \dots, \left\{\tilde{\mathbf{y}}_i^J\right\}_{i=1}^{N_J}\right), \quad (3)$$

where N is the number of object for the input image x after aggregation by NMS manner. Finally, we obtain the merged labels $\hat{\mathbf{y}}_i = (\hat{c}_i, \hat{p}_i, \hat{\mathbf{b}}_i)$. A more detailed discussion of bounding box aggregation is provided in the supplementary material.

Test-Time Augmentation. TTA can improve accuracy by executing image-transformation extensions to the data during model evaluation. As suggested by [40], by executing multi-scale data extension at the time of base model output, further accuracy improvement can be achieved. Therefore, this is also applied during the output of the base model to enhance the quality of the teacher data. The actual effectiveness of implementing TTA is discussed in more detail in Sect. 4.4.

3.2 Lightweight-Detector Training

Finally, a lightweight detector is trained using generated pseudo-labels mentioned in the previous section as teacher data. Unlike many unsupervised or weakly supervised methods, our framework does not restrict the type of detector or model architecture used, enabling users to choose a model that suits their task freely. There are also no constraints on the training method, enabling users to

execute data augmentation or fine-tuning of models as needed. In our framework, by explicitly separating these steps as the supervision-knowledge extraction and lightweight-detector training, we can simultaneously leverage the strengths of the VL models and modern network architectures and sophisticated training for closed-object detection.

4 Experiments

We describe the experimental settings then the three experiments we conducted to demonstrate the effectiveness of the proposed framework; i) the effectiveness of training-data generation by using multiple VL models, ii) performance of lightweight-detector training using the generated pseudo labels, and iii) ablation study and analysis of the proposed framework.

4.1 Settings

Datasets. We evaluated our method and existing methods on MS COCO [20] and Pascal VOC [8]. MS COCO is a large dataset containing 80 categories of objects and composed of train2017, val2017, and test-dev 2017 subsets respectively containing 118,287, 5,000 and 20,288 images. PASCAL VOC consists of VOC2007 and VOC2012 and contains objects in 20 categories. The 16,551 images in the trainval subsets of VOC2007 and VOC2012 are used for training, and the 4952 images in the test subset of VOC2007 are used for testing. We report the performance on the MS COCO datasets following the standard COCO metric, which includes several metrics, such as average precision (AP) and average recall (AR) with varying intersection-over-union (IoU) thresholds³.

Models and Implementation Details. We used GLIP-L [18] and Grounding DINO swin-T [21] as the VL models for supervision-knowledge extraction in our framework. These models, including the detector head, are publicly available. These pre-trained models have not been trained on Pascal VOC or MS COCO datasets. The confidence-score threshold after the score calibration is usually set around 0.4 to 0.6. In our experiments, we set it to 0.4 as pseudo labels for all cases. We used GLIP and Grounding DINO to create pseudo labels by inputting images and category names from publicly available datasets for accuracy validation. Note that, These VL models are used for only inference. Therefore, our framework does not require such an enormous computational cost during the training phase.

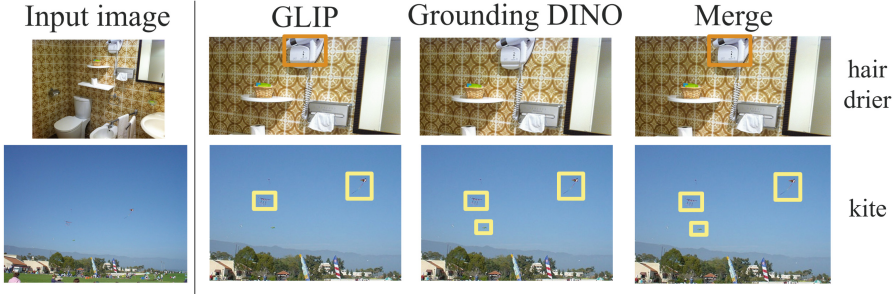
For object detection, we employ the well-known models Faster R-CNN [27], YOLOX [9], and CO-DINO [48]. These models are considered de facto standards. A more details of the detectors is provided in the supplementary material.

The training and evaluation of these models were implemented using MMDetection [6] ver 3.2.0, and all hyperparameters and training schedules of the models referred to default settings.

³ Unless otherwise specified, AP and AR indicate the IoU threshold range was set to 0.50 to 0.95, and the maximum number of objects was set to 100.

Table 3. Results of VL models’ knowledge extraction performance on training datasets from Pascal VOC trainval and MS COCO train2017.

	Pascal VOC		MS COCO	
	AP	AR	AP	AR
GLIP-Large [18]	74.6	86.8	51.4	69.2
Grounding DINO swin-T [21]	65.0	89.3	48.4	73.6
Merged (Ours)	75.6	89.3	52.0	74.3

**Fig. 4.** Examples merged results by the proposed framework.

4.2 Training Data Generation by Using Multiple VL Models

To evaluate whether the VL models can generate accurate annotated data, we conducted inference and evaluation on the training images of the dataset using GLIP [18] and Grounding DINO [21]. The experimental results are listed in Table 3. GLIP demonstrated better precision, while Grounding DINO exhibited better recall. By integrating them into the proposed framework, as described

Table 4. Comparison of accuracy of detector trained with proposed framework and conventional unsupervised method. * indicates that it was evaluated on VOC2007 trainval subset. AP^{50} means average precision when IoU threshold was 0.5. *All results were calculated on class-agnostic setting.*

Model	Pascal VOC			MS COCO		
	AP	AP^{50}	AR	AP	AP^{50}	AR
DETReg [1]	–	–	–	1.0	3.1	12.7
Exemplar-FreeSOLO [13]	12.6*	26.8*	–	12.6	17.9	17.9
CutLER [34]	20.2	36.9	44.3	12.3	21.9	32.7
Ours w/ Faster R-CNN	50.2	82.8	63.8	37.3	61.1	57.3
Ours w/ YOLOX	57.3	76.7	68.1	44.4	64.1	63.1
Ours w/ CO-DINO	71.7	92.9	83.5	51.6	70.9	70.9

in Sect. 3.1, we can obtain better results compared with using the VL models individually, as they complement each other in selecting the correct bounding boxes. Examples for qualitative evaluation are shown in Fig. 4. As shown in Table 2, for *hair drier* and *kite* classes, there are instances in which Groudning DINO and GLIP failed to detect objects individually. In contrast, after the complementary integration into the proposed framework, however, correct detection results could be incorporated into the pseudo labels.

4.3 Lightweight-Detector Training Using Generated Pseudo-Labels

The results of training several detectors using the generated training data mentioned in the previous section are listed in Table 4 and Table 5. In addition to the results of learning under fully supervised conditions, the results of conventional methods divided for each annotation type are also presented for comparison. Note that AP and AR significantly improved compared with conventional unsupervised methods. Conventional unsupervised methods lack background knowledge of target labels, leading to frequent non-detection or misdetection. Detectors trained with the proposed framework can transfer knowledge from the pseudo-labels generated with a the VL model as background knowledge, thus significantly improving accuracy even under the same problem setting.

Interestingly, the accuracy for AR in the proposed method is approaching, if not surpassing, that of the fully-supervised methods. This is thought to be

Table 5. Comparison of accuracy of detector trained with proposed framework and conventional method: annos column indicates the annotation types. AP⁵⁰ means average precision when the IoU threshold is 0.5. We report two results (val/test-dev) on MS COCO dataset. † indicates that was evaluated on MS COCO val-2014 subset.

annos	Model	Pascal VOC			MS COCO		
		AP	AP ⁵⁰	AR	AP	AP ⁵⁰	AR
Full	Faster R-CNN	51.6	82.7	62.1	37.4/37.7	58.1/58.7	51.7/52.5
	YOLOX	69.7	88.9	76.9	50.6/50.7	68.4/68.9	63.6/63.5
	CO-DINO	73.4	91.8	85.9	60.0/59.7	77.7/77.4	78.2/73.9
Semi	DETReg 5% [1]	–	–	–	24.8/–	–	–
	Semi-DETR(DINO) 5% [41]	–	–	–	40.1/–	–	–
	Semi-DETR(DINO) 30% [41]	65.2	86.1	–	–	–	–
	MixPL (DINO) 2% [7]	–	–	–	34.7/–	–	–
	MixPL (Faster R-CNN) 2% [7]	–	–	–	28.6/–	–	–
	MixPL (Faster R-CNN) 30% [7]	56.1	85.8	–	–	–	–
Weakly	Wetector [28]	–	54.9	–	12.6 [†] /–	26.1 [†] /–	24.7 [†] /–
	WSCL [29]	–	58.7	–	13.8/–	27.8/–	29.7/–
	WSTDN [35]	–	54.7	–	–	–	–
Free	Ours w/ Faster R-CNN	50.0	79.9	64.6	32.8/33.1	52.2/52.7	53.2/53.8
	Ours w/ YOLOX	66.4	85.1	79.2	42.2/42.0	59.4/59.0	61.9/61.3
	Ours w/ CO-DINO	71.2	87.8	86.8	50.4/51.6	68.1/68.1	74.0/72.1

due to the outputs of the VL models having a tendency to have over-detection. Thus, the proposed framework train from a larger number of instances than from supervised oracle annotations. A more details of the result is provided in the supplementary material.

Figure 5 shows a comparison of FPS and accuracy of VL models and our framework. This result indicates that, although the accuracy of the VL models is high, it is difficult to apply directly them to real-time processing due to the slow processing speed. In contrast, the object detector YOLOX trained with our framework, realizing processing speed, is applicable to real-time application scenarios while outperforming traditional un-, weakly and semi-supervised manners in terms of detection accuracy. Given that the proposed framework has no limitations on the type of detector or architecture, it is possible to achieve even faster inference while maintaining high accuracy by using conventional acceleration methods, such as Tensor RT [9, 36, 47].

4.4 Ablation Study

We evaluated the effectiveness of our three key aspects of our framework, i.e., the use of multiple VL models, score calibration, and TTA, described in Sect. 3.1 on the Pascal VOC dataset. The AP and AR of Faster R-CNN are shown in Table 6. Compared with the simple merging of multiple VL models, those with applied score calibration showed accuracy improvements of 1.5% and 1.4% in

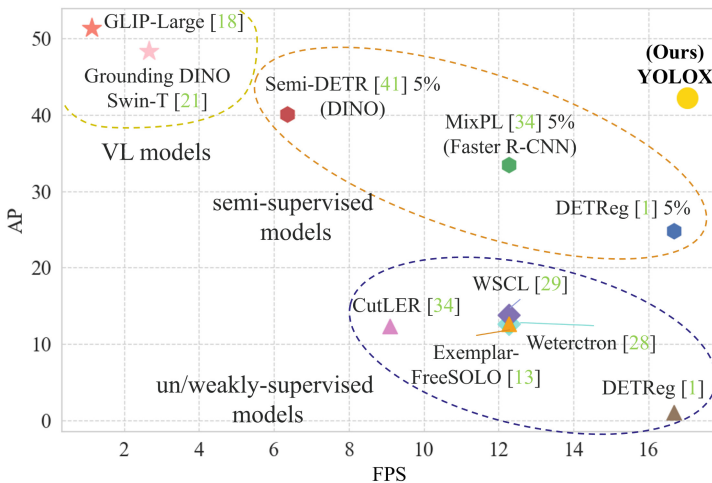


Fig. 5. Comparison of processing time and accuracy of VL models, object detectors trained with conventional methods and YOLOX trained with our framework on MS COCO dataset. Note that time of conventional methods are roughly estimated by referring from their backbone model size. FPS was measured with an NVIDIA Geforce 1080Ti.

Table 6. Ablation study of proposed framework based on detection accuracy of Faster R-CNN. In addition to simple integration of multiple VL models, introduction of score calibration and TTA is effective for improving detector accuracy.

Multiple VL models	Score calibration	TTA	AP	AR
✓			47.8	62.6
✓	✓		49.3	64.0
✓	✓	✓	50.0	64.6

AP and AR, respectively. In addition, using TTA during the GLIP data generation resulted in further improvements in accuracy, recording 0.7% and 0.6%, respectively. This ablation study clearly demonstrates that score calibration and TTA can enhance the performance of the proposed framework.

5 Conclusion

We proposed a simple-yet-effective framework for training lightweight object detectors by inheriting the rich expression power of multiple pre-trained visual-language models. The key is to obtain elaborate pseudo-labels for lightweight model training by extracting knowledge from multiple visual-language models, the biases of which are corrected by score correction. In contrast to current semi-supervised and unsupervised methods for object detection, our proposed framework is immediately applicable to state-of-the-art detectors. Comprehensive experiments on two public datasets demonstrated that the proposed framework is fast and lightweight while maintaining detection accuracy, and the results surpass those of supervised methods.

References

1. Bar, A., et al.: Detreg: unsupervised pretraining with region priors for object detection, pp. 14585–14595 (2021)
2. Bilen, H., Vedaldi, A.: Weakly supervised deep detection networks. In: CVPR, pp. 2846–2854 (2016)
3. Bodla, N., Singh, B., Chellappa, R., Davis, L.S.: Soft-nms – improving object detection with one line of code (2017)
4. 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
5. Chen, B., Li, P., Chen, X., Wang, B., Zhang, L., Hua, X.S.: Dense learning based semi-supervised object detection. In: CVPR, pp. 4805–4814 (2022)
6. Chen, K., et al.: MMDetection: open mmlab detection toolbox and benchmark. arXiv preprint [arXiv:1906.07155](https://arxiv.org/abs/1906.07155) (2019)
7. Chen, Z.Y., Zhang, W., Wang, X., Chen, K., Wang, Z.: Mixed pseudo labels for semi-supervised object detection. ArXiv [abs/2312.07006](https://arxiv.org/abs/2312.07006) (2023)

8. Everingham, M., Gool, L.V., Williams, C.K.I., Winn, J.M., Zisserman, A.: The pascal visual object classes (VOC) challenge. *IJCV* 303–338 (2010)
9. Ge, Z., Liu, S., Wang, F., Li, Z., Sun, J.: Yolox: exceeding yolo series in 2021. arXiv preprint [arXiv:2107.08430](https://arxiv.org/abs/2107.08430) (2021)
10. Gu, X., Lin, T.Y., Kuo, W., Cui, Y.: Open-vocabulary object detection via vision and language knowledge distillation. In: *ICLR* (2021)
11. Guo, Q., Mu, Y., Chen, J., Wang, T., Yu, Y., Luo, P.: Scale-equivalent distillation for semi-supervised object detection. In: *CVPR*, pp. 14502–14511 (2022)
12. Huang, Z., Zou, Y., Bhagavatula, V., Huang, D.: Comprehensive attention self-distillation for weakly-supervised object detection. ArXiv [abs/2010.12023](https://arxiv.org/abs/2010.12023) (2020)
13. Ishtiak, T., En, Q., Guo, Y.: Exemplar-freesolo: enhancing unsupervised instance segmentation with exemplars. In: *CVPR*, pp. 15424–15433 (2023)
14. Jia, C., et al.: Scaling up visual and vision-language representation learning with noisy text supervision. ArXiv [abs/2102.05918](https://arxiv.org/abs/2102.05918) (2021)
15. Kuznetsova, A., et al.: The open images dataset v4. *IJCV* 1956 – 1981 (2018)
16. Li, J., Li, D., Savarese, S., Hoi, S.C.H.: Blip-2: bootstrapping language-image pre-training with frozen image encoders and large language models. In: *ICML* (2023)
17. Li, J., Li, D., Xiong, C., Hoi, S.C.H.: Blip: bootstrapping language-image pre-training for unified vision-language understanding and generation. In: *ICML* (2022)
18. Li, L.H., et al.: Grounded language-image pre-training. In: *CVPR*, pp. 10955–10965 (2022)
19. Lin, T. Y., Dollar, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: *CVPR*, pp. 936–944 (2017)
20. 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
21. Liu, S., et al.: Grounding dino: marrying dino with grounded pre-training for open-set object detection. ArXiv [abs/2303.05499](https://arxiv.org/abs/2303.05499) (2023)
22. Lv, W., et al.: Detsr beat yolos on real-time object detection. vol. [abs/2304.08069](https://arxiv.org/abs/2304.08069) (2023)
23. Ning, C., Zhou, H., Song, Y., Tang, J.: Inception single shot multibox detector for object detection. In: *ICMEW*, pp. 549–554 (2017)
24. Ogawa, T., Shibata, T.: Frog-mot: fast and robust generic multiple-object tracking by iou and motion-state associations. In: *WACV* (2024)
25. Pan, J., Chen, S., Shou, M.Z., Liu, Y., Shao, J., Li, H.: Actor-context-actor relation network for spatio-temporal action localization. In: *CVPR*, pp. 464–474 (2021)
26. Radford, A., et al.: Learning transferable visual models from natural language supervision. In: *ICMR* (2021)
27. Ren, S., He, K., Girshick, R.B., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. *TPAMI* **39**, 1137–1149 (2015)
28. Ren, Z., et al.: Instance-aware, context-focused, and memory-efficient weakly supervised object detection. In: *CVPR*, pp. 10595–10604 (2020)
29. Seo, J., Bae, W., Sutherland, D.J., Noh, J., Kim, D.: Object discovery via contrastive learning for weakly supervised object detection. In: Avidan, S., Brostow, G., Cisse, M., Farinella, G.M., Hassner, T. (eds.) *Computer Vision – ECCV 2022*. *ECCV 2022*. LNCS, vol. 13691, pp. 312–329 . Springer, Cham (2022). https://doi.org/10.1007/978-3-031-19821-2_18
30. Solovyev, R., Wang, W., Gabruseva, T.: Weighted boxes fusion: ensembling boxes from different object detection models. *Image Vis. Comput.* 1–6 (2021)

31. Tang, J., Xia, J., Mu, X., Pang, B., Lu, C.: Asynchronous interaction aggregation for action detection. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12360, pp. 71–87. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58555-6_5
32. Tian, Z., Shen, C., Chen, H., He, T.: FCOS: fully convolutional one-stage object detection. In: ICCV, pp. 9626–9635 (2019)
33. Wang, X., et al.: FreeSOLO: learning to segment objects without annotations. arXiv preprint [arXiv:2202.12181](https://arxiv.org/abs/2202.12181) (2022)
34. Wang, X., Girdhar, R., Yu, S.X., Misra, I.: Cut and learn for unsupervised object detection and instance segmentation. In: CVPR, pp. 3124–3134 (2023)
35. Wang, Z., Zhang, W., Zhang, M.L.: Transformer-based multi-instance learning for weakly supervised object detection. ArXiv [abs/2303.14999](https://arxiv.org/abs/2303.14999) (2023)
36. Xia, X., et al.: TRT-VIT: tensorrt-oriented vision transformer (2022)
37. Yang, Q., Wei, X., Wang, B., Hua, X.S., Zhang, L.: Interactive self-training with mean teachers for semi-supervised object detection. In: CVPR, pp. 5937–5946 (2021)
38. Yang, S., Kim, Y., Kim, Y., Kim, C.: Combinational class activation maps for weakly supervised object localization. In: WACV, pp. 2930–2938 (2019)
39. Yao, L., et al.: Detclip: dictionary-enriched visual-concept paralleled pre-training for open-world detection. ArXiv [abs/2209.09407](https://arxiv.org/abs/2209.09407) (2022)
40. Zhang, H., et al.: Glipv2: unifying localization and vision-language understanding (2022)
41. Zhang, J., et al.: Semi-DETR: semi-supervised object detection with detection transformers. In: CVPR, pp. 23809–23818 (2023)
42. Zhang, Y., et al.: ByteTrack: multi-object tracking by associating every detection box. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) Computer Vision – ECCV 2022. ECCV 2022. LNCS, vol. 13682. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-20047-2_1
43. Zhou, B., Khosla, A., Lapedriza, À., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: CVPR, pp. 2921–2929 (2015)
44. Zhou, H., et al.: Dense teacher: dense pseudo-labels for semi-supervised object detection. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) Computer Vision – ECCV 2022. ECCV 2022. LNCS, vol. 13669, pp. 35–50. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-20077-9_3
45. Zhou, Q., Yu, C., Wang, Z., Qian, Q., Li, H.: Instant-teaching: an end-to-end semi-supervised object detection framework. In: CVPR, pp. 4079–4088 (2021)
46. Zhou, Y., Wen, S., Wang, D., Mu, J., Richard, I.: Object detection in autonomous driving scenarios based on an improved faster-rcnn. Appl. Sci. **11**(24) (2021)
47. Zhou, Y., Yang, K.: Exploring tensorrt to improve real-time inference for deep learning. In: HPCC/DSS/SmartCity/DependSys, pp. 2011–2018 (2022)
48. Zong, Z., Song, G., Liu, Y.: Detrs with collaborative hybrid assignments training. In: ICCV, pp. 6725–6735 (2022)



An Improved YOLOF for Scale Imbalance with Dilated Attention

Tsatsral Amarbayasgalan[✉], Mooseop Kim[✉], and Chi Yoon Jeong[✉]

Human Sensory Augmentation Research Section, Electronics and
Telecommunications Research Institute, Daejeon, Republic of Korea
{tsatsral,gomskim,iamready}@etri.re.kr

Abstract. Scale imbalance, where objects of different sizes are not equally represented in a dataset, is a common problem in real-world object detection scenarios that leads to significant performance degradation of object detection methods. Although several solutions have been proposed based on multilevel feature maps, these methods may not be suitable in real-time applications owing to their low speed and memory consumption. Recently, you only look one-level feature (YOLOF) was proposed based on a single-in-single-out (SiSo) architecture; the SiSo architecture is well suited for real-time applications with performance comparable to that of methods based on multilevel feature maps. However, they show limited performance when applied to real-world object detection scenarios with scale imbalance problems. Therefore, we propose a lightweight object detection method that can handle the scale imbalance problem while retaining the advantages of the SiSo framework. To mitigate the scale imbalance, we use dilated attention to extend the SiSo architecture and learn the scale range of objects. Extensive experiments on public datasets show the effectiveness of a dilated attention-based proposed method in scale-imbalanced scenarios. Our method achieves results comparable to those of the original YOLOF on the MS COCO and PASCAL VOC datasets. In particular, for imbalanced datasets, the proposed method outperforms the original YOLOF by 4.78% on the first-person-walking-livingroom dataset and by 1.38% on the imbalanced PASCAL VOC dataset in terms of average precision (AP)₅₀.

Keywords: Object detection · Scale-imbalance problem · YOLOF · Single-level feature · Dilated attention

1 Introduction

Object detection identifies objects with different scale ranges, from a single image or image sequences. Thus, it is an essential task in vision-based applications such as healthcare monitoring [39], robotics [12, 17, 21], and autonomous driving [3, 7]. Many studies have been conducted to identify objects in images accurately.

Object detection methods can be sorted into one- and two-stage detectors. The two-stage detectors generate a set of object proposals (candidate bounding

boxes) that may contain an object during the initial stage. In the second stage, the detector predicts the object class by extracting features from the generated proposals. A region convolutional neural network (R-CNN) [10] and its family [11, 29] are versions of two-stage detectors. Although two-stage detectors show state-of-the-art results, they are relatively slow owing to their enormous computational costs. In one-stage detectors, the bounding boxes and object classes are predicted using a single neural network without a proposal generation stage. Therefore, one-stage detectors significantly increase the detection speed, rendering them more suitable for real-time applications.

Among one-stage detectors, Redmon et al. [26] proposed you only look once (YOLO) algorithm in 2015, and its variants [5, 8, 28, 36] are commonly used in real-world applications owing to their excellent accuracy and high processing speed. Starting from YOLOv3 [28], multilevel feature maps have been used to enhance the detection of objects of different sizes; however, they sacrificed the processing speed owing to the computational overhead of multiple feature maps. YOLOX [8] was introduced by updating YOLOv3 architecture based on several advanced techniques: decoupled head, anchor-free, and advanced label assignment strategy. It uses three feature maps from a feature pyramid network (FPN) [18] to enhance object detection across multiple scales and increase its baseline YOLOv3 by 3.0% AP. RetinaNet [19] is also the FPN-based one-stage object detector. It proposes a Focal Loss function to address a class imbalance by focusing on misclassified examples and achieves higher accuracy than two-stage Faster-RCNN by leveraging multiple feature maps and the Focal Loss function.

Recently, you only look one-level feature (YOLOF) [5], which balances the accuracy and speed without using feature pyramids and transformer layers, has been proposed. The architecture of YOLOF is simple and efficient and utilizes only a single feature map extracted from the ResNet model [14]. Its excellent performance demonstrates that a single feature map can provide sufficient information for all object-scale ranges without feature fusion.

Existing object detection methods have achieved reasonable performance on well-prepared general object detection datasets such as MS COCO (common objects in contexts) [20] and PASCAL VOC (visual object classes) [6], which are scale-balanced datasets. When collecting real-world data, it is unlikely that the dataset is balanced across object categories and sizes. Thus, scale imbalance is a common problem in training object recognition models, particularly when a certain range of object sizes is over- or underrepresented in the dataset. A recent study [24] also indicated that many scale-imbalanced datasets exist, and scale imbalance is a challenging issue in object detection tasks.

We conducted controlled experiments to determine the impact of scale-imbalanced datasets on the performance of object detection models. We created a scale-imbalanced dataset from the PASCAL VOC 0712 dataset by increasing imbalance ratios. Figure 1a shows the percentage of small, medium, and large-scale objects in the original PASCAL VOC and imbalanced PASCAL VOC. We then trained YOLOF models on the original and imbalanced datasets and evaluated their performances using the PASCAL VOC test dataset. Our aim was

to demonstrate the degrading influence of imbalanced datasets on the detection performance, represented in Fig. 1. We can observe that when the imbalance ratio is increased (Fig. 1a), the performance is lowered (Fig. 1b).

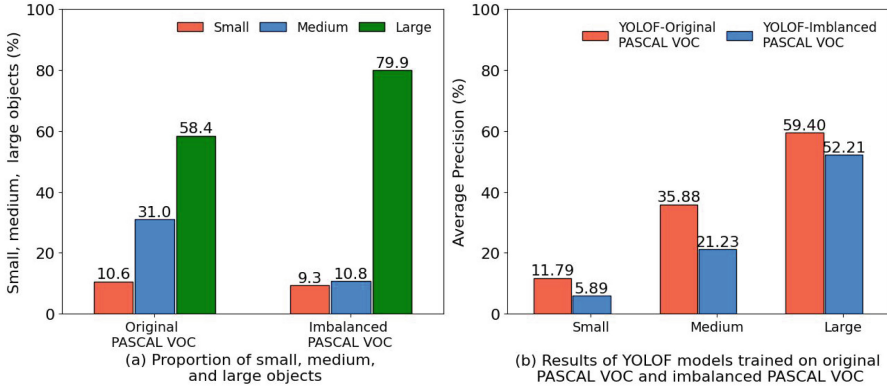


Fig. 1. Comparison of the average precision (AP) among the small, medium, and large object detection on the PASCAL VOC test dataset. We changed the original PASCAL VOC dataset by removing objects to increase scale imbalance.

Many studies have addressed this problem by making detection from image pyramids [25, 32, 33], feature hierarchies [23, 28], and feature pyramids [9, 18, 22]. Earlier object detection methods [32, 34] have used multiscale feature maps extracted from image pyramids to cover objects of different sizes. However, these methods are computationally expensive because they require the independent building of feature maps for all image scales. Another memory-efficient approach involves using deep convolutional neural networks (CNNs) to extract hierarchical feature maps in sequential layers. It uses features from shallow to higher layers to capture objects of various scales. The single shot detector (SSD) [23] was member of the initial methods to use a CNN-based multiscale feature hierarchy. Subsequently, the FPN was applied to many state-of-the-art detectors by combining shallow-level spatially rich features and deep-level semantic-rich features through a top-down architecture to enhance feature maps. However, it requires more resources and decelerate the speed because of its costly procedure, such as extracting multiple feature maps and performing object detection on each feature map.

Therefore, this study proposes a lightweight object detection method that uses a powerful single-level feature map for scale-imbalanced datasets. Inspired by the YOLOF algorithm, the proposed object detection method uses a single feature map. To address scale-imbalanced datasets, we propose an attention module based on several dilation rates to focus on all ranges of object scales without multiple feature maps. We summarized the main contributions as follows:

- We propose a lightweight object detection method with a powerful single-level feature map for scale-imbalanced datasets.
- We propose an attention module based on multiple dilation rates to focus on all ranges of object scales when improving the detection performance.
- We evaluated the proposed method on well-organized public datasets such as MS COCO and PASCAL VOC and the scale-imbalanced first-person-walking datasets. Experimental results show that the proposed method outperforms the original YOLOF.

The remainder of this study is organized as follows: Sect. 2 reviews existing methods in object detection; Sect. 3 details the method presented in this study; Sect. 4 introduces the experimental environment and datasets and then evaluates and compares the performance of the proposed method on the MS COCO, PASCAL VOC, and first-person-walking datasets; Sect. 5 concludes the paper.

2 Related Work

Current object detection methods rely on two or one-stage mechanisms. R-CNN [10] is a popular example of a two-stage detector that implements a region proposal strategy. The first stage produces candidate object locations called region proposals. The second stage uses deep CNN and support vector machine (SVM) models to predict object classes from the region proposals. However, this method works slower because of the requirement for a forward pass in the feature-extraction CNN model for each proposed region. In addition, the training pipeline is complicated because three different models must be trained separately for feature extraction, classification, and bounding box regression. Then, Fast-RCNN [11] addresses these problems using a region of interest pooling layer to learn features from the entire image by one forward pass. Another solution in fast-RCNN is to use a single CNN model for all tasks, ranging from feature extraction to classification. Faster-RCNN [29] proposed a more efficient training framework by combining the region proposal network into the entire network architecture instead of the external region proposal. This unified object-detection framework achieves faster speed with competitive accuracy than its predecessors.

For one-stage detectors, object detection is performed without a region proposal stage. SSD [23] and YOLO [26] are early representatives of one-stage detectors, demonstrating promising results at high speeds. SSD uses multiple layers of convolutional feature maps in a single CNN to predict object boxes and labels. Conversely, YOLO splits the input image into grids, where each grid predicts the bounding boxes and their confidence scores. These detectors significantly improved the speed, but their accuracy was low or similar to the two-stage methods.

Both one- and two-stage object detection methods encounter a scale imbalance problem during model training. Their performances were reasonably good for well-organized datasets but dropped significantly for imbalanced datasets. Numerous techniques have been proposed to solve scale imbalance problems in

object detection, as summarized in references [24, 40]. These methods typically involve extracting multiscale features from images of different scales or constructing pyramidal features from single-scale images to detect objects of various sizes.

In [32], the authors proposed scale normalization for image pyramids, developed several detectors on images of varying scales, and performed back-propagation on objects with selected sizes. A more efficient training schema was proposed in SNIPER [34] using an image-cropping approach. Although the image pyramid strategy can increase performance, it is unsuitable for real-time applications owing to its high memory consumption. Another memory-effective method compared with the image pyramid approach is to extract a multiscale feature hierarchy from a single-scale image, with each level of feature map representing different scales of objects. The SSD [23] is one of the first methods to use a CNN-based multiscale feature hierarchy. It creates a feature pyramid by adding several new layers to a pretrained VGG-16 [31] backbone network. However, the pyramid does not involve low-level features with a high resolution, which is significant for detecting small objects [18].

Recently, feature fusion techniques have become increasingly popular for obtaining feature maps rich in semantic information. A popular method for constructing feature pyramids is FPN, which fuses high- and low-level feature maps by a top-down architecture. Subsequently, many different versions of feature fusion techniques were introduced in various studies, such as the path aggregation network (PANet) [22] and NAS-FPN [9]. Popular object detection methods that use FPN as their feature extraction architecture include the RetinaNet [19], R-CNN families [10, 11, 29], YOLOv3 [28], and its successors. YOLOX [8] is a family of the YOLO series, which utilizes a combination of FPN and PANet to enhance multilevel feature representations. Although multilevel features enhance detection performance on different scales, they also lead to increased computation.

Earlier object detection methods such as R-CNN [10], YOLO [26], and YOLOv2 [27] used only one feature map, which is the top layer of the feature extraction network, to gather information about all objects. However, further improvements in speed and accuracy are still required to render these methods more efficient and accurate for real-time applications. YOLOF proposed a single-in-single-out (SiSo) architecture and has shown comparable results to that of RetinaNet [19] and detection transformer (DETR) [2] models using one feature map from the ResNet [14] model. They used dilated residual blocks to enlarge a single-feature map without multiscale feature fusion. Several studies have proposed enhancements based on YOLOF, such as the attention mechanism [30, 37] and feature fusion [16], to enhance the performance in specific areas. In this study, we focused on all ranges of object scales using a dilated attention without complex modification on SiSo architecture.

3 Method

In the Introduction, we explained that a scale-imbalanced dataset significantly affects YOLOF’s performance. Considering this shortcoming of SiSo architec-

ture, we improved it without significantly increasing the workload of the detector. This section briefly introduces the SiSo architecture used in the proposed method and details the dilated attention module applied to the SiSo framework.

3.1 Main Components of Base Framework

We used the SiSo architecture as the basic framework owing to its good performance and high speed. The SiSo architecture consists of three modules: backbone, encoder, and decoder. The detection pipeline in Fig. 2 shows the overall structure of the proposed method based on SiSo architecture.

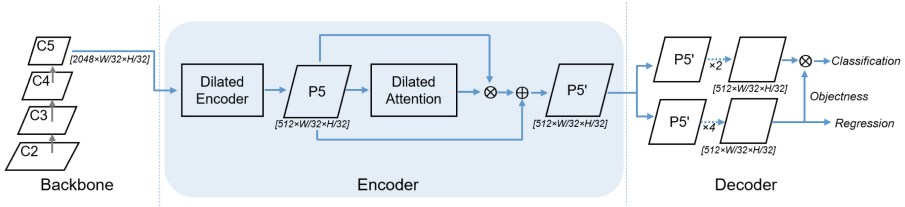


Fig. 2. Overall structure of the proposed method.

Backbone. The SiSo architecture uses a single feature map that can provide sufficient information to detect all the objects. We used a ResNet-50 model pretrained on ImageNet to learn the feature maps. Each level of the feature map in the backbone model decreases the resolution while increasing the number of channels. For subsequent analysis, we used only the highest-level C5 feature map from ResNet-50, which has 2048 channels and a downsampling rate 32.

Encoder. The encoder is responsible for feature enhancement before the detection process. The detection part of YOLOF uses only a one-level feature map, which can degrade accuracy. A dilated encoder was used to obtain multiscale information to address the loss of accuracy. The dilated encoder structure comprises a projector and residual blocks. The projector part reduces the number of channels to 512 using 1×1 convolutional layer and then generates a feature map for residual blocks using a 3×3 convolution layer. Subsequently, four consecutive residual blocks with different dilation rates of 2, 4, 6, 8 produced a final feature map capable of representing all scales of objects. Each residual block was composed of a 1×1 convolutional layer for channel reduction with a reduction rate of 4, followed by a 3×3 dilated convolutional layer to cover all objects on various scales; a 1×1 convolutional layer was used to restore the number of channels back to 512. In this study, we incorporated dilated attention into the neck structure after the encoder (Fig. 2).

Decoder. The decoder in this model has two separate heads, each with a different number of convolutions. One head is responsible for the classification,

whereas the other is responsible for the bounding box regression. The regression head has four 3×3 convolutions, whereas the classification head has two 3×3 convolutions. The two heads were calculated separately. At the end of the regression head, an implicit objectness prediction was added to each anchor box to determine whether the box contains an object. Finally, the classification confidence was estimated by multiplying the result of the classification head by the objectness score from the regression head.

3.2 Dilated Attention Module

We only used a one-level feature map in the detection process, which requires further feature improvement to represent multiscale objects. In YOLOF, the C5 feature map from the ResNet backbone is enlarged in the neck part using the encoder module before being used for detection. The encoder module enriches the feature representation using a projector and residual blocks with different dilated convolutional layers. However, scale-imbalanced datasets require additional analysis to achieve a good performance.

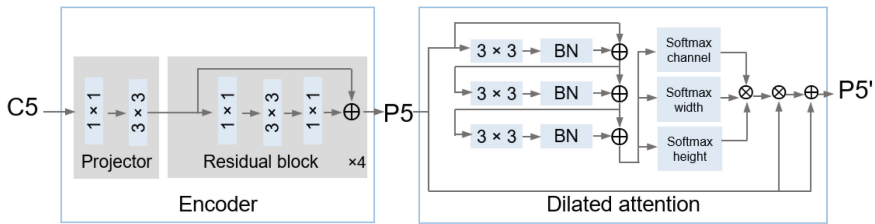


Fig. 3. Structure of the proposed feature enhancement (neck) module. It appends dilated attention to the YOLOF’s neck part. In the figure, 1×1 and 3×3 denote 1×1 and 3×3 convolution layers, respectively, and BN stands for a batch normalization layer.

In CNN-based object detectors, channel and spatial attention focus on the semantic information. The convolutional block attention module (CBAM) [38] and squeeze-and-excitation networks (SENet) [15] are the main techniques for attention in convolutional networks. For transformer-based methods, they utilize a multihead self-attention [35] to handle long-range dependencies between image patches. However, it introduces significant computation complexity, and techniques such as sparse self-attention and window-based local self-attention are proposed to reduce the complexity. The dilated neighborhood attention [13] is one approach to reduce complexity, applying a concept of dilated convolution to its self-attention mechanism for calculating sparse attention on a subset of patches in a dilated manner rather than being contiguous. Our proposed method uses a single-level feature; the attention module can focus on all scales of objects

on this feature using conventional dilated convolutions with several dilation rates without multiple feature maps.

In this study, we redesigned the YOLOF’s neck structure by appending a dilated attention section. One solution to obtain efficient information for objects of all sizes is to use an attention mechanism on the feature map. We focused on the output of the encoder module to effectively consider the pixel information. The input of the proposed dilated attention is obtained from the result of the last stage of the encoder module (Fig. 3). The dilated attention uses 3×3 convolutional layers with dilation rates of 2, 3, and 4. A batch normalization layer followed each convolutional layer. Finally, three softmax layers are aligned for the channel, width, and height dimensions to obtain useful semantic information from the feature map. The first softmax layer accounts for channel-wise attention, whereas the other two layers account for spatial attention.

The dilated attention module makes the detector focus on all scale objects, even if the dataset is scale-imbalanced, without degrading the performance on well-distributed datasets. The layers used in the attention module and their impact on the network are evaluated in detail in Sect. 4.

4 Experimental Study

4.1 Datasets

We evaluated the object detection models using three public datasets. The experimental study used the MS COCO 2017 [20], PASCAL VOC 0712 [6] datasets, commonly used as standard datasets in object-detection research, and the first-person-walking [1] imbalanced open datasets in living room, bathroom, and balcony environments.

The MS COCO 2017 dataset comprises 118,287 and 5,000 images for training and testing, respectively, and includes 80 object categories. For the PASCAL VOC 0712 dataset, a trainval split with 16,551 images was used for training, and a test split with 4,946 images was used for testing, with the objects divided into 20 categories. The first-person-walking datasets, available on the AI Hub website [1], was collected to train AI-based models to identify indoor/outdoor first-person walking environments, such as roads and alleys, and obstacles to walking, particularly for disabled and elderly individuals. This dataset consists of images captured from 18 different locations at distances of 165, 80, 60, and 40 cm. We used the datasets in living room, bathroom, and balcony environments with a point of view of 165 cm. The number of training and testing images of first-person-walking-livingroom, first-person-walking-bathroom, and first-person-walking-balcony are (11,986 and 2,508), (2,215 and 655), and (3,142 and 490); the numbers of object categories are 13, 9, and 12, respectively. Figure 4 shows the proportion of small, medium, and large objects to the total number of objects in the experimental datasets. The first-person-walking datasets are highly scale-imbalanced compared to the MS COCO 2017 and PASCAL VOC 0712 datasets, with over 90% of the objects belonging to large-scale objects.

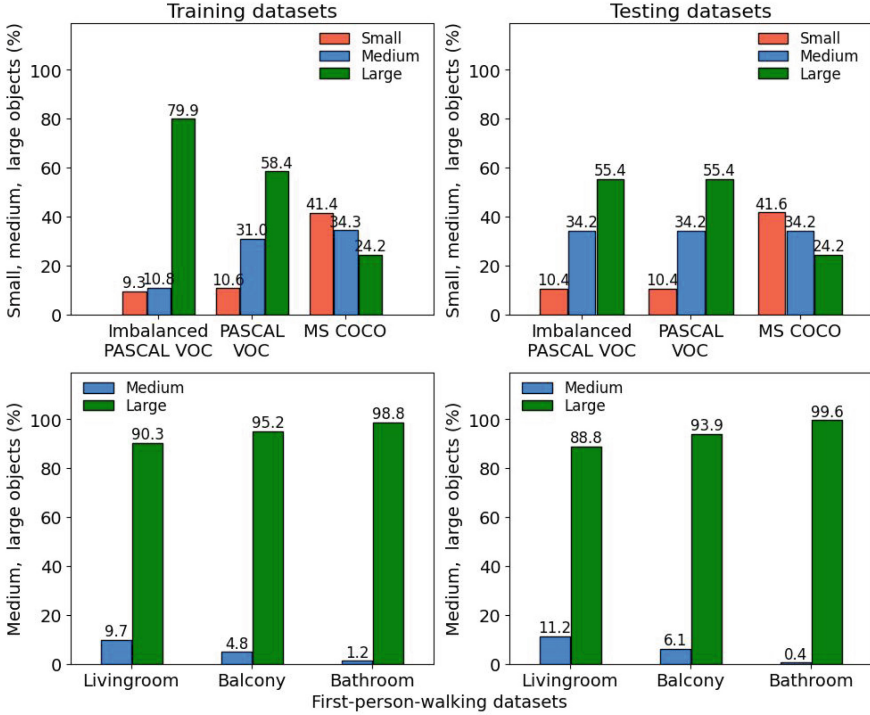


Fig. 4. Proportion of small, medium, and large objects in the overall count of objects. Objects are categorized based on their size: small ($\text{area} < 32^2$), medium ($32^2 < \text{area} < 96^2$), and large ($\text{area} > 96^2$).

4.2 Experimental Setup

The experiments were conducted on a server equipped with an NVIDIA RTX A6000 GPU. We used the MMDetection [4] object-detection toolbox based on Pytorch, and the parameter configurations were similar to those of MMDetection. To ensure stable training, we rescaled the learning rate values for our server. We used the ResNet-50 model as the backbone for all compared detectors. We used a stochastic gradient descent (SGD) optimizer with a weight decay of 0.0001 and a momentum of 0.9 to train the models. The training used the $1\times$ schedule, with a learning rate decay of 0.1 at epochs 8 and 11.

4.3 Experimental Results

We evaluated the proposed attention-based YOLOF on various datasets: MS COCO, PASCAL VOC, first-person-walking, and customized PASCAL VOC datasets. We compared this with the original YOLOF. To evaluate the proposed method on different scale-imbalanced datasets, images with objects of

particular sizes were removed from the PASCAL VOC dataset to increase the scale-imbalance ratio. The learning rate for stable training changed based on the number of GPUs. Both models used the same learning rate (0.06) on the MS COCO, PASCAL VOC, and imbalanced PASCAL VOC datasets, whereas the learning rates for the models trained on the first-person-walking datasets were 0.24. The ResNet-50 model was used as the backbone of the compared models on a 1x schema with 12 epochs.

Table 1 lists the performances of the original YOLOF model and the proposed model on MS COCO, PASCAL VOC, and imbalanced PASCAL VOC datasets. For the MS COCO and PASCAL VOC datasets, the proposed method slightly enhanced the detection of small and medium objects. For the imbalanced PASCAL VOC dataset, the attention-based YOLOF achieved better performance on all scales of objects, improving AP by 1.62% and AP on small, medium, and large objects by 1.06%, 1.97%, and 1.55%, respectively.

Table 1. Comparison with YOLOF on open datasets.

Dataset	Method	$AP(\%)$	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
MS COCO	YOLOF	37.56	56.90	40.34	18.80	42.32	52.78
	Ours	37.58	56.94	40.48	18.98	42.34	52.58
PASCAL VOC	YOLOF	49.68	76.97	53.99	11.79	35.88	59.40
	Ours	49.83	77.13	54.15	12.03	35.92	59.55
Imbalanced PASCAL VOC	YOLOF	40.34	67.42	41.81	5.89	21.23	52.21
	Ours	41.96	68.80	43.76	6.95	23.20	53.76

We validated the generalizability of our method on real-world datasets in different environments, and the results are listed in Table 2. The presented method outperformed YOLOF on all real-world datasets by improving AP_{50} between 2.46% and 4.78%. Particularly, first-person-walking-bathroom and first-person-walking-balcony are extremely scale-imbalanced datasets because they consist of only 1.2% and 4.8% of medium objects, respectively, and 98.8% and 95.2% of large objects, respectively. Our method improved the performances for medium and large-sized objects on the first-person-walking-bathroom dataset by 0.24% and 1.94%, respectively, and on the first-person-walking balcony dataset by 0.12% and 1.59%, respectively.

In addition, we compared the proposed method with other one- and two-stage algorithms, namely Faster-RCNN, RetinaNet, SSD300, SSD512, YOLOX, and DETR, on the first-person-walking-livingroom dataset. The results are presented in Table 3. The proposed method demonstrated an improvement in AP_{50} over the other algorithms: 4.12% over SSD512, 4.27% over Faster-RCNN, 6.69% over RetinaNet, 9.11% over SSD300, 11.44% over DETR, 22.01% over YOLOX, and 4.78% over our baseline YOLOF. Table 3 shows that SSD300, SSD512, DETR, and YOLOF have fewer FLOPs than the proposed method, but their AP is

Table 2. Comparison with YOLOF on real-world first-person-walking datasets. “-” represents that the corresponding area has no objects.

Environments	Method	$AP(\%)$	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Living room	YOLOF	16.92	31.79	16.88	-	2.41	17.84
	Ours	19.33	36.57	18.56	-	3.26	19.76
Bathroom	YOLOF	39.43	56.74	46.23	-	0.00	40.1
	Ours	41.37	59.43	48.62	-	0.24	42.04
Balcony	YOLOF	10.00	21.33	8.92	-	6.06	10.22
	Ours	11.49	23.79	9.68	-	6.17	11.81

lower. Conversely, Faster-RCNN, RetinaNet, and YOLOX are FPN-based object detectors that require more operations than ours, resulting in higher FLOPs. Our method has 50% fewer FLOPs than Faster-RCNN and RetinaNet.

Table 3. Comparisons of different object detection methods on the first-person-walking-livingroom test dataset. ResNet-50 serves as the backbone for the experimented models. Only the YOLOX model employed CSPDarknet as its backbone. “-” represents that the corresponding area has no objects.

Method	Epochs	#Params	FLOPs	$AP(\%)$	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Faster-RCNN	12	41.4 M	208 G	17.80	32.39	17.22	-	1.54	18.35
SSD300	12	25.4 M	31 G	12.53	27.46	9.36	-	0.82	13.07
SSD512	12	26.2 M	89 G	16.34	32.45	15.42	-	1.02	16.90
RetinaNet	12	36.6 M	211 G	15.54	29.88	14.66	-	2.36	16.01
DETR	500	41.6 M	97 G	12.66	25.13	11.23	-	0.24	12.98
YOLOX	300	99.0 M	141 G	8.47	14.56	8.54	-	0.7	8.99
YOLOF	12	42.6 M	99 G	16.92	31.79	16.88	-	2.41	17.84
Ours	12	49.7 M	106 G	19.33	36.57	18.56	-	3.26	19.76

We conducted a cost analysis of the proposed model, the original YOLOF model, and other counterparts to multilevel feature maps-based models, such as Faster-RCNN, RetinaNet and YOLOX. All the models except YOLOX used the ResNet-50 backbone, and YOLOX employed CSPDarknet as its backbone. To obtain the number of parameters, frames per second (FPS), and floating point operations (FLOPs), we used analysis tools from MMDetection [4]. The FLOPs were estimated for the first 100 images of the first-person-walking-livingroom test dataset, which had a size of 768×1344 . A single GPU with a batch size of one was used to calculate the FPS. The analysis of memory consumption during inference was measured by summing CPU and GPU memory usage for the first 100 images of the first-person-walking-livingroom test dataset. Table 4 compares

memory consumption, computation complexity, and speed between multilevel feature maps-based models and our proposed model.

Compared to the methods with multilevel feature maps, the memory consumption of the proposed method is lesser by 34 MB to 629 MB. Faster-RCNN and RetinaNet use multilevel feature maps from FPN, doubling their FLOPs from YOLOF and ours. In contrast, the proposed method requires 4578.19 MB of memory, outperforming all multilevel maps-based methods by FPS of 45.9 and AP_{50} of 36.57%.

Faster-RCNN and RetinaNet, as the models with the highest FLOPs, require 31.9 ms and 33.8 ms, respectively, to predict for a single image. YOLOX achieves the fastest speed among the multilevel feature maps-based detection models, with an inference time of 24.1 ms. YOLOF demonstrates the highest speed, and our method decreased it by less than 1 ms, adding an attention module. Our model requires 21.7 ms for inference, which is an acceptable result compared to the YOLOF, as our proposed model has superior accuracy. In the proposed method, we appended attention layers after the encoder module to the neck of YOLOF, which increased the number of parameters by 7M. The FPS decreased slightly by 4%, and FLOPs increased by 7%. Despite the dilated attention module increasing YOLOF’s memory consumption by 27.03 MB, it improved the baseline by 4.78% AP_{50} with minimal impact on speed, with only difference of 0.7 ms.

Table 4. Comparison of the FLOPs, AP, FPS, the number of parameters, and memory usage between the proposed model, YOLOF model, and multilevel feature maps-based models.

Method	FLOPs(G)	FPS	#Params(M)	AP_{50}	Memory(MB)
Faster-RCNN	208	31.0	41.4	32.39	4706.17
RetinaNet	211	29.4	36.6	29.88	4612.21
YOLOX	141	40.1	99.0	14.56	5207.96
YOLOF	99	47.2	42.6	31.79	4551.16
Ours	106	45.9	49.7	36.57	4578.19

Because the proposed method does not detect objects from multilevel features, the attention module focuses on all scales of objects on a single feature map. The proposed dilated attention uses several dilated convolutions to enhance important regions in the feature map for objects of varying sizes. We conducted experiments with different dilation values in the attention module to confirm that dilated attention on the single feature map improves the performance. Table 5 lists the results of different dilations in the attention module on the first-person-walking-livingroom dataset. As a result, dilations with values of (2, 3, 4) are better than the same dilations such as (1, 1, 1) or (2, 2, 2). However, performance starts to drop when using larger dilations with values of (3, 3, 3) and (3, 4, 5).

Table 5. Ablation study of dilation ratios on the proposed attention module. “–” represents that the corresponding area has no objects.

Dilations	AP(%)	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Baseline	16.92	31.79	16.88	–	2.41	17.84
(1,1,1)	17.90	33.18	17.21	–	2.88	18.32
(2,2,2)	17.57	33.91	16.71	–	2.87	18.02
(3,3,3)	17.33	32.47	16.7	–	3.03	17.85
(1,2,3)	17.97	34.64	16.94	–	2.98	18.39
(2,3,4)	19.33	36.57	18.50	–	3.26	19.76
(3,4,5)	17.28	32.59	16.83	–	2.85	17.77

We analyzed the impact of the proposed attention component on YOLOF in terms of their coverage of channel, width, and height. According to the results presented in Table 6, each type of attention positively impacts the performance of YOLOF. Attention to the channel using channel dimension as input to softmax improved both medium and small object detection, whereas attention to the width or height spatial dimension as input to softmax only enhanced medium object detection. The combination of width and height attention parts improved AP for medium object detection by 1%, whereas using all attention parts led to an overall improvement in performance, particularly in large object detection, by 1.92%.

Table 6. Ablation study of input dimensions on the proposed attention module. “–” represents that the corresponding area has no objects.

Method	AP(%)	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Baseline	16.92	31.79	16.88	–	2.41	17.84
+Attention channel	17.79	33.39	17.35	–	2.86	18.25
+Attention width	17.04	32.62	16.11	–	2.69	17.56
+Attention height	17.29	32.51	16.83	–	3.06	17.73
+Attention width and height	17.86	33.86	17.40	–	3.42	18.30
+Attention all	19.33	36.57	18.50	–	3.26	19.76

We replaced the proposed attention block in YOLOF with two more commonly used attention mechanisms: CBAM and SE. Figure 5 illustrates the impacts of these attentions on the YOLOF algorithm. The results show that the distributions of both the baseline and SE-based models were more widespread than those of the other models, which can be attributed to the relatively different results. The CBAM-based YOLOF showed a short distribution, but the mean and maximum values were lower than those of the proposed attention-based YOLOF, which had the shortest distribution and highest value.

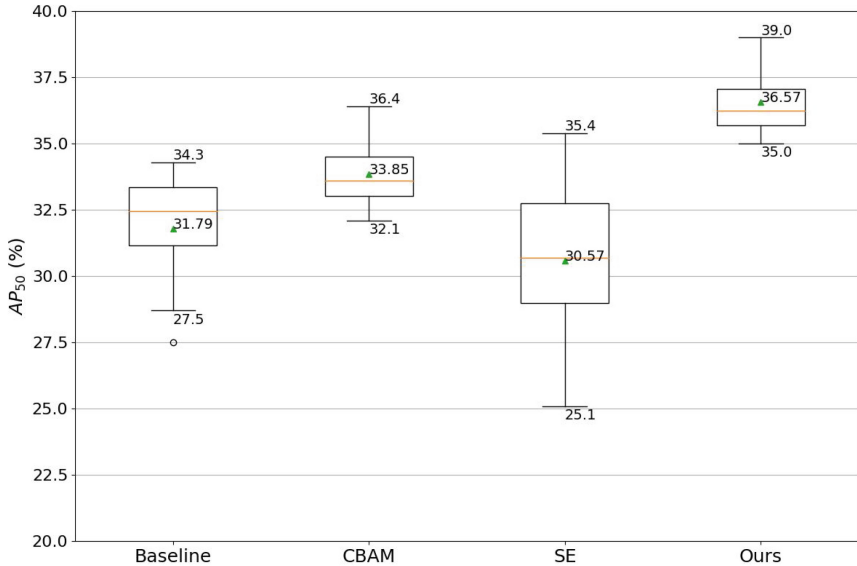


Fig. 5. Performance distribution of the compared attention-based YOLOF models on the first-person-walking-livingroom test dataset.

5 Conclusions

This study highlights the negative impact of datasets with imbalanced scales on detection accuracy. To address this problem, several studies have introduced techniques based on multiple feature maps for specific object scales. However, these techniques increase computational cost and reduce detection speed. In response, we propose a lightweight method based on a single feature map that extends YOLOF. We appended the dilated attention module to the neck part, which enhanced the detection performance on imbalanced datasets without significantly affecting the speed. We compared our method with YOLOF on scale-imbalanced and standard object detection datasets and demonstrated its efficiency. Our dilated attention-based YOLOF will serve as a robust model for imbalanced datasets in future research.

Acknowledgements. This work was supported by the Electronics and Telecommunications Research Institute (ETRI) grant funded by the Korean government (24ZB1200, Fundamental Technology Research for Human-Centric Autonomous Intelligent Systems).

References

1. AI-Hub: The open AI dataset project (2020). <https://www.aihub.or.kr>

2. 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
3. Carranza-García, M., Lara-Benítez, P., García-Gutiérrez, J., Riquelme, J.C.: Enhancing object detection for autonomous driving by optimizing anchor generation and addressing class imbalance. *Neurocomputing* **449**, 229–244 (2021). <https://doi.org/10.1016/j.neucom.2021.04.001>
4. Chen, K., et al.: Mmdetection: open mmlab detection toolbox and benchmark (2019)
5. Chen, Q., Wang, Y., Yang, T., Zhang, X., Cheng, J., Sun, J.: You only look one-level feature. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 13034–13043. IEEE Computer Society, Los Alamitos, CA, USA, June 2021. <https://doi.org/10.1109/CVPR46437.2021.01284>
6. Everingham, M., Eslami, S.M.A., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classes challenge: a retrospective. *Int. J. Comput. Vis.* **111**(1), 98–136 (2015). <https://doi.org/10.1007/s11263-014-0733-5>
7. Feng, D., Harakeh, A., Waslander, S.L., Dietmayer, K.: A review and comparative study on probabilistic object detection in autonomous driving. *IEEE Trans. Intell. Transp. Syst.* **23**(8), 9961–9980 (2022). <https://doi.org/10.1109/TITS.2021.3096854>
8. Ge, Z., Liu, S., Wang, F., Li, Z., Sun, J.: YOLOX: exceeding yolo series in 2021 (2021)
9. Ghiasi, G., Lin, T., Le, Q.V.: Nas-fpn: learning scalable feature pyramid architecture for object detection. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7029–7038. IEEE Computer Society, Los Alamitos, CA, USA (June 2019). <https://doi.org/10.1109/CVPR.2019.00720>
10. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 580–587. IEEE Computer Society, Los Alamitos, CA, USA (June 2014). <https://doi.org/10.1109/CVPR.2014.81>
11. Girshick, R.: Fast r-cnn. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1440–1448 (2015). <https://doi.org/10.1109/iccv.2015.169>
12. Griffin, B.A., Corso, J.J.: Depth from camera motion and object detection. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1397–1406. IEEE Computer Society, Los Alamitos, CA, USA (June 2021). <https://doi.org/10.1109/CVPR46437.2021.00145>
13. Hassani, A., Shi, H.: Dilated neighborhood attention transformer. arXiv preprint [arXiv:2209.15001](https://arxiv.org/abs/2209.15001) (2022)
14. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778. IEEE Computer Society, Los Alamitos, CA, USA (June 2016). <https://doi.org/10.1109/CVPR.2016.90>
15. Hu, J., Shen, L., Albanie, S., Sun, G., Wu, E.: Squeeze-and-excitation networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **42**(08), 2011–2023 (2020). <https://doi.org/10.1109/TPAMI.2019.2913372>
16. Jing, R., Zhang, W., Liu, Y., Li, W., Li, Y., Liu, C.: An effective method for small object detection in low-resolution images. *Eng. Appl. Artif. Intell.* **127**, 107206 (2024). <https://doi.org/10.1016/j.engappai.2023.107206>

17. Kim, G.S., Lee, H., Park, S., Kim, J.: Joint frame rate adaptation and object recognition model selection for stabilized unmanned aerial vehicle surveillance. *ETRI J.* **45**(5), 811–821 (2023)
18. Lin, T., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 936–944. IEEE Computer Society, Los Alamitos, CA, USA (July 2017). <https://doi.org/10.1109/CVPR.2017.106>
19. Lin, T., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **42**(02), 318–327 (2020). <https://doi.org/10.1109/TPAMI.2018.2858826>
20. 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
21. Liu, G., Hu, Y., Chen, Z., Guo, J., Ni, P.: Lightweight object detection algorithm for robots with improved yolov5. *Eng. Appl. Artif. Intell.* **123**(PA) (2023). <https://doi.org/10.1016/j.engappai.2023.106217>
22. Liu, S., Qi, L., Qin, H., Shi, J., Jia, J.: Path aggregation network for instance segmentation. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 8759–8768. IEEE Computer Society, Los Alamitos, CA, USA (June 2018). <https://doi.org/10.1109/CVPR.2018.00913>
23. 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
24. Oksuz, K., Cam, B.C., Kalkan, S., Akbas, E.: Imbalance problems in object detection: a review. *IEEE Trans. Pattern Anal. Mach. Intell.* **43**(10), 3388–3415 (2021). <https://doi.org/10.1109/TPAMI.2020.2981890>
25. Pang, Y., Wang, T., Anwer, R., Khan, F., Shao, L.: Efficient featurized image pyramid network for single shot detector. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7328–7336. IEEE Computer Society, Los Alamitos, CA, USA (June 2019). <https://doi.org/10.1109/CVPR.2019.00751>
26. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: unified, real-time object detection. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 779–788. IEEE Computer Society, Los Alamitos, CA, USA (June 2016). <https://doi.org/10.1109/CVPR.2016.91>
27. Redmon, J., Farhadi, A.: Yolo9000: better, faster, stronger. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6517–6525. IEEE Computer Society, Los Alamitos, CA, USA (July 2017). <https://doi.org/10.1109/CVPR.2017.690>
28. Redmon, J., Farhadi, A.: Yolov3: an incremental improvement (2018)
29. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. In: Cortes, C., Lawrence, N., Lee, D., Sugiyama, M., Garnett, R. (eds.) *Advances in Neural Information Processing Systems*, vol. 28. Curran Associates, Inc. (2015). <https://doi.org/10.1109/tpami.2016.2577031>
30. Ruan, Z., Cao, J., Wang, H., Guo, H., Yang, X.: Adaptive feedback connection with a single-level feature for object detection. *IET Comput. Vis.* **16**(8), 736–746 (2022). <https://doi.org/10.1049/cvi2.12121>
31. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition (2015)

32. Singh, B., Davis, L.S.: An analysis of scale invariance in object detection - snip. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3578–3587. IEEE Computer Society, Los Alamitos, CA, USA (June 2018). <https://doi.org/10.1109/CVPR.2018.00377>
33. Singh, B., Najibi, M., Sharma, A., Davis, L.S.: Scale normalized image pyramids with autofocus for object detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **44**(07), 3749–3766 (2022). <https://doi.org/10.1109/TPAMI.2021.3058945>
34. Singh, B., Najibi, M., Davis, L.S.: Sniper: efficient multi-scale training. In: Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R. (eds.) *Advances in Neural Information Processing Systems*, vol. 31. Curran Associates, Inc. (2018)
35. Vaswani, A., et al.: Attention is all you need. *Adv. Neural Inf. Process. Syst.* **30** (2017)
36. Wang, C., Bochkovskiy, A., Liao, H.: Yolov7: trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7464–7475. IEEE Computer Society, Los Alamitos, CA, USA (June 2023). <https://doi.org/10.1109/CVPR52729.2023.00721>
37. Wang, Q., Qian, Y., Hu, Y., Wang, C., Ye, X., Wang, H.: M2YOLOF: based on effective receptive fields and multiple-in-single-out encoder for object detection. *Expert Syst. Appl.* **213**, 118928 (2023). <https://doi.org/10.1016/j.eswa.2022.118928>
38. 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
39. Yücel, Z., Akal, F., Oltulu, P.: Mitotic cell detection in histopathological images of neuroendocrine tumors using improved yolov5 by transformer mechanism. *SIViP* **17**(8), 4107–4114 (2023). <https://doi.org/10.1007/s11760-023-02642-8>
40. Zou, Z., Chen, K., Shi, Z., Guo, Y., Ye, J.: Object detection in 20 years: a survey. *Proc. IEEE* **111**(3), 257–276 (2023). <https://doi.org/10.1109/JPROC.2023.3238524>



CLIP-Based Point Cloud Classification via Point Cloud to Image Translation

Shuvozit Ghose¹(✉), Manyi Li², Yiming Qian¹, and Yang Wang³

¹ University of Manitoba, Winnipeg, Canada
shuvozit.ghose@gmail.com

² Shandong University, Jinan, China

³ Concordia University, Montreal, Canada

Abstract. Point cloud understanding is an inherently challenging problem because of the sparse and unordered structure of the point cloud in the 3D space. Recently, Contrastive Vision-Language Pre-training (CLIP) based point cloud classification model i.e. PointCLIP has added a new direction in the point cloud classification research domain. In this method, at first multi-view depth maps are extracted from the point cloud and passed through the CLIP visual encoder. To transfer the 3D knowledge to the network, a small network called an adapter is fine-tuned on top of the CLIP visual encoder. PointCLIP has two limitations. Firstly, the point cloud depth maps lack image information which is essential for tasks like classification and recognition. Secondly, the adapter only relies on the global representation of the multi-view features. Motivated by this observation, we propose a Pretrained Point Cloud to Image Translation Network (PPCITNet) that produces generalized colored images along with additional salient visual cues to the point cloud depth maps so that it can achieve promising performance on point cloud classification and understanding. In addition, we propose a novel viewpoint adapter that combines the view feature processed by each viewpoint as well as the global intertwined knowledge that exists across the multi-view features. The experimental results demonstrate the superior performance of the proposed model over existing state-of-the-art CLIP-based models on ModelNet10, ModelNet40, and ScanobjectNN datasets.

Keywords: Contrastive Language-Image Pre-Training · Point Cloud Classification · Few shot Learning

1 Introduction

Point cloud understanding refers to the process of extracting meaningful information from 3D point clouds, which are sets of 3D coordinates representing the surface geometry of objects or scenes. The goal of point cloud understanding is to analyze and interpret the data contained in the point cloud in order to understand the objects or scenes that it represents. Point cloud understanding

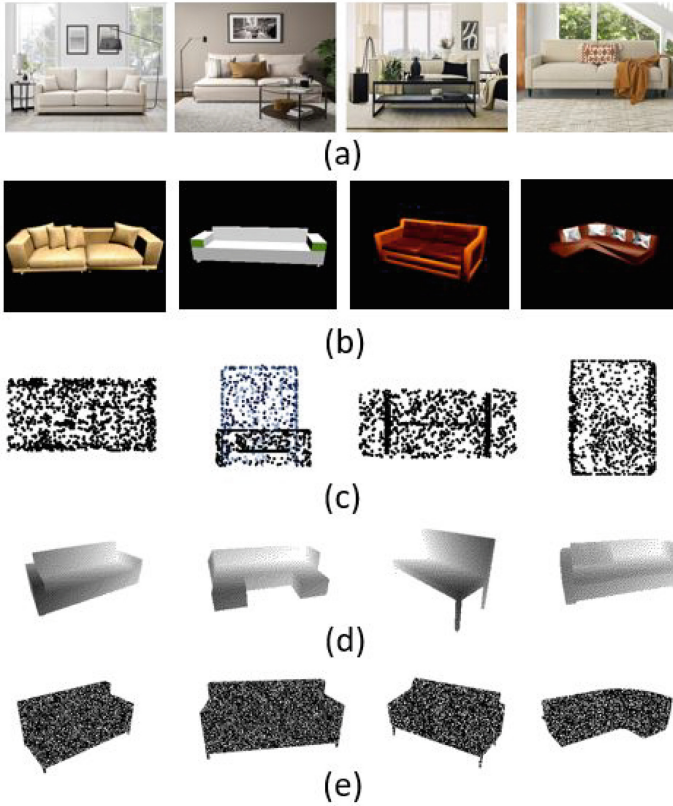


Fig. 1. Example of different image representations: (a) natural RGB images; (b) rendered RGB images; (c) point cloud depth maps; (d) 3D depth maps; (e) processed binary mask images.

has various applications in the real world, such as stereo reconstruction, indoor navigation, autonomous driving, augmented reality, and robotics perception etc. Although both 2D image understanding and point cloud understanding involve analyzing visual data, compared to the 2D image understanding [14], 3D point cloud understanding [11] is more challenging. A 2D image consists of a dense and regular pixel array. In contrast, a 3D point cloud only consists of sparse and unordered points in the 3D space. Moreover, point clouds often lack the rich texture and image information available in 2D images.

The success of deep learning in computer vision has also accelerated deep learning-based point cloud understanding and 3D-related research. While early deep learning methods had tried to propose some advanced architectures like PointNet [12], PointNet++ [13], RSCNN [9], DGCNN [18], CurveNet [20], the success of Contrastive Language-Image Pre-Training (CLIP) model has added new direction in the context of computer vision. CLIP has several advantages

over traditional deep learning methods. Firstly, the CLIP model is trained in a more generalizable manner, learning to associate images with natural language text in a way that can be applied to a wide range of downstream tasks. Whereas, traditional deep learning models are typically trained on specific tasks, such as image classification or object detection, and their performance can degrade significantly when applied to new, unseen tasks. Secondly, the CLIP model is trained on a large, unlabeled dataset of image-caption pairs, which does not require labeling efforts. On the other hand, traditional deep learning models often require large amounts of labeled training data to achieve good performance on a specific task. Finally, most importantly the CLIP model can be fine-tuned on new datasets and tasks with minimal additional training, making it a more flexible and adaptable solution to the downstream tasks compared to traditional deep learning models. Recently following the CLIP’s success on image and natural language domain, several works have been proposed to generalize pre-trained clip to 3D recognition. Some of these works focus on designing a small adapter network to CLIP [6, 23] and fine-tuning it for the downstream task. Other works focus on LLM-assisted 3D prompting and realistic shape projection [24] and cross-modal training framework [22] to bridge the gap between 2D image and point cloud. In general, the pipeline is as follows. Given a point cloud, the point cloud is first projected as a depth map. The depth map is then processed by the pre-trained CLIP visual encoder [14]. A small network called an adapter is added and fine-tuned for the downstream task.

Although these methods show some promising performance, they have certain limitations. It is due to the fact that the CLIP [14] is trained on RGB images whereas these models utilize point cloud depth maps for the point cloud understanding. Inherently, RGB images and depth maps are quite different from one another as depicted in Fig. 1. Point cloud depth maps represent depth information as a set of 3D points in space, with each point having an x, y, and z coordinate. This information is useful for 3D reconstruction and robotics navigation and manipulation. On the other hand, RGB images consist of red, green, and blue color channels, and each pixel in the image is represented by a combination of intensity values for these channels and captures color and texture information that is important for tasks like classification, recognition, and localization. In summary, the image information missing in the depth maps leads to the degrading performance of the state-of-the-art CLIP-based [14] point cloud models.

To transfer the image information to the CLIP-based point cloud models, one naive solution can be designing a network that maps depth maps to the corresponding natural RGB images. But, there does not exist any dataset that has depth maps and natural RGB image correspondence. However, there exists a dataset that has depth maps and rendered RGB image correspondence. In this direction, the next solution can be designing a network that maps depth maps to the corresponding rendered RGB images. Here the problems are three-fold. Firstly, CLIP is trained on natural RGB images. Rendered images differ from natural images in terms of realism, lighting, and Complexity depicted by Fig. 1(a,

b). Secondly, for a single depth map, there can be many possible corresponding rendered RGB images. For example, for a depth map of a sofa, the synthetic color changes in various parts as depicted in Fig. 1(b) in multiple rendered image instances. Finally, a 3D model depth map differs from a point cloud depth map as showed by [16]. A 3D model depth map typically represents depth information as a grayscale image shown in Fig. 1(d), with darker regions indicating greater distance from the viewer. In contrast, a point cloud depth map represents depth information as a set of 3D points in space, with each point having an (x, y, z) coordinate as depicted by Fig. 1(c).

In order to transfer image information to the CLIP [14] based point cloud model, we propose a novel Pretrained Point Cloud to Image Translation Network (PPCITNet) that produces generalized colored images along with additional salient visual cues to the point cloud depth maps. Here, the salient visual cues refer to additional color concentration to prominent or distinctive parts like an additional color concentration in the head and legs of a person (see Fig. 4). The target of our PPCITNet is to provide image information to the CLIP [14] model so that it can achieve promising performance on point cloud classification and understanding. To pre-train this Point Cloud to Image Translation Network (PCITNet), we utilize the binary mask images of the rendered RGB images. Binary mask images and point cloud depth maps are similar geometrically because of discrete and compact representation. But visually, they are slightly different (see Fig. 1(d, e)). To further bridge the gap, we preprocess the binary mask images by multiplying the binary image with a noise image to make the binary image sparse. The noise image is composed of 50% white pixel and 50% of black pixel sampled randomly. Through PPCITNet, the depth features of the point cloud can then be well aligned with the visual CLIP features.

To further adapt our network to the few-shot learning, we proposed a novel viewpoint adapter that combines the local feature processed by each viewpoint as well as the global intertwined knowledge that existed across the multi-view features. In our opinion, the local viewpoint information is crucial for point cloud classification. For example, to classify the point cloud of ‘airplane’ the viewpoint that contains the wing information is more crucial than any other parts. In summary, the contributions of our paper are as follows. 1) We propose a novel Pretrained Point Cloud to Image Translation Network (PPCITNet) that transfers image information to the point cloud depth maps so that it can achieve promising performance on point cloud classification and understanding. 2) We propose a novel viewpoint adapter that combines the view feature processed by each viewpoint as well as the global intertwined knowledge existing across the multi-view features. 3) Our methods achieve state-of-the-art results on few-shot point cloud classification tasks on ModelNet10, ModelNet40, and ScanobjectNN.

2 Related Works

Deep Learning in Point Clouds. Deep learning has revolutionized the field of point cloud classification and understanding. Categorically, deep learning-based

models are divided into three sections, including multi-view based methods [5], volumetric-based methods [8] and point-based methods [12]. Early works on deep learning primarily focused on multi-view-based methods [5], where the 2D image models are utilized for point cloud classification. In volumetric-based methods [8], point clouds are treated as voxel data. 3D convolution-based models are used for classification and segmentation. The state-of-the-art models are point cloud-based methods [12], where the raw points are processed and passed through the model without any transformation. PointNet [12] is the first point-based model that has encoded each point with a multi-layer perception. PointNet++ [13] further utilizes the max pooling operation to ensure permutation invariance. Recently, the success of CLIP for the downstream tasks on 2D has motivated the use of pre-trained CLIP for point cloud classification. Zhang et al. [23] propose PointCLIP which generalizes pre-trained CLIP to 3D recognition.

CLIP-Based Point Cloud Models. Recently several works have been proposed to generalize pre-trained Contrastive Language-Image Pre-Training (CLIP) to point cloud understanding tasks. For example, Zhang et al. first proposed PointCLIP [23] by extending the CLIP [14] for handling 3D point cloud data. In addition, they presented an inter-view adapter to capture the feature interaction between multiple views. In this direction, Zhu et al. [24] further introduced an efficient cross-modal adaptation method called PointCLIP V2 by proposing LLM-assisted 3D prompting and realistic shape projection. Next, Huang et al. [6] presented a novel Dual-Path adapter and contrastive learning framework to transfer CLIP knowledge to the 3D domain. Yan et al. [22] presented PointCMT, an point cloud cross-modal training framework that utilized the merits of color-aware 2D images and textures to acquire more discriminative point cloud representation and formulated point cloud analysis as a knowledge distillation problem.

3 Methodology

In this section, we first briefly revisit PointCLIP [23] for few-shot 3D classification (Sect. 3.1). Then we introduce our Pretrained Point Cloud to Image Translation Network (PPCITNet) framework (Sect. 3.2) that aligns image information to the point cloud depth map. Finally, we describe our proposed few-shot learning framework for few-shot point cloud classification (Sect. 3.3). The overall overview of our method is depicted in Fig. 3.

3.1 Revisit of PointCLIP

Similar to CLIP [14] which matches images and text by contrastive learning, PointCLIP [24] consists of one visual encoder and a textual encoder. For K class classification, PointCLIP uses a pre-defined template: “point cloud depth map of a big [CLASS]” and the textual encoder outputs $P \in R^{K \times C}$, where C is the channel of the text embedding. To feed point clouds to the CLIP’s visual encoder [14], point clouds are first projected onto depth maps $\{f_1, f_2, \dots, f_M\}$.

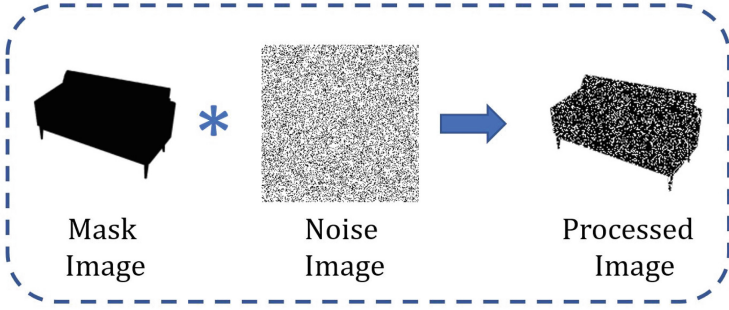


Fig. 2. For a binary mask image, we multiply the binary image with a noise image to make the binary image sparse. The noise image is composed of 50% white pixel and 50% of black pixel sampled randomly.

Here M denotes the number of views and $f_i \in R^{H \times W \times C}$ denotes each view of the point cloud, where H and W denote height and width respectively. Given the input $\{f_1, f_2, \dots, f_M\}$, the visual encoder in PointCLIP generates visual feature $\{F_1^I, F_2^I, \dots, F_M^I\}$, where $F_i^I \in R^{1 \times C}$ and C is the channel dimension of the embedding.

Zero-Shot Classification: In a zero-shot setup, there is no training stage. Each viewpoint generates a prediction by calculating the cosine similarity between the visual feature F_i^I and the textual feature P^T . The final prediction is the weighted sum of all viewpoint-wise predictions. Thus,

$$O_i = F_i^I P^T, i = 1, 2 \dots N \quad (1)$$

$$\hat{y} = \text{softmax}\left(\sum_{i=1}^N \alpha_i O_i\right) \quad (2)$$

where α_i is a hyper-parameter that describes the weighting importance of the view i .

Few-Shot Classification: For few-shot point cloud classification, PointCLIP proposes an inter-view adapter. The inter-view adapter extracts the global visual representation by combining the multi-view features produced by the visual encoder of PointCLIP. The global representation is then added back to the adapted features F_i^I . Thus, the adapter can be formulated as follows:

$$G = f_2(\text{ReLU}(f_1(\text{concat}(F_{i=1}^M)))) \quad (3)$$

$$F^g = \text{ReLU}(GW^T) \quad (4)$$

$$\hat{y} = \text{softmax}\left(\sum_i^M \alpha_i ((F_i^I + F^g)\{P^T\}^T)\right) \quad (5)$$

where P denotes textual information, α_i is a hyper-parameter that denotes importance of view i , W denotes learnable weights, and f_1, f_2 are MLP layers.

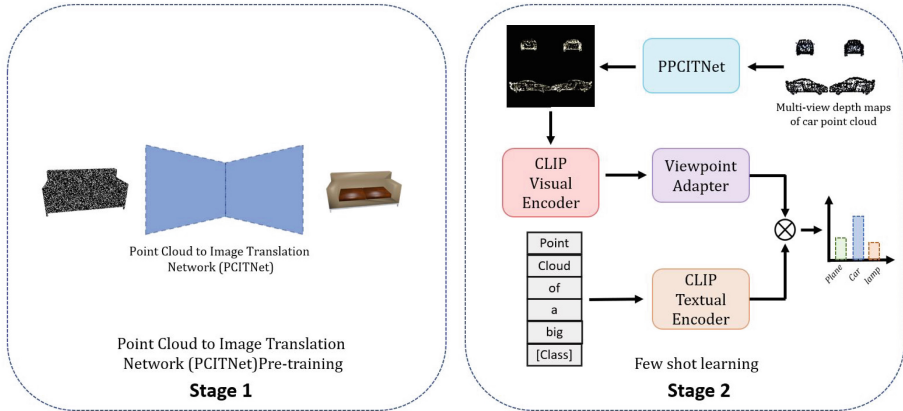


Fig. 3. The training of our approach is composed of two stages. In the first stage, we pre-train our PCITNet using the processed binary mask and RGB pairs. In the second stage, we perform a few shot learning on a viewpoint adapter utilizing PPCITNet and pre-trained CLIP.

3.2 Point Cloud to Image Translation Network Pre-training

Instead of directly applying CLIP [14] visual encoder to depth maps, we propose to learn a Point Cloud to Image Translation Network (PCITNet) for aligning point cloud depth features with CLIP visual features. In other words, we expect the extracted features of a rendered point cloud depth map to be consistent with the CLIP visual features of the corresponding image. Then CLIP [14] textual prompts can be directly adopted to match the depth features. Let $S = \{B_i, R_i\}_{i=1}^L$ denotes a pre-training dataset with L instances. Here B_i is a binary mask image and R_i denotes its corresponding rendered RGB image. We would like to learn a network $F_\theta(\cdot)$ that maps from a binary mask image to a rendered RGB image as follows:

$$\hat{R} = F_\theta(B) \quad (6)$$

Our goal is to learn the PCITNet F_θ that represents generalized image color distribution along with additional salient visual cues. As discussed earlier, binary mask images and point cloud depth maps are similar geometrically because of their discrete and compact representations. But visually, they are slightly different. To further bridge the gap, we pre-process a binary mask image by multiplying the binary image with a noise image to make the binary image sparse as depicted in Fig. 2. The noise image is composed of 50% white pixel and 50% of black pixel sampled randomly. To learn the generalized image information along with additional salient visual cues, we optimize the following objective function:

$$\mathcal{L}_c = \frac{1}{L} \sum_{i=1}^L (R_i - \hat{R}_i)^2 \quad (7)$$

Here, L is the total number of mask-RGB pairs in the dataset. The generalized image information along with additional salient visual cues information helps to encode a richer and more diverse set of visual features that can be used to discriminate between different objects. Without image information, CLIP [14] may have difficulty distinguishing between objects with similar shapes. For example, consider the task of classifying chairs based on their shape alone. Chairs have similar shape features to tables such as legs. Based on the shape alone, it is very difficult for CLIP [14] to distinguish between them. However, by incorporating image information in the classification process, we can identify additional features that can help differentiate between chairs and tables as the image information provides additional cues for the CLIP [14] as described by Bramaio et al. [1].

3.3 Few-Shot Learning

Settings. Let $\rho \in R^{P \times 3}$ denote the point cloud, where P denotes the number of points of the point cloud sample from the $N \times K$ few shot data. Here, N is the total number of classes and each class has K instances of point cloud. Given the PPCITNet and pre-trained CLIP [14] network, the goal is to train the viewpoint adapter so that it can boost the performance of the CLIP-based point cloud classification network.

Feature Extraction. For each $\rho \in R^{P \times 3}$, we need to project 3D coordinates to 2D coordinates. Following [6], we get the point cloud depth maps $f^d = \{f_1, f_2, \dots, f_M\}$. These depth maps are first passed through the PPCITNet, then the output feature is passed through CLIP’s visual encoder. The goal of our PPCITNet is to provide generalized image information along with additional salient visual cues to the CLIP model so that it can achieve promising performance on point cloud classification and understanding.

$$f^c = F_\theta(f^d), i = 1, 2 \dots M \quad (8)$$

$$f^v = F_V(f^c), i = 1, 2 \dots M \quad (9)$$

where i indicates the number of depth maps of a 3D point cloud captured from different perspectives, f_i^v denotes output for f_i^d depth map, f_i^c denotes generalized colored images, F_θ and F_V denote the PPCITNet and CLIP’s visual encoder [6] respectively.

Viewpoint Adapter. We propose a novel viewpoint adapter that combines the view feature processed by each viewpoint as well as the global intertwined knowledge that exists across the multi-view features. Given the extracted feature $f^v = \{f_1, f_2, \dots, f_M\}$, the view-specific view information is calculated using M MLP layers. Thus,

$$f_i^l = \phi(f_i^v W_{li}) \quad (10)$$

where W_{li} is the weight of an MLP layer and ϕ denotes the activation function. f_i^l captures the view-specific fine-grained visual features and generalized

image information along with additional salient visual cues that are relevant to a particular point cloud object. For example, to classify the point cloud of an airplane, the viewpoint that contains the wing information is more crucial than any other part. f_i^l encodes fine-grained wing information for the point cloud of the airplane. To get the global information of the M views, we perform the following operation:

$$f^g = \phi(\text{concat}(f_{i=1}^v{}^M)W_{g1}^T)W_{g2}^T \quad (11)$$

where $f^g \in R^{1 \times C}$ and $W_{g1} W_{g2}$ denote the two-layer weights in the viewpoint adapter. Here, the global knowledge captures the overall structure and organization of point clouds and provides a more holistic understanding of the point cloud objects. Finally, the classification is performed as follows:

$$\text{logits} = \text{softmax}(\sum_i^N \alpha_i ((f_i^l + f^g)\{P^T\}^T)) \quad (12)$$

where α_i denotes the learnable weight, and P denotes textual information. Note that, Only the viewpoint adapter is trained in few-shot learning. The features learned by the viewpoint adapter provide complementary information about the overall structure and view-specific fine-grained features of point cloud objects combining both view and global information.

4 Experiments

Pre-training Datasets. To pre-train our PCITNet network, we use the DISN 2D dataset released by Wang et al. [21]. This dataset is based on the ShapeNet Core dataset [2], which is a 3D dataset consisting of 13 object categories. While early work [4] of rendering this dataset utilizes 24 views with limited variation in terms of camera orientation for each model, DISN provides two types of settings: “easy” and “hard”. The easy setting consists of 36 renderings with smaller variations, The hard setting is composed of 36 renderings with larger variations. To train our PCITNet network, we sample 100k data from the easy setting randomly. From the RGBA image, we sample the mask image.

Downstream Datasets. Following PointCLIP [23], we evaluate our proposed model on three widely used benchmark datasets: ModelNet10 [19], ModelNet40 [19] and ScanObjectNN [17]. ModelNet10 and ModelNet40 have a training point cloud set of 3991 and 9,843 and a test point cloud set of 908 and 2,468 respectively. ScanObjectNN is a real-world point cloud dataset that includes 2,321 samples for training and 581 samples for testing the point cloud from 15 categories. Compare to the ModelNet, ScanObjectNN is more challenging because the CAD models are attached with backgrounds and partially presented. For all three datasets, we uniformly sample 1,024 points of each object as the PCITNet’s input.

Implementation Details. We use Unet architecture from [15] as our PCITNet network. To pre-train the PCITNet network, we resize the image to 224×224

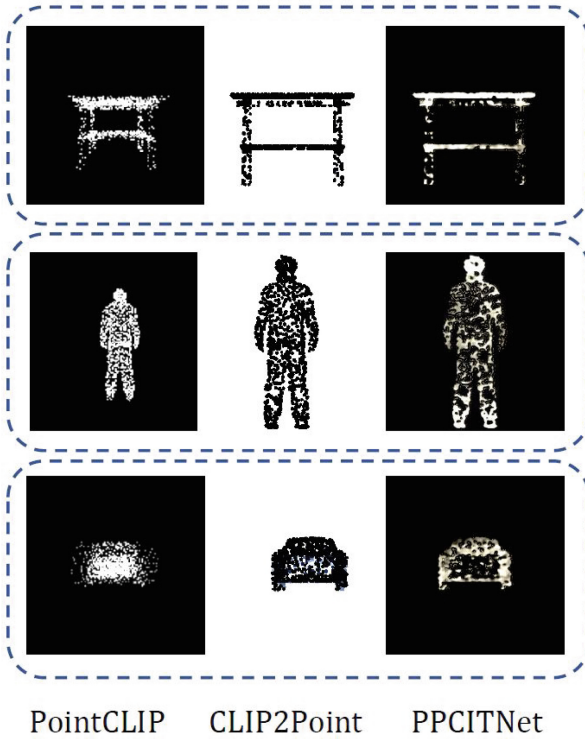


Fig. 4. Input Visualization. Our PPCITNet produces generalized colored images along with additional salient visual cues. The salient visual cues refer to additional color concentration to prominent or distinctive parts of the image.

and train our model in a 12 GB Nvidia Titan X GPU using PyTorch. In pre-training, we use the Adam optimizer [7] with decay of 1×10^{-4} and the initial learning rate of 1×10^{-3} . Our pre-training task takes 100 epochs with a batch size of 16. For few-shot learning, we utilize AdamW optimizer [10] with decay of 1×10^{-4} and the initial learning rate of 1×10^{-3} . The training batch size is 32 and it takes 100 epochs to train the network. Similar to [6, 23], we use the 6 orthogonal views: left, right, top, bottom, front, and back for few-shot learning.

4.1 Results

CLIP-based models [6, 23] are generally evaluated by comparing with state-of-the-art methods on few-shot learning and prompt engineering. In Table 1, we present the zero-shot and few-shot performance of PPCITNet on ModelNet40 using the prompt “point cloud of a big [CLASS]”. In Table 2, we present the few shot performance of PPCITNet and compare it with state-of-the-art 3D networks like PointNet [12], PointNet++ [13], CurveNet [20], SimpleView [3] as well as CLIP based models PointCLIP [23], CLIP2Point [6] on 16 shot setup.

As we can see from the table, PPCITNet with a viewpoint adapter outperforms PointCLIP and CLIP2Point by a margin of 3–5 % for 16 shot setup for prompt “point cloud of a big [CLASS]” on all three datasets. To further evaluate the transfer ability of PPCITNet, we show the performance for 1, 2, 4, 8, 10, 12, 16 shots in Fig. 5.

Table 1. Zeroshot and Few-shot results of PPCITNet on ModelNet40 using the prompt “point cloud of a big [CLASS]”.

Setup	Accuracy
Zeroshot	22.74
Few-shot	88.93

We can see from the graph, our PPCITNet surpasses all by a reasonable good margin. This is due to the additional visual cues provided by PPCITNet and the view and global information encoding of the viewpoint adapter. The large performance gain on ScanObjectNN indicates the robustness of PPCITNet under noisy real-world scenes.

Table 2. Performance (%) of PPCITNet with other methods in 16-shot setup using prompt “point cloud of a big [CLASS]”.

Model	ModelNet10	ModelNet40	ScanObjectNN
CurveNet	82.45	76.55	34.76
SimpleView	84.15	71.17	37.44
PointNet	73.98	67.34	36.18
PointNet++	84.62	77.13	51.62
PointCLIP	89.33	83.80	54.37
CLIP2Point	90.21	85.10	57.49
PPCITNet	94.30	88.93	63.22

The visualization in Fig. 4 further establishes our claims. While PointCLIP and CLIP2Point provide uniformly sampled point features to the CLIP’s visual encoder, our PPCITNet produces generalized colored images along with additional salient visual cues. Here, the salient visual cues refer to additional color concentration to prominent or distinctive parts like an additional color concentration in the head and legs of the human in Fig. 4. In Table 3, we compare our PPCITNet with PointCLIP for different prompt designs on ModelNet40, where [CLASS] represents the class token and ‘[Learnable Tokens]’ refers to the prompts with a fixed length that are capable of being learned during training. The large performance gain indicates the generability of our PPCITNet over PointCLIP for various prompt designs.

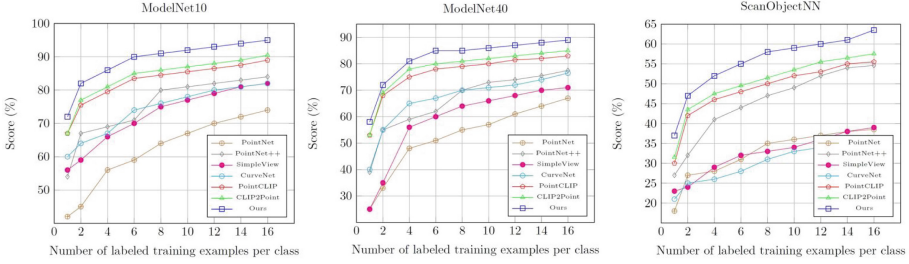


Fig. 5. Few-shot performance comparison under 1, 2, 4, 8, 10, 12, 14, and 16-shot settings.

Table 3. Performance (%) of PPCITNet with PointCLIP for different prompt designs on ModelNet40.

Prompts	PointCLIP	PPCITNet
“a photo of a [CLASS]”	81.78	86.63
“a point cloud photo of a [CLASS]”	82.02	87.33
“point cloud of a [CLASS]”	82.10	87.04
“point cloud of a big [CLASS]”	83.80	88.93
“point cloud depth map of a [CLASS]”	81.58	85.15
“[Learnable Tokens] + [CLASS]”	69.23	76.27

4.2 Ablation Studies

In this section, we evaluate the effect of our PPCITNet and the effect of view information on the viewpoint adapter. To observe the effect of PPCITNet, we conduct an experiment with PPCITNet and without PPCITNet on ModelNet40 as shown in Table 4.

Table 4. Effect of PPCITNet on ModelNet40 using prompt “point cloud of a big [CLASS]”.

Model	Accuracy
Without PPCITNet	84.27
With PPCITNet	88.93

From the table, it is evident that incorporating PPCITNet on the few-shot pipeline improves accuracy by 4.6 %. To analyze the view feature, we conduct experiments with the only view feature, with only global information, and with both view information and global information on PPCITNet on ModelNet40.

Although the performance drops significantly while utilizing only the view information, a combination of view and global information yields the best performance, specifically an improvement of 1.3% over global information as described in Table 5.

Table 5. Effect of view information for PPCITNet on ModelNet40 using prompt “point cloud of a big [CLASS]”.

View info.	Global info.	Accuracy
✓	–	82.34
–	✓	87.60
✓	✓	88.93

5 Conclusion

In conclusion, we present a novel pretrained point cloud to image translation network that transfers image information to the point cloud depth maps. In addition, we present a novel viewpoint adapter that combines the view feature processed by each viewpoint as well as the global intertwined knowledge existing across the multi-view features. The experiment results validate the superior performance of our approach compared to the other state-of-the-art models on the few-shot point cloud classification.

References

1. Bramão, I., Reis, A., Petersson, K.M., Faísca, L.: The role of color information on object recognition: a review and meta-analysis. *Acta Physiol. (Oxf)* **138**(1), 244–253 (2011)
2. Chang, A.X., et al.: ShapeNet: an information-rich 3D model repository. arXiv preprint [arXiv:1512.03012](https://arxiv.org/abs/1512.03012) (2015)
3. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: *International Conference on Machine Learning*, pp. 1597–1607. PMLR (2020)
4. Choy, C.B., Xu, D., Gwak, J., Chen, K., Savarese, S.: 3D-R2N2: a unified approach for single and multi-view 3D object reconstruction. In: *Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016, Proceedings, Part VIII 14*, pp. 628–644. Springer (2016)
5. Feng, Y., Zhang, Z., Zhao, X., Ji, R., Gao, Y.: GVCNN: group-view convolutional neural networks for 3D shape recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 264–272 (2018)
6. Huang, T., et al.: CLIP2Point: transfer CLIP to point cloud classification with image-depth pre-training. arXiv preprint [arXiv:2210.01055](https://arxiv.org/abs/2210.01055) (2022)
7. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. In: *ICLR* (2015)
8. Li, Y., Pirk, S., Su, H., Qi, C.R., Guibas, L.J.: FPN: field probing neural networks for 3D data. In: *Advances in Neural Information Processing Systems 29* (2016)

9. 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)
10. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: ICLR (2019)
11. Meng, Q., Wang, W., Zhou, T., Shen, J., Jia, Y., Van Gool, L.: Towards a weakly supervised framework for 3D point cloud object detection and annotation. *IEEE Trans. Pattern Anal. Mach. Intell.* (2021)
12. 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)
13. 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 30 (2017)
14. Radford, A., et al.: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning, pp. 8748–8763. PMLR (2021)
15. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015: 18th International Conference, Munich, Germany, 5–9 October 2015, Proceedings, Part III 18, pp. 234–241. Springer (2015)
16. Rothganger, F., Lazebnik, S., Schmid, C., Ponce, J.: 3D object modeling and recognition using local affine-invariant image descriptors and multi-view spatial constraints. *Int. J. Comput. Vis.* **66**(3), 231–259 (2006)
17. 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: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 1588–1597 (2019)
18. 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)
19. Wu, Z., et al.: 3D ShapeNets: a deep representation for volumetric shapes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1912–1920 (2015)
20. Xiang, T., Zhang, C., Song, Y., Yu, J., Cai, W.: Walk in the cloud: learning curves for point clouds shape analysis. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 915–924 (2021)
21. Xu, Q., Wang, W., Ceylan, D., Mech, R., Neumann, U.: DISN: deep implicit surface network for high-quality single-view 3D reconstruction. In: Advances in Neural Information Processing Systems 32 (2019)
22. Yan, X., et al.: Let images give you more: point cloud cross-modal training for shape analysis. arXiv preprint [arXiv:2210.04208](https://arxiv.org/abs/2210.04208) (2022)
23. Zhang, R., et al.: PointCLIP: point cloud understanding by CLIP. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8552–8562 (2022)
24. Zhu, X., Zhang, R., He, B., Zeng, Z., Zhang, S., Gao, P.: PointCLIP V2: adapting CLIP for powerful 3D open-world learning. arXiv preprint [arXiv:2211.11682](https://arxiv.org/abs/2211.11682) (2022)



BarBeR: A Barcode Benchmarking Repository

Enrico Vezzali¹, Federico Bolelli¹(✉), Stefano Santi², and Costantino Grana¹

¹ University of Modena and Reggio Emilia, Modena, Italy
{enrico.vezzali,federico.bolelli,costantino.grana}@unimore.it

² Datalogic, S.p.A, Bologna, Italy
stefano.santi@unimore.it

Abstract. Since their invention in 1949, barcodes have remained the preferred method for automatic data capture, playing a crucial role in supply chain management. To detect a barcode in an image, multiple algorithms have been proposed in the literature, with a significant increase of interest in the topic since the rise of deep learning. However, research in the field suffers from many limitations, including the scarcity of public datasets and code implementations, which hampers the reproducibility and reliability of published results. For this reason, we developed “BarBeR” (Barcode Benchmark Repository), a benchmark designed for testing and comparing barcode detection algorithms. This benchmark includes the code implementation of various detection algorithms for barcodes, along with a suite of useful metrics. It offers a range of test setups and can be expanded to include any localization algorithm. In addition, we provide a large, annotated dataset of 8748 barcode images, combining multiple public barcode datasets with standardized annotation formats for both detection and segmentation tasks. Finally, we share the results obtained from running the benchmark on our dataset, offering valuable insights into the performance of different algorithms.

Keywords: BarBeR · Barcodes · Benchmark · QR Codes · Public Dataset

1 Introduction

Barcodes, a prevalent form of machine-readable data representation, have revolutionized the accuracy and speed of data collection and identification [36]. Their cost-effectiveness and efficiency have led to their widespread use in various engineering applications. First of all, barcodes serve as a cornerstone of supply chain management [24], facilitating the flow of goods from manufacturers to consumers by enabling efficient inventory tracking and logistics management. Secondly, barcodes are extensively used in warehouses to automate the process of goods receipt, storage, and dispatch, helping in reducing manual errors and improving the speed of operations [19]. Other notable applications are component tracking in manufacturing, product recognition in retail [25], and robot

guidance [29]. Despite their inception over seven decades ago, barcodes continue to hold their ground in today’s digital age, and their use is forecasted to increase in the future [17]. This is reflected in the projected growth of the barcode reader market, which was valued at \$7.4 billion in 2022 and is expected to reach \$13.3 billion by 2032, growing at a CAGR of 6.3% from 2023 to 2032 [33]. Barcodes come in two categories: one-dimensional (1D or linear) and two-dimensional (2D). Linear barcodes encode data with lines of varying widths and spacing, but have limited data storage capacity. To overcome this issue, 2D barcodes were introduced. Their structure allows data to be stored on both vertical and horizontal axes, offering greater capacity compared to 1D barcodes [32]. The process of reading a barcode can usually be divided into two macro steps: localization and decoding. While some papers focus on both steps [10, 18] most of the publications just focus on the localization part [30, 38, 41]. Especially in recent times, it has become the norm to use public third-party libraries to handle the decoding step [37]. The two most used libraries are ZXing¹ and Zbar.² Each software tool can handle both 1D and 2D barcodes. Therefore, our primary focus from now on will be on localization. Until recently, real-time speed for a localization algorithm was achievable solely through the computation of hand-crafted features from the image. However, the recent advancements in edge deep learning fueled the interest in developing barcode localization solutions based on deep learning. Between the years 2015 and 2021, 25 publications introduced a method for barcode localization (either 1D, 2D, or both) that utilized deep learning techniques [37]. Despite the huge interest in the field, several issues prevent definitive conclusions about methods’ effectiveness and applicability. The first is that existing research relies on small datasets that do not reflect real-world scenarios accurately and make training deep learning models difficult. Then there is the problem of reproducibility. The lack of public code implementations makes replicating results challenging. Finally, different studies use different metrics, leading to contradictory comparisons even with identical algorithms and datasets.

To address these challenges, we have developed “BarBeR” (Barcode Benchmark Repository)—an open-source benchmark for barcode localization with standardized test protocols and evaluation metrics. BarBeR contains the implementation of multiple localization algorithms tailored for barcodes that we selected after a thorough review of the literature. In addition, we are publicly releasing a large annotated dataset of 8748 images of barcodes to be used with our benchmark. Our aim is to enhance reproducibility and facilitate more reliable algorithm comparisons within the research community.

2 Related Works

Early Barcode Localization Efforts. Joseph Woodland and Bernard Silver invented the linear barcode in 1949 and patented it in 1952. Early decoding

¹ <https://github.com/zxing/zxing>.

² <https://github.com/ZBar/ZBar>.

Table 1. List of the public datasets collected for the benchmark. The table reports the number of images per dataset and the resolution of the image with the minimum and the maximum number of pixels in the dataset respectively. # 1D and # 2D represent the number of linear and two-dimensional barcode instances in each dataset.

Dataset Name	# Images	Minimum Resolution	Maximum Resolution	# 1D	# 2D
Arte-Lab Medium 1D [39]	430	1 152 × 864	2 976 × 2 232	430	7
Arte-Lab Extended 1D [40]	155	648 × 488	648 × 488	165	3
Bodnár-Huawei QR [3]	98	1 600 × 1 200	1 600 × 1 200	0	98
DEAL KAIST Lab [7]	3 308	141 × 200	3 480 × 4 640	3 378	76
Dubská QR [8]	810	402 × 604	2 560 × 1 440	0	806
InventBar [16]	527	480 × 640	480 × 640	530	33
Muenster 1D [35]	1 055	1 600 × 1 200	2 592 × 1 944	1 068	1
OpenFood Facts [1]	185	390 × 520	5 984 × 3 376	187	5
ParcelBar [16]	844	1 108 × 1 478	1 478 × 1 108	1 196	17
Skku Inyong DB [38]	325	1 440 × 2 560	1 440 × 2 560	368	10
Szentandrási QR [31]	90	1 024 × 768	4 752 × 3 168	0	225
ZVZ-Real [41]	921	407 × 576	3 288 × 4 930	740	475
Total	8 748	200 × 141	5 984 × 3 376	8 062	1 756

methods relied on analog circuits, with laser scanners being the primary decoding method in the ‘70s. However, these systems required the reader to be directly aimed at the barcode. The 1990s saw the advent of 2D image barcode reading. A significant advantage of this approach is the ability to read a barcode from a wider field of view, but to do so, the barcode must first be located. Detection methods for linear barcodes included Sobel filters for texture analysis [34], Gabor filters [13], and even early machine learning techniques for texture classification [14]. The Hough Transform also gained popularity for linear barcodes [26], and gradient analysis was used for QR codes [27].

Recent Approaches. Research continued to address limitations and expand barcode localization applications. Methods explored skeletonization [4] and texture direction analysis [12], while the use of the Hough Transform was extended to the 2D barcodes [31]. Huge efforts were directed into increasing the algorithm’s speed, to allow the use on mobile phones [10].

The Deep Learning Era. Chou’s 2015 work marked a turning point with the introduction of CNNs for QR code detection [6]. Deep learning has since become dominant, with notable successes using YOLO [11] and Faster R-CNN [20]. Many also proposed custom CNN architectures adapted to the task [41].

3 Dataset

For this project, we required a large dataset to accurately compare algorithms and train object detection neural networks. For this reason, we conducted a thorough literature review to identify publicly available datasets of barcodes. Table 1 lists these datasets and their sources. The collected datasets account for a total of 8 748 images with 9 818 annotated barcodes, 8 062 linear, and

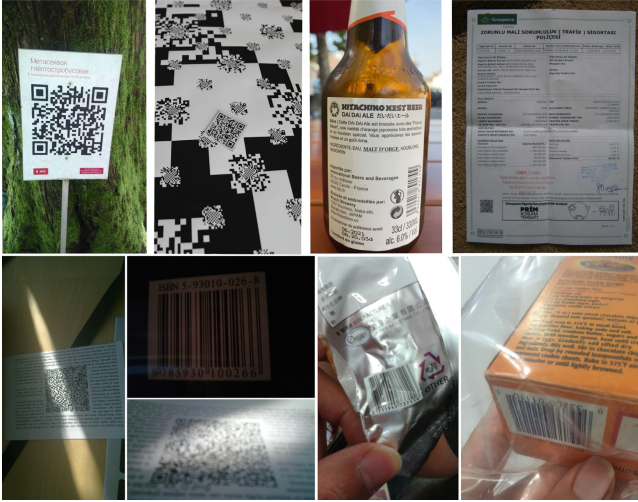


Fig. 1. Example of images taken from the proposed dataset. Multiple types of items are portrayed in different settings. In addition, we have some examples of hard cases, such as confusing patterns, small bounding boxes, variable lighting, underexposure, and blur. Some barcodes are also non-planar or partially obstructed.

1756 two-dimensional. A significant challenge was the lack of annotations in some datasets and the wide variation in annotation formats. To address this, we generated new annotations for all images using Datalogic’s proprietary software, which generates a 4-point polygon for each barcode read and provides additional information, such as its type, and the encoded string. In addition, we have information about the pixel density of the barcode, usually measured in pixels per element (PPE), i.e., the mean width of the smallest element in a barcode. This measure can also be referred to as pixels per module (PPM). While most codes were annotated in this way (8096), a few (1722) were un-decodable due to blur, noise, or incorrect scale. These codes were manually annotated, and thus they lack some information like the PPE. Since the annotations use polygons instead of boxes, they are suitable for both detection and segmentation.

The final dataset presents an extensive diversity of subjects and environments. It contains barcodes of 18 categories, 14 of which are considered linear symbologies (Code 128, Code 39, EAN-2, EAN-8, EAN-13, GS1-128, IATA 2 of 5, Intelligent Mail Barcode, Interleaved 2 of 5, Japan Postal Barcode, KIX-code, PostNet, RoyalMail Code, and UPC) and 4 are considered 2D symbologies (Aztec, Datamatrix, PDF417, and QR Code). The images have been captured with different devices such as a Nokia N95 [35], a Huawei Smartphone from 2014 [3], and a 15MP professional camera [8]. The dataset also features different settings and subjects. Skku Inyong DB [38] was captured inside a supermarket and represents items found there. DEAL KAIST Lab [7] also represents market items, but the settings change widely, some indoors, others outdoors. Dubska [8]

dataset represents mostly QR codes printed on paper, captured indoors in a controlled setting. ZVZ-Real [41] is one of the most diverse datasets in the collection, with images taken indoors and outdoors, with subjects ranging from market items, product labels, receipts, and letters as well as book photos and scans. In addition, our dataset contains both planar barcodes and skewed or warped barcodes. The dataset contains barcodes in different lighting conditions, some are underexposed or overexposed, and others have variable lighting throughout the code. Other codes have specular reflections. Finally, some codes are affected by blur and noise or are partially covered or obstructed, as shown in Fig. 1.

4 Benchmark Description

As part of this project, we have developed BarBeR, a benchmark for barcode localization algorithms available on GitHub.³ It includes various detection methods and scripts to train neural networks for barcode detection. Our dataset, used for running our tests, can be downloaded from the same GitHub repository or from our website.⁴

4.1 Tests and Metrics

The repository is equipped with a variety of test scripts, each supporting diverse configurations. Here is a breakdown of the test scripts and their main configuration parameters:

- **Single Class Detection:** runs all the selected algorithms considering only images with the selected type of barcodes. It can be tailored to permit only linear or two-dimensional barcodes. It is also possible to include only images with a single Region Of Interest (ROI) or multiple ROIs per image. In addition, we can decide the target resolution used to rescale the images in the test set. Finally, we can specify which algorithms to use in the test and with which arguments;
- **Multi-Class Detection:** runs all the selected algorithms on all the images of the test set. As for Single Class Detection, we can choose the resizing resolution and which algorithms are included in the test;
- **Timing Performance:** measures the time required to run the algorithms. The times can be taken from the average times on all datasets or a subsection of it. It is possible to measure the algorithms’ performance on a single core or multiple cores as well as on GPU.

All test scripts are written in Python and take as input argument a YAML configuration file and output a YAML file containing multiple metrics for every tested algorithm. The available metrics are precision, recall, and F1-score at different IoU scores. For algorithms that also output a confidence score, the Benchmark

³ <https://github.com/Henzezz95/BarBeR>.

⁴ <https://ditto.ing.unimore.it/barber>.

Table 2. Characteristics of the deep-learning models used in our tests. Two-stage detectors first propose regions, then classify and refine bounding boxes. One-stage detectors perform detection and classification in a single step.

Network	Type	Backbone	# Parameters [M]	GFlops @ (640 × 640)
Zharkov <i>et al.</i> [41]	One-Stage	dilated-net	0.0424	1.528
Faster R-CNN [28]	Two-Stage	resnet50_fpn_3x	41.755	134.38
RetinaNet [21]	One-Stage	resnet50_fpn_3x	34.014	151.54
YOLO-v8 [15]	One-Stage	yolov8 medium	25.903	39.66
YOLO-v8 Nano [15]	One-Stage	yolov8 nano	3.157	4.429
RT-DETR [23]	One-Stage	HGNetv2-L	31.005	54.17

computes the Average Precision (AP@.5, AP@[.5:.95]) for each class, the mean Average Precision (mAP@.5, mAP@[.5:.95]) and the Average Recall (AR100, AR10, AR1). Finally, the benchmark allows to filter these metrics depending on the size of the ground truth and its pixel density. The repository also contains bash scripts used to run a pipeline of tests. This is useful, for example, for k-fold cross-validation.

4.2 Available Localization Methods

Gallo *et al.* The localization method proposed by Gallo and Manduchi in 2011 is a rapid algorithm that localizes a single 1D barcode per image. It assumes the barcode is horizontally positioned with vertically aligned parallel lines and is not rotation invariant. The process begins by calculating a heatmap $I_e(n)$, representing the difference between the magnitudes of the horizontal and vertical derivatives. After smoothing and binarizing the heatmap, the blob containing the pixel that maximizes $I_e(n)$ is used to compute the barcode’s bounding box.

Soros *et al.* This algorithm was proposed in 2013 by Sörös and Flörkemeier. It is a method designed for both 1D and 2D barcodes that is orientation invariant and is quite resistant to blur [30]. However, this method can only output a single ROI for each barcode type. It is based on the UNIVAR detector and OMNIVAR detector proposed by Ando [2]. The first detector finds areas with strong unidirectional edges and can be used to find linear barcodes, while the latter can find corners and is useful for 2D barcode localization.

Zamberletti *et al.* The method introduced by Zamberletti *et al.* in 2013 is capable of detecting multiple linear barcodes. It generates several rotated boxes, all sharing the same angle of rotation. This proves beneficial in scenarios where a single label contains multiple barcodes, each exhibiting the same rotational angle. It uses a multi-layer perceptron to process the Hough Transform of the image and predict the angle of the barcodes in the image. Once the angle is found, the technique of Galamhos *et al.* [9] is used to find all lines with that angle of orientation. Finally, the areas with the highest concentration of these lines are located using a method based on histograms and labeled as barcodes.

Table 3. Precision, Recall and F1-score with an IoU threshold of 0.5. Employed images contain a single 1D barcode and were resized to have their longest side of 640 pixels.

Detection Method	Precision \uparrow	Recall \uparrow	F1-score \uparrow
Gallo <i>et al.</i> [10]	0.533	0.533	0.533
Soros <i>et al.</i> [30]	0.658	0.658	0.658
Zamberletti <i>et al.</i> [40]	0.234	0.340	0.278
Yun <i>et al.</i> [38]	0.806	0.714	0.757
Zharkov <i>et al.</i> [41]	0.725	0.952	0.823
Faster R-CNN [28]	0.981	0.996	0.989
RetinaNet [21]	0.988	0.991	0.990
YOLO Nano [15]	0.978	0.997	0.987
YOLO Medium [15]	0.984	0.998	0.991
RT-DETR [23]	0.987	0.999	0.993

Yun *et al.* This detection method was described in 2017 by Yun and Kim. The algorithm is designed for the detection of linear barcodes and supports multiple detections per image. For detecting the salient regions, the entropy scheme is used [5]. The idea is to divide the image into non-overlapping cells, and for each cell the local orientation histogram is computed. The histogram is used to compute the entropy of the cell. Cells with high entropy have high directionality and a high probability of being part of a barcode.

Zharkov *et al.* In 2019, Zharkov et al. proposed a custom Convolutional Neural Network for 1D and 2D barcodes segmentation employing dilated convolution. The network is trained using a loss function that prioritizes high recall over high precision.

Open Source Object Detection Models. In addition, we included five open-source object detection models in our benchmark. Each model was pre-trained on the MS COCO dataset [22] and fine-tuned on our training set. The selected architectures are Faster R-CNN, RetinaNet, YOLO-v8 Medium, and Nano and RT-DETR. The details of the selected architectures are presented in Table 2.

5 Benchmark Results

5.1 Methodology

We assess the detection accuracy using 5-fold cross-validation for both single-class and multi-class modes. End-to-end deep learning models are trained with 75% of the training set, using the remaining 25% as a validation set for early stopping. Zamberletti’s method leverages a pre-trained MLP trained on the Arte-Lab Rotated dataset, that is not included in our dataset, thus preventing any unfair comparison. Timing measurements are taken as the best of three runs to minimize external factor interference.

5.2 Single 1D Barcode Localization

First, we tested the available detection algorithms by considering just images of a single class, linear barcodes, or 2D barcodes. This evaluation focuses on images containing a single linear barcode, allowing us to test all the available algorithms. The total number of images included in this test was 6811. For this test, we resized all images to have their longest side of 640 pixels. This is the same size used to test the methods of Gallo [10] and Zamberletti [40] in their original paper. This is also the default resolution for YOLO-v8 [15] and other object detection networks. At this resolution, our dataset comprises 42 small objects (area $< 32^2$), 2665 medium objects ($32^2 < \text{area} < 96^2$), and 4104 large objects (area $> 96^2$). Traditional methods often rely on some form of texture detection for localization, where barcode texture depends on the number of pixels per element (PPE). After resizing, the PPE ranges from 0.35 to 5.13, with most barcodes in the dataset having a pixel density between 1 and 3 pixels per element. Additionally, there are 1044 barcodes without PPE information, suggesting that the automatic labeler was unable to decode them.

Since not all methods generate a confidence score, we used precision, recall, and F1-score as metrics for a fair comparison. In Table 3 we can see the results of the different methods considering an IoU threshold of 0.5. Gallo and Soros' algorithms produce a single prediction every time, so their precision, recall, and F1 scores are always the same. However, considering a single IoU threshold could not be enough for a fair comparison. A more complete evaluation is displayed in

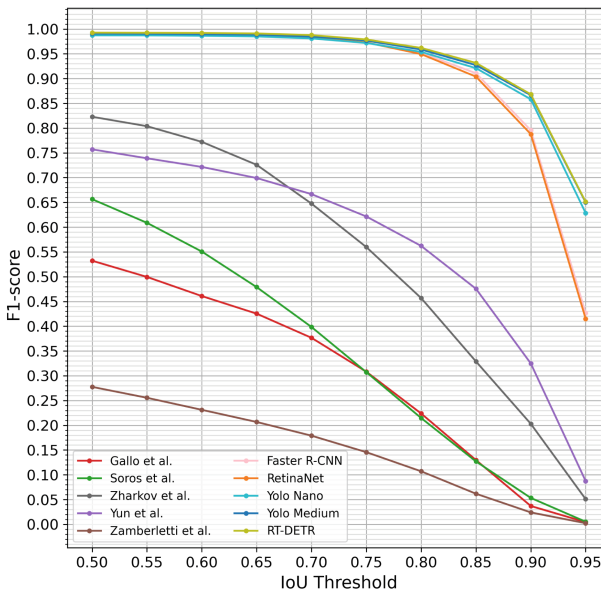


Fig. 2. F1-score of detection algorithms at different thresholds. Employed images contain a single 1D barcode and were resized to have the longest side equal to 640 pixels.

Table 4. Precision, Recall and F1-score with IoU threshold of 0.5. All images contain a single 2D barcode and were resized to have their longest side equal to 640 pixels.

Detection Method	Precision \uparrow	Recall \uparrow	F1-score \uparrow
Soros <i>et al.</i> [30]	0.140	0.140	0.140
Zharkov <i>et al.</i> [41]	0.727	0.900	0.804
Faster R-CNN [28]	0.981	0.992	0.987
RetinaNet [21]	0.981	0.995	0.988
YOLO Nano [15]	0.962	0.989	0.975
YOLO Medium [15]	0.980	0.990	0.985
RT-DETR [23]	0.972	0.997	0.984

Fig. 2, with the F1-score curves at different values of T_{IoU} . Apart from Zharkov *et al.*, all the other end-to-end neural networks always outperform the other methods. This was expected since these methods are more computationally intensive and adept at complex detection problems. Among the tested classic algorithms, Yun *et al.* is by far the one that performs better at every IoU threshold, making it a valid choice when a neural network is too resource-heavy. The methods of Gallo and Soros have similar performance, with a moderate edge in favor of the second one at low T_{IoU} . Zamberletti’s method is the weakest performer overall. Zharkov *et al.* reaches a very high recall, much higher than what is achieved by the classic algorithms, but scores lower in precision. All the other deep-learning-based methods reach a near-perfect precision and recall for $T_{IoU} < 0.75$. Despite being the two biggest models, Faster R-CNN and RetinaNet underperform a bit compared to other networks for $T_{IoU} > 0.75$, meaning that the generated boxes are less precise. Overall, T-DETR leads the leaderboard, albeit by a small margin. Interestingly, YOLO Nano, despite having nearly 10 times fewer parameters, performs similarly to YOLO Medium and RT-DETR, suggesting that smaller networks can excel in this detection task without sacrificing accuracy.

5.3 Single 2D Barcode Localization

In this test, we only include examples with a single two-dimensional barcode. Soros’s method [30] is the only non-deep-learning-based method available that also detects 2D barcodes. The employed dataset contains 1164 images, resized to a maximum edge length of 640 pixels. At this resolution, our dataset included 19 small objects (area $< 32^2$), 202 medium objects ($32^2 < \text{area} < 96^2$), and 943 large objects (area $> 96^2$). Alongside the object’s area, module density remains crucial for determining the dataset’s difficulty. After resizing, the PPE ranges from 0.48 to 9.98, with most codes being uniformly distributed in the range 1.5 to 7.0. Additionally, 90 barcodes lack PPE information. As for the linear barcode case, we present the values of precision, recall, and F1-score of the tested methods considering an IoU threshold of 0.5. The results are presented in Table 4.

It is clear that the Soros *et al.* method, with an F1 score of 0.14, is not a reliable 2D barcode detector. To better understand how the other methods perform at different IoU thresholds, we present their F1 curves in Fig. 3.

Zharkov *et al.* achieves good results, especially in recall, but falls short of the other deep learning architectures. At $T_{IoU} < 0.75$, RetinaNet performs the best in terms of F1-score, while YOLO Medium and RT-DETR have the highest score for $T_{IoU} > 0.75$. YOLO Nano has a similar performance to YOLO Medium, but now the gap is a bit larger with respect to the 1D case.

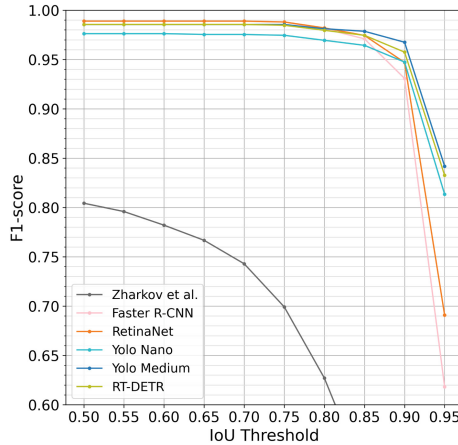


Fig. 3. F1-score curves of 2D barcode detection algorithms at different values of IoU threshold. Employed images contain a single 2D barcode and were resized to have their longest side of 640 pixels.

Table 5. Number of objects per class and size category across the entire dataset, with images resized at different resolutions.

Longest Side Resolution	Type	Small Objects	Medium Objects	Large Objects	Total
640 px	1D	172	3 613	4 277	8 062
	2D	85	611	1 060	1 756
	Total	257	4 224	5 337	9 818
480 px	1D	478	4 789	2 795	8 062
	2D	157	712	887	1 756
	Total	635	5 501	3 682	9 818
320 px	1D	1 813	5 447	802	8 062
	2D	421	574	761	1 756
	Total	2 234	6 021	1 563	9 818

Table 6. Average precision scores for the tested models across all images of the dataset resized at different scales.

Longest Side Resolution	Model	1D barcodes		2D barcodes		Average	
		AP@0.5 \uparrow	AP@[.5:.95] \uparrow	AP@0.5 \uparrow	AP@[.5:.95] \uparrow	mAP@0.5 \uparrow	mAP@[.5:.95] \uparrow
640 px	Zharkov <i>et al.</i>	0.905	0.536	0.741	0.468	0.823	0.502
	YOLO Nano	0.986	0.902	0.960	0.910	0.973	0.906
	YOLO Medium	0.988	0.909	0.976	0.930	0.982	0.920
	RT-DETR	0.989	0.914	0.973	0.930	0.981	0.922
	Faster R-CNN	0.982	0.857	0.967	0.866	0.974	0.862
	RetinaNet	0.973	0.848	0.968	0.894	0.970	0.871
	480 px	Zharkov <i>et al.</i>	0.380	0.180	0.661	0.465	0.521
YOLO Nano		0.982	0.889	0.961	0.901	0.972	0.895
YOLO Medium		0.988	0.899	0.966	0.917	0.977	0.908
RT-DETR		0.987	0.900	0.968	0.919	0.977	0.910
Faster R-CNN		0.979	0.843	0.953	0.843	0.966	0.843
RetinaNet		0.963	0.830	0.948	0.866	0.955	0.848
320 px		Zharkov <i>et al.</i>	0.530	0.254	0.571	0.382	0.551
	YOLO Nano	0.976	0.860	0.947	0.872	0.961	0.866
	YOLO Medium	0.975	0.853	0.946	0.862	0.960	0.857
	RT-DETR	0.980	0.875	0.955	0.893	0.968	0.884
	Faster R-CNN	0.929	0.764	0.928	0.787	0.928	0.775
	RetinaNet	0.887	0.740	0.89	0.793	0.888	0.766

5.4 Multi-class Detection

We expand our analysis to the entirety of the dataset, encompassing both 1D and 2D barcode classes. The task is now not only about detection, but also classification. The available methods for multi-class and multi-ROI detection are the deep-learning-based models. As previously observed, deep-learning models significantly outperform classical methods in this domain. However, implementing them in industrial applications could be challenging due to the high computational costs. A potential solution is to detect barcodes at a lower resolution and execute the decoding phase at full resolution. We thus decided to run our tests at three different resolutions, to test the viability of this strategy. First, all the images are resized to have their longest side equal to 640 pixels, then to 480 pixels and 320 pixels. For each scale, we re-trained the models using a training set with the same scale used for testing. In Table 5 we see the number of instances divided by class and size. In total, 8 748 images are included, with 8 062 instances of 1D barcodes and 1 756 instances of 2D barcodes. To evaluate model performance, we calculated the Average Precision at an IoU threshold of 0.5 (AP@0.5) and the Average Precision across IoU thresholds from 0.5 to 0.95 with a step size of 0.05 (AP@[.5:.95]) for each class. In addition, we considered the corresponding mean Average Precision values (mAP@0.5 and mAP@[.5:.95]) for each model. The results are presented in Table 6. Zharkov *et al.*'s model, while not as robust as the others, achieves a respectable mAP@0.5 score of 0.823 at the 640 pixels scale. However, its performance drops significantly at the other two scales. Other models perform well at all tested resolutions. The performance drop from 640 pixels to 480 pixels is small for most models, while downscaling to 320 pixels has

Table 7. Average time required for detection on PC and on Raspberry PI. All images have been resized to have the longest side to 640 pixels. The ∞ symbol indicates that there was not enough RAM to run the algorithm.

Detection Method	Times on PC (ms)			Times on Raspberry PI (ms)	
	Single-Thread CPU ↓	Multi-Thread CPU ↓	GPU ↓	Single-Thread CPU ↓	Multi-Thread CPU ↓
Gallo <i>et al.</i> [10]	1.63	–	–	53.45	–
Soros <i>et al.</i> [30]	11.25	–	–	397.53	–
Zamberletti <i>et al.</i> [40]	48.20	–	–	1 360.23	–
Yun <i>et al.</i> [38]	7.59	–	–	146.31	–
Zharkov <i>et al.</i> [41]	25.85	5.97	1.45	2 120.43	1 949.08
YOLO Nano [15]	64.99	17.40	18.66	3 034.27	1 803.09
YOLO Medium [15]	478.92	51.36	23.91	20 083.87	15 813.46
RT-DETR [23]	985.41	141.06	37.55	39 882.45	33 224.15
Faster R-CNN [28]	1 271.93	237.91	30.27	∞	∞
RetinaNet [21]	1 124.11	105.20	36.00	∞	∞

a more noticeable impact. At the 640 pixels scale, Faster R-CNN and RetinaNet achieve lower scores than other models, while YOLO Medium and RT-DETR deliver the highest mAP@0.5 and mAP@[.5:.95], respectively. At the other two scales, the scores of Faster R-CNN and RetinaNet decrease more than those of YOLO and RT-DETR. RT-DETR is the best model across all metrics considered, with an increase in lead at the lowest resolution. Surprisingly, YOLO Nano has better metrics across all categories compared to YOLO Medium at 320 pixels resize, while this is not true for the other scales.

5.5 Time Measurement

In this section, we evaluate barcode detection algorithm inference times. This analysis is essential for applications running on devices with limited resources. For a comprehensive assessment, we benchmark the algorithms on two contrasting platforms: a high-end PC and a Raspberry Pi 3B+. The algorithms we tested, implemented in C++, were not specifically optimized for multi-threading, but employ a few OpenCV functions capable of multi-threaded execution. To provide a clear understanding of their performance, we ran these methods on a single CPU thread. For a balanced comparison, we also recorded the inference times of deep-learning methods running on a single CPU thread. In addition, we also report the times of deep-learning methods when running on GPU or CPU with multi-threading enabled. All C++ implementations were compiled with -O3 optimization for maximum performance. For this benchmark, we run all the detection methods on all the images of the dataset. To reduce the impact of the background processes, we repeat detections three times per image and take the lowest time. The final time is the average for every image.

Table 8. Average times required for detection on PC and on Raspberry PI, using a single thread on the CPU, at different longest side resolutions. The ∞ symbol indicates that there was not enough RAM to run the algorithm.

Detection Method	Times on PC (ms)			Times on Raspberry PI (ms)		
	Time at 640px ↓	Time at 480px ↓	Time at 320px ↓	Time at 640px ↓	Time at 480px ↓	Time at 320px ↓
Gallo <i>et al.</i> [10]	1.63	0.92	0.41	53.45	32.04	14.31
Soros <i>et al.</i> [30]	11.25	6.26	2.78	397.53	205.51	92.02
Zamberletti <i>et al.</i> [40]	48.20	29.66	17.42	1 360.23	1 357.17	855.78
Yun <i>et al.</i> [38]	7.59	4.49	2.17	146.31	103.84	52.80
Zharkov <i>et al.</i> [41]	25.85	14.56	6.72	2 120.43	882.50	340.92
YOLO Nano [15]	64.99	40.20	20.82	3 034.27	2 108.00	1 050.38
YOLO Medium [15]	478.92	284.62	135.24	20 083.87	12 091.44	5 570.13
RT-DETR [23]	985.41	604.01	329.26	39 882.45	25 371.39	13 427.26
Faster R-CNN [28]	1 271.93	892.33	599.15	∞	∞	∞
RetinaNet [21]	1 124.11	665.03	319.17	∞	∞	∞

Time on PC. We measured times when running on a PC with a 24-core AMD Ryzen Threadripper Pro 5965WX CPU, 128 GB of DDR4 RAM, and an RTX 4090 GPU. All the tests were conducted after scaling the images to have their longest side of 640 pixels. In total, we have 8 748 images, with a mean resolution of 0.284 Megapixels after resizing. Inference is conducted on a single image at a time. Table 7 presents the times required to run detection methods on a single CPU thread. For deep-learning methods, we also report multi-threaded performance and GPU performance. Focusing on single-threaded performance on the CPU, there’s a significant difference between the methods, with Gallo *et al.* being the fastest (1.63 ms). This was expected since this is the oldest method, and its main focus was to run on limited hardware. Yun *et al.* is the second fastest method (7.59 ms), despite having a better detection accuracy than Soros and Zamberletti’s algorithms. Zharkov *et al.* is the only deep-learning model that could run in real-time on a single core with a recorded time of 25.85 ms. YOLO Nano is also quite faster than the other deep-learning models with a mean execution time of 64.99 ms. YOLO Medium is much slower at 478.9 ms in single-thread. As expected, RT-DETR is slower with a time of 985.4 ms, and both RetinaNet and Faster R-CNN require even more time (1 124 ms and 1 272 ms respectively). Using multiple threads, all neural networks become 5–10 times faster, except YOLO Nano which becomes only 4 times faster with a time of 17.4 ms. On GPU, the ranking remains the same, but bigger models receive a bigger boost than smaller models. The fastest model is still Zharkov *et al.* at 1.45 ms while the slowest one is RetinaNet at 36 ms. All barcode detection methods could be used for real-time applications on a high-end PC. However, it is hard to find a real-world application where this makes economic sense.

We also recorded the single-thread performance when resizing the longest side to 480 pixels and 320 pixels, as deep-learning-based detectors work well even at lower resolutions. The results are shown in Table 8. At lower resolution, the ranking remains the same, but shorter times are required. Indeed, time scales more or less linearly with the amount of pixels.

Time on Embedded Device. Many barcode reading applications rely on embedded CPUs, such as identification marking and retail automatic checkouts. The use of embedded devices instead of PCs ensures a reduction in costs, latency, and space requirements. To measure the performance on embedded devices we run our benchmark on a Raspberry PI 3B+ (1.2 GHz quad-core ARMv8 CPU, 1 GB DDR2 RAM). Since the tested system is now much slower, we had to test on a subset of 500 randomly selected images of the dataset, to make the test run in a reasonable time. The mean area remained 0.284 Megapixels. Single-core CPU tests were conducted for all detection algorithms, with deep-learning methods also tested using all four cores of the CPU. Results are presented in Table 7. Compared to the PC results, execution times increased by 30–50×. Insufficient RAM prevented Faster R-CNN and RetinaNet from running. No method currently achieves real-time performance, with Gallo’s method being close. The comparison between the various methods in terms of timings remains unchanged. Gallo’s method is the fastest (53.45 ms), followed by Yun’s (146.3 ms), Soros’ (397.5 ms), and Zamberletti’s (1 360 ms) algorithms. All the deep-learning methods are slower. Zharkov *et al.* is still the fastest network at 2 120 ms, followed by YOLO Nano (3 034 ms). YOLO Medium and RT-DETR are incredibly slow, with processing times of 20 084 ms and 39 882 ms respectively. Multi-core execution yielded a modest speed-up of roughly 1.5×, potentially limited by unoptimized libraries or system bottlenecks such as RAM. We also recorded the single-thread performance when resizing the longest side to 480 pixels and 320 pixels. The results are shown in Table 8. The ranking remains the same, apart from Zharkov *et al.* surpassing Zamberletti *et al.* at 320 pixels scaling. At this resolution, the time required by the smaller neural networks, Zharkov *et al.* and YOLO Nano, becomes more reasonable (340.9 ms and 1 050 ms respectively), but still far from the real-time applications target.

It is crucial to acknowledge that the speed of these methods could be significantly enhanced through optimization. For instance, the C++ methods we have tested could be optimized with SIMD intrinsics and multi-threaded code, while the use of software toolkits for Edge AI or techniques like quantization and pruning can be employed to boost the speed of neural networks with minimal impact on accuracy. However, this goes beyond the scope of our paper.

6 Conclusion

The paper contributions include a comprehensive review of the field of barcode localization, the release of a large dataset of 8 748 images of barcodes with

standardized annotations, and the public release of our benchmark. This benchmark includes multiple localization algorithms, scripts for training deep learning models, and diverse performance metrics. This ensures transparency and enables researchers to easily replicate and expand upon our work. Finally, we performed multiple tests with our benchmark, using our dataset and trained models, from which we can draw some interesting conclusions. First, our tests confirmed the significant accuracy advantage of deep learning methods over hand-crafted approaches. However, the computational complexity of most deep learning models remains a challenge for real-time embedded applications, since even fairly small models require more than one second per detection. Downscaling the image before localization gives a huge speed-up, but does not solve the problem entirely. Our findings suggest that small neural networks, such as YOLO Nano, perform nearly as well as much bigger architectures like RT-DETR and RetinaNet. Our tests also highlight the big advantage of using pre-trained general models, like YOLO or RetinaNet, over custom-built models like Zharkov's. Lastly, among the methods tailored to barcodes, Yun *et al.* proposal offers an optimal blend of accuracy and speed, surpassing Soros' and Zamberletti's methods in both metrics. The fastest method was Gallo *et al.*, showing that decent accuracy could be achieved even on very constrained devices.

As a closing remark, we hope this benchmark will be a valuable asset for further research in this field. Its modular design facilitates the integration of new algorithms, metrics, and data. We welcome feedback and contributions to further enhance the proposed benchmark.

Acknowledgement. This work was supported by the University of Modena and Reggio Emilia and Fondazione di Modena, through the FAR 2023 and FARD-2023 funds (Fondo di Ateneo per la Ricerca).

References

1. Generate a large labelled dataset of barcodes from open food facts data (2018). <https://github.com/openfoodfacts/openfoodfacts-ai/issues/15>
2. Ando, S.: Image field categorization and edge/corner detection from gradient covariance. *IEEE Trans. Pattern Anal. Mach. Intell.* **22**(2), 179–190 (2000)
3. Bodnár, P., Grósz, T., Tóth, L., Nyúl, L.G.: Efficient visual code localization with neural networks. *Pattern Anal. Appl.* **21**, 249–260 (2018)
4. Chai, D., Hock, F.: Locating and decoding EAN-13 barcodes from images captured by digital cameras. In: 2005 5th International Conference on Information Communications & Signal Processing, pp. 1595–1599 (2005)
5. Chang, S.K., Yang, C.C.: Picture information measures for similarity retrieval. *Comput. Vis. Graph. Image Process.* **23**(3), 366–375 (1983)
6. Chou, T.H., Ho, C.S., Kuo, Y.F.: QR code detection using convolutional neural networks. In: International Conference on Advanced Robotics and Intelligent Systems (ARIS), pp. 1–5 (2015)
7. Do, T., Kim, D.: Quick browser: a unified model to detect and read simple object in real-time. In: 2021 International Joint Conference on Neural Networks (IJCNN), pp. 1–8 (2021)

8. Dubská, M., Herout, A., Havel, J.: Real-time precise detection of regular grids and matrix codes. *J. Real-Time Image Proc.* **11**, 193–200 (2016)
9. Galamhos, C., Matas, J., Kittler, J.: Progressive probabilistic Hough transform for line detection. In: 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 554–560 (1999)
10. Gallo, O., Manduchi, R.: Reading 1D barcodes with mobile phones using deformable templates. *IEEE Trans. Pattern Anal. Mach. Intell.* **33**(9), 1834–1843 (2010)
11. Hansen, D.K., Nasrollahi, K., Rasmussen, C.B., Moeslund, T.B.: Real-time barcode detection and classification using deep learning. In: International Joint Conference on Computational Intelligence, pp. 321–327 (2017)
12. Hu, H., Xu, W., Huang, Q.: A 2D barcode extraction method based on texture direction analysis. In: 2009 Fifth International Conference on Image and Graphics, pp. 759–762 (2009)
13. Jain, A.K., Chen, Y.: Bar code localization using texture analysis. In: Proceedings of 2nd International Conference on Document Analysis and Recognition (ICDAR 1993), pp. 41–44 (1993)
14. Jain, A.K., Karu, K.: Learning texture discrimination masks. *IEEE Trans. Pattern Anal. Mach. Intell.* **18**(2), 195–205 (1996)
15. Jocher, G., Chaurasia, A., Qiu, J.: Ultralytics YOLOv8 (2023)
16. Kamnardsiri, T., Charoenkwan, P., Malang, C., Wudhikarn, R.: 1D barcode detection: novel benchmark datasets and comprehensive comparison of deep convolutional neural network approaches. *Sensors* **22**(22), 8788 (2022)
17. Kapsambelis, C.: Bar codes aren't going away! (2005)
18. Klimek, G., Vamossy, Z.: QR code detection using parallel lines. In: 2013 IEEE 14th International Symposium on Computational Intelligence and Informatics (CINTI), pp. 477–481 (2013)
19. Kubáňová, J., Kubasáková, I., Čulík, K., Štítik, L.: Implementation of barcode technology to logistics processes of a company. *Sustainability* **14**(2), 790 (2022)
20. Li, J., Zhao, Q., Tan, X., Luo, Z., Tang, Z.: Using deep ConvNet for robust 1D barcode detection. In: Advances in Intelligent Systems and Interactive Applications: Proceedings of the 2nd International Conference on Intelligent and Interactive Systems and Applications (IISA 2017), pp. 261–267 (2018)
21. 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)
22. Lin, T.Y., et al.: Microsoft COCO: common objects in context. In: Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, 6–12 September 2014, Proceedings, Part V 13, pp. 740–755 (2014)
23. Lv, W., et al.: DETRs beat YOLOs on real-time object detection. In: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2024)
24. McCathie, L.: The advantages and disadvantages of barcodes and radio frequency identification in supply chain management. Ph.D. thesis, School of Information Technology and Computer Science (2004)
25. Melek, C.G., et al.: Datasets and methods of product recognition on grocery shelf images using computer vision and machine learning approaches: an exhaustive literature review. *Eng. Appl. Artif. Intell.* **133** (2024)
26. Muniz, R., Junco, L., Otero, A.: A robust software barcode reader using the Hough transform. In: Proceedings 1999 International Conference on Information Intelligence and Systems (Cat. No. PR00446), pp. 313–319 (1999)

27. Ottaviani, E., et al.: A common image processing framework for 2D barcode reading. In: 1999 Seventh International Conference on Image Processing and Its Applications (Conf. Publ. No. 465), vol. 2, pp. 652–655 (1999)
28. 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 28 (2015)
29. Soliman, A., et al.: AI-based UAV navigation framework with digital twin technology for mobile target visitation. *Eng. Appl. Artif. Intell.* **123**, 106318 (2023)
30. Sörös, G., Flörkemeier, C.: Blur-resistant joint 1D and 2D barcode localization for smartphones. In: Proceedings of the 12th International Conference on Mobile and Ubiquitous Multimedia, pp. 1–8 (2013)
31. Szentandrás, I., Herout, A., Dubská, M.: Fast detection and recognition of QR codes in high-resolution images. In: Proceedings of the 28th Spring Conference on Computer Graphics, pp. 129–136 (2012)
32. Taveerad, N., Vongpradhip, S.: Development of color QR code for increasing capacity. In: 2015 11th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS), pp. 645–648 (2015)
33. Mate, V.S., Mutreja, S.: Barcode reader market size, share, competitive landscape and trend analysis report by type, by application: global opportunity analysis and industry forecast, 2023–2032 (2023)
34. Viard-Gaudin, C., Normand, N., Barba, D.: A bar code location algorithm using a two-dimensional approach. In: Proceedings of 2nd International Conference on Document Analysis and Recognition (ICDAR 1993), pp. 45–48 (1993)
35. Wachenfeld, S., Terlunen, S., Jiang, X.: Robust recognition of 1-D barcodes using camera phones. In: 2008 19th International Conference on Pattern Recognition, pp. 1–4 (2008)
36. Weng, D., Yang, L.: Design and implementation of barcode management information system. In: Information Engineering and Applications: International Conference on Information Engineering and Applications, pp. 1200–1207 (2012)
37. Wudhikarn, R., Charoenkwan, P., Malang, K.: Deep learning in barcode recognition: a systematic literature review. *IEEE Access* **10**, 8049–8072 (2022)
38. Yun, I., Kim, J.: Vision-based 1D barcode localization method for scale and rotation invariant. In: TENCON - IEEE Region 10 Conference, pp. 2204–2208 (2017)
39. Zamberletti, A., et al.: Neural image restoration for decoding 1-D barcodes using common camera phones. In: VISAPP (1), pp. 5–11 (2010)
40. Zamberletti, A., et al.: Robust angle invariant 1D barcode detection. In: 2013 2nd IAPR Asian Conference on Pattern Recognition, pp. 160–164 (2013)
41. Zharkov, A., Zagaynov, I.: Universal barcode detector via semantic segmentation. In: 2019 International Conference on Document Analysis and Recognition (ICDAR), pp. 837–843 (2019)



PCGAUNet: Pixel Correlation and Gaussian Attention Driven Network for Text Segmentation

Ayush Roy¹, Shivakumara Palaiahnakote²(✉), Umapada Pal¹,
Apostolos Antonacopoulos², and Raghavendra Ramachandra³

¹ Computer Vision and Pattern Recognition, Indian Statistical Institute, Kolkata, India
umapada@isical.ac.in

² School of Science, Engineering and Environment, University of Salford, Manchester, UK
s.palaiahnakote@salford.ac.uk,
a.antonacopoulos@primaresearch.org

³ Norwegian University of Science and Technology, Trondheim, Norway
raghavendra.ramachandra@ntnu.no

Abstract. Text-line segmentation is still considered challenging for complex background scene images. The success of text detection and recognition depends on the success of the text segmentation. This study presents a new method for text segmentation to facilitate reliable detection and recognition. Therefore, we introduce a new model called Pixel Correlation and Gaussian Attention Driven Network (PCGAUNet) for text segmentation. To extract pixel correlation, we modified the MultiResUnet architecture, which leverages pixel-wise correlation to effectively highlight foreground pixels. In addition, the proposed model utilizes the prior spatial statistics of bottleneck features to create a learnable Gaussian distribution, which guides the decoder for accurate text segmentation. Experimental results on three standard scene text segmentation datasets, ICDAR13 FST, Total Text, and COCO-TS, show that the proposed model outperforms existing methods. Furthermore, the results for the underwater dataset UTS-55 show that our model is robust and generic.

Keywords: Text segmentation · Attention mechanism · Pixel correlation · Gaussian distribution · Underwater scene text images

1 Introduction

In practical applications, such as autonomous vehicles navigating in various conditions, challenging factors such as rainy weather and rapid movement of background elements such as trees, other vehicles, and pedestrians, adversely impact the effectiveness of text detection and recognition [1–7]. In addition, text detection and recognition may be necessary in a variety of other extreme situations, such as underwater or aerial images. To address this complex problem, rather than detecting and recognizing text outright, we propose to first segment the text and then select and apply appropriate detection and recognition methods, improving the detection and recognition performance irrespective

of the above-mentioned challenges. This is because segmentation does not require text-specific features, such as those specific to characters, words, and text. Segmentation considers general textual features, such as the correlation between pixels and the distribution of text, for separating text from non-text [8, 9]. This approach can be applied to any type of text and background combination. In the case of text detection and recognition, models focus on extracting the characteristics of characters, words, and lines to detect text accurately. This approach may not work well in complex scenes; hence, generalization is questionable for those models. Thus, to ensure that the detection and recognition models work for any image without any constraints and assumptions, text segmentation before detection and recognition becomes very important.

Recent methodologies have sought to address these challenges by incorporating prior knowledge of text to guide models in producing segmentation results that closely align with text characteristics [8, 9]. Despite considerable progress made in text segmentation methodologies, numerous existing strategies are plagued by the following shortcomings. The effectiveness of segmentation relies solely on visual indicators from foreground elements. However, these cues are highly susceptible to interference from the background noise. It is evident from the illustration shown in Fig. 1, where it is noted that the existing transformer-based model [10], which extracts inter- and intra-textual features at various granularities by modeling global and local dependencies for text segmentation, fails to output proper segmentation results compared to the Ground Truth (GT), especially for underwater and low-contrast images. The key reason for these poor results is that the model depends significantly on the augmented data and parameter settings. On the other hand, the proposed model exhibited superior performance.



Fig. 1. Examples of scene and underwater images where TextFormer [10] fails to produce satisfactory segmented text mask whereas the proposed model performs well.

It is confirmed from the illustration in Fig. 1 that developing a generalized method that can work for all possible real-world scenarios is challenging. Therefore, we introduce PCGAUNet, a novel approach that leverages pixel-level correlation to accentuate the foreground regions. The text then employs an attention mechanism that utilizes a learnable Gaussian distribution derived from the spatial information of the bottleneck features. It is important to note that the pixel correlation and Gaussian distribution are global features that are capable of representing any type of text. Moreover, these features are unique in their ability to represent text as opposed to non-text regions. The proposed model successfully segments text by utilizing pixel-wise correlation to emphasize foreground pixels and employing distribution-aided conditioned decoding to diminish irrelevant noisy features.

The key contributions of the proposed study are as follows. (i) Modifying MultiResUnet to extract pixel correlation features. (ii) Use of Gaussian distribution derived from spatial statistics through an attention mechanism to accurately segment the text in scene images. The remainder of this paper is organized as follows. The related semantic and text segmentation methods are discussed in Sect. 2. The modified MultiResUnet architecture, called PCGAUNet, and attention modules are presented in Sect. 3. Section 4 discusses the results of our own and the standard datasets to validate the proposed and existing methods. Section 5 summarizes the findings of this work.

2 Related Works

Because of the methods of semantic segmentation text segmentation and relevant approaches, we review the different methods in the same categories.

Semantic Segmentation: For instance, the method in [11] employs dilated convolutions to expand the receptive field, whereas [12] integrates boundary information to prevent individual pixels from being overshadowed by the global scene. In addition, attention mechanisms and encoder-decoder structures have been effectively utilized in semantic segmentation. The method in [13] combines the transformer architecture with a lightweight multilayer perception decoder to enhance the performance. However, despite these advancements, integrating text characteristics into semantic segmentation remains a daunting task. In summary, the primary objective of the methods is not text segmentation; therefore, these methods are not suitable for text segmentation in scenes and underwater images.

Text Segmentation: TexRNet [9] evaluated the readability of the segmentation results using a single-word text recognizer as a discriminator. The approach in [14] integrated text segmentation with text region detection for co-optimization, while ARM-Net [15] proposed a module that highlighted text regions and provided higher-level semantic information about the text. PGTSNet [16] introduced pluggable text detection and text-line recognition modules to enhance text perception. Additionally, [17] pioneers text instance segmentation based on attention mechanisms. PSPNet [18], which was originally designed for semantic segmentation, is a deep convolutional neural network that is also used for text segmentation [19]. This network is built on the ResNet model for image classification and uses a set of dilated convolutions to replace the standard convolutions

in the ResNet part of the network to enlarge the receptive field of the neural network. To deal with the specificity of scene text segmentation, [20] proposed a Segmentation Multi-scale Attention Network composed of three main components: a ResNet encoder, a multiscale attention module, and a convolutional decoder. The encoder is based on PSPNet [18] for semantic segmentation, which is a deep fully convolutional neural network that repurposes ResNet, originally designed for image classification. [21] presents an object-contextual representation approach for semantic segmentation in which the label of a pixel is the label of the object in which the pixel lies. Pixel representation is strengthened by characterizing each pixel using the corresponding object region representation.

Overall, despite the existence of models for text segmentation, the problem of text segmentation remains an open challenge. This is because the scope of most methods is limited to scene images, not poor-quality realistic images such as underwater images. In other words, the existing methods are ineffective for multiple domains.

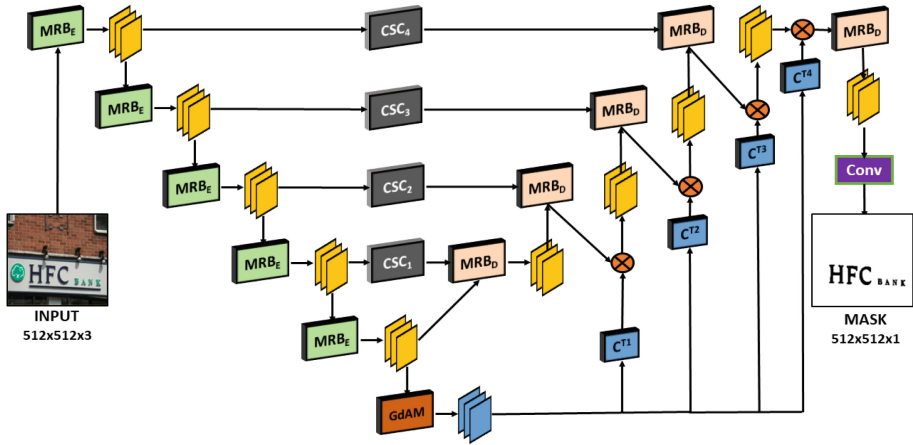


Fig. 2. The block diagram representation of PCGAUNet

3 Methodology

The proposed model, PCGAUNet, uses MultiResUNet [22] as its backbone for text segmentation. The effectiveness of MultiResUNet has been proven in various domains like biomedical imaging [23], remote sensing [24], biometrics [25], semantic scene segmentation [26], etc. Within PCGAUNet, the MRB_E and MRB_D serve as MultiResUNet Blocks, operating on the encoder and decoder sides, respectively. Introducing a Correlation-aided Skip Connection (CSC) module enables the capture of pixel-wise correlations between foreground and background pixels, creating a correlation space. As information traverses from the encoder to the decoder via skip connections, it undergoes a projection onto this correlation space before reaching the decoder module. Moreover, the Gaussian Distribution Attention Module (GdAM) leverages spatial means and standard deviations across all channels of the bottleneck layer to generate a 2D Gaussian

distribution. Features extracted from this distribution inform the trans-posed convolution layers (C^T), generating attention maps. These maps, in turn, undergo element-wise multiplication with decoder feature maps, effectively highlighting the relevant spatial regions. An illustration of PCGAUNet is shown in Fig. 2.

3.1 Correlation-Aided Skip Connection

In text segmentation, discerning foreground (text) and background pixels is a critical task. However, in natural scenes, the relationship between adjacent pixels, especially between the text and background regions, can be intricate and subtle. The CSC module addresses this challenge by capturing the inherent correlation patterns between foreground and background pixels, thus enhancing the model’s ability to focus on relevant regions. Consider an image containing text overlaid on various background elements. Pixels corresponding to the text exhibit distinctive correlation patterns compared to those representing the background. By analyzing these correlation patterns, the model can effectively distinguish between text and non-text regions, leading to precise segmentation outcomes. The detailed architecture for extracting pixel correlation is shown in Fig. 3.

The CSC module employs a multi-step approach to exploit pixel correlations. It begins by computing the mean feature map F' across channels of the input feature F as shown in Eq. (1). This operation effectively summarizes the spatial distribution of features within the input image.

$$F' = 1/C \sum_{c=1}^C F_c \quad (1)$$

By performing element-wise multiplication between the F' and its transpose, the module constructs a correlation space S_{corr} as shown in Eq. (2). This space encapsulates the interdependencies between adjacent pixels, highlighting regions where pixel correlations are particularly strong.

$$S_{corr} = F' \odot F'^T \quad (2)$$

Through the application of a convolutional layer, the module derives attention weights that accentuate regions exhibiting high pixel correlations while suppressing noise and irrelevant details. The obtained attention weights are then applied to modulate the original feature map. This process amplifies features in regions where pixel correlations are significant, effectively directing the model’s focus toward salient text regions (foreground pixels) while attenuating background noise. This is demonstrated in Eq. (3) where $f_{sigmoid}^{1 \times 1}$ represents a separable convolution layer with a kernel size of 1 and a sigmoid activation function.

$$F_{skip} = F \odot f_{sigmoid}^{1 \times 1}(S_{corr}) \quad (3)$$

This attention-aided feature, F_{skip} , was then passed on to the MRB_E for further processing. By integrating correlation-based attention mechanisms, the CSC module facilitates a nuanced understanding of image context, enabling the model to discern

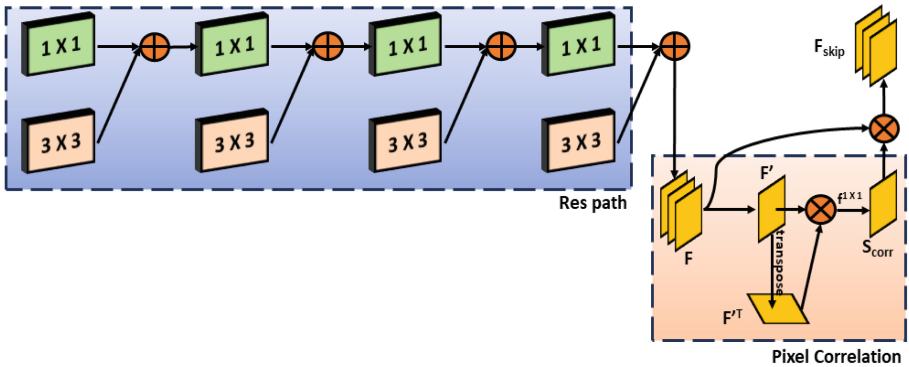


Fig. 3. Correlation-aided Skip Connection

text regions from complex backgrounds. This correlation-driven approach enables the model to make informed segmentation decisions, leading to more accurate and robust text segmentation outcomes. This reinforcement of the attention-aided skip features from the encoder side helps the decoder generate better correlation-oriented feature maps. A block diagram of the CSC module is shown in Fig. 3. The foreground pixel correlations of the images shown in Fig. 1 can be observed in the heat maps, as shown in Fig. 4.

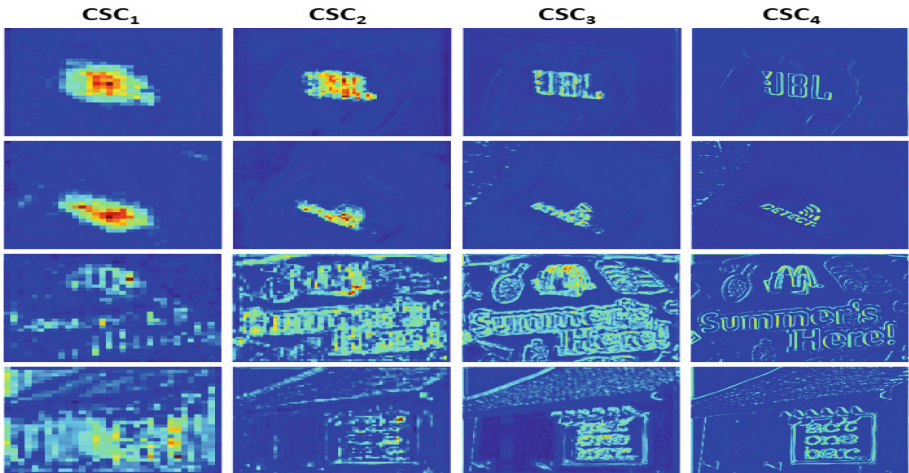


Fig. 4. Heatmaps showing the correlated regions highlighted by the CSC module for images shown in Fig. 1.

3.2 Gaussian Distribution Attention Module

The Gaussian distribution Attention Module (GdAM) plays a pivotal role in enhancing text segmentation by leveraging the bottleneck features to derive a 2D Gaussian distribution and extract an attention mask. Bottleneck feature (B) encapsulates high-level

representations extracted from the input image, providing a comprehensive understanding of its salient features. GdAM utilizes spatial statistical information across all channels of the bottleneck feature to generate a two-dimensional (2D) Gaussian distribution. The mean (μ) and standard deviation (σ) for the distribution, $\mathcal{N}(\mu, \sigma)$, are calculated using Eq. (4) and Eq. (5), respectively. The encapsulation of the statistical properties of the bottleneck feature space enables the intricate encoded information of the input image to be represented within the distribution.

$$\mu = f^{1 \times 1}(B) \tag{4}$$

$$\sigma = f_{softplus}^{1 \times 1}(B) \tag{5}$$

The GdAM employs transposed convolution layers to upsample B , enhancing spatial resolution while preserving essential features. The upsampled features are B_1, B_2, \dots, B_n ($n = 4$ which can be observed in Fig. 2). Subsequently, the distribution function is applied to each upsampled feature map, generating corresponding 2D Gaussian distributions $\mathcal{N}(\mu_n, \sigma_n)$ where μ_n and σ_n denote the spatial mean and standard deviation across the channels of B_n . The attention masks derived from the Gaussian distributions serve as guidance signals, highlighting relevant spatial regions within the decoder features. By modulating the decoder features (D_n) with attention masks, the model focuses its segmentation efforts on regions deemed significant based on the learned spatial coherence and variability from the bottleneck features. This is shown in Eq. (6).

$$D_{GdAM} = D_n \odot f_{sigmoid}^{1 \times 1}(\mathcal{N}(\mu_n, \sigma_n)) \tag{6}$$

By integrating bottleneck features to derive 2D Gaussian distributions and extract attention masks, the model gains a deeper understanding of spatial relationships within

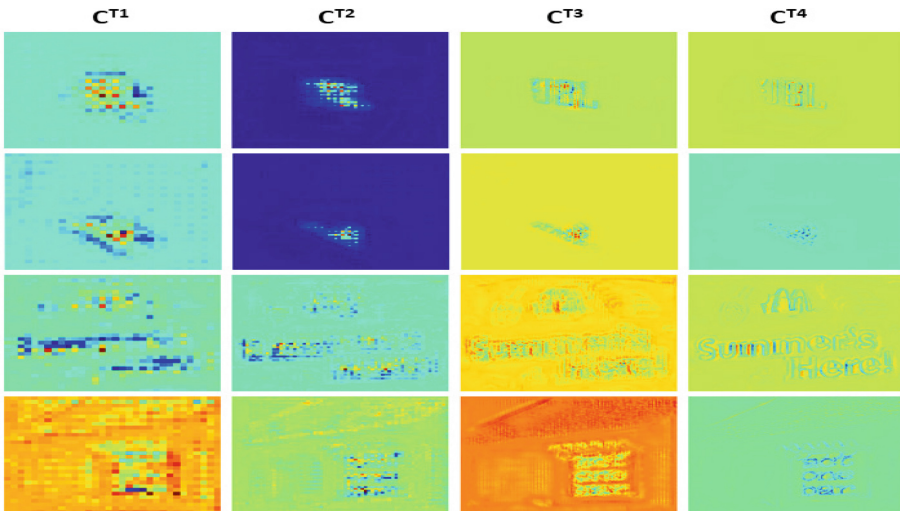


Fig. 5. Heatmaps showing the focus of GdAM for images shown in Fig. 1.

the input image, particularly in text-rich regions. By leveraging learned spatial characteristics, the model can discern text regions amidst cluttered backgrounds, facilitating more precise and reliable segmentation outcomes. Figure 5 demonstrates the heatmaps of D_{GdAM} corresponding to $C^{T1}-C^{T2}$.

After the incorporation of both the GdAM and CSC modules, the enhancement in the feature extraction of the proposed model can be qualitatively assessed by visualizing the heatmaps of the encoder and decoder layers. Heatmaps corresponding to the encoder last layer (stated as Encoder last in Fig. 6). The bottleneck layer (stated as Bottleneck in Fig. 6), and the decoder last layer (stated as the decoder last in Fig. 6) can be seen to focus on the relevant text regions. As we traverse from the encoder to the decoder layers, that is, delving deeper into the model’s architecture, notable enhancements and clarity in relevant information extraction can be observed concurrently with the suppression of unnecessary details. This progression underscores the effectiveness of the proposed architecture, demonstrating the gradual refinement of information across the depths of the model.

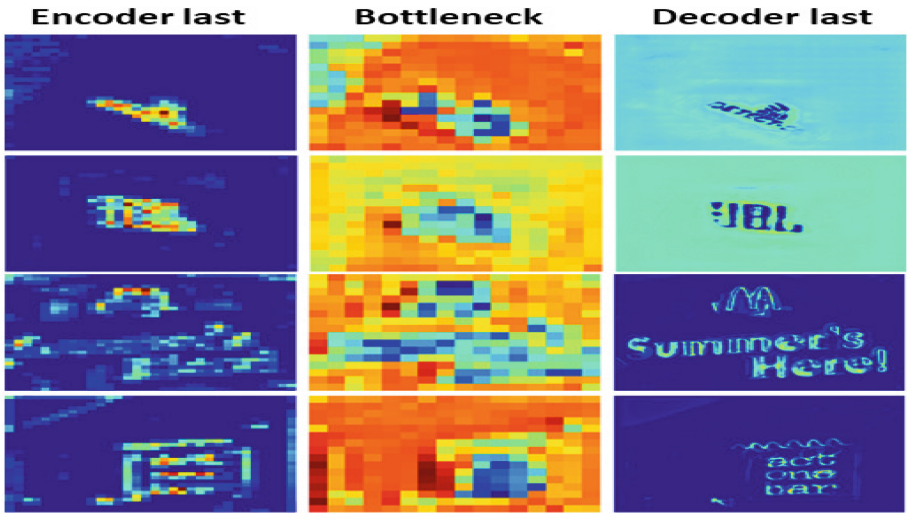


Fig. 6. Heatmaps of the encoder and decoder layers after using the GdAM and CSC module for images shown in Fig. 1.

4 Experimental Results

To evaluate the proposed model in terms of usefulness, fairness, and effectiveness, we consider four datasets, namely, the two standard datasets for scene text segmentation, ICDAR13 FST [27], and Total-Text [28]. To demonstrate fairness, the underwater text segmentation dataset has been used, UTS-55, which validates the effectiveness of the proposed method in challenging situations. Furthermore, to validate the generalization ability, the proposed method is tested on an additional dataset, COCO-TS [19], which

consists of scene, printed, and handwritten text images. Therefore, the generalization properties of the proposed method were validated by testing on datasets from above mentioned different domains of the data.

4.1 Dataset and Evaluation

The details of the ICDAR13 FST [27], Total-Text [28], UTS-55, and COCO-TS [19] datasets are listed in Table 1. Sample images for the respective datasets are shown in Fig. 7, where each sample indicates different characteristics, complexity, and nature. Hence, these four datasets represent four different domains. Therefore, these datasets are used to evaluate the usefulness, fairness, effectiveness, and the generic nature of the proposed method. We used standard metrics, which are defined as the Foreground Intersection-Over-Union (fgIoU) and F-score on foreground pixels to measure the performance of the methods.

Table 1. A summary of the number of images used for training and testing for the three datasets and the additional COCO-TS dataset.

Dataset	Training	Testing
Scene Text-ICDAR13 FST [27]	229	233
Scene Text-Total-Text [28]	1254	300
Underwater Text-UTS-55	35	20
Scene + Printed + Handwritten-COCO-TS [19]	43686	10000

Implementation Details: We use $256 \times 256 \times 3$ dimensioned images as our input. We train the models using the Adam optimizer for 100 iterations, a learning rate of 0.00001, and a batch size of 4. We have used a linear combination of dice loss and BCE loss for training the model. During training, we used a system with an Intel Core i7 processor, with 8 GB RAM and an NVIDIA P100 GPU. Python version 3.7.4 is used for implementation.

4.2 Ablation Study

The main components of the proposed method to segment text in different domains are the modified architecture of MultiResUnet as the baseline UNet for feature extraction, Correlation-Aided Skip Connection (CSC) for defining pixel correlation, and Gaussian distribution attention module (GdAM) for selecting features that represent text for text segmentation. To validate the effectiveness of each key step, we conducted the following experiments on the ICDAR13 FST [27] dataset: (i) This experiment used the baseline architecture of MultiResUnet [22] to show that the baseline is not effective in solving complex segmentation. (ii) This experiment included the baseline architecture with



Fig. 7. Sample images of three datasets, Total Text, ICDAR13 FST, and UTS-55 and the additional COCO-TS dataset.

GdAM to demonstrate the contribution of GdAM to text segmentation. (iii) This experiment included the baseline + GdAM and CSC to validate the effectiveness of CSC for text-line segmentation. (iv) This experiment includes the modified MultiResUnet + GdAM + CSC, which is the proposed method. It is clear from Table 2 that the baseline architecture is not sufficient to achieve the best result compared with the proposed method. In the same way, the results of experiments (ii) and (iii) show that GdAM and CSC contribute equally to achieving the best text segmentation. This is evident from the results of the proposed method, which combines all the steps and achieves the best result compared to the baseline and individual key steps.

Table 2. Ablation study to analyze the effect of GdAM and CSC module on the performance of the model (in %)

#	Configuration	fgIoU	F-score
(i)	MultiResUnet	63.31	75.58
(ii)	MultiResUnet + GdAM	74.34	86.60
(iii)	MultiResUnet + GdAM + CSC	75.13	87.49
(iv)	Modified MultiResUnet + GdAM + CSC (PCGAUNet)	75.29	87.57

4.3 Comparison with the State-of-the-Art

A qualitative analysis of the PCGAUnet output is in Fig. 8 where the segmented output mask is shown along with the original image and the ground truth. As shown in Fig. 8, our proposed method correctly segmented text for all four datasets. This demonstrates that the proposed method can address the challenges of text in multiple domains.

Quantitative results of the proposed and existing methods on the datasets of different domains are presented in Table 3, where it is observed that for all four datasets, the proposed model is superior to the existing models. In addition, it demonstrates superior accuracy in segmenting unclear and densely packed text regions, leveraging its capability to model intricate pixel-level relationships between foreground and background elements. Furthermore, our approach incorporates a learnable Gaussian distribution based on spatial statistics derived from the richly encoded features in the bottleneck layer. This adaptation enables the model to effectively focus on text regions within complex under-water images, addressing challenges such as the poor quality and noise prevalent in such environments. Therefore, we can infer that the proposed model is domain-independent. The reason for the poor results of the existing methods is that they were limited to a particular type and hence lack generalization ability.

Table 3. Comparison of PCGAUnet with the state-of-the-art segmentation models (in %)

Model	ICDAR13 FST [27]		Total Text [28]		UTS-55		COCO-TS	
	fgIoU	F-score	fgIoU	F-score	fgIoU	F-score	fgIoU	F-score
PSPNet [18, 19]	–	79.71	–	74.00	–	68.00	–	74.0
SMANet [20]	–	78.52	–	77.00	–	71.00	–	77.0
DeepLab V3+ [11]	69.27	80.20	74.44	82.42	70.82	78.26	72.07	64.1
HRNetV2-W48+OCR [21]	72.45	83.00	76.23	83.28	76.01	79.50	69.54	62.7
TexRNet [9]	73.38	85.00	78.47	84.80	76.73	81.74	72.39	72.0
TextFormer [10]	72.27	83.80	81.56	88.70	77.15	83.38	73.20	74.5
Ours	75.29	87.57	86.54	90.23	81.51	88.23	73.42	74.5

To further demonstrate the robustness of the proposed model, cross-dataset experimentation is shown in Table 4. Three setups were used, FST-TT (trained on ICDAR13 FST [27] and tested on Total Text [28]) and FST-UTS (trained on ICDAR13 FST [27] and tested on UTS-55), TT-FST (trained on Total Text [28] and tested on ICDAR13 FST [27]) and TT-UTS (trained on Total Text [28] and tested on UTS-55), and UTS-FST (trained on UTS-55 and tested on ICDAR13 FST [27]) and UTS-TT (trained on UTS-55 and tested on Total Text [28]). It can be seen that the proposed model performs better than other models. This demonstrates the robustness of the model and the effectiveness of the learnable sampled attention weights from the distribution in GdAM along with

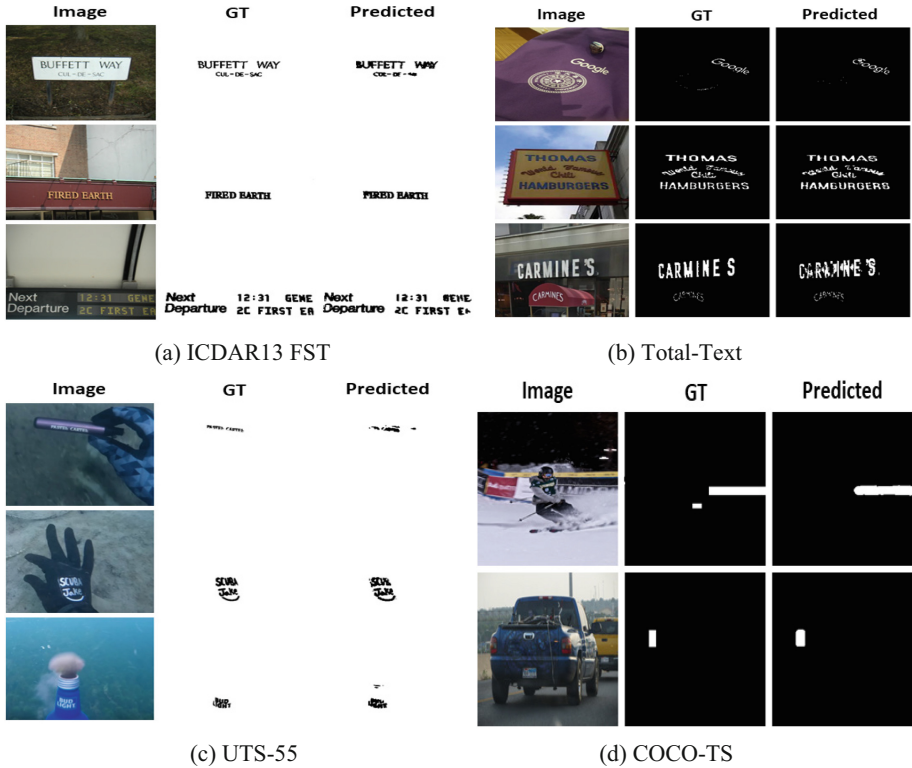


Fig. 8. Predicted mask of PCGAUNet for the images shown in Fig. 7.

the spatial focus of the CSC module. It is important to note that in conducting the experiments presented in Tables 4 and 5, we utilized only three datasets and did not incorporate the COCO-TS dataset. This is because we separately benchmark the proposed model on a larger, more diverse text segmentation dataset to demonstrate its performance under conditions of both data scarcity and data availability.

Table 4. Cross-dataset validation of the proposed and existing methods with one to one set up evaluation (in %)

Model	FST-TT		FST-UTS		TT-FST		TT-UTS		UTS-FST		UTS-TT	
	fgIoU	F-score	fgIoU	F-score	fgIoU	F-score	fgIoU	F-score	fgIoU	F-score	fgIoU	F-score
TexRNet [9]	68.46	71.50	63.52	59.85	68.47	70.75	60.69	62.98	60.73	57.74	60.57	63.19
TextFormer [10]	70.43	76.30	64.47	63.80	72.18	70.35	61.56	60.10	63.27	59.92	61.15	63.55
Ours	77.67	74.69	68.59	67.23	75.02	72.51	63.70	65.35	65.69	61.78	66.96	64.11

To further test the robustness of the models, we used a two-one setup where the three datasets mentioned above are used for evaluation. The symbolic representation is XX-YY-ZZ where XX dataset is used for training, YY dataset is used for validation, and ZZ dataset is used for testing. Table 5 demonstrates the results of this setup where the proposed model showcases better results in most of the cases than the state-of-the-art.

Table 5. Cross-dataset validation of the proposed and existing methods with two-one set-up evaluations (in %)

Model	FST-TT-UTS		FST-UTS-TT		TT-FST-UTS		TT-UTS-FST		UTS-FST-TT		UTS-TT-FST	
	fgIoU	F-score	fgIoU	F-score	fgIoU	F-score	fgIoU	F-score	fgIoU	F-score	fgIoU	F-score
TexRNet [9]	65.41	67.22	59.86	57.43	64.51	66.25	58.29	57.90	58.55	52.34	55.91	57.25
TextFormer [10]	65.43	66.50	60.13	59.11	66.35	67.81	58.15	57.46	62.41	51.72	55.85	57.55
Proposed	67.31	64.89	62.39	60.77	65.13	67.91	59.03	58.68	62.35	54.82	54.89	59.85

To test the efficiency of the proposed method, we estimated the number of parameters and GFLOP involved in the text segmentation process. Table 6 presents a computational comparison between the proposed model and state-of-the-art methods. The number of trainable parameters in a deep learning model refers to the total number of parameters (weights and biases) that can be adjusted during the training process. These parameters are updated through the backpropagation algorithm based on the loss function to minimize the error in predictions. GFLOPs are a measure of the computational complexity of a deep learning model. It represents the number of billion (giga) floating-point operations the model performs during inference (forward pass). This metric helps to understand the computational resources needed to run the model. The proposed model outperforms the state-of-the-art methods in both metrics, demonstrating superior computational efficiency.

Table 6. Computational comparison of the proposed and existing methods

Models	No. of Parameters	GFLOPs
PSPNet [18, 19]	46.6 M	357.17
DeepLab V3+ [11]	59.5 M	178.72
HRNetV2-W48 + OCR [21]	65.8 M	174.043
TexRNet [9]	67.2 M	–
Proposed	28.9 M	113.95

4.4 Error Cases

Although PCGAUNet exhibits superior performance compared to state-of-the-art models, there are still areas for improvement. Figure 9 illustrates the specific images that result in an erroneous segmentation output. These images present challenges such as light reflection, hazy foreground, and intricate background-to-foreground relationships, making it challenging for the model to accurately focus on text regions and generate precisely segmented masks. The distortions and missed pixels evident in the segmented outputs underscore the extreme difficulties posed by scenes and underwater images. Addressing these challenges necessitates advancements in handling complex lighting conditions, mitigating hazy foregrounds, and improving the ability of the model to discern text regions amidst intricate backgrounds. However, this is beyond the scope of the proposed study. However, instead of the spatial domain, if we consider the frequency and polar domains and explore language models to integrate the features extracted from multiple domains, the above challenges can be addressed, which will be our future work.

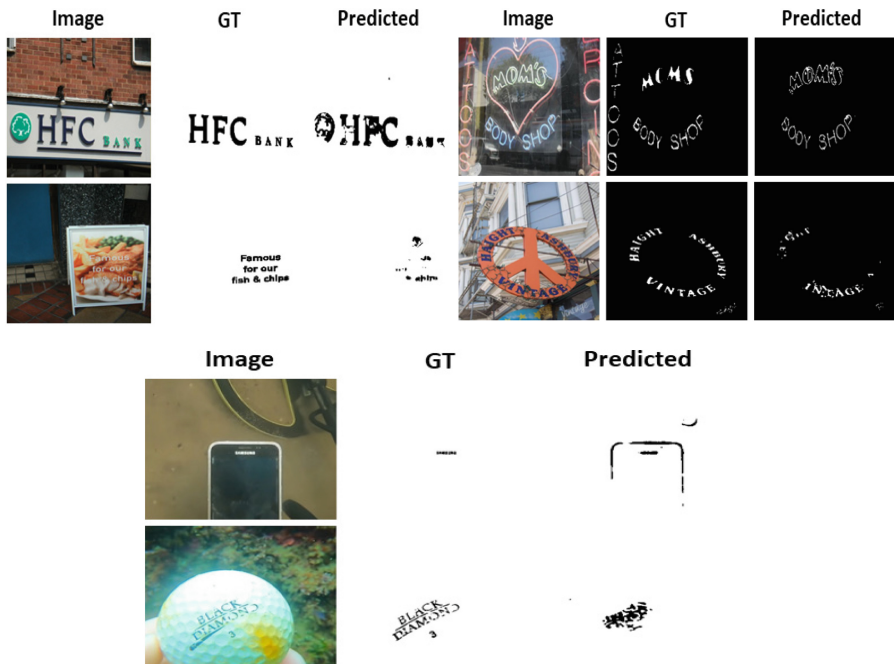


Fig. 9. Error cases of PCGAUNet.

5 Conclusions and Future Works

In this paper, we introduced a novel model called PCGAUNet for text segmentation by leveraging a modified MultiResUnet as the backbone architecture, correlation-aided skip connections (CSC), and a Gaussian Distribution Attention Module (GdAM). The

CSC module harnesses pixel-wise correlations to provide essential attention to relevant spatial regions, whereas the GdAM module uses spatial statistics derived from highly enriched bottleneck features. These modules collectively contributed to enhancing the performance of the model. The experiments on four different datasets, namely, ICDAR13 FST, which is a standard dataset for scene text segmentation, Total-Text, which is a standard dataset for scene text detection and segmentation, UTS-55, which is an underwater scene text dataset and COCO-TS, which is a general large scale dataset including scene, printed and handwritten text images, show that the proposed method is domain-independent and generalizable. In addition, the lower number of GFLOPs and trainable parameters required show that the proposed method is more efficient for text segmentation than existing methods. Overall, the performance of the proposed method is superior to that of existing methods for all four datasets in terms of robustness, generalization, and efficiency. However, for certain images that suffer from severe degradation, blur, and poor contrast, the proposed method does not perform as well, as discussed in the Experimental section. To solve this problem, we plan to explore a language model that integrates features extracted from multiple domains.

References

1. Dzida, M., Vukadin, D., Silic, M., Delac, G., Vladimir, K.: An overview of state-of-the-art solutions for scene text detection. In: Proceedings of MIPRO, pp. 947–952 (2023)
2. Dang, Q.V., Lee, G.S.: Scene text segmentation by paired data synthesis. In: Proceedings of ICIP, pp. 545–549 (2023)
3. Pal, S., Roy, A., Shivakumara, P., Pal, U.: Adapting a swin transformer for license plate number and text detection in drone images. *Artif. Intell. Appl.* **1**(3), 145–154 (2023)
4. Roy, A., Shivakumara, P., Pal, U., Mokayed, H., Liwicki, M.: Fourier feature-based CBAM and vision transformer for text detection in drone images. In: Proceedings of ICDAR, pp. 257–271 (2023)
5. Pal, S., Roy, A., Shivakumara, P., Pal, U.: A robust SLIC based approach for segmentation using canny edge detector. *Artif. Intell. Appl.* (2022)
6. Jain, J., Li, J., Chiu, M.T., Hassani, A., Orlov, N., Shi, H.: OneFormer: one transformer to rule universal image segmentation. In: Proceedings of CVPR, pp. 2989–2998 (2023)
7. Alkhaled, L., Roy, A., Palaiahnakote, S.: An attention-based fusion of ResNet50 and InceptionV3 model for water meter digit recognition. *Artif. Intell. Appl.* (2023). <https://doi.org/10.47852/bonviewAIA32021197>
8. Zhang, Y., et al.: Inversion-based style transfer with diffusion models. In: Proceedings of CVPR, pp. 10146–10156 (2023)
9. Xu, X., Zhang, Z., Wang, Z., Price, B., Wang, Z., Shi, H.: Rethinking text segmentation: a novel dataset and a text-specific refinement approach. In: Proceedings of CVPR, pp. 12045–12055 (2021)
10. Wang, X., Wu, C., Yu, H., Li, B., Xue, X.: TextFormer: component-aware text segmentation with transformer. In: Proceedings of ICME, pp. 1877–1882 (2023)
11. Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoderdecoder with atrous separable convolution for semantic image segmentation. In: Proceedings of ECCV, pp. 801–818 (2018)
12. Ding, H., Jiang, X., Liu, A.Q., Thalmann, N.M., Wang, G.: Boundary-aware feature propagation for scene segmentation. In: Proceedings of CVPR, pp. 6819–6829 (2019)

13. Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J.M., Luo, P.: SegFormer: simple and efficient design for semantic segmentation with transformers. In: *Advances in Neural Information Processing Systems*, vol. 34, pp. 12077–12090 (2021)
14. Wang, C., et al.: Semisupervised pixel-level scene text segmentation by mutually guided network. *IEEE Trans. Image Process.* **30**, 8212–8221 (2021)
15. Ren, Y., Zhang, J., Chen, B., Zhang, X., Jin, L.: Looking from a higher-level perspective: attention and recognition enhanced multi-scale scene text segmentation. In: *Proceedings of ACCV*, pp. 3138–3154 (2022)
16. Xu, X., Qi, Z., Ma, J., Zhang, H., Shan, Y., Qie, X.: BTS: a bi-lingual benchmark for text segmentation in the wild. In: *Proceedings of CVPR*, pp. 19152–19162 (2022)
17. Zu, X., Yu, H., Li, B., Que, X.: Weakly-supervised text instance segmentation. *arXiv preprint arXiv:2303.10848* (2023)
18. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: *Proceedings of CVPR*, pp. 2881–2890 (2017)
19. Bonechi, S., Andreini, P., Bianchini, M., Scarselli, F.: COCO_TS dataset: pixel-level annotations based on weak supervision for scene text segmentation. In: *Proceedings of ICANN 2019: Image Processing*, pp. 238–250 (2019)
20. Bonechi, S., Bianchini, M., Scarselli, F., Andreini, P.: Weak supervision for generating pixel-level annotations in scene text segmentation. *Pattern Recogn. Lett.* **138**, 1–7 (2020)
21. Yuan, Y., Chen, X., Wang, J.: Object-contextual representations for semantic segmentation. In: *Proceedings of ECCV*, pp. 173–190 (2020)
22. Ibtehaz, N., Rahman, M.S.: MultiResUNet: rethinking the U-Net architecture for multimodal biomedical image segmentation. *Neural Netw.* **121**, 74–87 (2020)
23. Yang, J., Zhu, J., Wang, H., Yang, X.: Dilated MultiResUNet: Dilated multiresidual blocks network based on U-Net for biomedical image segmentation. *Biomed. Signal Process. Control* **68**, 102643 (2021)
24. Li, R., et al.: Multiattention network for semantic segmentation of fine-resolution remote sensing images. *IEEE Trans. Geosci. Remote Sens.* **60**, 1–13 (2021)
25. Roy, A., Mohiuddin, S., Sarkar, R.: A similarity-based positional attention aided deep learning model for copy-move forgery detection. *IEEE Trans. Artif. Intell.* (2024)
26. Abdollahi, A., Pradhan, B.: Integrating semantic edges and segmentation information for building extraction from aerial images using UNet. *Mach. Learn. Appl.* **6**, 100194 (2021)
27. Karatzas, D., et al.: ICDAR 2013 robust reading competition. In: *Proceedings of ICDAR*, pp. 1484–1493 (2013)
28. Ch'ng, C.K., Chan, C.S.: Total-text: a comprehensive dataset for scene text detection and recognition. In: *Proceedings of ICDAR*, pp. 935–942 (2017)



DATR: Domain Agnostic Text Recognizer

Kunal Purkayastha^{1,3}, Shashwat Sarkar¹, Shivakumara Palaiahnakote^{2(✉)},
Umapada Pal¹, and Palash Ghosal³

- ¹ Computer Vision and Pattern Recognition Unit, Indian Statistical Institute, Kolkata, India
umapada@isical.ac.in
- ² School of Science, Engineering and Environment, University of Salford, Manchester, UK
s.palaiahnakote@salford.ac.uk
- ³ Department of Information Technology, Sikkim Manipal Institute of Technology, Sikkim
Manipal University, Gangtok, Sikkim, India

Abstract. Recognizing text extracted from multiple domains is complex and challenging because complexities vary from one domain to another. Most existing methods focus either on natural scene text or specific text type but not text of multiple domains, namely, scene, underwater, and drone texts. In addition, the state-of-the-art models ignore the vital cues that exist in multiple instances of the text. This paper presents a new method called the Student-Teacher-Assistant (STA) network, which involves dual CLIP models to exploit cues in multiple text instances. The model that uses ResNet50 in its image encoder is called helper CLIP, while the model that uses ViT in its image encoder is called primary CLIP. The proposed work processes both models simultaneously to extract visual and textual features through image and text encoders. Our work uses cosine similarity for the randomly chosen input image to detect instances similar to the input image. The input and similar instances are supplied to primary and helper CLIPs for visual and textual feature extraction. The outputs of dual CLIPs are fused in a different way through the alignment step for recognizing text accurately, irrespective of domains. To demonstrate the proposed model's significance, experiments are conducted on a set of standard natural scene text datasets (regular and irregular), underwater images, and drone images. The results on three different domains show that the proposed model outperforms the state-of-the-art recognition models. The datasets and code for public use in training and testing shall be made available on GitHub.

Keywords: Visual encoder · textual encoder · CLIP · Knowledge Distillation · Domain Agnostic · Text recognition

1 Introduction

When we consider traditional applications such as image retrieval, understanding, labeling, and machine translation, the methods developed in the past work well by addressing the challenges of arbitrarily oriented and arbitrarily sized text. However, when we consider real-world applications, such as self-driving vehicles, surveillance and monitoring,

namely, theft vehicle tracking and tracking scuba divers under the ocean, the existing models may not be effective (Mokayed et al., 2022). This is because the challenges of above-such images are different compared to natural scene text images. As mentioned, the scene text images pose arbitrary orientations and shaped text. In contrast, underwater images pose poor quality, low contrast, and drone images pose low quality, distortions of tiny text, and loss of text due to occlusion. Furthermore, in the case of drone images, this work considers only license plate numbers (Mokayed et al., 2022; Alkhaled et al., 2023). Indeed, the license plate number does not provide semantic information unlike scene text. Therefore, proposing a model for addressing those challenges is an elusive goal for the researchers. In addition, developing a single domain-independent model makes the problem more complex and challenging.

The sample images of the scene, drone, and underwater images are shown in Fig. 1, where the state-of-the-art models fail to recognize the text in drone and underwater images, while the same models perform well for the text in the scene images. The reason for the poor results by the existing method is that lack of generalization ability and limited scope. On the other hand, the proposed method performs well for images of three different domains. Therefore, one can infer that although the existing models explore the deep learning approaches, the existing methods are not effective for drone and underwater images. In the same way, the proposed new Student-Teacher-Assistant (STA) is effective and domain-independent.

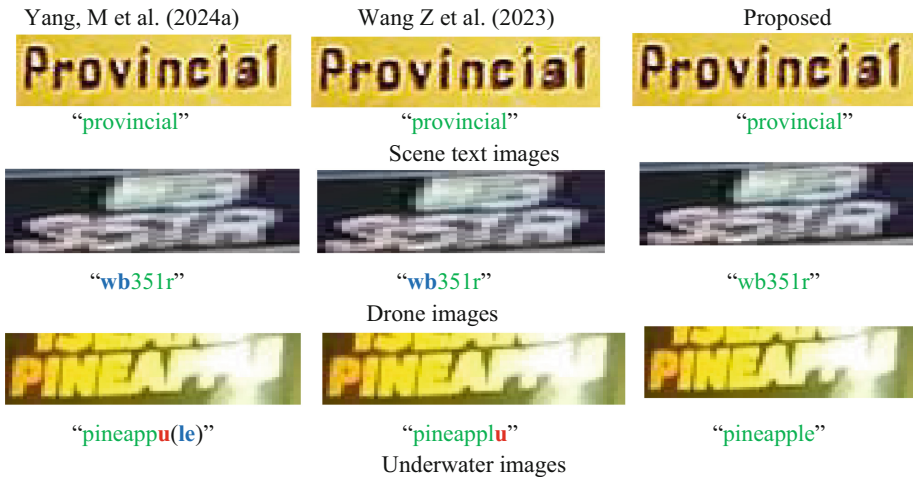


Fig. 1. Performance of the proposed method and the state-of-the-art models for scene, drone and underwater images. Characters in green represent the correct predictions, whereas characters in red represent the incorrect predictions, blue represents missing predictions.

Thus, in this work, inspired by the ability of the CLIP (Radford, A. et al. 2021) model that integrates visual and textual information to extract the fine-tuned features, we propose dual CLIPs for encoding visual and textual features to solve the complex problem of recognition in images of multiple domains. In addition, the proposed work

uses vital information in multiple instances of similar text images. One can find multiple instances of the exact text in the images. This was ignored by the existing models for recognition. Therefore, the proposed work introduces dual CLIPs, one for a primary set, which is input images, and one more for the helper set, which is similar instances of the input image. The way the proposed work designs visual and textual encoders and decoders of the dual CLIPS extract robust and invariant features to achieve the best recognition rate for text of multiple domains. Therefore, the following are the key contributions of the proposed work.

- Introducing a novel student-teacher assistant pipeline network that utilizes primary and helper modules to recognize text with improved confidence scores and accuracy.
- Proposing a new fusion network for integrating merits of visual and textual encoder and decoders of dual CLIPs.

The structure of the paper is organized as follows. The existing methods of scene text recognition are reviewed in Sect. 2. Section 3 presents the proposed framework, including our student-teacher-assistant network. Experimental results are discussed in Sect. 4, and finally, the conclusion and future work are provided in Sect. 5.

2 Related Works

In recent years, we have seen significant progress in scene text recognition. The existing methods addressed several challenges of scene text recognition. However, the scope and objectives of the existing models are limited to a single domain.

Cheng et al. (2023) introduced LISTER, a Length-Insensitive Scene Text Recognizer, which employs a Neighbor Decoder and a Feature Enhancement Module to recognize text regardless of its length accurately. This approach demonstrates superiority in recognizing long text and exhibits length extrapolation capabilities. Yan et al. (2023) proposed an adaptive n-gram transformer for multi-scale scene text recognition (ANT-STR). ANT-STR utilizes an adaptive n-gram embedding to explore semantic correlations between neighboring visual patches and a patch-based n-gram attention mechanism to process feature maps for multi-scale texts. To integrate the advantages of both permuted language modeling (PLM) and masked language modeling (MLM), Yang et al. (2024b) proposed a masked and permuted implicit context learning network for scene text recognition. The model achieves superior performance on popular benchmarks by unifying PLM and MLM within a single decoder and employing perturbation training. Zhang et al. (2023) introduced DPF-S2S, a dual-pathway-fusion-based sequence-to-sequence learning model for text recognition in the wild. DPF-S2S focuses on enriching spatial information and extracting high-dimensional representation features to assist decoding. X. -Y. Ding et al. (2023) introduced a text recognition model tackling unsupervised domain adaptation. It uses dual adaptation on global (text layout) and local (character) features. Adaptive Feature Clustering enhances local adaptation by leveraging source domain knowledge, improving recognition of fine-grained characters across domains.

Wang Z et al. (2023) introduced a Symmetrical Linguistic Feature Distillation strategy that uniquely leverages both visual and linguistic features of CLIP for Scene Text Recognition (STR). It establishes a novel image-to-text feature flow, enhancing STR

accuracy through progressive, layer-by-layer optimization and a new Linguistic Consistency Loss. Nguyen et al. (2024) proposed Diffusion in the Dark (DiD), a diffusion model for low-light image reconstruction for text recognition. DiD provides competitive reconstructions while preserving high-frequency details in boisterous and dark conditions. Jiang et al. (2023) revisited scene text recognition from a data-oriented perspective by consolidating a large-scale real scene text recognition dataset called Union14M. Aberdam et al. (2023) developed CLIPTEr, a framework that harnesses the representative capabilities of modern vision-language models to provide scene-level information to crop-based recognizers. Banerjee et al. (2024) proposed E2EMVSTR, combining cycle consistency, Siamese networks, and semi-supervised attention for scene text recognition. It employs NLP and genetic algorithms for error correction and restoring missing characters, enhancing recognition accuracy from multiple views. Yang et al. (2024c) explored adversarial training’s impact on STR models, proposing a regularization-based method enhancing robustness and accuracy, especially in low-resolution images. Yang et al. (2024a) developed a Class-Aware Mask-guided refinement (CAM) for scene text recognition, using standard font-generated glyph masks to reduce background and style noise, enhancing feature distinction. They also designed an alignment and fusion module with mask guidance for further refinement.

In summary, the above-review shows that although the models successfully addressed several challenges, such as diverse text lengths, multi-scale texts, low-light conditions, the method’s scope is limited to natural scene text images or single domain. Therefore, it is not sure whether the methods work well for the text of multiple domains, such as text of underwater and drone images. Hence, our work aims at developing a novel method based on CLIPS for recognizing text of multiple domains.

3 Proposed Methodology

As discussed in the previous section, we propose a model for recognizing text in scenes and underwater and drone images, which we consider as three domains. Each domain exhibits different characteristics and complexities. In addition to the challenges of scene text recognition, text in underwater images needs better visibility, quality, contrast, and low resolution. In contrast, drone images suffer from loss of text, lack of semantic information, and tiny text. Since afore-mentioned three types of images exhibit different nature, characteristics and complexities, text recognition in three types of images is considered as text recognition in multiple domains. Therefore, developing a single model that can work well for three domains is complex and challenging. To address such a complex problem, motivated by the performance of the CLIP model that fuses visual and textual features for text recognition, we explore the same CLIP models for recognizing text of multiple domains in a novel way. Unlike existing models, the proposed method exploits the intuition of the presence of multiple instances of the same text or similar text. This observation motivated us to introduce dual CLIPs, one for extracting features from the input image (primary) chosen randomly and one more for the feature extraction from the helper image. To implement such an idea, we use the ResNet50 and ViT in CLIP encoders and define them as primary and helper CLIPS, respectively. The two CLIPs are processed simultaneously to extract features from input and helper text images. Further,

the proposed work fuses the features extracted from primary and helper CLIPs for recognizing text in the images of multiple domains, named the Student-Teacher-Assistant (STA) network. The framework of the proposed model is presented in Fig. 2.

In Fig. 2, we use CLIP (Contrastive Language–Image Pre-training) as the image and text encoders inside the Teacher-Assistant pipeline to generate visual and linguistic guidance. It uses the strengths of two CLIP models, giving an efficient dual-modality learning approach. It consists of a Vision Transformer (ViT) based recognition encoder and a decoder in the student pipeline. Its inclusion of CLIP encoders demonstrates more significant interrelatedness between visual and textual data, improving model accuracy in complex recognition tasks.

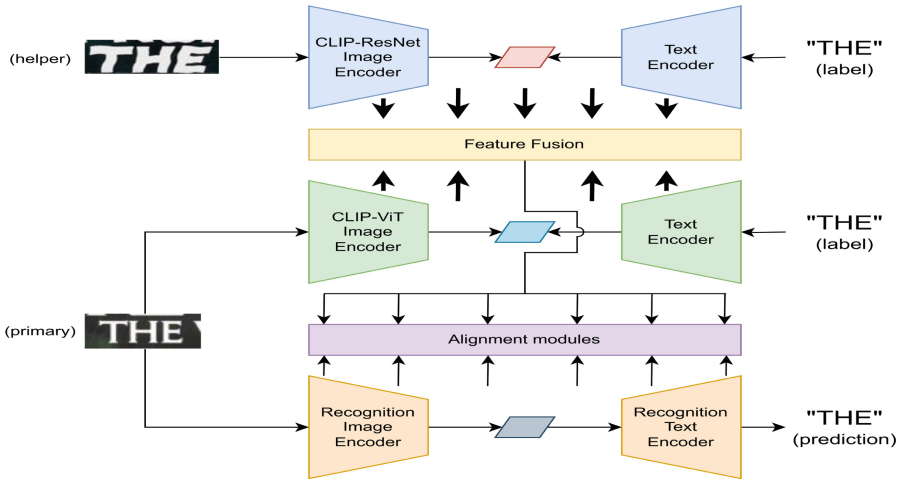


Fig. 2. Framework of the proposed model

3.1 Teacher-Assistant Pipeline

The Teacher-Assistant pipeline features extraction comprises two CLIP models; one uses the ResNet50, and another uses ViT for image and text encoders, with a patch size of 16. Therefore, the proposed work defines the CLIP model with the ResNet50 architecture as helper CLIP and the ViT architecture as primary CLIP. During training, it processes two different images simultaneously using two CLIPs. The helper images and their labels pass through the helper CLIP’s image and text encoder, respectively. Similarly, the primary image and its label pass through the primary CLIP’s image and text encoders. In this way, the proposed dual-pathway approach effectively encodes visual and textual data, which integrates the strengths of two different modalities.

The selection process for the helper and primary batches of images and labels in our Student-Teacher-Assistant(STA) network pipeline is designed to optimize learning efficiency, diverging from random categorization. The primary batch of images and labels are chosen at random. Subsequently, to select the helper batch, we use cosine similarity

measures to identify images resembling those in the primary batch, ensuring thematic or contextual alignment between the pairs. This strategic selection process ensures that both the helper and primary batches are not only relevant but also complementary to each other. Once selected, the helper batch of images and their corresponding labels are processed through the helper CLIP model and, similarly, the primary batch through the primary CLIP model, as previously described. This pairing and processing mechanism significantly enhances the model’s ability to learn from nuanced similarities and differences between the images, fostering a more robust and accurate recognition capability that leverages the full potential of contrastive learning through visual and textual data alignment across multiple domains.

To effectively combine features from two different CLIP models without altering their shapes for use in a recognition pipeline, a Hybrid Feature Fusion (HFF) layer is used. The HFF layer integrates helper information into the main feature pathway without changing the feature shape. For each pair of feature vectors $F_{primary}$ and F_{helper} , a linear transformation is applied to F_{helper} to align its dimensionality with $F_{primary}$. A concatenation of the features is done after the transformation to get all the possible features from the CLIPs, which is named the final concatenated feature vector as F_{cmb} .

3.2 Student Pipeline

The student pipeline approach consists of the recognition encoder and decoder, which are used to save the predictions and the recognition feature list. It is noted that the traditional distillation techniques presume that the input and output formats of the teacher and student models must be identical to perform consistent supervision. However, the proposed dual CLIP models, which comprise only two encoders and lack a decoder, face structural mismatches when distilling knowledge into encoder-decoder models. Since CLIP employs word-level tokenization, which does not guide the character level, we divide the word-level labels into character-level lists for tokenization.

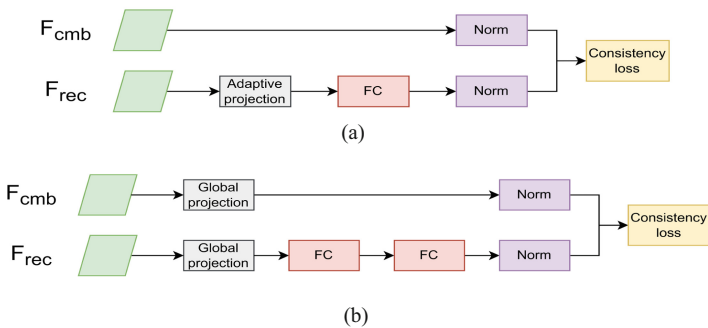


Fig. 3. Structures of adaptive and global alignment modules. (a) represents the adaptive align module and (b) represents the global alignment module

This approach enables us to capture more detailed feature sequences crucial for effective feature distillation. This way, the proposed work matched features from the

Teacher-Assistant pipeline and the Student pipeline to the Adaptive Alignment Modules (AAMs) and Global Alignment Modules (GAMs) (Wang Z et al., 2023). Next, we calculate the loss of the overall architecture, which will be discussed in Sect. 3.3. Aligning the student model directly with the teacher-assistant model may hinder generalization ability. To overcome this problem, we propose Symmetrical Distillation Strategy (Wang et al., 2023), as shown in Fig. 3.

Let’s assume the recognition feature to be F_{rec} with the shape of (N_{rec}, D_{rec}) and the feature from the teacher-assistant pipeline be F_{cmb} with the shape of (N_{cmb}, D_{cmb}) . The AAM first uses an adaptive trainable projection matrix $\mathcal{M} \in (N_{cmb}, N_{rec})$ and one linear layer $W_1 \in (D_{rec}, D_{cmb})$ followed by a ReLU activation layer to project F_{rec} to F_{cmb} ’s feature space and adjust their shapes to be the same. Then a normalization layer is added to undo the effect of magnitude in computation of the consistency loss (Fig. 4).

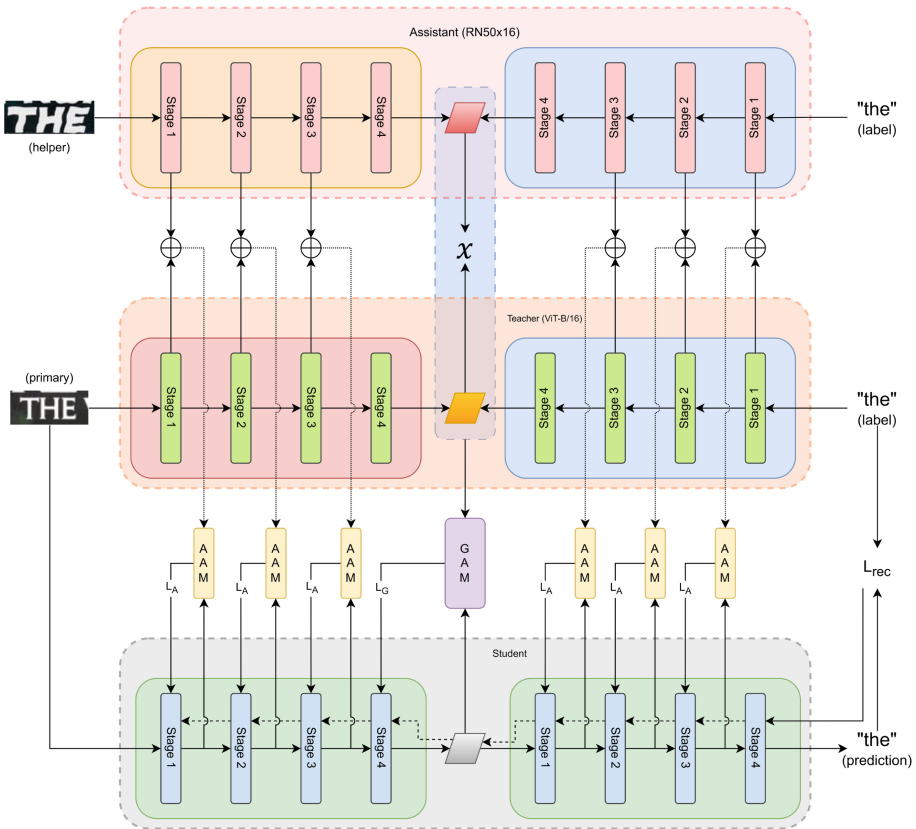


Fig. 4. Overall architecture of the proposed model. Here L_{rec} represents the regular recognition loss, L_A represents the adaptive alignment loss and L_G represents the global alignment loss. AAM is the adaptive alignment module and GAM is the global alignment module. The pretrained assistant CLIP uses ResNet50 in the image encoder and the pretrained teacher CLIP uses ViT in the image encoder. χ is the weight associated with the CLIP embeddings where all the CLIPs and recognition components are divided into four stages.

The formulation of the distillation loss L_A for AAM is defined in Eq. (1).

$$L_A(F_{rec}, F_{cmb}) = L_1(\text{norm}(\text{ReLU}(\mathcal{M} \times F_{rec} \times W_1)), \text{norm}(F_{cmb})) \quad (1)$$

Here, L_1 is a predefined consistency L_1 loss, norm denotes the normalization layer and \times represents matrix multiplication. The GAM is formulated to support the notion of gradually aligning the class token’s guidance of the CLIP image encoder towards the text encoder through joint alignment characteristics. Unlike the AAM, GAM uses another linear layer $W_2 \in (D_{rec}, D_{cmb})$ for projecting the feature space. As defined in Eq. (2), GAM uses a similar approach like AAM but only uses the class token for global projection.

$$L_G(F_{rec}, F_{cmb}) = L_1\left(\text{norm}\left(\text{ReLU}\left(F_{rec}^{cls} \times W_1\right) \times W_2\right), \text{norm}\left(F_{cmb}^{cls}\right)\right) \quad (2)$$

In the above equation, F_{rec}^{cls} and F_{cmb}^{cls} represents the class tokens from the recognition model and the teacher-assistant pipeline.

The overall distillation loss L_D after applying AAM and GAM on the image-to-text flow can be formulated as follows.

$$L_D = \sum_{i=1}^{n-1} L_A\left(F_{rec}^E{}_i, F_{cmb}^I{}_i\right) + L_G\left(F_{rec}^E{}_n, F_{cmb}^T{}_n\right) + \sum_{i=1}^{n-1} L_A\left(F_{rec}^D{}_i, F_{cmb}^T{}_{n-i}\right) \quad (3)$$

Here, F_{rec}^E, F_{rec}^D are the recognition encoder and decoder’s features, F_{cmb}^I, F_{cmb}^T are the combined image and text features from the teacher-assistance pipeline. i is the stage index and n is the number of stages in encoder and decoder which in our case is 4.

3.3 Similarity Based Weighted Linguistic Consistency Loss

Motivated by Wang Z et al. (2023), the proposed work derives the Linguistic Consistency Loss by incorporating a weighted loss measurement system. This novel system considers the similarity between the helper and primary CLIP embeddings and quantifies this relationship using a sophisticated weighting mechanism. Specifically, the similarity score between the CLIP embeddings, derived from the teacher-assistant network, is computed utilizing cosine similarity, a measure reflecting the degree to which the embeddings align in the vector space as formulated as defined in Eq. (4).

$$S_C(P_{CE}, H_{CE}) = \frac{P_{CE} \cdot H_{CE}}{\|P_{CE}\| \|H_{CE}\|} \quad (4)$$

where S_C is the cosine similarity ranging from 0 to 1. P_{CE} and H_{CE} are the embeddings of the primary and secondary CLIP respectively. Subsequently, the sigmoid function determines the weighing of these similarity scores by applying a nuanced approach to evaluating linguistic consistency as defined in Eq. (5).

$$\chi = \sigma(S_C) \quad (5)$$

where χ is the weight of the similarity between the primary and secondary clip embeddings. The weight is calculated by taking the sigmoid, σ of the cosine similarity between the embeddings. This methodology not only enhances the precision of loss calculation but also significantly contributes to the robustness and efficacy of the model by ensuring a more granular and context-sensitive assessment of linguistic features. The consistency of the first order statistics is considered by general distillation losses such as L1 loss. For our task we use L1 loss as defined in Eq. (6).

$$\mathcal{L}_{L1}(F_{rec}, F_{cmb}) = \frac{1}{ND} \|F_{rec} - F_{cmb}\| \quad (6)$$

where $F_{rec}, F_{cmb} \in (N, D)$ are the feature sequences of the student and teacher-assistant pipeline from AAM and GAM alignment modules. To enhance the linguistic knowledge learning efficiency from AAM and GAM, we use the Linguistic Consistency Loss introduced by Wang et al. (2023). There, we combine two kinds of intra and inter losses, which are nothing but contrastive learning loss and the cross-attention map between the recognition and CLIP features. We combine these losses to formulate the final LCL as defined in Eq. (7).

$$\mathcal{L}_{LCL}(F_{rec}, F_{cmb}) = \lambda_1 \mathcal{L}_{intra}(F_{rec}, F_{cmb}) + \lambda_2 \mathcal{L}_{inter}(F_{rec}, F_{cmb}) \quad (7)$$

Here, λ_1 and λ_2 are hyperparameters determined experimentally to fit best with the LCL. Then the linguistic consistency loss is applied to Eq. (1) and Eq. (2). Finally, the total loss is formulated as defined in Eq. (8).

$$L = L_{rec} + (L_D \chi) \quad (8)$$

where L is the total loss, L_{rec} is the regular character level cross-entropy recognition loss, L_D is the distillation loss and χ is the weight of the similarity between the primary and secondary clip embeddings.

4 Experimental Results

Since we aim to evaluate the proposed model for recognizing text in multiple domains, we combine the standard regular and irregular datasets of scene text recognition as the scene domain, our underwear image dataset, and drone images as two more domains, respectively. Therefore, we conducted experiments on each domain to test the performance of the proposed method, which will be discussed in subsequent sub-sections.

4.1 Dataset and Evaluation

Natural Scene Domain. This includes benchmark datasets of IIIT5k, SVT, SVTP, IC13, and IC15. IIIT5k offers 3000 web images, highlighting font, size, and background complexity. SVT, derived from Google Street View, includes 647 images marked by occlusion and low resolution. SVTP, with 645 images, emphasizes perspective distortions in text. IC13 provides 1015 images featuring varied text orientations and scales, while IC15, with 1811 images, focuses on incidental text affected by motion blur and uneven lighting. Collectively, these datasets test the robustness and adaptiveness of text recognition models in natural scenes.

Underwater Domain. The underwater image domain consists of 488 images for training and 141 images for testing, which we created. This domain poses unique challenges, such as turbidity, absorption, and scattering, which degrade image clarity. Successful recognition requires techniques that can handle reduced visibility, distortion, and small text size, making underwater environments crucial for developing robust text recognition algorithms suited to harsh conditions.

Drone Domain. This domain includes our own dataset, which provides 991 cropped license plate number images captured from aerial drone views for training and 278 for testing. Challenges include crowded or multiple license plates, distortions, and varied angles due to drone altitude changes. Addressing these is crucial for enhancing text recognition algorithms in aerial imaging, where text distortion and occlusion demand sophisticated techniques for accurate and reliable recognition.

Implementation Details: Our model is trained with real text-recognition instances from ArT (2019), COCOv2 (2018), LSVT (2019), MLT (2019), RCTW (2017), ReCTS (2019) and UberText (2017) datasets. We trained it on the NVIDIA RTX 4060 GPU system with 36-charset configuration and batch size of 160 with $9e-4$ learning rate. The model is further tested for multiple domains, with the natural scene domain consisting of 6 benchmark datasets, three regulars (IIIT5k, SVT, IC13), and three irregular (IC15, SVTP, CUTE80) datasets. Then, the model is finetuned and tested with the instances of Drone and Underwater domain. For evaluating, we use standard measures, namely, recognition accuracy. For all the existing methods used for comparative study, the same evaluation scheme and process have been used.

4.2 Ablation Study

In this work, we propose a novel Student-Teacher-Assistant (STA) network where the features from the assistant pipeline help the teacher to pass down similar, related, and meaningful features to the student to learn more efficiently. For this process, we conducted experiments on our model with multiple components, each from the teacher and assistant pipeline. We trained our model by using/freezing the primary and helper CLIP embeddings and concatenated text and image features of the Teacher-Assistant pipeline. The recognition accuracy is calculated for only Primary CLIP embeddings, only Helper CLIP embeddings, Primary CLIP embeddings with Image and Textual features, and Helper CLIP embeddings with image and textual features. Furthermore, the recognition accuracy is calculated for combining all the components, which is the proposed model. Results for all the experiments on three domains are presented in Table 1, detailing the model’s performance across different configurations and emphasizing the benefits of using concatenated image/text features.

It is observed from Table 1 that when the Primary CLIP embeddings combine with textual features, the results are improved compared to the Primary CLIP embeddings with image features. The same conclusion can be drawn from the combination of Helper CLIP embeddings with image and textual features. Therefore, one can conclude that input images and their instances provide vital clues for improving the recognition performance of the method. In the same way, the combination of image and textual features helps us

to achieve stable results irrespective of domain. At the same time, Table 1 shows that each component contributes equally and effectively achieves the best performance in the three domains. This approach provides insightful revelations into the contributions of individual and combined feature sets, illustrating the STA network’s capacity to enhance learning through the strategic integration of multimodal information. It is also noted from Table 1 that the recognition accuracy of individual components is not greater than the proposed model. This indicates that individual components cannot cope with the challenges of recognition in multiple domains.

Table 1. Comparison table of the model trained with different active components of the proposed model. P_{CE} refers to the primary CLIP embeddings, H_{CE} refers to the helper CLIP embeddings. IF and TF are both the CLIP’s combined image and text features. **Bold** text represents the result with best model configuration.

Model components				Accuracy (domain)		
P _{CE}	H _{CE}	IF	TF	Natural Scene	Drone	Underwater
✓	-	✓	-	92.94	96.03	65.99
✓	-	-	✓	92.21	95.41	66.73
✓	-	✓	✓	93.20	97.66	69.64
✓	✓	-	-	89.82	93.68	61.40
-	✓	-	✓	93.28	98.28	65.25
-	✓	✓	-	92.11	97.48	68.09
✓	✓	✓	✓	93.34	98.92	70.21

As discussed in the proposed methodology section, Weighted Linguistic Consistency Loss (WLCL) is one more critical step in coping with the challenges of text recognition in multiple domains. To assess the effectiveness of the proposed WLCL, we conducted experiments on all the domains with weighted loss and without weighted loss, as reported in Table 2. The difference in accuracy demonstrates that the similarity weight between the CLIP embeddings affects the model’s learning, resulting in improved recognition accuracy. We trained and tested the model on all the domains to validate the proposed loss function’s efficiency.

Table 2. Assessing the effectiveness of linguistic consistency loss (LCL) and weighted LCL for text recognition in different domains in terms of Accuracy.

Models	Loss	Natural Scene	Drone	Underwater
DATR (Our method)	LCL	92.07	96.40	68.79
DATR (Our method)	Weighted LCL	93.34	98.92	70.21

4.3 Experiments on Domains

Qualitative results of the proposed method on samples of three different domains are shown in Fig. 5, where it can be seen that the text in underwater and drone images is of poor quality compared to the text in scene images. For all images of different domains, the proposed model recognizes text accurately. This shows that the proposed model is domain-agnostic and generic, which can handle the challenges of text recognition of multiple domains.

The above statement can be verified through quantitative results of the proposed and existing methods recorded in Table 3 for the datasets of three domains. The results from our experiments make it clear that our method works better than the current state-of-the-art methods, especially when trained and tested on various domains. When we compare the results of the three domains, the accuracy for the drone datasets is higher than that of the natural scene and underwater image datasets. This is because although the drone domain suffers from tiny, blurry, and noisy text, most of the text has uniform size and horizontal directions in contrast to the scene domain, which has more arbitrarily oriented text texts. In the case of the underwater domain, the visibility of text is very poor compared to the drone and scene text domain. Therefore, the proposed and existing methods report lower results for the underwater domain than the scene and drone domains. However, the existing models report poor results for drone and underwater domains compared to the scene test domain. Since Wang et al. (2023) uses CLIP models for recognizing text in scene images while Yang et al. (2024a) do not, the results of Wang et al. (2023) are better for almost all three domains. This clearly indicates that the CLIP model has the ability to recognize the text affected by adverse effects. When we consider the overall performance of the existing methods on three domains, both methods are inferior to the proposed method. The key reason is that the models were developed for scene text images or single domain, and hence, the methods lack generalization and domain independence abilities. In addition, none of the models use cues in multiple instances for recognition as the proposed method. In our case, the same CLIP model has been explored in a novel way with the help of information in multiple instances, and the proposed method is the best compared to the existing methods for all three domains.

Table 3. Recognition accuracy of the proposed and existing methods across domains. **Bold** text represents the best results.

Models	Year	Natural Scene	Drone	Underwater
X. -Y. Ding et al. (2023)	2023	84.41	-	-
Wang et al. (2023)	2023	92.11	69.06	66.67
Yang, et al. (2024a)	2024	85.15	88.49	55.32
DATR (Our method)	2024	93.34	98.92	70.21



Fig. 5. Proposed model’s recognition results on multiple domains. Characters in green represent the correct predictions, whereas characters in red represent the wrong predictions.

4.4 Experiments on Cross Domain Validation

To validate the domain agnosticism of our proposed model, we conducted multiple sets of experiments by training and fine tuning on a specific domain and testing it on the rest of the domains. It is noted from Table 4 that when we train the proposed model on drone and underwater domains and test it on scene text domain, the performance is higher than that of other combinations. This makes sense because the drone and underwater domains have more diversified samples compared to the natural scene domain. Similarly, when we train on drone and test it on the underwater domain or vice versa, the performance of the proposed model is low. This is due to insufficient samples for learning to address the challenges of drone and underwater domains. The results are better and more reasonable when we train on the natural scene domain and test it on drone or underwater domain.

Table 4. Detailed cross-domain validation results on the proposed method and the existing state-of-the-art methods, where the domain on left represents the domain on which the model is trained/fine-tuned and the domain on right represents the domain on which the model is tested.

#	Cross-Domain Validation	Accuracy		
		Wang et al.	Yang, et al.	Proposed
(i)	Natural Scene → Drone	58.87	67.63	85.97
(ii)	Natural Scene → Underwater	60.79	54.61	68.79
(iii)	Drone → Natural Scene	92.11	74.68	93.07
(iv)	Drone → Underwater	57.45	41.13	66.67
(v)	Underwater → Natural Scene	93.25	83.73	93.30
(vi)	Underwater → Drone	51.08	50.11	54.32

Overall, when we consider the performance of the proposed method on different combinations, the accuracy is promising and reasonable. Therefore, one can assert that the proposed model is domain-agnostic and domain-independent.



Fig. 6. Failure cases of the proposed model. Characters in green represent the correct predictions, whereas characters in red represent the wrong predictions.

It is noted from samples shown in Fig. 6 that the proposed model does not recognize the text accurately. This is because the texts in the samples shown in Fig. 6 are not visible, and even humans cannot read the text in the images with naked eyes. The samples are affected by severe blur and degradation. The proposed method fails to extract compelling features to predict the text in these cases. This is beyond the scope of the proposed work. A possible remedy is integrating the feedback enhancement model with the sequence of CLIPs. This will be our future work.

5 Conclusion and Future Work

We have proposed a novel model called the Student-Teacher-Assistant network for recognizing text in multiple domains, namely, scene text, underwater, and drone domains. Unlike most existing models that focus on a single domain, such as natural scene images or specific types of images, the proposed work focused on three domains. Our model is built based on the fact that multiple instances of text share the same characteristics. This observation motivated us to introduce dual CLIPs called primary for input image and one more called helper for support image. The helper image is obtained using cosine similarity. The features extracted from the image and text encoder of dual CLIPs are fused differently through the alignment approach for recognizing text accurately, irrespective of domains. Experiments on ablation study, each domain, and cross-dataset validation show that the proposed method outperforms the existing methods. The results also show that the proposed model is domain-independent and agnostic. However, as noted from the experimental section, our model fails to recognize the text when the text in the images is not visible and readable even from human eyes. This can be solved by integrating an

image enhancement module with the sequence of CLIPs, which can be explored in the near future.

Acknowledgement. This work is partially supported by IDEAS-TIH, ISI Kolkata.

References

- Aberdam, A., Bensaid, D., Golts, A., et al.: CLIPTER: looking at the bigger picture in scene text recognition. In: Proceedings of ICCV, pp. 21706–21717 (2023)
- Alkhaled, L., Roy, A., Palaiahnakote, S.: An attention-based fusion of ResNet50 and InceptionV3 model for water meter digit recognition. *artificial intelligence and applications* (2022). <https://doi.org/10.47852/bonviewAIA32021197>
- Banerjee, A., Shivakumara, P., Bhattacharya, S., Pal, U., Liu, CL.: An end-to-end model for multi-view scene text recognition. *Pattern Recogn.* **149**, 110206 (2024)
- Cheng, C., Wang, P., Da, C., Zheng, Q., Yao, C.: LISTER: neighbor decoding for length-insensitive scene text recognition. In: Proceedings ICCV, pp. 19541–19551 (2023)
- Karatzas, D., et al.: Icdar 2013 robust reading competition. In ICDAR (2013)
- Karatzas, D., et al.: Icdar 2015 competition on robust reading. In: ICDAR (2015)
- Jiang, Q., Wang, J., Peng, D., Liu, C., Jin, L.: Revisiting scene text recognition: a data perspective. In: Proceedings of ICCV, pp. 20543–20554 (2023)
- Wang, K., Babenko, B., Belongie, S.: End-to-end scene text recognition. In: International Conference on Computer Vision, pp. 1457–1464 (2011)
- Mishra, A., Alahari, K., Jawahar, C.V.: Scene text recognition using higher order language priors. In: Proceeding of BMVC (2012)
- Mokayed, H., Palaiahnakote, S., Alkhaled, L., AL-Masri, A.N.: License plate number detection in drone images. *artificial intelligence and applications* (2022). <https://doi.org/10.47852/bonviewAIA2202421>
- Nayef, N., et al.: Icdar2019 robust reading challenge on multi-lingual scene text detection and recognition—rrc-mlt-2019. In: Proceedings of ICDAR, pp 1582–1587 (2019)
- Nguyen, C.M., Chan, E.R., Bergman, A.W., Wetzstein, G.: Diffusion in the dark: a diffusion model for low-light text recognition. In: Proceedings of WACV, pp. 4146–4157 (2024)
- Phan, T.Q., Shivakumara, P., Tian, S., Tan, C.L.: Recognizing text with perspective distortion in natural scenes. In: Proceedings of ICCV (2013)
- Radford, A., et al.: Learning transferable visual models from natural language supervision. In: Proceedings of PMLR, pp. 8748–8763 (2021)
- Risnumawan, A., Shivakumara, P., Chan, C.S., Tan, C.L.: A robust arbitrary text detection system for natural scene images. *Exp. Syst. Appl.* **41**(18), 8027–8048 (2014)
- Shi, B., Yao, C., Liao, M., et al.: ICDAR2017 competition on reading Chinese text in the wild (RCTW-17). In: Proceedings of ICDAR, pp. 1429–1434 (2017)
- Sun, Y., et al.: ICDAR 2019 competition on large-scale street view text with partial labeling-rrc-lsvt. In: Proceedings of ICDAR, pp. 1557–1562 (2019)
- Quy Phan, T., Shivakumara, P., Tian, S., Lim Tan, C.: Recognizing text with perspective distortion in natural scenes. In: Proceedings of ICCV, pp. 569–576 (2013)
- Veit, A., Matera, T., Neumann, L., et al.: Coco-text: Dataset and benchmark for text detection and recognition in natural images. arXiv preprint [arXiv:1601.07140](https://arxiv.org/abs/1601.07140) (2016)
- Wang, Z., Xie, H., Wang, Y., Xu, J., Zhang, B., Zhang, Y.: Symmetrical linguistic feature distillation with CLIP for scene text recognition. arXiv. (2023). <https://doi.org/10.48550/arXiv.2310.04999>
- Wang, K., Belongie, S.: Word spotting in the wild. In: Proceedings of ECCV, pp 591–604 (2010)

- Yan, X., Fang, Z., Jin, Y.: An adaptive n-gram transformer for multi-scale scene text recognition. *Knowl. Based Syst.* (2023)
- Yang, M., Yang, B., Liao, M., Zhu, Y., Bai, X.: Class-aware mask-guided feature refinement for scene text recognition. *Pattern Recogn.* **149**, 110244 (2024)
- Yang, X., Qiao, Z., Wei, J., Yang, D., Zhou, Y.: Masked and permuted implicit context learning for scene text recognition. *IEEE Sig. Process. Lett.* 31, 964–968 (2024b). <https://doi.org/10.1109/LSP.2024.3381893>
- Yang, X., Yang, D., Qiao, Z., Zhou, Y.: Accurate and robust scene text recognition via adversarial training. In: *Proceedings of ICASSP*, pp 4275–4279 (2024c)
- Zhang, Y., Gueguen, L., Zharkov, I., Zhang, P., Seifert, K., Kadlec, B.: Uber-Text: a large-scale dataset for optical character recognition from street-level imagery. In: *SUNw: Scene Understanding Workshop - CVPR 2017* (2017)
- Zhang, Y., Wu, P., Li, H., Liu, Y., Alsaadi, F.E., Zeng, N.: DPF-S2S: a novel dual-pathway-fusion-based sequence-to-sequence text recognition model. *Neurocomputing.* 182–190 (2023)
- Zhang, R., et al.: ICDAR 2019 robust reading challenge on reading Chinese text on signboard. In: *Proceedings of ICDAR*, pp 1577–1581 (2019)
- Ding, X.-Y., Liu, X.-Q., Luo, X., Xu, X.-S.: DOC: text recognition via dual adaptation and clustering. *IEEE Trans. Multimedia* **25**, 9071–9081 (2023). <https://doi.org/10.1109/TMM.2023.3245404>



DEYOLO: Dual-Feature-Enhancement YOLO for Cross-Modality Object Detection

Yishuo Chen¹ , Boran Wang¹ , Xinyu Guo¹ , Wenbin Zhu¹ ,
Jiasheng He¹ , Xiaobin Liu^{1,2}, and Jing Yuan^{1,2} 

¹ College of Artificial Intelligence, Nankai University, Tianjin 300350, China
wangbr1025@gmail.com

² Engineering Research Center of Trusted Behavior Intelligence, Ministry of
Education, Nankai University, Tianjin 300350, China

Abstract. Object detection in poor-illumination environments is a challenging task as objects are usually not clearly visible in RGB images. As infrared images provide additional clear edge information that complements RGB images, fusing RGB and infrared images has potential to enhance the detection ability in poor-illumination environments. However, existing works involving both visible and infrared images only focus on image fusion, instead of object detection. Moreover, they directly fuse the two kinds of image modalities, which ignores the mutual interference between them. To fuse the two modalities to maximize the advantages of cross-modality, we design a dual-enhancement-based cross-modality object detection network DEYOLO, in which semantic-spatial cross-modality and novel bi-directional decoupled focus modules are designed to achieve the detection-centered mutual enhancement of RGB-infrared (RGB-IR). Specifically, a dual semantic enhancing channel weight assignment module (DECA) and a dual spatial enhancing pixel weight assignment module (DEPA) are firstly proposed to aggregate cross-modality information in the feature space to improve the feature representation ability, such that feature fusion can aim at the object detection task. Meanwhile, a dual-enhancement mechanism, including enhancements for two-modality fusion and single modality, is designed in both DECA and DEPA to reduce interference between the two kinds of image modalities. Then, a novel bi-directional decoupled focus is developed to enlarge the receptive field of the backbone network in different directions, which improves the representation quality of DEYOLO. Extensive experiments on M³FD and LLVIP show that our approach outperforms SOTA object detection algorithms by a clear margin. Our code is available at <https://github.com/chips96/DEYOLO>.

Keywords: Object detection · Visible-infrared · Dual-enhancement

1 Introduction

As a fundamental task of computer vision, object detection in complex scenes still encounters various challenges. Due to the limited wavelength range of visible light, it is difficult to obtain object information in complex environments with poor illumination (*e.g.* heavy smoke). To address this problem, infrared information has been widely introduced. However, due to the low quality of infrared images, it is hard to extract useful texture and color information for general detectors from infrared images. Thus, it is difficult for them to support the detection task alone.

In contrast, utilizing the complementary information in the cross-modality of visible-infrared images can improve the performance in object detection. The commonly used methods adopt fusion-and-detection strategies, which means the image fusion network uses the object detection results as the validation metric. However, the fusion-and-detection methods have several deficiencies. Firstly, fusion of two-modality images does not focus on object detection tasks. Secondly, their redundant model structures (*e.g.* two separate models for fusion and detection, respectively) cause increased training cost as well. Thirdly, although being rich in structure information, infrared (IR) images have a drawback of missing texture. Thus, fusion models usually focus on enriching the texture information while eliminating the complex brightness information of the object. On the contrary, they seldom take the mutual interference between the two modal images into account. *e.g.* infrared images maybe offset the visible imaging quality in fusion process. Only direct image pair fusion without cross-modality enhancement is not sufficient to improve the object detection performance.

Most existing RGB-IR detection models either construct a four-channel input or maintain RGB and infrared images in two separate branches, merging their features downstream. These multi-modality information fusion strategies enhance detection performance to some extent. However, we believe that the interaction between the two modalities is insufficient in these methods. There is a clear boundary between the processing of single-modality images and the feature fusion, resulting in insufficient utilization of cross-modality information. Furthermore, they lack compound interactions at the channel and spatial dimensions, overlooking the potential relationship between semantic and structural information.

To this end, we propose a cross-modality feature fusion approach to dually enhance the feature map of visual and infrared images for detection tasks. This enhancement strategy is able to guide the fusion process of two-modality features from different scales to ensure the integrity of feature information and optimal information extraction. Aiming at object detection, DECA and DEPA are designed to enrich semantic and structure information contained in the feature maps respectively. Moreover, for the purpose of highlighting the modality-specific characteristics, we insert a novel bi-directional decoupled focus in the backbone. It improves the receptive field in the feature extraction stage of DEYOLO multi-directionally, yielding better results. Figure 1 shows the detection results by DEYOLO and DetFusion [23], IRFS [29], PIAFuse [25], SeaFusion [24]

U2Fusion [31]. It can be observed that the proposed DEYOLO achieve better detection results. The contributions of this work are three-fold:

1. We propose the DEYOLO based on YOLOv8 [11], which performs cross-modality feature fusion between the backbone and the detection heads. Different from other fusion methods which directly fuse two-modality images, we fuse two-modality information in feature space and focus on object detection tasks.
2. We propose two modules DECA and DEPA utilizing dual-enhancement mechanism. They reduce interference between two kinds of modalities and achieve semantic and spatial information enhancement by redistributing the weights of channels and pixels.
3. To make the features extracted by the backbone more adaptive to our dual-enhancement mechanism, we design the bi-direction decoupled focus. It down-samples shallow feature maps in different directions, increasing the receptive fields without losing surrounding information.

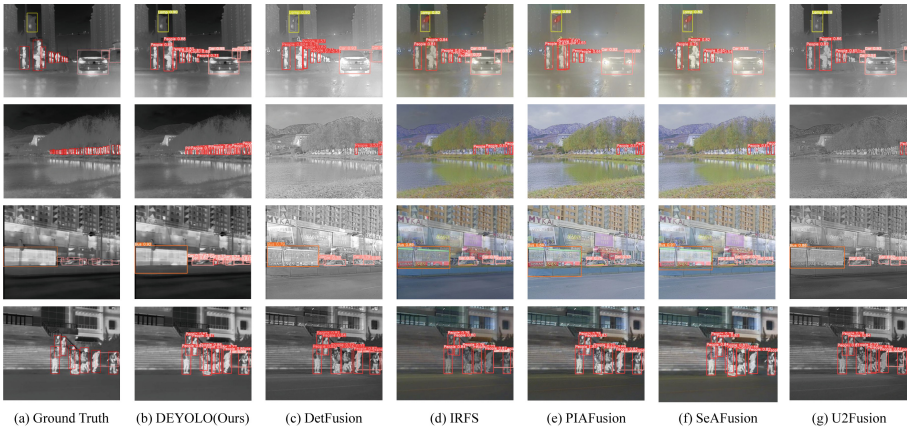


Fig. 1. Detection results of different methods

2 Related Work

In this section, we review the commonly used single-modality object detection algorithms first. Then, some recent visible and infrared image fusion methods are introduced.

2.1 Single-Modality Object Detection

Recently, deep neural networks have been proposed to improve accuracy in object detection tasks, including CNN and its variants, *e.g.* Sparse R-CNN [22], Center-Net2 [36] and the YOLO series [2, 20, 28], as well as Transformer-based models, *e.g.* DETR [3] and Swin Transformer [17]. Although the outstanding performance can be achieved by these models, they all merely utilize information from single-modality images. In addition, these models heavily rely on the texture of the image, which hinders their detection capabilities for infrared images.

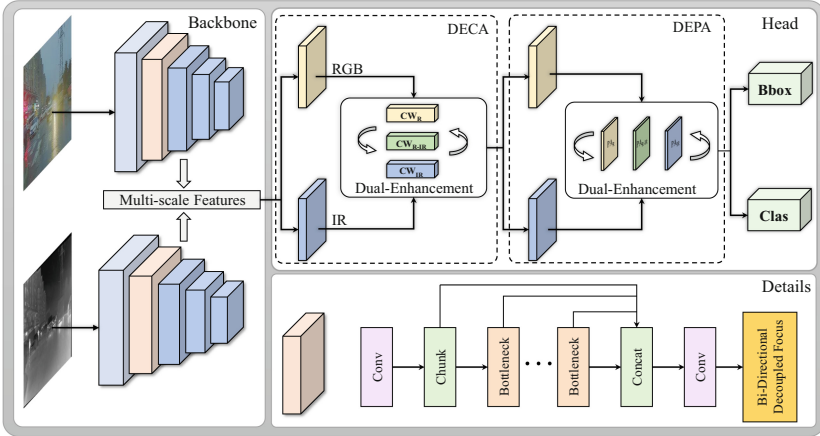


Fig. 2. The framework of the proposed DEYOLO. We incorporate dual-context collaborative enhancement modules (DECA and DEPA) within the feature extraction streams dedicated to each detection head in order to refine the single-modality features and fuse multi-modality representations. Concurrently, the Bi-direction Decoupled Focus is inserted in the early layers of the YOLOv8 backbone to expand the network’s receptive fields.

To handle infrared object detection problems, researchers are continuously introducing different network structures and mechanisms. ALCNet [5] uses backbone to extract the high-level semantic features of the image and a model-driven encoder to learn the local contrast features. ISTDU-Net [30] effectively integrates the encoding and decoding stages and facilitates the transfer of information through hopping connections. This structure is able to increase the receptive field while maintaining a high resolution. IRSTD-GAN [34] treats infrared targets as a special kind of noise. It can predict infrared small targets from the input image based on the data distribution and hierarchical features learned by the GAN. These models only take infrared images into account without extracting information from visible images.

The above single-modality methods are not well suitable for object detection under complex illumination conditions. In contrast, two-modality fusion can extract complementary information from both visible and infrared images, and thus has less over-dependence on texture information.

2.2 Fusion-and-Detection Methods

Considering that infrared images are less vulnerable to poor lighting conditions, various visible and infrared image fusion methods have been proposed.

U2Fusion [31] is an unsupervised end-to-end image fusion network that can solve different fusion problems. It uses feature extraction and information measurement to automatically estimate the importance of the corresponding source images and proposes adaptive information preservation degree. PIAFusion [25] takes the illumination factor into account using an illumination-aware loss. Swin-Fusion [18] involves fusion units based on self-attention [27] and cross-attention, in order to mine long dependencies within the same domain and across domains. CDDFuse [35] introduces a Transformer-CNN extractor and succeeds in decomposing desirable modality-specific and modality-shared features. After the fusion process, the obtained image are fed to a separate model to detect objects.

Although these models can produce convincing results that preserve the adaptive similarity between the fusion result and source images, they don't directly aim at the object detection task. Another drawback is that there may exist conflicts in the fusion results (*e.g.* the textureless patches of infrared images ruin the originally texture-rich ones of visible images), which is harmful to detection accuracy. In contrast, DEYOLO only focuses on object detection and the newly designed dual-enhancement mechanism can tackle the conflict problem.

3 Method

As shown in Fig. 2, to process the multi-scale features extracted from the two-modality images, we add newly designed modules DECAs and DEPAs (Fig. 3) between the backbone and the necks of the YOLOv8 [11] model. Through a specific dual-enhancement mechanism, the fusion of semantic and spatial information makes two-modality features more harmonious. Meanwhile, for the backbone network, to better extract and retain the useful features of both modalities of images, we propose a novel bi-directional decoupled focus strategy. It increases the receptive field of the backbone in different orientations and ensures no leakage of origin information.

3.1 DECA: Dual Semantic Enhancing Channel Weight Assignment Module

The dual enhancement mechanism here refers to the enhancement for two-modality fusion result with single-modality information between the channels and further enhancement for single modality with complementary information from two-modality fusion. Therefore, DECA is able to emphasize the semantic information by distributing weights according to the importance of each channel.

The first enhancement aims to use the single-modality feature to improve the two-modality fusion results of both RGB-IR features, which may contain conflicts. Let $\mathbf{F}_{V_0} \in \mathbb{R}^{b \times c \times h \times w}$ and $\mathbf{F}_{IR_0} \in \mathbb{R}^{b \times c \times h \times w}$ be the feature maps of

visible and infrared images calculated by the backbone, respectively. At first, to get the comprehensive information of RGB and IR images, we concatenate the two features along the channel dimension. Then, a convolution operation will make the combined feature map change to the previous size, filtering the redundant information. As a result, the mixed feature map $\mathbf{F}_{Mix_0} \in \mathbb{R}^{b \times c \times h \times w}$ is obtained:

$$\mathbf{F}_{Mix_0} = conv(concat(\mathbf{F}_{V_0}, \mathbf{F}_{IR_0})) \quad (1)$$

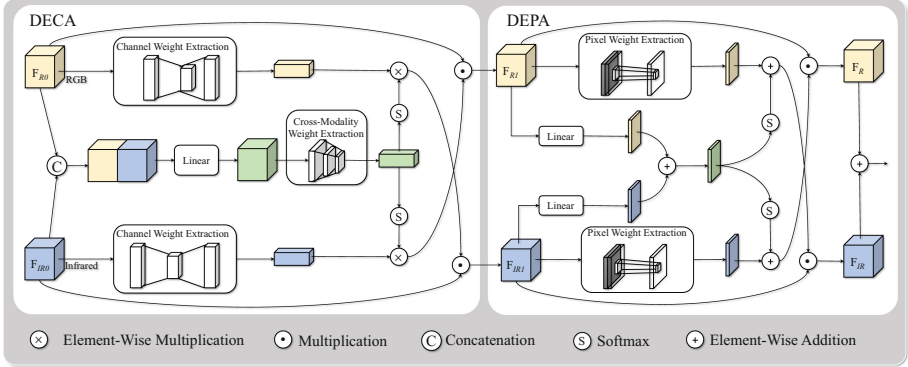


Fig. 3. The concrete structure of DECA and DEPA. These modules utilize both single-modality and cross-modality information through a dual enhancement mechanism. DECA enhances the cross-modality fusion results by leveraging dependencies between channels within each modality and outcomes are then used to reinforce the original single-modal features, highlighting more discriminative channels. Similarly, DEPA is able to learn dependency structures within and across modalities to produce enhanced multi-modality representations with stronger positional awareness

Next, we propose a novel weight-encoding method through convolution. An encoder is designed to squeeze \mathbf{F}_{Mix_0} in the spatial dimension progressively to the size of $\mathbb{R}^{b \times c \times 1 \times 1}$:

$$\mathbf{W}_{Mix_0} = CMWE(\mathbf{F}_{Mix_0}) \in \mathbb{R}^{b \times c \times 1 \times 1} \quad (2)$$

where $CMWE(\cdot)$ refers to the cross-modality weight extraction operation in Fig. 3.

On the other hand, we need to acquire the specific feature of each modality. The SE block [7] explicitly models the interdependencies between the channels of its convolutional features for improving the quality of the feature map representation. Motivated by this idea, we feed this structure with visible and infrared images to get the feature blocks of size $\mathbb{R}^{b \times c \times 1 \times 1}$, which represents the weight values of different channels:

$$\begin{cases} \mathbf{W}_{V_0} = CWE(F_{V_0}) \in \mathbb{R}^{b \times c \times 1 \times 1} \\ \mathbf{W}_{IR_0} = CWE(F_{IR_0}) \in \mathbb{R}^{b \times c \times 1 \times 1} \end{cases} \quad (3)$$

where $CWE(\cdot)$ refers to the channel weight extraction block in Fig. 3. \mathbf{W}_{V_0} and \mathbf{W}_{IR_0} can enhance the mixed feature of the two modalities by element-wise multiplication to redistribute weights, which is able to highlight significant channels:

$$\begin{cases} \mathbf{W}_{enV_0} = \mathbf{W}_{V_0} \otimes \text{softmax}(\mathbf{W}_{Mix_0}) \\ \mathbf{W}_{enIR_0} = \mathbf{W}_{IR_0} \otimes \text{softmax}(\mathbf{W}_{Mix_0}) \end{cases} \quad (4)$$

For the second enhancement, we attempt to make each feature map of RGB and IR fully utilize the respective advantages of another modality. To this end, \mathbf{F}_{V_0} and \mathbf{F}_{IR_0} will multiply the corresponding feature weights acquired in the first enhancement to get semantic and textural information from another modality:

$$\begin{cases} \mathbf{F}_{IR_1} = \mathbf{F}_{IR_0} \odot \mathbf{W}_{enV_0} \\ \mathbf{F}_{V_1} = \mathbf{F}_{V_0} \odot \mathbf{W}_{enIR_0} \end{cases} \quad (5)$$

where \odot is multiplication in channel dimension. The enhancement results $\mathbf{F}_{V_1} \in \mathbb{R}^{b \times c \times w \times h}$ and $\mathbf{F}_{IR_1} \in \mathbb{R}^{b \times c \times w \times h}$ will pass through the DEPA described below.

3.2 DEPA: Dual Spatial Enhancing Pixel Weight Assignment Module

Similar with DECA, DEPA adopts the dual enhancement mechanism as well. Re-encoded in the spatial dimension, DEPA emphasizes important pixel positions while minimizing the irrelevant ones.

Specifically, to obtain the mixed feature including global information, we perform a shape transformation for the two feature maps \mathbf{F}_{V_1} and \mathbf{F}_{IR_1} using convolution. Then, an element-wise multiplication is applied on the result of each other:

$$\mathbf{W}_{Mix_1} = \text{conv}(\mathbf{F}_{V_1}) \otimes \text{conv}(\mathbf{F}_{IR_1}) \quad (6)$$

Afterwards, a softmax operation is performed on \mathbf{W}_{Mix_1} . In order to fully obtain the feature specific to each modality in spatial dimension, we maintain the differences in spatial information learned by different convolutional kernel sizes.

$$\begin{cases} \mathbf{W}_{IR_1temp} = \text{concat}(\text{conv}_1(\mathbf{F}_{IR_1}), \text{conv}_2(\mathbf{F}_{IR_1})) \\ \mathbf{W}_{V_1temp} = \text{concat}(\text{conv}_1(\mathbf{F}_{V_1}), \text{conv}_2(\mathbf{F}_{V_1})) \end{cases} \quad (7)$$

In Eq. (7), two convolution operations are used to extract the pixel weights from distinct scales. By concatenating them in the channel dimension, we can obtain $\mathbf{W}_{IR_1} \in \mathbb{R}^{b \times 2 \times w \times h}$ and $\mathbf{W}_{V_1} \in \mathbb{R}^{b \times 2 \times w \times h}$. Then, we compress the feature by reducing the number of channels by half and obtain $\mathbf{W}_{IR_1} \in \mathbb{R}^{b \times 1 \times w \times h}$ and $\mathbf{W}_{V_1} \in \mathbb{R}^{b \times 1 \times w \times h}$. The element-wise multiplication by the softmaxed \mathbf{F}_{Mix_1} is applied on \mathbf{W}_{IR_1} and \mathbf{W}_{V_1} :

$$\begin{cases} \mathbf{W}_{enIR_1} = \mathbf{W}_{IR_1} \otimes \text{softmax}(\mathbf{F}_{Mix_1}) \\ \mathbf{W}_{enV_1} = \mathbf{W}_{V_1} \otimes \text{softmax}(\mathbf{F}_{Mix_1}) \end{cases} \quad (8)$$

The second enhancement is implemented by an element-wise multiplication operation between the input feature maps and the results of first enhancement:

$$\begin{cases} \mathbf{F}_{IR} = \mathbf{F}_{IR_1} \odot \mathbf{W}_{enV_1} \\ \mathbf{F}_V = \mathbf{F}_{V_1} \odot \mathbf{W}_{enIR_1} \end{cases} \quad (9)$$

Equation (9) aims to extract structural feature from another modality in spatial dimension. In the end, we do element-wise addition on \mathbf{F}_{IR} and \mathbf{F}_V for the object detection.

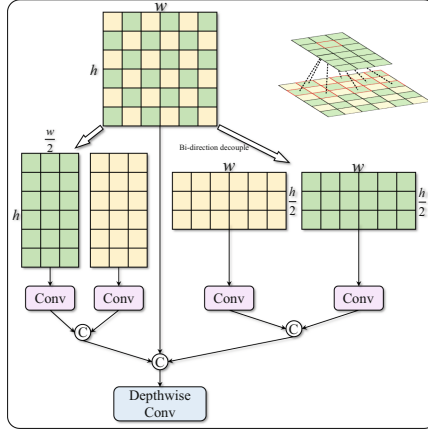


Fig. 4. Bi-direction decoupled focus.

3.3 Bi-direction Decoupled Focus

In this subsection, we tend to improve the performance of object detection from the perspective of the single modality. In order to enhance the capability of extracting targets, the bi-direction decoupled focus is designed to enlarge the receptive field of the backbone in DEYOLO while minimizing the loss of surrounding pixels.

The focus block in YOLOv5 [10] is a slicing operation, which is improved from the passthrough layer in YOLOv2 [21]. This specific operation gets a pixel in an image with an interval by one pixel and thus can provide a two-fold downsampled feature map without an information loss.

Inspired by this downsampling method, we design bi-direction decoupled focus to retain the information adequately in multi-directions. Specifically, we adopt two specific sampling and encoding rules implemented horizontally and vertically. As shown in Fig. 4, we divide the pixels into two groups for convolution. Each group focuses on the adjacent and remote pixels at the same time. Finally, we concatenate the original feature map in the channel dimension and make it go through a depth-wise convolution [4] layer.

4 Experiments

4.1 Datasets

Since infrared images are obtained by measuring the heat radiation emitted from objects, they are susceptible to noises in the environment. In fact, only a small number of high-quality datasets composed of infrared and visible images are available, such as TNO [26] and RoadScene [32]. However, these datasets often aim at infrared and visible image fusion tasks, rather than object detection, thus the labels for object detection are absent. The FLIR [1] dataset provides annotations for object detection but it lacks pixel-level alignment. Therefore, we choose the public datasets M³FD [15], LLVIP [9], and KAIST [8] which are pixel-wise aligned for infrared-visible image pairs and contain annotations for object detection. Among these, the M³FD dataset comprises 4,200 image pairs, totaling 8,400 images. The LLVIP dataset includes 16,836 image pairs, amounting to a total of 33,672 images. Considering the original KAIST dataset contains noisy annotations, we use a cleaned version of the training set (7,601 examples) and the testing set (2,252 examples).

4.2 Implementation Details

In this subsection, two sets of experiments are conducted to verify the effectiveness of DEYOLO. One is the comparison with the SOTA single-modality object detection algorithms and the other is the comparison with the fusion-and-detection algorithms. When training single-modality detection algorithms, we use infrared and visible images to train the model, respectively. For the sake of experimental fairness, we also combined the visible and infrared images from the datasets to serve as the training set of these detector. For the fusion-and-detection algorithms, the pre-trained image fusion models for cross-modality fusion are adopted in the comparison algorithms, and then the fused images are further used to train YOLOv8 [11]. The training is performed on eight NVIDIA RTX 4090 GPUs. The number of epochs for training is 800, the batch size is 64, the initial and final learning rates are 1×10^{-2} and 1×10^{-4} , respectively. And, we evaluate our method on the validation set and use the mean average precision (mAP) with the IoU threshold of 0.5 and Log Average Miss Rate (LAMR) as the evaluation metric.

4.3 Ablation Studies

To validate the impact of the key components in DEYOLO, we conducted a number of experiments on the M³FD [15] dataset to investigate how they affect our final performance.

Firstly, we verify the impact of the use of the bi-directional decoupled focus, DECA and DEPA modules on the model, respectively. The experimental results are shown in Table 1. It can be seen that DECA and DEPA improve the detection accuracy of the model more obviously. The use of DECA and DEPA modules

Table 1. Ablation studies on the M³FD dataset. Bi-direction stands for using bi-direction decoupled focus on the backbone. DECA stands for using the DECA module. DEPA stands for using the DEPA module.

Bi-direction	DECA	DEPA	mAP ₅₀	mAP ₅₀₋₉₅
			80.8	54.3
	✓		85	58.7
		✓	84.4	57.8
	✓	✓	85.2	58.9
✓	✓	✓	86.6	59.6

alone improves mAP₅₀ by 4.2% and 3.6%, as well as mAP₅₀₋₉₅ by 4.4% and 3.5%, compared to the baseline network trained merely by visible images. While the improvement of DECA is more obvious than that of DEPA. The joint use of them improves mAP₅₀ by 4.4% and mAP₅₀₋₉₅ by 4.6%, respectively. Moreover, the object detection accuracy is further improved using all three modules at the same time, with the two metrics improving by 5.8% and 5.3%, respectively.

In the DECA and DEPA modules, the channel weights and spatial pixel weights, which incorporate both semantic and spatial information from two modalities, are utilized to respectively enhance the semantic and structural information within the single-modality channel weights and spatial pixel weights. The enhanced weights are then applied to the single-modality feature maps to achieve dual enhancement. By fully leveraging the advantages of each modality and their complementary information within the feature space, the use of DECA and DEPA results in improving the performance of cross-modality object detection. Since we are utilizing deep features, each feature map contains stronger semantic information compared to spatial information. As a result, the enhancement effect of DECA on the model is more pronounced compared to that of DEPA.

Furthermore, in order to investigate how to make the dual enhancement mechanism in DECA and DEPA relieves the interference between two-modality images and obtain cross-modality channel weights and pixel weights better, we choose different hyperparameters in the feature mixing part in DEPA and cross-modality weight extraction part in DECA, respectively.

Table 2. Performance of different kernel sizes used in DEPA to get the mixed feature.

Layer	Kernel Size	mAP ₅₀	mAP ₅₀₋₉₅
Conv	3 × 3	85.3	58.9
	5 × 5	85.1	58.4
	7 × 7	85.1	58.1

For DEPA, we use different convolution kernel sizes to get the spatial pixel weights of two modalities. The results are shown in Table 2. We believe that as the convolution kernel size increases, more and more redundant information within each single modality is also integrated, thereby increasing mutual interference between the two modalities and hindering feature enhancement. It is found that for feature maps with different scales, when the number of convolutional layers is the same, the kernel size of 3×3 can better model the spatial pixel information.

Table 3. Performance of different ways to generate W_{Mix_0} through Cross-Modality Weight Extraction in DECA.

Layer	Number of Layers	mAP ₅₀	mAP ₅₀₋₉₅
Conv	1	X	X
	2	84.5	58.1
	3	84.9	57.8
Depth-wise Conv	2	84.5	58.3
	3	85.2	58.9

For DECA, we try to use different types of convolutions with different numbers of layers for cross-modality channel weight extraction. The experiment results are shown in Table 3. We firstly attempt to directly extract the weights of each channel through one layer of convolution with the same size as the original feature map. However, we find that the model cannot converge if the layer number is set to 1. Then, we set the number of convolution layers to 2 and 3 successively, and find that the weights of each channel can be better extracted when it is 3. For channel weight extraction, we find that the depth-wise convolution [4] is more suitable for guiding the training process because of its fast convergence rate, which demonstrates its advantages.

4.4 Comparison with State-of-the-Arts Models

At last, we compare DEYOLO with recent state-of-the-art fusion models and object detection models on the M³FD [15] and LLVIP [9] datasets. Here we select YOLOv8-n and YOLOv8-l as our baseline.

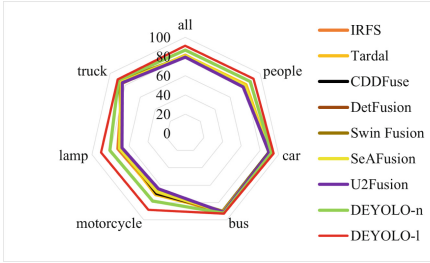
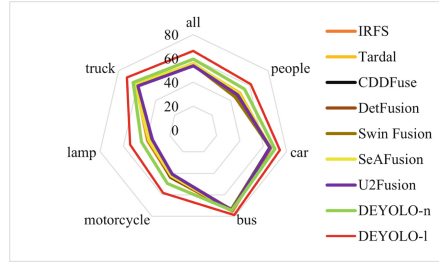
As shown in Table 4, due to utilization of different information from two modalities, DEYOLO outperforms all single-modality object detection models. In addition, mAPs of the detectors trained using visible images are higher than those of the detectors trained with infrared images. But none of the single-modality detectors can surpass DEYOLO, which uses the dual feature enhancement mechanism. Particularly, DEYOLO outperforms ViT-based models, such as Swin Transformer [17] and Sparse RCNN [22]. The ViT-based models only considers single-modality global correlation, while DEYOLO additionally uses

Table 4. Performance comparison with other detectors. Visible stands for training the model using visible images, infrared stands for training the model using infrared images. Cross-modality stands for using two-modality images for training.

Method	Modality	mAP ₅₀	mAP ₅₀₋₉₅
Swin Transformer [17]	visible	76.4	44.9
	infrared	72.6	41.9
	cross-modality	73.8	42.6
CenterNet2 [36]	visible	78.5	52.4
	infrared	65.3	42.4
	cross-modality	70.2	46.5
Sparse RCNN [22]	visible	82.4	49.6
	infrared	76.4	44.8
	cross-modality	78.2	47.3
YOLOv7-tiny [28]	visible	82.1	51.6
	infrared	78.1	48.4
	cross-modality	80.1	49.8
YOLOv7 [28]	visible	90.4	61.3
	infrared	87.9	58.3
	cross-modality	88.3	59.6
YOLOv8n [11]	visible	80.8	54.3
	infrared	78.3	52.3
	cross-modality	79.2	52.8
YOLOv8l [11]	visible	88.3	61.8
	infrared	86.5	59.6
DEYOLO-n (ours)	Cross-modality	86.6	58.9
DEYOLO-l (ours)	Cross-modality	91.2	66.3

the complementary information between two modalities extracted by DECA and DEPA without conflicts.

It can be observed that some fusion-and-detect methods, such as DetFusion [23] and U2Fusion [31], as shown in Fig. 1(b) and (d), produce fused images which look more like the infrared images, lacking partial texture and color information required for detection tasks. On the other hand, the fused images obtained by the other methods including SeAFusion [24] and Tardal [15], do not effectively capture rich structural information in the infrared image (e.g., Fig. 1(c)). The comparison methods fail to balance the texture and structure information of both modalities to improve the detection accuracy. In contrast, DEYOLO first exploits the advantages of both modalities through bi-direction decoupled focus and then utilizes the DECA and DEPA modules based on a dual-enhancement mechanism to reduce the mutual interference between the two modalities, thereby improving the detection accuracy.

Fig. 5. mAP_{50} in specific categoriesFig. 6. mAP_{50-95} in specific categories

As shown in Table 5, the performance of our method on both datasets is better than that of the state-of-the-art fusion-and-detection methods. Specifically, in M³FD [15] dataset the mAP_{50} and mAP_{50-95} of DEYOLO-n are higher than those of the other models by 5.4% and 3.1% at least, respectively. And the improvement of the mAP_{50} and mAP_{50-95} of DEYOLO-l can reach more than 10.0% and 10.5%, respectively. Meanwhile, in LLVIP [9] dataset, we observe at least 0.6% and 1.4 % improvement on the mAP_{50} and mAP_{50-95} of DEYOLO-

Table 5. Performance comparison with fusion-and-detection works.

Dataset	Method	Modality	mAP_{50}	mAP_{50-95}
M ³ FD [15]	IRFS [29]	cross-modality	81.2	55.8
	Tardal [15]		81.0	54.9
	CDDFuse [35]		80.3	54.9
	PIAFusion [25]		80.6	54.9
	Swin Fusion [18]		80.2	54.7
	DetFusion [23]		80.6	55.0
	SeAFusion [24]		80.7	55.4
	U2Fusion [31]		79.2	53.8
	DEYOLO-n (ours)		86.6	58.9
	DEYOLO-l (ours)		91.2	66.3
LLVIP [9]	IRFS [29]	cross-modality	94.0	60.7
	Tardal [15]		94.5	63.3
	CDDFuse [35]		92.1	57.5
	PIAFusion [25]		96.1	62.4
	Swin Fusion [18]		93.3	59.4
	MFEIF [16]		95.8	64.0
	SeAFusion [24]		96.2	64.0
	U2Fusion [31]		92.2	58.3
	DEYOLO-n (ours)		96.8	65.4

n, respectively. In addition, in Fig. 5 and Fig. 6, the detection results of every category in M³FD dataset also shows the superiority of our method. We have re-split the datasets into training, validation, and test sets in a 3:1:1 ratio. After dividing the test set as described above, the mAP₅₀ on the test/validation sets of the two datasets are 85.7%/86.6% and 96.4%/96.8%, respectively.

To validate the generalization ability of our model, experiments were conducted on the KAIST dataset, as shown in Table 6. Unlike the M³FD and LLVIP datasets, KAIST consists of pairs of RGB and thermal images. Thermal images, unlike infrared images studied in our research, exhibit lower imaging quality and significant differences. Therefore, these experiments serve as an extended validation of our model. From Table 6, it is evident that our method does not achieve state-of-the-art (SOTA) performance but outperform the majority of existing methods.

Table 6. Comparison with other RGB-T detectors on KAIST dataset.

Methods	ALL	Day	NIGHT
RPN+BDT [14]	29.83	30.51	27.62
TC-DET [12]	27.11	34.81	10.31
Halfway Fusion [19]	25.75	24.88	26.59
IATDNN [6]	26.37	27.29	24.41
IAF R-CNN [13]	20.59	21.85	18.96
CIAN [33]	14.12	14.77	11.13
DEYOLO (ours)	15.45	17.23	12.23

5 Conclusion

In this paper, we propose DEYOLO using the dual enhancement mechanism for cross-modality object detection in complex-illumination environments. DECA and DEPA are designed to fuse the feature maps of two modalities between the backbone and the detection heads. And the bi-direction decoupled focus is proposed in the backbone to improve the feature extraction capability. The superiority of this method is verified on two datasets. It is worthwhile to point out that, both DECA and DEPA proposed in this paper can be used as a plug-and-play module for wider applications in other models to solve the problem of object detection in complex environments. And this will be the topic in our future work.

Acknowledgement. This work was supported in part by the Natural Science Foundation of China under Grant U21A20486, 62473208 and 62401294, in part by the Tianjin Science Fund for Distinguished Young Scholars under Grant 20JCJQJC00140, in part by the major basic research projects of the Natural Science Foundation of Shandong

Province under Grant ZR2019ZD07, in part by the Postdoctoral Fellowship Program of CPSF under Grant GZC20240753, and in part by the Fundamental Research Funds for the Central Universities under Grant 078-63243158.

References





1. FLIR: FLIR thermal dataset for algorithm training (2018). <https://www.flir.in/oem/adas/adas-dataset-form>
2. Bochkovskiy, A., Wang, C.Y., Liao, H.Y.M.: YOLOv4: optimal speed and accuracy of object detection. arXiv preprint [arXiv:2004.10934](https://arxiv.org/abs/2004.10934) (2020)
3. 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)
4. Chollet, F.: Xception: deep learning with depthwise separable convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1251–1258 (2017)
5. 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)
6. 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)
7. 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)
8. Hwang, S., Park, J., Kim, N., Choi, Y., So Kweon, I.: Multispectral pedestrian detection: benchmark dataset and baseline. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1037–1045 (2015)
9. Jia, X., Zhu, C., Li, M., Tang, W., Zhou, W.: LLVIP: a visible-infrared paired dataset for low-light vision. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 3496–3504 (2021)
10. Jocher, G.: YOLOv5 by ultralytics (2020). <https://doi.org/10.5281/zenodo.3908559>. <https://github.com/ultralytics/yolov5>
11. Jocher, G.: ultralytics/yolov8: v8.1.0 - YOLOv8 oriented bounding boxes (OBB) (2024). <https://github.com/ultralytics/ultralytics>
12. Kieu, M., Bagdanov, A.D., Bertini, M., Del Bimbo, A.: Task-conditioned domain adaptation for pedestrian detection in thermal imagery. In: European Conference on Computer Vision, pp. 546–562. Springer (2020)
13. Li, C., Song, D., Tong, R., Tang, M.: Illumination-aware faster R-CNN for robust multispectral pedestrian detection. *Pattern Recogn.* **85**, 161–171 (2019)
14. Liu, J., Zhang, S., Wang, S., Metaxas, D.N.: Multispectral deep neural networks for pedestrian detection. arXiv preprint [arXiv:1611.02644](https://arxiv.org/abs/1611.02644) (2016)
15. Liu, J., et al.: Target-aware dual adversarial learning and a multi-scenario multi-modality benchmark to fuse infrared and visible for object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5802–5811 (2022)
16. Liu, J., Fan, X., Jiang, J., Liu, R., Luo, Z.: Learning a deep multi-scale feature ensemble and an edge-attention guidance for image fusion. *IEEE Trans. Circuits Syst. Video Technol.* **32**(1), 105–119 (2021)

17. 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)
18. Ma, J., Tang, L., Fan, F., Huang, J., Mei, X., Ma, Y.: SwinFusion: cross-domain long-range learning for general image fusion via swin transformer. *IEEE/CAA J. Autom. Sin.* **9**(7), 1200–1217 (2022)
19. Park, K., Kim, S., Sohn, K.: Unified multi-spectral pedestrian detection based on probabilistic fusion networks. *Pattern Recogn.* **80**, 143–155 (2018)
20. 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)
21. Redmon, J., Farhadi, A.: YOLO9000: better, faster, stronger. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7263–7271 (2017)
22. Sun, P., et al.: Sparse R-CNN: end-to-end object detection with learnable proposals. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 14454–14463 (2021)
23. Sun, Y., Cao, B., Zhu, P., Hu, Q.: DetFusion: a detection-driven infrared and visible image fusion network. In: Proceedings of the 30th ACM International Conference on Multimedia, pp. 4003–4011 (2022)
24. Tang, L., Yuan, J., Ma, J.: Image fusion in the loop of high-level vision tasks: a semantic-aware real-time infrared and visible image fusion network. *Inf. Fusion* **82**, 28–42 (2022)
25. Tang, L., Yuan, J., Zhang, H., Jiang, X., Ma, J.: PIAFusion: a progressive infrared and visible image fusion network based on illumination aware. *Inf. Fusion* **83**, 79–92 (2022)
26. Toet, A.: The TNO multiband image data collection. *Data Brief* **15**, 249–251 (2017)
27. Vaswani, A., et al.: Attention is all you need. In: *Advances in Neural Information Processing Systems* 30 (2017)
28. 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)
29. Wang, D., Liu, J., Liu, R., Fan, X.: An interactively reinforced paradigm for joint infrared-visible image fusion and saliency object detection. *Inf. Fusion* **98**, 101828 (2023)
30. Hou, Q., Zhang, L., Tan, F., Xi, Y., Zheng, H., Li, N.: ISTDU-Net: infrared small-target detection U-Net. *IEEE Geosci. Remote Sens. Lett.* **19**, 1–5 (2022). Art no. 7506205. <https://doi.org/10.1109/LGRS.2022.3141584>
31. Xu, H., Ma, J., Jiang, J., Guo, X., Ling, H.: U2Fusion: a unified unsupervised image fusion network. *IEEE Trans. Pattern Anal. Mach. Intell.* **44**(1), 502–518 (2020)
32. Xu, H., Ma, J., Le, Z., Jiang, J., Guo, X.: FusionDN: a unified densely connected network for image fusion. In: Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence (2020)
33. Zhang, L., et al.: Cross-modality interactive attention network for multispectral pedestrian detection. *Inf. Fusion* **50**, 20–29 (2019)
34. Zhao, B., Wang, C., Fu, Q., Han, Z.: A novel pattern for infrared small target detection with generative adversarial network. *IEEE Trans. Geosci. Remote Sens.* **59**(5), 4481–4492 (2020)

35. Zhao, Z., et al.: CDDFuse: correlation-driven dual-branch feature decomposition for multi-modality image fusion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5906–5916 (2023)
36. Zhou, X., Koltun, V., Krähenbühl, P.: Probabilistic two-stage detection. arXiv preprint [arXiv:2103.07461](https://arxiv.org/abs/2103.07461) (2021)



MarUCOD: Unknown but Concerned Object Detection in Maritime Environments

Hajung Yoon¹ , Yoonji Lee¹ , Hwijun Lee¹ , Daeho Um² ,
Hong Seok Choi³ , and Jin Young Choi²  

- ¹ Automation and Systems Research Institute (ASRI), Interdisciplinary Program in Artificial Intelligence, Seoul National University, Seoul 08826, South Korea
² Automation and Systems Research Institute (ASRI), Department of Electrical and Computer Engineering, Seoul National University, Seoul 08826, South Korea
jychoi@snu.ac.kr
³ NexReal Inc., 178, Digital-ro, Geumcheon-gu, Seoul, Republic of Korea

Abstract. Detection of unknown but concerned objects such as diverse unknown suspicious objects in the sea area is a critical problem in military defense applications, but the problem is challenging because 1) a pre-trained detector cannot easily detect unknown but concerned objects, 2) it detects too many unconcerned objects such as usual objects in the coastal land area, and 3) we cannot easily establish a well-performing discriminator to divide the detected objects into unknown and known objects, as well as concerned and unconcerned objects because the unknown objects are not available, whereas the concerned and unconcerned objects are not clearly defined. To tackle this challenge, this paper proposes a real-time framework for unknown but concerned object detection in maritime environments by integrating object detection, segmentation, and out-of-distribution (OOD) detection techniques. In our framework, an object detector finds all object-like foregrounds by setting a low threshold and a segmentation deep-learning network filters out unconcerned foregrounds detected in the coastal land area. After that, to discriminate known or unknown objects among concerned objects detected in the sea area, a discriminator performs unsupervised OOD detection using bisecting K-means clustering. To boost the performance of the proposed framework, we apply a pre-processing scheme and a contrastive separation loss for segmentation. The proposed framework achieves a high detection rate of unknown but concerned objects with minimal detection of unconcerned objects (*i.e.*, minimal false positives), surpassing baseline methods and demonstrating potential for enhanced maritime safety and security. The codes are open at <https://github.com/AIX-Coast-Defense-PIL/MarUCOD>.

H. Yoon, Y. Lee and H. Lee—These authors contributed equally.

Supported by IITP grant funded by Korea government(MSIT) [No.2021-0-01343, Artificial Intelligence Graduate School Program (Seoul National University); No. A1101-23-1002, Information and Communication Promotion].

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
A. Antonacopoulos et al. (Eds.): ICPR 2024, LNCS 15317, pp. 253–268, 2025.
https://doi.org/10.1007/978-3-031-78447-7_17

Keywords: Maritime surveillance · unknown but concerned object detection · area segmentation · out-of-distribution detection

1 Introduction

In maritime environments, detecting unknown objects that are not usually visible is crucial for military defense. Among unknown objects, those floating on the sea are the primary concern in maritime surveillance. These unknown but concerned objects including suspicious objects such as naval mines or rafts can serve as means of enemy attack or intrusion. Implementing an automated system for detecting unknown but concerned objects has a wide range of applications along coastlines and can significantly reduce the workload of sentries. Despite the significance of the problem, the detection of unknown but concerned maritime objects remains under-explored.

While object detection models [1, 2] have achieved remarkable success in accurately identifying and localizing objects, most of these models are limited to detecting only known objects that belong to classes seen during training. However, unknown objects frequently emerge in real-world situations, leading to crucial problems in safety-critical applications (*e.g.*, autonomous driving, military defence) when relying on conventional object detection models. To address this issue, models capable of detecting unknown objects have been developed [3, 4].

Despite the potential of existing unknown object detection models, detecting unknown objects in maritime environments still remains a challenging task. Although object detection models used in maritime environments are often trained with datasets of known objects such as various type of ships, false alarms are frequently generated due to various unconcerned objects such as vehicles in the coastal land area. In maritime applications, such as autonomous unmanned surface vehicles (USVs) and maritime surveillance, the primary interest is to detect unknown but concerned objects floating on the sea area. However, existing object detection models cannot selectively detect unknown but concerned objects in the sea area, because of yielding many false positives detected on coastal land area. To mitigate the aforementioned difficulty, identifying the circumstances surrounding each unknown object can help prevent false alarms.

In this paper, we propose a real-time framework for unknown but concerned object detection in maritime environments, named MarUCOD, which integrates object detection, segmentation, and out-of-distribution (OOD) detection techniques. The object detection employs an off-the-shelf object detector trained across a vast spectrum of classes to act as a class-agnostic detector, and outputs many object boxes by setting low threshold. The segmentation identifies the circumstances surrounding the object boxes and filters out unconcerned objects detected in the coastal land. Finally, the OOD detection discriminates unknown objects among the remaining concerned objects. To maximize performance of proposed MarUCOD in maritime environments, we devise a new brightness pre-processing scheme and a novel contrastive separation loss for segmentation.

The main contributions are summarized as: (1) We propose a real-time framework for unknown but concerned object detection in maritime environments,

which integrates object detection, segmentation, and OOD detection techniques. (2) We apply bisecting K-means clustering to the unsupervised OOD detection to enhance the performance of proposed MarUCOD. In addition, we suggest brightness pre-processing schemes and a separation loss for optimizing MarUCOD to maritime environments. (3) We demonstrate the effectiveness of MarUCOD through comprehensive analyses on each component of MarUCOD. Despite the lightweight of each component, MarUCOD achieves state-of-the-art performance in unknown but concerned object detection in maritime environments.

2 Proposed Method

The overall scheme of the proposed MarUCOD is depicted in Fig. 1, which includes an object detection module, a segmentation module, and an OOD detection module. First, an input image is fed into both the object detection module and the segmentation module. The object detection module produces sufficient object boxes by using a low confidence threshold during detection. Concurrently, the segmentation module determines which pixels are located in the sea area. Based on the results from the segmentation module, the unconcerned objects are filtered to retain only the concerned objects located in the sea area. These concerned objects are fed into the OOD detection module that discriminates unknown objects from known objects belonging to in-distribution. The detected boxes on unknown but concerned objects are the final output of MarUCOD.

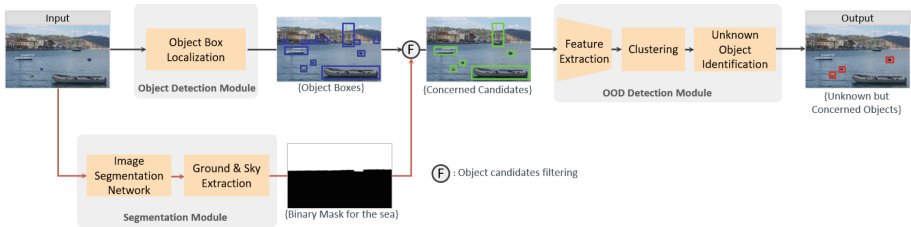


Fig. 1. Overall Scheme of MarUCOD.

2.1 Object Detection Module

When MarUCOD receives an input image, it aims to detect as many objects as possible, regardless of their classes. To achieve this class-agnostic object detection, we employ an object detection model with a low confidence score threshold, which has been trained on a wide range of classes. For the model, we utilize a YOLOv7 [1] model trained on the COCO datasets with 80 classes. This paper specifically employs YOLOv7, chosen for its fast and stable state-of-the-art performance, for real-time object localization. Using the YOLOv7 model, we produce object boxes by combining all detected objects across classes.

The YOLOv7 [5] model provides a confidence score for each predicted object and outputs boxes containing objects with scores greater than a threshold α . Since our objective is to identify objects regardless of classes, we intentionally reduce the confidence score threshold to $\beta < \alpha$. This adjustment makes the model less sensitive to classification and meets the goal of detecting object boxes regardless of their classes.

2.2 Segmentation Module

If the object detection module is used alone in maritime environments, false alarms (boxes of known or unconcerned objects) frequently occur, that is, lots of unconcerned objects in the coastal land or in the sky are detected. To tackle this issue, we adopt a segmentation module to remove the unconcerned object boxes obtained by the object detection module. The segmentation module comprises two stages: image segmentation network and ground and sky extraction.

Image Segmentation Network. As a backbone network for image segmentation, we use WODIS [6], which is for image segmentation in marine environments and requires light computation burden. WODIS has an encoder-decoder architecture and classifies each pixel of an input image into three categories: water, sky, and obstacle (others excluding water and sky). Here, the encoder is based on ResNet [7] convolutional layers and the decoder classifies each pixel from features extracted by the encoder. To boost the performance of the image segmentation, we introduce a brightness pre-processing scheme and a novel loss function.

Brightness Pre-processing. The dynamic lighting conditions of the maritime environment, ranging from sun-drenched horizons to murky twilight, pose a significant challenge for reliable image analysis. Furthermore, inconsistency in brightness can impede computer vision tasks of deep learning such as object detection or image segmentation. By harmonizing pixel brightness across diverse lighting scenarios, brightness pre-processing fosters consistent data representations that enhance performance on learning tasks. In this work, brightness directly corresponds to the V channel in HSV color space. That is, after converting an image from RGB to HSV, each value in V channel represents the brightness of pixels. Given a dataset, we propose two pre-processing schemes to mitigate inconsistency in brightness as follows.

Brightness shift adjusts the brightness mean of pixels in an input image to match the brightness mean of pixels in all images in the training dataset. It aims to learn more robust model against various brightness spectrum by making bright images darker and dark images brighter. The shifted brightness of each pixel is obtained by

$$\tilde{\mathbf{b}}_{ij}^{\mathbf{x}} = \min(\max(0, \mathbf{b}_{ij}^{\mathbf{x}} - (\mu_{\mathbf{b}^{\mathbf{x}}} - \mu_{\mathbf{b}^{\mathcal{D}}}), 255), \quad (1)$$

where $\mathbf{b}_{ij}^{\mathbf{x}}$, $\mu_{\mathbf{b}^{\mathbf{x}}}$ and $\mu_{\mathbf{b}^{\mathcal{D}}}$ denote the brightness of the i -th row and the j -th column pixel in an input image \mathbf{x} , the brightness mean of the input image, and the brightness mean of the dataset, respectively.

Brightness Normalization normalizes the brightness distribution of a brightness-shifted input image according to the brightness statistics of a dataset.

$$\bar{\mathbf{b}}_{ij}^{\mathbf{x}} = \frac{\mathbf{b}_{ij}^{\mathbf{x}} - \mu_{\mathbf{b}^{\mathcal{D}}}}{\sigma_{\mathbf{b}^{\mathcal{D}}}}, \quad (2)$$

where $\sigma_{\mathbf{b}^{\mathcal{D}}}$ is the brightness standard deviation of a dataset.

Contrastive Separation Loss Function. To enhance the robustness of the segmentation network to background diversity, we propose a contrastive separation loss function that creates an effective feature space for segmentation. Let \mathbf{v} be a vector obtained by flattening N_c -channel features from the third layer of the encoder and let R_1 and R_2 represent regions corresponding to two different classes from $\{\text{water, sky, obstacle}\}$. We define the values of features belonging to R_1 in the c -th channel as $\{\mathbf{v}_i^c\}_{i \in R_1}$. Additionally, we define the batch mean values of c -th channel features belonging to R_1 and R_2 as $\mu_{R_1}^c$ and $\mu_{R_2}^c$, respectively.

The proposed loss function aims to make the features of each component distinct from each other in the feature space. Specifically, the loss function makes the features corresponding to R_1 closer to their mean value, $\mu_{R_1}^c$, and at the same time farther away from the mean value, $\mu_{R_2}^c$, corresponding to R_2 . The contrastive separation loss function can be expressed by

$$\mathcal{L}_{R_1}^{R_2} = \frac{1}{N_c} \sum_{c=1}^{N_c} \frac{\sum_{i \in R_1} (\mathbf{v}_i^c - \mu_{R_1}^c)^2}{\sum_{i \in R_1} (\mathbf{v}_i^c - \mu_{R_2}^c)^2}. \quad (3)$$

Depending on the components assigned to R_1 and R_2 , different loss functions with distinct meanings can be formulated. For instance, a contrastive water-obstacle separation loss (CWOL) can be defined as $\mathcal{L}_{R_1}^{R_2} = \mathcal{L}_{R_{\text{water}}}^{R_{\text{obstacle}}}$. Unlike the water-obstacle separation loss (WSL) [8], our proposed loss function considers class information for each pixel by measuring class-wise distances in a feature space. Furthermore, while WSL uses pixels from two different classes as an input simultaneously, ours focuses on pixels belonging to a single class, resulting in more favorable features for segmentation.

The final loss function includes not only the proposed contrastive separation loss but also a focal loss [9] for emphasizing regions prone to misclassification during training. The final loss function can be expressed by

$$\mathcal{L} = \mathcal{L}_{\text{focal}} + \lambda \mathcal{L}_{R_1}^{R_2}. \quad (4)$$

Ground and Sky Identification. Utilizing class information for each pixel obtained during the segmentation step, we filter out unconcerned objects in land and sky. Pixels belonging to the obstacle class can be considered as unconcerned objects in the land. To identify pixels corresponding to land area, we apply the Connected Component Labeling (CCL) [10] algorithm only to obstacle pixels. This process enables the identification of connected pixels as a single component, as illustrated in Fig. 2(c). Considering that the ground generally lies

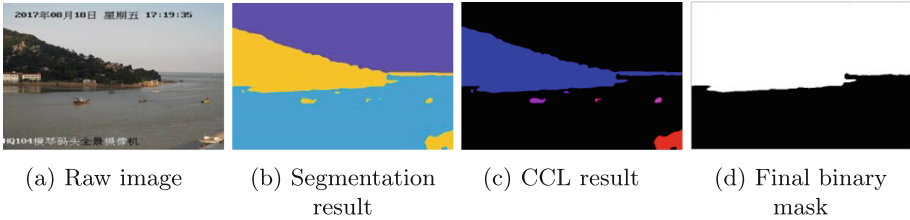


Fig. 2. Result images at each step of the ground & sky extraction algorithm

between the sky and the sea in surveillance camera compositions, class information of surrounding pixels is utilized for each component. If the number of pixels corresponding to the sky among the surrounding pixels of a component exceeds a certain threshold, it is classified as a land component. As shown in Fig. 2(d), a binary mask is created where the pixels belonging to the land and the sky are assigned a value of 1, while all other pixels are assigned a value of 0. Ultimately, among object boxes detected by the object detection module, those located in parts corresponding to the value of 1 in the mask are filtered out, leaving only concerned objects floating on the sea. These concerned object boxes are then fed into the OOD detection module to identify unknown objects.

2.3 OOD Detection Module

The OOD detection module categorizes the concerned object boxes into known and unknown ones. Only objects falling into the unknown object boxes become the final detection results. The OOD detection process consists of three steps: feature extraction, Bisecting K-Means clustering, and unknown object identification.

Feature Extraction. To determine unknown objects, we first extract feature vectors from each concerned object boxes. In this process, the concerned object boxes are cropped from the image, referred to as object patches. These object patches are then resized to a fixed patch size and inputted into a feature extractor. Using a feature extractor trained solely on ID data may lead to difficulties in extracting feature vectors for OOD objects which represent entirely different objects from ID data. Accordingly, we utilize an ImageNet [11] pre-trained ResNet50 [7] model as the feature extractor for the OOD detection module.

Bisecting K-Means Clustering. To distinguish unknown objects from known objects, we employ unsupervised clustering to form clusters of known objects within a feature space. During training, we use feature vectors of ID objects without class information to learn the distribution of ID objects. After learning ID objects, OOD objects are identified by measuring the distance between feature vectors of concerned object boxes and the cluster centroids in the feature space. For unsupervised clustering that largely affects the performance, We utilize bisecting K-Means clustering [12]. Bisecting K-Means is a hierarchical

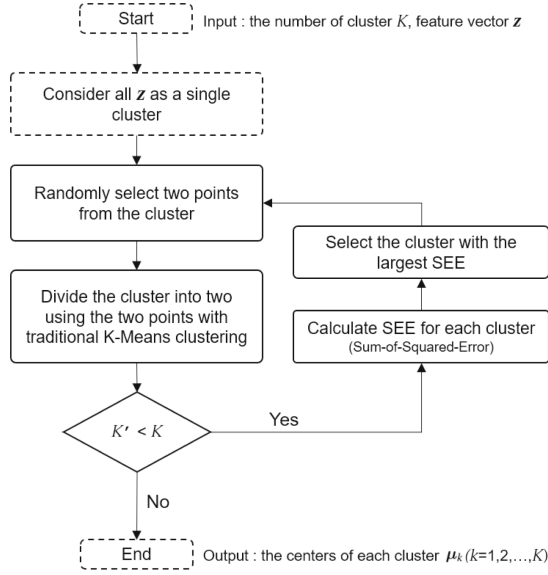


Fig. 3. A flow chart of Bisecting K-Means clustering

clustering algorithm, derived from K-Means clustering [13], which categorizes a given extracted feature vector \mathbf{z} into K clusters.

K-Means clustering is an iterative clustering algorithm that selects K centroids and assigns each data point to the nearest cluster until convergence. Bisecting K-Means clustering [12] performs the K-Means clustering in a hierarchical manner. Unlike traditional K-Means clustering that divides all \mathbf{z} into K clusters at ones, Bisecting K-Means repeatedly divides each cluster into two clusters until the total number of clusters reaches K . The structure of this algorithm is outlined in Fig. 3. First, Bisecting K-Means clustering receives as input the final number of cluster K and feature vectors z . Then, it treats all z as one cluster and divides it into two clusters using standard K-Means clustering. Among the K' clusters formed up to that point, the cluster with the highest SEE(Sum-of-Squared-Error) is selected. Subsequently, the standard K-Means clustering is repeated until the number of clusters reaches K .

Through Bisecting K-Means clustering, centroids $\boldsymbol{\mu}_k (k = 1 \cdots K)$ of K clusters representing training data are identified and stored. These $\boldsymbol{\mu}_k$ values symbolize the distribution of the training data (*i.e.*, ID data).

Unknown Object Identification. After obtaining the centroids $\boldsymbol{\mu}_k$ of each cluster, the OOD score s of \mathbf{z}^{test} is computed as

$$s = \min_{k=1 \dots K} \left(1 - \frac{\boldsymbol{\mu}_k \cdot \mathbf{z}^{test}}{\|\boldsymbol{\mu}_k\| \|\mathbf{z}^{test}\|} \right), \quad (5)$$

where \mathbf{z}^{test} is an extracted feature vector of an object box and s represents the minimum cosine distance between the centroids μ_k of K clusters and \mathbf{z}^{test} . Thus, s signifies the distance between the distribution of training data and \mathbf{z}^{test} , with higher values indicating greater differences between the training data and \mathbf{z}^{test} .

During training, OOD scores, denoted by \mathbf{s}^t for t th training sample, are used to determine the OOD score threshold θ to be utilized during detecting OOD objects. \mathbf{s}^t is sorted in ascending order, and θ is defined by the lowest OOD score among those greater than $p\%$ scores in the ascending order. The threshold θ is determined by selecting p that maximizes the overall OOD detection accuracy.

During detecting OOD objects, when an object feature vector \mathbf{z}^{test} is received, if the OOD score of \mathbf{z}^{test} is greater than the OOD score threshold θ , it is considered OOD; otherwise, it is considered ID. Accordingly, only objects classified as OOD among feature vectors of object boxes are categorized as unknown but concerned objects finally.

3 Experiments

3.1 Out-of-Distribution Detection Module

Datasets. The OOD detection module is trained using the set of known and concerned objects seen at the sea area, *i.e.*, SeaShips dataset [14]. SeaShips is an image dataset containing six types of ships, and we train the OOD detection module using images of ships from this dataset. Therefore, in MarUCOD, ships are considered as ID data (known and concerned objects), and objects in the sea area other than ships are considered as unknown but concerned objects. The performance is evaluated using MID [15] and MODD [16] datasets. The MID and MODD datasets consist of 7 and 12 videos of maritime environments, respectively. Videos without unknown objects are excluded, resulting in the use of 5 and 8 videos for evaluation, respectively.

Object Detection Model. To detect unknown but concerned objects, it is essential to first locate as many objects as possible regardless of their class and thereafter classify them as unknown but concerned objects. To swiftly detect various objects, we utilize YOLOv7, which demonstrates state-of-the-art performance in real-time object detection, trained on the COCO dataset containing 80 diverse classes. However, if we have used this model as is, objects outside of the trained 80 classes have not been detected. Therefore, to mitigate the influence of classes in our work, we drastically reduce the confidence score threshold of the model from the default threshold 0.25 (α) to 0.05 (β).

Clustering Method Selection. To select the clustering method for the detection of unknown objects, we conduct experiments comparing the performance depending on clustering method. The experiments utilize a total of six clustering methods: K-Means [13], Gaussian Mixture [17], Mean Shift [18], Affinity Propagation [19], HDBSCAN [20], and Bisecting K-Means [12]. During clustering the features of training data, we employ the clustering scheme and code from

Table 1. Performance of OOD detection depending on clustering methods.

Clustering Method	F1-score (%)	Recall (%)	Precision (%)	FPS
KMeans [13]	48.25	54.16	58.60	31.95
GaussianMixture [17]	44.88	54.39	50.87	33.31
MeanShift [18]	44.34	54.67	49.71	32.43
AffinityPropagation [19]	44.09	52.25	54.97	23.72
HDBSCAN [20]	44.47	54.11	52.86	28.78
BisectingKMeans [12]	50.57	58.11	57.75	32.23

Table 2. Performance of OOD detection depending on score function.

Score Function	F1-score (%)	Recall (%)	Precision (%)	FPS
Euclidean [6]	50.57	58.11	57.75	32.23
Cosine [7]	60.82	65.53	64.03	32.74
Mahalanobis [8]	54.23	72.96	50.02	19.85

Scikit-Learn [21], utilizing the default settings for each clustering method. Consequently, the number of clusters (K) is set to 8 for K-Means and Bisecting K-Means, the number of mixture components for Gaussian Mixture is set to 1, the bandwidth for Mean Shift is set to 96, the dumping factor for Affinity Propagation is set to 0.5, and the minimum distance between neighbors (EPS) for HDBSCAN is set to 0.5. The performance of each clustering method is measured, and the results are presented in Table 1. Table 1 shows that the Bisecting K-Means clustering method achieves the highest F1 score. Consequently, we choose Bisecting K-Means clustering method for the detection of unknown objects.

Score Function Selection. The performance can vary depending on the method used to measure OOD scores. Therefore, we compare performance of 3 distance measurement formulas: Euclidean Distance (Eq. 6), Cosine Distance (Eq. 7), and Mahalanobis Distance (Eq. 8). In these equations, $\boldsymbol{\mu}_k$ ($k = 1 \dots K$) represents the cluster-wise centroid vectors, Σ denotes the covariance, n dimensional feature vector is denoted by $\mathbf{z}^{test} (= [\mathbf{z}_1^{test}, \mathbf{z}_2^{test}, \dots, \mathbf{z}_n^{test}]^\top)$, and s is the OOD score for \mathbf{z}^{test} .

$$s = \min_{k=1 \dots K} \sqrt{\sum_{i=1}^n (\boldsymbol{\mu}_{k_i} - \mathbf{z}_i^{test})^2}, \quad (6)$$

$$s = \min_{k=1 \dots K} \left(1 - \frac{\boldsymbol{\mu}_k \cdot \mathbf{z}^{test}}{\|\boldsymbol{\mu}_k\| \|\mathbf{z}^{test}\|}\right), \quad (7)$$

$$s = \min_{k=1 \dots K} \sqrt{(\mathbf{z}^{test} - \boldsymbol{\mu}_k) \Sigma^{-1} (\boldsymbol{\mu}_k - \mathbf{z}^{test})^\top}. \quad (8)$$

The performance for each score function is presented in Table 2. Table 2 demonstrates that the Cosine Distance achieves the highest F1 score and also

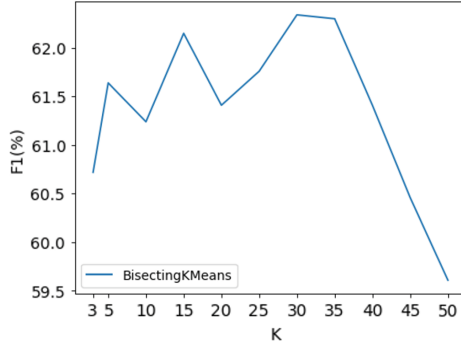


Fig. 4. Performance on OOD according to the number of cluster (K), in terms of F1 score.

Table 3. Performance on OOD detection depending on patch size.

Patch Size	F1-score (%)	Recall (%)	Precision (%)	FPS
32	42.42	74.79	33.28	62.39
64	46.17	64.10	42.56	50.29
128	62.34	68.16	63.54	32.37
256	43.46	49.99	46.54	13.38
512	40.17	85.68	29.60	4.10

the fastest FPS. Therefore, we choose the Cosine Distance (Eq. 7) as the final formula for calculating the OOD score.

Performance Depending on the Number of Cluster K . Bisecting K-Means clustering [12] is an algorithm that iteratively divides each cluster into two clusters until the total number of clusters reaches K . Therefore, the performance can vary significantly depending on the chosen value of K . We experiment with different values of K , specifically 3, 5, 10, 15, 20, 25, 30, 35, 40, 45, and 50, measuring the performance for each. The results are presented in Fig. 4. Looking at the Fig. 4, it is observed that the highest F1 scores are achieved when the number of clusters K is 30. Therefore, we have selected $K = 30$ as the final number of clusters.

Object Patch Size Selection. When classifying objects using clusters, we resize various size of an object bounding box into a fixed patch size to apply it the classifier. The performance and speed of the classifier can vary depending on the size of the patch size. As shown in Table 3, the highest F1 scores are achieved when the object patch size is 128.

3.2 Segmentation Module

Datasets. The segmentation network in the segmentation module is trained and evaluated using the LaRS dataset [22]. The LaRS dataset consists of a total of 4,006 images, but only 2,803 images with segmentation labels are utilized for training and evaluation. Additionally, LaRS consists of datasets from various sources, among which 1,323 images corresponding to MaStr1325 [23] are used for training, while the remaining 1,480 images are employed for evaluation, ensuring that images from the same source are not used for both training and evaluation.

Table 4. Comparison of preprocessing cases.

Rank	Train Preprocess	Test Preprocess	Acc. (%)	Obst. Iou (%)	Water Iou (%)	Sky Iou (%)	Mean Iou (%)
1	$ImageNet_N + Br_N$	$ImageNet_N + Br_N$	98.74	90.63	97.77	98.95	95.79
2	$ImageNet_N + Br_\mu$	$ImageNet_N + Br_\mu$	98.67	90.9	97.85	98.52	95.76
3	$ImageNet_N + Br_\mu$	$ImageNet_N$	98.64	90.8	97.81	98.43	95.68
4	Br_N	Br_N	98.36	90.94	96.88	98.07	95.29
5	$ImageNet_N$	$ImageNet_N$	98.19	91.04	96.69	97.51	95.08
6	$ImageNet_N$	$ImageNet_N + Br_\mu$	98.04	90.38	96.42	97.31	94.70

Brightness Pre-processing. This section shows the results of different preprocessing cases when learning WODIS. In training and testing, the performance is compared with different preprocess combinations. The preprocessing consists of a total of 6 cases as below:

- No preprocessing is applied: No Normalization and no brightness shift.
- ImageNet normalization which is commonly used as preprocessing: $ImageNet_N$.
- Brightness Normalization: Br_N .
- Brightness Shift: Br_μ .
- Both ImageNet Normalization and Brightness Normalization are applied: $ImageNet_N + Br_N$.
- Both ImageNet Normalization and Brightness Shift are applied: $ImageNet_N + Br_\mu$.

After conducting 36 experiments with all possible combinations, only top 6 results are reported in Table 4 including existing ImageNet normalization as baseline. As can be seen in Table 4, four combinations show improved performance in terms of Accuracy and Mean IoU compared to the ImageNet normalization. It shows Obstacle IoU, which is slightly lower than ImageNet Normalization, on the other hand, increases by 1%+ in both Water and Sky IoU. The combination of ImageNet Normalization with Brightness Shift leads to improved performance. Nevertheless, the best result is achieved by fusing Brightness Normalization and ImageNet Normalization.

Contrastive Separation Loss Function. As discussed in the Sect. 2.2, the proposed contrastive separation loss function can be applied to diverse compositions of the loss function based on the selection of components for R_1 and R_2 . Among these, we compare the performance of Contrastive Sky-Obstacle separation Loss (CSOL; $\mathcal{L}_{R_{\text{sky}}}^{R_{\text{obstacle}}}$), Contrastive Sky-Water separation Loss (CSWL; $\mathcal{L}_{R_{\text{sky}}}^{R_{\text{water}}}$), Contrastive Obstacle-Sky separation Loss (COSL; $\mathcal{L}_{R_{\text{obstacle}}}^{R_{\text{sky}}}$), Contrastive Obstacle-Water separation Loss (COWL; $\mathcal{L}_{R_{\text{obstacle}}}^{R_{\text{water}}}$), Contrastive Water-Sky separation Loss (CWSL; $\mathcal{L}_{R_{\text{water}}}^{R_{\text{sky}}}$), and Contrastive Water-Obstacle separation Loss (CWOL; $\mathcal{L}_{R_{\text{water}}}^{R_{\text{obstacle}}}$).

Table 5 presents the results of segmentation performance based on different loss functions. We compare the performance of our loss functions with WSL that shows the highest performance among the loss functions proposed in WaSR [8]. The comparison includes accuracy, Intersection over Union (IoU) for water, IoU for obstacles, IoU for the sky, and the average of these three IoU values. Notably, among our loss functions, CSOL and CSWL outperform WSL in both accuracy and IoU, indicating a significant improvement in segmentation performance.

Table 5. Performance on image segmentation with different loss functions.

Loss Function	Accuracy (%)	Obstacle IoU (%)	Water IoU (%)	Sky IoU (%)	Mean IoU (%)
WSL [8]	89.29	74.22	81.31	88.91	81.48
CSOL	90.62	77.25	83.60	89.30	83.38
CSWL	90.06	73.91	83.44	90.46	82.60
COSL	89.93	76.60	82.59	87.35	82.18
COWL	88.75	74.43	80.21	86.98	80.54
CWSL	88.74	75.06	80.66	84.80	80.18
CWOL	87.87	72.65	80.05	83.30	78.67

Table 6. Performance per segmentation filtering.

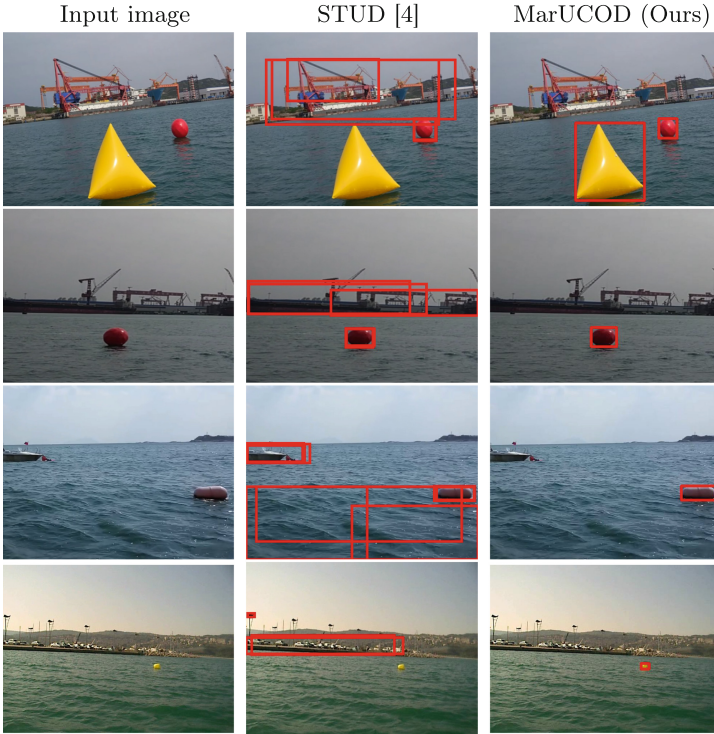
	F1-score (%)	Recall (%)	Precision (%)	FPS
without Seg Filter	62.34	68.16	63.54	32.37
with Seg Filter	67.72	65.03	76.82	15.52

3.3 Performance of MarUCOD

In the main stream of MarUCOD, the object detection module estimates the locations of all unspecified objects and the OOD detection module discriminates them into known or unknown objects. The final performance of the two detection modules corresponds to “without Seg Filter” in Table 6. This includes setting the

Table 7. Comparison with a state-of-the-art method for unknown but concerned object detection.

	F1-score (%)	Recall (%)	Precision (%)	FPS
STUD [4]	25.63	48.67	19.99	14.28
MarUCOD(Ours)	67.72	65.03	76.82	15.52

**Fig. 5.** Qualitative comparison with a state-of-the-art method.

confidence score threshold β for YOLOv7 to 0.05, the object patch size to 128, employing Bisecting K-Means with a cluster count K of 30, and setting the OOD threshold θ determined by $p = 95\%$. When the segmentation module is used for filtering unconcerned object boxes, all the performance metrics largely increase as shown in the case of “with Seg Filter” in Table 6.

The comparison between the performance of existing research and our MarUCOD is presented in Table 7. We have investigated various out-of-distribution detection techniques [24–26]. However, among with STUD [4], which focuses on object-level OOD detection, these methods primarily address image-level OOD detection. Since STUD is the only state-of-the-art (SOTA) in our problem setting, we perform a performance comparison with this paper. In our study, we

adapt this approach to the maritime domain by training on the SeaShips dataset and evaluating its performance on the MID and MODD datasets.

As shown in Table 7, significant improvements are observed in F1 score, which includes precision and recall. The F1 score increases from 25.63% to 67.72%, recall from 48.67% to 65.03%, and precision from 63.54% to 76.82%. Additionally, the final speed increases slightly to 15.52 fps compared to STUD's 14.28 fps. Given that the conventional target frame rate for real-time processing in surveillance footage is 15 fps [27], MarUCOD's speed ensures real-time performance. This performance enhancement can also be observed in Fig. 5.

We develop a user interface (UI) using the PyQT5 library, enabling users to easily train MarUCOD and tune hyperparameters depending on their maritime environments. The codes and the UI for MarUCOD is available at <https://github.com/AIX-Coast-Defense-PIL/MarUCOD>.

4 Conclusion

We proposed a novel real-time framework for unknown but concerned object detection (MarUCOD) in maritime scenes. By integrating object detection, segmentation, and OOD detection techniques, along with optimizing the performance of each technique to meet maritime environments through extensive experiments, MarUCOD achieves state-of-the-art performance in unknown but concerned object detection in maritime settings. We believe that our framework will be applied to a variety of maritime applications, including unmanned surface vehicles and maritime surveillance. However, in extremely dark environments, all pixels of a RGB input image may have a value of 0, leading to performance degradation. Therefore, exploring unknown object detection using alternative channels (*e.g.*, infrared) is left for future research.

References

1. 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)
2. 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)
3. Joseph, K., Khan, S., Khan, F.S., Balasubramanian, V.N.: Towards open world object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5830–5840 (2021)
4. Du, X., Wang, X., Gozum, G., Li, Y.: Unknown-aware object detection: learning what you don't know from videos in the wild. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 13678–13688 (2022)
5. 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)

6. Chen, X., Liu, Y., Achuthan, K.: WODIS: water obstacle detection network based on image segmentation for autonomous surface vehicles in maritime environments. *IEEE Trans. Instrum. Meas.* **70**, 1–13 (2021)
7. 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)
8. Bovcon, B., Kristan, M.: WaSR-a water segmentation and refinement maritime obstacle detection network. *IEEE Trans. Cybern.* **52**(12), 12661–12674 (2021)
9. 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)
10. Bolelli, F., Allegretti, S., Baraldi, L., Grana, C.: Spaghetti labeling: directed acyclic graphs for block-based connected components labeling. *IEEE Trans. Image Process.* **29**, 1999–2012 (2019)
11. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: *Advances in Neural Information Processing Systems* 25 (2012)
12. Di, J., Gou, X.: Bisecting k-means algorithm based on k-valued selfdetermining and clustering center optimization. *J. Comput.* **13**(6), 588–595 (2018)
13. Arthur, D., Vassilvitskii, S., et al.: k-means++: the advantages of careful seeding. In: *SODA*, vol. 7, pp. 1027–1035 (2007)
14. Shao, Z., Wu, W., Wang, Z., Du, W., Li, C.: SeaShips: a large-scale precisely annotated dataset for ship detection. *IEEE Trans. Multimedia* **20**(10), 2593–2604 (2018)
15. Liu, J., Li, H., Luo, J., Xie, S., Sun, Y.: Efficient obstacle detection based on prior estimation network and spatially constrained mixture model for unmanned surface vehicles. *J. Field Robot.* **38**(2), 212–228 (2021)
16. Kovačič, S., Kristan, M., Kenk, V.S., Perš, J.: Fast image-based obstacle detection from unmanned surface vehicles. *IEEE Trans. Cybern.* (2015)
17. Rasmussen, C.: The infinite Gaussian mixture model. In: *Advances in Neural Information Processing Systems* 12 (1999)
18. Comaniciu, D., Meer, P.: Mean shift: a robust approach toward feature space analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* **24**(5), 603–619 (2002)
19. Frey, B.J., Dueck, D.: Clustering by passing messages between data points. *Science* **315**(5814), 972–976 (2007)
20. Campello, R.J., Moulavi, D., Sander, J.: Density-based clustering based on hierarchical density estimates. In: *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 160–172. Springer (2013)
21. Pedregosa, F., et al.: Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011)
22. vZust, L., Perš, J., Kristan, M.: LaRS: a diverse panoptic maritime obstacle detection dataset and benchmark. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 20304–20314 (2023)
23. Bovcon, B., Muhovič, J., Perš, J., Kristan, M.: The MaSTr1325 dataset for training deep USV obstacle detection models. In: *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE (2019)
24. Liu, W., Wang, X., Owens, J., Li, Y.: Energy-based out-of-distribution detection. In: *Advances in Neural Information Processing Systems* 33, pp. 21464–21475 (2020)
25. Papadopoulos, A.-A., Rajati, M.R., Shaikh, N., Wang, J.: Outlier exposure with confidence control for out-of-distribution detection. *Neurocomputing* **441**, 138–150 (2021)

26. Sun, Y., Ming, Y., Zhu, X., Li, Y.: Out-of-distribution detection with deep nearest neighbors. In: International Conference on Machine Learning, pp. 20827–20840. PMLR (2022)
27. Cao, D., et al.: MicroEdge: a multi-tenant edge cluster system architecture for scalable camera processing. In: Proceedings of the 23rd ACM/IFIP International Middleware Conference, pp. 322–334 (2022)



Identifying Impurities in Liquids of Pharmaceutical Vials

Gabriele Rosati¹, Kevin Marchesini¹, Luca Lumetti¹, Federica Sartori²,
Beatrice Balboni², Filippo Begarani², Luca Vescovi², Federico Bolelli¹ (✉),
and Costantino Grana¹

¹ Università degli Studi di Modena e Reggio Emilia, Modena, Italy
{gabriele.rosati,kevin.marchesini,luca.lumetti,federico.bolelli,
costantino.grana}@unimore.it

² PBL S.r.l., Parma, Italy
{federica.sartori,beatrice.balboni,filippo.begarani,
luca.vescovi}@pblsrl.it

Abstract. The presence of visible particles in pharmaceutical products is a critical quality issue that demands strict monitoring. Recently, Convolutional Neural Networks (CNNs) have been widely used in industrial settings to detect defects, but there remains a gap in the literature concerning the detection of particles floating in liquid substances, mainly due to the lack of publicly available datasets. In this study, we focus on the detection of foreign particles in pharmaceutical liquid vials, leveraging two state-of-the-art deep-learning approaches adapted to our specific multiclass problem. The first methodology employs a standard ResNet-18 architecture, while the second exploits a Multi-Instance Learning (MIL) technique to efficiently deal with multiple images (sequences) of the same sample. To address the issue of no data availability, we devised and partially released an annotated dataset consisting of sequences containing 19 images for each sample, captured from rotating vials, both with and without impurities. The dataset comprises 2,426 sequences for a total of 46,094 images labeled at the sequence level and including five distinct classes. The proposed methodologies, trained on this new extensive dataset, represent advancements in the field, offering promising strategies to improve the safety and quality control of pharmaceutical products and setting a benchmark for future comparisons.

Keywords: Vial Liquid inspection · Multi-Instance Learning · Convolutional Neural Network · Classification · Prediction

1 Introduction

Control over visible particles represents an important aspect in various fields, such as pharmaceuticals, food and beverages, and manufacturing, because they have a significant effect on the quality of the products. Impurities found in food can have different forms: physical, chemical, and biological contaminants, like

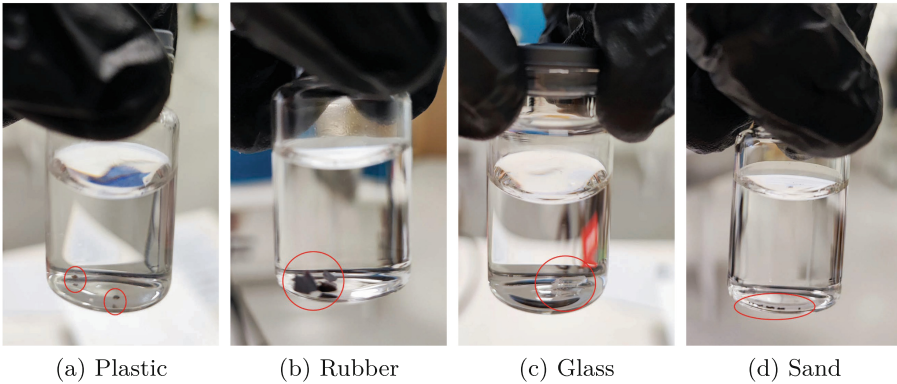


Fig. 1. An example of some impurities (circled in red) that can occur in a liquid vial. (a) shows brown plastic particles, in (b) black rubber particles are present, (c) illustrates the presence of a piece of glass, and (d) shows residual sand at the bottom of the vial.

small metal fragments or pesticides [30]. These impurities pose significant health risks to humans, potentially leading to severe illnesses and affecting the quality and taste of food. In manufacturing processes, impurities can alter desired product characteristics and performance. For example, metallic contaminants in semiconductor manufacturing can alter product mechanical properties, leading to the development of weak points, thus impairing the functionality of final products [9]. Our work is focused on foreign particles in pharmaceutical products. These impurities can lead to various consequences, including reduced effectiveness due to interference with active ingredients, safety risks from toxic substances or allergens, and regulatory issues resulting in product recalls or legal penalties [3]. These particles can arise from injection of the bottles, packaging, collisions, or filtration, and they can pose serious health risks when injected into the bloodstream, potentially resulting in thrombosis, phlebitis, tumors, and anaphylactic reactions [15]. The detection of these particles is particularly challenging because they can occur in various forms, such as dust, plastic, rubber and silicone particles, glass fragments, and sand residues as illustrated in Fig. 1.

Traditionally, identifying particles and impurities relied on manual inspection, which has been proven to be inefficient due to its time-consuming nature, subjectivity, low repeatability, and susceptibility to errors. Several factors influence the likelihood of visually detecting particles, such as the particle's size, composition, and shape, as well as the product formulation, the vials, the filled volume, and inspection conditions [29]. In manual detection, typically, inspectors position the injection bottle under a high brightness and planar light source, then rotate and tilt the container manually (or with the assistance of machinery) to observe any visible foreign substances inside. Based on their inspection experience, they decide whether these substances are acceptable or not. Such an approach often exhibits poor efficiency since it strongly depends on light conditions and other external factors, and it is not exactly repeatable [10].

Advancements in imaging technology and computer vision have led to the establishment of automated particle detection systems, which are increasingly reliable, removing human error. These systems typically leverage image processing [21], machine learning techniques [42], and well-known vision algorithms [18]. The main challenge to overcome is to find a method capable of efficiently extracting fine-grained features from images that may be captured under suboptimal lighting conditions and contain various sources of interference or noise. In summary, the issues to be tackled when designing particle detection methods in liquid vials are as follows:

1. The **appearance of particles** can be influenced by different lighting conditions, leading to variations in color, especially if some particles are transparent, as is often the case with glass fragments. Images of pharmaceutical containers are taken using a camera positioned beneath a mobile tracking device, operating synchronously, and changes in illumination within the image may occur due to ambient light conditions and vibrations from the machine;
2. Particles come in a **variety of forms**, ranging from small spots to bigger shapes. They can exhibit different textures and surface characteristics, from smooth to rough or irregular. The diversity in particle properties poses a significant challenge for detection and classification systems, requiring robust algorithms capable of effectively distinguishing between different particle types under various circumstances;
3. The presence of **noisy elements** on the bottle wall and bubbles [40] within the liquid can pose challenges in classifying the foreign particles, as they share similar visual characteristics.

In recent years, Convolutional Neural Networks (CNNs) have been extensively used for multiple applications [4, 7, 33, 34, 38, 39], including industrial defect detection [8, 11, 20, 41]. They have shown promising results in overcoming the aforementioned issues. CNN models can perform various tasks thanks to their strong capability to represent robust features. While recent literature focused on developing tools for detecting particles in liquids using deep learning methods, there is an absence of publicly available datasets for this task, mainly due to the preservation of industrial secrets. For this reason, previous research in this area relied only on private datasets, making the comparison with existing approaches impractical.

Paper Contributions. To partially cope with this literature gap, this paper releases a small set of images that can be employed for future comparison.¹ Unfortunately, for the same aforementioned reasons, the entire training set cannot be released.

¹ Test data are available at <https://ditto.ing.unimore.it/residual>.

More specifically, this paper tackles the problem of identifying different kinds of impurities in pharmaceutical vial liquid by smartly leveraging two existing state-of-the-art deep learning approaches, namely ResNet-18 [16] and DSMIL [22]. To cope with the previously identified issues 1 and 2, instead of dealing with a single image per sample, we opted for acquiring sequences of 20 images for each vial, suitably subjected to machinery-supported rotation. Such an approach, which is also feasible in modern inline injection machines, allows for mitigating particle appearance issues and the presence of noisy elements. However, it introduces additional challenges in the automatic detection algorithms. In order to achieve satisfactory performance without sacrificing computation time, our approach advocates for ResNet-18 by directly feeding it with multiple channels, each corresponding to a sequence frame.

Additionally, to achieve similar results, although tackling it in a different way, a Multi-Instance Learning Approach (MIL) is employed by treating each sequence as a *bag* composed of multiple images *instances*. This way, the model can deal with moving objects in the sequence without requiring expensive tracking strategies as previously proposed in literature [48].

In both cases, our proposed pipeline achieves outstanding results without requiring pixel-level annotations.

2 Related Works

Product quality is crucial for pharmaceutical products, given their impact on people's health. To ensure this quality, various works have been made on vial inspection, with the goal of detecting defects such as tilting and sinking of the cap or cracks in the glass, which may negatively affect the product quality [44]. Although this is a slightly different task with respect to the detection of liquid defects, our approach follows similar steps and employs comparable techniques to those used in these studies.

The first works in this field employed traditional computer vision techniques. Liuet *al.* [25] have proposed an inspection method that used the watershed transform to find defective areas and a fuzzy SVM ensemble combined with an ensemble of genetic algorithms to classify the type of imperfection. Also, Liuet *al.* [27] used the SVM classifier to inspect vials for flaws, fed with local binary pattern (LBP) features extracted from the region of interest of the image, grouped using k-means clustering to have a compact representation of them. Several other studies have utilized SVM for classifying defects on the surface of the rolled steel [19], in the industrial pavements [28], and in textile materials [1]. The key difference in existing approaches lies in the method used for feature extraction. More recently, Zhouet *al.* [49] proposed two different techniques to find defects in glass bottles using traditional vision algorithms: a template-matching-based method with multiscale filtering, and a region-growing Euclidean saliency method, with the integration of superpixel segmentation and geodesic saliency detection algorithms.

Regarding the analysis of liquid solutions, Wang *et al.* [45] developed a method to find unwanted glass fragments in the liquid by shaking the container, exploiting the fact that the glass pieces are heavier, so they cannot move smoothly with liquid and other particles. Thus, they took several images in sequence and used the optical flow algorithm to perform the detection. In the same year, Ge *et al.* [12] presented an automated system for checking ampoule injections for tiny foreign particles. They developed a custom hardware platform for transportation and agitation, capturing images for analysis. The computation of trajectories of moving objects within liquid allowed them to differentiate foreign particles in the images; then impurity types were classified through multiple features, including particle area, mean gray value, and geometric invariant moments.

The advent of deep learning has been a breakthrough in visual detection tasks, including defect detection [41]. Its ability to autonomously learn complex features from datasets enabled algorithms to accurately identify patterns and objects with more precision. One of the first approaches regarding foreign particle inspection is another work of Ge *et al.* [11]. They successfully explored the usage of a modified version of Pulse-Coupled Neural Networks (PCNN) [20] to identify undesired particles in glucose or sodium chloride injection liquids. PCNNs are non-trained neural networks where each neuron receives as input the corresponding pixel intensity and other inputs from its neighboring neurons. These stimuli are added together, accumulating them until they surpass a dynamic threshold, triggering a pulse output. This process, iteratively performed, generates a series of binary images as outputs. Neighboring neurons' connections lead to pixels of the image with similar intensity values pulsing together. Thus, it is possible to obtain image segmentation by identifying pixels corresponding to synchronously pulsing neurons. The main drawback of this technique lies in its dependence on the choice of thresholds. The author of the paper suggested an adaptive approach to find the best hyperparameters.

Since the middle of the 2010s, many neural network architectures have been developed for detection tasks, such as R-CNN [14], Faster R-CNN [36], YOLO [35], SSD [26], and ResNet [16] and have become widely popular. Examples of application of these networks can be found in defect detection addressing various domains, such as the inspection of flat surfaces [46] using a combination of Fast and Faster R-CNN, the particle detection in complex biomedical images [13] through a ResNet-based architecture and the detection of cracks in aircraft structures through the usage of YOLOv3-Lite [23]. In the work of Ding *et al.* [8], a defect-detecting Single-Shot Detector (SSD) is devised for wood inspection, using DenseNet [17] as the backbone to improve the extraction of deep features and mitigate gradient vanishing issues of the original SSD backbone. Furthermore, the integration of a feature fusion function to combine multi-layer feature maps from the backbone enhances the classification of wood defects. Ritter *et al.* [37] presented a new method to identify and track fluorescent particles in microscopy images. Their approach leveraged the Deconvolution Network [31], a CNN similar to an encoder-decoder architecture for particle detection, along with a bidirectional long short-term memory for tracking, which also aided in

particle classification. A less conventional deep learning approach was used by Zhang *et al.* [48], who developed a particle inspection system for liquid vials. They captured eight sequential images and used fuzzy cellular neural networks for precise position and segmentation, introducing an adaptive tracking system based on a sparse model for determining the presence of foreign particles. In one of the most recent works, Yi *et al.* [47] explored the usage of the attention mechanism on pharmaceutical foreign particle detection. They developed an end-to-end deep architecture with adaptive convolution and multiscale attention to identify and classify foreign particles.

Based on the results reported in the aforementioned papers, we can state that deep learning detection methods outperform traditional approaches in particle detection liquids. For these reasons, in this work, we choose to employ two state-of-the-art deep learning architectures: ResNet [16], which we employed in a new fashion to handle sequences rather than individual images, and DSMIL [22], a method not previously investigated for multiclass particle detection. The specific details of our architectures and the results on our dataset are outlined in the following sections.

3 Methods

As said, to face the task of recognizing defective vials, we decided to explore two different paths; the former is based on the use of ResNet [16] in a slightly different way than the standard one, in order to deal with the entire sequence of images, the latter is a Multi-Instance Learning (MIL) [5] based technique.

3.1 ResNet-18

Residual Neural Network (also known as ResNet [16]) is a family of deep learning models in which the weights layers learn residual functions based on the layer inputs. This is possible through the residual connections that execute identity mappings and are added to layer outputs. In our study we employed ResNet-18.

As ResNet operates on individual images, we had to adapt its architecture to our problem, where we deal with a sequence of images for each rotating vial. Our goal was to capture the collective information across the sequence of frames acquired during the vial's rotation. To perform classification at the sequence level, we explored two different aggregating methods. In the first approach, we learned to predict a class for each frame within the sequence and subsequently determined the class for the entire sequence through a majority voting approach. Secondly, we investigated an alternative approach wherein we independently extracted features from each frame using ResNet convolutional layers. Then, these features were concatenated along a new dimension before being fed to the fully connected layers, resulting in a single prediction for the entire sequence. As a loss function, we used cross-entropy.

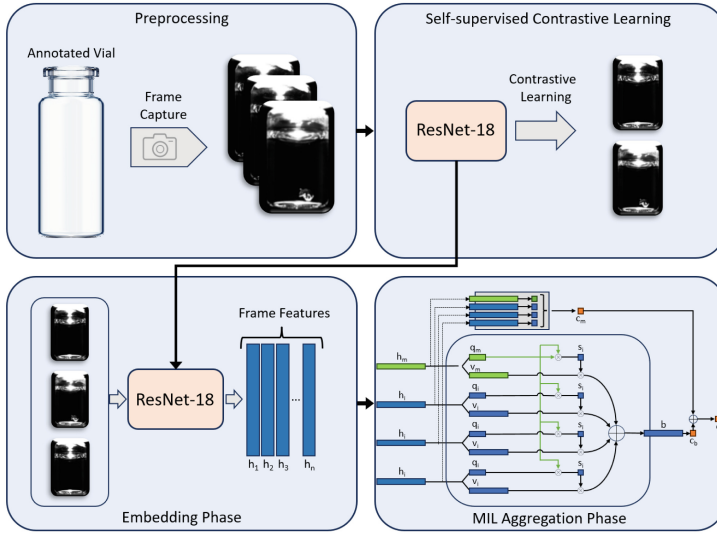


Fig. 2. Representation of the proposed MIL-based pipeline, divided into four main steps: preprocessing, self-supervised training of ResNet-18, Embedding phase, and the MIL phase.

3.2 MIL-Based Approach

Multiple instance learning is an extensively used weakly supervised learning algorithm [2, 24, 32] where a subset of examples from the training set is arranged as a set (bag) composed of multiple instances. If we deepen the case of binary classification, let $B = \{(x_1, y_1), \dots, (x_n, y_n)\}$ be a bag where $x_i \in X$ are instances with labels $y_i \in \{0, 1\}$, the label of B is given by:

$$c(B) = \begin{cases} 1, & \text{if } \exists y_i \in B : y_i = 1 \\ 0, & \text{otherwise} \end{cases}$$

The challenge of detecting defects in the liquid inside vials can be seen as a multiple-instance learning problem if the detection method involves capturing a series of images of the rotating vials and the labeling is done based on the entire sequence. The sequence-level labeling method is usually the standard because while foreign particles may not always be visible in every frame if they appear at least in one frame, the vial should be classified as defective.

The problem of multi-instance learning for a bag-level classification can be approached by training a model that assigns a probability $c(X)$ of the bag being labeled as positive ($Y = 1$). The function $c(X)$, can be formulated as follows:

$$c(X) = g(\sigma(f(x_1), \dots, f(x_n)))$$

where the function f is a feature extractor transforming single instances into a lower-dimensional embedding; σ is a permutation-invariant aggregation function

(often referred to as MIL pooling), which derives the bag representation; and g apply a final transformation to obtain the bag probability. Both functions f and g can be parameterized by neural networks, which can be trained end-to-end through backpropagation. The only other requirement is that the MIL pooling operation σ must be differentiable.

In our case, each image sequence is considered a bag, while each single frame composing the sequence is treated as an instance. We used the MIL architecture developed by Liet *al.* [22] called Dual-Stream Multiple Instance Learning (DSMIL). This network, depicted in Fig. 2, learns from both instances and bag embeddings at the same time. The first stream works at instance-level. It extracts an embedding from each instance and classifies each embedding, giving a single score in case of a binary classification problem. Then, the classification step is followed by a max-pooling operation to identify the instance with the highest score, referred to as the *critical instance*.

In a more exhaustive way, let $X = x_1, \dots, x_n$ denote a sequence (bag) of frames of a rotating vial. Given f as feature extractor, each frame x_i can be projected into an embedding $h_i = f(x_i) \in \mathbb{R}^{L \times 1}$. The first stream uses a frame classifier on each frame embedding, followed by max-pooling on the scores:

$$c_m(X) = g_m(f(x_1), \dots, f(x_n)) = \max\{W_0 h_1, \dots, W_0 h_n\}$$

where W_0 is a weight vector. The max-pooling stream provides the frame with the highest score (the *critical instance*).

The second stream aggregates the above frame embeddings into a single sequence embedding, which is further scored by a bag classifier. It transforms each instance embedding h_i , obtained in the first stream (including the critical instance embedding h_m) into two vectors, query $q_i \in \mathbb{R}^{L \times 1}$ and information $v_i \in \mathbb{R}^{L \times 1}$, which are given respectively by:

$$q_i = W_q h_i, \quad v_i = W_v h_i, \quad i = 0, \dots, N - 1$$

where W_q and W_v are learnable weight matrices. Then, a distance measurement U , which has a similar structure and meaning of the attention operation used in Transformers architecture [43], is defined as follows:

$$U(h_i, h_m) = \frac{\exp(\langle q_i, q_m \rangle)}{\sum_{k=0}^{N-1} \exp(\langle q_k, q_m \rangle)}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product of two vectors. As we can see from the formulation, the distance is computed only between the critical instance and all the instances in the bag. This ensures a linear complexity of $O(n)$ rather than quadratic like the attention mechanism.

Overall bag representation b is computed by combining the information vectors v_i of all instances using a weighted sum, where the weights are determined by the distances to the critical instance:

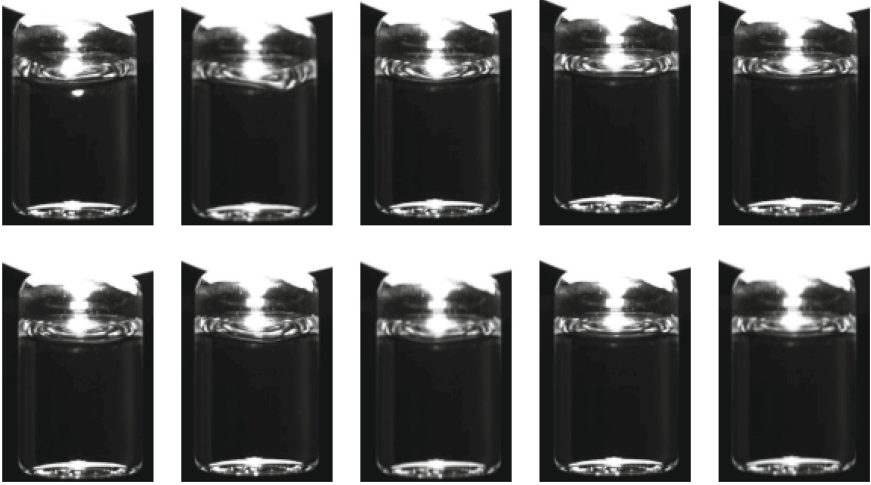


Fig. 3. Sample images from a sequence of a vial containing no impurities.

$$b = \sum_{i=1}^n U(h_i, h_m)v_i$$

The bag score of the second stream is obtained through a final linear layer. This score, averaged with the one of the first stream $c_m(B)$, produced the final score.

Since our research tackles a multiclass problem, DSMIL has been adapted accordingly. We use max-pooling to determine the critical instance of each class, and then we compute attention weights for each class individually with respect to the corresponding critical instance. As a result, the bag embedding b becomes a matrix with dimensions $L \times C$, where C represents the number of classes. In this matrix, each entry is a weighted sum of the instance information vectors v_i . The final fully connected layer for the classification has C output channels.

DSMIL exploited SimCLR [6], which stands for Simple Contrastive Learning Representation, to produce a robust feature extractor in an unsupervised learning setting. In our case, SimCLR trains a ResNet-18 to drastically reduce the input size of each frame by embedding it into a vector. It randomly selects pairs of images from the sequences, applies random augmentations to improve the robustness, and trains the network to maximize similarity between images belonging to the same sequence while minimizing similarity between images from different sequences. After training, ResNet-18 is used to generate the embeddings for single frames within the first stream of DSMIL.

4 Experiments and Results

Dataset. The samples under examination are glass vials with silicone caps filled with distilled water. Image acquisition was performed on a rotating test bench

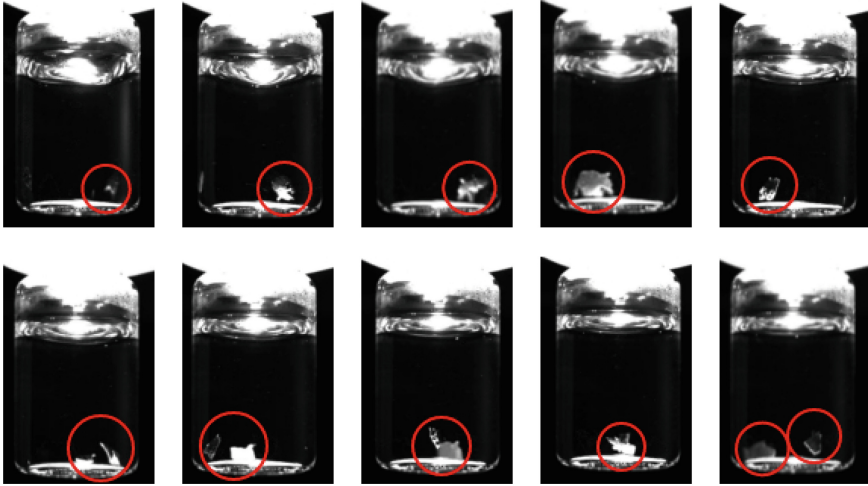


Fig. 4. Sample images from a sequence of a vial containing glass impurities.

using a Matrix Vision camera with a bottom white illuminator and a single LED. The vials were rotated at a speed of 200 rpm with an acceleration and deceleration time of 400 ms. Image acquisitions of each sample occurred after a rotation of the vial, with a delay of approximately 100 ms.

Before the acquisition procedure, each vial was cleaned on the outside with alcohol to remove marks and residual particles from the glass. The dataset used to train and evaluate our models is composed of 2,426 vial sequences, where each sequence consists of 19 frames, for a total of 46,094 images.

The dataset contains annotations for five different classes. One class represents *good* vials, indicating the absence of impurities. The other four classes refer to different types of foreign particles: *brown* impurities, corresponding to burnt plastic particles, *black* defects, corresponding to rubber or silicone particulates, a class is for *glass* pieces of various sizes, and the last class for *sand* residues. These are essentially the defects shown in Fig. 1. Samples from a clean sequence are reported in Fig. 3, while images extracted from sequences containing glass and sand impurities are depicted in Fig. 4 and Fig. 5.

Pre-processing. Each frame in the dataset encompassed a pre-processing phase consisting of a center crop to a fixed dimension of 325×268 pixels to isolate the vial, followed by a rotation to ensure a consistent vial alignment.

Implementation Details. The experiments were conducted for both the presented methods by dividing the dataset into 4 separate and non-overlapping sequence splits. For each split, each training set consists of 2,000 sequences, while each test set consists of 426 sequences.

For what concerns ResNet-18 with voting and concat, we used SGD with momentum as optimizer, a learning rate of 0.01, ReduceLRonPlateau as sched-

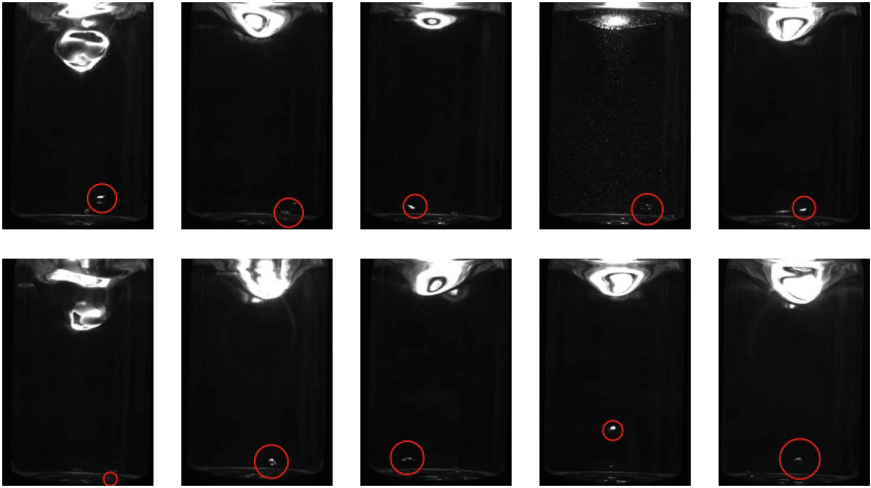


Fig. 5. Sample images from a sequence of a vial containing sand impurities.

uler, and a batch size of 4 sequences. In this case, convergence is achieved after a total of 30 epochs.

For DSMIL, instead, we used Adam as optimizer, keeping a fixed learning rate of 0.0002 during the training and a batch size of 1. The feature extractor (ResNet-18) is trained using the SimCLR framework on each frame of all the sequences. To achieve convergence, DSMIL is trained for a total of 120 epochs. Both DSMIL and ResNet are trained using NVIDIA Tesla K80 as GPU.

Results. The classification results are summarized in Table 1. For each method, we reported accuracy, precision, recall, and F1-score computed on the test set, averaged across the five classes. Additionally, we computed the average inference time on a single sequence. We used 4-fold cross-validation to evaluate the model’s performance more robustly and mitigate the risk of overfitting to a specific subset of the data. Thus, the reported results consist of the average metrics computed across all the folds. The results suggest that all models reach good performance on this classification task; in particular, the best-performing method is DSMIL, which reaches an accuracy of 99.53%. DSMIL misclassifies only a few sequences confusing *brown* particles sample as vials’ without impurities. This occurred because these types of impurities consist of very tiny burnt

Table 1. Comparison of different methods on our dataset.

Model	Accuracy	Precision	Recall	F1-Score	Time [ms]
ResNet (voting)	0.9835 ± 0.0071	0.9829 ± 0.0062	0.9851 ± 0.0069	0.9840 ± 0.0064	1257
ResNet (concat)	0.9903 ± 0.0046	0.9899 ± 0.0042	0.9918 ± 0.0048	0.9908 ± 0.0046	1328
DSMIL	0.9953 ± 0.0023	0.9948 ± 0.0020	0.9957 ± 0.0024	0.9952 ± 0.0022	1639

plastic pieces. Comparing the two aggregation methods used for ResNet experiments, we observed that concatenation is slightly more effective than majority voting. We noticed that instances where majority voting failed were due to a misclassification of vials with an impurity as pure vials. This happens because very small impurities (Fig. 5) are only visible in specific frames of the sequence, leading to most of them being assigned the “no impurities” class. Thus, we can conclude that, particularly for challenging-to-detect defects, using concatenation before the ResNet-18 fully connected layers is preferable.

5 Conclusion

In conclusion, this study addresses the critical issue of detecting visible particles in pharmaceutical liquid vials using advanced deep-learning techniques. Over the years, some traditional algorithms, such as SVM and k-means clustering have been explored. More recently, deep learning techniques have outperformed the latter, improving the safety of the final products. In this work, we introduce two methodologies, leveraging ResNet-18 and DSMIL, to classify four types of impurities. To gap the absence of publicly available dataset we also create a new dataset (which is partially released) comprising sequences of images captured from rotating vials, enhances research in this area by providing valuable data for future comparisons. Our methodologies, trained on this dataset, reaches impressive results, with a maximum accuracy of 99.53%, and 99.52% of F1-score.

Future Work. The proposed methodologies exhibited exceptional performance in the designated task, achieving near-optimal scores in multi-class classification. Future research will pivot towards the localization and detection of impurities rather than solely focusing on classification, thereby augmenting the pipeline with explanatory capabilities. Moreover, this allows to classify each detection with its own class, and identify different kind of impurities within the same sample. Another direction of research could focus on improving the inference time in order to obtain real-time performance in a production environment.

Acknowledgement. This work was supported by the University of Modena and Reggio Emilia and Fondazione di Modena, through the FAR 2023 and FARD-2023 funds (Fondo di Ateneo per la Ricerca).

References

1. Abdellah, H., Ahmed, R., Slimane, O.: Defect detection and identification in textile fabric by SVM method. *IOSR J. Eng.* **4**(12), 69–77 (2014)
2. Bontempo, G., Porrello, A., Bolelli, F., Calderara, S., Ficarra, E.: DAS-MIL: distilling across scales for MIL classification of histological WSIs. In: Greenspan, H., et al. (eds.) *MICCAI 2023*. LNCS, vol. 14220, pp. 248–258. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-43907-0_24
3. Bukofzer, S., et al.: Industry perspective on the medical risk of visible particles in injectable drug products. *PDA J. Pharm. Sci. Technol.* **69**(1), 123–139 (2015)

4. Calvo, C., Micarelli, A., Sanginetto, E.: Automatic annotation of tennis video sequences. In: Van Gool, L. (ed.) DAGM 2002. LNCS, vol. 2449, pp. 540–547. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-45783-6_65
5. Carbonneau, M.A., Cheplygina, V., Granger, E., Gagnon, G.: Multiple instance learning: a survey of problem characteristics and applications. *Pattern Recogn.* **77**, 329–353 (2018)
6. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: *International Conference on Machine Learning*, pp. 1597–1607. PMLR (2020)
7. Cornia, M., Baraldi, L., Serra, G., Cucchiara, R.: SAM: pushing the limits of saliency prediction models. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1890–1892 (2018)
8. Ding, F., Zhuang, Z., Liu, Y., Jiang, D., Yan, X., Wang, Z.: Detecting defects on solid wood panels based on an improved SSD algorithm. *Sensors* **20**(18), 5315 (2020)
9. Dobashi, K., Saito, M., Hayashi, T.: Advanced quality control of quartz parts for semiconductor equipment based on the food industry’s well-established QC methodology (HACCP). In: *2008 International Symposium on Semiconductor Manufacturing (ISSM)*, pp. 29–32. IEEE (2008)
10. Fang, J., Wang, Y., Wu, C.: Binocular automatic particle inspection machine for bottled medical liquid examination. In: *2013 Chinese Automation Congress*, pp. 397–402. IEEE (2013)
11. Ge, J., Wang, Y., Zhou, B., Zhang, H.: Intelligent foreign particle inspection machine for injection liquid examination based on modified pulse-coupled neural networks. *Sensors* **9**(05), 3386–3404 (2009)
12. Ge, J., et al.: A system for automated detection of ampoule injection impurities. *IEEE Trans. Autom. Sci. Eng.* **14**(2), 1119–1128 (2015)
13. Ge, Y., Liu, Y., Xu, C.: Particle detection of complex images based on convolutional neural network. In: *2022 41st Chinese Control Conference (CCC)*, pp. 7228–7233. IEEE (2022)
14. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580–587 (2014)
15. Gross, M.A.: The danger of particulate matter: in solutions for intravenous use. *Drug Intell.* **1**(1), 12–14 (1967)
16. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: *2016 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778 (2016)
17. Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: *2017 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4700–4708 (2017)
18. Islam, M.J., Basalamah, S.M., Ahmadi, M., Sid-Ahmed, M.A.: Computer vision-based quality inspection system of transparent gelatin capsules in pharmaceutical application. *Am. J. Intell. Syst.* **2**(1), 14–22 (2012)
19. Jia, H., Murphey, Y.L., Shi, J., Chang, T.S.: An intelligent real-time vision system for surface defect detection. In: *Proceedings of the 17th International Conference on Pattern Recognition, ICPR 2004*, vol. 3, pp. 239–242. IEEE (2004)
20. Johnson, J.L., Padgett, M.L.: PCNN models and applications. *IEEE Trans. Neural Netw.* **10**(3), 480–498 (1999)

21. Kekre, H., Mishra, D., Desai, V.: Detection of defective pharmaceutical capsules and its types of defect using image processing techniques. In: 2014 International Conference on Circuits, Power and Computing Technologies [ICCPCT], pp. 1190–1195. IEEE (2014)
22. Li, B., Li, Y., Eliceiri, K.W.: Dual-stream multiple instance learning network for whole slide image classification with self-supervised contrastive learning. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 14318–14328 (2021)
23. Li, Y., Han, Z., Xu, H., Liu, L., Li, X., Zhang, K.: YOLOv3-lite: a lightweight crack detection network for aircraft structure based on depthwise separable convolutions. *Appl. Sci.* **9**(18), 3781 (2019)
24. Lin, T., Yu, Z., Hu, H., Xu, Y., Chen, C.W.: Interventional bag multi-instance learning on whole-slide pathological images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 19830–19839 (2023)
25. Liu, H., Wang, Y., Duan, F.: Glass bottle inspector based on machine vision. *Int. J. Comput. Inf. Eng.* **2**(8), 2682–2687 (2008)
26. 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
27. Liu, Y., Chen, S., Tang, T., Zhao, M.: Defect inspection of medicine vials using LBP features and SVM classifier. In: 2017 2nd International Conference on Image, Vision and Computing (ICIVC), pp. 41–45. IEEE (2017)
28. Mathavan, S., Kumar, A., Kamal, K., Nieminen, M., Shah, H., Rahman, M.: Fast segmentation of industrial quality pavement images using laws texture energy measures and k-means clustering. *J. Electron. Imaging* **25**(5), 053010 (2016)
29. Mazaheri, M., et al.: Monitoring of visible particles in parenteral products by manual visual inspection reassessing size threshold and other particle characteristics that define particle visibility. *J. Pharm. Sci.* **113**(3), 616–624 (2024)
30. Meenu, M., Kurade, C., Neelapu, B.C., Kalra, S., Ramaswamy, H.S., Yu, Y.: A concise review on food quality assessment using digital image processing. *Trends Food Sci. Technol.* **118**, 106–124 (2021)
31. Noh, H., Hong, S., Han, B.: Learning deconvolution network for semantic segmentation. In: 2015 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1520–1528 (2015)
32. Panariello, A., Porrello, A., Calderara, S., Cucchiara, R.: Consistency-based self-supervised learning for temporal anomaly localization. In: Karlinsky, L., Michaeli, T., Nishino, K. (eds.) ECCV 2022. LNCS, vol. 13805, pp. 338–349. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-25072-9_22
33. Pollastri, F., et al.: Confidence calibration for deep renal biopsy immunofluorescence image classification. In: 2020 25th International Conference on Pattern Recognition (ICPR), pp. 1298–1305. IEEE (2021)
34. Pollastri, F., et al.: A deep analysis on high resolution dermoscopic image classification. *IET Comput. Vis.* **15**(7), 514–526 (2021)
35. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: unified, real-time object detection. In: 2016 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 779–788 (2016)
36. Ren, 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)

37. Ritter, C., Spilger, R., Lee, J.Y., Bartenschlager, R., Rohr, K.: Deep learning for particle detection and tracking in fluorescence microscopy images. In: 2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI), pp. 873–876. IEEE (2021)
38. Roberti, I., Lovino, M., Di Cataldo, S., Ficarra, E., Urgese, G.: Exploiting gene expression profiles for the automated prediction of connectivity between brain regions. *Int. J. Mol. Sci.* **20**(8), 2035 (2019)
39. Roy, S., Sanginetto, E., Demir, B., Sebe, N.: Deep metric and hash-code learning for content-based retrieval of remote sensing images. In: 2018 IEEE International Geoscience and Remote Sensing Symposium, IGARSS 2018, pp. 4539–4542. IEEE (2018)
40. Szatmári, I., Schultz, A., Rekeczky, C., Kozek, T., Roska, T., Chua, L.O.: Morphology and autowave metric on CNN applied to bubble-debris classification. *IEEE Trans. Neural Netw.* **11**(6), 1385–1393 (2000)
41. Tulbure, A.A., Tulbure, A.A., Dulf, E.H.: A review on modern defect detection models using DCNNs-Deep convolutional neural networks. *J. Adv. Res.* **35**, 33–48 (2022)
42. Unnikrishnan, S., Donovan, J., Macpherson, R., Tormey, D.: Machine learning for automated quality evaluation in pharmaceutical manufacturing of emulsions. *J. Pharm. Innov.* **15**, 392–403 (2020)
43. Vaswani, A., et al.: Attention is all you need. In: *Advances in Neural Information Processing Systems*, vol. 30 (2017)
44. Vishwanatha, C.R., Asha, V., More, S., Divya, C., Keerthi, K., Rohaan, S.P.: A survey on defect detection of vials. In: Saraswat, M., Chowdhury, C., Kumar Mandal, C., Gandomi, A.H. (eds.) *Data Science and Applications. LNNS*, vol. 551, pp. 171–186. Springer, Singapore (2023). https://doi.org/10.1007/978-981-19-6631-6_13
45. Wang, S., Zhuo, Q., et al.: Detection of glass chips in liquid injection based on computer vision. In: 2015 International Conference on Computational Intelligence and Communication Networks (CICN), pp. 329–331. IEEE (2015)
46. Wang, Y., Liu, M., Zheng, P., Yang, H., Zou, J.: A smart surface inspection system using faster R-CNN in cloud-edge computing environment. *Adv. Eng. Inform.* **43**, 101037 (2020)
47. Yi, J., Zhang, H., Mao, J., Chen, Y., Zhong, H., Wang, Y.: Pharmaceutical foreign particle detection: an efficient method based on adaptive convolution and multi-scale attention. *IEEE Trans. Emerg. Top. Comput. Intell.* **6**(6), 1302–1313 (2022)
48. Zhang, H., et al.: Automated machine vision system for liquid particle inspection of pharmaceutical injection. *IEEE Trans. Instrum. Meas.* **67**(6), 1278–1297 (2018)
49. Zhou, X., et al.: Automated visual inspection of glass bottle bottom with saliency detection and template matching. *IEEE Trans. Instrum. Meas.* **68**(11), 4253–4267 (2019)



RGB-T Object Detection via Group Shuffled Multi-receptive Attention and Multi-modal Supervision

Jinzhong Wang¹, Xuetao Tian^{1,2}, Shun Dai¹, Tao Zhuo³, Haorui Zeng¹,
Hongjuan Liu², Jiaqi Liu², Xiuwei Zhang¹(✉), and Yanning Zhang¹

¹ Northwestern Polytechnical University, Xi'an 710072, China
{xwzhang, ynzhang}@nwpu.edu.cn

² Xi'an ASN Technology Group Co., Ltd., Xi'an 710065, China

³ Northwest A & F University, Yangling 712100, China

Abstract. Multispectral object detection, utilizing both visible (RGB) and thermal infrared (T) modals, has garnered significant attention for its robust performance across diverse weather and lighting conditions. However, effectively exploiting the complementarity between RGB-T modals while maintaining efficiency remains a critical challenge. In this paper, a very simple Group Shuffled Multi-receptive Attention (GSMA) module is proposed to extract and combine multi-scale RGB and thermal features. Then, the extracted multi-modal features are directly integrated with a multi-level path aggregation neck, which significantly improves the fusion effect and efficiency. Meanwhile, multi-modal object detection often adopts union annotations for both modals. This kind of supervision is not sufficient and unfair, since objects observed in one modal may not be seen in the other modal. To solve this issue, Multi-modal Supervision (MS) is proposed to sufficiently supervise RGB-T object detection. Comprehensive experiments on two challenging benchmarks, KAIST and DroneVehicle, demonstrate the proposed model achieves the state-of-the-art accuracy while maintaining competitive efficiency.

Keywords: Multispectral object detection · Attention mechanism · Group shuffle · Multi-modal supervision

1 Introduction

As an integral branch of computer vision, object detection has a wide range of applications in real-world scenarios. However, unimodal object detection methods often encounter limitations from unfavorable conditions, such as dim lighting, fog, or occlusion [37]. To address this challenge, a common approach is to fuse complementary information of different modals, which has been widely used in tasks such as video surveillance [1] and autonomous driving [23]. For example, visible cameras typically capture complex details such as color and texture under sufficient lighting. But in dark scenarios, their effectiveness will

be significantly reduced. In contrast, thermal cameras specialize in capturing the thermal radiation emitted by objects and are almost unaffected by changes in lighting and weather conditions. Nevertheless, the resolution of thermal images is lower, and the texture and color of objects are absent. Consequently, the sufficient fusion of complementary information of RGB and thermal modals is critical.

Feature-level fusion, also known as middle fusion, has been widely explored since its excellent performance. It often adopts two separated sub-networks to extract feature maps from RGB and thermal modals and employs methods such as channel concatenation [11] and weighted fusion [19] for further fusion. Researchers also explored more complex fusion modules to fully utilize the potential complementary information between RGB and thermal modal, such as illumination-aware techniques [7] and attention modules [31]. However, these methods are typically based on two-stage R-CNN variants [10, 32] and with overly complex designs fail to achieve an optimal balance between accuracy and efficiency. In addition, prevalent studies often utilize union annotations [38] as detection supervision. It may cause the network easily affected by noise in weak alignment or modal-absent situations. Moreover, using union annotations to supervise two modals is unfair, since objects observed in one modal may not be seen in the other modal. It may cause confusion and failure to fully utilize the precise information of each modal.

In this paper, we propose a novel one-stage SAMS-YOLO network to address the problems mentioned above. Specifically, for significant and efficient multi-modal feature fusion, we introduce a lightweight multi-scale attention module to extract RGB-T multi-receptive field features and combine them via a novel parameter-free group shuffle operation. Through the multi-level path aggregation neck [16], the combined multi-modal features are effectively and sufficiently fused. Additionally, to ensure robust and accurate object detection, we propose a multi-modal supervision strategy consisting of three branches for detection, i.e., RGB, thermal, and fusion, which is supervised by visible, thermal, and union annotations separately. It can solve the problem of unfair supervision caused by union annotations. By integrating the aforementioned lightweight and efficient modules into the one-stage YOLOv5 [9] framework, we achieve a good balance between detection accuracy and efficiency. Extensive experiments are conducted on the KAIST and DroneVehicle datasets, the results demonstrate superior detection performance. The contributions of this paper are summarized as follows:

- 1) A simple Group Shuffled Multi-receptive Attention (GSMA) module is proposed to effectively extract and combine multi-modal multi-receptive field features. Through the integration with the top-down and bottom-up PANet [16], multi-modal features are efficiently and sufficiently fused. With this module, we achieve a reduction of 2.07%, 2.28%, and 1.93% on MR^{-2} across all-day, day, and night subsets of the KAIST dataset, respectively.
- 2) A Multi-modal Supervision (MS) strategy is proposed to effectively guide the network to learn precise and robust feature representations by leveraging

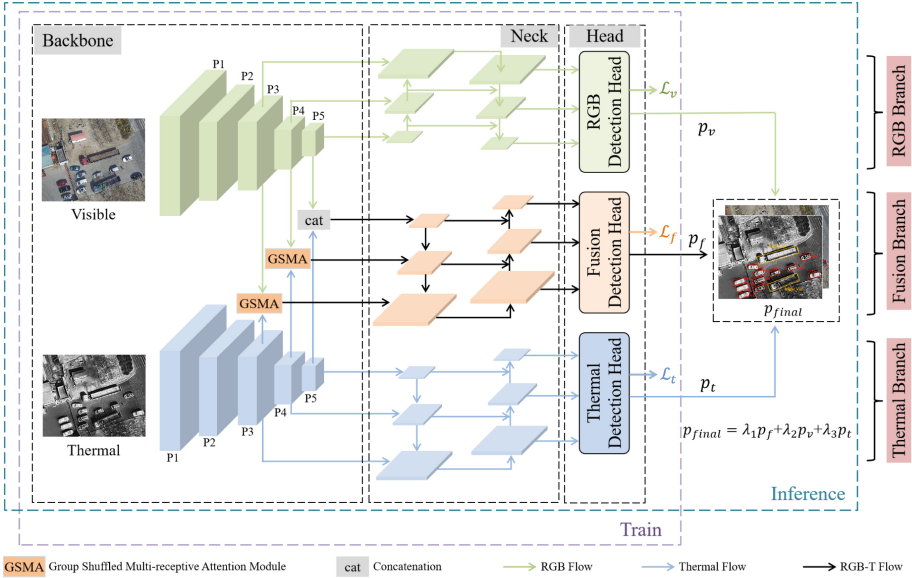


Fig. 1. Architecture of the proposed SAMS-YOLO. The multi-modal supervision strategy is applied to the RGB, thermal, and fusion branches. During training, the RGB, thermal, and union annotations are used as supervision to calculate detection loss. During inference, a decision-level fusion is applied to fuse the RGB, thermal, and fusion branch results.

visible, thermal, and union annotations as supervision. This strategy leads to a reduction of 2.66%, 2.87%, and 2.25% on MR^{-2} across all-day, day, and night subsets of the KAIST dataset, respectively.

- 3) The proposed RGB-T object detection method, namely SAMS-YOLO, integrates GSMA and MS into YOLOv5, enhancing the detection ability of small targets, night scenes, and occlusion situations. It achieves the state-of-the-art results on two challenging datasets: KAIST multispectral pedestrian dataset and DroneVehicle remote sensing dataset, while maintaining a fast processing speed.

2 Related Work

2.1 Multispectral Object Detection

Due to the significant advantages offered by collaborative detection in visible and thermal domains, multispectral object detection has made remarkable progress. Liu et al. [15] adopted a two-stage method Faster R-CNN [18] as the framework, and incorporated two separate pedestrian detectors on visible and thermal images, respectively. SDS-RCNN [2], MSDS-RCNN [11], and I^2 MDet [34] leveraged ground truth bounding boxes as weak segmentation annotations to

facilitate supervised learning. IAF R-CNN [12] integrated illumination-aware modules into the detection network, enabling dynamic weight adjustment for different input modals based on varying light conditions. AR-CNN [32] and TSFADet [28] proposed modal alignment operations to solve the problem of temporal and spatial misalignment between RGB-T modals. Additionally, Zhou et al. [38] investigated the issue of multi-modal imbalance resulting from the inadequate fusion of modal information in their MBNet framework. UA-CMDet [19] tackled the quantification of uncertainty associated with multi-modal targets through uncertainty perception and illumination estimation. Li et al. [14] proposed multiscale cross-modal homogeneity enhancement and confidence-aware feature fusion in their MCHE-CF. Notably, most of the approaches mentioned above are based on two-stage R-CNN variants, which suffer from slow detection speeds due to their intricate architecture and multiple stages involved.

2.2 Multi-modal Features Fusion

Fusing multi-modal features is a crucial aspect of multispectral object detection, which can be categorized into four types: early fusion, middle fusion, late fusion, and decision-level fusion. Among these, middle fusion strategies have been widely explored and demonstrated to be more effective, as they are more flexible in design and enable deeper feature fusion [5, 17]. MSDS-RCNN [11] and UA-CMDet [19] employed a simple channel concatenation approach for feature fusion. CSAA [3] combined channel switching and channel concatenation. CIAN [31] introduced a cross-modal interaction attention module to adaptively recalibrate channel responses. MBNet [38] leveraged the differences between modals to design a differential modal-aware fusion module. SC-MPD [5] incorporated a spatial-contextual feature aggregation block to efficiently utilize multiple source features. DCMNet [24] improved feature complementarity through dynamic local and non-local feature aggregation modules. C²Former-S²ANet [29] employed an intermodality cross-attention module to obtain the calibrated and complementary features between the RGB-T modals. However, a simple concatenation fusion method cannot guarantee accuracy, and complex modules significantly result in high memory usage and latency. To address these challenges, we propose a novel group shuffled multi-receptive attention module that considers both channel and multi-spatial level attention while ensuring low computational costs.

3 Method

Figure 1 illustrates the overall architecture of our SAMS-YOLO model. It comprises three primary components: the dual-stream feature extractor backbone, three-branch detection neck, and three-branch detection head. In the training phase, we utilize RGB, thermal, and union annotations of the RGB-T modals for detection supervision. For inference, we adopt a decision-level fusion strategy to weigh the prediction from the RGB, thermal, and fusion branches.

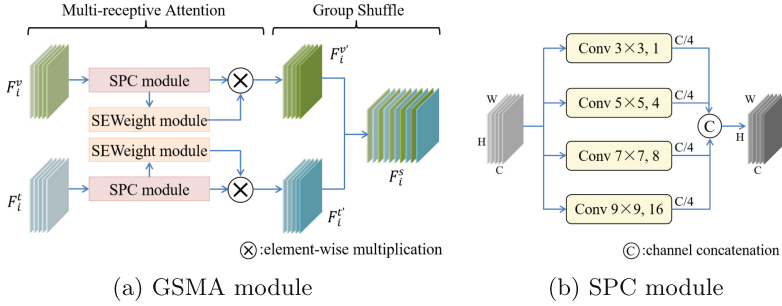


Fig. 2. The structure of Group Shuffled Multi-receptive Attention module. (a) shows the data flow structure of the GSMA. (b) shows the SPC structure in (a).

3.1 Framework Overview

As shown in Fig. 1, the network takes RGB-T image pairs as input, and a dual-stream backbone with five layers (denoted as P1 to P5) to extract hierarchical feature maps. The size of features generated by P1 to P5 is 2, 4, 8, 16, and 32 times downsampling of the input images, respectively. The neck and head are designed with a three-branch structure of RGB, thermal, and fusion. Firstly, the visible and thermal features obtained by P3 to P5, are passed to the corresponding neck branch, i.e., the top one and the bottom one, for refined feature representation and prediction, respectively. Secondly, two GSMA modules are employed to extract RGB-T multi-receptive field features and fully combine them at the P3 and P4 stages. Thirdly, the concatenated features at the P5 stage and the fused features enhanced by GSMA are then fed into the fusion neck and head (middle branch) for further prediction. The neck here is the top-down and bottom-up PANet [16]. Subsequently, as expressed in Eq. 1, the final prediction result p_{final} is obtained by taking the weighted average of fusion detection p_f , visible detection p_v , and thermal detection p_t . λ_1 , λ_2 , and λ_3 are hyper-parameters, and detailed in the experimental implementation.

$$p_{final} = \lambda_1 p_f + \lambda_2 p_v + \lambda_3 p_t \tag{1}$$

3.2 Group Shuffled Multi-receptive Attention Module

The motivation of this work is to build an efficient and effective multi-modal attention mechanism to improve multi-modal feature fusion. As illustrated in Fig. 2(a), the structure of the GSMA module is quite straightforward, it contains two parts: multi-receptive attention and group shuffle.

Multi-receptive Attention. Previous studies have rarely focused on the impact of multi-receptive field features on multi-modal feature fusion. Inspired by [30], we introduce a multi-receptive attention mechanism to effectively extract the multi-modal multi-scale spatial information. As shown in Fig. 2(a), two

Squeeze Pyramid Concat (SPC) modules [30] are adopted to obtain multi-receptive field feature maps on channel-wise of the input features F_i^v and F_i^t ($i \in \{3, 4\}$). Then, two SEWeight modules [30] are applied to extract channel-wise attention weights for the RGB-T multi-scale features obtained from the SPC module. It is worth noting that the SEWeight module [30] can encode global information and adaptively recalibrate channel-wise relationships through squeezing and excitation operations. After that, element-wise multiplication is applied to recalibrate the weights and corresponding feature maps. Finally, the refined features $F_i^{v'}$ and $F_i^{t'}$ at different receptive fields are combined by the group shuffle operation to obtain F_i^s .

The structure of the SPC module is shown in Fig. 2 (b). The input features are extracted by multi-scale group convolution kernels to capture information regarding different spatial resolutions and depths. The multi-scale group convolution kernel sizes are 3×3 , 5×5 , 7×7 , and 9×9 , with corresponding convolution groups set as 1, 4, 8, and 16. Then, the multi-receptive features are merged through channel concatenation.

Group Shuffle. The RGB-T features contain rich complementary information such as color, texture, and contour. To efficiently learn modal correlations, we propose a new representation module called group shuffle. As shown in Fig. 3, we first split and group the RGB and thermal features along the channel dimension, and then combine them through alternating merging. This parameter-free operation not only preserves the similarity of intra-group modal but also makes inter-group modal responses more diverse, effectively improving the fusion of multi-modal features. Assuming that both RGB and thermal features have C channels, we split these channels into K groups, each containing N channels, where $N = C/K$. The channel index j of F_j^v and F_j^t is mapped to new position j' according to Eq. 2. It should be noted that when $K = 1$, group shuffle is equivalent to channel concatenation, and when $K = C$, it is equivalent to channel shuffle [35].

$$j' = \begin{cases} j \bmod N + \lfloor \frac{j}{N} \rfloor \times 2N, & F_j^v \in F^v \\ j \bmod N + \lfloor \frac{j}{N} \rfloor \times 2N + N, & F_j^t \in F^t \end{cases} \quad (2)$$

The multi-modal features after the group shuffle are fully mixed with each other at different multi-receptive fields. By aggregating through multi-level top-down and bottom-up path neck, a comprehensive fusion of multi-modal and multi-scale features can be achieved, and the detection ability of small targets, night scenes, and occlusion situations can be improved.

3.3 Multi-modal Supervision Strategy

Due to the possibility of spatial misalignment between visible and thermal images, the position of the same object may be different in the two modals. Simply using the union annotations of RGB-T modals may lead the network to be subjected to biased supervised information due to misalignment, which is

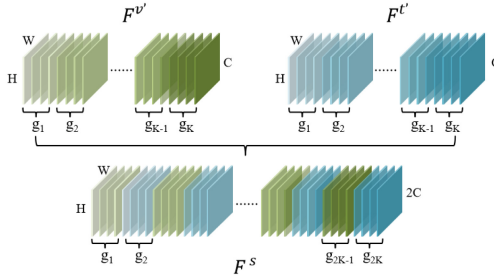


Fig. 3. The structure of the Group Shuffle.

not conducive to learning more accurate feature representations. Furthermore, in situations where specific modal are not available, such as a visible camera in dark night environment, the union annotations used to supervise the visible branch may introduce noise disturbance. Therefore, we use visible annotations, thermal annotations, and the union annotations of RGB-T modals to provide more accurate feature learning supervision for feature extraction and detection. As shown in Fig. 1, during training, the visible, thermal, and union annotations are used to supervise the prediction of the RGB, thermal, and fusion branches, respectively.

Additionally, to guide the network to extract more accurate feature representations, following [11], we add segmentation prediction heads in the dual-stream backbone and use the RGB and thermal ground truth bounding boxes as segmentation supervision. This method is only used during training and does not affect the network inference speed.

3.4 Loss Function

The loss function is built upon YOLOv5 [9], incorporating the multi-modal supervision loss and segmentation supervision loss. As shown in Eq. 3, \mathcal{L}^f , \mathcal{L}^v and \mathcal{L}^t are fusion, visible and thermal detection loss, respectively. \mathcal{L}_{cls} , \mathcal{L}_{obj} , \mathcal{L}_{bbox} represents the object classification loss, object confidence loss, and object coordinate position loss, respectively. \mathcal{L}_{seg} is segmentation loss of binary cross-entropy. λ_{cls} , λ_{obj} , λ_{bbox} and λ_{seg} are correction factors and are detailed in the experimental implementation.

$$\mathcal{L}_{total} = \lambda_{cls} (\mathcal{L}_{cls}^f + \mathcal{L}_{cls}^v + \mathcal{L}_{cls}^t) + \lambda_{obj} (\mathcal{L}_{obj}^f + \mathcal{L}_{obj}^v + \mathcal{L}_{obj}^t) + \lambda_{bbox} (\mathcal{L}_{bbox}^f + \mathcal{L}_{bbox}^v + \mathcal{L}_{bbox}^t) + \lambda_{seg} \mathcal{L}_{seg} \quad (3)$$

4 Experiments

In this section, the experimental datasets are introduced firstly. Then, the implementation details and comparison experiments are presented. Finally, ablation studies are conducted to verify the effectiveness of each component in our approach.

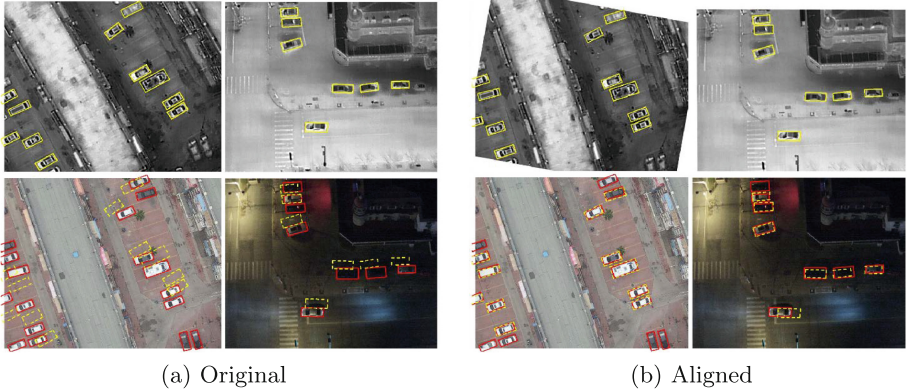


Fig. 4. Illustration of the modal misalignment problem in the DroneVehicle training set. (a) and (b) depict the original and aligned RGB-T image pairs, where the top images are thermal images and the bottom images are visible images. Yellow boxes indicate annotations on the thermal images, while red boxes indicate annotations on the visible images. Both modal annotations are visualized on the visible image. (Color figure online)

4.1 Datasets and Evaluation Metric

Our approach is evaluated on two challenging datasets: KAIST multispectral pedestrian dataset [8] and DroneVehicle remote sensing dataset [19]. These two datasets along with their related evaluation metrics are depicted as follows.

KAIST Dataset. The KAIST dataset [8] consists of 95,328 RGB-T image pairs with 103,128 pedestrian annotations. Following [15, 32], we use 7,095 image pairs for training and 2,252 image pairs for testing. Specifically, The test images contain 1,455 day-time images (‘Day’) and 797 night-time images (‘Night’). We use the standard MR^{-2} (log-average Miss Rate over false positive per image range of $[10^{-2}, 10^0]$) [8] and FPS (frames per second) to evaluate the performance. Note that a lower MR^{-2} indicates a better detection performance.

DroneVehicle Dataset. The DroneVehicle dataset [19] is a large-scale drone-based image dataset of oriented vehicles. It contains 28,439 RGB-T image pairs with 953,087 instances covering urban roads, residential areas, parking lots, and other scenarios. Specifically, it contains five categories, i.e., car, truck, bus, van, and freight car.

Due to the lack of union annotations in the DroneVehicle training set, to verify the effectiveness of our method, we established union annotations for the first time by taking the union of independent annotations of RGB and thermal modals. Specifically, due to the cross-modal misalignment problems [28], we use the method proposed by [36] to register 2,441 RGB-T image pairs in the training set. As shown in Fig. 4, the position misalignment problem has been addressed compared with the original image pairs. Finally, we use the aligned training set

Table 1. Evaluation results on the KAIST dataset.

Method		MR^{-2}			Platform	FPS
		All-Day	Day	Night		
ACF [8]	RGB+IR	47.32	42.57	56.17	MATLAB	0.37
Halfway Fusion [15]	RGB+IR	25.75	24.88	26.59	TITAN X	2.33
IATDNN + IASS [7]	RGB+IR	14.95	14.67	15.72	TITAN X	4.00
CIAN [31]	RGB+IR	14.12	14.77	11.13	1080Ti	14.29
MSDS-RCNN [11]	RGB+IR	11.34	10.53	12.94	TITAN X	4.55
AR-CNN [32]	RGB+IR	9.34	9.94	8.38	1080Ti	8.33
CMPD [13]	RGB+IR	8.16	8.77	7.31	1080Ti	9.09
MBNet [38]	RGB+IR	8.13	8.28	7.86	1080Ti	14.29
SC-MPD [5]	RGB+IR	8.07	8.16	7.51	Tesla P6	10
BAANet [26]	RGB+IR	7.92	8.37	6.98	1080Ti	14.29
UGCML [10]	RGB+IR	8.18	6.96	7.89	1080Ti	11.11
CPFM [20]	RGB+IR	7.09	5.61	6.62	3090Ti	-
MCHE-CF [14]	RGB+IR	6.71	7.58	5.52	-	-
DCMNet [24]	RGB+IR	5.84	6.48	4.60	3090	7.14
YOLOv5 [9]	RGB	18.72	13.48	28.45	2080Ti	83.33
	IR	16.90	22.33	6.34	2080Ti	83.33
YOLOv5 [9] (early fusion)	RGB+IR	17.61	20.67	12.18	2080Ti	58.48
SAMS-YOLO (ours)	RGB+IR	5.26	6.00	3.81	2080Ti	19.31

for training and the original test set for evaluation. Following [19], we evaluate the detection performance by utilizing the mean average precision (mAP) under different IoU thresholds as the evaluation metric. Specifically, we select $mAP_{0.5}$ and mAP in our experiments. Here, the mAP indicates that the IoU threshold is set from 0.50 to 0.95 with a step of 0.05. Note that the evaluation performance of RGB and thermal modal are averaged as the final evaluation results in our experiment.

4.2 Implementation Details

The proposed SAMS-YOLO is based on YOLOv5 [9]. During training, mosaic data enhancement, random HSV enhancement, and horizontal flip are adopted to enhance RGB-T image pairs. The input images in both datasets are resized to 640×640 pixels. The optimizer employed is stochastic gradient descent (SGD) for 150 epochs with a learning rate of 0.001 and a batch size of 6. Weight decay and momentum are set to 0.0001 and 0.937, respectively. The hyper-parameters λ_1 , λ_2 , and λ_3 in Eq. 1 are set to 0.5, 0.25 and 0.25, respectively. The correction factors λ_{cls} , λ_{obj} , λ_{bbox} , and λ_{seg} in Eq. 3 are set to 0.5, 1.0, 0.05, and 0.25, respectively.

Table 2. Evaluation results on the DroneVehicle dataset.

Method	car	truck	van	bus	freight car	mAP _{0.5}	mAP	Platform	FPS
UA-CMDet [19]	87.5	60.7	38.0	87.1	46.8	64.00	-	3090	9.12
Oriented R-CNN [25]	89.9	56.6	46.9	89.6	54.4	67.52	42.60	-	-
RoI Transformer [6]	90.1	60.4	52.2	89.7	58.9	70.29	43.57	-	-
CIAN(OBB) [31]	89.98	62.47	49.59	88.9	60.22	70.23	-	GV100	21.7
AR-CNN(OBB) [32]	90.08	64.82	51.51	89.38	62.12	71.58	-	GV100	18.2
TSFADet [28]	89.88	67.87	53.99	89.81	63.74	73.06	-	GV100	18.6
ViT-B+RVSA [21]	89.7	52.3	44.4	88.0	51.0	65.07	42.63	-	-
C ² Former-S ² ANet [29]	90.2	68.3	58.5	89.8	64.4	74.2	-	TITAN V	-
I ² MDet [34]	96.3	73.4	58.6	93.2	65.0	77.30	46.20	-	-
DTNet-B [33]	90.2	78.1	65.7	89.2	67.9	78.23	52.85	3090	-
SAMS-YOLO-OBB (ours)	97.00	79.57	67.50	95.95	63.75	80.75	57.13	2080Ti	17.83

Table 3. Effect of K in group shuffle. We tune the group hyperparameter K to {1, 2, 4, 8, 16, 32, C}.

Subset	MR^{-2}						
	K = 1	K = 2	K = 4	K = 8	K = 16	K = 32	K = C
All-day	7.30	8.11	7.63	6.89	6.48	7.70	6.77
Day	8.33	9.40	8.92	8.77	7.86	10.03	8.46
Night	5.66	6.07	5.11	3.26	3.94	4.10	3.64

4.3 Comparison on the KAIST Dataset

The performance of SAMS-YOLO on the KAIST Dataset is presented in Table 1. Compared with mainstream multispectral object detection algorithms, our method achieves the best accuracy, reaching 5.26%, 6.00%, and 3.81% MR^{-2} on the reasonable all-day, day, and night subsets, respectively. At the same time, our detector achieves the fastest detection speed of 19.31 FPS on 2080Ti GPU. Compared with unimodal YOLOv5 [9] and early fused YOLOv5 [9, 15], our method has achieved significant performance improvement. This indicates that the proposed GSMA module enhances the fusion ability of complementary information of RGB-T modals, and improves the localization accuracy while maintaining high efficiency. Meanwhile, through the proposed MS strategy, the detection results from the RGB, thermal, and fusion branches ensure the model’s recall rate, achieving superior consequences.

4.4 Comparison on the DroneVehicle Dataset

Since the DroneVehicle is an oriented bounding box detection dataset, to achieve the detection of oriented objects, we modified our model referring to [27], named SAMS-YOLO-OBB. The experimental results are shown in Table 2. Our method

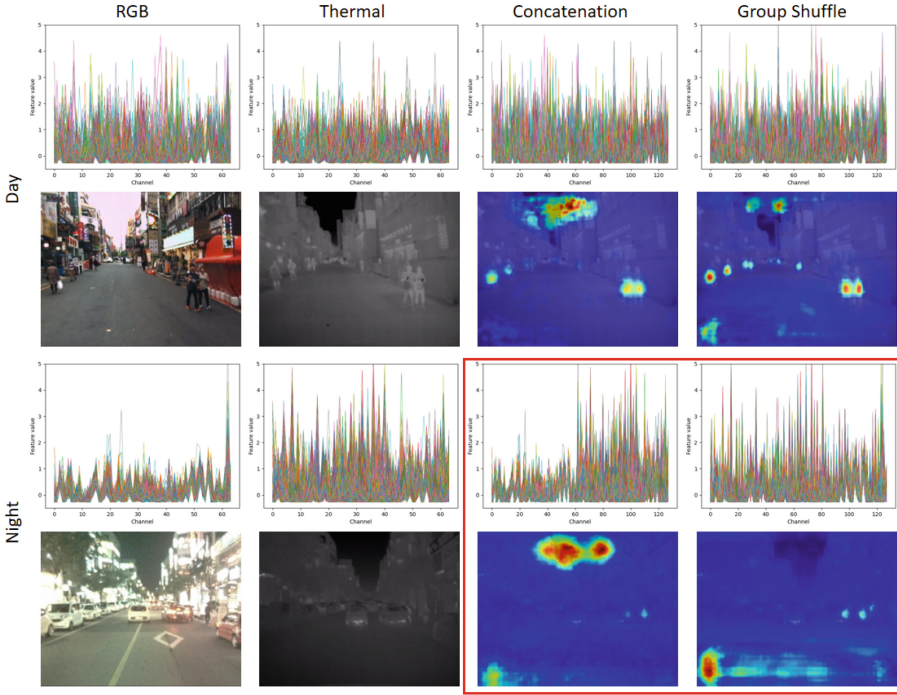


Fig. 5. Impacts of RGB-T feature concatenation and group shuffle. The top and third rows depict the feature map values along the channel dimension, while the second and fourth rows display RGB-T images and corresponding heatmaps. The top two rows showcase day-time scenes, whereas the bottom two rows depict night-time scenes. Notably, the response of RGB features diminishes during night-time. As observed in the feature maps and heatmaps in the bottom right corner, compared to simple concatenation, the group shuffle operation achieves more comprehensive multi-modal feature mixing. Through the GSMA module and multi-path aggregation fusion, the network exhibits heightened attention towards pedestrian areas.

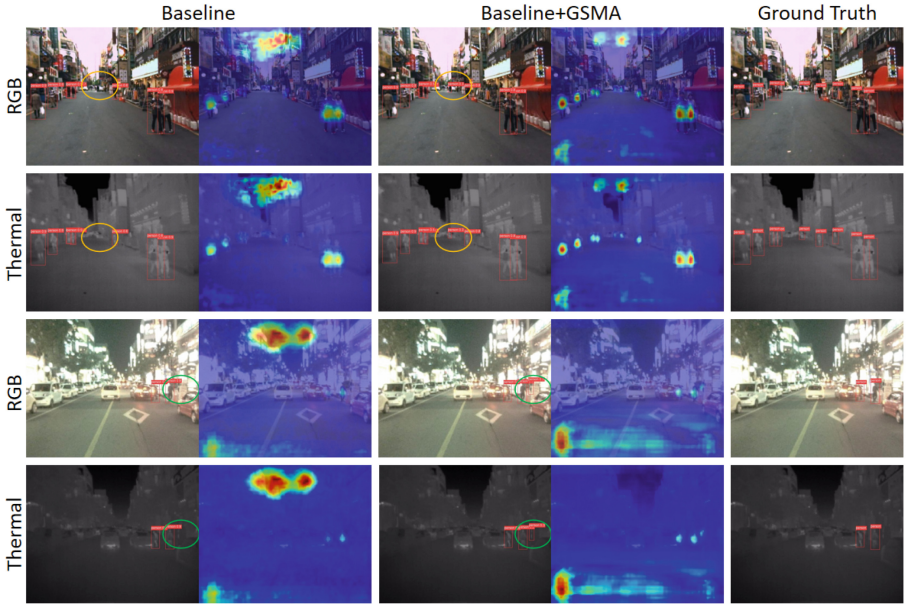
significantly outperforms others, achieving the best performance. Specifically, we improved $mAP_{0.5}$ and mAP metrics by 2.52% and 4.28%, respectively, compared with the previous best method DTNet-B. Furthermore, we achieved the highest scores in the subcategories of car, truck, van, and bus, with $mAP_{0.5}$ increasing by 0.70%, 1.47%, 1.8%, and 2.75% compared to the previous state-of-the-art methods.

4.5 Ablation Study

Ablation experiments are conducted on the KAIST dataset to verify the effect of the GSMA and MS modules. We employ a framework without the GSMA module and RGB and thermal branches in the MS strategy as the baseline in the experiment.

Table 4. Effects of GSMA module. MA means multi-receptive attention and GS means group shuffle operation.

Method	MR^{-2}								
	All-day	Day	Night	Near	Medium	Far	None	Partial	Heavy
Baseline	8.55	10.14	5.87	0.00	13.23	41.28	22.75	28.53	50.02
MA without GS	7.30 _(-1.25)	8.33 _(-1.81)	5.66 _(-0.21)	0.00	12.56 _(-0.67)	40.63 _(-0.65)	22.13 _(-0.62)	29.04 _(0.51)	52.00 _(1.98)
GS before MA	7.39 _(-1.16)	9.78 _(-0.36)	3.90_(-1.97)	0.00	11.46_(-1.77)	40.82 _(-0.46)	21.15 _(-1.60)	26.83 _(-1.70)	47.07 _(-2.95)
GS after MA	6.48_(-2.07)	7.86_(-2.28)	3.94 _(-1.93)	0.00	11.67 _(-1.56)	38.51_(-2.77)	21.00_(-1.75)	24.86_(-3.67)	47.03_(-2.99)
CBAM [22]	7.57 _(-0.98)	9.49 _(-0.65)	4.45 _(-1.42)	0.00	14.22 _(0.99)	44.13 _(2.85)	24.04 _(1.29)	27.86 _(-0.67)	54.36 _(4.34)
GCB [4]	8.48 _(-0.07)	10.58 _(0.44)	5.07 _(-0.80)	0.00	12.93 _(-0.30)	39.64 _(-1.64)	21.86 _(-0.89)	29.84 _(1.31)	52.11 _(2.09)

**Fig. 6.** Examples of detection and heatmaps of baseline method and the addition of GSMA module on the KAIST pedestrian dataset. As shown in the orange and green elliptical areas, the GSMA module enhances the detection ability of small and occluded objects. (Color figure online)**Table 5.** Effects of GSMA module and MS strategy evaluated on KAIST dataset.

GSMA MS		MR^{-2}								
		All-day	Day	Night	Near	Medium	Far	None	Partial	Heavy
×	×	8.55	10.14	5.87	0.00	13.23	41.28	22.75	28.53	50.02
✓	×	6.48 _(-2.07)	7.86 _(-2.28)	3.94 _(-1.93)	0.00	11.67 _(-1.56)	38.51 _(-2.77)	21.00 _(-1.75)	24.86 _(-3.67)	47.03 _(-2.99)
×	✓	5.89 _(-2.66)	7.27 _(-2.87)	3.62 _(-2.25)	0.00	9.21_(-4.02)	34.18 _(-7.10)	17.70_(-5.05)	23.64 _(-4.89)	47.13 _(-2.89)
✓	✓	5.26_(-3.29)	6.00_(-4.14)	3.81 _(-2.06)	0.00	9.91 _(-3.32)	36.25 _(-5.03)	19.05 _(-3.70)	23.04_(-5.49)	46.87_(-3.15)

Effectiveness of GSMA. We first conduct experiments on the effects of different group configurations and the operation position of group shuffle. As illustrated in Table 3, optimal performance is observed when $K = 16$. As shown in Table 4, placing group shuffle (GS) after multi-receptive attention (MA) achieves the best performance. Besides, the proposed GSMA module exhibits prominent superiority, when compared to existing attention methods such as CBAM [22] and GCB [4]. As shown in Table 5, when adding the GSMA into the baseline, it achieves the reductions of 2.07%, 2.28%, and 1.93% on MR^{-2} across the reasonable all-day, day, and night subset, respectively. Figure 5 exhibits some typical images and the corresponding visualized feature maps, we can see that through the group shuffle operation, the fused RGB-T features is inclined to highlight pedestrian regions. Relevant detection result examples are shown in Fig. 6. This indicates that the GSMA facilitates a complementary fusion of RGB-T features and enhances detection accuracy in night scenes and situations involving occlusion.

Effectiveness of MS Strategy. As indicated in Table 5, after incorporating the MS strategy into the baseline, we observe decreases of 2.66%, 2.87%, and 2.25% on MR^{-2} across reasonable all-day, day, and night conditions, respectively. These results indicate that the supervision by utilizing independent annotations for RGB, thermal, and fusion modal is more sufficient and can fully utilize the precise information of each modal.

The combination of the GSMA module and MS strategy also verifies their effectiveness. Finally, the baseline was reduced by 3.29%, 4.14%, and 2.06% on the reasonable all-day, day, and night subsets via applying the GSMA module and MS strategy which obtained the best performance.

5 Conclusions

In this paper, we propose a novel multispectral object detection network named SAMS-YOLO, which can effectively improve multi-modal detection accuracy while maintaining high efficiency. Particularly, we design a group shuffled multi-receptive attention module to fully extract and combine multi-scale RGB-T features and promote deeper multi-modal feature fusion. In addition, we propose a multi-modal supervision strategy to guide the network in learning more accurate and robust feature representations, as well as improving object detection. Comprehensive comparison and ablation experiments on KAIST and DroneVehicle datasets demonstrate the effectiveness of the proposed framework and its components. The proposed method can be applied to unmanned driving, video surveillance, and other RGB-T object detection domains.

Acknowledgements. This work was supported by Natural Science Basic Research Program of Shaanxi (2024JC-YBQN-0719), Key R & D Program of Shaanxi province, China (2023-YBGY-012), National Key Research and Development Program of China (2023YFC3209304, 2023YFC3209305), Shandong Excellent Young Scientists Fund Program (Overseas, Grant number: 2023HWYQ-114).

References

1. Alldieck, T., Bahnsen, C.H., Moeslund, T.B.: Context-aware fusion of RGB and thermal imagery for traffic monitoring. *Sensors* **16**(11), 1947 (2016)
2. Brazil, G., Yin, X., Liu, X.: Illuminating pedestrians via simultaneous detection & segmentation. In: *ICCV*, pp. 4950–4959 (2017)
3. Cao, Y., Bin, J., Hamari, J., Blasch, E., Liu, Z.: Multimodal object detection by channel switching and spatial attention. In: *CVPR*, pp. 403–411 (2023)
4. Cao, Y., Xu, J., Lin, S., Wei, F., Hu, H.: GCNet: non-local networks meet squeeze-excitation networks and beyond. In: *ICCV Workshops* (2019)
5. Dasgupta, K., Das, A., Das, S., Bhattacharya, U., Yogamani, S.: Spatio-contextual deep network-based multimodal pedestrian detection for autonomous driving. *TITS* **23**(9), 15940–15950 (2022)
6. Ding, J., Xue, N., Long, Y., Xia, G.S., Lu, Q.: Learning ROI transformer for oriented object detection in aerial images. In: *CVPR*, pp. 2849–2858 (2019)
7. 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)
8. Hwang, S., Park, J., Kim, N., Choi, Y., So Kweon, I.: Multispectral pedestrian detection: benchmark dataset and baseline. In: *CVPR*, pp. 1037–1045 (2015)
9. Jocher, G.: YOLOv5 release v6.1 (2020). <https://github.com/ultralytics/yolov5/releases/tag/v6.1>
10. Kim, J.U., Park, S., Ro, Y.M.: Uncertainty-guided cross-modal learning for robust multispectral pedestrian detection. *IEEE Trans. Circuits Syst. Video Technol.* **32**(3), 1510–1523 (2021)
11. Li, C., Song, D., Tong, R., Tang, M.: Multispectral pedestrian detection via simultaneous detection and segmentation. arXiv preprint [arXiv:1808.04818](https://arxiv.org/abs/1808.04818) (2018)
12. Li, C., Song, D., Tong, R., Tang, M.: Illumination-aware faster R-CNN for robust multispectral pedestrian detection. *Pattern Recogn.* **85**, 161–171 (2019)
13. Li, Q., Zhang, C., Hu, Q., Fu, H., Zhu, P.: Confidence-aware fusion using Dempster-Shafer theory for multispectral pedestrian detection. *IEEE Trans. Multimed.* (2022)
14. Li, R., Xiang, J., Sun, F., Yuan, Y., Yuan, L., Gou, S.: Multiscale cross-modal homogeneity enhancement and confidence-aware fusion for multispectral pedestrian detection. *IEEE Trans. Multimed.* **26**, 852–863 (2024)
15. Liu, J., Zhang, S., Wang, S., Metaxas, D.N.: Multispectral deep neural networks for pedestrian detection. arXiv preprint [arXiv:1611.02644](https://arxiv.org/abs/1611.02644) (2016)
16. Liu, S., Qi, L., Qin, H., Shi, J., Jia, J.: Path aggregation network for instance segmentation. In: *CVPR*, pp. 8759–8768 (2018)
17. Qingyun, F., Zhaokui, W.: Cross-modality attentive feature fusion for object detection in multispectral remote sensing imagery. *Pattern Recogn.* **130**, 108786 (2022)
18. 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)
19. Sun, Y., Cao, B., Zhu, P., Hu, Q.: Drone-based RGB-infrared cross-modality vehicle detection via uncertainty-aware learning. *IEEE Trans. Circuits Syst. Video Technol.* **32**(10), 6700–6713 (2022)
20. Tian, C., Zhou, Z., Huang, Y., Li, G., He, Z.: Cross-modality proposal-guided feature mining for unregistered RGB-thermal pedestrian detection. *IEEE Trans. Multimed.* **26**, 6449–6461 (2024)

21. Wang, D., et al.: Advancing plain vision transformer toward remote sensing foundation model. *TGARS* **61**, 1–15 (2022)
22. Woo, S., Park, J., Lee, J.Y., Kweon, I.S.: CBAM: convolutional block attention module. In: *ECCV*, pp. 3–19 (2018)
23. Xiao, Y., Codevilla, F., Gurram, A., Urfalioglu, O., López, A.M.: Multimodal end-to-end autonomous driving. *TITS* **23**(1), 537–547 (2020)
24. Xie, J., et al.: Learning a dynamic cross-modal network for multispectral pedestrian detection. In: *Proceedings of the 30th ACM International Conference on Multimedia*, pp. 4043–4052 (2022)
25. Xie, X., Cheng, G., Wang, J., Yao, X., Han, J.: Oriented R-CNN for object detection. In: *ICCV*, pp. 3520–3529 (2021)
26. Yang, X., Qian, Y., Zhu, H., Wang, C., Yang, M.: BAANet: learning bi-directional adaptive attention gates for multispectral pedestrian detection. In: *ICRA*, pp. 2920–2926. *IEEE* (2022)
27. Yang, X., Yan, J.: Arbitrary-oriented object detection with circular smooth label. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) *ECCV 2020*. LNCS, vol. 12353, pp. 677–694. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58598-3_40
28. Yuan, M., Wang, Y., Wei, X.: Translation, scale and rotation: cross-modal alignment meets RGB-infrared vehicle detection. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) *ECCV 2022*. LNCS, vol. 13669, pp. 509–525. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-20077-9_30
29. Yuan, M., Wei, X.: C²former: calibrated and complementary transformer for RGB-infrared object detection. *TGARS* (2024)
30. Zhang, H., Zu, K., Lu, J., Zou, Y., Meng, D.: EPSANet: an efficient pyramid squeeze attention block on convolutional neural network. In: *Proceedings of the Asian Conference on Computer Vision*, pp. 1161–1177 (2022)
31. Zhang, L., et al.: Cross-modality interactive attention network for multispectral pedestrian detection. *Inf. Fusion* **50**, 20–29 (2019)
32. Zhang, L., Zhu, X., Chen, X., Yang, X., Lei, Z., Liu, Z.: Weakly aligned cross-modal learning for multispectral pedestrian detection. In: *ICCV*, pp. 5127–5137 (2019)
33. Zhang, N., Liu, Y., Liu, H., Tian, T., Ma, J., Tian, J.: DTNet: a specialized dual-tuning network for infrared vehicle detection in aerial images. *IEEE Trans. Geosci. Remote Sens.* (2024)
34. Zhang, N., Liu, Y., Liu, H., Tian, T., Tian, J.: Oriented infrared vehicle detection in aerial images via mining frequency and semantic information. *TGARS* (2023)
35. Zhang, X., Zhou, X., Lin, M., Sun, J.: Shufflenet: an extremely efficient convolutional neural network for mobile devices. In: *CVPR*, pp. 6848–6856 (2018)
36. Zhang, X., Li, Y., Qi, Z., Sun, Y., Zhang, Y.: Learning multi-domain feature relation for visible and long-wave infrared image patch matching (2023)
37. Zheng, Y., Blasch, E., Liu, Z.: *Multispectral Image Fusion and Colorization*, vol. 481. SPIE Press Bellingham, Washington (2018)
38. Zhou, K., Chen, L., Cao, X.: Improving multispectral pedestrian detection by addressing modality imbalance problems. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) *ECCV 2020*. LNCS, vol. 12363, pp. 787–803. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58523-5_46



Enhancing Object Detection by Leveraging Large Language Models for Contextual Knowledge

Amirreza Rouhi^(✉), Diego Patiño, and David K. Han

Drexel University, Philadelphia, PA 19104, USA
{ar3755, dp3324, dkh42}@drexel.edu

Abstract. The adoption of deep learning-based object detection models has proliferated across numerous applications. However, their efficacy is significantly constrained under challenging imaging conditions like fog or occlusion. In response to these limitations, we present a novel approach that transcends these hurdles by exploiting scene contextual knowledge distilled from Large Language Models (LLMs). This methodology empowers our model to deduce and anticipate object presence within a scene by leveraging contextual knowledge akin to human perception, thereby overcoming the constraints of direct visual cues. Our method synergizes the capabilities of object detection models with the contextual interpretation and predictive capacity of LLaMA, an advanced LLM. Our framework operates exclusively on the labels and positional information provided by a detection algorithm, sidestepping the reliance on pixel-level image data both during training and inference. The effectiveness of our approach is validated through extensive experiments conducted on the COCO-2017 dataset, including a modified version simulating reduced visibility conditions. The empirical findings underscore the superior performance of our integrated model compared to standalone YOLO models, particularly evident in adverse conditions, where notable enhancements in detection accuracy are observed across various object sizes.

Keywords: Object Detection · Scene Understanding · Deep Learning · Large Language Models (LLMs)

1 Introduction

While state-of-the-art computer vision algorithms have demonstrated remarkable proficiency in recognizing diverse objects and their locations, it is widely

This work was possible by the support provided by the Bruce Eisenstein Endowment Funds.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-78447-7_20.

recognized that their performance often falters when tasked with images afflicted by poor lighting, obscurants like fog, or occlusions. In contrast, humans excel in locating objects within such images by leveraging contextual knowledge, extending their perception beyond what is visible. One such example is driving a vehicle on dimly lit, foggy, or rainy roads at night. Visibility in such conditions is significantly poorer compared to driving in clear daylight. Yet, many drivers maneuver through these challenging circumstances, detecting and evading potential hazards without additional aid, reaching their destination with little trouble.

The use of contextual knowledge to infer and predict, even with limited visual data, is a unique aspect of human perception that current computer vision systems do not possess. Our work aims to address this deficiency by integrating contextual information from transformer-based deep learning architectures such as Large Language Models (LLMs). The ability to capture long-ranging dependencies and extract contextual information has made transformers critical in natural language processing tasks. The contextual understanding ability of Transformer architectures (TAs) comes from the self-attention mechanism, which enables these models to determine the relative importance of other words in the input text with respect to any given word. Under this, the transformer has a mechanism to capture contextual information from the whole input sequence rather than just exploiting local context.

In this paper, we develop a new technique that strengthens object detection algorithms in challenging situations by embedding them with Large Language Models (LLMs), with an emphasis on contextual scene understanding. We start by employing an object detector, such as YOLO [10, 11, 21], to identify and locate objects within a visually challenging scene. The object detector produces a list of objects along with their bounding boxes, accompanied by a range of confidence scores. From the objects detected with high confidence, we assert that a scene context can be established. Our aim is to predict the remaining objects present in the scene, likely detected with lower confidence. To achieve this, we task the Large Language Model (LLM) to generate a list of objects anticipated to occupy the bounding box locations of low detection confidence, guided by the context established from the detected objects of high confidence. If any object in the generated list aligns with those initially detected by the YOLO detector with low confidence, it is deemed as a detected object. Conversely, it is disregarded. In summary, our approach restricts the sample space of the objects in the image by placing a contextual condition on possible objects not fully detected by the pixel based object detector.

In Fig. 1, we exemplify the benefits of augmenting current state-of-the-art (SotA) object detectors with the aid of LLM. The left image displays objects detected by YOLOv8 [10]. However, due to image distortion caused by raindrops, YOLOv8 fails to detect the **Car** in the bottom-right corner and one of the **Traffic Lights** in the upper left quadrant. In contrast, our integrated approach accurately finds these missed objects, as depicted in the image on the right. By combining visual cues with contextual knowledge, we posit that object detection, even in challenging scenes, can be significantly enhanced.



Fig. 1. An example of qualitative results using YOLOv8 on the COCO-2017 dataset: (left) object detection using an out-of-the-box YoloV8 model, where YOLOv8 fails to detect the *Car* in the bottom-right corner and one of the *Traffic Lights* in the upper left quadrant. (right) YOLOv8 enhanced with LLaMA 2_{pt}, where our integrated approach accurately finds these missed objects. The green boxes in the right image highlight the correctly detected objects which were missed by the standard YOLOv8 model. (Color figure online)

2 Related Work

The integration of language models with visual object detection has been explored extensively in the literature, focusing on enhancing visual recognition through contextual understanding and multi-modal learning. Our review here aims to highlight these approaches, emphasizing their limitations and how our proposed method addresses these shortcomings.

2.1 Visual Learning Using Language

Textual information integration to the visualization process has greatly improved object detection approaches by incorporating image and text attributes. For example, ImageBERT [19] and ViLT [13] employ the complementarity of the context to enhance the performance of the systems regarding various scenes. These developments have been important for approaches such as Flamingo [1] and SIMVLM [25] that utilize vision and text to improve scene understanding and object recognition respectively. Some recent models including CLIP [20], which applies contrastive learning to match image captions with the corresponding images, have slightly shifted focus onto zero shot learning, but have primarily focused on matching attributions. This approach expanded the capability of the system to detect objects not included in the training corpus through the translation of textual prompt into classifiers [24]. However, these models are not precisely optimized for the cases where visual data are smeared by low illumination or occlusions. Recent methods have made use of the open-ended vocabulary-matching power of CLIP to integrate attribute data via descriptions created

by large language models (LLMs) [17, 18]. However, these models often struggle with attribute recognition and contextual understanding in complex scenes.

Despite these advancements, our approach is unique in that it leverages LLMs to enhance object detection specifically by providing contextual knowledge for low-confidence predictions. Unlike previous methods that focus on zero-shot learning and attribute integration, our method directly improves detection accuracy under challenging conditions by utilizing the contextual understanding of LLMs.

2.2 Contextual Knowledge in Object Detection

Recent advances in object detection are increasingly incorporating contextual understanding by resorting to transformer models and Large Language Models [23, 29]. However, most of such methods, although very creative, have limitations addressed by our approach.

Among these, Zang et al. [29] proposed ContextDET, which uses Multimodal Large Language Models for contextual object detection. Although this interface between the human operators and AI systems is vastly improved when ContextDET connects visual objects to linguistic cues, it does not really imbibe the full potential of LLMs in improving detection accuracy within a spatially complex or ambiguous setting. Additionally, they also fed the image features to the LLM, which increases the complexity of the model.

In 2020, Ilharco et al. [9] examined the language alignment with visual representations of concrete nouns. Since their findings show that the visual-semantic representation is compatible with text data, the motivation from their work is more towards model selection rather than improving operation performance in object detection. Our approach actively uses this finding to exploit improvements in the prediction at hand from object detection systems with the addition of context, explicitly from the LLMs, in interpreting the diversity within complex scenes. More recently, Xue et al. [27] constructed DIAG-TR, which employs a dual network structure to extract global and local features from transformers. DIAG-TR was verified on remote sensing image datasets, where it demonstrated the value of feature hierarchies but was still limited to the specificity of different kinds of images.

Large image-caption datasets and text embeddings have enabled the upsurge of methods with easier supervision techniques and cheaper vocabulary expansion [3, 12, 14, 31], while the generation of large image-caption datasets and weakly supervised techniques have made cheaper supervision approaches [6, 28]. Similarly, open-vocabulary detection methods [2, 7, 8, 26, 30] generalize the functionalities of object detection systems, enabling them to identify and label a vast number of objects beyond those included in the training set. This expansion significantly broadens the scope of detectable objects. Methods along both lines oftentimes pay the price for the varying quality of image-caption pairs, a common issue in real-life scenarios.

Bravo et al. [4] demonstrated an open-vocabulary detection method based on matching image-caption pairs. Although that was useful in detecting new

objects, its applications in real-life scenarios get restricted due to the availability and quality of the data. We circumvent these limitations by using contextual information gathered from LLMs to enhance detection at no large computational cost traditionally involved in processing image datasets of this scale.

Most object detection techniques make heavy use of fine-grained visual inputs, thereby limiting performances in scenarios when objects could be perceived as highly occluded, blurred, or tiny. These are some of the factors that lower the area of clear visual information, making it hard to perform object detection. Moreover, the direct use of LLMs in processing image data may bring about immense computational loads, more so in real-time applications. Such a method is very resource-intensive, as image data is large in size and complex in nature, hence hampering the real-world implementation of such a system where responses are expected in real-time.

3 Proposed Approach

To address the limitations of current object detection models under challenging conditions, we propose a novel method that integrates YOLO with Large Language Models (LLMs), enhancing detection through contextual knowledge. Our approach begins by detecting visual objects within a scene using the YOLO model. Following the detection, we extract and segregate labels and bounding box data from the output, categorizing them into high and low-confidence groups based on their scores. For the detections marked with low confidence, we apply the LLaMA 2 model [22], leveraging its advanced contextual comprehension capabilities, to validate whether these labels fit with the context established by the detected objects of high confidence. The final step of our method involves aligning the original detection results with these predictions from LLaMA 2, integrating the enhanced label predictions to refine and improve the overall detection results. Figure 2 illustrates the architecture of our proposed approach. In this section, we describe our approach’s workflow in detail and all the specific processes involved in each step of our proposed methodology, including the computational techniques and the integration mechanism between YOLO and LLaMA 2.

3.1 Object Detection with YOLO

Using the YOLO algorithm, an input image I is analyzed to detect a set of objects \mathcal{O} , with each object $o_i \in \mathcal{O}$ defined by a bounding box $b_i = (x_i, y_i, w_i, h_i)$ and a confidence score c_i . Objects are then classified based on their confidence scores into two categories: high and low confidence. Objects in the high category are all elements of \mathcal{O} with their confidence score greater than a threshold t_{high} , following

$$\mathcal{O}_{\text{high}} = \{o_i \mid c_i > t_{\text{high}}\}.$$

Similarly, the detected objects are categorized into the low confidence category following

$$\mathcal{O}_{\text{low}} = \{o_i \mid t_{\text{low}} < c_i \leq t_{\text{high}}\},$$

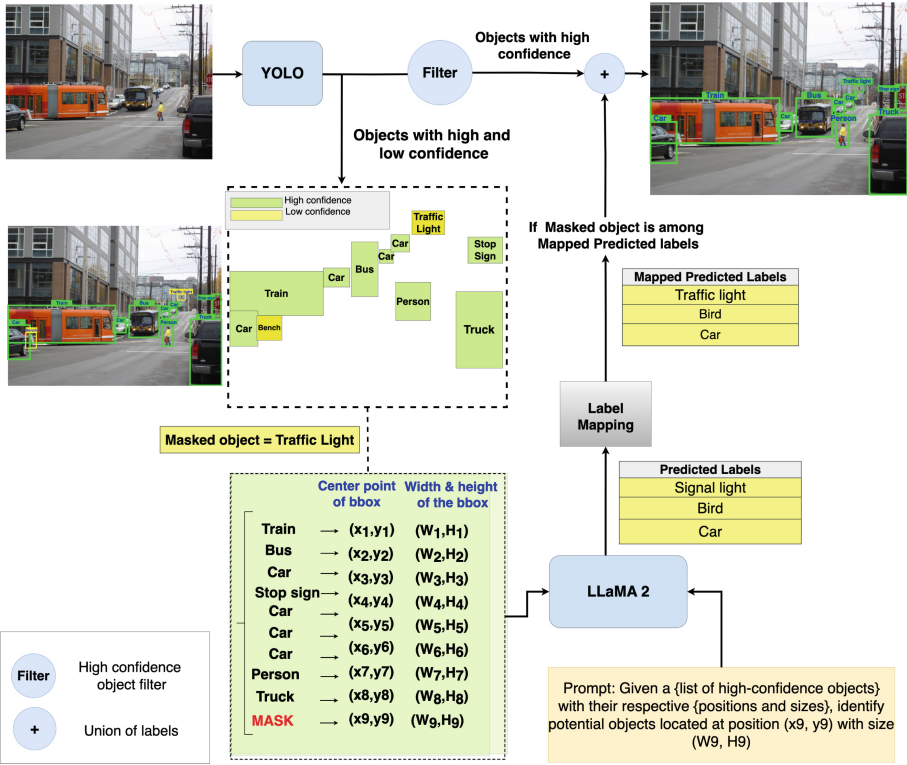


Fig. 2. Pipeline illustration showing the integration of YOLO and LLaMA 2 for enhanced object detection. The image demonstrates the detection of **Traffic Light** and **Bench** objects with low confidence by YOLO. These objects are then masked and passed into LLaMA 2, along with high-confidence objects. LLaMA 2 predicts potential objects for the given location and size. The process includes a mapping step to align LLaMA 2 predictions with dataset objects. If the masked label (e.g., **Traffic Light**) is among the mapped predictions from LLaMA 2, we add it to the list of detected objects; otherwise, we disregard it. Note that the process is repeated for each low-confidence object. The “Bench” object is input to the model in the next round.

where we set $t_{high} = 0.25$ and $t_{low} = 0.1$. The t_{high} value matches YOLO’s default detection threshold. Objects with a confidence level below t_{low} are not considered successful detections; thus, we ignore them.

3.2 Integration with LLaMA

In our methodology, for each object detected with a low confidence score, we mask out its label while retaining its bounding box information. This information forms part of the input prompt that we feed to the LLaMA model. The prompt includes the names and bounding boxes of objects detected with high confidence, providing a contextual backdrop for the model, see Fig. 2. To prepare the data for

this prompt, we use a function f that specifically formats the bounding box and high-confidence object data. Then, we use the prompt as the complete query we send to the LLaMA model. This process is repeated individually for each object in the low confidence set \mathcal{O}_{low} . Mathematically, the prompt function and the prediction process for each low-confidence object are represented as follows:

$$q_{\text{masked}} = f(b_{\text{masked}}, \mathcal{O}_{\text{high}}, \mathcal{B}_{\text{high}}), \quad \forall o_{\text{masked}} \in \mathcal{O}_{\text{low}} \quad (1)$$

$$\mathcal{O}_{\text{predicted}} = \text{LLaMA}(q_{\text{masked}}), \quad (2)$$

where the function f is responsible for preparing the data by structuring the bounding box information and details of high-confidence detected objects into a prompt format q_{masked} suitable for the LLaMA model. The LLaMA model aims to predict the identity of an object detected with low confidence, referred to as o_{masked} . The variable o_{masked} is an object selected from the low-confidence group \mathcal{O}_{low} and its bounding box b_{masked} provides its spatial position and size. The context for these predictions is enriched by $\mathcal{O}_{\text{high}}$ and $\mathcal{B}_{\text{high}}$, which represent the objects detected with high confidence and their corresponding bounding boxes. Finally, $\mathcal{O}_{\text{predicted}}$ comprises the set of possible object labels that LLaMA predicts for the position and context of o_{masked} , representing the model’s best guess based on the provided prompt.

This streamlined process ensures that LLaMA uses both the location and contextual clues from high-confidence detections to enhance the accuracy of low-confidence object identification.

3.3 Word Embedding and Mapping

One issue of simply combining Large Language Models like LLaMA with object detection systems stems from the disparity between the set of words in the object detection dataset and the vocabularies contained in LLaMA. Typically, the object detection dataset is comprised of a vastly smaller set of words compared to the extensive vocabulary of LLaMA. The main goal is to align LLaMA’s natural language predictions with the predefined object categories implemented in our dataset, such as the COCO label dataset. To bridge this misalignment, we employed a word embedding approach utilizing a Word2Vec model. This approach helps determine the closest semantic match between the output categories of visual detector and LLaMA. Using the Word2Vec [5] model, we transform both the predicted labels and labels of the COCO dataset into high-dimensional vectors. We subsequently compute the cosine similarity between the predicted label vector and vectors of the COCO labels. The label from the COCO dataset that exhibits the highest similarity to the predicted label is selected as the best match and included in the set of mapped labels $\mathcal{O}_{\text{mapped}}$, provided that the cosine similarity exceeds a threshold of 0.5. This ensures that the matched labels share a significant semantic similarity, thus avoiding incorrect or arbitrary matches.

$$\mathcal{O}_{\text{mapped}} = \{\text{Match}(o_{\text{predicted}}) \mid o_{\text{predicted}} \in \mathcal{O}_{\text{predicted}}\} \quad (3)$$

with

$$\text{Match}(o_{\text{predicted}}) = \underset{l \in L_{\text{COCO}}}{\text{argmax}} \text{Sc}(V(o_{\text{predicted}}), V(l)), \quad \text{if } \text{Sc} > 0.5 \quad (4)$$

In Eq. 3 and 4, $\mathcal{O}_{\text{mapped}}$ represents the set of mapped objects, $o_{\text{predicted}}$ is a member of the predicted object labels set $\mathcal{O}_{\text{predicted}}$, L_{COCO} denotes all possible labels within the COCO dataset, $V(\mathcal{O}_{\text{predicted}})$ and $V(l)$ are the vector representations of the predicted label and the COCO label obtained through Word2Vec word embedding, and Sc is the cosine similarity between two vector embeddings.

This embedding and mapping step is crucial for developing stable object detection. It avoids the removal of an instance just because LLaMa forecasted a synonym-such as **Vehicle** when the visual detector output label was **Car**. This alignment is particularly crucial in the later stages when we compare the outputs of YOLO and LLaMA’s predictions.

3.4 Alignment and Inclusion of Detected Objects

After applying the mapping on LLaMA’s predictions, we check whether o_{masked} is suitable to be added to the final set of detections, according to the rule in Eq. 5. If o_{masked} is among any of the top three mapped predictions of LLaMA, we validate it and consider o_{masked} as a detected object; otherwise, we ignore it.

$$\text{is_valid}(o_{\text{masked}}) = \begin{cases} \text{true} & \text{if } o_{\text{masked}} \in \mathcal{O}_{\text{mapped}} \\ \text{false} & \text{otherwise} \end{cases} \quad (5)$$

When it passes this alignment, we include an object in the final set of detections such that

$$\mathcal{O}_{\text{final}} = \mathcal{O}_{\text{high}} \cup \{o_{\text{masked}} \mid \text{is_valid}(o_{\text{masked}})\} \quad (6)$$

Figure 2 demonstrates the enhancement of a low-confidence detection, specifically a **Traffic Light**, within an urban street scene analyzed by the YOLOv8 algorithm. The initial detection classifies the **Traffic Light** with a lower-than-desired confidence score, designating it as o_{masked} .

The LLaMA model, upon receiving the spatial and size details of o_{masked} alongside the high-confidence object data $\mathcal{C}_{\text{high}}$, provides a list of three potential object labels for the given location and size: **Signal Light**, **Bird**, and **Car**. We apply word embedding transformations to semantically match these predictions with standard object categories and calculate their cosine similarities with the COCO dataset labels. This process yields a corresponding list of COCO dataset classes: **Traffic Light**, **Bird**, and **Car**.

In the final alignment step, we confirm whether the original masked object o_{masked} , which YOLOv8 detected as **Traffic Light**, is included in the list of mapped predictions of LLaMA. Given that **Traffic Light** is present, we conclude that LLaMA’s context-aware prediction aligns with the initial detection,

and thus, we incorporate the **Traffic Light** into the final set of detected objects $\mathcal{O}_{\text{final}}$.

This case underscores the strength of our methodology in leveraging the capabilities of a Large Language Model to enhance the precision and confidence of object detection in complex, real-world scenarios.

4 Experimental Results

4.1 Datasets and Evaluation Metrics

We evaluate our method on the validation set of the well-known **COCO-2017 dataset** [15]. This dataset consists of 118k training images and 5k validation images spanning across various object sizes. The dataset classifies objects into small (S) when the objects are less than 32×32 pixels, medium (M) when they are between 32×32 and 96×96 pixels, and large (L) when they are greater than 96×96 pixels. This classification is based on the object’s bounding box area.

Moreover, We have modified the COCO 2017 dataset to create **COCO-2017-Blurred**, simulating real-world scenarios with reduced visibility. This variation tests model performance in challenging conditions. In this variant, one-third of the objects are randomly blurred using a 21×21 Gaussian kernel, increasing the difficulty for object detection systems.

4.2 Methodology

We conducted comprehensive experiments to evaluate the performance of various YOLO models integrated with LLaMA 2, assessing their ability to detect objects of different sizes with precision. The YOLO models tested in our experiments included YOLOv3, YOLOv7, and YOLOv8. For the integration with LLaMA 2, we utilized two specific versions of the model: LLaMA 2_{pt}, which is the pre-trained version, and LLaMA 2_{ft}, which is a fine-tuned version using the LoRA technique specifically adapted to the COCO dataset’s labels and bounding box information.

4.3 Results

Table 1 summarizes the AP scores for each configuration on the standard COCO-2017 and the COCO-2017 Blurred datasets. The results indicate a consistent improvement in object detection across all sizes when YOLO models are augmented with LLaMA. Notably, the fine-tuned LLaMA versions generally outperformed the pre-trained ones, emphasizing the value of tailoring the LLM to the specific dataset.

A significant aspect to highlight is that the pre-trained model, LLaMA2_{pt}, was not trained on the experimental COCO dataset. This indicates that the model’s capabilities in enhancing object detection are robust and generalizable, as it can effectively apply learned contextual knowledge from different datasets to improve detection accuracy in unseen environments.

We report further experiments with YOLO models whose confidence threshold is set below the default values to prove that improvements in mean Average Precision (mAP) were due to Large Language Models’ integration and not just due to the lowering of the confidence threshold. We designed these experiments to specifically compare the performance of the models under diminished confidence thresholds of YOLO models against those enhanced by LLMs at close threshold conditions. The approach will help isolate the effect of LLM integration from other potential confounding factors due to threshold manipulation, hence providing a clearer insight into the real value added by incorporating contextual knowledge from LLMs into the object detection process.

Table 1. Enhancement in Object Detection Across Various Sizes using YOLO and LLaMA Integration: The table compares the Average Precision (AP) for small, medium, and large objects across the standard COCO-2017 and its altered counterpart, COCO-2017 Blurred. The metrics are reported for IoU thresholds from 0.5 to 0.95, highlighting improvements when YOLO models are combined with both pre-trained and fine-tuned LLaMA 2 models.

	COCO-2017			COCO-2017 Blurred		
	AP-S	AP-M	AP-L	AP-S	AP-M	AP-L
YOLOv3 $_{Th=0.25}^*$	0.21	0.42	0.49	0.06	0.18	0.23
YOLOv3 $_{Th=0.1}$	0.19	0.41	0.47	0.05	0.16	0.20
YOLOv3 + LLaMA2 $_{pt}$	0.28	0.48	0.54	0.11	0.24	0.28
YOLOv3 + LLaMA2 $_{ft}$	0.30	0.51	0.54	0.17	0.26	0.30
YOLOv7 $_{Th=0.001}^*$	0.35	0.55	0.66	0.10	0.26	0.41
YOLOv7 $_{Th=0.0001}$	0.34	0.55	0.66	0.08	0.25	0.40
YOLOv7 + LLaMA2 $_{pt}$	0.37	0.57	0.70	0.19	0.29	0.44
Yolov7 + LLaMA2 $_{ft}$	0.39	0.57	0.71	0.20	0.30	0.44
YOLOv8 $_{Th=0.25}^*$	0.36	0.59	0.70	0.11	0.27	0.50
YOLOv8 $_{Th=0.1}$	0.325	0.56	0.67	0.07	0.24	0.48
YOLOv8 + LLaMA2 $_{pt}$	0.39	0.61	0.71	0.20	0.30	0.52
YOLOv8 + LLaMA2 $_{ft}$	0.40	0.61	0.72	0.20	0.31	0.52

Models marked with * use the default threshold value as presented in the original paper.

The enhanced performance is especially pronounced in the COCO-2017 Blurred dataset, underlining the proposed method’s robustness against reduced visibility conditions. The fine-tuned LLaMA models exhibit superior Average Precision (AP) scores, indicating that the supplementary contextual knowledge gained through the fine-tuning process proves advantageous in challenging detection scenarios.

Figures 3 and 1 showcase visual comparisons highlighting the enhanced object detection capabilities achieved by integrating YOLOv8 with LLaMA 2’s cognitive processing. This integration markedly improves the detection of small

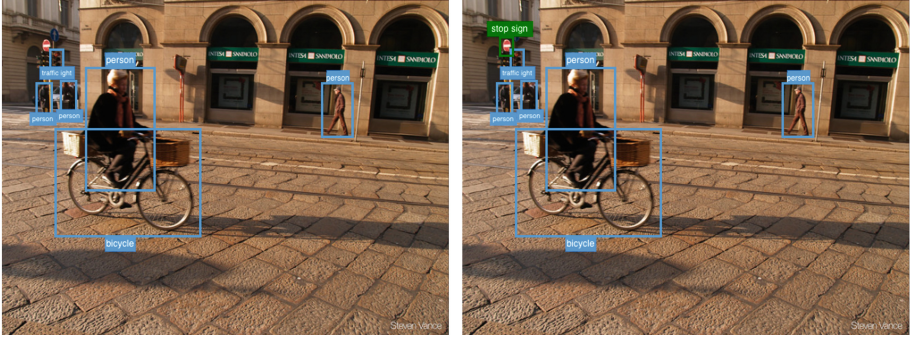


Fig. 3. Comparison of the object detection outcomes using the YOLOv8 with and without using an LLM on a sample from the COCO-2017 dataset [16]. The left image showcases detection with YOLOv8 only, while the right image demonstrates YOLOv8 [10] with LLM. Notice how YOLOv8 without the LLM fails to detect the **Stop Sign**, whereas the YOLOv8+LLM integration successfully accurately them.

or visually ambiguous objects. For example, as illustrated in Fig. 3, the model effectively finds the **Stop Sign**, while it was missed by the baseline YOLOv8. Figure 1 depicts a realistic scenario on a rainy day, where YOLOv8 alone struggles to detect a **Car** and a **Traffic Light** due to low detection confidence but succeeds when augmented with LLaMA 2.

For more qualitative examples illustrating the effectiveness of our integration of YOLOv8 with LLaMA 2_{pt}, please refer to Fig. 4 and the supplementary materials.

4.4 Ablation Study

The goal of this analysis is to compare the detection accuracy of the baseline YOLO models with their counterparts integrated with LLaMA 2 under standard and visually impaired conditions provided by the COCO-2017 and COCO-2017 Blurred datasets, respectively.

Performance on COCO-2017: The integration of YOLO with LLaMA 2, both pre-trained (LLaMA 2_{pt}) and fine-tuned (LLaMA 2_{ft}), significantly enhances detection accuracy for all object sizes on the COCO-2017 dataset, as evidenced by the improved AP scores in Table 1. Notably, there is a marked increase in AP for small objects, with the AP-small for YOLOv8+LLaMA 2_{ft} reaching **0.40**, a clear improvement over the baseline YOLOv8 model’s AP of 0.36. Medium and large objects also see commendable performance gains, underscoring the integrated models’ effectiveness over a range of object dimensions.

Performance on COCO-2017 Blurred: The COCO-2017 Blurred dataset introduces additional complexity to object detection tasks. However, the combined YOLO models and LLaMA 2 models demonstrate robust performance enhancements, particularly in the recognition of small objects. The integration

of LLaMA 2, both in its pre-trained and fine-tuned forms, with the YOLOv8 architecture results in a notable improvement in AP for small objects. Specifically, the AP-small metric increases to **0.20** for the integrated models from an AP of 0.11 for the standalone YOLOv8. This increase highlights LLaMA 2’s adeptness at employing contextual information effectively to bolster detection accuracy, even under challenging conditions where visibility is compromised.

Comparative Effectiveness of LLaMA Integration and Model Efficiency: Based on the results shown in Table 1, across all variants of YOLO (YOLOv5, YOLOv7, and YOLOv8), on both dataset, integrating LLaMA has consistently aided the models in detecting objects that were previously missed. This enhancement underscores the significant impact of incorporating contextual knowledge through LLMs on the object detection process. Moreover, we can see that the performance of the pre-trained LLaMA model (LLaMA2_{pt}) closely matches that of its fine-tuned counterpart (LLaMA2_{ft}) in many cases. Despite the expectation that fine-tuning would boost performance, the marginal improvements suggest the pre-trained LLaMA model already possesses a considerable degree of the necessary contextual knowledge for this task. This finding highlights the pre-trained model’s efficacy, indicating it is a robust choice for enhancing object detection without additional fine-tuning.

5 Limitations

While our approach has demonstrated significant improvements in object detection by leveraging contextual clues from Large Language Models, it is not without limitations. A key dependency of our method is the availability of a sufficient number of high-confidence detections to establish a robust scene context. In scenarios where the object detector yields few high-confidence detections, our model may struggle to generate accurate predictions for low-confidence objects due to insufficient contextual data. This limitation highlights the importance of having reliable initial detections and suggests that our method may be less effective in extremely challenging visual conditions where few objects are detected with high confidence.

Furthermore, adding an LLM to the model increases computational load to the overall process. The input to the LLM in our case, however, is purely text composed the names and bounding box information of the found objects. The computational load in our case, therefore, is significantly lower compared to those methods where the whole image is processed by LLM, like [29]. Nevertheless, we have to consider few parameters, such as maximum token count, which will be self-optimizing in an LLM for its performance feature. While the inclusion of an LLM does increase computational load, efforts in optimizing the integrated process and fine tuning would lead to many practical applications.

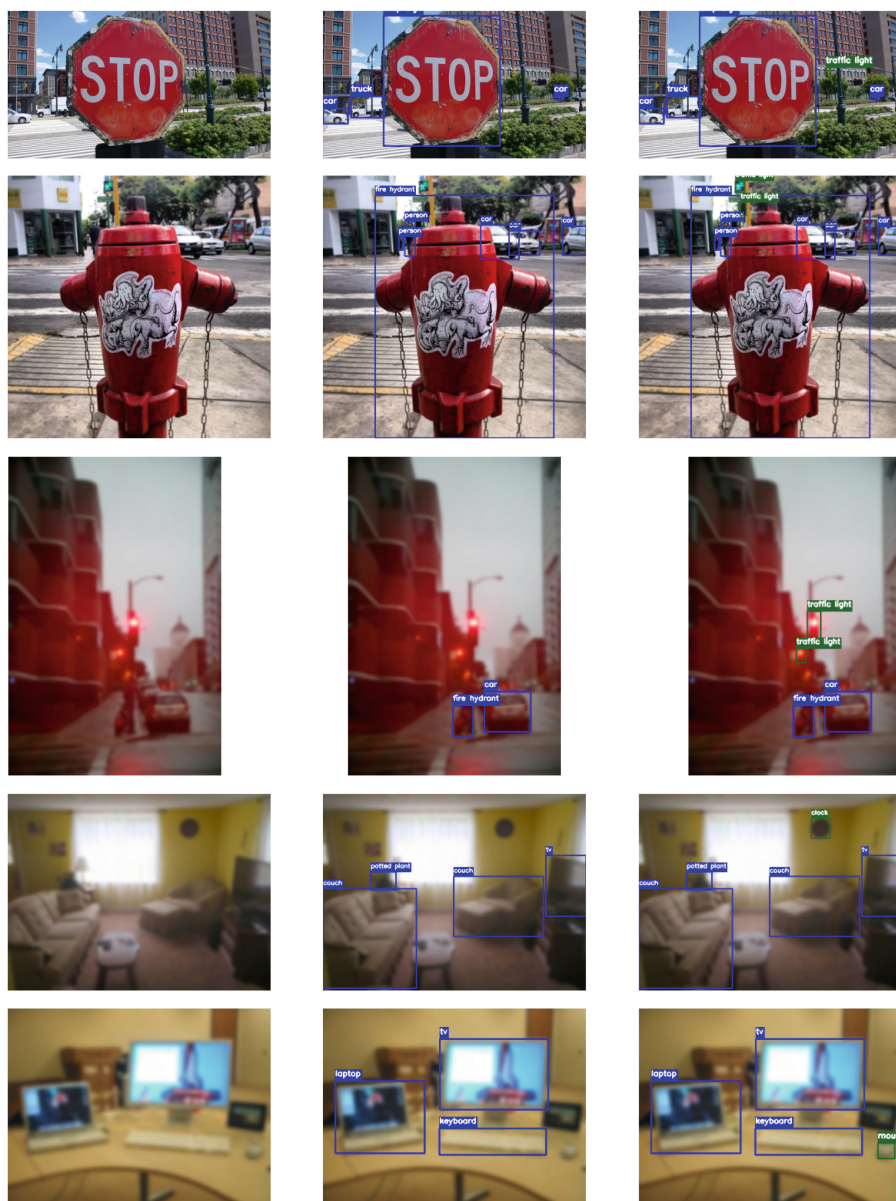


Fig. 4. Qualitative Result Samples from COCO-2017 (top three rows) and COCO-2017 Blurred (bottom three rows) Datasets: Original images are displayed on the left, detections by YOLOv8 without LLaMA 2_{pt} integration in the middle, and detections by YOLOv8 with LLaMA 2_{pt} integration on the right

6 Conclusion

In this work, we presented a novel strategy to augment the capabilities of state-of-the-art visual object detectors by integrating them with learn-based contextual knowledge models. Through this integration, our goal was to emulate the human ability to comprehend and interpret complex visual scenes, even under conditions of uncertainty or incomplete information. Our method achieves this by exploiting contextual knowledge from the scene. Our strategy shows an improvement in the detection performance of visual object detectors, such as YOLO, when paired with attention-based transformer architectures such as LLaMA 2.

Our experiments clearly showcased the benefits of exploiting scene contextual knowledge in object detection.

In conclusion, this paper aimed to demonstrate that by utilizing the contextual knowledge capabilities of Large Language Models (LLMs) alongside traditional object detection methods, we can enhance the performance of these detection models. This is achieved without the need to process entire images within the LLM, relying solely on textual information of objects. Thus, LLMs can significantly boost the efficacy of conventional object detection models without adding substantial complexity to the system.

References

1. Alayrac, J.B., et al.: Flamingo: a visual language model for few-shot learning. In: *Advances in Neural Information Processing Systems*, vol. 35, pp. 23716–23736 (2022)
2. Bangalath, H., Maaz, M., Khattak, M.U., Khan, S.H., Shahbaz Khan, F.: Bridging the gap between object and image-level representations for open-vocabulary detection. In: *Advances in Neural Information Processing Systems*, vol. 35, pp. 33781–33794 (2022)
3. Bansal, A., Sikka, K., Sharma, G., Chellappa, R., Divakaran, A.: Zero-shot object detection. In: *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 384–400 (2018)
4. Bravo, M.A., Mittal, S., Brox, T.: Localized vision-language matching for open-vocabulary object detection. In: Andres, B., Bernard, F., Cremers, D., Frintrop, S., Goldlücke, B., Ihrke, I. (eds.) *DAGM GCPR 2022*. LNCS, vol. 13485. Springer, Cham (2022)
5. Church, K.W.: Word2vec. *Nat. Lang. Eng.* **23**(1), 155–162 (2017)
6. Desai, K., Johnson, J.: Virtex: learning visual representations from textual annotations. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11162–11173 (2021)
7. Du, Y., Wei, F., Zhang, Z., Shi, M., Gao, Y., Li, G.: Learning to prompt for open-vocabulary object detection with vision-language model. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14084–14093 (2022)
8. Gao, M., et al.: Open vocabulary object detection with pseudo bounding-box labels. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) *ECCV 2022*. LNCS, vol. 13670, pp. 266–282. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-20080-9_16

9. Ilharco, G., Zellers, R., Farhadi, A., Hajishirzi, H.: Probing contextual language models for common ground with visual representations. arXiv preprint [arXiv:2005.00619](https://arxiv.org/abs/2005.00619) (2020)
10. Jocher, G., Chaurasia, A., Qiu, J.: YOLO by Ultralytics (2023). <https://github.com/ultralytics/ultralytics>
11. Jocher, G., et al.: Ultralytics/YOLOv5: v5. 0-YOLOv5-p6 1280 models, AWS, supervise. LY and Youtube integrations. Zenodo (2021)
12. 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)
13. Kim, W., Son, B., Kim, I.: ViLT: vision-and-language transformer without convolution or region supervision. In: International Conference on Machine Learning, pp. 5583–5594. PMLR (2021)
14. 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)
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. Loh, Y.P., Chan, C.S.: Getting to know low-light images with the exclusively dark dataset. *Comput. Vis. Image Underst.* **178**, 30–42 (2019). <https://doi.org/10.1016/j.cviu.2018.10.010>
17. 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)
18. Pratt, S., Covert, I., Liu, R., Farhadi, A.: What does a platypus look like? Generating customized prompts for zero-shot image classification. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 15691–15701 (2023)
19. Qi, D., Su, L., Song, J., Cui, E., Bharti, T., Sacheti, A.: ImageBERT: cross-modal pre-training with large-scale weak-supervised image-text data. arXiv preprint [arXiv:2001.07966](https://arxiv.org/abs/2001.07966) (2020)
20. Radford, A., et al.: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning, pp. 8748–8763. PMLR (2021)
21. 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)
22. Touvron, H., et al.: Llama 2: open foundation and fine-tuned chat models. arXiv preprint [arXiv:2307.09288](https://arxiv.org/abs/2307.09288) (2023)
23. Wang, W., et al.: VisionLLM: large language model is also an open-ended decoder for vision-centric tasks. In: Advances in Neural Information Processing Systems, vol. 36 (2024)
24. Wang, W., et al.: Image as a foreign language: beit pretraining for all vision and vision-language tasks. arXiv preprint [arXiv:2208.10442](https://arxiv.org/abs/2208.10442) (2022)
25. 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)
26. Wu, S., Zhang, W., Jin, S., Liu, W., Loy, C.C.: Aligning bag of regions for open-vocabulary object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 15254–15264 (2023)

27. Xue, J., He, D., Liu, M., Shi, Q.: Dual network structure with interweaved global-local feature hierarchy for transformer-based object detection in remote sensing image. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **15**, 6856–6866 (2022)
28. Ye, K., Zhang, M., Kovashka, A., Li, W., Qin, D., Berent, J.: Cap2det: learning to amplify weak caption supervision for object detection. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9686–9695 (2019)
29. Zang, Y., Li, W., Han, J., Zhou, K., Loy, C.C.: Contextual object detection with multimodal large language models. arXiv preprint [arXiv:2305.18279](https://arxiv.org/abs/2305.18279) (2023)
30. 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)
31. Zhang, H., et al.: Glipv2: unifying localization and vision-language understanding. In: *Advances in Neural Information Processing Systems*, vol. 35, pp. 36067–36080 (2022)



YOLO-RSOD: Improved YOLO Remote Sensing Object Detection

Yang Xu^{1,2,3} and Jun Lu^{1,2,3}(✉)

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

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. Remote sensing object detection has important application value in fields such as environmental monitoring and resource detection and analysis. However, the current universal object detectors are not very effective in detecting remote sensing objects. To this end, this paper proposes an efficient, low-complexity and anchor-free remote sensing object detection framework YOLO-RSOD based on YOLOv7. First, an additional Tiny Object Head is proposed for better detection of micro-remote sensing objects. The original Head is then replaced with Decoupled Head (DH) to explore the detection potential of the decoupled detection head structure. Then the Explicit Vision Center (EVC) in the Centralized Feature Pyramid Network (CFP) is added to further improve the detection ability of remote sensing objects. Finally, this article also integrates a global attention module (GAM) to find attention areas in dense object scenes. Ablation experiments on the general remote sensing target detection dataset VisDrone2021 demonstrate the effectiveness of several modules introduced in this paper in remote sensing target detection. On the VisDrone2021 data set, YOLO-RSOD can achieve accuracy rates of 30.7% AP50:95 and 51.7% AP50, which are 3.1% and 3.2% higher than the baseline model respectively.

Keywords: Object Detection · Remote Sensing Object Detection · Attention Mechanism · Feature Pyramid Networks · Decoupled Head

1 Introduction

With the increasing performance of computing hardware, deep neural network-based computer vision technology has been rapidly developing in the past decade. Object detection is an important part of computer vision technology [1], and remote sensing object detection is one of the most challenging tasks in the field of object detection. Currently, there are two mainstream object detection strategies. One is a two-stage strategy represented by the R-CNN family [2–5], and

the other is a one-stage strategy with YOLO [6] as one of the most popular frameworks. On common object detection datasets (MS COCO2017), models using the two-stage strategy perform somewhat better than those using the one-stage strategy. However, due to the inherent limitations of the two-stage framework, it is far from meeting the real-time requirements on traditional computing devices and is likely to face the same situation on most high-performance equipment. In contrast, one-stage object detectors can maintain a balance between real-time metrics and performance, and thus have received more attention from researchers. However, all current YOLO family models are designed and optimized based on general object detection, and no work has been done specifically for remote sensing object detection. Directly applying previous models to solve the remote sensing object detection task leads to three main problems, which are visually illustrated by some cases in Fig. 1.

In Fig. 1, row 1 illustrates the large variation in the size of remote sensing object images. Row 2 illustrates the high density characteristic of remote sensing objects, which usually results in occlusion between objects. Row 3 then illustrates that the coverage of remote sensing object is usually large and contains a wide variety of complex background information. The above three issues make remote sensing object detection very challenging.

In this paper, an improved model YOLO-RSOD is proposed based on the one-stage object detector YOLOv7, so as to solve the above three problems.

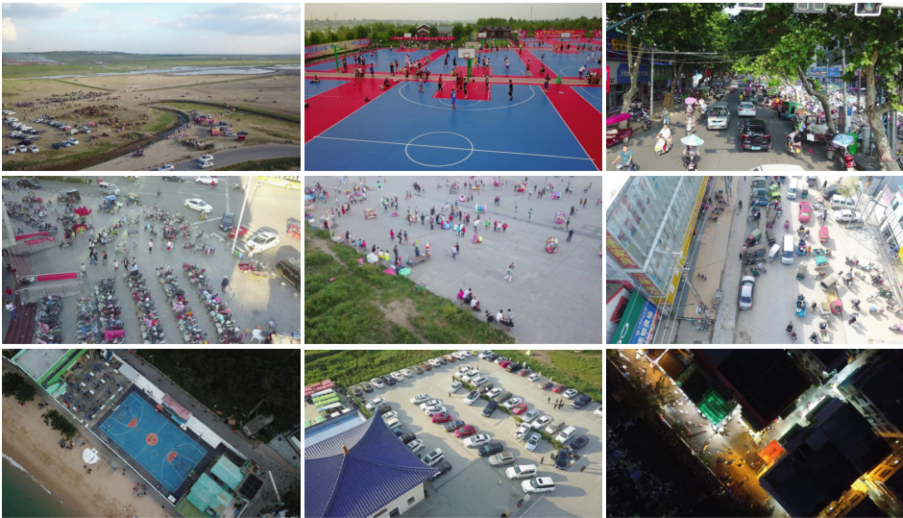


Fig. 1. Three main problems of object detection in remote sensing object images.

The contributions of this paper are as follows: (1) The remote sensing object detection framework YOLO-RSOD is proposed to deal with remote sensing objects.

(2) YOLO-RSOD integrates the Tiny Object Head, Decoupled-Idetect (DID), the Explicit Visual Center (EVC) in the Centralized Feature Pyramid Network (CFP), and the Global Attention Module (GAM). These modules can effectively improve the framework's detection effect on remote sensing objects. (3) On the VisDrone2021 dataset [7], the YOLO-RSOD proposed in this paper achieves 30.7% AP50:95 and 51.7% AP50, which is an improvement of 3.1% and 3.2% compared to the baseline model, respectively.

2 Related Work

Since the beginning of object detection research, the problem of remote sensing image detection has received widespread attention. As the proportion of objects in the image decreases, the pixel information used to express the objects also decreases. Large objects often occupy dozens or even hundreds of times more information than small objects, while the detection accuracy of small objects is often significantly lower than that of large objects. Therefore, the key difficulty of remote sensing object detection lies in improving the detection accuracy of small objects.

There have been many previous efforts aimed at improving small object detection performance. Some research has focused on optimizing the overall architecture of the YOLO series detector neck assembly. For example, J. Shang [8] replaced Neck in YOLO with weighted bidirectional feature pyramid multi BIFPN, and H. Liu [9] introduced a new feature fusion method PB-FPN in the neck of YOLO.

However, both methods choose to change the entire structure of the neck to achieve better feature fusion, which results in greater computational cost. On the contrary, this paper only uses a lightweight display vision center (EVC) [10] in the neck for feature integration. At the same time, this module can also fuse global and local information. This improvement aims to achieve less computational cost and Higher precision improves the performance of the YOLO neck.

In addition, some studies have tried to use the attention mechanism [11]. The attention mechanism can help the model better understand and process the structure and characteristics of the input data, so that the model can more accurately focus on the key parts of the image, such as the object and its surrounding area, thereby improving the accuracy and speed of object detection. There are many types of attention mechanisms, such as channel attention, spatial attention, temporal attention, branch attention, etc. Compared with these methods, YOLO-RSOD adds a global attention module (GAM) [9] mechanism in the transmission module between the neck and the head. The purpose is to reduce the computational cost, improve the model's ability to detect image occlusions that often exist in remote sensing images, and pay more attention to essential information when extracting features.

The conflict between classification and regression tasks is a well-known problem [12,13]. Therefore, decoupled heads for classification and localization are

widely used in most one-stage and two-stage detectors. However, with the continuous development of the backbone and feature pyramids of the YOLO series (e.g., FPN, PAN [12]), they generally adopt coupled detection heads. However, this design will lead to performance degradation, so this article adopts a decoupled structure in the detection head part, decomposing the original single detection head into two independent parts, one is responsible for the position information of the prediction box, and the other is responsible for the category information of the prediction box. This design improves the flexibility and generalization ability of the model while reducing the amount of calculation and memory consumption.

3 Approach

3.1 Baseline Model Selection

In the past few years, the YOLO series of models have become the most widely used and high-performing methods in the field of real-time object detection. Currently, the most commonly used YOLO series models include YOLOv5, YOLOv6, YOLOv7 and YOLOv8 methods. The performance comparison of these models on the general object detection data set COCO is shown in Fig. 2. The abscissa is the amount of model calculations, which directly affects the model running speed. The ordinate is the detection accuracy of the model in the COCO data set, reflecting the model performance. Taken together, the YOLOv7 model can achieve the best performance and computational cost balance compared to other models, so this article chooses YOLOv7 as the baseline.

3.2 YOLO-RSOD

In order to improve the performance of remote sensing object detection, this paper improves the original YOLOv7 model, thus forming the YOLO-RSOD model, and the framework of YOLO-RSOD is shown in Fig. 3.

In Fig. 3, the Backbone part adopts the structure of the original YOLOv7. The Neck part adds the EVC, while the GAM attention mechanism is introduced in the ELEN-H module and an additional Tiny Object Head is proposed. The Head part adopts a decoupled structure based on the head of the original structure.

Tiny Object Head. This paper studies and analyzes various remote sensing object datasets and finds that these remote sensing object datasets contain many very small instances. Therefore, this paper proposes a prediction head to predict remote sensing objects. This new prediction head is specifically used to detect tiny objects. It receives low-level, high-resolution feature maps as input. The newly added prediction head is combined in parallel with the original three prediction heads to form a four-head detection structure. During the training

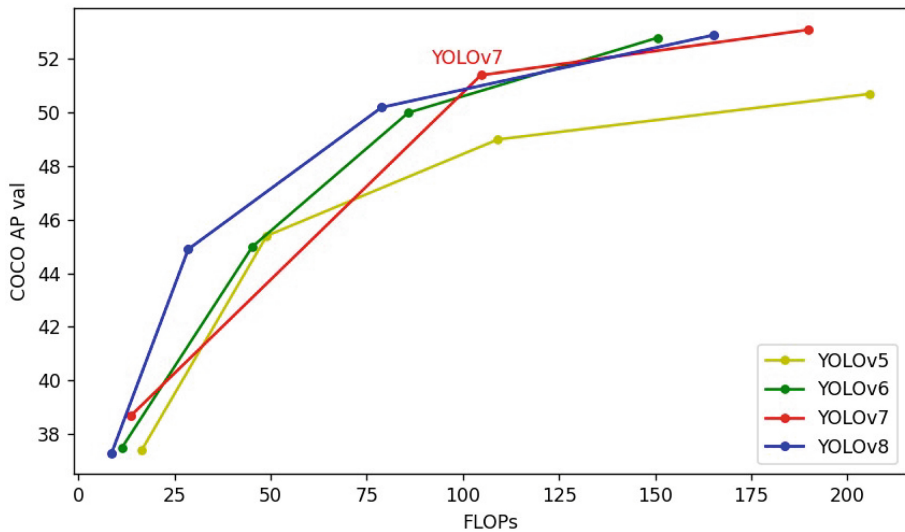


Fig. 2. YOLO series model performance comparison chart.

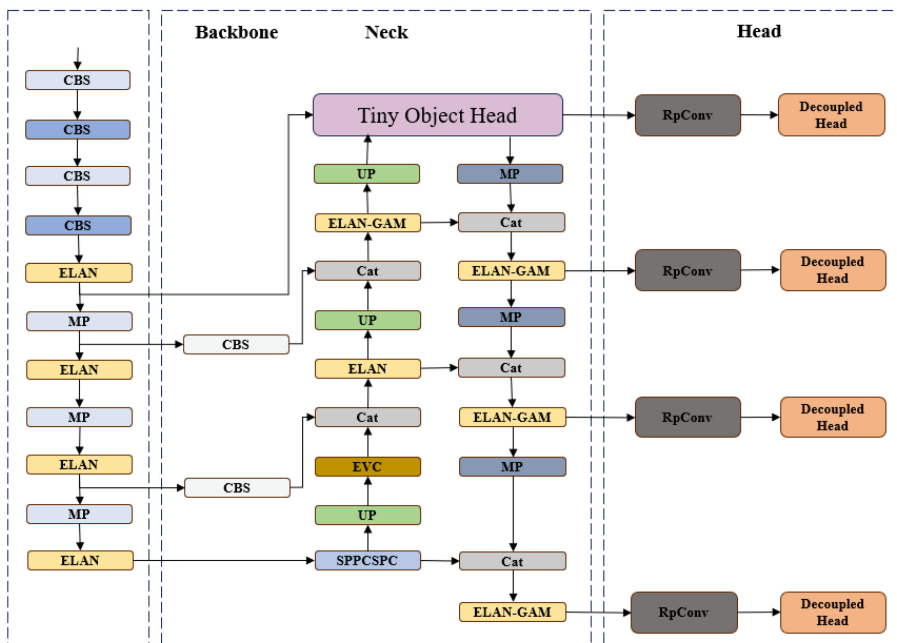


Fig. 3. The overall architecture of YOLO-RSOD.

process, the four prediction heads jointly learn features and perform object detection, giving full play to their respective advantages.

As shown in Fig. 4, Tiny Object Head, a prediction head specifically used to detect tiny objects, allows image features to be passed to the feature fusion module at a shallower level of the model, so that the model can obtain more detailed information about tiny objects in the image. This method allows the model to capture more features of tiny objects, thereby improving the detection accuracy of tiny objects in remote sensing images.

Although adding tiny object head will bring certain computational and memory overhead, this trade-off is worth accepting compared to the improved tiny object detection performance. Because for scenes such as remote sensing images that contain a large number of tiny objects, improving the tiny object detection effect is very critical, and subsequent experimental results can prove this.

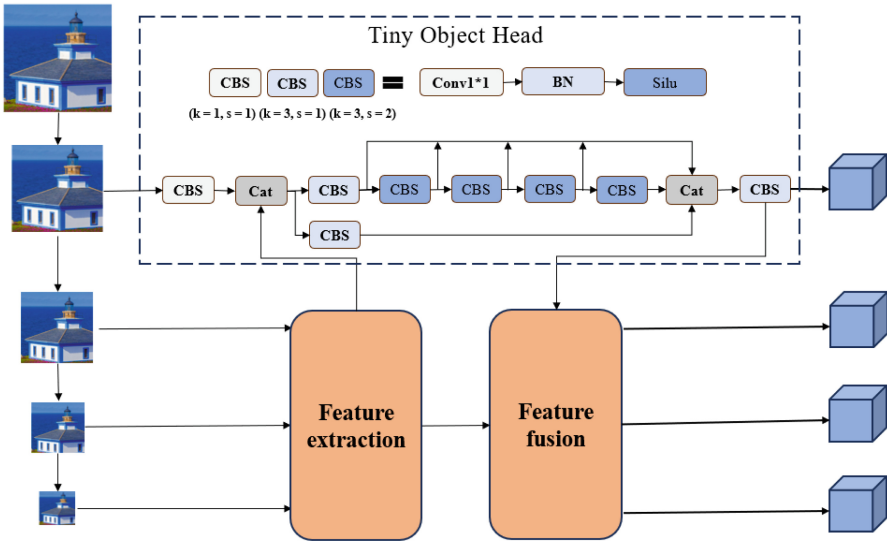


Fig. 4. Added tiny object head.

Decoupled Head. Traditional detection models, such as YOLOv5, use a single detection head that predicts both the object category and the location of the box. This design has a problem: combining category prediction and location prediction in one head may cause the error of one task to affect the other task. Category prediction and location prediction have different problem domains and require different loss functions and network layers.

The decoupled head separates category prediction and location prediction, and uses two independent network branches to process them respectively. This can optimize the loss function of each task separately, improve model flexibility, and avoid mutual interference between different tasks.

The decoupled detection head structure used in this paper is shown in Fig. 5. The category prediction branch uses a fully connected layer to output various probabilities. The location prediction branch uses a convolutional layer to generate bounding box coordinates, and IoU is used as an evaluation indicator to measure the quality of the prediction results, and non-maximum suppression is applied in the post-processing stage.

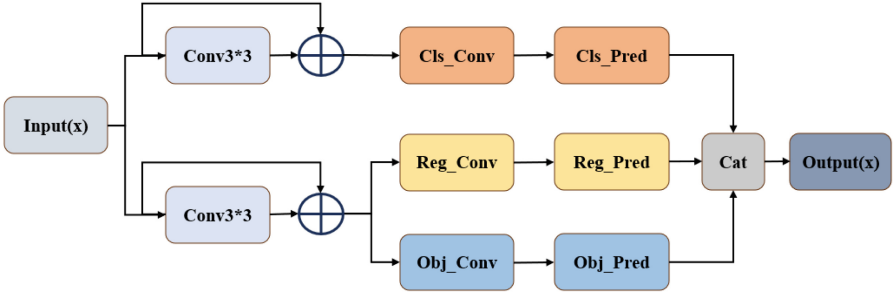


Fig. 5. The overall architecture of Decoupled Head.

Centralized Feature Pyramid. The feature pyramid is a basic neck network in modern recognition systems that can be effectively and efficiently used to detect objects at different scales. Overall, feature pyramid can handle the problem of multi-scale changes in object recognition without increasing computational overhead, and the extracted features can generate multi-scale feature representations that include some high-resolution features. Compared with existing feature pyramids, the CFP not only captures global long-range dependencies, but also efficiently obtains comprehensive and discriminative feature representations. The structure of the core block EVC in the CFP is shown in Fig. 6.

From Fig. 6, Between the top-level features X_{in} and EVC, there is a Stem Block for feature smoothing. The Stem Block consists of a 7×7 convolution with an output channel size of 256, followed by a batch normalization layer and an activation function layer. The above process can be expressed by X_{sb} as formula (1).

$$X_{sb} = BN(Conv7 * 7(X_{in})) \quad (1)$$

It can be seen that EVC mainly consists of two parallel-connected blocks, where a lightweight MLP is used to capture the global information of the top-level feature X_{in} . Implement a learnable visual centering mechanism on X_{in} using Learnable Visual Center (LVC) to aggregate local region features within layers. The resulting feature maps of these two blocks are concatenated along the

channel dimension as the output of EVC for downstream recognition. It can be expressed as formula (2).

$$X_{out} = Cat(MLP(Xsb), LVC(Xsb)) \tag{2}$$

In Fig. 6, the architecture of EVC uses a lightweight MLP module that can capture remote dependencies and a parallel LVC to aggregate local corner regions of the input image. The integrated features incorporate the benefits of both the MLP and LVC modules, allowing the detection model to learn comprehensive and discriminative feature representations. Experiments show that the EVC architecture improves the detection capability of the model in this paper.

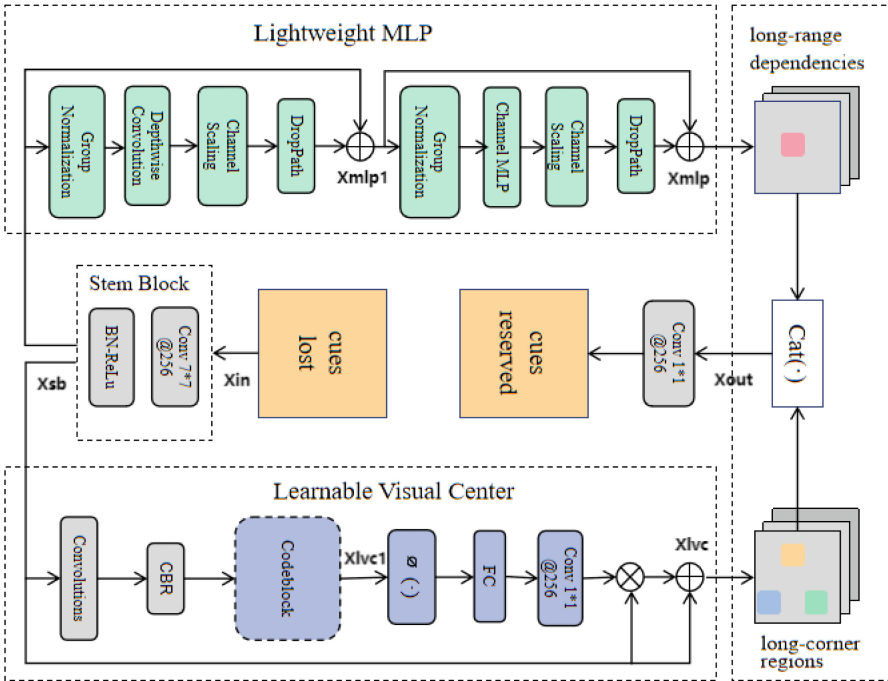


Fig. 6. The overall architecture of Explicit Vision Center.

Global Attention Module. Global Attention Module (GAM) is a simple but effective module for attention. It is a lightweight module that can be integrated into the best known CNN architectures and can be trained in an end-to-end manner. Given a feature map, GAM has two modules, channel attention and spatial attention. Channel attention uses a 3D arrangement to preserve information in three dimensions, while to focus on spatial information, two convolutional layers are used in the spatial attention sub-module for spatial information

fusion. GAM sequentially infers the attention map along the two separate dimensions, channel and spatial, and then the attention map multiplied by the input feature map to performs adaptive feature refinement. The structure of the GAM module is shown in Fig. 7, the execution process can be expressed as formula (3).

$$F_1 = RC * H * W, F_2 = M_c(F_1)F_1, F_3 = M_s(F_2)F_2 \quad (3)$$

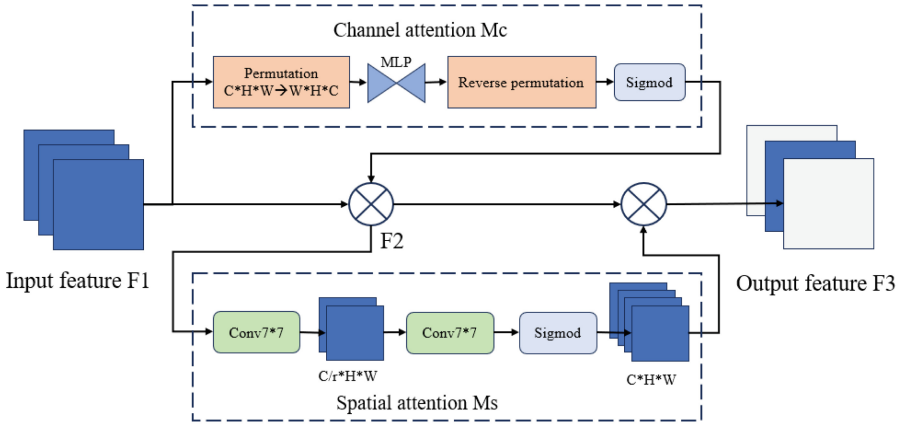


Fig. 7. The overall architecture of GAM.

In formula (3), F_1 is the input feature map, F_2 is the intermediate state and F_3 is the output. M_c is the channel attention map and M_s is spatial attention map.

According to the experiments in the paper, the performance of the model is greatly improved after integrating GAM into different models on different classification and detection datasets. In remote sensing object detection, there is usually a variety of complex and puzzling background information contained in large scale images. Using GAM can extract the attention region and help YOLORSOD to eliminate the negative effects of complex background information. It makes model focus on useful objects.

4 Experiments

4.1 Implementation Details

- (1) Dataset section. This paper uses VisDrone2021 dataset and COCO2017 dataset for experiments. VisDrone2021 includes training set (6471 images), validation set (548 images) and test set (1610 images), and the maximum image resolution is 2000×1500 . COCO2017 includes train (118287 images), val (5000 images) and test (40670 images).

- (2) Training section. Training is performed on GPU server with 4 RTX 3090, using SGD optimizer, maximum learning rate 0.005, weight decay 0.0005, momentum 0.9, a total of 300 epochs, image size 640*640, batch size 32.

4.2 Results and Comparison

This paper selects YOLOV7 as the baseline model and compares the YOLOR-SOD detector with the YOLO series detectors and some popular object detectors on the VisDrone2021 dataset, as shown in Table 1. It is compared with the YOLO series object detectors and recent remote sensing object detectors on the COCO2017 dataset, as shown in Table 2. Table 3 shows the performance improvement of YOLO-RSOD compared to YOLOv7 in various categories of the VisDrone2021 dataset.

Table 1. Comparison of different object detectors in VisDrone2021.

Model	Size	AP50:95	AP50
YOLOv5-X	640*640	22.6%	38.6%
YOLOX-X	640*640	25.8%	43.2%
YOLOv6-L	640*640	27.3%	47.1%
YOLOv7	640*640	27.5%	48.6%
SF-YOLO	640*640	18.2%	34.3%
YOLO-IMP	640*640	20.1%	36.4%
EdgeYOLO	640*640	26.4%	44.8%
TPH-YOLO	640*640	28.3%	47.4%
YOLO-RSOD(ours)	640*640	30.7%	51.7%

Table 2. Comparison of different object detectors in COCO2017.

Model	Param	Size	AP50:95	AP50
YOLOv5-X	86.7M	640*640	50.7%	68.9%
YOLOX-X	99.1M	640*640	51.1%	69.3%
YOLOv6-L	59.6M	640*640	51.8%	69.2%
YOLOv7	36.9M	640*640	51.2%	69.7%
SF-YOLO	2.24M	640*640	32.3%	50.6%
YOLO-IMP	15.3M	640*640	42.7%	58.7%
EdgeYOLO	40.5M	640*640	50.6%	69.8%
TPH-YOLO	53.6M	640*640	51.2%	70.1%
YOLO-RSOD(ours)	39.8M	640*640	52.7%	70.1%

As can be seen from Table 1, YOLO-RSOD achieves 30.7% and 51.7% on AP50:95 and AP50, respectively. Specifically, YOLO-RSOD outperforms the

YOLOv7 model by 3.2% and 3.1% on AP50:95 and AP50. At the same time, the model outperforms the most popular remote sensing object detector, the TPH-YOLO model, by 2.4% on AP50:95 and is 4.3% ahead on the AP50 metric. The same conclusion is also verified on the COCO2017 dataset, where Table 2 shows that YOLO-RSOD leads all other models in AP50:95 and AP50. For the baseline model YOLOv7, AP50:95 improves by 1.5%, while the number of parameters only increases by 2.9M.

Table 3. Comparison of YOLO-RSOD and YOLOv7 on Specific Categories in the VisDrone2021 Dataset.

Class	YOLO-RSOD	YOLO-RSOD	YOLOv7	YOLOv7
	AP50:95	AP50	AP50:95	AP50
all	30.7%	51.7%	27.5%	48.6%
pedestrian	26.4%	54.7%	22.6%	51.2%
people	19%	44.3%	15.6%	40.7%
bicycle	11.5%	25.9%	8.5%	22.1%
car	62.5%	89.6%	60.5%	87.6%
van	38.2%	56.1%	35.7%	53.8%
truck	32.1%	48.5%	28.7%	46.1%
tricycle	22.9%	41.1%	20.2%	38.6%
awning-tricycle	14.3%	26.2%	12.2%	23.4%
bus	51.5%	72.2%	47.5%	68.5%
motor	27.4%	57.4%	23.7%	53.8%

For the improvement of specific categories, it can be concluded from Table 3 that YOLO-RSOD achieves good detection results for all 10 detection object categories included in the VisDrone2021 dataset. Specifically, the performance improvement for both large-sized and small-sized objects is significant. For example, in the AP50:95 indicator, the detection performance for buses improves by 4%, and the performance for pedestrians improves by 3.8%. There is also a corresponding improvement in the detection of dense and complex objects; for instance, in the AP50:95 indicator, the detection performance of people improves by 3.4%, and the performance of awning-tricycles improves by 2.1%. The performance improvement for complex scenes is reflected in the overall performance enhancement.

The above experimental conclusions fully verify the improved object detection performance of YOLO-RSOD compared to YOLOv7, especially for common problems in remote sensing object detection, such as large size changes, high density situations, and complex backgrounds, which have been effectively solved.

4.3 Ablation Studies

In this paper, the importance of each component is analyzed on the VisDrone2021 validation set, and the experiments show that each component has

Table 4. Ablation experiments.

Model	Decoupled Head	GAM	EVC	Tiny Object Head	AP50:95	AP50
YOLO-RSOD					27.5%	48.6%
YOLO-RSOD*					27.9%	49.3%
YOLO-RSOD*		*			28.1%	49.6%
YOLO-RSOD*		*	*		28.5%	50.1%
YOLO-RSOD*		*	*	*	30.7%	51.7%

some improvements on the remote sensing object detection capability. Table 4 lists the impact on YOLO-RSOD detection performance with the addition of each component.

According to Table 4, it can be seen that the four improvement measures adopted in this article can effectively improve the detection performance of remote sensing image objects. The specific performance is as follows: (1) After adding the decoupling structure to the original IDetect Head, the model’s AP50:95 increased by 0.4%, and AP50 increased by 0.7%. (2) After replacing the convolution block of the ELAN structure of the network with the GAM attention mechanism, the model’s AP50:95 increased by 0.2%, and AP50 increased by 0.3%. (3) By adding the EVC block of the feature pyramid to the NECK part of the network, the network’s ability to extract multi-scale features is enhanced. The model’s AP50:95 is increased by 0.4%, and AP50 is increased by 0.5%. (4) After using an additional tiny object head, the model’s AP50:95 increased by 2.2%, and AP50 increased by 1.6%. It can be seen that the additional remote sensing object detection heads contribute the most to the performance of the model. The reason is that the remote sensing objects in the data set are too small and there are too many types. Only when the additional detection heads can detect these objects first can other modules be more accurate. Good performance.

4.4 Visualization

The YOLO-RSOD proposed in this paper is better than YOLOv7 on remote sensing object datasets. In order to prove this more intuitively, this paper visualizes the detection results of YOLO-RSOD and YOLOv7 for the remote sensing object dataset. Specifically, some representative remote sensing object images in the validation set of VisDrone2021 dataset are first selected, and then YOLORSOD and YOLOv7 are used to detect these images respectively. The detection results are shown in Fig. 8 with each row displaying different remote sensing object detection scenarios and each column displaying the detection performance of different models, from left to right: YOLOv7, YOLO-RSOD, Ground Truth.

As shown in Fig. 8, for four groups of very representative remote sensing object scenes, it can be clearly seen that the detection effect of YOLO-RSOD is better than that of YOLOv7. Specifically, the first group of images is a common low-density larger object scene, that is, a traditional object detection scene. The

detection effect of this scene can well illustrate the model's detection performance for common objects. It can be seen from the detection results that for larger objects, YOLOv7 and YOLO-RSOD perform equally well, and both achieve excellent detection results. For example, both detectors can successfully detect all objects to be identified in the scene.

The second group of images shows a common multi-size dense object scene in remote sensing object detection. This scene shows the characteristics of high density and large size variation of remote sensing objects. From the detection results, it can be seen that for densely distributed and small objects, YOLOv7 can detect larger objects in the foreground very well, but the detection effect for smaller objects farther away is very poor, and some categories are not successfully identified. YOLO-RSOD can achieve excellent detection results for both large and small objects.



Fig. 8. Visual comparison of detection results.

The third group shows one of the important application scenarios in remote sensing object detection - aerial photography detection. This scenario shows the large coverage characteristic of remote sensing objects. From this group of comparative experiments, it can be seen that YOLOv7 is basically unable to

detect very distant objects in similar pictures, while the YOLO-RSOD proposed in this paper performs much better than YOLOv7 in remote sensing object detection, especially aerial photography detection.

The fourth group shows the remote sensing object scene at night, which is mainly to reflect the robustness of the model. This set of comparative experiments shows that the detection ability of YOLO-RSOD at night is also stronger than that of YOLOv7.

Through the comparison experiment of the detection effects in Fig. 8, it can be seen that the three main problems of remote sensing object detection at this stage are large-scale changes, high density and large coverage. In addition, this paper also conducts night scene detection to verify robustness. The YOLO-RSOD proposed in this paper can effectively overcome these problems and performs much better than YOLOv7.

5 Conclusion

This paper proposes a remote sensing object detection framework YOLO-RSOD. This model is more effective for remote sensing image datasets and is optimized and improved on the framework of the state-of-the-art one-level object detector YOLOv7. First, an effective remote sensing object data prediction head is proposed by combining the GAM attention mechanism and EVC module in the feature pyramid at the neck. Finally, a more effective decoupling head structure is adopted in the detection head structure, resulting in a detector with excellent performance. The YOLO-RSOD model is particularly good at object detection in remote sensing image scenes. Tests on the VisDrone2021 dataset show that YOLO-RSOD achieves excellent performance on the remote sensing image dataset, indicating that YOLO-RSOD provides better results in the analysis and processing of remote sensing image scenes.

The YOLO-RSOD proposed in this article still has some room for improvement in terms of model complexity and calculation speed. In the future, the author will further optimize the model structure, reduce computational complexity, and improve detection speed to meet the needs of more practical application scenarios.

References

1. Hmidani, O., Alaoui, E.M.I.: A comprehensive survey of the R-CNN family for object detection. In: 2022 5th International Conference on Advanced Communication Technologies and Networking (CommNet), pp. 1–6. IEEE (2022)
2. Li, C., Li, L., Jiang, H., et al.: YOLOv6: a single-stage object detection framework for industrial applications. arXiv preprint [arXiv:2209.02976](https://arxiv.org/abs/2209.02976) (2022)
3. Ge, Z., Liu, S., Wang, F., et al.: Yolox: exceeding yolo series in 2021. arXiv preprint [arXiv:2107.08430](https://arxiv.org/abs/2107.08430) (2021)
4. 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)

5. Liang, S., Wu, H., Zhen, L., et al.: Edge YOLO: real-time intelligent object detection system based on edge-cloud cooperation in autonomous vehicles. *IEEE Trans. Intell. Transp. Syst.* **23**(12), 25345–25360 (2022)
6. 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
7. Cao, Y., He, Z., Wang, L., et al.: VisDrone-DET2021: the vision meets drone object detection challenge results. In: *Proceedings of the IEEE/CVF International conference on computer vision*, pp. 2847–2854 (2021)
8. Shang, J., Wang, J., Liu, S., Wang, C., Zheng, B.: Remote sensing object detection algorithm for UAV aerial photography based on improved YOLOv5s. *Electronics* (2023)
9. Liu, Y., Shao, Z., Hoffmann, N.: Global attention mechanism: Retain information to enhance channel-spatial interactions. *arXiv preprint [arXiv:2112.05561](https://arxiv.org/abs/2112.05561)* (2021)
10. Quan, Y., Zhang, D., Zhang, L., et al.: Centralized feature pyramid for object detection. *IEEE Trans. Image Process.* (2023)
11. Zhu, X., Lyu, S., Wang, X., et al.: TPH-YOLOv5: improved YOLOv5 based on transformer prediction head for object detection on drone-captured scenarios. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2778–2788 (2021)
12. Ma, Y., Liu, S., Li, Z., et al.: IQDet: instance-wise quality distribution sampling for object detection. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1717–1725 (2021)
13. Song, G., Liu, Y., Wang, X.: Revisiting the sibling head in object detector. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11563–11572 (2020)



All-Weather Vehicle Detection and Classification with Adversarial and Semi-Supervised Learning

Yi-Chao Huang¹ and Huei-Yung Lin^{1,2}(✉)

¹ Department of Electrical Engineering, National Chung Cheng University, Chiayi 621, Taiwan

yc.daniel.huang@gmail.com

² Department of Computer Science and Information Engineering, National Taipei University of Technology, Taipei 106, Taiwan

lin@ntut.edu.tw

Abstract. The images taken under varying lighting or adverse weather conditions exhibit different distributions in high-dimensional space, and make object detection networks perform poorly. In this paper, we propose a domain adaptation method based on adversarial learning to ensure the features extracted from a similar distribution, even when the input images belong to different domains. Considering the scarcity of images taken under certain weather conditions in the existing dataset, a semi-supervised learning framework is incorporated to enhance the detection performance through training with unlabeled images. The experiments conducted on public and private datasets show that our proposed adversarial learning technique outperforms the recent traffic scene object detection networks in all different domains. Source code and datasets are available at <https://github.com/daniel851218/all-weather-vehicle-detector>.

Keywords: All Weather Object Detection · Adversarial Learning · Semi-Supervised Learning

1 Introduction

In the past decades, numerous computer vision tasks have exceeded the performance of conventional algorithms due to the explosive growth of deep neural networks. The advances in these technologies have enhanced the ability of machines to perceive the real world, which leads to the development of more comprehensive advanced driver assistance systems. It provides both drivers and pedestrians with a safer environment. Current object detection networks, whether one-stage models like YOLOv7 [20], or two-stage models such as Faster R-CNN [17], can perform the target identification successfully in various driving datasets (e.g., KITTI [10], CityScapes [4], and BDD100K [21]). These datasets contain images primarily acquired under daytime with normal weather conditions.



Fig. 1. Road scene images captured under adverse conditions for various application scenarios.

For a broader aspect, the lighting and weather conditions are highly variable in the real world. In general, the images captured in the nighttime might be excessively dark due to under-exposure. The headlights of coming vehicles can cause backlighting, which results in significant variations between bright and dark areas of the image. In addition, the increase of exposure time will lead to the effect of motion blur in the images. As shown in Fig. 1, image blurring is also prone to occur due to the light refraction caused by raindrops during the rainy weather. Since conventional cameras are limited by the intensity dynamic range in image acquisition, they cannot capture bright and clear scenery under all-weather scenarios. Consequently, the object detect networks trained on daytime and normal weather data face challenges in the applicability to general situations due to the domain shift between training and testing data.

An advanced driver assistance system should be capable of detecting objects under any lighting and weather conditions [13]. It is clear that training multiple object detection networks for different scenarios using the images captured under various conditions is an inefficient approach. Collecting and labeling image data is a time-consuming and labor-intensive task. To improve the detection performance of all-weather conditions, infrared or thermal cameras are adopted, and combined with RGB images for cross-modality deep learning techniques [5, 8, 23, 24]. Although the contour characteristics of objects could be perceived in nighttime or dense foggy weather, the equipment is typically high-cost, and presents issues such as lower resolution and distance limitations.

In this work, we propose an all-weather detection network based on adversarial and semi-supervised learning for traffic scenes. The objective is to adopt the same network model to identify targets in the RGB images captured under different situations. We consider four scenarios based on the lighting variations and weather conditions, more specifically, daytime normal, daytime rainy, night normal, and night rainy. For the feature extraction of traffic scenes associated with a similar distribution but belonged to different domains, we incorporated daytime/night and normal/rainy domain classifiers for adversarial learning. To deal with imbalanced data due to the significantly less rainy images in available datasets, teacher-student frameworks for semi-supervised learning are adopted to enable the object detection network trained on imbalanced samples. The experiments carried out on SHIFT, BDD100K and our datasets have demonstrated the effectiveness of the proposed approach. Our source code and datasets are available at <https://github.com/daniel851218/all-weather-vehicle-detector>.

2 Related Work

To improve the performance of visual tasks under adverse conditions, some researchers utilized GAN-based approaches for image style transformation. In previous works, Anoosheh *et al.* proposed ToDayGAN [1] to transfer image styles from night to daytime. Based on CycleGAN and image retrieval, it was able to perform the accurate localization of robots with 6-DOF. ForkGAN proposed by Zheng *et al.* employed a fork-shape module composed of one encoder and two decoders. It was used to disentangle the domain-invariant and domain-specific image features across different scenarios [22]. Data augmentation using ForkGAN to transform labeled daytime images into nighttime reduced data imbalance and enhanced visual localization, semantic segmentation, and object detection in the nighttime. However, a major drawback of GAN-based methods is the uncertainty of the generated contents. During the process of image style transfer, small objects such as pedestrians, motorcycles, and distant vehicles are easy to disappear. This is a crucial flaw for development of advanced driver assistance systems, and particularly noticeable in dark nighttime images.

The images captured under adverse conditions are usually with low quality. This makes network detection results prone to false positives and misses. Thus, some previous researches proposed image enhancement methods to improve the quality before feeding the images into the object detection networks. One typical approach is to train the neural networks for direct adjustments of parameters to enhance the image quality. Guo *et al.* proposed an end-to-end training framework “enhance before detect” [11]. This approach converts the RGB images into the YCbCr color space to adjust exposure and then feed them into Faster R-CNN for object detection. In [14], Liu *et al.* proposed Image-Adaptive YOLO by using a differentiable processing module to perform dehazing, sharpening, contrast enhancement and white balance adjustment for object detection. Nayak *et al.* proposed ObjectRL, which trained an agent by deep reinforcement learning to adjust the image properties for improvements [16]. It utilized a pre-trained

network and provided feedback to the agent based on IoU and F1 scores. While these approaches are intuitive and adaptable to various image processing algorithms, they often struggle to increase the object detection performance when the image quality is fairly poor.

When the images are captured under different lighting and weather conditions, the domain shift problem occurs as their distributions are different in high-dimensional space. In order to enable object detection networks to perform cross-domain detection, Chen *et al.* proposed Domain Adaptive Faster R-CNN to divide domain shift into image-level and instance-level [3]. The approach utilized adversarial learning based on H-divergence to extract features with similar distributions, even if input images came from different domains. Saito *et al.* claimed that aligning the global features of images from different domains would be beneficial, and proposed strong-weak distribution alignment to enhance the performance of Domain Adaptive Faster R-CNN [18]. To address the issue of insufficient labeled images in the target domain, Deng *et al.* proposed an unbiased mean teacher method [6]. It utilized CycleGAN to generate source-like or target-like images, and send them into the mean teacher model for semi-supervised learning. Jiao *et al.* noted that in the semi-supervised learning framework using current teacher-student networks, the model weight updates of the teacher network overly depend on the student network. Hence, a dual instance-consistent network was proposed to learn and extract the features independently from the source and target domains [12].

This paper leverages the advantages of adversarial learning to align features from different domains. In addition, semi-supervised learning is employed for training with unlabeled images to enhance the detection network. The target objects can then be identified under the adverse lighting and weather conditions.

3 Approach

To reduce the effect of domain shift due to varying lighting and address the problem of insufficient image quantity under certain weather conditions, this work first utilizes adversarial learning to extract the features with a consistent distribution by an object detection network. Simultaneously, we incorporate a semi-supervised learning framework to train the object detection network using unlabeled images for minimizing the efforts of data collection and annotation. In order to alleviate the performance degradation of object detection network due to the application scenario change, we classify input images into four domains: daytime normal, daytime rainy, nighttime normal, and nighttime rainy. Faster R-CNN is utilized as our backbone for object detection with two domain classifiers for time and weather. Each kind of domain classifiers are further broken down into image-level and instance-level classifiers. Hence, there are totally four different classifiers to recognize the images belonged to which domain and at different feature level.

The features derived from the feature extractor consist of the global characteristics of input images, and are referred to as image-level features. Similarly,

the feature maps obtained from the RoI align represent the local characteristics of input images, and are hence termed instance-level features. Before feeding both types of features into the corresponding domain classifiers, they are processed with the gradient reversal layer [9] to minimize the detection loss and maximize the domain classification loss at the same time. Since the image-level and instance level features originate from the same input image, the classification results of different domain classifiers must also be the same.

The losses of daytime and weather domain classifiers are given by

$$L_d^{adv}(X, \hat{C}_d) = BCE(C_d^{ins}, \hat{C}_d) + BCE(C_d^{img}, \hat{C}_d) + MSE(C_d^{img}, C_d^{ins}) \quad (1)$$

and

$$L_w^{adv}(X, \hat{C}_w) = BCE(C_w^{ins}, \hat{C}_w) + BCE(C_w^{img}, \hat{C}_w) + MSE(C_w^{img}, C_w^{ins}), \quad (2)$$

where \hat{C}_d and \hat{C}_w denote the ground truth labels, C_d^{img} and C_w^{img} are the image-level predictions of two classifiers, and C_d^{ins} and C_w^{ins} represent the instance-level predictions. Equations (1) and (2) consist of three loss terms. The first two are used to calculate the domain classification loss of each classifier with the binary cross entropy (*BCE*). The third term is used to compute the consistency loss between the image-level and instance-level classifiers by mean squared error (*MSE*).

This adversarial learning based method enables the feature extractor of Faster R-CNN to extract the features with similar distributions. Consequently, it can achieve cross-domain object detection under different lighting and weather domains. Figure 2 shows the overall architecture of the proposed Adversarial Faster R-CNN. Based on Eqs. (1) and (2), the total loss is defined by

$$L_{det}^{adv}(X, \hat{C}, \hat{B}) = L_{det}(X, \hat{C}_{obj}, \hat{B}) + \lambda_d \cdot L_d^{adv}(X, \hat{C}_d) + \lambda_w \cdot L_w^{adv}(X, \hat{C}_w) \quad (3)$$

where λ_d and λ_w are used to control the influence of daytime and weather classifiers, respectively.

3.1 Semi-Supervised Adversarial Object Detection Network

Due to the frequency of different weather conditions, the collection of driving images could easily lead to a significant quantity imbalance in the dataset. In the BDD100K training dataset, the images of daytime rainy and night rainy account for only 5% and 4% of the entire dataset, respectively. The number of rainy images is almost ten times less than normal weather images. To address the data imbalance issue caused by the images from different domains, we collect our driving data and YouTube video clips for specific weather conditions as a private dataset for model training. The image annotation is conducted by incorporating the teacher-student network in Adversarial Faster R-CNN. Based on this network structure, Semi-Supervised Adversarial Faster R-CNN is established as shown in Fig. 3. Note that the architecture of student network is almost identical to teacher network, except for the domain classifiers attached for adversarial learning.

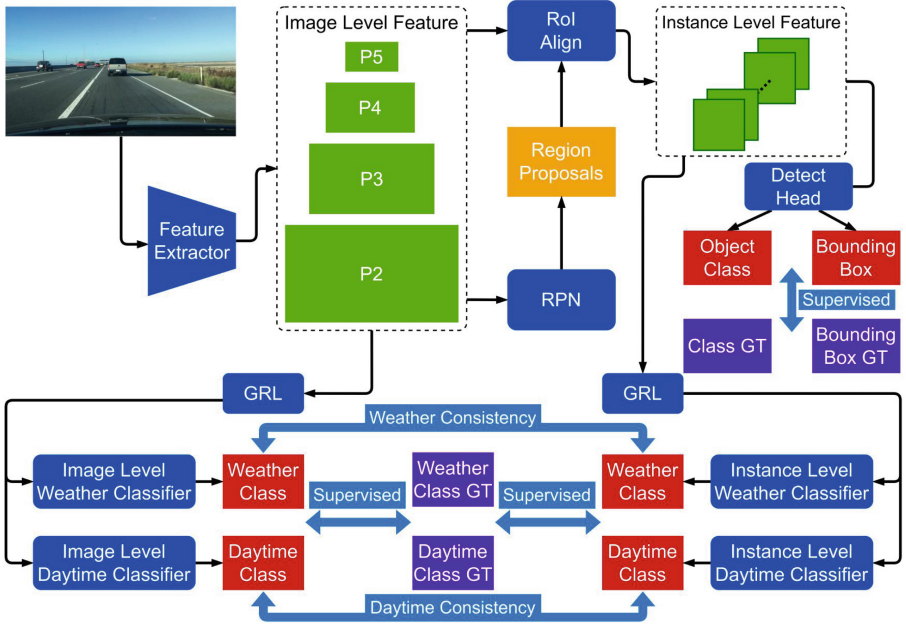


Fig. 2. The architecture of the proposed Adversarial Faster R-CNN. The total loss contains weighted influence of daytime and weather classifiers.

In general, the object detection networks cannot generate enough accurate pseudo labels for the unlabeled low-quality images. This work employs the concept of curriculum learning [2] by selecting the easier images for training prior to the difficult ones. The student network is trained with labeled data for each iteration first to make the training process more stable. It is followed by performing data augmentation on the unlabeled image to obtain two images with less disturbance. One is fed into the teacher network to generate pseudo labels, and the other is sent into the student network for supervised learning with pseudo labels. After several iterations, student network updates its weights to the teacher network using the exponential moving average

$$\theta_t \leftarrow \alpha \cdot \theta_t + (1 - \alpha) \cdot \theta_s \tag{4}$$

where θ_t and θ_s denote the weights of the teacher and student networks, respectively.

Too many false detections in the pseudo labels will cause the student network learned the incorrect information. As the training iteration increases, the object detection results may become even worse. To deal with this problem, we filter out the low-confidence pseudo labels using double-thresholding and voting mechanism to derive more reliable ones. For the bounding boxes with confidence scores larger than the high threshold or smaller than the low threshold, they are kept or abandoned respectively. For the ones with confidence scores between

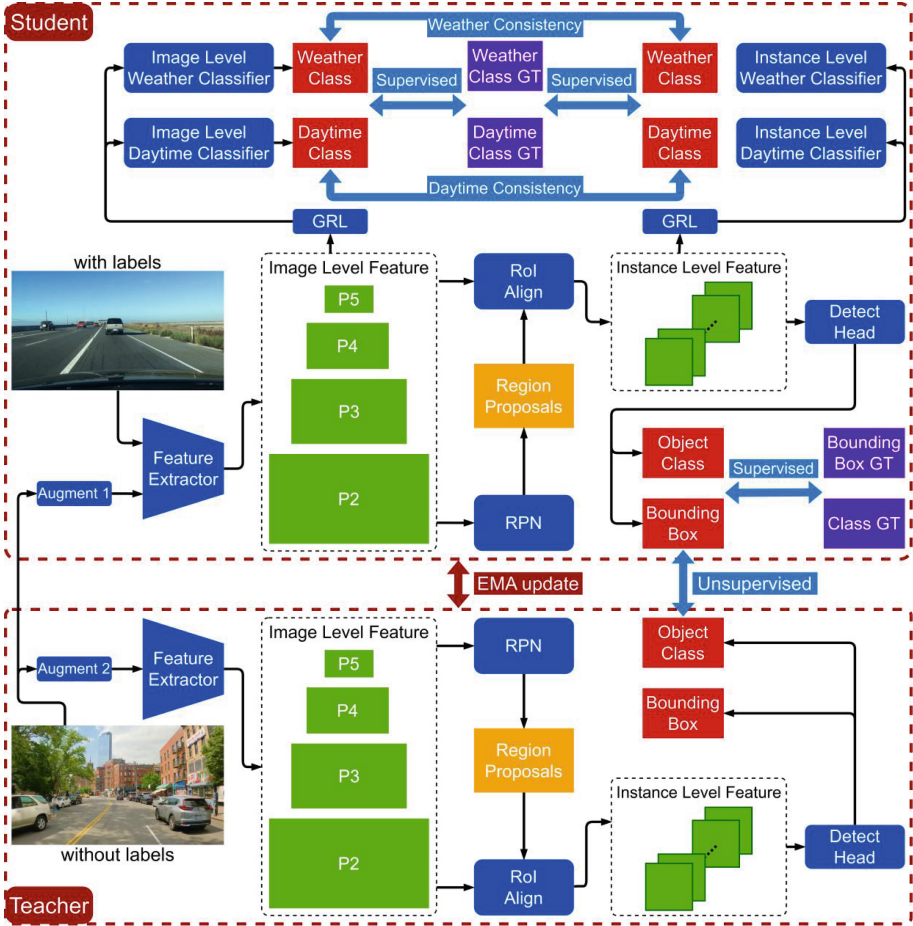


Fig. 3. The architecture of our Semi-Supervised Adversarial Faster R-CNN. The student network has the structure similar to the teacher network, except for the domain classifiers are attached for adversarial learning.

the low and high thresholds, the voting based on the width, height, aspect ratio and area of each ground truth label in the public dataset is conducted. After sorting with these four characteristics, the pseudo labels in the first quartile or their aspect ratios in the fourth quartile will receive one vote. When the number of votes exceeds the voting threshold, the pseudo label is likely a false positive, and is abandoned from the student network training.

The loss function for the proposed Semi-Supervised Adversarial Faster R-CNN is defined by

$$L_{det}^{ssl}(X^{sup}, X^{unsup}, \hat{C}, \hat{B}) = L_{det}^{sup}(X^{sup}, \hat{C}, \hat{B}) + \lambda_{unsup} \cdot L_{det}^{unsup}(X^{unsup}, C^T, B^T) + L_{adv}^{unsup}(X, \hat{C}) \quad (5)$$

Algorithm 1. Training Process of Semi-Supervised Learning

Input: Images with labels X^{sup}
Input: Images without labels X^{unsup}
Input: $F(x)$ is Data Augmentation Function
Input: $G(\{C_j, B_j\})$ is Pseudo Label Filter Function

- 1: Get pre-trained weight θ_{init} by training Adversarial Faster R-CNN with X^{sup}
- 2: Initialize Student Network $S(x)$ with θ_{init}
- 3: Initialize Teacher Network $T(x)$ with θ_{init}
- 4: **for** $i = 0$ to MAX_EPOCH **do**
- 5: **for** $x_{sup} \in X^{sup}, x_{unsup} \in X^{unsup}$ **do**
- 6: Train Student Network with $F(x_{sup})$
- 7: Compute Supervised Loss L_{det}^{sup}
- 8: Get $x_1^{unsup} = F(x^{unsup})$
- 9: Get $x_2^{unsup} = F(x^{unsup})$
- 10: Get Pseudo Label $\{C_j^T, B_j^T\} = T(x_1^{unsup})$
- 11: $\{C_j'^T, B_j'^T\} = G(\{C_j^T, B_j^T\})$
- 12: Train Student Network with x_2^{unsup} and $\{C_j^T, B_j^T\}$
- 13: Compute Adversarial Loss L_{adv}^{unsup}
- 14: Compute Unsupervised Loss L_{det}^{unsup}
- 15: Total Loss $L_{det}^{ssl} = L_{det}^{sup} + \lambda_{unsup} L_{det}^{unsup} + L_{adv}^{unsup}$
- 16: **end for**
- 17: Update Student Parameter θ_s
- 18: **if** $i \% \text{NUM_UPDATE} == 0$ **then**
- 19: Update Teacher Parameter $\theta_t \leftarrow \alpha \cdot \theta_t + (1 - \alpha) \cdot \theta_s$
- 20: **end if**
- 21: **end for**

where X^{sup} and X^{unsup} denote the labeled and unlabeled data, C^T and B^T are the pseudo labels of object category and bounding box position, and λ_{unsup} is used to control the influence of the unsupervised learning. It consists of three components, with first the supervised loss derived from the labeled data, second the unsupervised loss obtained from the pseudo labels, and last the adversarial loss calculated from cross-domain learning. The overall training process is shown in Algorithm 1.

4 Experiments

The proposed detection techniques for all-weather driving are evaluated on two public datasets, SHIFT, BDD100K, and our recorded driving videos. We compare the performance of the networks, Faster R-CNN, Adversarial Faster R-CNN, and Semi-Supervised Adversarial Faster R-CNN. The images in the datasets are categorized to four domains: daytime sunny, daytime rainy, nighttime sunny, and nighttime rainy. In the experiments, we consider six target objects: car, bus, bicycle, truck, motorcycle, and pedestrian.

SHIFT Dataset. The SHIFT dataset contains synthetic images generated using the CARLA simulator [7,19]. It incorporates various onboard sensors to create virtual driving images with diverse lighting and weather conditions. SHIFT is mainly designed for domain adaptation applied to various autonomous driving tasks.

BDD100K Dataset. The BDD100K dataset consists of large-scale driving data released by UC Berkeley DeepDrive [21]. It includes a wide range of road scenes, weather conditions, and lighting, with data annotation provided for various autonomous driving evaluation. Nevertheless, the dataset still covers a large part of images taken in the normal weather as illustrated in Fig. 4.

Private Dataset. To address the issue of limited number of rainy weather images in the BDD100K dataset, in this paper we include additional images of rainy scenes from our driving recording and downloads from YouTube. The images are served as unlabeled data in the training process of Semi-Supervised Adversarial Faster R-CNN.

The SHIFT and BDD100K datasets are used for training and testing of the backbone network and adversarial learning. In the training phase of semi-supervised learning, both of the BDD100K and private datasets are utilized. The BDD100K dataset provides ground truth data, which helps maintain the training stability. On the other hand, the private image dataset is used to expose the network to a wider variety of data. In the testing phase, the BDD100K and private datasets are used for qualitative and quantitative experiments, respectively.

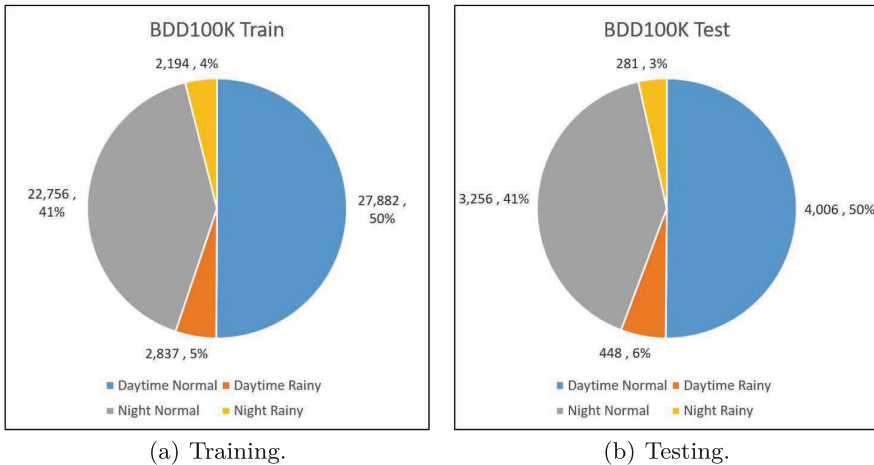


Fig. 4. The distributions of the numbers of images provided by the four domains in the BDD100K dataset.

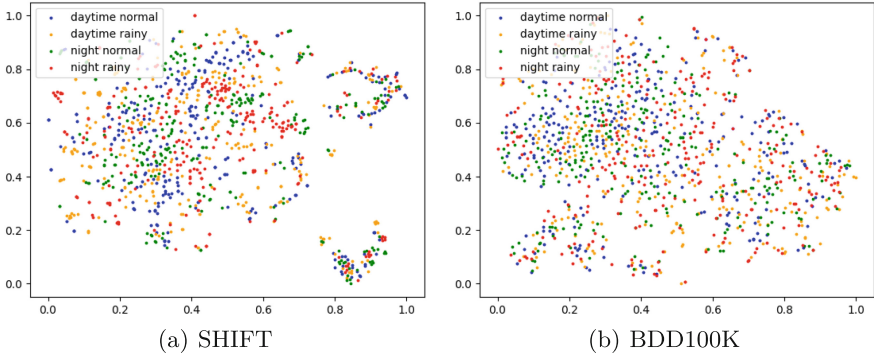


Fig. 5. The feature distributions of the SHIFT and BDD100K datasets after adversarial learning.

Table 1. The results of adversarial learning on the SHIFT dataset for different domains and classes.

P/R/AP	Car	Truck	Pedestrian	Bus	Bicycle	Motorcycle
Daytime Normal	-0.0157/ 0.0248 / 0.0110	-0.0132/ 0.0187 / 0.0334	0.0445 / 0.0088 / 0.0202	0.0414 / 0.0315 / 0.0621	0.0324 / 0.0420 / 0.0229	0.0589 / -0.0030/ 0.0288
Daytime Rainy	0.0197 / 0.0144 / 0.0098	-0.0332/ 0.0593 / 0.0429	0.0230 / 0.0195 / 0.0248	0.0318 / 0.0930 / 0.0868	0.0091 / 0.0309 / 0.0115	-0.0452/ 0.0710 / 0.0097
Night Normal	-0.0062/ 0.0377 / 0.0043	0.0593 / 0.0114 / 0.0314	-0.0106/ 0.0480 / 0.0318	0.0544 / 0.0276 / 0.0424	0.0409 / -0.0037/ 0.0402	0.0756 / 0.0126 / 0.0099
Night Rainy	0.0287 / 0.0069 / 0.0086	0.0466 / 0.0916 / 0.0553	0.0640 / 0.0281 / 0.0396	-0.0644/ 0.0523 / 0.0346	0.1702 / 0.0607 / 0.0899	0.0717 / 0.0357 / 0.0554

4.1 Results of Adversarial Learning

We use Precision (P), Recall (R), and AP ($IoU = 0.5$) as evaluation metrics. The improvement of adversarial driving scene detection network is evaluated separately on the SHIFT and BDD100K datasets. Figure 5 shows the feature distribution after the adversarial learning visualized through t-SNE [15]. It can be observed that the domain shift issue among images is indeed alleviated by adversarial learning. Tables 1 and 2 represent the improvements achieved by Faster R-CNN with adversarial learning on the SHIFT and BDD100K datasets, respectively, with positive values indicate improvement and negative values indicate degradation.

Table 1 shows that Adversarial Faster R-CNN effectively improves Precision, Recall, and AP on the images from four different domains compared to Faster R-CNN in the SHIFT dataset. In these four domains, the night rainy scene exhibits

Table 2. The results of adversarial learning on the BDD100K dataset for different domains and classes.

P/R/AP	Car	Truck	Pedestrian	Bus	Bicycle	Motorcycle
Daytime Normal	0.0087 / 0.0023 / 0.0017	0.0574 / -0.0360/ -0.0021	0.0121 / 0.0002 / 0.0038	0.1055 / -0.0410/ 0.0093	0.0789 / 0.0401 / 0.0548	0.1266 / -0.0430/ 0.0214
Daytime Rainy	-0.0151/ 0.0261 / 0.0093	-0.0418/ -0.0033/ -0.0143	0.0278 / -0.0317/ -0.0164	-0.3293/ 0.1182 / 0.0095	0.0593 / 0.0714 / 0.0613	-0.0627/ 0.0000/ -0.0927
Night Normal	0.0112 / -0.0021/ -0.0010	-0.1072/ 0.0397 / 0.0017	-0.0106/ -0.0097/ -0.0047	-0.0536/ 0.0273 / 0.0099	0.0618 / 0.0263 / 0.0735	0.0686 / 0.0864 / 0.1498
Night Rainy	0.0228 / 0.0066 / 0.0017	0.0158 / -0.0139/ 0.0161	0.1333 / -0.0455/ 0.0850	-0.0957/ -0.0383/ -0.0905	0.4615 / -0.1333/ 0.0048	-0.5714/ 0.1000 / -0.0448



(a) Ground Truth



(b) w/o adv



(c) w/ adv

Fig. 6. The result of Adversarial Faster R-CNN in Night Rainy obtained from the SHIFT dataset.

the most significant improvement. Especially, the bicycle and motorcycle classes increase 17.02% and 7.17% in precision, and truck and bicycle increase 9.16% and 6.07% in recall, respectively. Figure 6 shows the results from night rainy images with rainy blur and low light scene. Compared to the ground truth labels depicted in Fig. 6(a), Faster R-CNN misses two small targets on the right (car

Table 3. The performance evaluation of Faster R-CNN and the proposed semi-supervised adversarial learning technique on daytime normal images.

P/R/AP	Car	Truck	Person	Bus	Bicycle	Motorcycle	Average
Faster R-CNN	0.8354/ 0.6537/ 0.7349	0.5765/ 0.5169 / 0.5096	0.6594/ 0.4930/ 0.5271	0.6445/ 0.4675/ 0.5117	0.4796/ 0.2361/ 0.2473	0.3850/ 0.3008 / 0.2402	0.5967/ 0.4447 / 0.4618
w/adv	0.8441/ 0.6561/ 0.7366	0.6338 / 0.4809/ 0.5075	0.6715/ 0.4932 / 0.5310	0.7500 / 0.4265/ 0.5210	0.5586 / 0.2762/ 0.3021	0.5116 / 0.2578/ 0.2616	0.6616 / 0.4318/ 0.4766
w/adv & ssl	0.8447 / 0.6584 / 0.7378	0.6180/ 0.4914/ 0.5098	0.7111 / 0.4751/ 0.5313	0.6288/ 0.4795 / 0.5197	0.5482/ 0.2784 / 0.3007	0.5000/ 0.2578/ 0.2629	0.6418/ 0.4401/ 0.4771

and pedestrian, see Fig. 6(b)), while our Adversarial Faster R-CNN is able to detect all four objects (two trucks, one car and one pedestrian) as shown in Fig. 6(c).

As tabulated in Table 2, the improvements of Adversarial Faster R-CNN are less significant on the BDD100K dataset. This is mainly because the highly diverse real-world images cannot be entirely replicated by synthetic images generated by the driving scene simulator. In particular, there is a notable gap between the rainy effects produced by the simulation and real-world scenes. Furthermore, the number of images across the four different domains is relatively balanced in the SHIFT dataset. For the BDD100K dataset, there is a clear imbalance in terms of the images across different domains as depicted in Fig. 4. Consequently, it limits the increasing performance of Adversarial Faster R-CNN on the daytime rainy and night rainy scenes.

4.2 Results of Semi-Supervised Learning

Tables 3, 4, 4, 5 and 6 present the detection results of three models on the images from four different domains in the BDD100K dataset. It can be observed that no object detection network significantly outperforms the others for the daytime normal, daytime rainy, and night normal scenes. However, networks with semi-supervised learning consistently achieve the best or second-best performance in these three evaluation metrics. This demonstrates the capability of semi-supervised learning for the improvement on driving scene detection performance of CNN-based networks.

While Table 6 shows that, on the average, Faster R-CNN provides better performance for the night rainy scene, some categories (such as car and truck) have a significantly lower representation in the dataset. This could result in a high false (positive and negative) detection rate. As illustrated in Fig. 7 for the outputs of the private dataset, Faster R-CNN can only detect the vehicles near the image center. However, Adversarial Faster R-CNN with domain classifiers is able to identify additional targets at the cost of some false positives. After incorporating the teacher-student framework for Semi-Supervised Adversarial

Table 4. The performance evaluation of Faster R-CNN and the proposed semi-supervised adversarial learning technique on daytime rainy images.

P/R/AP	Car	Truck	Person	Bus	Bicycle	Motorcycle	Average
Faster R-CNN	0.8417 / 0.6621/ 0.7465	0.6255 / 0.5621/ 0.5726	0.7300/ 0.4929 / 0.5402	0.9756 / 0.3636/ 0.5015	0.4074/ 0.2619/ 0.2374	0.5333/ 0.4000 / 0.3406	0.6856 / 0.4571/ 0.4898
w/adv	0.8265/ 0.6882 / 0.7558	0.5836/ 0.5588/ 0.5584	0.7578/ 0.4612/ 0.5237	0.6463/ 0.4818/ 0.5111	0.4667 / 0.3333/ 0.2987	0.4706/ 0.4000 / 0.2479	0.6253/ 0.4872/ 0.4826
w/adv & ssl	0.8305/ 0.6851/ 0.7545	0.5682/ 0.5719 / 0.5641	0.7624 / 0.4628/ 0.5237	0.6154/ 0.5091 / 0.5102	0.4167/ 0.3571 / 0.2993	0.7000 / 0.3500/ 0.2657	0.6489/ 0.4893 / 0.4863

Table 5. The performance evaluation of Faster R-CNN and the proposed semi-supervised adversarial learning technique on night normal images.

P/R/AP	Car	Truck	Person	Bus	Bicycle	Motorcycle	Average
Faster R-CNN	0.7756/ 0.6497/ 0.7261	0.6488 / 0.5000/ 0.5000	0.6124 / 0.4719 / 0.4930	0.6298 / 0.4453/ 0.4608	0.4382/ 0.2566/ 0.2206	0.5000/ 0.2716/ 0.2329	0.6008/ 0.4325/ 0.4389
w/adv	0.7868/ 0.6476 / 0.7251	0.5415/ 0.5397 / 0.5017	0.6018/ 0.4622/ 0.4883	0.5762/ 0.4727 / 0.4706	0.5000 / 0.2829/ 0.2941	0.5686/ 0.3580 / 0.3827	0.5958/ 0.4605 / 0.4771
w/adv & ssl	0.7997 / 0.6413/ 0.7261	0.5675/ 0.5293/ 0.5023	0.5944/ 0.4681/ 0.4888	0.5930/ 0.4609/ 0.4670	0.4234/ 0.3092 / 0.2999	0.6364 / 0.3457/ 0.3965	0.6024 / 0.4591/ 0.4801

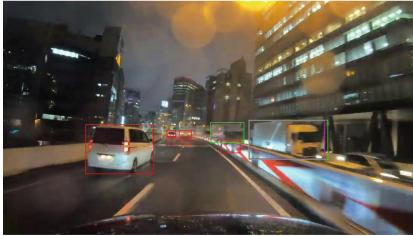
Table 6. The performance evaluation of Faster R-CNN and the proposed semi-supervised adversarial learning technique on night rainy images.

P/R/AP	Car	Truck	Person	Bus	Bicycle	Motorcycle	Average
Faster R-CNN	0.7844/ 0.6188/ 0.7084	0.8478/ 0.5417/ 0.6037	0.5568 / 0.4170 / 0.4075	0.8667/ 0.5909 / 0.6523	0.5385/ 0.4667 / 0.4187	1.0000 / 0.2000/ 0.2284	0.7657 / 0.4725 / 0.5032
w/adv	0.8072 / 0.6254/ 0.7100	0.8636 / 0.5278/ 0.6198	0.4611/ 0.3787/ 0.3170	1.0000 / 0.5455/ 0.7374	1.0000 / 0.3333/ 0.4235	0.4286/ 0.3000 / 0.1836	0.7601/ 0.4518/ 0.4985
w/adv & ssl	0.8023/ 0.6283 / 0.7116	0.6984/ 0.6111 / 0.6202	0.4754/ 0.3702/ 0.3182	1.0000 / 0.5455/ 0.7376	1.0000 / 0.3333/ 0.4190	0.4286/ 0.3000 / 0.1847	0.7341/ 0.4647/ 0.4986

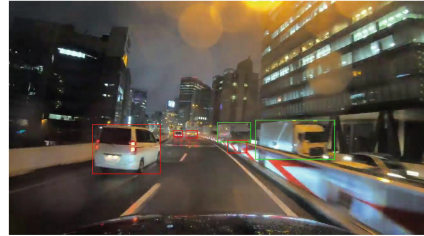
Faster R-CNN, the false positives are reduced and the best detection result is achieved. All in all, the proposed semi-supervised adversarial network possesses the better generalization capability for diverse input data.



(a) Faster R-CNN



(b) w/ adv



(c) w/ adv & ssl

Fig. 7. The result of Semi-Supervised Adversarial Faster R-CNN in Night Rainy obtained from our private dataset.

5 Conclusion

To perform road scene object detection without individual training on the images acquired under different lighting and weather conditions, this paper proposes a method to employ adversarial learning to alleviate domain shifts among images. Since the approach does not involve domain classifiers during model inference or testing, the computation time for testing can be reduced. In the experiments conducted on public and private datasets, the proposed adversarial learning technique outperforms the recent traffic scene object detection networks in all different domains. It also demonstrates most significant improvements in the night rainy scenario.

References

1. Anoosheh, A., Sattler, T., Timofte, R., Pollefeys, M., Van Gool, L.: Night-to-day image translation for retrieval-based localization. In: 2019 International Conference on Robotics and Automation (ICRA), pp. 5958–5964. IEEE (2019)
2. Bengio, Y., Louradour, J., Collobert, R., Weston, J.: Curriculum learning. In: Proceedings of the 26th Annual International Conference on Machine Learning, pp. 41–48 (2009)
3. 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)

4. Cordts, M., et al.: The cityscapes dataset for semantic urban scene understanding. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3213–3223 (2016)
5. Deng, F., et al.: Feanet: feature-enhanced attention network for rgb-thermal real-time semantic segmentation. In: 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 4467–4473. IEEE (2021)
6. Deng, J., Li, W., Chen, Y., Duan, L.: Unbiased mean teacher for cross-domain object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4091–4101 (2021)
7. Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., Koltun, V.: Carla: an open urban driving simulator. In: Conference on Robot Learning, pp. 1–16. PMLR (2017)
8. El Ahmar, W., et al.: Enhanced thermal-rgb fusion for robust object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 365–374 (2023)
9. Ganin, Y., Lempitsky, V.: Unsupervised domain adaptation by backpropagation. In: International Conference on Machine Learning, pp. 1180–1189. PMLR (2015)
10. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 3354–3361. IEEE (2012)
11. Guo, H., Lu, T., Wu, Y.: Dynamic low-light image enhancement for object detection via end-to-end training. In: 2020 25th International Conference on Pattern Recognition (ICPR), pp. 5611–5618. IEEE (2021)
12. Jiao, Y., Yao, H., Xu, C.: Dual instance-consistent network for cross-domain object detection. IEEE Trans. Pattern Anal. Mach. Intell. (2022)
13. Lin, H.Y., Dai, J.M., Wu, L.T., Chen, L.Q.: A vision-based driver assistance system with forward collision and overtaking detection. Sensors **20**(18), 5139 (2020)
14. Liu, W., Ren, G., Yu, R., Guo, S., Zhu, J., Zhang, L.: Image-adaptive yolo for object detection in adverse weather conditions. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 36, pp. 1792–1800 (2022)
15. Van der Maaten, L., Hinton, G.: Visualizing data using t-sne. Journal of machine learning research **9**(11) (2008)
16. Nayak, S., Ravindran, B.: Reinforcement learning for improving object detection. In: Computer Vision–ECCV 2020 Workshops: Glasgow, UK, 23–28 August 2020, Proceedings, Part V 16, pp. 149–161. Springer (2020)
17. 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)
18. Saito, K., Ushiku, Y., Harada, T., Saenko, K.: Strong-weak distribution alignment for adaptive object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6956–6965 (2019)
19. Sun, T., et al.: Shift: a synthetic driving dataset for continuous multi-task domain adaptation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 21371–21382 (2022)
20. 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)
21. Yu, F., et al.: Bdd100k: a diverse driving dataset for heterogeneous multitask learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2636–2645 (2020)

22. Zheng, Z., Wu, Y., Han, X., Shi, J.: Forkgan: seeing into the rainy night. In: *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020, Proceedings, Part III 16*, pp. 155–170. Springer (2020)
23. Zhou, W., Guo, Q., Lei, J., Yu, L., Hwang, J.N.: Ecffnet: effective and consistent feature fusion network for rgb-t salient object detection. *IEEE Trans. Circuits Syst. Video Technol.* **32**(3), 1224–1235 (2021)
24. Zhu, Y., Sun, X., Wang, M., Huang, H.: Multi-modal feature pyramid transformer for rgb-infrared object detection. *IEEE Trans. Intell. Transp. Syst.* (2023)



Who Should Have Been Focused: Transferring Attention-Based Knowledge from Future Observations for Trajectory Prediction

Seokha Moon¹, Kyuhwan Yeon², Hayoung Kim², Seong-Gyun Jeong²,
and Jinkyu Kim¹(✉)

¹ Korea University, 145 Anam-ro, Seoul 02841, Republic of Korea

jinkyukim@korea.ac.kr

² 42dot, 20 Changeop-ro 40beon-gil, Seongnam-si, Gyeonggi-do 13449,
Republic of Korea

Abstract. Accurately predicting the trajectories of dynamic agents is crucial for the safe navigation of autonomous robotics. However, achieving precise predictions based solely on past and current observations is challenging due to the inherent uncertainty in each agent's intentions, greatly influencing their future trajectory. Furthermore, the lack of precise information about agents' future poses leads to ambiguity regarding which agents should be focused on for predicting the target agent's future. To solve this problem, we propose a teacher-student learning approach. Here, the teacher model utilizes actual future poses of other agents to determine which agents should be focused on for the final prediction. This attentional knowledge guides the student model in determining which agents to focus on and how much attention to allocate when predicting future trajectories. Additionally, we introduce a Lane-guided Attention Module (LAM) that considers interactions with local lanes near predicted trajectories to enhance prediction performance. This module is integrated into the student model to refine agent features, thereby facilitating a more accurate emulation of the teacher model. We demonstrate the effectiveness of our proposed model with a large-scale ArgoVerse motion forecasting dataset, improving overall prediction performance. Our model can be used plug-and-play, showing consistent performance gain. Additionally, it generates more human-intuitive trajectories, e.g., avoiding collisions with other agents, keeping its lane, or considering relations with other agents.

Keywords: Trajectory Prediction · Motion Forecasting · Autonomous Driving

1 Introduction

Predicting the future poses of dynamic agents is critical for autonomous vehicles (or robots) to navigate safely and avoid collisions. Various methods have been

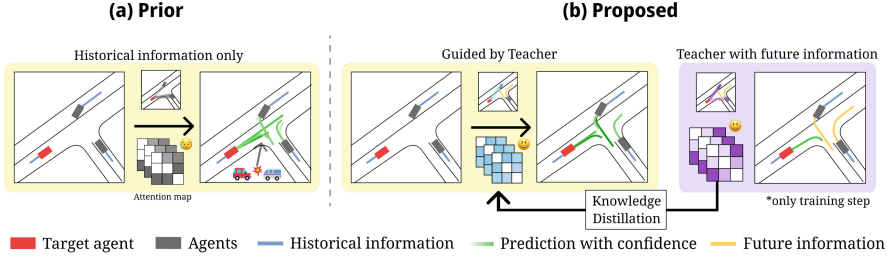


Fig. 1. (a) Existing approaches depend only on past and current observations, which presents challenges in addressing intentional uncertainties of other agents and optimizing joint behavior distributions. (b) To address this, we propose a teacher-student learning approach. The teacher model uses future observations of other agents to determine should-be-attended agents for the final prediction. This attentional knowledge is transferred to the student model, guiding it to learn agent-wise relations efficiently.

developed to predict agents' future trajectories using map-based contexts [7, 11, 14, 34] and considering agent-wise interactions [25, 28]. Despite promising results, the uncertainty in agents' intentions remains a significant obstacle to accurate prediction. This unpredictability challenges the forecasting of interactions and complicates the optimization of collective behavior among agents.

However, by considering the future positions of other agents (though unrealistic assumptions during inference time), uncertain intentions can be clarified. This approach offers a more accurate understanding of other agents that should be focused as they proceed with the interaction when predicting the future trajectory of the target agent. Inspired by this observation, in this paper, we aim to leverage the attentional knowledge of a teacher model, which utilizes other agents' future information, as supervision for determining the extent of interaction between each agent.

To this end, we utilize a teacher-student learning approach where knowledge can be transferred from the teacher to the student model as shown in Fig. 1. Among various knowledge transfer approaches, we focus on attention-based knowledge transfer, where the agent-wise attention (i.e., which agents should be attended to) distribution is transferred. Specifically, our teacher model possesses critical attentional knowledge for agent-specific future predictions, as all agents determine which agents should be focused on in order to predict trajectories by leveraging the future poses of other agents, excluding themselves, respectively. Our student model is not allowed to use that future information but is regularized to mimic its attention to be similar to that of the teacher model, i.e., minimizing attentional differences between the teacher and student models. We empirically observe that such guidance notably improves the overall quality of trajectory prediction, focusing on other agents that the teacher model attends to.

Further, we propose a novel Lane-guided Attention Module (LAM) that refines the predicted trajectories of each agent through interaction with confidence-augmented lane near the predicted trajectories.

To demonstrate the effectiveness of our proposed model, we conducted experiments with the publicly available large-scale Motion Forecasting Dataset [3]. Our model, which is applied to existing HiVT [35] and LangeGCN [14] models, provides a significant improvement. To our best knowledge, this is the first work to transfer attentional knowledge learned from other agents' future information.

2 Related Work

2.1 Trajectory Prediction

Recent works in the trajectory prediction domain have introduced methods for predicting agents' trajectories using their past trajectories and high-definition map (HD map). However, challenges remain, such as (i) dealing with trajectory uncertainty related to human intentions and (ii) accurately predicting interactions with other elements present in the driving scene such as road components and other agents. Notable approaches to address (i) include optimizing Gaussian location uncertainties and integrating agents' predicted goal positions to better understand their intentions [2, 11, 24, 30, 31, 34]. To resolve (ii), attention-based architectures [9, 20, 21, 26] have been increasingly chosen as the method for fusing multimodal data and considering interactions between various components. Employing an attention-based architecture allows for the joint prediction of all agents' trajectories simultaneously, encompassing the entire scene. Studies such as [11, 31] leverage predicted goal points to consider the intent of each agent and incorporate this information into the interaction between agent features. However, predicting future trajectories of agents affected by human intentions based solely on historical information is challenging, leading to uncertainty in the interactions between agents. To address this challenge, Sun et al. [28] uses distance-based rules to classify the types of interactions between agents (e.g., first pass, next pass, or unrelated relationships) and integrate this information into model learning. However, this method incurs ambiguity, where unrelated agents can be classified as related, and additional costs in classifying interaction types. Additionally, it is limited to considering only predefined relationships between agents. Therefore, in this study, the process of classifying interaction types is omitted. Instead, interactions are autonomously considered using future trajectory information from other agents in the teacher model. This information is used as a supervision in the student model, guiding which agents will focus on when interacting.

2.2 Knowledge Distillation

Knowledge distillation [12] is a widely used method in various fields of computer vision and natural language processing, aimed at transferring knowledge from

high-performing models with a large number of parameters to smaller models with fewer parameters [10, 23]. With the increasing attention towards knowledge distillation, there have been numerous studies and research efforts aimed at enhancing its performance [1, 13, 22, 29]. [4] demonstrates that larger or more accurate teacher models do not necessarily result in better student models during the knowledge distillation process, also propose that early stopping of teacher model’s training can mitigate this problem. [18] highlight the limitation of conventional knowledge distillation, especially when the size gap between the teacher and the student model is significant. To overcome this, they propose a multi-step knowledge distillation method utilizing an intermediate-sized ‘Teacher Assistant’, effectively bridging the gap and improving the performance of the student model. In recent years, knowledge distillation has been extensively studied and applied in the field of autonomous vehicle application. For example, Su *et al.* [27] proposes a model that is not affected by the number of agents through knowledge transfer from an agent-centric model (teacher) that has high performance but increases computational cost geometrically with the number of agents to a scene-centric model (student). Monti *et al.* [19] proposes an approach that exploits only a few observation inputs to increase prediction performance and eliminate noise probability in the detection phase. In this study, we propose a method of transferring the knowledge of a teacher model, which can better understand the interaction between the target agent and other agents by referencing the future trajectory of other agents, to a student model. This approach allows predicting the interactions between agents based not only on history information but also on distilled features. And this allows for the efficient use of computation resources and improves the performance of the student model.

3 Method

As shown in Fig. 2, our model uses a teacher-student learning approach where (i) the teacher model \mathcal{T} leverages other agents’ ground-truth future poses to determine which agents to be attended (or focused) for predicting the target agents’ trajectories. Such attentional knowledge is then transferred to (ii) the student model \mathcal{S} by forcing it to mimic the teacher model’s attention distributions. This learning approach can be flexibly applied to a variety of attention-based trajectory prediction models. Therefore, we extend our approach by building upon an existing attention-based model as a baseline.

3.1 Preliminary

A sequence of poses (x, y) in 2D coordinates for an agent $i \in [1, N]$ is split into the observation trajectory $X_i = \{(x, y)_i^t | t \in \{1, 2, \dots, T_{\text{in}}\}\}$ and the future trajectory $Y_i = \{(x, y)_i^t | t \in \{T_{\text{in}} + 1, \dots, T_{\text{in}} + T_{\text{out}}\}\}$ where T_{in} and T_{out} are the observation and the prediction horizons, respectively. Local Encoder f_{local} generates per-agent contextual embeddings $\mathbf{c}_i \in \mathbb{R}^d$ for $i \in [1, N]$ given agents’ past observation trajectory X_i and map data. We utilize an attention-based Local

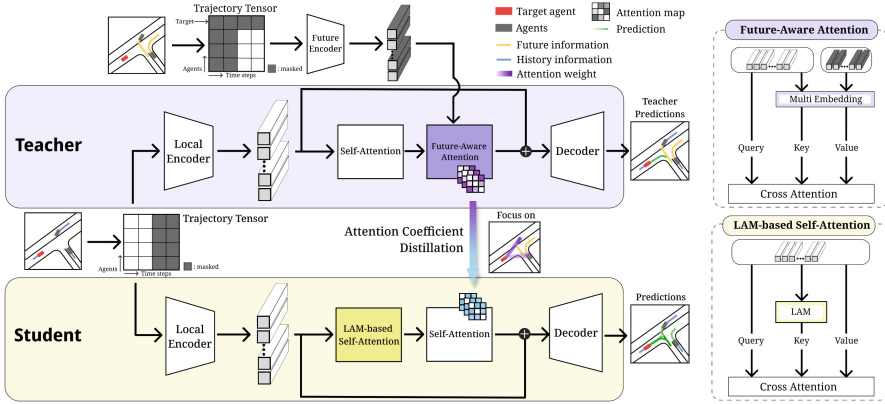


Fig. 2. An overview of our proposed method. Our model is built upon a teacher-student learning model where (top) the teacher model can leverage the ground-truth future observations of other agents while (bottom) the student model cannot use them. The knowledge is transferred from the teacher model to the student model via attention-based knowledge distillation, i.e., which agents should be *attended* to and be *interacted* with for the final prediction.

Encoder [14,35] that concentrates on agent-centric local regions, learning temporal dependencies and agent-map relations without considering interactions between agents.

3.2 Teacher Network

Our Teacher network \mathcal{T} leverages other agents’ ground-truth “future” trajectories Y_j for $j \neq i$ to determine which agents should be truly considered in predicting future trajectories. To facilitate this process, we propose utilizing two components as shown in Fig. 2 (top): (1) Future Encoder and (2) Future-Aware Attention module. Our Future Encoder embeds the displacement of agents’ real future trajectories at each time step $\{p_i^t - p_i^{t-1}\}_{t=T_{in}^{out}+T_{in}}$ and concatenates it with learnable token. Subsequently, this concatenated feature is fed into the temporal attention block, enabling the extraction of overall temporal information about the agent’s future from the learnable token \mathbf{o}_j such as BERT [5] and ViT [6].

The Future-Aware-Attention module is utilized to learn which agents the target agent should focus on, taking into account the future trajectories of all other agents except the target. Through this module, each agent, acting as a target agent in turn, accesses the future information and learns more accurately which agents to consider in order to predict their future trajectories. Therefore, when the module is stacked multiple times, the knowledge of the target agent’s future is integrated into \mathbf{c}_j , which serves as both keys and values. Subsequently, this future information is exposed to the target agent’s feature \mathbf{c}_i . However, this could be perceived as a form of cheating when predicting the future trajectory

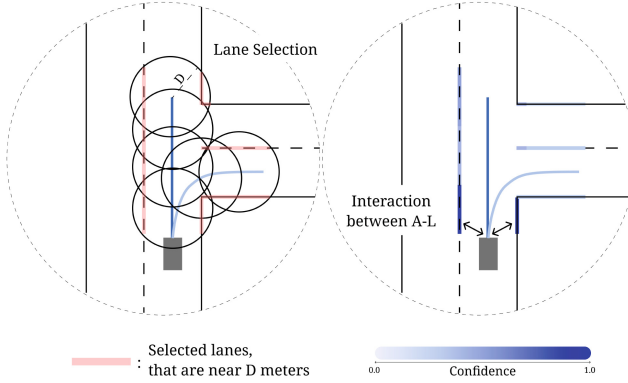


Fig. 3. An overview of Lane-guided Attention Module (LAM), which predicts future trajectories of agents using their features, selects lanes within \mathcal{D} meters of the predicted trajectories and assigns corresponding confidence value to the lanes. This module incorporates the interaction between the generated lane features and agents, aiming to reduce uncertainties regarding the future trajectories of the agents.

of the target agent. Thus, the teacher network generates an attention coefficient α_i using the Future-Aware Attention module only once, and learns the feature of the appropriate target agent. We further employ this module that utilizes the following query, key, and value with learnable parameters W_Q , W_K , and W_V and the attention coefficient α_i is also obtained as follows:

$$Q_i = W_Q \mathbf{c}_i, K_j = W_K \phi(\mathbf{c}_j + \mathbf{o}_j), V_j = W_V \phi(\mathbf{c}_j + \mathbf{o}_j), \quad (1)$$

$$\alpha_i = \text{softmax}\left(\frac{Q_i^T}{\sqrt{d}} [K_j]_{j \neq i}\right). \quad (2)$$

3.3 Attention-Based Knowledge Distillation

The attention coefficients α obtained from the teacher network provide crucial insights on which other agents should be attended to and the attention allocation required for accurate trajectory prediction of the target agent. To transfer this attentional knowledge from the teacher to the student network, we employ the knowledge distillation technique introduced in MINILM [32] by minimizing the following distillation loss:

$$\mathcal{L}_{\text{distill}} = \frac{1}{N} \sum_{i=1}^N \mathcal{D}_{KL}(\alpha_i || \alpha'_i). \quad (3)$$

where N is the number of agents. α_i and α'_i are the attention coefficients from the teacher model and the student model, respectively.

3.4 Lane-Guided Attention Module

As shown in Fig. 3, our proposed Lane-guided Attention Module (LAM) first predicts the future trajectory \hat{p}_i for each agent $i \in [1, N]$ using their respective features as input. Given this, the lane segments around each agent’s predicted future trajectories are extracted, and a corresponding confidence value is assigned to each extracted lane segment using the following equation:

$$C_{i\xi} = \sum_{k=1}^6 \begin{cases} \Pi^k, & \text{if } \exists \hat{p}_{ik}^t \mid |\hat{p}_{ik}^t - p_{\xi,0}| < \mathcal{D}, \\ 0, & \text{otherwise} \end{cases}, \tag{4}$$

where \hat{p}_{ik}^t represents the position of the i -th agent’s k -th mode of the t -th step in the predicted path, $p_{\xi,0}$ represents the starting position of lane ξ , and Π^k represents the confidence of the k -th mode in the i -th agent’s predicted path. The lanes to be passed by each agent are obtained within \mathcal{D} meter of the predicted trajectory, and each lane reflects the confidence of the trajectory. So, we can extract features for the lane segment related to the i -th agent as follows:

$$\mathbf{c}_{i\xi} = \phi_{lane} \left(\left[R_i^T(p_{\xi,1} - p_{\xi,0}), R_i^T(p_{\xi,0} - p_i^{T_{in}}), C_{i\xi}, a_\xi \right] \right), \tag{5}$$

where ϕ_{lane} and $p_i^{T_{in}}$ respectively represent the outputs of the MLP layer and the position of the i -th agent at time T_{in} . We define a 2×2 rotation matrix for conversion to the central coordinates of the i -th agent as $R_i \in \mathbb{R}^{2 \times 2}$. The starting and ending positions of lane segment ξ can be represented as $p_{\xi,0} \in \mathbb{R}^2$ and $p_{\xi,1} \in \mathbb{R}^2$, respectively. The semantic attributes of lane segment ξ are denoted as a_ξ . Then the confidence-weighted lane features are combined with the agent features using cross-attention with the following query, key, and value:

$$\bar{Q}_i = \bar{W}_Q \mathbf{c}_i, \bar{K}_{i\xi} = \bar{W}_K \mathbf{c}_{i\xi}, \bar{V}_{i\xi} = \bar{W}_V \mathbf{c}_{i\xi}. \tag{6}$$

3.5 Student Network

Similar to the teacher network, the student network \mathcal{S} utilizes a self-attention module to consider for interactions between agents. However, each agent feature in the student model is comparatively coarse compared to the teacher model, which utilizes future information of other agents except for itself. Thus, the refined agent context feature $\tilde{\mathbf{c}}_j$ obtained from LAM is utilized as the key to minimize this disparity between teacher model and student model. Therefore, the query, key, and value of the agent interaction attention in the student model are as follows:

$$\tilde{Q}_i = \tilde{W}_Q \mathbf{c}_i, \tilde{K}_j = \tilde{W}_K \tilde{\mathbf{c}}_j, \tilde{V}_j = \tilde{W}_V \mathbf{c}_j. \tag{7}$$

Since the teacher model solely utilizes the future information of other agents except for itself, and the student model only use the attention coefficients of the teacher model as supervision, no cheating occurs, and the mode diversity of the student model is preserved.

3.6 Decoder

Conditioned on the agent’s contextual embedding, our Decoder f_{decoder} learns Gaussian Mixture Model distributions across all time steps after T_{in} . This process generates future locations $\hat{\mathbf{p}}_i^t \in \mathbb{R}^2$ and their associated confidences $\mu_i^t \in \mathbb{R}^2$ for agent i at the forthcoming time step t , all within the agent-centric coordinate system. Such a decoder is commonly used to model both intent uncertainty (about the agents’ desired goal) and control uncertainty (about agents’ future states to satisfy its intent) in a single shot [2, 35].

3.7 Loss Function

Concretely, we minimize the following loss \mathcal{L} to train our student network:

$$\mathcal{L} = \mathcal{L}_{\text{traj}} + \mathcal{L}_{\text{distill}}, \quad (8)$$

to optimize the predicted trajectories by treating it as M distributions along with the confidence assigned to each distribution. To achieve this, we minimize the loss $\mathcal{L}_{\text{traj}}$ to optimize both the predicted trajectory and the confidence associated with it. Specifically, the trajectory loss $\mathcal{L}_{\text{traj}}$ is computed by minimizing the negative log-likelihood function between the ground-truth trajectories and the predicted trajectories derived from Gaussian mixture components for every agent, time step, and M mode, considering the confidence assigned to each predicted pose:

$$\mathcal{L}_{\text{traj}} = -\frac{1}{N} \sum_{i=1}^N \log \left(\sum_{m=1}^M \mu_{im} \cdot \frac{1}{\sqrt{2b^2}} \exp \left(-\frac{(\mathbf{p}_i - \hat{\mathbf{p}}_{im})^2}{2} \right) \right), \quad (9)$$

where \mathbf{p}_i denotes the actual future position of the i -th agent, $\hat{\mathbf{p}}_{im}, \mu_{im}$, denote the predicted position of the i -th agent of m -th mode and confidence for $\hat{\mathbf{p}}_{im}$ respectively, and b is the scale parameter. This trajectory loss is applied to the predictions from our LAM and our decoder f_{decoder} .

4 Experiments

4.1 Dataset

The Argoverse dataset [3] is a valuable resource for training and evaluating trajectory prediction models for autonomous vehicles. It includes 324,557 scenario samples, each lasting 5 s (2 s for past observation and 3 s for the future), sampled at a rate of 10 Hz, with high-definition maps, providing comprehensive data for trajectory prediction research. This dataset, which covers Pittsburgh and Miami in the United States, offers 205,942 training samples, 39,272 validation samples, and 78,143 testing samples. The motion forecasting task in the Argoverse dataset involves predicting the future trajectories of agents over a 3-second time horizon based on their past trajectories over a 2-second time span.

4.2 Metrics

Our model is evaluated using standard metrics for trajectory prediction, which includes minimum Average Displacement Error (minADE), minimum Final Displacement Error (minFDE), and Miss Rate (MR). The minADE measures the average displacement error between the ground truth trajectory and the best predicted sample out of $K=6$ joint samples. The minFDE metric, on the other hand, metric measures the final displacement error between the ground truth trajectory’s end position and the best predicted end position from $K = 6$ joint samples. The MR refers to the percentage of scenarios where the distance between the ground truth trajectory’s endpoint and the best predicted endpoint is above 2meter threshold.

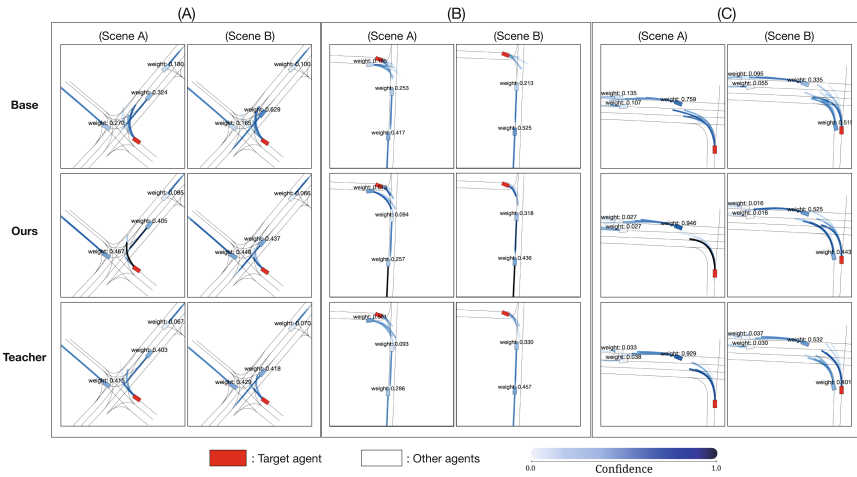


Fig. 4. Examples of trajectory prediction result in three complex scenarios, which demand agent-wise interactions to avoid collisions. To better understand the model’s ability to consider interactions with other agents, we (A) change the trajectories of another agent, (B) remove, or (C) add another agent (see changes from Scene A to B). We use the numeric information and the darkness of the color painted over each agent to indicate how much the target agent is focusing on the other agents.

4.3 Experiment Details

We conducted training for 64 epochs using a single RTX 3090 Ti GPU, employing the AdamW optimizer [17]. The model was trained with a batch size of 32, an initial learning rate of 5×10^{-4} , weight decay set to 1×10^{-4} , and a dropout rate of 0.1. To manage the learning rate, we utilized the cosine annealing scheduler [16]. Given that our model follows a Teacher-Student framework, we first trained the teacher model under the aforementioned conditions. Afterward, we froze the teacher model and employed it while training the student model. To

maintain consistency with standard practices, we set the number of predicted modes, denoted as F , to 6. In particular, any ensemble techniques were never used in both training and testing.

4.4 Effect on Learning Agent-Wise Relations

We observe that our proposed attentional knowledge distillation generally improves the quality of trajectory predictions, preventing them from violating physical occupancy constraints such as collisions or irrational trajectories, while directly confirming the results of the experiment.

In Fig. 4, we experiment how the target agent responds to tasks such as agent movement, addition, and deletion. Scenario A shows how the trajectory of the target agent and another agent changes when the position of another correlated agent changes at an intersection, and scenarios B and C show how well the interaction between agents is predicted and how reasonably the trajectory change is predicted when an agent is added or removed that affects the driving path of target agent. Our model effectively avoids a potential collision with other agents, while the base model does not. The base model’s predicted trajectories often remain mostly the same regardless of changes in another agent.

Further, in Fig. 5, we provide five examples of the predicted trajectories in various scenarios, including turning, intersection passing, and congested areas. Compared to Base model, our model learns to focus more on highly correlated agents, predicting better in various scenarios and avoiding collisions with other agents and not deviating from the lane centers. This may confirm the effectiveness of our proposed attentional knowledge distillation and Lane-guided Attention Module.

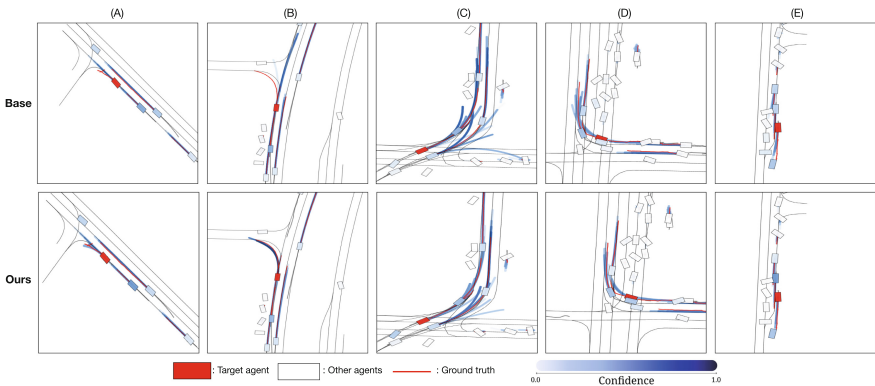


Fig. 5. Examples of trajectory prediction results from our base model and our proposed model. We provide typical examples in various driving scenarios: (A) slowing down, (B) turning left, (C) Crossing an intersection, (D) turning right, and (E) congestions. Our model generally predicts trajectories that avoid a potential collision with other agents, does not deviate from lane centers, and attend more relevant agents.

4.5 Quantitative Comparison with Other Existing Approaches

We observe in Table 1 that our model clearly outperforms other well-known models, including THOMAS [8], TNT [34], DenseTNT [11], LaneGCN [14], mmTransformer [15], DSP [33], SceneTransformer [21], HiVT [35], Multipath++ [30], in terms of minADE and minFDE. In this evaluation, we use the test split of the Argoverse motion forecasting dataset. As we start from the HiVT model, we report our reproduced scores (other scores are brought from the Argoverse motion forecasting leaderboard). Further, we observe in Fig. 6 that our student model produces attention distributions that are closer to those of the teacher model, while our base model does not (compare (a) vs. Base Model and (b) vs. Student Model (ours)). This confirms that our student model can successfully learn the teacher model’s attentional knowledge even without other agents’ future information. Also, it can be seen in Table 3 that mimicking the attentional knowledge of the teacher network in student network directly helps improve the trajectory prediction performance of the target agent.

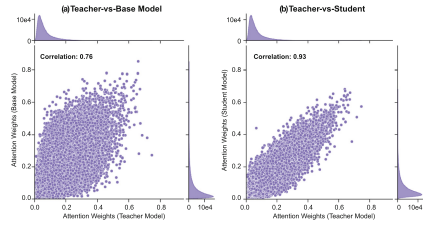


Fig. 6. Relations between attention weights (how much the target agent focus on which agent): (a) teacher-vs-base model and (b) teacher-vs-student (ours). We also provide the correlation scores

Table 1. Argoverse motion forecasting leaderboard scores on the trajectory prediction performance in terms of three widely-used metrics: minADE, minFDE, and MR (lower is better). We compare ours with the other existing approaches. † indicates our reproduced results. Data: Argoverse test set

Method	minADE (↓)	minFDE (↓)	MR (↓)
TNT [34]	0.9097	1.4457	0.1656
DenseTNT [11]	0.8817	1.2815	0.1258
LaneGCN [14]	0.8703	1.3622	0.1620
mmTransformer [15]	0.8436	1.3383	0.1540
DSP [33]	0.8194	1.2186	0.1303
SceneTransformer [21]	0.8026	1.2321	0.1255
Multipath++ [30]	0.7897	1.2144	0.1324
HiVT† [35]	0.7995	1.2321	0.1357
+ Ours	0.7867	1.2028	0.1319

Table 2. Performance comparison with our base model in terms of three different target agent’s maneuvers: driving straight, right turn, and left turn. Moreover, we extended our analysis to encompass comparisons among all agents, not just the target agent. Data: Argoverse validation set

method	Maneuver	Target	minADE (\downarrow)	minFDE (\downarrow)	MR (\downarrow)
Base [35]	All	Target agent	0.687	1.030	0.1024
	Straight	Target agent	0.615	0.857	0.0691
	Right-turn	Target agent	1.039	1.925	0.2700
	Left-turn	Target agent	1.045	1.860	0.2674
	All	All	1.070	2.071	0.3229
Ours	All	Target agent	0.676(1.60% \downarrow)	1.008(2.14% \downarrow)	0.0987(3.61% \downarrow)
	Straight	Target agent	0.605(1.63% \downarrow)	0.839(2.10% \downarrow)	0.0669(3.18% \downarrow)
	Right-turn	Target agent	1.024(1.44% \downarrow)	1.881(2.29% \downarrow)	0.2537(6.04% \downarrow)
	Left-turn	Target agent	1.029(1.53% \downarrow)	1.821(2.10% \downarrow)	0.2581(3.48% \downarrow)
	All	All	1.035(3.27% \downarrow)	1.909(7.82% \downarrow)	0.3072(4.86% \downarrow)

4.6 Effect on Different Maneuvers and Overall Agent Prediction

Further, in Table 2, we evaluate the performance in terms of three different maneuvers, driving straight, right turn, and left turn (where turning requires more complicated agent-wise interactions). Our model shows an improved performance in all scenarios. In comparison to solely predicting the behavior of the target agent, we observed performance improvements of 1.60%, 2.14%, and 3.61% for the minADE, minFDE, and MR metrics, respectively. When contrasted with predictions made for all agents, the improvements were even more pronounced, standing at 3.27%, 7.82%, and 4.86% for the same metrics. These

Table 3. Comparison of trajectory prediction performance based on LaneGCN [14] and HiVT [35]. Each model uses 128dim, 64dim. We also provide our ablation study without our proposed two modules: (i) Lane-guided Attention Module (LAM) and (ii) the use of attentional knowledge distillation loss $\mathcal{L}_{\text{distill}}$. We also represent performance of the Teacher network. Data: Argoverse validation set. Inference times are reported in milliseconds (msec), measured based on 12 agents using a single RTX 3090 Ti GPU

Base Model	Distillation	LAM	Teacher	minADE (\downarrow)	minFDE (\downarrow)	MR (\downarrow)	Time(ms) (\downarrow)
LaneGCN [14]				0.7118	1.075	0.1030	37
	V			0.7067	1.059	0.0968	37
HiVT [35]				0.6868	1.030	0.1024	35
	V			0.6830	1.013	0.1005	35
	V	V		0.6804	1.014	0.0995	42
		V		0.6758	1.008	0.0987	42
LaneGCN [14]			V	0.6494	0.9683	0.0888	41
HiVT [35]			V	0.6270	0.9210	0.0891	40

results demonstrate that our model can extract robust results by leveraging interaction information with other agents even with some information loss for the agent being predicted. In addition, our model shows distinct performance improvement in all maneuvers: straight, right-turn, and left-turn.

4.7 Ablation Study

Our model learns which agents should interact with the target agent by utilizing a teacher network that leverages the future trajectories of other agents. The student model is then trained to mimic this interaction knowledge from the teacher model. Table 3 presents the performance of the teacher model, demonstrating that accurately predicting interactions with other agents can significantly enhance overall trajectory prediction performance. This emphasizes the importance of accurate interaction prediction with other agents and demonstrates why the student model should learn interaction knowledge from the teacher model. Importantly, the student model is trained to accurately predict interactions among agents using the distillation method without additional network complexity, leading to performance improvements without incurring extra computational costs during the inference phase (see the Distillation and Time(ms) columns). Furthermore, in Table 3, when using the LAM (see LAM column), it is confirmed that each agent’s predicted trajectories can be refined using localized confidence-augmented lane information, and that using the information improves performance. In the 6-th row, it is shown that there is a more enhancement in performance when both modules are employed simultaneously. This demonstrates that leveraging the LAM to refine the predicted trajectory and then mimicking the teacher’s attention focus coefficient helps improve performance. Additionally, by focusing only on interactions with highly relevant lanes with the agent, rather than considering interactions with all lanes, we achieved performance improvements with minimal additional cost. We conducted experiments on both LaneGCN [14] and HiVT [35], and we observed that each module contributes to its own performance gain.

4.8 Analyzing Distillation Loss Weight Ratios

To investigate the impact of different weight ratios between the loss terms, we conducted a series of experiments. Specifically, we varied the weight of $\mathcal{L}_{\text{distill}}$ by factors of 0.2, 0.5, 1.0 and 2.0. As shown in Table 4, our experiments demonstrated that

Table 4. Comparison different distillation loss weight ratios based on HiVT [35]

Distillation loss weight ratio	minADE (\downarrow)	minFDE (\downarrow)	MR (\downarrow)
0.0 (Base [35])	0.6868	1.030	0.1024
0.2	0.6888	1.020	0.1007
0.5	0.6916	1.025	0.1013
1.0	0.6830	1.013	0.1005
2.0	0.6873	1.022	0.1018

a 1:1 ratio between the two loss terms provides the most balanced and optimal performance. This finding led us to adopt the 1:1 ratio in our main experiments.

5 Conclusion

Accurately predicting the future movements of surrounding traffic agents remains challenging for fully-autonomous driving. In this paper, we demonstrate that future information can be effectively utilized during training with the help of teacher-student technique and attention-based knowledge distillation. This enables the model to effectively learn which agents should be interacted with target agents. Also, our proposed Lane-guided Attention Module (LAM) enhances the transfer of attention coefficients from the network to the student network by bridging the information gap between both models. Our model generally outperforms baseline models on the Argoverse Motion Forecasting dataset, effectively reducing uncertainty and resulting in improved interaction predictions.

Acknowledgement. This work was supported by 42dot. Also, this work was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(NRF-2021R1A6A1A13044830, 15%) and supported by Institute of Information & communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT) (RS-2022-II220043, Adaptive Personality for Intelligent Agents, 15%, IITP-2024-RS-2024-00397085, Leading Generative AI Human Resources Development, 15%).

References

1. Beyer, L., Zhai, X., Royer, A., Markeeva, L., Anil, R., Kolesnikov, A.: Knowledge distillation: a good teacher is patient and consistent. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10925–10934 (2022)
2. Chai, Y., Sapp, B., Bansal, M., Anguelov, D.: Multipath: multiple probabilistic anchor trajectory hypotheses for behavior prediction. arXiv preprint [arXiv:1910.05449](https://arxiv.org/abs/1910.05449) (2019)
3. Chang, M.F., et al.: Argoverse: 3D tracking and forecasting with rich maps. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8748–8757 (2019)
4. Cho, J.H., Hariharan, B.: On the efficacy of knowledge distillation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4794–4802 (2019)
5. 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)
6. Dosovitskiy, A., et al.: An image is worth 16×16 words: transformers for image recognition at scale. In: ICLR (2021)
7. Gao, J., et al.: VectorNet: encoding HD maps and agent dynamics from vectorized representation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11525–11533 (2020)
8. Gilles, T., Sabatini, S., Tsishkou, D., Stanculescu, B., Moutarde, F.: THOMAS: trajectory heatmap output with learned multi-agent sampling. arXiv preprint [arXiv:2110.06607](https://arxiv.org/abs/2110.06607) (2021)

9. Girgis, R., et al.: Latent variable sequential set transformers for joint multi-agent motion prediction. arXiv preprint [arXiv:2104.00563](https://arxiv.org/abs/2104.00563) (2021)
10. Gou, J., Yu, B., Maybank, S.J., Tao, D.: Knowledge distillation: a survey. *Int. J. Comput. Vision* **129**, 1789–1819 (2021)
11. Gu, J., Sun, C., Zhao, H.: DenseTNT: end-to-end trajectory prediction from dense goal sets. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 15303–15312 (2021)
12. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. arXiv preprint [arXiv:1503.02531](https://arxiv.org/abs/1503.02531) (2015)
13. Kim, Y., Rush, A.M.: Sequence-level knowledge distillation. arXiv preprint [arXiv:1606.07947](https://arxiv.org/abs/1606.07947) (2016)
14. Liang, M., Yang, B., Hu, R., Chen, Y., Liao, R., Feng, S., Urtasun, R.: Learning lane graph representations for motion forecasting. In: ECCV 2020, Part II, pp. 541–556. Springer (2020)
15. Liu, Y., Zhang, J., Fang, L., Jiang, Q., Zhou, B.: Multimodal motion prediction with stacked transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7577–7586 (2021)
16. Loshchilov, I., Hutter, F.: SGDR: stochastic gradient descent with warm restarts. arXiv preprint [arXiv:1608.03983](https://arxiv.org/abs/1608.03983) (2016)
17. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. arXiv preprint [arXiv:1711.05101](https://arxiv.org/abs/1711.05101) (2017)
18. Mirzadeh, S.I., Farajtabar, M., Li, A., Levine, N., Matsukawa, A., Ghasemzadeh, H.: Improved knowledge distillation via teacher assistant. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 5191–5198 (2020)
19. Monti, A., Porrello, A., Calderara, S., Coscia, P., Ballan, L., Cucchiara, R.: How many observations are enough? Knowledge distillation for trajectory forecasting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6553–6562 (2022)
20. Nayakanti, N., Al-Rfou, R., Zhou, A., Goel, K., Refaat, K.S., Sapp, B.: Wayformer: Motion forecasting via simple & efficient attention networks. In: 2023 IEEE International Conference on Robotics and Automation (ICRA), pp. 2980–2987. IEEE (2023)
21. Ngiam, J., et al.: Scene transformer: a unified architecture for predicting future trajectories of multiple agents. In: International Conference on Learning Representations (2021)
22. Park, W., Kim, D., Lu, Y., Cho, M.: Relational knowledge distillation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3967–3976 (2019)
23. Phuong, M., Lampert, C.: Towards understanding knowledge distillation. In: International Conference on Machine Learning, pp. 5142–5151. PMLR (2019)
24. Salzmann, T., Ivanovic, B., Chakravarty, P., Pavone, M.: Trajectron++: dynamically-feasible trajectory forecasting with heterogeneous data. In: ECCV 2020, Part XVIII, pp. 683–700. Springer (2020)
25. Sheng, Z., Xu, Y., Xue, S., Li, D.: Graph-based spatial-temporal convolutional network for vehicle trajectory prediction in autonomous driving. *IEEE Trans. Intell. Transp. Syst.* **23**(10), 17654–17665 (2022)
26. Shi, S., Jiang, L., Dai, D., Schiele, B.: MTR-A: 1st place solution for 2022 Waymo open dataset challenge—motion prediction. arXiv preprint [arXiv:2209.10033](https://arxiv.org/abs/2209.10033) (2022)
27. Su, D.A., Douillard, B., Al-Rfou, R., Park, C., Sapp, B.: Narrowing the coordinate-frame gap in behavior prediction models: Distillation for efficient and accurate

- scene-centric motion forecasting. In: 2022 International Conference on Robotics and Automation (ICRA), pp. 653–659. IEEE (2022)
28. Sun, Q., Huang, X., Gu, J., Williams, B.C., Zhao, H.: M2I: from factored marginal trajectory prediction to interactive prediction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6543–6552 (2022)
 29. Tang, J., et al.: Understanding and improving knowledge distillation. arXiv preprint [arXiv:2002.03532](https://arxiv.org/abs/2002.03532) (2020)
 30. Varadarajan, B., et al.: Multipath++: efficient information fusion and trajectory aggregation for behavior prediction. In: 2022 International Conference on Robotics and Automation (ICRA), pp. 7814–7821. IEEE (2022)
 31. Wang, M., et al.: GANet: goal area network for motion forecasting. In: 2023 IEEE International Conference on Robotics and Automation (ICRA), pp. 1609–1615. IEEE (2023)
 32. Wang, W., Wei, F., Dong, L., Bao, H., Yang, N., Zhou, M.: MiniLM: deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Adv. Neural. Inf. Process. Syst.* **33**, 5776–5788 (2020)
 33. Zhang, L., Li, P., Chen, J., Shen, S.: Trajectory prediction with graph-based dual-scale context fusion. In: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 11374–11381. IEEE (2022)
 34. Zhao, H., et al.: TNT: target-driven trajectory prediction. In: Conference on Robot Learning, pp. 895–904. PMLR (2021)
 35. Zhou, Z., Ye, L., Wang, J., Wu, K., Lu, K.: HiVT: hierarchical vector transformer for multi-agent motion prediction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8823–8833 (2022)



VA-OCC : Enhancing Occupancy Dataset Based on Visible Area for Autonomous Driving

Yang Li¹, Weng Feng¹, Ge Gao^(✉)¹, Jun Chang^(✉)¹, and Ming Li^(✉)¹

School of Computer Science, WuHan University, Wuhan 430072, China
{gaoge, changjun, liming}@whu.edu.cn

Abstract. In the field of autonomous driving, the importance of the occupancy grid data structure cannot be ignored. The occupancy grid has advantages such as reducing data complexity, improving computational efficiency, and facilitating path planning. By constructing an accurate occupancy grid dataset, researchers can better understand and analyze the distribution of objects in the environment, providing strong support for tasks such as object detection and path planning. This paper proposes a new method for constructing an occupancy dataset, which first constructs dense voxels based on point cloud data, then extracts semantics through two methods, and finally filters the grid based on the visible area to obtain the ground truth of the Occupancy dataset (Named as VA-OCC dataset.). By replacing the existing dataset in the paper with the VA-OCC dataset, better IOU scores and visualization effects can be achieved.

Keywords: Autonomous Driving · Occupancy Grid Dataset

1 Introduction

Occupancy Grid is a commonly used environmental modeling method in the field of autonomous driving. It divides the environment into small grids (or cells) and tracks the state of each cell to achieve perception and understanding of the environment.

In the field of autonomous driving, Occupancy Grid technology has shown unique advantages. Firstly, it can provide comprehensive and detailed environmental representation, helping vehicles to perceive and understand the surrounding environment more accurately. Secondly, Occupancy Grid technology has stronger robustness against occlusion. In real-life scenarios, vehicles may encounter obstacles such as buildings and trees, which can affect the accuracy of perception. However, with Occupancy Grid technology, vehicles can still perceive the surrounding environment relatively completely, avoiding information loss caused by occlusions. In addition, Occupancy Grid technology also promotes more efficient sensor fusion.

Existing Occupancy Grid ground truth datasets are generated based on existing datasets containing full semantic segmentation point cloud ground truth. However, the manual and time costs of annotating full semantic segmentation point cloud ground truth are high and may not be suitable for all cases. This paper proposes a complete pathway to construct Occupancy Grid ground truth datasets, which can extract semantics from full semantic segmentation point cloud ground truth and from full semantic segmentation 2D camera surround view images.

Furthermore, in practical scenarios, drivers can only observe a part of objects, which can be referred to as the foreground within the foreground. Unlike existing datasets, VA-OCC dataset retains only the part of the foreground within the foreground, enabling the model to focus more on the areas that are truly observable and have an impact on autonomous driving during training. This significantly improves the accuracy of occupancy prediction and lays a solid foundation for the next steps in planning and control.

The main contributions of this paper are:

1. Proposed a complete pathway for constructing VA-OCC dataset, including a new method for assigning semantics to occupancy voxels.
2. Utilized visual selection to generate Occupancy Grid ground truth that better reflects the actual observation conditions of human drivers in autonomous driving scenarios.
3. Demonstrated through experiments the superiority of the datasets generated using this method.

2 Related Work

2.1 3D and 2D Semantic Segmentation Datasets

[1–3] have road scene point cloud ground truth data with full point cloud annotations, containing rich semantic information. While datasets like [2, 4] provide multi-camera surround view image data, they lack 2D semantic segmentation annotations for these images. [5–7] are several widely used 2D semantic segmentation datasets, but they do not include annotations for surround view images.

2.2 Occupancy Dataset

[8] integrates multiple frames of LiDAR point clouds, employs Poisson reconstruction to fill holes, and voxelizes the grid to obtain dense occupancy labels. [9] addresses the issue of some occupancy labels being overlooked due to the sparsity of point clouds, and introduces the Augmentation and Purification (AAP) pipeline for densification of annotations, requiring approximately 4000 h of manual work. [10] additionally generates LiDAR visibility masks and camera visibility masks to indicate whether each voxel is observed in the current LiDAR or multi-camera view.

3 Methodology

VA-OCC dataset requires a point cloud dataset as the data source, and the dataset used in this paper is the nuscnets dataset (Fig. 1).

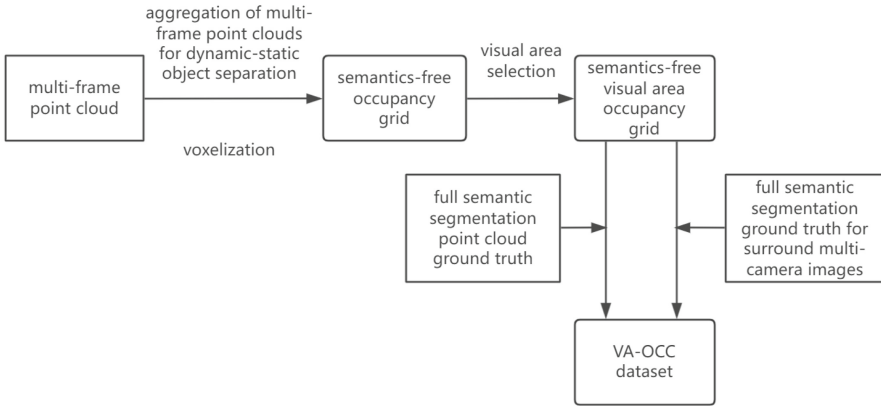


Fig. 1. VA-OCC dataset construction process.

3.1 Generating Dense Occupied Grids

In the context of autonomous driving scenarios, due to the sparsity of a single frame of LiDAR point cloud, in order to generate dense and distinguishable occupied grids, multiple frames of point clouds need to be aggregated first.

In each scene, dynamic objects and static backgrounds are separated based on the 3D object bounding box true values in each frame. For the static background, the static background of each frame is transformed to the vehicle coordinate system of the first frame of the scenario and aggregated to obtain a dense static background point cloud of the entire scene. For dynamic objects, each object is identified with a unique token (the token remains the same when the same object appears in multiple frames). By traversing the entire scene and aggregating the point clouds of the same object in different frames based on the token, dense point clouds for each object are obtained.

After the separation is completed, the dense static background point cloud of the scene is first transformed to the current frame based on the ego-position. Then, the point cloud of the corresponding object is placed based on the object token list obtained from the dataset, points outside the set perception range are removed, further repair is done through Poisson reconstruction (mainly to repair road surface holes), and finally, Gaussian smoothing is applied to flatten the road surface to obtain a complete dense point cloud for the current frame. A voxel grid of size (200,200,16) is generated with a voxel size of 0.5 m.

3.2 Visual Area Selection

Recently proposed occupancy prediction models often take images as input, where only a part of the object is visible in the image (i.e., the visual area). Given this reality, it is unreasonable to calculate loss based on the complete occupancy as the ground truth; moreover, distant buildings, plants, and objects with other large obstacles between them and the ego-vehicle have less impact on the vehicle’s movement. Filtering out these parts when constructing the VA-OCC dataset allows the model to focus more on the parts that affect the vehicle’s movement during training. Therefore, we propose a visual area selection algorithm to further process the occupancy.

Taking the voxel at the driver’s head position as the origin, a ray is emitted to each voxel in the occupied space. The occupancy of each voxel is judged every unit distance along each ray, and only the first occupied voxel encountered on each ray is recorded. The recorded voxels are then consolidated to obtain a preliminary visual occupancy ground truth without semantics.

3.3 Semantic Assignment to Occupancy

There are two methods to obtain semantic information based on 3D or 2D available data.

3D Semantics. If has fully semantic segmented point cloud ground truth, semantics can be assigned to the occupancy ground truth using the nearest neighbor algorithm.

2D Semantics. Because the occupied ground truth of VA-OCC dataset only considers the visual surface, no need to consider the depth of the inner voxel, if there are fully surrounding semantic segmented multi-camera image ground truth, the centroids of the voxels can be projected to the corresponding image positions using the transformation matrix from point cloud to image, and semantic information can be assigned to the voxels at those positions.

3.4 Scene Completion and Refinement

Due to the large voxel size, the completeness of the road surface will be greatly affected after visual area selection. Therefore, it is necessary to complete the road surface and prevent the removal of valid information in the subsequent connected component detection. Finally, a connectivity testing is performed starting from the road surface where the ego-vehicle is located using breadth-first search (BFS), removing objects that only have their upper half remaining and distant plants, buildings, and backgrounds that do not affect the vehicle’s movement (see Fig. 2).

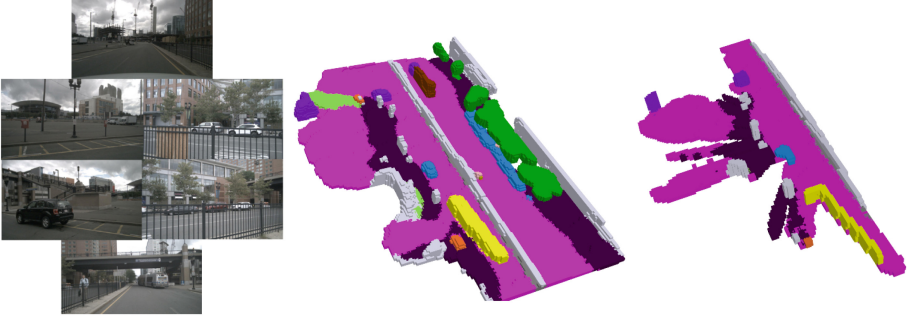


Fig. 2. Original six camera images (left), inference results of the model trained on the SurroundOcc dataset (middle), inference results of the model trained on the VA-OCC dataset (right).

4 Experiments

4.1 2D Semantic Segmentation Ground Truth Construction

Datasets such as [1, 2] contain 3D full point cloud semantic segmentation ground truth data, with [1] proposing a complete annotation process. However, there is currently no complete ground truth data for surrounding multi-camera 2D top-view semantic segmentation in existing datasets. Existing AI annotation methods such as [11–13] lack accuracy, while manual annotation is costly. In order to efficiently demonstrate the effectiveness of the proposed method in this paper, a method for generating 2D full semantic segmentation ground truth based on high-quality occupancy ground truth is proposed. Firstly, based on the original images and the transformation matrix from LiDAR to the camera, the direction vector of each pixel in the image corresponding to the point cloud space is obtained. Then, this vector is transformed into occupancy space. Finally, starting from the camera position along this vector, the semantic of the first encountered voxel with semantics is assigned to the corresponding pixel. After traversing each pixel, complete ground truth data for surrounding multi-camera 2D top-view semantic segmentation can be obtained (see Fig. 3).

4.2 Visualization Processing with CUDA Acceleration

By emitting rays from the camera’s optical center to the voxel grid ground truth, we can detect which voxels in the camera’s field of view are completely occluded. These occluded voxels are then marked as “free” status, allowing us to visualize the camera view of the occupancy grid. However, this process requires traversing a large number of ray directions and determining whether they intersect with voxels with semantics based on the distance traversed by the rays, leading to significant time consumption. By parallelizing this process using CUDA programming, assigning each CUDA core to traverse one ray, the required processing time



Fig. 3. The full semantic segmentation ground truth for camera images constructed according to the method in this article, the semantic free part in the figure is both the part that exceeds the occupied grid range.

can be significantly reduced. This parallel operation among cores does not interfere with each other. Through testing, using CUDA acceleration to process the occupancy grid ground truth results in a speed improvement of approximately 100 times compared to traditional serial processing, greatly improving processing efficiency.

4.3 Training Results and Comparison

The baseline model used in the comparative experiment is SurroundOcc. In order to highlight the superiority of our dataset, a voxel size of 0.5 m was used for horizontal comparison in the experiment. Training was conducted using VA-OCC dataset replacing the dataset used in the original paper (see Fig. 4). For evaluation metrics, just like the original paper, we use the intersection over union (IoU) of occupied voxels, ignoring their semantic class as the evaluation metric of the scene completion (SC) task and the mIoU of all semantic classes for the SSC task.

$$\text{IoU} = \frac{TP}{TP + FP + FN}$$

$$\text{mIoU} = \frac{1}{C} \sum_{i=1}^C \frac{TP_i}{TP_i + FP_i + FN_i}$$

where TP, FP, FN indicate the number of true positive, false positive, and false negative predictions. C is the class number.

This paper validates the effectiveness of the VA-OCC dataset based on two benchmark algorithms, SurroundOcc and OCC3D. The OCC3D dataset is widely used by most occupancy grid algorithms at present. (see Table 1) Training on our dataset improved IoU by 41.5% compared to the SurroundOcc dataset, and by 10.9% compared to the more general OCC3D dataset.

Table 1. Replacing the dataset in SurroundOcc and OCC3D with the VA-OCC dataset (* indicates the use of the VA-OCC dataset) resulted in an improvement in the IoU evaluation metrics.

	SurroundOcc	SurroundOcc*	OCC3D	OCC3D*
IoU	31.49	44.56	41.75	46.32
barrier	20.59	30.26	39.33	42.12
bicycle	11.68	14.56	20.56	22.38
bus	28.06	31.45	38.29	42.11
car	30.86	39.22	42.24	45.86
construction-vehicle	10.20	8.45	16.93	16.65
motorcycle	15.14	16.22	24.52	23.67
pedestrian	14.09	16.39	22.72	24.75
traffic-cone	12.06	20.17	21.05	22.93
trailer	14.38	14.56	31.11	32.37
driveable-surface	37.29	51.43	53.33	58.19
other-flat	23.20	27.80	33.84	34.58
sidewalk	24.49	31.95	37.98	41.74
terrain	22.77	27.28	33.23	35.11
manmade	14.86	18.35	20.79	22.31
vegetation	21.86	18.35	18.01	19.65
mIoU	20.30	30.34	28.53	33.53

4.4 Road Completion Comparison Experiment

Due to the relatively large voxel size, occlusion between voxels is more severe, especially when vehicles are on slopes, which is particularly evident on the road surface. After visualization filtering, the continuity of the road surface is severely disrupted, with many gaps in the road surface. This can lead to incorrect guidance to the model during training, although the model can fill in the gaps during prediction, it still results in a decrease in model performance(see Table 2).

4.5 Connectivity Screening Comparison Experiment

In scenes where the lower half of vehicles, pedestrians, foreground objects, plants, buildings, and other background objects are occluded, after visualization filtering, the lower parts are omitted due to occlusion, leaving only the higher parts. These lower parts of objects do not actually affect the vehicle’s driving, and training the model to complete them is meaningless. Starting from the road surface at the self-position, connectivity filtering is performed using the BFS algorithm, retaining only the connected parts. As shown in the Table 2, this approach allows the model to focus on the parts that truly affect the vehicle’s driving, improving prediction accuracy (see Fig. 5).

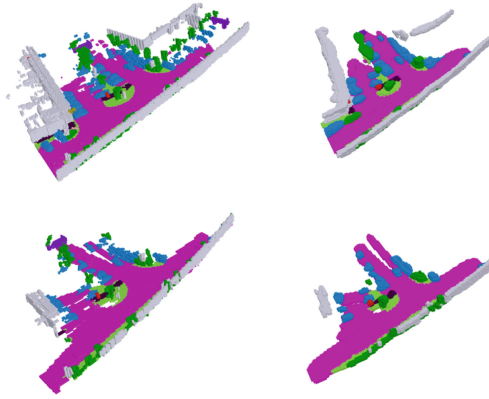


Fig. 4. The upper and lower rows respectively show the ground truth and model training results of the original dataset and the dataset used in this paper. The model trained on the dataset in this paper achieves higher inference accuracy while inferring valid information, as evidenced by the clearer vehicles in the inference results shown in the figure below.

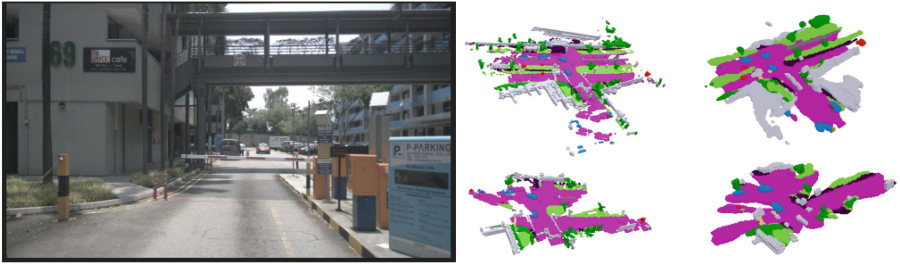


Fig. 5. The image on the left shows the rear-view camera image of the vehicle, indicating that the lifting pole is obstructing the rear road. It can be observed from the comparison in the right part that the model trained based on the dataset in this paper (below) is able to more clearly identify the drivable area and objects that affect the vehicle's movement

Table 2. Effects of various improvement strategies on experimental results

Visual-Area Filtering	Road completion	Connectivity Screening	IoU	driveable-surface
			31.49	37.29
✓			38.67	41.95
✓	✓		40.37	50.15
✓	✓	✓	44.56	51.43

5 Conclusion

This paper proposes a novel approach to construct a complete occupancy grid dataset. Building upon existing methods, a new method is introduced to extract semantic information from 2D semantic segmentation for occupancy semantic acquisition. The effectiveness of occupancy ground truth visualization is further enhanced through visual filtering and road surface completion methods. By replacing the existing dataset with the VA-OCC dataset, better IOU indices and visualization effects can be achieved.

Funding. This work was supported by the Key RD Program in Hubei Province(Grant No. 2022BAA079) and the Key Project of Hubei Province (Grant No. 2021BAA179).

References

1. Behley, J., et al.: SemanticKITTI: a dataset for semantic scene understanding of LiDAR sequences. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV) (2019)
2. Caesar, H., et al.: nuScenes: a multimodal dataset for autonomous driving. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2020)
3. Hackel, T., Savinov, N., Ladicky, L., Wegner, J.D., Schindler, K., Pollefeys, M.: Semantic3D.net: a new large-scale point cloud classification benchmark. In: ISPRS, pp. 91-98 (2017)
4. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? The KITTI vision benchmark suite. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition (2012)
5. Pascal VOC. <http://host.robots.ox.ac.uk/pascal/voc/>
6. Cordts, M., et al.: The cityscapes dataset for semantic urban scene understanding. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (016)
7. ADE20K. <https://groups.csail.mit.edu/vision/datasets/ade20k/index.html>
8. Wei, Y., Zhao, L., Zheng, W., Zhu, Z., Zhou, J., Lu, J.: SurroundOcc: multi-camera 3D occupancy prediction for autonomous driving. In: ICCV (2023)
9. Wang, X., et al.: OpenOccupancy : a large scale benchmark for surrounding semantic occupancy perception (2023)
10. Tian, X., Jiang, T., Yun, L., Wang, Y., Wang, Y., Zhao, H.: Occ3D: a large-scale 3D occupancy prediction benchmark for autonomous driving (2023)
11. Kirillov, A., et al.: Segment anything. In: ICCV (2023)
12. Liang, F., et al.: Open-vocabulary semantic segmentation with mask-adapted clip. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7061–7070 (2023)
13. Xiong, Y., et al.: EfficientSAM: leveraged masked image pretraining for efficient segment anything (2023)



Large Models in Dialogue for Active Perception and Anomaly Detection

Tzoulio Chamiti¹, Nikolaos Passalis²(✉), and Anastasios Tefas¹

¹ Computational Intelligence and Deep Learning Group, AIIA Laboratory,
Department of Informatics, Aristotle University of Thessaloniki, 541 24 Thessaloniki,
Greece

{t.chamiti,tefas}@csd.auth.gr

² Department of Chemical Engineering, Aristotle University of Thessaloniki,
541 24 Thessaloniki, Greece
passalis@auth.gr

Abstract. Autonomous aerial monitoring is an important task aimed at gathering information from areas that may not be easily accessible by humans. At the same time, this task often requires recognizing anomalies from a significant distance and/or not previously encountered in the past. In this paper, we propose a novel framework that leverages the advanced capabilities provided by Large Language Models (LLMs) to actively collect information and perform anomaly detection in novel scenes. To this end, we propose an LLM-based model dialogue approach, in which two deep learning models engage in a dialogue to actively control a drone to increase perception and anomaly detection accuracy. We conduct our experiments in a high fidelity simulation environment where an LLM is provided with a predetermined set of natural language movement commands mapped into executable code functions. Additionally, we deploy a multimodal Visual Question Answering (VQA) model charged with the task of visual question answering and captioning. By engaging the two models in conversation, the LLM asks exploratory questions while simultaneously flying a drone into different parts of the scene, providing a novel way to implement active perception. By leveraging LLM's reasoning ability, we output an improved detailed description of the scene going beyond existing static perception approaches. In addition to information gathering, our approach is utilized for anomaly detection and our results demonstrate the proposed method's effectiveness in informing and alerting about potential hazards.

Keywords: Active Anomaly Detection · LLM · VQA · Aerial Monitoring

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-78447-7_25.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
A. Antonacopoulos et al. (Eds.): ICPR 2024, LNCS 15317, pp. 371–386, 2025.
https://doi.org/10.1007/978-3-031-78447-7_25

1 Introduction

In the last few years, drones have witnessed numerous technological advancements, as well as great commercial exposure for their ability to perform difficult tasks, such as surveillance, anomaly detection, and aerial monitoring in challenging environments. To effectively support these tasks and ensure the efficient and autonomous operation of robots, large informative datasets, e.g., containing drone images, action states, and/or anomalies, were necessary in order to cover every possible scenario that could occur [1-3]. These approaches primarily focused on collecting a large quantity of data and employing different learning techniques to detect possible anomalies in autonomous drone flying scenarios.

With the major advancements in deep learning across numerous domains, there have been multiple attempts to incorporate these modern, more effective technologies for the sake of enhancing autonomous systems' efficiency and capability. By deploying larger, more advanced deep learning models a substantial improvement in performance was witnessed [4,5]. Nevertheless, these methods lack the ability to *actively perceive* the scene in order to issue the appropriate control commands and further improve the perception accuracy based on the current conditions. Such active perception approaches have shown promising

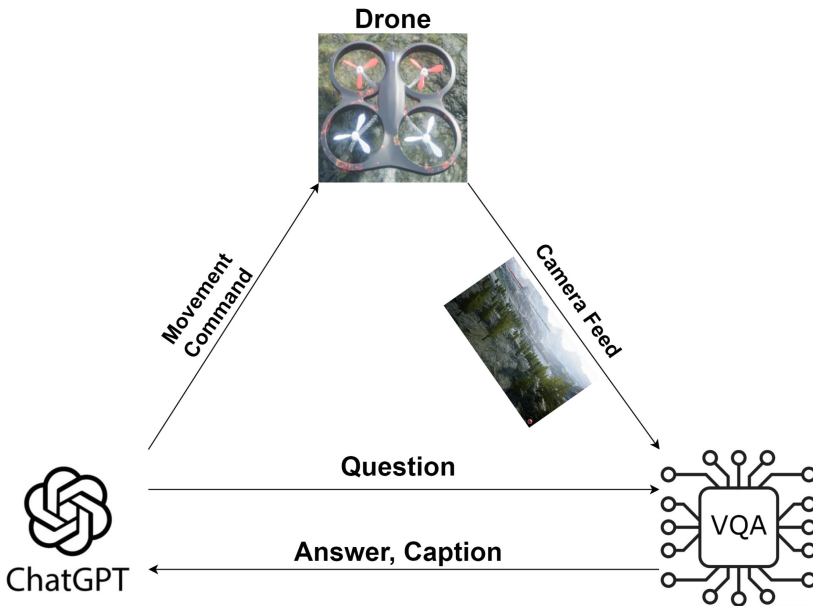


Fig. 1. Overview of the proposed model dialogue approach. First a drone captures an image. This image, along with an appropriate question, is fed to the employed VQA model. Then, the VQA model provides a response that is fed to the LLM model which in turn issues a movement command and a new exploratory question.

results in other relevant domains in recent years [6–8]. However, it is not trivial to implement such methods in open-world setups.

The main contribution of this paper is a novel approach for active perception and anomaly detection that leverages the capabilities of recent Large Language Models (LLMs) by developing a *model dialogue* approach in which two deep learning models interact in order to continuously improve the final prediction. To this end, we equip the employed LLM with complete navigational control through a set of specific textual commands that ultimately navigate a drone in real time, implementing an active perception scheme in which the drone explores the scene and exploits potential hazardous scenarios and anomalies. Furthermore, we incorporate a Visual Question Answering (VQA) model in order to engage the two models in interactive conversation from which the LLM acts as a controller that can extract meaningful textual information about the unknown scene in which the drone operates. Our goal is to provide a detailed description of the scene gathered throughout the conversation along with explanations that led to these decisions. This dialogue process leads to an active perception pipeline in which we can gather additional information about the scene, as well as validate the scene details. The conducted experimental evaluation shows that the proposed approach can indeed enable a drone to successfully navigate an unknown open environment and provide an explainable and detailed description of the scene in a zero-shot fashion, as well as detect anomalies and output potential safety measures in response to potentially hazardous observations. The code used for the conducted experiments, including detailed prompts and experimental results are provided at https://github.com/Tzoulio/Large_Models_Dialogue_for_Active_Perception.

The rest of the paper is structured as follows. Section 2 introduces the related work, while the proposed method is presented in Sect. 3. The experimental evaluation is provided in Sect. 4, while Sect. 5 concludes the paper.

2 Related Work

The task of Visual Question Answering [9] has increased in popularity in recent years, with the ability to combine computer vision with Natural Language Processing (NLP) resulting in a system that can process two types of different modalities at the same time. Such an ability is crucial in robotics applications considering they are often applied to scenarios and environments that require handling such multimodal data. By giving a robot the ability to process multiple data together at once, they increase the quality and quantity of information they acquire, which in turn expands their overall knowledge of the world. As a result, there have been multiple attempts at applying VQA in robotics. Some works focus on having the robot interact with the environment and come up with an answer to a specific question, mimicking the VQA task. Deng et al. [10] uses VQA in a robotic manipulation scenario. They train a Deep Q Network (DQN) and through reinforcement learning teach the robot to continuously manipulate objects until they come up with the right answer. In [11] a Hierarchical Interactive Memory Network (HIMN) was deployed as a controller that allows the

system to store and retrieve information hierarchically in the form of memory and enables the robot to provide an answer by interacting with its environment in real-time. EmbodiedQA [12] is another approach that deploys a robot in an unknown environment in which the robot learns to navigate through using imitation learning and ultimately gathers the appropriate information to answer the question. Our work leverages the recent advances in VQA as a fundamental part of the proposed pipeline by employing a VQA model which acts as the *sensing* model, which processes the data acquired from the world and answers questions regarding these.

After the breakthrough that LLMs made in the field of AI, researchers have been constantly finding ways to utilize them in robotic applications. A lot of works leverage the LLMs' reasoning capabilities and language understanding ability to act as a communicator between the human operator who issues a command in natural language and the robot who executes the command in the form of code [13–16]. These approaches either directly map specific commands to code snippets that are applied on the robot directly or provide enough resources to the LLM to construct code and make specific API calls that will produce the correct result on the robot, as specified in the natural language prompt. Generally, a lot of research is focused on advancing the LLM capabilities further, by implementing different modules together with the LLM in an attempt to give it multi-modal capabilities [17–19]. This resulted in a lot of works which combined multi-modal variations of LLMs into robot task planning [20–22]. These works utilize imitation learning to teach a control agent how to perform the natural language tasks which are learned from a dataset consisting of sets of demonstrations during different timestamps. In other works, such as [13], users are able to control an aerial drone through natural language and prompt engineering. The proposed method goes beyond these approaches by employing a dialogue-based approach, in which only one model has full access to the visual modality and the other model can interact with this model through textual prompts.

The proposed method is more closely related to recent attempts to combine LLMs with VQA models. Some works [23–26] focus on initiating a conversation between the two models to enhance the VQAs ability in the captioning task. They start with a general caption of a query image and through ChatGPT's ability of understanding and generalising textual information an active dialogue between the LLM and the VQA module is initiated. During the dialogue, ChatGPT makes inquiries about possible information that the image might contain. Afterwards, the VQA model answers by confirming or denying and providing additional information for the scene. The process continues until ChatGPT outputs a detailed description containing all the knowledge it gathered through the conversation. Other methods follow a similar approach [27–29] by providing complementary knowledge to the LLM in the form of captions. This enhances the quality and flow of information, resulting in better answers and captions for the query images. Our method builds on this idea, going beyond these approaches by implementing *active perception* through the drone's navigation scheme. We collect a different image of the scene each time the drone reaches a new position.

At the same time, the employed LLM asks an exploratory question with each movement command and the VQA model provides an answer and a caption. This way, we are able to gather more information (extracted by the different captions we get in every position) as well as explore parts of the initial image that the camera could not see either by them being obscured or simply by being too far away.

3 Proposed Method

In this work, we aim to equip a drone with active perception and anomaly detection capabilities in order to provide a robust scene description, as depicted in Fig. 1. First, the drone leverages a VQA model which provides descriptions of the environment through captions. In this way, the VQA model provides a way for the LLM to “sense” the environment through text. Additionally, the VQA model outputs an image-caption matching score in order to help the LLM distinguish between good and bad captions. Then, the LLM validates the gathered textual information through the VQAs question-answering module combined with active perception and ultimately provides a generalized scene description together with explainable attention maps. The outline of the proposed approach is shown through an example in Fig. 2. This example should be used as a reference point through the description provided in this Section, since it further clarifies how the proposed method works.

For the VQA model, we incorporate the Plug-and-Play VQA (PnP-VQA) [30] framework, as shown in Fig. 3. To perform the task of image captioning, image-question pairs are processed by a pre-trained vision-language model called BLIP [31] which is also able to output a similarity score between the image and the question. The image is split into K patches and through GradCAM [32], a feature-attribution interpretability technique, they are able to provide the most relevant image patches. Finally, the image captioning module of BLIP is combined with top- k sampling to generate captions only for the relevant patches. Subsequently, the produced caption and question are fed into the question answering module to produce the answer. For the LLM, we employed the GPT3.5 as our model [33].

Let the LLM model denoted by $f(\mathbf{A}, \mathbf{C})$, which takes two distinct text sequences as input $\mathbf{A} = [A_1, A_2, \dots, A_n]$, $\mathbf{C} = [C_1, C_2, \dots, C_m]$ and outputs a response sequence $\mathbf{Q} = [Q_1, Q_2, \dots, Q_k]$, in the form of a question i.e., $\mathbf{Q} = f(\mathbf{A}, \mathbf{C})$, where \mathbf{A} denotes the answer to a previous question by the VQA model (if exists) and \mathbf{C} denotes a textual description (caption) of the current scene. In this work, we employed the GPT3.5 model to implement $f(\cdot)$, while we feed the concatenated \mathbf{A} and \mathbf{C} to the model. We assume A_i , C_i and Q_i denote the indices of words, while n , m and k denote the corresponding sequence lengths. Similarly, the VQA network $g(\mathbf{Q}, \mathbf{I})$ takes as input the output sequence of the LLM \mathbf{Q} , as well as an image \mathbf{I} , producing two different textual sequences $\mathbf{A}, \mathbf{C} = g(\mathbf{Q}, \mathbf{I})$, where \mathbf{A} is the answer to the question and \mathbf{C} denotes the caption for the image. Then, these outputs are fed to the LLM and this process repeats in an iterative fashion.

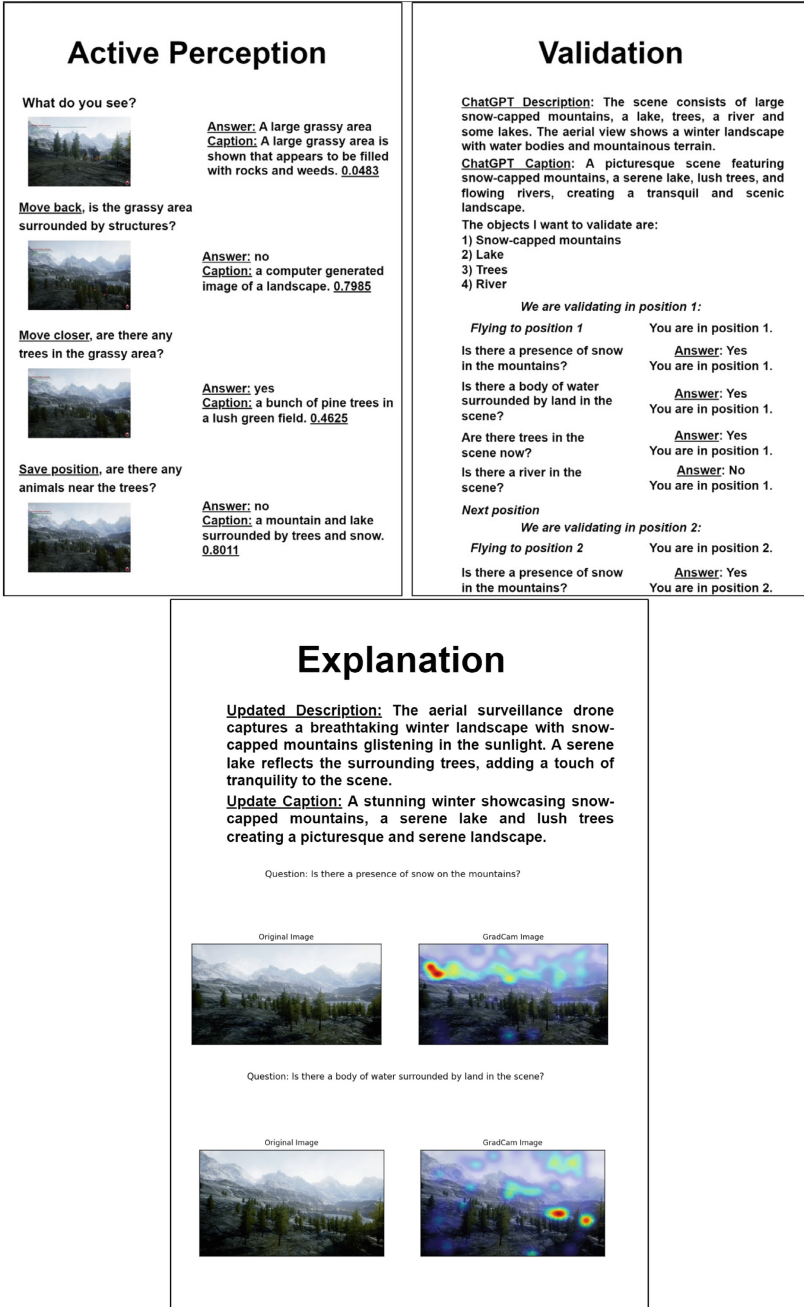


Fig. 2. A typical example of the operation of the proposed method. During active perception, the two models engage in a conversation and exchange information. In validation, a premature description and caption are chosen together and information is validated by revisiting the saved positions. Then, in the explanation mode, the final description and caption are provided together with attention maps.

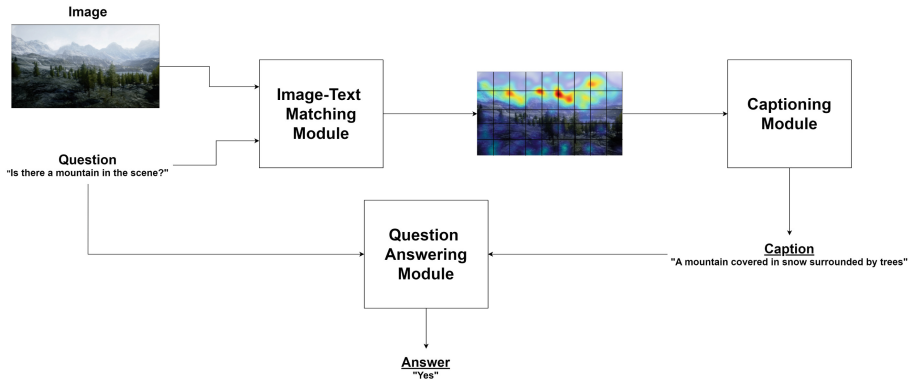


Fig. 3. The employed VQA architecture.

To grant the LLM control of the drone we first define a set of diverse functions, each one in charge of a specific navigational output. Afterwards, we provide the drone with a detailed prompt consisting of a set of commands mapped to a specific function apiece, certain rules the GPT3.5 outputs must follow, the general goal of the task and tips on how to filter and extract information from captions. Additionally, to prevent hallucination [34], i.e., imaginative and fabricated outputs from the controller, we begin the prompt by informing LLM that it is in a game scenario, the commands serve as its controls and the goal is to provide a detailed description of the observed scene while looking out for any possible anomalies that could lead to hazardous situations. The list of commands is split into:

- (i) Active perception commands
 - (a) Move closer, to move 10 m forward.
 - (b) Move back, to move 5 m backwards.
 - (c) Move right, to move 10 m to the right.
 - (d) Move left, to move 10 m to the left.
- (ii) General control commands
 - (a) Save position, to save the current position of the drone.
 - (b) Ask a question, to ask exploratory questions.
 - (c) I know enough, to return to the starting position.

Additionally, we divide the diverse list of rules the LLM must follow into:

- (i) General Rules, to make sure LLM outputs the commands and questions correctly.
- (ii) Active Perception rules, which ensure the proper movement of the drone.
- (iii) Visual Question Answering rules, in order to utilize the captions and answers as efficiently as possible and optimize the procedure.

The propose pipeline consists of the following: an *active perception* mode, a *validation* mode and an *explanation* mode. Throughout active perception mode, the drone’s camera takes snapshots of the observed scene and the controller asks questions while simultaneously issuing different movement commands. The process always starts with the question “What do you see?”. Consequently, the VQA model returns an answer, a caption and a percentage indicating if the caption matches the specific image to help distinguish between accurate and inaccurate captions. Through multiple diverse captions from different angles of the scene, the LLM model is able to gain knowledge and by leveraging its language understanding capabilities it is able to generalize and understand the context, as well as output possible safety measures for the specific scene. Then, during exploration mode, we encourage the LLM (by providing the appropriate prompt) to use the command *save position* whenever it deems it necessary in order to save the current drone position and revisit it during validation mode. The process continues until the LLM uses the command *I know enough* and transitions to validation mode.

During the validation mode, we ask the LLM to output a description and a caption of its current knowledge, along with which parts it wants to validate. We add random Gaussian noise to the saved positions, in order to gain different question-image pairs before inputting them to the VQA model again. In each new position, the controller asks one validating question for each targeted piece of information it wants to validate and we also save the question-image pairs which hold the highest matching score percentage for explanation mode. Afterwards, the controller compiles all the answers in each revisited position and leverages an ensemble approach to update the scene description and caption. In the end, the drone returns to its starting position outputting the final description, caption and the safety rules about the scene.

Finally, in order to provide the ability to explain the conclusions drawn by the developed pipeline, we extract the GradCAM’s visualization from our VQA model in order to output attention maps on the validated images, as shown in Fig. 2. As a result, when the drone returns to its starting position it is able to output the question-image pairs through an attention mask, highlighting the parts of the image that lead to its decisions on the captioning and question-answering tasks.

4 Experimental Evaluation

All the experiments were conducted using the Airsim simulation environment [35]. It is built upon Unreal Engine 4 and consists of a physics engine and different environmental, vehicular and sensory models. By testing out the quadrotor vehicular model in multiple environments we can simulate a plethora of scenarios that provide physical and visual feedback adjacent to the real world. Specifically, our experiments take place in typical surveillance environments such as a mountain landscape, a lake, a public square and a snowy road, as shown in Fig. 4.



Fig. 4. Four different environments were used for the conducted experiments: a mountain landscape, a snowy road, a public square and a lake.

To quantitatively evaluate the performance of the proposed method we compute the caption-image matching score (using the VQA model) at the drone’s spawn position and at every subsequent position revisited during the validation module. We then calculate the average caption-image matching score across all positions for ten independent experiments. The results are reported in Table 1, where we compare the baseline score (directly using the description at the starting position of the drone), and the final validated result of the proposed method (“Proposed”). Our results indicate that in different environments, the proposed method consistently enhances the caption-image matching score, suggesting that the generated captions provide more relevant information that aligns well with the scene. Furthermore, we present the average run time required, to obtain a validated, detailed scene description with explainable attention maps. Given that the average experiment time is approximately 12 min and recognizing that such a duration is impractical in hazardous situations, we introduce a special rule in our prompt. This rule stipulates that whenever the proposed method detects a potential anomaly, it must immediately stop exploration and proceed with validation and result generation. By implementing this rule, we reduce the average experiment time to under 5 min in anomaly induced scenarios.

Table 1. Average image-caption matching score (calculated over ten runs) for each of the employed environments.

Environment	Baseline	Proposed	Time of Experiment
Mountain Landscape	0.384	0.585	12 min 57 s
Public Square	0.361	0.699	12 min 28 s
Snow road	0.458	0.629	11 min 48 s
Lake	0.451	0.690	13 min 26 s

Additionally, we assess our system’s performance on the task of anomaly detection. By introducing potential hazards or dangerous elements, such as fires and car crashes, into each scene (refer to Fig. 5 for some example anomalies), we evaluate the baseline framework’s ability to accurately identify anomalies, comparing it with the performance of our proposed system following the active perception and validation phases. We consider the system successful in anomaly detection when it identifies the anomaly in its captions in a coherent and grammatically logical manner. To evaluate the proposed method in scenes that contain anomalies, we deploy hazards in three distinct scenarios. Initially, we position a potential hazard within the range of the drone’s spawn point. Subsequently, we increase the distance between the drone’s spawn point and the hazard. Finally, we place the hazard in an obscured view from the initial drone spawn point necessitating movement to locate it. We conduct the experiments ten times for each



Fig. 5. Example anomalies in the four different environments. Note that some anomalies are challenging to detect and require very careful inspection of the input frame.

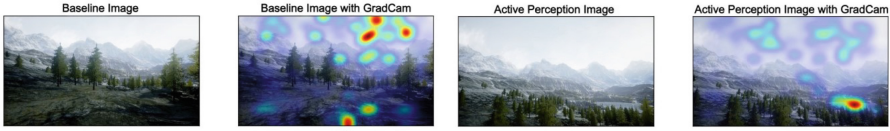
environment and present the accuracy of anomaly detection (averaging the ten runs over the three setups), comparing the baseline and the proposed method, in Table 2.

Table 2. Comparing anomaly detection accuracy between baseline and the proposed method.

Method	Environment	Anomaly Detection Score
Baseline	Mountain Landscape	0.53
Proposed	Mountain Landscape	0.90
Baseline	Public Square	0.43
Proposed	Public Square	0.73
Baseline	Lake	0.26
Proposed	Lake	0.76
Baseline	Snow	0.20
Proposed	Snow	0.83

These results indicate that the drone succeeded in providing a description and caption about the unknown scene whilst only relying on outputs from the VQA model in the form of text. Moreover, when hazardous anomalies are introduced, altering the scene to an unsafe condition, our system successfully identifies the danger and suggests necessary safety precautions. Finally, the proposed pipeline can also provide interpretable attention maps, leveraging GradCAM’s capabilities, both for the intermediate and final questions/captions, which showcase the validated information in order for a human operator to assess. Two indicative examples are shown in Fig. 6, highlighting the improved explainability capabilities provided by the proposed method. Furthermore, in Table 3, we compare the captions provided by the baseline model with the captions provided by the proposed framework and in Table 4 we provide the detailed scene descriptions leveraged by our proposed framework. Note that in most cases the proposed method leads to a more accurate description. However, hallucinations can still occur despite the validation process. Increasing the number of examination points and/or adding additional validation steps could help further reduce these occurrences.

Question: Is there a body of water surrounded by land in the scene?



Question: Is there a fire burning from flames in the scene?

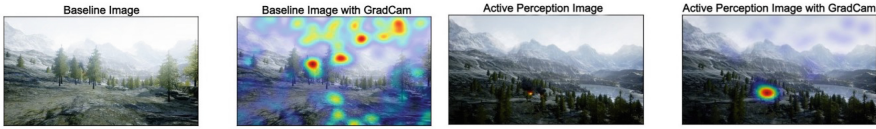


Fig. 6. Two examples for two different questions, indicating the additional explainability capabilities that can be provided by the proposed pipeline.

Table 3. Caption examples provided by the baseline VQA model and the proposed model. We highlight the correct pieces of information with the color green, the wrong ones with the color red and the ambiguous ones with orange.

Scene	Baseline	Proposed
Mountain Landscape	A view of rocky mountain peaks that looks into the horizon	A serene mountainous landscape with mist, snow-capped mountains, and trees.
Snowy road in mountainside	The snowy mountain is covered in a thick blanket of snow.	A snowy mountain with a road leading into glacier water.
Public Square	A fountain park filled with lots of water.	A lively fountain park shrouded in dense fog with water shoots creating a mysterious atmosphere
Lake	A group of tall vegetation on a river.	A tranquil lake setting with ducks, tall vegetation, and lush green plants, offering a picturesque natural landscape.
Mountain Landscape with fire	A huge flame and a cloud of black smoke.	A devastating forest fire consumes the valley, threatening the green vegetation and trees in its path.
Lake with a car fire	The steam rises in the clouds on a foggy day.	A car crash has occurred, with a truck damaged after crashing into a river emitting smoke, individuals trying to move the stuck truck.

Table 4. We showcase our methods ability to provide descriptions of the scenes, after the information was gathered through Active Perception and after it was validated through our validation module. We highlight the correct pieces of information with the color green, the wrong ones with the color red and the ambiguous ones with orange.

Environment	Proposed Final Description
Mountain Landscape	The aerial surveillance drone has captured a serene mountain landscape with trees covering its slopes. While there is no visible forest in the scene, a clear lake adds to the natural beauty of the surroundings. The absence of human activity enhances the peacefulness of the environment.
Mountain Landscape with fire	The aerial surveillance drone captures a dramatic scene with a group of mountains featuring rocky peaks in the background. In the foreground, a fire rages with red lava and flames, casting a fiery glow. On the left side, a majestic mountain stands tall, adding to the rugged landscape. Meanwhile, on the right side, another fire burns with smoke billowing into the sky. The background displays a computer artwork, adding a surreal touch to the overall view.
Snowy road in mountainside	The scene depicts a tranquil snowy landscape with no specific objects or anomalies present. The serene setting is characterized by the peacefulness of the snow-covered terrain and the absence of any notable features.
Snowy road in mountainside with car crash	The scene depicts a snowy road with a truck traveling on it. The road is covered in snow, and there is a mountain nearby covered in heavy snow. The presence of the truck on the snowy road indicates a potential hazardous situation that needs to be approached with caution.
Public Square	The scene features a round, red tiled courtyard enveloped in fog, creating an eerie and mysterious atmosphere. The fog obscures the surroundings, adding to the sense of obscurity and intrigue. The digital object, previously mentioned, is no longer present in the scene leaving behind a solitary and enigmatic courtyard.
Public Square with fire	The scene features a small fountain with water spraying, and an outdoor fountain with a fire display, and a fire torch made of metal. Both the small fountain and fire display have been confirmed to be present in the scene. The fire torch made of metal is also part of the scene, adding to the overall ambiance.
Lake	The scene portrays a tranquil river flowing with ripples at its center. Along the riverbank, the trees stand tall and healthy, framing the water's edge without any nearby structures interrupting the natural beauty. Across the river lies a park merging into a dense forest, enhancing the scene's idyllic charm. A blanket of fog envelops the surroundings, lending an air of mystery and serenity to the landscape.
Lake with fire	The scene features a body of water with a small boat floating in the middle. In front of the boat, a tree is engulfed in flames, emitting orange burning flames. The fire has spread to the bush tucker on a field with trees. However, there is no floating island engulfed by flames as previously mentioned. Smoke rises from the burning objects, creating a hazardous environment.

5 Conclusion

In this paper, we presented a novel framework that employs LLMs to actively collect information and detect anomalies, even in unprecedented situations. We propose a method where two deep learning models engage in dialogue to control a drone and improve anomaly detection accuracy. We test our approach in a realistic simulation environment, where the LLM follows natural language commands to move the drone, while a VQA model answers questions about images. By combining these models, the LLM asks questions while guiding the drone through the scene, providing a unique way to improve perception accuracy, as well as detect potential anomalies. At the same time, by leveraging the explainability capabilities of the employed VQA model, the proposed method can also further improve the explainability of the perception process. By providing four different types of scenes, with different hazardous situations in them and without requiring any fine-tuning or retraining of the models, we demonstrate the potential of the proposed method for handling open-ended adaptation in-the-wild. Additionally, to the best of our knowledge, there is currently no other established way to implement and evaluate active perception in unstructured open-world setups. Therefore, this work opens several research directions, including effective evaluation of approaches that extend beyond static perception and pave the way for applications in other areas as well.

Acknowledgements. The work presented here has been partially supported by the RoboSAPIENS project funded by the European Commission’s Horizon Europe programme under grant agreement number 101133807. This publication reflects the authors’ views only. The European Commission is not responsible for any use that may be made of the information it contains.

References









1. Mishra, B., Garg, D., Narang, P., Mishra, V.: Drone-surveillance for search and rescue in natural disaster. *Comput. Commun.* **156**, 1–10 (2020)
2. Chriki, A., Touati, H., Snoussi, H., Kamoun, F.: UAV-based surveillance system: an anomaly detection approach. In: *Proceedings of the IEEE Symposium on Computers and Communications (ISCC)*, pp. 1–6 (2020)
3. Gasparini, R., et al.: Anomaly detection, localization and classification for railway inspection. In: *Proceedings of the International Conference on Pattern Recognition (ICPR)*, pp. 3419–3426 (2021)
4. Zhai, X., Liu, K., Nash, W., Castineira, D.: Smart autopilot drone system for surface surveillance and anomaly detection via customizable deep neural network. In: *IPTC International Petroleum Technology Conference*, 14 January 2020, vol. Day 2 Tue, p. D021S053R001 (2020)
5. Unlu, E., Zenou, E., Riviere, N., Dupouy, P.E.: An autonomous drone surveillance and tracking architecture. In: *2019 Autonomous Vehicles and Machines Conference, AVM 2019*, vol. 2019, pp. 35-1–35-7 (2019)
6. Bajcsy, R., Aloimonos, Y., Tsotsos, J.K.: Revisiting active perception. *Auton. Robot.* **42**, 177–196 (2018)

7. Saito, N., Ogata, T., Funabashi, S., Mori, H., Sugano, S.: How to select and use tools?: active perception of target objects using multimodal deep learning. *IEEE Robot. Autom. Lett.* **6**(2), 2517–2524 (2021)
8. Manousis, T., Passalis, N., Tefas, A.: Enabling high-resolution pose estimation in real time using active perception. In: *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, pp. 2425–2429 (2023)
9. Agrawal, A., et al.: VQA: visual question answering. [arXiv:1505.00468](https://arxiv.org/abs/1505.00468) (2016)
10. Deng, Y., Guo, D., Guo, X., Zhang, N., Liu, H., Sun, F.: MQA: answering the question via robotic manipulation. In: *Robotics: Science and Systems XVII, RSS2021. Robotics: Science and Systems Foundation* (2021)
11. Gordon, D., Kembhavi, A., Rastegari, M., Redmon, J., Fox, D., Farhadi, A.: IQA: visual question answering in interactive environments. [arXiv:1712.03316](https://arxiv.org/abs/1712.03316) (2018)
12. Das, A., Datta, S., Gkioxari, G., Lee, S., Parikh, D., Batra, D.: Embodied question answering. [arXiv:1711.11543](https://arxiv.org/abs/1711.11543) (2017)
13. Vemprala, S., Bonatti, R., Bucker, A., Kapoor, A.: ChatGPT for robotics: design principles and model abilities. [arXiv:2306.17582](https://arxiv.org/abs/2306.17582) (2023)
14. Tazir, M.L., Mancas, M., Dutoit, T.: From words to flight: integrating OpenAI ChatGPT with px4/gazebo for natural language-based drone control. In: *Proceedings of the 13th International Workshop on Computer Science and Engineering* (2023)
15. Ye, Y., You, H., Du, J.: Improved trust in human-robot collaboration with chatGPT. [arXiv:2304.12529](https://arxiv.org/abs/2304.12529) (2023)
16. Liang, J., et al.: Code as policies: language model programs for embodied control. [arXiv:2209.07753](https://arxiv.org/abs/2209.07753) (2023)
17. Wu, C., Yin, S., Qi, W., Wang, X., Tang, Z., Duan, N.: Visual ChatGPT: talking, drawing and editing with visual foundation models. [arXiv:2303.04671](https://arxiv.org/abs/2303.04671) (2023)
18. Shen, Y., Song, K., Tan, X., Li, D., Lu, W., Zhuang, Y.: HuggingGPT: solving AI tasks with chatgpt and its friends in hugging face. [arXiv:2303.17580](https://arxiv.org/abs/2303.17580) (2023)
19. Wu, S., Fei, H., Qu, L., Ji, W., Chua, T.-S.: Next-GPT: any-to-any multimodal LLM. [arXiv:2309.05519](https://arxiv.org/abs/2309.05519) (2023)
20. Shridhar, M., Manuelli, L., Fox, D.: CLIPort: what and where pathways for robotic manipulation. [arXiv:2109.12098](https://arxiv.org/abs/2109.12098) (2021)
21. Bucker, A., Figueredo, L., Haddadin, S., Kapoor, A., Ma, S., Bonatti, R.: Reshaping robot trajectories using natural language commands: a study of multi-modal data alignment using transformers. [arXiv:2203.13411](https://arxiv.org/abs/2203.13411) (2022)
22. Stepputtis, S., Campbell, J., Phielipp, M., Lee, S., Baral, C., Amor, H.B.: Language-conditioned imitation learning for robot manipulation tasks. [arXiv:2010.12083](https://arxiv.org/abs/2010.12083) (2020)
23. Zhu, D., Chen, J., Haydarov, K., Shen, X., Zhang, W., Elhoseiny, M.: ChatGPT asks, blip-2 answers: automatic questioning towards enriched visual descriptions. [arXiv:2303.06594](https://arxiv.org/abs/2303.06594) (2023)
24. Rotstein, N., Bensaid, D., Brody, S., Ganz, R., Kimmel, R.: FuseCap: leveraging large language models for enriched fused image captions. [arXiv:2305.17718](https://arxiv.org/abs/2305.17718) (2023)
25. Levy, M., Ben-Ari, R., Darshan, N., Lischinski, D.: Chatting makes perfect: chat-based image retrieval. [arXiv:2305.20062](https://arxiv.org/abs/2305.20062) (2023)
26. Ricci, R., Bazi, Y., Melgani, F.: Machine-to-machine visual dialoguing with chatGPT for enriched textual image description. *Remote Sens.* **16**(3) (2024)
27. Hu, Y., Hua, H., Yang, Z., Shi, W., Smith, N.A., Luo, J.: PromptCap: prompt-guided task-aware image captioning. [arXiv:2211.09699](https://arxiv.org/abs/2211.09699) (2023)

28. Yu, Z., Ouyang, X., Shao, Z., Wang, M., Yu, J.: Prophet: prompting large language models with complementary answer heuristics for knowledge-based visual question answering. [arXiv:2303.01903](https://arxiv.org/abs/2303.01903) (2023)
29. Ravi, S., Chinchure, A., Sigal, L., Liao, R., Shwartz, V.: VLC-BERT: visual question answering with contextualized commonsense knowledge. [arXiv:2210.13626](https://arxiv.org/abs/2210.13626) (2022)
30. Tiong, A.M.H., Li, J., Li, B., Savarese, S., Hoi, S.C.H.: Plug-and-play VQA: zero-shot VQA by conjoining large pretrained models with zero training. [arXiv:2210.08773](https://arxiv.org/abs/2210.08773) (2023)
31. Li, J., Li, D., Xiong, C., Hoi, S.: BLIP: bootstrapping language-image pre-training for unified vision-language understanding and generation. [arXiv:2201.12086](https://arxiv.org/abs/2201.12086) (2022)
32. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-CAM: visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 618–626 (2017)
33. Brown, T., et al.: Language models are few-shot learners. In: Proceedings of the Advances in Neural Information Processing Systems, vol. 33, pp. 1877–1901 (2020)
34. Zhang, Y., et al.: Siren’s song in the AI ocean: a survey on hallucination in large language models. [arXiv:2309.01219](https://arxiv.org/abs/2309.01219) (2023)
35. Shah, S., Dey, D., Lovett, C., Kapoor, A.: AirSim: high-fidelity visual and physical simulation for autonomous vehicles. [arXiv:1705.05065](https://arxiv.org/abs/1705.05065) (2017)



Dense Road Surface Grip Map Prediction from Multimodal Image Data

Jyri Maanpää^{1,2} , Julius Pesonen¹ , Heikki Hyyti¹ ,
Iaroslav Melekhov² , Juho Kannala^{2,3} , Petri Manninen¹ ,
Antero Kukko¹ , and Juha Hyyppä¹ 

¹ Finnish Geospatial Research Institute FGI, National Land Survey of Finland,
02150 Espoo, Finland

{jyri.maanpaa,julius.pesonen}@nls.fi

² Department of Computer Science, Aalto University, 02150 Espoo, Finland

³ University of Oulu, 90570 Oulu, Finland

Abstract. Slippery road weather conditions are prevalent in many regions and cause a regular risk for traffic. Still, there has been less research on how autonomous vehicles could detect slippery driving conditions on the road to drive safely. In this work, we propose a method to predict a dense grip map from the area in front of the car, based on postprocessed multimodal sensor data. We trained a convolutional neural network to predict pixelwise grip values from fused RGB camera, thermal camera, and LiDAR reflectance images, based on weakly supervised ground truth from an optical road weather sensor.

The experiments show that it is possible to predict dense grip values with good accuracy from the used data modalities as the produced grip map follows both ground truth measurements and local weather conditions, such as snowy areas on the road. The model using only the RGB camera or LiDAR reflectance modality provided good baseline results for grip prediction accuracy while using models fusing the RGB camera, thermal camera, and LiDAR modalities improved the grip predictions significantly.

Keywords: Grip prediction · Autonomous driving · Convolutional neural networks

1 Introduction

Harsh winter conditions pose unique challenges to autonomous driving. According to the Road Weather Management Program by the U.S. Department of

J. Maanpää and J. Pesonen—Shared an equal contribution to this work as first authors.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-78447-7_26.

© The Author(s) 2025

A. Antonacopoulos et al. (Eds.): ICPR 2024, LNCS 15317, pp. 387–404, 2025.

https://doi.org/10.1007/978-3-031-78447-7_26

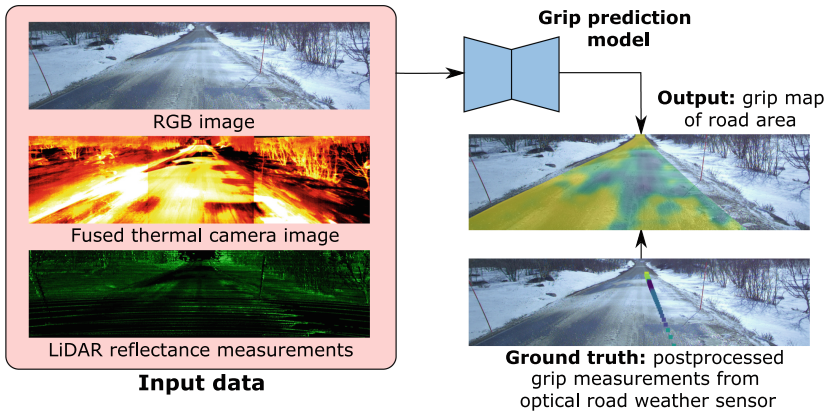


Fig. 1. Our work presents a grip prediction model, which operates on pixelwise fused RGB camera, thermal camera, and LiDAR reflectance measurements and predicts a dense grip map of the road area. The ground truth for training is obtained with an optical road weather sensor that provides road grip measurements which are postprocessed with GNSS trajectories and external calibrations to match the input data.

Transportation, 24 % of weather-related vehicle crashes in the U.S. occur on snowy, slushy, or icy pavement, and 15 % happen during snowfall or sleet each year [20]. Besides low visibility, significant challenges posed by winter conditions include changes in road surface slipperiness. Snowy and icy road surfaces, in particular, can drastically reduce the friction between vehicle wheels and the road compared to dry and wet roads. Therefore, autonomous driving systems must be capable of distinguishing such scenarios, requiring specialized sensing solutions.

Several approaches exist for estimating the grip on the road. However, the greatest shortcoming of most of these methods in the sense of autonomous driving has been the lack of ability to sense the road ahead of the vehicle, thus only allowing the vehicle to react in situations where the slippery conditions have already affected the driving. To enable sensing of the road further ahead, cameras or other forward-facing, longer-range sensors must be deployed.

In addition, the grip can often vary between different sections of the road depending on local snow, ice, and water layer thicknesses. For example, snowy roads with frequent traffic usually have clear tire tracks, which human drivers follow to avoid the snowy areas on the road. Human drivers can also distinguish sudden icy or wet areas on the road allowing them to either avoid these areas or decrease driving speeds accordingly. Therefore producing dense grip predictions would enable autonomous vehicles to react to the sensed conditions in a manner that human drivers can achieve.

Besides traditional vision by RGB cameras, even more accurate grip predictions could be achieved by combining measurements from other long-range sensors. For example, many LiDARs measure the return intensity of the sent infrared laser pulse, which could be used to differentiate ice and water on the

road due to their different optical properties. In addition, a thermal camera might be able to differentiate some road surface layer types, such as snowy and clear areas on the road. These sensors are commonly used in autonomous vehicles, so their positive effect on grip prediction could be easily adopted by the industry.

In this paper, we introduce a pixelwise road surface grip prediction model based on a convolutional neural network (CNN) to generate dense grip map predictions based on fused images from front-facing RGB and thermal cameras as well as LiDAR reflectance measurements. The ground truth grip values were provided by an optical road weather sensor, the data of which were projected on the other sensor data during postprocessing using 3D transformations between different sensors and the postprocessed trajectory of the data collection. The main idea and different sensor modalities are presented in Fig. 1. We collected a 37 h (1538 km) dataset with our autonomous research platform ARVO (Fig. 2) within different adverse weather conditions and preprocessed it for the aims of this study.

This work extends the previous work by Pesonen [19] and provides a new ablation study of the grip prediction accuracy between different input data modalities. The previous approach was also improved with a more consistent training and validation setup and extended testing. The capability of the dense grip map prediction was measured using quantitative error measurements and qualitative analysis for road areas where ground truth measurements could not be obtained. The study shows that the dense grip predictions are improved with the fused RGB, thermal camera, and LiDAR inputs, while the model relying on the sole RGB inputs, already, greatly improves the resolution of any prior camera-based grip prediction methods.

Our contributions to the state of the art are: 1) We developed a novel method to collect and process a dataset with pixelwise matching of multimodal images and sparse road grip measurements. 2) We proposed a model to predict a dense grip map of the road area in diverse weather conditions. 3) We compared the grip prediction accuracies of models using RGB images, thermal images, and LiDAR reflectance measurements as model input modalities both separately and with every combination using multi-encoder-fusion.

We shared a demo of our models in a Gitlab repository to allow readers to test our methods.¹

2 Background

While dense road surface grip map prediction has only been proposed in our earlier work [19], methods for grip prediction have been proposed before using various sensor setups. Road surface grip measurement methods can be roughly divided into non-contact and contact-based measurements, which have been addressed in surveys by Ma et al. [12] and Acosta et al. [1] respectively. Even

¹ <https://gitlab.com/fgi.nls/public/grip-prediction>.

though contact-based grip sensors and models relying on vehicle information, such as wheel rotation speeds, are incapable of producing the required predictions for grip in front of the car, their use has been essential for evaluating the later-developed non-contact methods. According to the survey by Ma et al. [12] the most prominent non-contact-based methods rely on infrared spectroscopy, computer vision, optical polarisation, or radar detection.

Most of the proposed camera-based road surface grip prediction methods have relied on classification of different road surface conditions without providing scalar estimates of the surface grip [18, 22] or by using a two-part process where the classification result is further used to generate a scalar estimate of the road surface grip [4, 10, 24, 28]. Few models have also been suggested for directly generating scalar grip estimates [2, 5, 16]. However, as a common limitation, all of the models rely on either generating a single prediction for the whole input image or for small regions of interest in predefined shapes. In some of the studies the ground truth labeling was generated by expert annotators [22, 28], in one using a portable pendulum tester [4], in one using friction wheel trailer measurements [10], in one with vehicle response [2], and in one with an optical sensor [16].

Models generating pixelwise outputs have become popular in many tasks, such as semantic segmentation and monocular depth estimation. In semantic segmentation, models are trained to classify each pixel of the input image. Solutions proposed for the task, some of which have also found use in many other problems, include U-net [21], FPN [11] and DeepLabV3+ [3].

Monocular depth estimation is a task more similar to the one presented in this paper, as the labels are scalar distance values instead of discrete classes as in the case of segmentation. In addition, the training labels could originate from sparse measurements such as LiDAR readings. Such weak supervision has been applied to monocular depth estimation with sparse labels by Guizilini et al. [6] showing similarity to our grip prediction task due to the comparable sparsity of the ground truth labels.

As both optical road weather sensors and LiDARs use lasers to measure the return intensity of the measured object, the use of LiDARs for road surface condition prediction shows potential. Ruiz-Llata et al. [23] and Shin et al. [26] showed that different road surface conditions can be detected using LiDAR measurements. Sebastian et al. [25] proposed the use of a LiDAR-based CNN for simultaneous road condition and weather classification. While the use of 3D LiDARs was proposed in the studies, their use for dense road surface grip prediction was not investigated in depth.

As noted, the prior literature is concentrated on low-resolution grip predictions using individual sensor inputs. This study aims to fill the gap by introducing both data and methods for generating dense predictions using multimodal input data.

3 Data

In this section, we describe the collection and preprocessing of the data used to train and evaluate the proposed grip prediction methods.

3.1 Dataset Collection

We collected a 37 h or 1538 km dataset of different driving conditions with the sensor setup in our autonomous driving research vehicle ARVO, of which an older version is presented in our earlier study [13]. The dataset includes various driving conditions, such as daytime and nighttime, snowfall, snow-covered roads, slushy conditions, rain, and wet roads. It also contains data from several road types, such as highways, urban roads, and paved and unpaved local roads. The dataset was collected mostly in the capital region of Finland during fall and winter, with a smaller part collected in Western Lapland during spring. The dataset was postprocessed to contain samples at a frequency of 2 fps and after automatic filtering of low-quality data, the dataset had 237 067 samples.

For this project, we used data from a forward-facing RGB camera (Basler MED ace 2.3 MP 164 color), three forward-facing thermal cameras (FLIR ADK, 24° FOV), a roof-mounted 128-beam rotating LiDAR (Velodyne Alpha Prime VLS-128), a GNSS Inertial Navigation System (INS) (Novatel PwrPak7-E1), and a mobile road weather sensor (Vaisala Mobile Detector MD30). An image of the car with highlighted sensor locations is shown in Fig. 2. The left and right thermal cameras were horizontally tilted approximately 23° outwards from the center camera to achieve a combined field of view covering a larger horizontal angle. All sensors were synchronized with GNSS INS triggering signals except the road weather sensor, which was synchronized manually during postprocessing.

We chose these long-range sensor modalities for this research due to their common use in autonomous driving development. We also noted in our preliminary studies that different types of snow can have type-specific features in thermal cameras. LiDAR reflectance measurements (laser pulse return intensity amplitude which is normalized with distance internally by LiDAR sensor) could also provide single-band spectral information on the surface material as the LiDAR sensor used in this study uses 903 nm wavelength lasers, which has



Fig. 2. The research vehicle ARVO used for data collection. The long-range sensors shown in box A are 1. LiDAR, 2. RGB camera in a weatherproof housing and 3. thermal cameras. The road weather sensor is shown in box B.

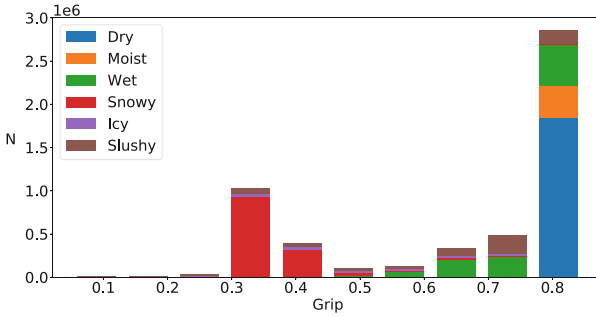


Fig. 3. The distribution of grip values and road surface states provided by the road weather sensor measurements in the complete unprocessed dataset. We observe that most of the data is collected within dry, wet, or snowy conditions.

different extinction coefficients between water and ice according to the data by Palmer and Williams [17] and the data by Warren and Brandt [27].

The road weather sensor Vaisala Mobile Detector MD30 is an optical sensor that estimates water, ice, and snow layer thicknesses using three laser intensity measurements in different wavelengths. The operating principle of the sensor is not publicly available, but an earlier sensor prototype is presented in a master’s thesis [9]. The sensor uses an internal model for calculating the grip estimate of the road, most likely based on the three surface layer thickness values. Based on earlier studies on optical sensors [14, 15], we assume that an optimistic upper limit of the sensor grip estimate accuracy is 0.1. However, the grip estimate can be more accurate within clear conditions with constant grip. The surface layer thickness and grip estimates are measured with a 40 fps sampling rate. In addition, the sensor provides road surface condition class, road and air temperatures, and other meteorological measurements. In our analysis, we have assumed that the sensor’s grip measurements are sufficiently accurate that it is reasonable to imitate the sensor measurements for a dense grip map, even though the sensor grip values are likely to contain some inaccuracies as the grip between the tires and the road surface is a complex physical phenomenon.

The grip and road surface condition distribution in our dataset is visualized in Fig. 3. We observe that the two most prominent road states are dry and snowy conditions. These conditions have two distinct grip coefficients, which are 0.82 for dry road and 0.35 for snowy road. Additionally, we note that wet roads usually have a grip of less than 0.8, and the smallest grip of 0.1 is observed on icy roads or roads with very thick layers of water.

3.2 Datasplit

The dataset includes data collections from 18 days, many of which shared the same data collection locations. To ensure that the training, validation, and test sets were collected from different locations while maintaining similar weather

condition distributions, we used a geofencing-based approach to choose the validation and test sets from the full dataset. We chose circular areas within approximately one-kilometer intervals from the data collection area so that all samples collected within these areas are included either in the validation or the test set. The rest of the data is included in the training set, except for any positions less than 55 m from any border of the chosen circular areas. This 55-m gap assures that no observations are shared between validation, testing, or training. With this data split, we achieved a qualitatively similar distribution of weather conditions between training, validation, and test sets. Additional qualitative data filtering was also done at this stage. In the end, the training set has 159 801 samples (79.1%), the validation set has 15 343 samples (7.6%) and the test set has 26 783 samples (13.3%).

There is a possibility that some conditions of the input data, such as illumination, would allow the model to fit to these conditions and learn the general grip conditions on specific data collection dates. Therefore, we used three separate data collections, with 16 139 samples in total, as additional test drives to demonstrate the accuracy of the model regardless of this effect.

3.3 Pixelwise Matching of Modalities

To obtain pixelwise pairs of image data and ground truth road weather measurements, we used the following preprocessing approach. We calibrated all cameras intrinsically and extrinsically and measured the 3D locations and orientations of each sensor. Due to the hardware-based synchronization, we also know the time correspondences between each of the sensors. The GNSS trajectory was postprocessed using base-station data to increase its accuracy.

We chose the RGB camera image as the reference frame of the data as it has the highest resolution regarding the front area of the car. The road weather sensor measurements were overlaid on the RGB images with the following procedure: first, we used the postprocessed trajectory and the external transformation between the INS reference frame and the road weather sensor measurement locations to project the road weather sensor measurement positions to a 3D trajectory. This trajectory of the measurements is then transformed to the RGB camera coordinates and projected to the RGB camera image plane. Therefore, we obtained RGB camera images where the road weather measurement points, which were recorded soon after the RGB camera capture time, are overlaid. To improve the data quality, we only included road weather measurement points within 50 m of the cameras and excluded the measurement points behind any obstacles.

The LiDAR point clouds were motion-corrected with the postprocessed trajectory and projected to the RGB camera pixel coordinates. We also accumulated more LiDAR points from the lower part of the three previous scans to include more reflectance measurements from the nearby road area.

The thermal cameras required a more complex pixelwise matching with the RGB camera. Initially, we generated approximate range images from the LiDAR point clouds projected onto the RGB camera. For each RGB camera pixel, we

identified a corresponding 3D point from the range image, projected that point onto a single thermal camera frame, and determined the corresponding thermal value from the thermal camera image. As a result, we obtained thermal camera images projected onto each RGB camera frame. We normalized the pixel values of the left and right thermal cameras to match the scale of the center thermal camera, ensuring a value distribution close to the shared image border.

As the thermal pixel values correspond to the thermal flux in the pixel with a varying scale due to online calibration, the raw thermal values were not considered suitable for this work. Therefore, we normalized the thermal camera pixel values within each frame so that a sample area from the road has a consistent distribution with zero mean and unit variance. In addition, the borders of the thermal camera images had lower values within cold conditions due to the operation of the thermal camera sensor. We alleviated this effect by determining the systematic error distribution for each thermal camera image and subtracting that error to obtain an image with a more homogenous value distribution. The data preprocessing is described in more detail in the preliminary results of our work [19].

An example of the pixelwise matched sensor data can be seen in Fig. 1. The pixelwise matching quality varies between frames and occasionally distant areas or tall objects closer to the camera might appear unaligned between different sensors. We considered this effect negligible for this work, as the road surface is mostly well aligned and the road surface is usually large and homogenous, alleviating any problems that could be caused by the slight unalignment.

4 Methods

In this section, we present our model for the grip prediction, the training setup, and the performance evaluation methods.

4.1 Model

To generate dense predictions of the road surface grip using the multimodal input data, we propose using a convolutional neural network trained with the sparse pixelwise matched road weather measurements as the ground truth labels. Our models are based on Feature Pyramid Network (FPN) [11] which is adapted to predict pixelwise scalar values for regression. The FPN model was chosen as it was shown efficient for the task in our preliminary studies.

We trained our models with every combination of the collected input modalities to measure their effect on grip prediction accuracy. The models utilizing a single input modality are based on the standard FPN implementation which takes an image tensor as the input. However, the multimodal models include separate encoders for each input modality, and their features are concatenated channel-wise within each feature scale before being forwarded to the decoder. We implemented this feature-level fusion approach due to finding occasional lower-quality samples in some input modalities, meaning it was useful for the model to

learn to discard these features in the corresponding situations. For each of the model encoders, we used ResNet-18 [7].

The outputs of the model are the predicted grip and the predicted water, ice, and snow layer thicknesses for each pixel. The grip prediction is the primary task of the model and the prediction of different surface layer thicknesses is used as an auxiliary task to support the learning, as it has been shown to improve the prediction accuracy of the obtained model in our prior experiments [19]. The model architecture and the training scheme are illustrated in Fig. 4.

In most frames, more road weather measurement points were visible further away from the car. These distant points also contain less information as the resolution of the RGB camera and other sensors concerning the road surface was smaller. We alleviated the effect of these distant points by weighting the road weather measurements within each image based on their y -coordinate in the RGB image plane: the weight of each measurement point decreases linearly from the bottom of the image to the estimated horizon level. With this approach, we could approximately balance the prediction accuracy over the whole road area. For validation and testing the pixelwise weights were normalized within each frame so that their mean is one. For the training, the normalization was performed on the unfiltered road weather measurements, which included some overlapping positions, leading to slightly larger weights on average.

The predicted grip and surface layer thickness values are compared to the sparse ground truth values from the postprocessed road weather sensor data

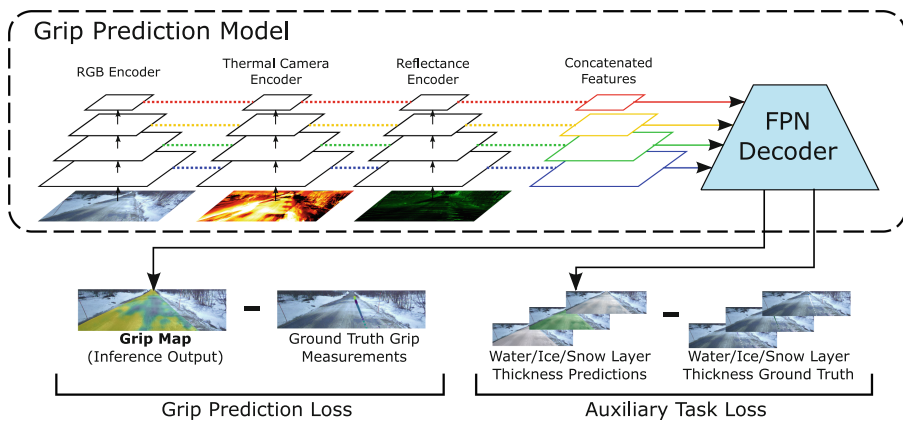


Fig. 4. The model architecture and training scheme for the model using all data modalities. Each input data modality has a separate encoder and their features are concatenated within each feature scale before the FPN decoder. The loss is evaluated both for the grip and the surface layer thickness prediction tasks.

with the following loss function:

$$\mathcal{L}(x, y_p, y_a, w|\theta) = \frac{1}{N} \sum_{i=1}^N w^i (y_p^i - f_p^i(x|\theta))^2 + \frac{\lambda}{3N} \sum_{l=1}^3 \sum_{i=1}^N w^i (y_a^{l,i} - f_a^{l,i}(x|\theta))^2, \quad (1)$$

where x is the input image tensor, N is the number of pixels containing ground truth grip values in the sample, w^i is the weight for pixel i , y_p^i , and $f_p^i(x|\theta)$ are the ground truth and model output for grip value at pixel i and $y_a^{l,i}$ and $f_a^{l,i}(x|\theta)$ are the ground truth and model output for surface layer l value at pixel i . The first term denotes the weighted mean square error for the grip prediction task and the second term denotes the similarly weighted mean square error for the prediction of surface layer thicknesses. The parameter λ is used to adjust the effect of the supportive auxiliary task and in our experiments, it was set to 1.0.

4.2 Training Setup

Each model was trained using the Adam optimizer [8] for 38 epochs with a batch size of 32 and a learning rate of $1e-3$. The FPN model used a dropout rate of 20% in its last layer. The models were compared using the instances which achieved the best validation loss during the training.

As our method was designed to predict the grip and the layer thicknesses based on the surface appearance, we avoided excessive augmentation to maintain accurate predictions. Some augmentations were still used to ensure appropriate generalization. Therefore, it was chosen to apply small random scale and rotation augmentation with a 30% probability, horizontal flip with a 50% probability, small random blur with a 30% probability, and random color jitter to the RGB images with a 30% probability.

4.3 Performance Evaluation

The model performance was evaluated with a root mean square error (RMSE) between the predicted and the ground truth grip values. A similar weighting as in the training loss (1) was applied during validation and test error evaluation as we wanted to measure the grip prediction accuracy balanced over the road area. Due to this weighting, the mean square error was evaluated for each frame separately, and these sample-wise square errors were averaged before evaluating the square root. The test accuracy is reported both for the test set from the main data collection and the three extra test drives with no correspondence to the main dataset.

In addition to the main evaluation metric RMSE, we assessed the test set performance using mean absolute error (MAE), pixelwise unweighted RMSE, and grip weighted RMSE. The pixelwise unweighted RMSE is similar to the main RMSE error, except it omits the weighting in the equation (1). To calculate the grip weighted RMSE, the grip range is divided into ten bins. The weight for each

bin is calculated from the inverse of the number of ground truth measurements in that bin, with all weights scaled to a mean of one. The framewise evaluated mean squared error within each bin is then weighted with the corresponding bin weight, and the square root of the mean of these weighted errors is calculated to obtain the grip weighted RMSE. This process imitates the error for a dataset with a uniform distribution of grip measurements. The grip weighted RMSE is also weighted in a pixelwise manner, similar to the training loss.

However, the error evaluation alone could not show if the grip predictions are valid over the road areas which rarely contain ground truth measurements. Therefore we also performed qualitative analysis on the model output to estimate how well the grip map follows the slipperiness expected by human drivers.

5 Results

In this section, we first analyze the quantitative errors from our validation and test sets and then inspect the qualitative performance of different models.

5.1 Validation and Test Set Errors

We performed the experiments by training the model with different sensor modalities as the input to observe the effect of each sensor on the grip prediction accuracy. The validation set, test set, and separate test drive dataset information and RMSEs achieved with each model are found in Table 1. The additional metrics for the test set are listed in Table 2.

For the validation and test set all obtained errors are significantly smaller than the standard deviation of the dataset, which insists that the models could learn to predict useful grip values. In most experiments, the best or second-to-best results are achieved with the model that uses all data modalities. Using RGB images provides the best accuracy when compared to other data modalities, but the model using only the LiDAR reflectance achieves comparable results with the RGB model. While the combination of RGB and thermal images does not improve performance over using RGB data alone, combining thermal and reflectance data provides similar improvements as the combination of RGB and reflectance. This indicates that the RGB and thermal information may overlap significantly, but also provide information unavailable from LiDAR reflectance alone. Therefore, almost all of the best or second-best results in Table 1 are achieved using some combination of LiDAR reflectance and a higher-resolution image input.

The separate test drive results confirm that the use of several data modalities improves the accuracy and the models have not noticeably overfit to the training data. Even though the standard deviation in each test drive is close to the model errors, it should be noted that the conditions in a single test drive are mostly constant and the models have predicted at least the general conditions in the test drive. However, there is variance and inconsistency in the separate test drive results as the amount of data is small, and adverse effects in one modality could

decrease the performance of a single model. Some differences between the results could also be explained by the specific driving conditions, as the dark conditions in Test drive 1 might benefit the performance of the models using reflectance.

The results of the additional test set metrics in the Table 2 follow the expected previous results with RMSE. As the grip weighted RMSE has larger values than the previous RMSE, we conclude that less prevalent grip values are more difficult to predict than the prevalent values. However, the grip weighted RMSE is below 0.1 which is relatively accurate as grip prediction performance.

We also verify the FPN model choice by comparing the performance of U-net and DeepLabV3+ using only RGB inputs. The results are shown in Table 3. While DeepLabV3+ performed best on the test set and two of the separate test drives, the comparable results confirm that our validation loss-based model choice should not hinder our results and further model optimization can be left as future work along other hyperparameters.

In addition, a scatter plot of the grip and different surface layer thickness predictions in the test set is shown in Fig. 5. The surface layer thickness predictions mostly follow the ground truth values within a relatively small error range while the predicted grip values have a larger error distribution, partly due to misinterpretation of snowy conditions.

5.2 Qualitative Performance

Besides the error evaluation based on the ground truth road weather sensor measurements, we evaluated the grip map prediction over the complete road area qualitatively. Several example scenarios and grip map predictions from the final proposed model in different road weather conditions are shown in Fig. 6. Additionally, examples from the other introduced models and a comparison of

Table 1. Dataset information and grip prediction RMSE for different models on the validation set, test set, and separate test drives. Different data modalities are abbreviated where RGB denotes RGB camera, T denotes thermal camera and R denotes LiDAR reflectance measurements. The best-achieved error in each set is in bold and the second-to-best is underlined.

	Validation set	Test set	Test drive 1	Test drive 2	Test drive 3
Weather condition	Varying	Varying	Snowy,snowfall, dark	Snowy	Wet, slushy
Grip mean	0.6474	0.659	0.399	0.557	0.649
Grip SD	0.2037	0.201	0.104	0.140	0.159
# samples	15 343	26 783	5 746	2 042	8 351
Modalities	RMSE	RMSE	RMSE	RMSE	RMSE
RGB	0.0657	0.0589	0.1041	0.1497	0.1062
T	0.0794	0.0772	0.1248	0.1670	0.1361
R	0.0677	0.0591	<u>0.0992</u>	0.1262	0.0944
RGB + T	0.0655	0.0605	0.1024	0.1416	0.1069
RGB + R	<u>0.0638</u>	0.0565	0.1038	0.1418	<u>0.0917</u>
T + R	0.0664	0.0586	0.1056	0.1038	0.0906
RGB + T + R	0.0632	<u>0.0575</u>	0.0974	<u>0.1118</u>	0.0994

Table 2. Additional test set metrics of the models using different modalities.

Modalities	MAE	Pixelwise unweighted RMSE	Grip weighted RMSE
RGB	0.0242	0.0698	0.0912
T	0.0323	0.0875	0.0955
R	<u>0.0236</u>	0.0734	0.0947
RGB+T	0.0248	0.0720	<u>0.0909</u>
RGB+R	0.0248	<u>0.0680</u>	0.0939
T+R	0.0231	0.0720	0.0974
RGB+T+R	0.0238	0.0674	0.0908

Table 3. Comparison of different model architectures with only RGB input. Errors in RMSE.

Model	Validation set	Test set	Test drive 1	Test drive 2	Test drive 3
FPN (proposed)	0.0657	<u>0.0589</u>	0.1041	<u>0.1497</u>	<u>0.1062</u>
U-net	0.0692	0.0604	0.1065	0.1466	0.1127
DeepLabV3+	<u>0.0673</u>	0.0583	<u>0.1061</u>	0.1462	0.1026

the impact of different modalities on the qualitative results are shown in Fig. 7 and in the supplementary material. In all figures in this work, the road area is segmented manually as the model does not differentiate the road area from the input data.

In general, we observe that the model output is smooth and is often constant when there are no variations in road weather conditions, such as when the road is completely dry, completely wet, or completely covered in snow. The model predictions could also mostly follow the boundaries between snowy and clear areas as seen in scenarios presented in Fig. 6 where clear tire tracks can be seen on otherwise snowy roads. Some conditions are still difficult to detect, such as the second scenario on the right column, in which the model could not detect the low grip of an area covered with deep water.

In addition, the model performance is unclear in some conditions that are further from the usual ground truth data locations, such as on the adjacent lane. The model output also could not follow sharp changes in grip values as the model seems to average the grip on relatively large prediction areas. This is likely due to the sparsity and varying data quality of the ground truth road weather measurements.

In Fig. 7 we show performance differences between models using different data modalities as inputs. In some examples, the thermal and reflectance-based single modality models misclassify the grip conditions of the whole scene as the data modality can not differentiate the current condition correctly. However, the model using each data modality seems to combine the correct predictions from the single modalities into a consistent representation of the grip map.

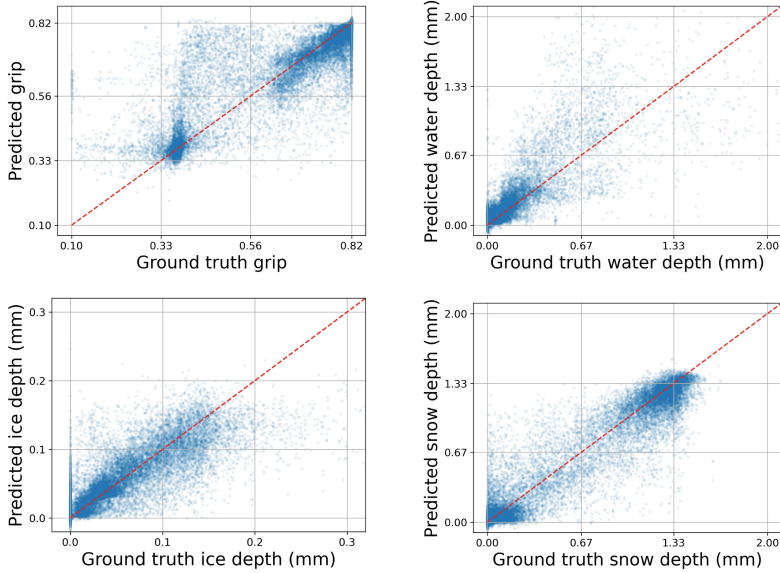


Fig. 5. Scatter plots of predicted grips and layer thicknesses produced by the best, proposed model (RGB+T+R). The x-axis represents the ground truth values and the y-axis the predictions. The plots were generated using 50 000 random measurements and corresponding predictions from the test set. The red dashed line represents the position of correct predictions. (Color figure online)

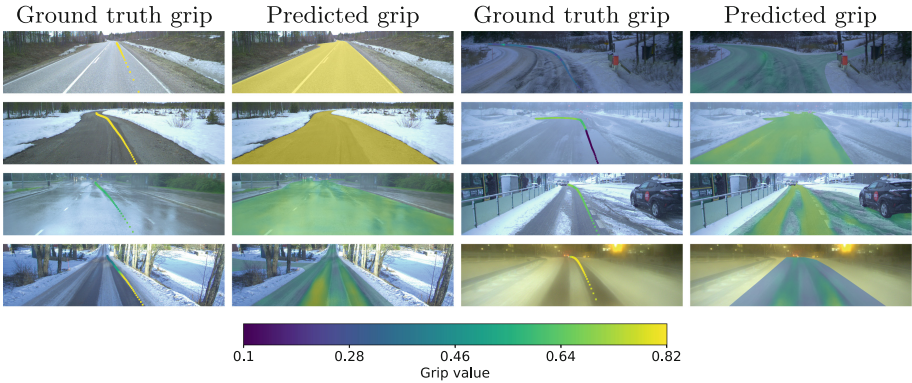


Fig. 6. Visualisations of the qualitative performance of the final model (RGB+T+R). The ground truth labels are shown using 14-by-14-pixel colored squares drawn on the RGB input image.

6 Discussion

The results support the original hypothesis on the accuracy of the dense grip map and the benefit of additional data modalities besides the RGB camera. Even

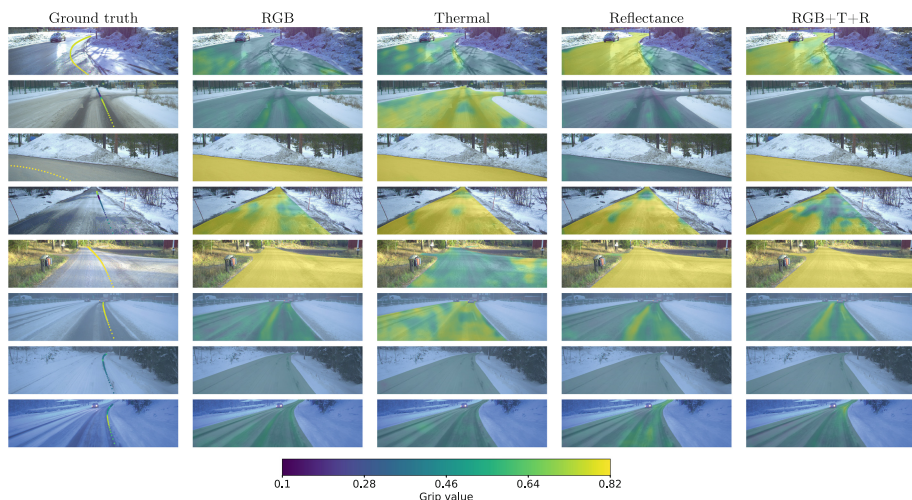


Fig. 7. Example grip prediction visualizations for each single-modality model and the RGB+T+R model. Some samples were chosen to highlight the performance differences between models.

though a large portion of the accuracy is obtained by separating the dry and snowy conditions from each other, it seems the model can perform in other road weather conditions as well.

However, one has to consider several error sources, as the optical road weather sensor is not designed to measure all the complex phenomena that could affect the grip between the tires and the road. In addition, the correct synchronization and alignment of the road weather sensor data was challenging. There is also a risk that the data split into training, validation, and test sets could cause some samples in different sets to have too many similarities meaning it's possible that some overfitting could not be observed from the validation and test set results. However, the results on the data from the three extra test drives defend the validation and test set results. In general, one would need an even larger representation of different weather conditions in the dataset to obtain a model with less bias and higher accuracy in several real-life road weather conditions. Despite these limitations, our results show evidence of the performance of our method.

7 Conclusions

This study presents a novel method to predict a dense grip map of the road area from multimodal image data with a convolutional neural network. The models using RGB or 3D LiDAR reflectance measurements provide the best baseline results, whereas the highest accuracy predictions are achieved with sensor fusion using modality-wise encoders. The use of thermal camera images also

shows potential, while their contribution is smaller than that of the RGB and reflectance measurements. The results follow those of earlier studies in proving that the RGB camera is a powerful tool for detecting road surface conditions while also providing major steps in using 3D LiDAR reflectance measurements for dense grip prediction both alone and alongside RGB cameras.

The best model configuration using a combination of all three input modalities achieves an RMSE of 0.0632 and an RMSE of 0.0575 on the diverse validation and test sets respectively. The results from separate test drives also prove the system's usability in unseen conditions. In addition, the qualitative results show the model recognizing various shapes of snow, ice, and water layer distributions affecting the grip prediction. These qualitative results were also improved with the model that uses multimodal inputs with the fusion of encoder features.

To achieve a reliable implementation of this method for autonomous driving, one should collect a large dataset with improved sensor data quality and an even more diverse and balanced set of road and weather conditions. It could also be investigated if one could improve the prediction accuracy by switching the reference image plane from the presented RGB camera frame to another plane, such as the bird's-eye view of the road area or even the 3D frame of the LiDAR. Finally, one should develop methods to predict the uncertainty of the grip prediction output to fuse the output from this method reliably with autonomous driving systems.

Acknowledgements. Funded by the European Union (grant no. 101069576). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Climate, Infrastructure and Environment Executive Agency (CINEA). Neither the European Union nor the granting authority can be held responsible for them.

The contribution of the authors is as follows: Maanpää and Pesonen planned the study, performed experiments, and wrote the manuscript. Maanpää also preprocessed the dataset and Pesonen performed preliminary experiments before this study. Manninen, Maanpää and Hyyti developed the research vehicle for data collection with the research group and Maanpää collected the dataset. Hyyti, Melekhov, and Kannala advised in the planning of the study. Kukko and Hyyppä supervised the project.

In addition, we would like to thank Eugeniu Vezeteu for his help in data collection and sensor calibration and Paula Litkey for participating in the vehicle development. We would also like to thank Eero Ahokas for GNSS trajectory processing and Josef Taher for their advice during this work.

References

1. Acosta, M., Kanarachos, S., Blundell, M.: Road friction virtual sensing: a review of estimation techniques with emphasis on low excitation approaches. *Appl. Sci.* **7**(12), 1230 (2017)
2. Cech, J., Hanis, T., Kononisky, A., Rurtle, T., Svancar, J., Twardzik, T.: Self-supervised learning of camera-based drivable surface roughness. In: 2021 IEEE Intelligent Vehicles Symposium (IV), pp. 1319–1325. IEEE (2021)

3. Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 833–851. Springer (2018)
4. Du, Y., Liu, C., Song, Y., Li, Y., Shen, Y.: Rapid estimation of road friction for anti-skid autonomous driving. *IEEE Trans. Intell. Transp. Syst.* **21**(6), 2461–2470 (2019)
5. Du, Z., Skar, A., Pettinari, M., Zhu, X.: Pavement friction evaluation based on vehicle dynamics and vision data using a multi-feature fusion network. *Transp. Res. Rec. J. Transp. Res. Board* **2677**(11), 219–236 (2023)
6. Guizilini, V., Li, J., Ambrus, R., Pillai, S., Gaidon, A.: Robust semi-supervised monocular depth estimation with reprojected distances. In: Conference on Robot Learning, vol. 100, pp. 503–512. PMLR (2020)
7. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778. IEEE (2016)
8. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. In: 3rd International Conference on Learning Representations, (ICLR) (2015)
9. Kivi, J.: Algorithmic road state modelling. Master’s thesis, Aalto University School of Science (2019)
10. Langstrand, J.P., Randem, H.O., Thunem, H., Hoffmann, M.: Using deep learning to classify road surface conditions and to estimate the coefficient of friction. In: 2023 IEEE Intelligent Vehicles Symposium (IV), pp. 1–8. IEEE (2023)
11. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2117–2125. IEEE (2017)
12. Ma, Y., Wang, M., Feng, Q., He, Z., Tian, M.: Current non-contact road surface condition detection schemes and technical challenges. *Sensors* **22**(24), 9583 (2022)
13. Maanpää, J., Taher, J., Manninen, P., Pakola, L., Melekhov, I., Hyypä, J.: Multimodal end-to-end learning for autonomous steering in adverse road and weather conditions. In: 2020 25th International Conference on Pattern Recognition (ICPR), pp. 699–706. IEEE (2021)
14. Malmivuo, M.: Comparison study of mobile optical friction and temperature meters 2013. Publications by Finnish Transport Agency (2013)
15. Malmivuo, M.: Test of optical MD30 sensors within Mäntsälä contract – winter season 2021–22 (in Finnish). Publications by Finnish Transport Infrastructure Agency (2023)
16. Ojala, R., Seppänen, A.: Enhanced winter road surface condition monitoring with computer vision. arXiv preprint [arXiv:2310.00923](https://arxiv.org/abs/2310.00923) (2023)
17. Palmer, K.F., Williams, D.: Optical properties of water in the near infrared. *J. Opt. Soc. Am.* **64**(8), 1107–1110 (1974)
18. Panhuber, C., Liu, B., Scheickl, O., Wies, R., Isert, C.: Recognition of road surface condition through an on-vehicle camera using multiple classifiers. In: Proceedings of SAE-China Congress 2015: Selected Papers, pp. 267–279. Springer (2016)
19. Pesonen, J.: Pixelwise Road Surface Slipperiness Estimation for Autonomous Driving with Weakly Supervised Learning. Master’s thesis, Aalto University School of Science (2023)
20. Road Weather Management Program of U.S. Department of Transportation: Snow and ice (2023). https://ops.fhwa.dot.gov/weather/weather_events/snow_ice.htm. Accessed 8 Feb 2024

21. Ronneberger, O., Fischer, P., Brox, T.: U-net: convolutional networks for biomedical image segmentation. In: *Medical Image Computing and Computer-Assisted Intervention*, pp. 234–241. Springer (2015)
22. Roychowdhury, S., Zhao, M., Wallin, A., Ohlsson, N., Jonasson, M.: Machine learning models for road surface and friction estimation using front-camera images. In: *2018 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8. IEEE (2018)
23. Ruiz-Llata, M., Rodríguez-Cortina, M., Martín-Mateos, P., Bonilla-Manrique, O.E., López-Fernández, J.R.: Lidar design for road condition measurement ahead of a moving vehicle. In: *2017 IEEE Sensors*, pp. 1–3 (2017)
24. Šabanovič, E., Žuraulis, V., Prentkovskis, O., Skrickij, V.: Identification of road-surface type using deep neural networks for friction coefficient estimation. *Sensors* **20**(3), 612 (2020)
25. Sebastian, G., Vattem, T., Lukic, L., Bürgy, C., Schumann, T.: Rangeweathernet for lidar-only weather and road condition classification. In: *2021 IEEE Intelligent Vehicles Symposium (IV)*, pp. 777–784. IEEE (2021)
26. Shin, J., Park, H., Kim, T., et al.: Characteristics of laser backscattering intensity to detect frozen and wet surfaces on roads. *J. Sens.* 8973248 (2019)
27. Warren, S.G., Brandt, R.E.: Optical constants of ice from the ultraviolet to the microwave: a revised compilation. *J. Geophys. Res. Atmos.* **113**(D14) (2008)
28. Zhao, T., Guo, P., Wei, Y.: Road friction estimation based on vision for safe autonomous driving. *Mech. Syst. Signal Process.* **208**, 111019 (2024)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.





Video-Based Semi-automatic Drivable Area Segmentation

Zhengyun Cheng[✉], Guanwen Zhang^(✉), Changhao Wang, and Wei Zhou

Northwestern Polytechnical University, Xi'an, China
guanwen.zh@nwpu.edu.cn

Abstract. Drivable area segmentation is a crucial task in autonomous driving. While current works mainly focus on analyzing single image and overlook the intra-video associations, due to the limited availability of video-based datasets. We present a novel prototype-based approach, named DASeg, to tackle the challenge of annotating intra-video query images using annotated support images as guidance. The primary obstacle lies in aggregating representative prototypes while ensuring resilience to variations in appearance and position across the video. Our method consists of three key components: position embedding for utilizing positional priors, soft-pooling for alleviating the limited coverage of intra-class variations from the support provided, and prototype regularization for generalizability enhancement. We augmented the lane detection dataset VIL-100 by incorporating drivable area annotations, resulting in a new dataset named VDA-100, which was employed to evaluate the performance of the proposed method. Experiments show that our method achieves mIoU score of 88.3% with the pre-trained backbone from lane detection model, and 89.1% when trained from scratch. Our code and dataset is available at <https://github.com/CZY-Code/DASeg>.

Keywords: Drivable area segmentation · Positional prior · Soft-pooling · Prototype regularization

1 Introduction

Drivable area division refers to the identification of flat and obstacle-free regions on the road where vehicles can maneuver while adhering to traffic regulations. The availability of drivable area information is essential for local path planning at all levels of autonomous driving and plays a crucial role in decision-making tasks, including steering and lane changes. The performance of drivable area segmentation technology has a profound impact on the safety of intelligent vehicles.

Vision sensors remain crucial for acquiring information in autonomous driving. However, the imaging capabilities of vision sensors are susceptible to illumination and weather conditions, making it challenging to maintain robustness in complex and dynamic scenarios. Traditional road segmentation approaches like threshold methods [23] and clustering methods [31] rely on hand-designed

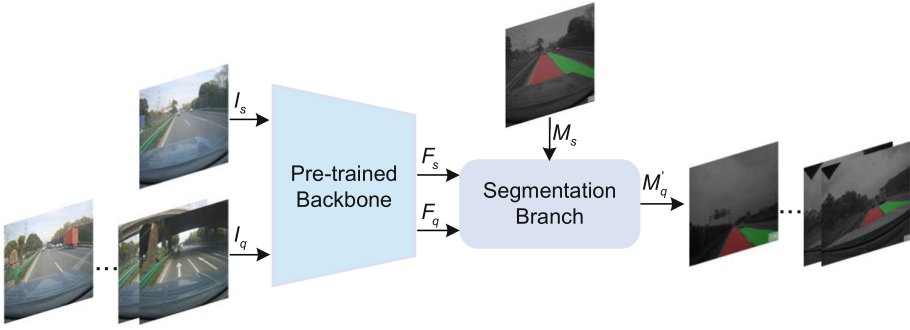


Fig. 1. Problem formulation. Given support image I_s and its mask M_s , DASeg predicts the mask M'_q of intra-video query image I_q . The backbone for extracting features F_s and F_q is pre-trained and kept frozen during training.

features which constrained by expert knowledge. Traditional methods struggle to capture the variability of complex traffic environments and the diversity of road structures, limiting their applicability to specific environments. Current approach for road segmentation mainly relies on deep learning [7, 9, 26, 29]. Deep neural networks automatically extract semantic features from the image data, while reducing reliance on prior knowledge and eliminating constraints related to road types and obstacle types. Therefore, it progressively become the mainstream solution for the road perception.

While deep learning models provide high accuracy, their substantial parameters and computational demands make exclusive deployment inefficient for onboard vehicle systems with strict resource constraints. Previous methods also primarily utilize single images due to dataset limitations, failing to leverage the temporally dynamic nature of drivable areas evident across video sequences captured during continuous driving. However, incorporating inter-frame context could enhance segmentation robustness, especially under complex conditions involving transient occlusions or dynamic scenes. The observation inspires us to introduce a lightweight video-based framework for real-time drivable area segmentation by incorporating inter-frame context. Our contributions in this work can be summarized as:

- We propose DASeg to obtain coarse segmentation annotations of query images within a video with annotated support image. The proposed method achieves the state-of-the-art performance compared with recent methods
- Our method shares the backbone with the lane detection method [5], and introduces only 0.43M additional parameters, aiming to reduce the computational burden on on-vehicle chip.
- We manually annotated the detailed mask of the drivable area, based on the public video-based lane detection dataset VIL-100 [35]. Our new dataset VDA-100 is open source and could facilitate the community.

2 Related Work

2.1 Fully-Automatic Segmentation

Related with our work, we divide The fully-automatic segmentation method into two categories: General semantic segmentation and road segmentation.

General semantic segmentation, the task of assigning semantic labels to every pixel in an image has been a fundamental problem in computer vision with wide-ranging applications. One of the pioneering works in this area is the DeepLab series, which has made remarkable contributions. DeepLab V3 [4] employed an Atrous Spatial Pyramid Pooling module to capture multi-scale contextual information, the ASPP module uses parallel atrous convolutions with different dilation rates to extract features at multiple scales. Another notable approach is SegFormer [33], which introduced a multi-scale transformer encoder to capture both local and global contextual information, each transformer blocks operates on features at a different resolution.

Road segmentation method can be divided according to the type of sensor used, and the main sensors used are monocular camera, binocular camera, and lidar. The existing drivable area segmentation methods can be classified based on the type of sensor employed. The primary sensors utilized in these methods include monocular camera, binocular camera, and LiDAR. In terms of the fusion of RGB images and LiDAR point clouds, LiDAR-camera [9] built two pipelines for daytime and nighttime respectively. The unsupervised method [18] integrates image coordinates and LIDAR information to generate a Delaunay triangulation that captures the spatial relationship among obstacle points. The method of fusing RGB images with surface normal maps has also produced promising results. The cross-modal domain adaptation framework [29] introduces the collaborative cross-guidance module to enable cross-modal in-domain sample supplementation, and a selective feature alignment module is introduced to bridge the domain gap between the source domain and the target domain. SNE-RoadSeg [7] first introduces a surface normal estimator to infer surface normal map from dense depth image and proposes a data-fusion module to extract and fuse features from both RGB images and inferred surface normal map. DFM-RTFNet [26] proposed a dynamic fusion module to dynamically fuse two different modalities of features in a multi-scale fashion, the fused feature is processed by five decoder layers and a softmax layer to output the result.

2.2 Semi-automatic Segmentation

The semi-automatic segmentation method can be divided into two categories: video object segmentation and few-shot semantic segmentation.

Semi-automatic video object segmentation methods primarily lie in the setting of first-frame mask propagation. These methods can be categorized based on how they utilize the object masks provided at test time. As an online Fine-tuning method [1] start with a pre-trained base CNN for image labeling on

ImageNet, then train a parent network to improve but are not focused on a specific object, finally fine-tuning on a segmentation example for the specific target object in a single frame. The propagation-based method VPns [11] uses the previous frame mask to infer the current mask, which combines two components, a temporal bilateral network for dense and video adaptive filtering, followed by a spatial network to refine features and increased flexibility. The matching-based methods aim to distinguish the target area from the background based on the pixel-level similarity between two object units, [22] proposed a siamese network that uses features from different depth layers to take advantage of both the spatial details and semantic information. RANet [30] employed an encoder-decoder framework to learn pixel-level similarity and segmentation in an end-to-end manner and proposed a ranking attention module, which automatically ranks and selects these maps for fine-grained performance. To alleviate the demand for large-scale, pixel-wise annotated training samples, several un-/weakly-supervised learning-based methods were recently developed. LIIR [13] exploits cross-video affinities as extra negative samples within a unified, inter-and intra-video reconstruction scheme.

Few-shot semantic segmentation, the task of adapting a segmentation model to novel categories given only a few examples. This task poses unique challenges, as models must quickly learn to segment new classes with limited training data. PPNet [17] introduces a part-aware prototype learning mechanism, which extracts region-level features from support images and aggregates to construct class-level prototypes. PANet [27] proposed prototype generation module to construct class-level prototypes from the support images, and introduced the prototype alignment module to align prototypes with the query image features to produce the final segmentation outputs. The key innovation of DGPNNet [12] is the use of Dense Gaussian Processes (DGPs) to model the task-specific segmentation distribution. DGPs can capture rich contextual dependencies in the segmentation maps, allowing for accurate adaptation with limited data. Motivated by the simple Gestalt principle that pixels belonging to the same object are more similar than those to different objects of same class, SSP [6] uses query prototypes to match query features, where the query prototypes are collected from high-confidence query predictions.

3 Preliminary

3.1 Problem Formulation

Classical fully-automatic segmentation methods need to use a large amount of training data to learn the standard method:

$$\hat{f} \in \arg \min_{f \in \mathcal{F}} \frac{1}{N_t} \sum \varepsilon(f(I_t), M_t), \quad (1)$$

where \mathcal{F} denotes the hypothesis space, and ε is an error function that evaluates the estimate $f(\cdot)$ against corresponding label. The I_t and M_t denote the training

data, which are selected from the entire dataset and are not restricted to the same video or scenario, the N_t represents the number of training samples, and it is usually very large, taking the BDD100K dataset [34] as an example, $N_t = 80K$. However, through the use of temporal consistency prior and supporting image guidance, our method can be trained on a small number of labeled data sets to obtain a high-performance annotation tool for the drivable area.

Our video-based semi-automatic segmentation models are typically learned in a fully supervised manner, requiring N_q input training samples and their annotations, as shown in Fig. 1. The standard method for evaluating learning outcomes follows an empirical loss minimization formulation:

$$\hat{f} \in \arg \min_{f \in \mathcal{F}} \frac{1}{N_q} \sum \varepsilon(f(I_q, I_s, M_s), M_q). \quad (2)$$

Given the query image I_q , we aim to generate a segmentation of the drivable area M'_q by leveraging the support image I_s and its well-annotated mask M_s . Both the support image $I_s \in \mathbb{R}^{1 \times 3 \times H \times W}$ and the query image $I_q \in \mathbb{R}^{N_q \times 3 \times H \times W}$ are assumed to be from the same video. To make \hat{f} a good approximation, current fully-automatic segmentation methods directly use the desired output M_q , as the prior knowledge, with the price of requiring vast amounts of well-annotated data. Through the pattern of formula 2, our semi-automatic method only requires 0.8K of annotation data, far less than the 80K of the fully-automatic methods.

3.2 Dataset

The public datasets for drivable area segmentation as shown in Table 1. SYNTHIA and R2D are collected in the virtual simulator, KITTI Road and BDD100K are collected in the real world. KITTI Road offers RGB-D data but only 289 images are collected. The widely used BDD100K is not a video-based dataset, which does not contain temporal information, and does not meet our needs. VIL-100 is a video-based dataset for lane detection, but does not contain drivable area annotation.

Table 1. Related datasets. The DA and Lane columns show whether there are annotations of drivable area and lane lines, respectively.

Dataset	Domain	Number	DA	Lane	Type
KITTI Road [8]	Real	289	✓	✗	Image
SYNTHIA [21]	Synthetic	~13K	✓	✗	Image
R2D [7]	Synthetic	~11K	✓	✗	Image
BDD100K [34]	Real	80K	✓	✓	Image
VIL-100 [35]	Real	10K	✗	✓	Video
VDA-100	Real	10K	✓	✓	Video

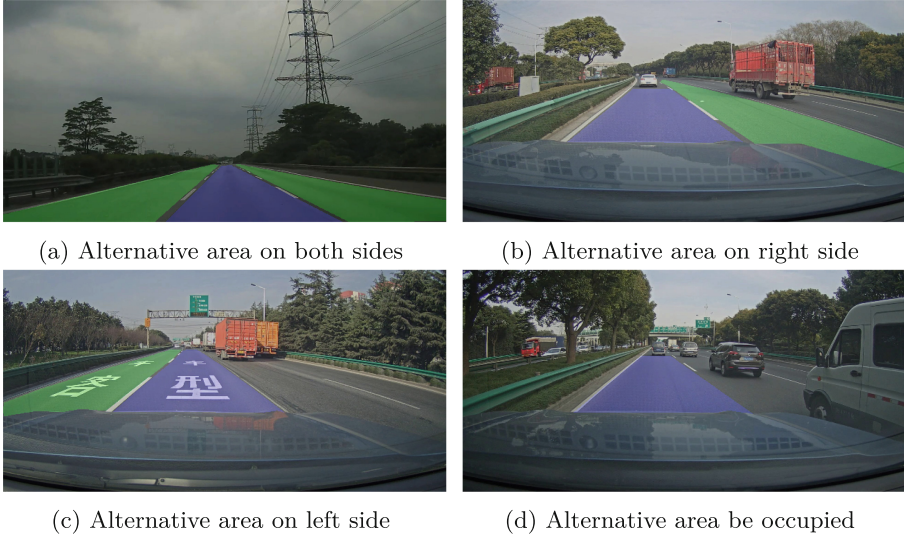


Fig. 2. Annotations of drivable area. Different from the definition of road surface, the drivable area must consider traffic regulations. The side of the direct area is divided by a dotted line and is not occupied, so it is marked as an alternative area, as shown in Fig. 2a and Fig. 2b. If the side of the direct area is divided by a solid line or is already occupied, it cannot be marked as an alternative area, as shown in Fig. 2c and Fig. 2d.

To obtain an available dataset, we manually annotate detailed masks of drivable areas based on the VIL-100 [35]. The new dataset called VDA-100 was used for training and evaluation. As shown in Fig. 2, we selected four annotated images to illustrate annotation principles.

The first 10 frames of each video was manually annotated, with 80% of the total 100 videos allocated for training and the remaining 20% for evaluation. During training, a single frame is randomly selected from the manually annotated frames as the support image, while the rest of the intra-video images serve as query images. The proposed method aims to use the limited number of manual annotations to achieve coarse annotating of the rest unlabeled intra-video images.

4 Proposed Method

Inspired by prototype matching [22, 30], we construct an embedded space that incorporates appearance information and position information, as shown in Fig. 3. The embedded space is used to store modified query prototypes obtained through soft pooling. Subsequently, labels are assigned to each pixel based on the similarity between query image feature and prototypes within the embedded space.

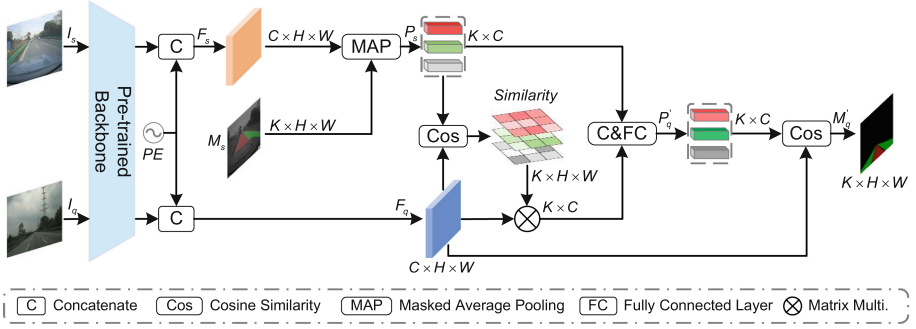


Fig. 3. Proposed network architecture.

4.1 Pixel-Wise Position Embedding

Unlike other visual concepts such as cats or dogs, the concept of a drivable area cannot be defined based on its appearance alone, positional information is also required. Thus, we introduce position embedding to improve inter-class discriminability by incorporating explicit positional priors. Previous segmentation methods [14, 20] solve the problem by training a discriminative classifier $p(Class|Appearance)$, which neglects the underlying data distribution $p(Position|Class)$. In our observation, the direct area and the alternative area are composed of similar road surface pixels. That is, the distribution of appearance features $p(Appearance|Class_D)$ and $p(Appearance|Class_A)$ are close. Therefore, it is not discriminative to solely rely on the appearance feature.

To solve the above problem, we use positional information to make image features more discriminative. Based on our observations, the direct area predominantly occupies the central region of the image, whereas the alternative area is typically situated on the left or right sides. Namely, the gap between $p(Position|Class_A)$ distribution and $p(Position|Class_D)$ distribution is large. With the prior mentioned above, we solve the drivable area segmentation problem by training a discriminative classifier $p(Class|(Appearance, Position))$. In this work, we use sine and cosine functions of different frequencies to encode 2D positional information of each pixel in appearance feature:

$$\begin{aligned}
 PE_{(x,y,4i)} &= \sin\left(\frac{x}{T^{4i/D}}\right), \\
 PE_{(x,y,4i+1)} &= \cos\left(\frac{x}{T^{4i/D}}\right), \\
 PE_{(x,y,4i+2)} &= \sin\left(\frac{y}{T^{4i/D}}\right), \\
 PE_{(x,y,4i+3)} &= \cos\left(\frac{y}{T^{4i/D}}\right),
 \end{aligned} \tag{3}$$

where the T is a hand-design temperature, a larger T results in a more flattened attention map, and vice versa [24]. To make sure the max wavelength $2\pi \cdot T$ is always larger than the spatial dimension $\max(H, W)$, we choose $T = 16$ in

our method. D represents the channel size of the appearance feature which is extracted by backbone $\mathcal{B}(I) \in \mathbb{R}^{D \times H \times W}$, $i \in [0, 1, \dots, 4/D - 1]$ is the channel index, x, y is the location of pixel. The position embedding has the same size as the appearance feature.

To decouple the appearance and position contributions to the prototype-to-feature similarity computed as dot product between query and support, we concatenate the position embedding and appearance feature in channel dimension as query feature F_s and support feature F_s for subsequent processes:

$$\begin{aligned} F_s &= \text{concat}(\mathcal{B}(I_s), PE_s), \\ F_q &= \text{concat}(\mathcal{B}(I_q), PE_q), \end{aligned} \quad (4)$$

where $F_s, F_q \in \mathbb{R}^{2D \times H \times W}$, for simplicity, we denote $C = 2D$ in the following sections. Both the positional embedding in queries and supports are generated based on 2D coordinates, which makes it more consistent to compare the positional similarity.

4.2 Prototype Soft Pooling

Queries in DETR series [2, 16, 28] can be interpreted as soft-pooling feature from a feature map based on the query-to-feature similarity, which considers both the appearance and position information. While the appearance similarity is for pooling semantically support feature, the positional similarity is to provide a positional constraint for pooling feature around the query position. Inspired by the SSP [6], we proposed soft-pooling to gather more representative query prototypes for alleviating the limited coverage of intra-class variations from the support provided, with the principle that pixels belonging to the same foreground are more similar than those from different foregrounds.

First, we use masked average pooling to collect support prototypes P_s with support mask M_s :

$$P_s^i = \frac{1}{\|M_s^i\|_1} \sum M_s^i \otimes F_s^T, \quad (5)$$

where the $i \in [0, 1, \dots, K]$ is the class index, We set $K = 3$ to denote the number of classes, with each class representing a specific area: background, direct area, and alternative area. M_r^i and $\|M_s^i\|_1$ represent the mask and the total number of pixels, respectively, belonging to the i th class. F_s is the position-encoded support feature, $P_s \in \mathbb{R}^{K \times C}$ denote support prototypes specifically correspond to the three different classes.

The cosine similarity $\text{CosSim}(P_s, F_q) \in \mathbb{R}^{K \times H \times W}$ between query feature F_q and support prototypes P_s decides how much the query prototype should soft-pooling from each pixel of the query feature itself, the cosine similarity can be formulated as:

$$\text{CosSim}(P, F)_{i,j} = \frac{P^i \cdot F^j}{\|P^i\|_2 \cdot \|F^j\|_2}. \quad (6)$$

We employ the cosine distance due to its enhanced stability and superior performance compared to other distance metrics, such as squared Euclidean distance.

Additionally, cosine distance is bounded, which facilitates ease of optimization. The query prototype P'_q used to classify each pixel is calculated by weighting the soft-pooled prototype and the support prototype in the channel dimension through a fully connected layer, the whole procedure can be formulated as:

$$P'_q = MLP(\text{concat}(P_s, \text{CosSim}(P_s, F_q) \otimes F_q)). \quad (7)$$

Then, we use the generated query prototypes $P'_q \in \mathbb{R}^{K \times C}$ to perform self-matching with each pixel of query feature by calculating the cosine similarity:

$$M'_q = \text{CosSim}(P'_q, F_q), \quad (8)$$

where the $M'_q \in \mathbb{R}^{K \times H \times W}$ means the probability that each pixel is classified into K semantic classes.

4.3 Prototype Regularization

The preceding subsections outline the inference pipeline for utilizing an annotated noise-free support set $\{I_s, M_s\}$ to predict the query mask M'_q of query image I_q . In order to enhance the generalization ability of the model by appropriately increasing the input noise, we use the query image I_q and the predicted noise mask M'_q to realize the robust prediction of support mask M'_s . We call the reverse segment pipeline as prototype regularization, which introduces noise into solid prototype to mimic realistic prototype generation, the prototype regularization can be formulated as:

$$\begin{aligned} P'_s &= MLP(\text{concat}(P_q, \text{CosSim}(P_q, F_s) \otimes F_s)), \\ M'_s &= \text{CosSim}(P'_s, F_s), \end{aligned} \quad (9)$$

where the P_q is obtained by masked average pooling with query feature I_q and predicted mask M'_q , the F_s is the support feature obtained as described in Sect. 4.1.

After computing the probability masks M'_q and M'_s of the query image and support image through two opposite pipelines, we calculate the segmentation loss \mathcal{L}_{seg} as follows:

$$\mathcal{L}_{seg}(M, M') = -\frac{1}{HW} \sum_{i=1}^K \sum_{j=1}^{HW} M_{(i,j)} \log(M'_{(i,j)}), \quad (10)$$

where M is the ground truth and the M' is the prediction. Optimizing the above loss will derive suitable prototype for each class. The total training loss consists of two parts of segment loss:

$$\mathcal{L}_{total} = \lambda_q \cdot \mathcal{L}_{seg}(M_q, M'_q) + \lambda_s \cdot \mathcal{L}_{seg}(M_s, M'_s), \quad (11)$$

where the λ_q and λ_s are the weights used to balance the influence of two opposite pipelines on network parameters.

5 Experiments

5.1 Implementation Details

The input images are resized to 320×640 pixels and the batchsize is set to 8. During training, the AdamW optimizer is employed over 60 epochs with the momentum value of 0.9, the weight decay value of $5e-4$, and the initial learning rate of $1e-3$. Cosine annealing scheduler is used to gradually decrease the learning rate to 0. We did the tuning experiment and found that the performance is best when the loss balance weights $\lambda_q, \lambda_s \in [0, 1]$ are set to 0.4 and 0.6 respectively. We employ vanilla ResNet [10] with FPN [15] as the backbone. Data augmentation contains random affine transformations (translation, rotation, and scaling) and random horizontal flips. All experiments were performed on a machine equipped with an Intel i7-10700K processor and a single RTX 2080Ti GPU.

5.2 Results

Table 2. Comparison on VDA-100 dataset. The IoU_D and IoU_A denotes the IoU metric of the direct area and alternative area, respectively, and the mIoU is the mean IoU of segmented classes.

Type	Method	mIoU \uparrow	$\text{IoU}_D\uparrow$	$\text{IoU}_A\uparrow$	#Params \downarrow
Fully-automatic	DeepLabV3+ [4]	70.0	72.4	67.5	15.4M
	SegFormer [33]	69.2	70.7	67.7	7.2M
	YOLOP [32]	72.3	75.6	69.0	5.53
	Sparse U-PDP [25]	72.9	74.2	71.6	18.8M
	TwinLiteNet ⁺ [3]	72.6	73.8	71.3	0.44M
Semi-automatic	PPNet [17]	82.4	83.0	81.8	31.5M
	DGPNNet [12]	80.5	82.5	78.4	20.9M
	SSP [6]	82.5	85.3	79.6	8.7M
	PANet [27]	83.8	87.0	80.6	14.7M
	LIIR [13]	85.0	85.6	84.3	13.4M
	VPNs [11]	82.6	84.2	80.9	21.1M
	RANet [30]	84.0	86.5	81.5	14.5M
	DASeg	88.3	90.7	85.8	0.43M

As shown in Table 2, the proposed method achieves promising performance with an IoU score of 90.7% for direct area and 85.8% for alternative area, outperform the previous few-shot based methods and road segmentation methods. Comparing to direct area, the position of alternative area exhibits greater flexibility and is affected by the driving environment and traffic regulations. Thus, the experiments indicate that the IoU_A score is approximately 5% lower than the IoU_D score.

Table 3. Ablation studies. For simplicity, PE means position embedding, SP means soft-pooling, PR means prototype regularization.

Method	mIoU \uparrow	IoU $_D$ \uparrow	IoU $_A$ \uparrow	Params \downarrow
w/o PE	83.5	89.0	77.9	0.35
w/o SP	70.3	90.0	50.6	0.21
w/o PR	85.5	90.2	80.7	0.43
DASeg	88.3	90.7	85.8	0.43M

5.3 Ablation Study

To compare the effects of the three components on performance, we utilize the T-SNE [19] to visualize the embedding space, and the channel dimension of image features and prototypes are reduced from 256 to 3, as shown in Fig. 4. Comparing Fig. 4a with Fig. 4b, the gap between red and green points in the embedded space increases, which means that the position embedding makes the two types of foreground features more discriminative. Comparing Fig. 4c with Fig. 4a, a large number of green points are not included in the green sphere, which means that the soft-pooling strategy can make the prototype of the alternative area more representative, which is achieved by reducing the distance between the prototype and the feature cluster in the embedding space. Comparing Fig. 4d with Fig. 4a, the number of green points outside the green sphere increases, thus the noise introduced by the prototype regularization procedure improves the generalization of the proposed method.

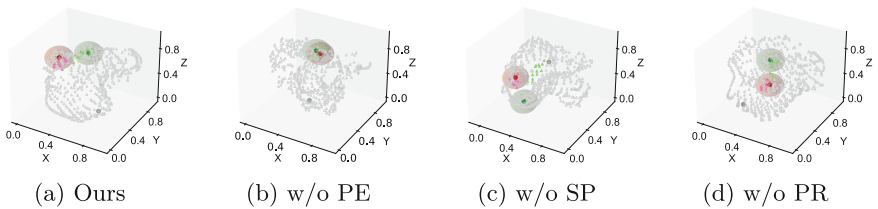


Fig. 4. Feature visualization. The points in gray, red, and green correspond to the pixels from the background, direct area, and alternative area, respectively. The red and green spheres depict the prototypes of the direct area and alternative area. (Color figure online)

To quantitatively analyze the effects of position embedding, soft-pooling, and prototype regularization proposed in this work, we take the ResNet18 as the baseline and conduct ablation experiments as shown in Table 3. The ablation experiment shows that the performance of three components bring gain of 4.8%, 18%, and 2.8% on the mIoU score, respectively. Notably, the soft-pooling strategy achieves a 35.8% gain in performance for alternative areas.

In our observation, the lanes serve as the contours of the drivable area in real driving scenarios. Therefore, the extracted image feature employed in lane detection and drivable area segmentation can be shared. To verify the feasibility of sharing the backbone between the lane detection method and the drivable area segmentation method, we incorporate the proposed segmentation branch with the pre-trained backbone from DIIane [5].

Table 4. Feasibility verification of the shared backbone. Comparing the performance of using the backbone trained from scratch and pre-trained by lane detection task.

Pre-trained	mIoU \uparrow	IoU $_D$ \uparrow	IoU $_A$ \uparrow	Params \downarrow
\times	89.1	91.2	86.9	11.77M
\checkmark	88.3	90.7	85.8	0.43M

The comparative experimental results are shown in Table 4, employing the pre-trained backbone requires training only 0.43M parameters, leading to a very slight decrease of 0.5% in IoU $_D$ score and a 1.1% decrease in IoU $_A$ score. Therefore, the pre-trained backbone of lane detection has solid generalization for the task of drivable area segmentation. Therefore, the proposed method can be used as a plug-and-play branch to integrate with lane detection applications, and only 0.43M more parameters are added.

To investigate the impact of temporal distance on the results, we randomly selected two images from the evaluation set videos as the support and query images, respectively. The maximum number of frames in each video was 100. To facilitate the analysis, we recorded the mIoU indicators at different distances between the support and query images. We then standardized the mIoU values using the formula $\frac{mIoU - \mu_{mIoU}}{\sigma_{mIoU}}$, where μ_{mIoU} and σ_{mIoU} are the mean and standard deviation of the mIoU, respectively. Finally, we performed curve fitting and visualization, as shown in Fig. 5.

The performance of the proposed method only decreases slightly as the distance increases. Because of prototype regularization proposed in Sect. 4.3, noise is introduced to enhance generalization.

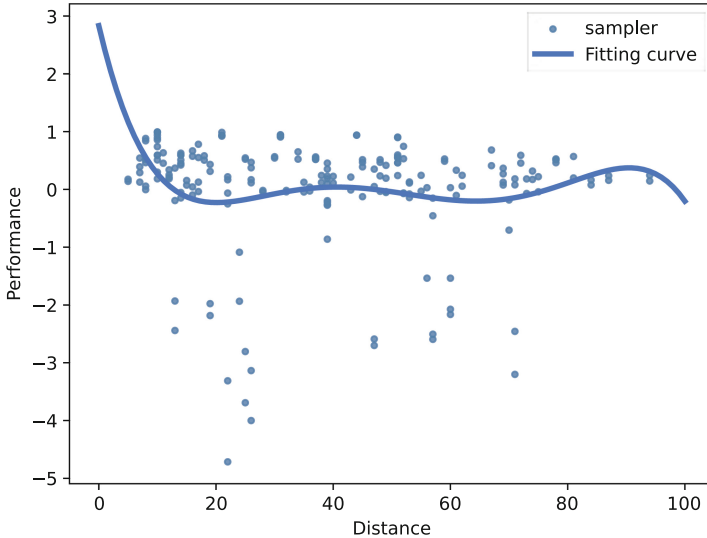


Fig. 5. Impact of the temporal distance. The horizontal axis represents the number of frames between the support and query image in the video, and the vertical axis represents the normalized mIoU performance.

6 Conclusion

We introduce a video-based semi-automatic drivable area segmentation method along with VDA-100 dataset. Our method requires only one annotated image to achieve coarse annotation for intra-video images. To alleviate the computational burden on on-vehicle chip, our method can function as a tiny plug-and-play branch, sharing the backbone with the lane detection method and achieving a promising performance.

Acknowledgments. This work was partially supported by a grants from the National Key R&D Program of China (2018AAA0102801 and 2018AAA0102803), the National Natural Science Foundation of China (61772424, 61702418, and 61602383), and the Air Science Foundation (2018ZE53052).

References

1. Caelles, S., Maninis, K.K., Pont-Tuset, J., Leal-Taixé, L., Cremers, D., Van Gool, L.: One-shot video object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 221–230 (2017)
2. 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)
3. Che, Q.H., Le, D.T., Pham, M.Q., Nguyen, V.T., Lam, D.K.: Twinlitenetplus: a stronger model for real-time drivable area and lane segmentation. arXiv preprint [arXiv:2403.16958](https://arxiv.org/abs/2403.16958) (2024)

4. Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proceedings of the European conference on computer vision (ECCV), pp. 801–818 (2018)
5. Cheng, Z., Zhang, G., Wang, C., Zhou, W.: Dilane: dynamic instance-aware network for lane detection. In: Proceedings of the Asian Conference on Computer Vision, pp. 2075–2091 (2022)
6. Fan, Q., Pei, W., Tai, Y.W., Tang, C.K.: Self-support few-shot semantic segmentation. In: European Conference on Computer Vision, pp. 701–719. Springer (2022)
7. Fan, R., Wang, H., Cai, P., Liu, M.: Sne-roadseg: incorporating surface normal information into semantic segmentation for accurate freespace detection. In: European Conference on Computer Vision, pp. 340–356. Springer (2020)
8. Fritsch, J., Kuehnl, T., Geiger, A.: A new performance measure and evaluation benchmark for road detection algorithms. In: 16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013), pp. 1693–1700. IEEE (2013)
9. Gu, S., Yang, J., Kong, H.: A cascaded lidar-camera fusion network for road detection. In: 2021 IEEE International Conference on Robotics and Automation (ICRA), pp. 13308–13314. IEEE (2021)
10. 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)
11. Jampani, V., Gadde, R., Gehler, P.V.: Video propagation networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 451–461 (2017)
12. Johnander, J., Edstedt, J., Felsberg, M., Khan, F.S., Danelljan, M.: Dense gaussian processes for few-shot segmentation. In: European Conference on Computer Vision, pp. 217–234. Springer (2022)
13. Li, L., Zhou, T., Wang, W., Yang, L., Li, J., Yang, Y.: Locality-aware inter-and intra-video reconstruction for self-supervised correspondence learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8719–8730 (2022)
14. Liang, C., Wang, W., Miao, J., Yang, Y.: GMMSeg: gaussian mixture based generative semantic segmentation models. *Adv. Neural. Inf. Process. Syst.* **35**, 31360–31375 (2022)
15. 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)
16. Liu, S., et al.: DAB-DETR: dynamic anchor boxes are better queries for DETR. In: International Conference on Learning Representations (2021)
17. Liu, Y., Zhang, X., Zhang, S., He, X.: Part-aware prototype network for few-shot semantic segmentation. In: Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IX 16, pp. 142–158. Springer (2020)
18. Liu, Z., Yu, S., Wang, X., Zheng, N.: Detecting drivable area for self-driving cars: an unsupervised approach. *arXiv preprint [arXiv:1705.00451](https://arxiv.org/abs/1705.00451)* (2017)
19. Van der Maaten, L., Hinton, G.: Visualizing data using t-sne. *J. Mach. Learn. Res.* **9**(11) (2008)
20. Oliveira, G.L., Burgard, W., Brox, T.: Efficient deep models for monocular road segmentation. In: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 4885–4891. IEEE (2016)

21. Ros, G., Sellart, L., Materzynska, J., Vazquez, D., Lopez, A.M.: The synthia dataset: a large collection of synthetic images for semantic segmentation of urban scenes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3234–3243 (2016)
22. Shin Yoon, J., Rameau, F., Kim, J., Lee, S., Shin, S., So Kweon, I.: Pixel-level matching for video object segmentation using convolutional neural networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2167–2176 (2017)
23. Suhr, J.K., Jung, H.G.: Noise-resilient road surface and free space estimation using dense stereo. In: 2013 IEEE Intelligent Vehicles Symposium (IV), pp. 461–466. IEEE (2013)
24. Vaswani, A., et al.: Attention is all you need. In: Advances in Neural Information Processing Systems, vol. 30 (2017)
25. Wang, H., Qiu, M., Cai, Y., Chen, L., Li, Y.: Sparse u-pdp: a unified multi-task framework for panoptic driving perception. *IEEE Trans. Intell. Transp. Syst.* (2023)
26. Wang, H., Fan, R., Sun, Y., Liu, M.: Dynamic fusion module evolves drivable area and road anomaly detection: a benchmark and algorithms. *IEEE Trans. Cybern.* **52**(10), 10750–10760 (2021)
27. Wang, K., Liew, J.H., Zou, Y., Zhou, D., Feng, J.: Panet: few-shot image semantic segmentation with prototype alignment. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 9197–9206 (2019)
28. Wang, Y., Zhang, X., Yang, T., Sun, J.: Anchor detr: query design for transformer-based detector. In: Proceedings of the AAAI Conference on Artificial Intelligence (2022)
29. Wang, Y., et al.: Cross-modality domain adaptation for freespace detection: a simple yet effective baseline. In: Proceedings of the 30th ACM International Conference on Multimedia, pp. 4031–4042 (2022)
30. Wang, Z., Xu, J., Liu, L., Zhu, F., Shao, L.: Ranet: ranking attention network for fast video object segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 3978–3987 (2019)
31. Wedel, A., Badino, H., Rabe, C., Loose, H., Franke, U., Cremers, D.: B-spline modeling of road surfaces with an application to free-space estimation. *IEEE Trans. Intell. Transp. Syst.* **10**(4), 572–583 (2009)
32. Wu, D., et al.: Yolop: you only look once for panoptic driving perception. *Mach. Intell. Res.* **19**(6), 550–562 (2022)
33. 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)
34. Yu, F., et al.: Bdd100k: a diverse driving dataset for heterogeneous multitask learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2636–2645 (2020)
35. Zhang, Y., et al.: Vil-100: a new dataset and a baseline model for video instance lane detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 15681–15690 (2021)



CASPFormer: Trajectory Prediction from BEV Images with Deformable Attention

Harsh Yadav¹(✉), Maximilian Schaefer², Kun Zhao², and Tobias Meisen¹

¹ University of Wuppertal, Wuppertal, Germany
{harsh.yadav,meisen}@uni-wuppertal.de

² Aptiv Services Deutschland GmbH, Wuppertal, Germany
{maximilian.schaefer,kun.zhao}@aptiv.com

Abstract. Motion prediction is an important aspect for Autonomous Driving (AD) and Advance Driver Assistance Systems (ADAS). Current state-of-the-art motion prediction methods rely on High Definition (HD) maps for capturing the surrounding context of the ego vehicle. Such systems lack scalability in real-world deployment as (HD) maps are expensive to produce and update in real-time. To overcome this issue, we propose Context Aware Scene Prediction Transformer (CASPFormer), which can perform multi-modal motion prediction from rasterized BEV images. Our system can be integrated with any upstream perception module that is capable of generating BEV images. Moreover, Context Aware Scene Prediction Transformer (CASPFormer) directly decodes vectorized trajectories without any post-processing. Trajectories are decoded recurrently using deformable attention, as it is computationally efficient and provides the network with the ability to focus its attention on the important spatial locations of the BEV images. In addition, we also address the issue of mode collapse for generating multiple scene-consistent trajectories by incorporating learnable mode queries. We evaluate our model on the nuScenes dataset and show that it reaches state-of-the-art across multiple metrics.

Keywords: Autonomous Driving · Multi-Modal Trajectory Prediction · Deformable Attention

1 Introduction

In recent years, AD and ADAS technologies have gained huge attention as they can significantly improve the safety and comfort standards across the automotive industry [20]. The current approach to these self-driving tasks is to divide them into multiple independent sub-tasks, mainly i) perception, ii) motion prediction, and iii) motion planning, and optimize each task individually [7]. The perception task deals with the detection and segmentation of surrounding dynamic and static environment contexts. The dynamic context captures the motion of the

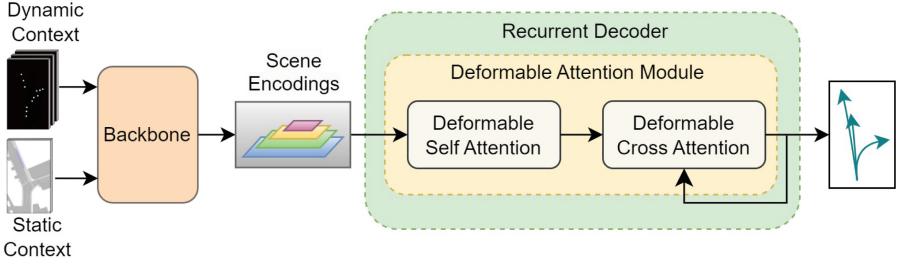


Fig. 1. Shows an overview of the CASPFormer architecture. The backbone uses CNN and convolution RNN to generate the scene encodings. The scene encodings have a pyramid structure with increasing resolution from top to bottom. The deformable self-attention module applies a multi-scale feature fusion on the scene encodings, while the deformable cross-attention module recurrently decodes the trajectories. The output of the previous time step is used to update the position of the reference point and the query embeddings in deformable cross-attention.

dynamic agents in the scene e.g. pedestrians, cyclists, vehicles, traffic lights, etc., while the static context includes stationary elements of the scene e.g. road and lane boundaries, pedestrian crossings, traffic signs, parked vehicles, construction sites etc. As defined by Cui et al. [9], the motion prediction task involves predicting multi-modal future trajectories for agents in a scene. The prediction of multiple future trajectories enables the model to account for uncertainties in the dynamic context. In addition, to ensure safety critical operation, the predicted trajectories must adhere to the static and dynamic contexts. Lastly, the objective of the motion planning task is to generate the control actions for the ego vehicle to navigate it through the scene while adhering to the traffic rules and dynamics of the vehicle.

Current state-of-the-art models [17, 24, 26, 29, 35, 36] in motion prediction require HD maps for static context with centimeter-level accuracy. Such a strict constraint on HD maps leads to high production costs [4]. Thus, these models suffer from the problem of scalability in a real-world deployment. A cost-effective and scalable alternative is to construct BEV images from a vision perception system deployed on the ego vehicle, as proposed by Li et al. [19] in their BEVFormer model. To efficiently decode trajectories and learn spatial attention on the feature maps of BEV images, we opt for the deformable attention mechanism proposed in Deformable Detection Transformer (DETR) [37]. Furthermore, to generate a diverse set of modes in multi-modal trajectory prediction, we incorporate learnable embeddings into our architecture. Contrary to previous studies [17, 31, 35], which use one set of learnable embeddings, our network consists of two sets of learnable embeddings. The first set, temporal queries, is responsible for capturing the temporal correlation in the output trajectories, and the second set, mode queries, aims to address the issue of mode collapse. Following the works [29, 35], we recurrently decode the multi-modal trajectories. This

allows the network to update the reference point for deformable attention and the temporal queries through feedback loops of the recurrent decoder.

A depiction of our proposed network Context Aware Scene Prediction Transformer (CASPPFormer) is shown in Fig. 1. Furthermore, Fig. 2 highlights the components of the recurrent decoder. The contributions of our work are summarised as follows:

- A novel motion prediction architecture is introduced that generates multi-modal vectorized trajectories from BEV images.
- It incorporates two sets of learnable embeddings: temporal queries for capturing the temporal correlation in the output trajectories and mode queries for overcoming the issue of mode collapse.
- The trajectory decoding is done recurrently using deformable attention where the feedback loops update the reference point for deformable attention and the temporal queries.
- We evaluate our method on the nuScenes motion prediction benchmark [25] and show that it achieves state-of-the-art performance across various metrics.

2 Related Work

In this section, we highlight the corresponding related work. Section 2.1 categorized the previous studies based on how their scene representation is constructed. Section 2.2 highlights various methods for generating multi-modal prediction. Section 2.3 discusses several transformer-based attention mechanisms that can be used to extract meaningful representations from BEV images.

2.1 Input Scene Representation

The scene representation in the motion prediction task can be divided into two categories, rasterized scene representation and vectorized scene representation. The studies with rasterized scene representation [5, 9, 14, 29] take advantage of matured practices in Convolution Neural Networks (CNN) to extract scene encodings. On the other hand, the vectorized representation was first introduced by LaneGCN [20] which identified that HD maps have an underlying graph structure that can be exploited to learn long-range and efficient static scene encodings with Graph Neural Network (GNN). VectorNet [13] later showed that not only the static context, but also the dynamic context can also be represented in vectorized format. Follow-up studies [17, 31, 35, 36] have provided several motion prediction methods that receive both static and dynamic contexts in vectorized form.

2.2 Multi-modal Prediction

To accommodate uncertainties in traffic scenarios, autonomous vehicles must predict various scene-consistent trajectories adhering to the static and dynamic

context. One approach [3, 8] employs a variational auto-encoder to learn multiple latent representations of the entire scene and then decodes these latent representations generating multiple trajectories corresponding to each agent. However, these methods require multiple forward passes during both training and inference and are prone to mode collapse. Other approaches [14, 29] use spatial-temporal grids to predict the future position for each agent and sample multiple goal positions. Thereafter, scene-consistent trajectories are generated which connect the proposed goal positions with the current position of the agents. These approaches learn multi-modality inherently without a specific training strategy, however, post-processing is required to generate trajectories from the grid. Alternatively, Multipath [5] utilizes fixed anchors corresponding to different modes. It constructs multiple trajectories by generating the offsets and probability distribution corresponding to each one of these anchors. A potential limitation of Multipath is that most of the fixed anchors are not relevant for particular scenes. This issue is addressed in the follow-up studies [17, 31, 35], which learn the anchors during the training with the help of learnable embeddings and predict a diverse set of modes.

2.3 Transformer-Based Attention in Image Domain

In recent years, transformer-based attention [32] mechanisms have achieved huge success in the image domain. The studies [11, 22, 33] establish the foundation for transformer-based encoders for image processing. Since these approaches lack decoder networks, their application is limited only to feature extraction. DETR [2] introduces a transformer-based encoder-decoder architecture capable of end-to-end object detection. However, DETR suffers from two major problems: slow convergence and low performance in detecting small objects, as its encoder is limited to processing features with very small resolution due to its quadratic computational complexity with the size of feature maps.

Deformable DETR [37] overcomes these problems by sparsifying the selection of values and computing the attention solely based upon queries whilst eliminating the need for keys. The decrease in computational cost allows both the encoder and decoder to attend to every feature map in the feature pyramid generated by the backbone. Deformable DETR thus significantly reduces training time while increasing performance in detecting small objects. Follow-up studies [18, 28] on Deformable DETR establish that a large part of its computational cost comes from the deformable self-attention module, and therefore propose to reduce this cost by limiting the number of queries which undergo self-attention. We compare training time with and without deformable self-attention modules in ablation studies because computational cost plays an important role in the deployment of models on edge devices operating in vehicles.

3 Methods

This section will explain the methods which are utilized in our work and in particular our contribution to the current state of the art. Section 3.1 describes

the formulation of the input and output of the network. Section 3.2 focuses on network architecture of CASPFormer and its components. Section 3.3 illustrates the loss formulation.

3.1 Input-Output Formulation

CASPFormer receives static and dynamic contexts of the surrounding region of the ego vehicle and outputs multi-modal vectorized trajectories.

Static Context Input. The static context is rasterized into a grid-based input of shape (H, W) . The feature dimension of rasterized static context contains binary feature maps consisting of information about the drivable area, center lines, driving lanes, road boundaries, and pedestrian crossing. The input of static context can be depicted as follows:

$$I_s \in \mathbb{R}^{H \times W \times |F_s|}, \quad (1)$$

where H is the height of the grid, W is the width of the grid, and $|F_s|$ is the number of input features of static context.

Dynamic Context Input. The dynamic context contains the motion information of all the road agents for the past T_i time steps. Corresponding to each time step T_i , a grid of shape (H, W) is created. The motion attributes of each dynamic agent are filled at the nearest voxel to their current position within each temporal grid. These attributes include the dynamic agent’s velocity, acceleration, location offset, height, width, and heading information. The rasterized input of dynamic context thus can be depicted as:

$$I_d \in \mathbb{R}^{T_i \times H \times W \times |F_d|}, \quad (2)$$

where, $|F_d|$ is the number of input features of dynamic context.

Output. The predicted trajectories contain the position information i.e. (x, y) of the ego vehicle, and the output tensor can thus be represented as:

$$Y \in \mathbb{R}^{M \times T_o \times 2}, \quad (3)$$

where, M is the number of modes, T_o is the number of future time steps.

3.2 Network Architecture

The overall network architecture is shown in Fig. 1. The network consists of a backbone and a recurrent decoder. For our work, the backbone architecture is adopted from Context Aware Scene Prediction Network (CASPNet) [29], as it is currently state-of-the-art in the nuScenes dataset [1]. It receives static and dynamic contexts in rasterized formats to generate multi-scale scene encodings. It is important to note that the CASPFormer is not limited to a particular backbone and can be extended to other transformer or CNN based backbones. The

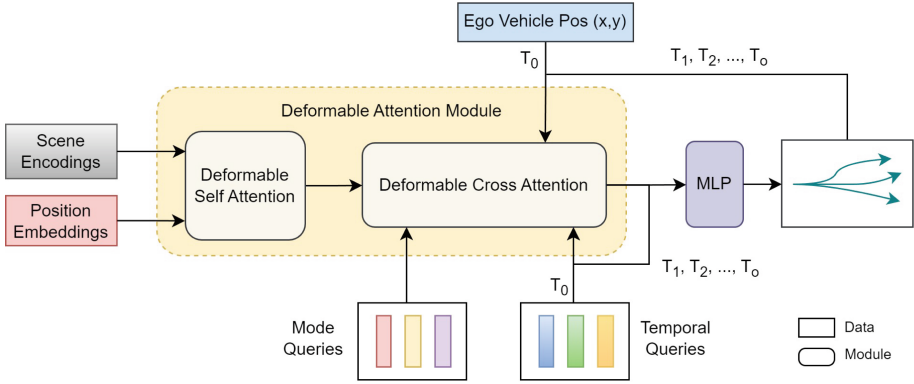


Fig. 2. A depiction of the recurrent decoder network architecture. The position embeddings are non-learnable and help the network in learning the location of features. The mode queries serve the purpose of producing multiple scene-consistent trajectories in the multi-modal output. The temporal correlation in the predicted trajectories is captured with temporal queries. The position of the reference point for the deformable attention is set to the ego vehicle position in the scene. The recurrent architecture updates the ego vehicle position and the temporal queries at every recurrent step.

works [29, 35] suggests that decoding the trajectory in a recurrent fashion results in better prediction capabilities. Inspired by this observation, we also decode the trajectory recurrently from the multi-scale scene encodings. A detailed schematic of the recurrent decoder is depicted in Fig. 2.

The recurrent decoder employs deformable attention [37] to gather essential information from the scene encodings. The deformable attention module consists of deformable self-attention and deformable cross-attention modules. Thereby, the scene encodings are first encoded in the deformable self-attention module, which performs multi-scale feature fusion. The position information in the scene encodings is captured with non-learnable sinusoidal positional embeddings [37]. The fused scene encodings are then processed by a deformable cross-attention module, in which the attention map is learned through a linear transformation of queries. During our initial experiments, we only introduced temporal queries corresponding to each mode. The objective of the temporal queries was two-fold, first, they must learn the temporal correlation across the different time steps in the predicted trajectories, and second, they must distinguish between different modes as illustrated in previous works [17, 31, 35]. However, our preliminary experiments showed that this setup results in mode collapse (see the left column of Fig. 3). We observed that although the different modes do correspond to different speeds, they miss out on other possible scene-consistent trajectories. To overcome this issue, we use another set of queries, called mode queries, in our network architecture. The results show that mode queries significantly improve the diversity of modes (see the right column of Fig. 3). Another aspect of the original deformable cross-attention [37] is that it utilizes reference points to help

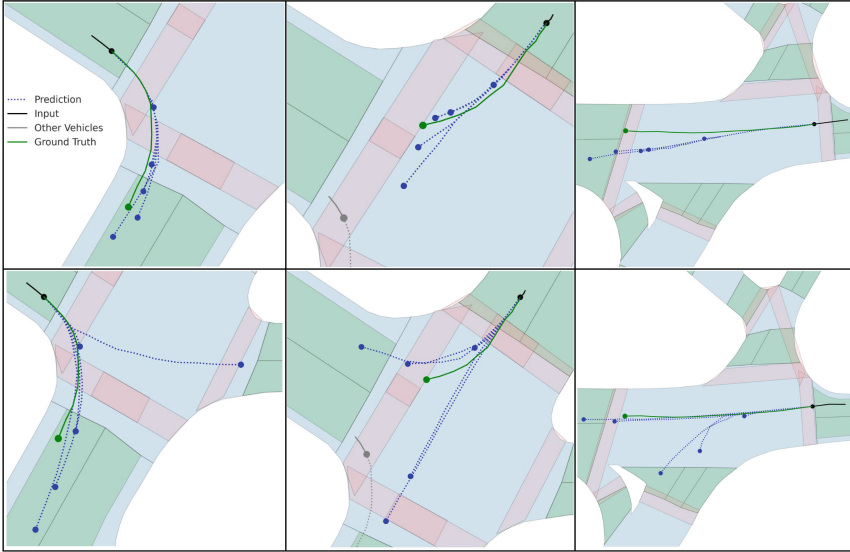


Fig. 3. First row shows the predicted trajectories by the network without mode queries. The second row shows the corresponding scenarios after the mode queries are incorporated into the network. The generalization capability of the network improves with mode queries as the network can predict the trajectories that can follow multiple scene-consistent paths.

the network focus its attention at a particular location in the image. We exploit this property of deformable cross-attention and set the reference point to the ego vehicle position based on the recurrent predicted trajectory output.

The recurrent behavior in the decoder is achieved by incorporating a feedback loop into the deformable cross-attention module. It outputs queries corresponding to individual modes, which are then transformed into multi-modal trajectories using Multi-Layer Perceptron (MLP). To capture the temporal correlation in the predicted trajectories, the temporal queries are updated to output queries of the previous iteration. In addition, the reference point is updated to the end point of the predicted trajectories from the previous iteration.

The working mechanism of the deformable cross-attention module is shown in Fig. 4. It consists of multiple iterations of deformable cross-attention layers between queries and fused scene encodings. The mode queries are added to the temporal queries before every deformable cross-attention layer.

3.3 Loss Formulation

We use the loss function proposed by HiVT [36]. It encourages diversity in predicted trajectories by optimizing only the best mode. The selection of the best mode is done based on the minimum l_2 between the ground truth and the pre-

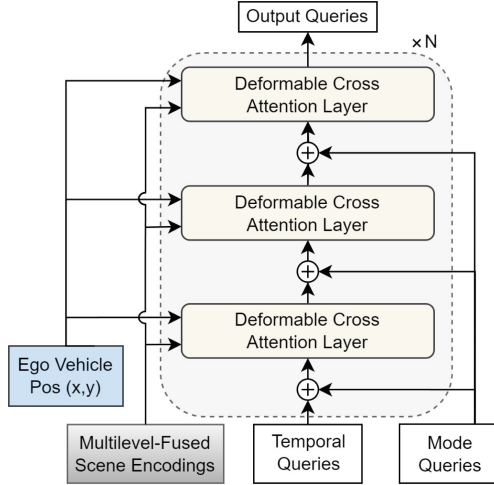


Fig. 4. An illustration of the proposed deformable cross-attention module. The offsets in the deformable cross-attention layer are computed with the linear transformation of the queries (as is done in original deformable attention [37]). These queries are generated by summing up temporal queries and mode queries. Values are then sampled from the multi-scale fused scene encodings at these offset locations and a weighted sum of the sampled values is computed. This process is repeated N times to produce the output queries.

dicted trajectories, averaged over all time steps. The loss function comprises of a regression loss \mathcal{L}_{reg} and a classification loss \mathcal{L}_{cls} :

$$\mathcal{L} = \mathcal{L}_{reg} + \mathcal{L}_{cls}, \quad (4)$$

Regression loss optimizes negative log-likelihood with the probability density function of the Laplace distribution, $\mathbb{L}(\cdot | \cdot)$, as follows:

$$\mathcal{L}_{reg} = -\frac{1}{T_o} \sum_{t=1}^{T_o} \log[\mathbb{L}(P_t | \mu_t, b_t)], \quad (5)$$

where μ_t , and b_t are the position and uncertainty at each time step of the predicted best mode trajectory respectively, and P_t are the ground truth trajectory positions. The classification loss aims to optimize only the mode probabilities $\pi(k)$ corresponding to mode k using the cross-entropy loss:

$$\mathcal{L}_{cls} = -\frac{1}{M} \sum_{k=1}^M \log(\pi(k)) \mathbb{L}(P_{T_o,k} | \mu_{T_o,k}, b_{T_o,k}), \quad (6)$$

4 Experiments

This section focuses on the experiments conducted using CASPFormer. Section 4.1 illustrates the dataset, metrics, and other experimental setting.

Section 4.2 provides a detailed comparison with the current state-of-the-art. Section 4.3 explains the design context of the ablation studies and the corresponding results.

4.1 Experimental Setup

Dataset. We test CASPFormer on the publicly available nuScenes dataset [1], which contains 1000 twenty-second-long traffic scenes from Boston and Singapore. The dataset consists of various traffic situations.

Metrics. We report the performance of CASPFormer using minADE_k , MR_k , minFDE_k , and OffRoadRate . minADE_k computes the average of pointwise l_2 distance in meters between the ground truth and the predicted modes and then chooses the minimum value across all k modes. minFDE_k computes the l_2 distance between the ground truth and predicted modes for the last time step only, and then selects the minimum amongst all k modes. MR_k is defined as the fraction of misses, where a miss occurs if the maximum pointwise l_2 distance between the ground truth and the predicted modes is more than two meters. OffRoadRate measures the fraction of predicted trajectories that lie outside the driving area.

Implementation Details. CASPFormer is trained on an Nvidia A100 GPU with a batch size of 64 using AdamW optimizer [23]. The number of past time steps for dynamic context is set to $T_i = 3$, which is equivalent to 1 s of input trajectory as the sampling rate is 2 Hz. The number of future time steps for the output is set to $T_o = 12$, which is equivalent to 6 s of prediction. The number of modes is set to $M = 5$. These hyperparameters are adopted from nuScenes motion prediction challenge [25].

The static and dynamic contexts cover a region of size $152\text{ m} \times 96\text{ m}$ with a resolution of 1 m, leading to the input grid sizes of (152, 96). The ego vehicle is placed at (122, 48) pointing upward in this grid. We perform data augmentation on the rasterized inputs during training. The inputs are randomly rotated in between $[-60^\circ, 60^\circ]$, and randomly translated in between $[-3, 3]$ with a probability of 0.75. The value of repetitions of deformable attention layers N , as depicted in Fig. 4, is set to four. The number of feature levels in the feature pyramid is also set to four, and the hidden dimension of all feature maps is set to 64. The network hyperparameters have been tuned to achieve the best performance of the network on nuScenes motion prediction challenge [25].

4.2 Results

We compare our work against the state-of-the-art on the nuScenes Motion Prediction Challenge [25] in Table 1. CASPFormer achieves the best performance in minADE_5 , MR_5 , and OffRoadRate . It should be noted that we have not included the work by Yao et al. [34] in our comparison, as their model Goal-LBP performs significantly worse on minFDE_1 (9.20) and OffRoadRate (0.07) in comparison to

Table 1. Comparison with state-of-the-art on the nuScenes prediction test split.

Method	minADE ₅ ↓	MR ₅ ↓	minFDE ₁ ↓	OffRoadRate↓
GOHOME [15]	1.42	0.57	6.99	0.04
Autobot [17]	1.37	0.62	8.19	0.02
THOMAS [16]	1.33	0.55	6.71	0.03
PGP [10]	1.27	0.52	7.17	0.03
MacFormer [12]	1.21	0.57	7.50	0.02
LAFormer [21]	1.19	0.48	6.95	0.02
FRM [27]	1.18	0.48	6.59	0.02
Q-EANet_v2 [6]	1.18	0.48	6.77	0.03
CASPNet++ [30]	1.16	0.50	6.18	0.01
CASPFormer (ours)	1.15	0.48	6.70	0.01

all other methods mentioned in Table 1. Moreover, this study is published after the conclusion of our work and therefore its methods could not have been verified and considered in our approach. Furthermore, we would also like to point out that the primary goal of CASPFormer is to remove the post-processing pipeline in CASPNet++ [30] and directly generate the vectorized trajectories for downstream planning task. However, it can be noticed that minFDE₁ in CASPFormer worsens in comparison to CASPNet++ and we hypothesised that this phenomenon happens due to a change in the direction of trajectory generation. CASPNet++ first generates the goal position of the agent and then generates a trajectory to connect the goal position with the current position of that agent. In contrast CASPFormer generates the trajectory from the current position towards the goal position of the agent, such a setup would lead to an accumulation of errors from previous time steps in estimating the goal position, and thus a comparatively worse minFDE₁.

Our qualitative results are illustrated in Fig. 5, which shows that CASPFormer can predict multiple modes consistent with the scene. In addition, we discover that each mode corresponds to a different driving speed of the ego vehicle. A potential limitation is that in some cases the trajectories are not well aligned with the lanes and we aim to tackle this in our future work.

4.3 Ablation Studies

We perform ablation studies on the nuScenes prediction validation split. The results of our ablation study are shown in Table 2. Where experiment #1 represents the baseline network architecture, which includes all modules, as presented in Fig. 2. In the following, we discuss the experimental setting of all the ablation studies and their results:

Importance of Mode Queries. To show the significance of mode queries, we conduct an experiment, in which the mode queries are not provided as input

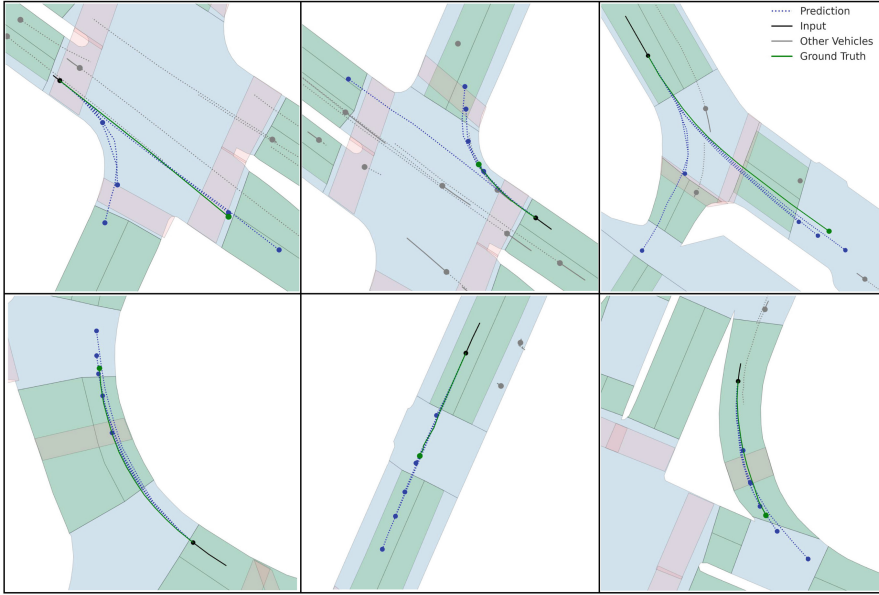


Fig. 5. Qualitative results on nuScenes prediction validation split. The blue region represents the drivable area and the green overlays portray the driving lanes. The pedestrian crossing is shown in red color. The current position of the ego vehicle is indicated with the black dot at the end of the input trajectory. The network can predict multiple scene-consistent trajectories in diverse scenarios such as intersections and crossings. (Color figure online)

to deformable cross-attention module, as presented in Fig. 2. The results of this experiment illustrate that the network performs worse on all metrics especially on minADE_5 when the mode queries are not provided in comparison to when they are (see experiments #1 and #2 in Table 2). The corresponding qualitative results of the experiment #2 are illustrated in Fig. 3, which indicate that even though the modes retain the property of capturing various speeds of the ego vehicle, they follow the same path and miss out on other possible paths, thus leading to mode collapse. Therefore, we deduce that the introduction of mode queries helps avoid mode collapse in CASPFormer.

Effect of Deformable Self-Attention. The studies [18, 28] point out that a significant computational cost in deformable attention comes from its deformable self-attention module. In our experiments, we also discover that if the deformable self-attention module is removed, the training time reduces by 60.3%, while minADE_5 , MR_5 and minFDE_1 increase by 11.5%, 15.2% and 7.6% respectively (see experiments #1 and #3 in Table 2). This can be a reasonable trade-off depending on the constraints for the motion prediction module. When removing the deformable self-attention module, we sum up the positional embeddings and

Table 2. Ablation Study on nuScenes Prediction Validation Split

#	Mode Embeddings	Deformable Self Attention	Recurrent Architecture	Ego Vehicle Position	minADE ₅ ↓	MR ₅ ↓	minFDE ₁ ↓
1.	✓	✓	✓	✓	1.13	0.46	6.43
2.	-	✓	✓	✓	1.72	0.60	6.60
3.	✓	-	✓	✓	1.26	0.53	6.92
4.	✓	✓	-	✓	1.21	0.48	6.63
5.	✓	✓	✓	-	1.15	0.48	6.51

scene encodings along the channel dimension and provide it directly as input into the deformable cross-attention module.

Importance of Recurrent Architecture. We also test whether the recurrent feedback loops help the network in performing better across the various metrics. Thus we remove both feedback loops from our baseline network (as shown in Fig. 2) and decode the complete 6s trajectories in a single forward pass. The results of this experiment show that the performance of the network decreases across all the metrics when the feedback loops are not present in the network (see experiments #1 and #4 in Table 2). This confirms the findings of the works [29, 35] that the recurrent architecture improves multimodal trajectory prediction.

Importance of Providing Ego Vehicle Position. The results of our experiments show that setting the reference point to the ego vehicle position does not improve the network performance by any significant degree (see experiments #1 and #5 in Table 2), where in the experiment #5, the reference points are directly learned via linear transformation of mode embeddings as is the case with the original deformable attention [37]. Nevertheless, we speculate that setting the reference point to the position of the agent in the scene can play an important role in multi-agent joint motion prediction, and leave a detailed study of this for future work.

5 Conclusion

In this study, a novel network architecture, CASPFormer, is proposed which performs multi-modal trajectory prediction from BEV images of the surrounding scene. CASPFormer employs a deformable attention mechanism to decode trajectories recurrently. Moreover, our work illustrates a mechanism to incorporate mode queries, which prevents the mode collapse and enables the network to generate scene-consistent multi-modal trajectories. We also identify that excluding the deformable self-attention module leads to a significant decrease in computational cost, without much effect on the network performance. Thus, in our future work, we aim to remove or modify the deformable self-attention module. Moreover, our future work would involve further study of the effect of vectorized

dynamic context and the impact of reference points in multi-agent joint motion prediction.

References

1. 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)
2. 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)
3. Casas, S., Gulino, C., Suo, S., Luo, K., Liao, R., Urtasun, R.: Implicit latent variable model for scene-consistent motion forecasting. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIII 16, pp. 624–641. Springer (2020)
4. Casas, S., Sadat, A., Urtasun, R.: Mp3: a unified model to map, perceive, predict and plan. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 14403–14412 (2021)
5. Chai, Y., Sapp, B., Bansal, M., Anguelov, D.: Multipath: Multiple probabilistic anchor trajectory hypotheses for behavior prediction. CoRR [abs/1910.05449](https://arxiv.org/abs/1910.05449) (2019). <http://arxiv.org/abs/1910.05449>
6. Chen, J., Wang, Z., Wang, J., Cai, B.: Q-eonet: implicit social modeling for trajectory prediction via experience-anchored queries. IET Intell. Transp. Syst. (2023)
7. Chen, L., Wu, P., Chitta, K., Jaeger, B., Geiger, A., Li, H.: End-to-end autonomous driving: challenges and frontiers. arXiv preprint [arXiv:2306.16927](https://arxiv.org/abs/2306.16927) (2023)
8. Cui, A., Casas, S., Sadat, A., Liao, R., Urtasun, R.: Lookout: diverse multi-future prediction and planning for self-driving. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 16107–16116 (2021)
9. Cui, H., et al.: Multimodal trajectory predictions for autonomous driving using deep convolutional networks. In: 2019 International Conference on Robotics and Automation (ICRA), pp. 2090–2096. IEEE (2019)
10. Deo, N., Wolff, E., Beijbom, O.: Multimodal trajectory prediction conditioned on lane-graph traversals. In: 5th Annual Conference on Robot Learning (2021)
11. Dosovitskiy, A., et al.: An image is worth 16x16 words: transformers for image recognition at scale. In: International Conference on Learning Representations (2021). <https://openreview.net/forum?id=YicbFdNTTy>
12. Feng, C., et al.: Macformer: map-agent coupled transformer for real-time and robust trajectory prediction. IEEE Robot. Autom. Lett. (2023)
13. Gao, J., et al.: Vectornet: encoding HD maps and agent dynamics from vectorized representation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11525–11533 (2020)
14. Gilles, T., Sabatini, S., Tsishkou, D., Stanciulescu, B., Moutarde, F.: Home: heatmap output for future motion estimation. In: 2021 IEEE International Intelligent Transportation Systems Conference (ITSC), pp. 500–507. IEEE (2021)
15. Gilles, T., Sabatini, S., Tsishkou, D., Stanciulescu, B., Moutarde, F.: Gohome: graph-oriented heatmap output for future motion estimation. In: 2022 International Conference on Robotics and Automation (ICRA), pp. 9107–9114. IEEE (2022)

16. Gilles, T., Sabatini, S., Tsishkou, D., Stanciulescu, B., Moutarde, F.: THOMAS: trajectory heatmap output with learned multi-agent sampling. In: International Conference on Learning Representations (2022). <https://openreview.net/forum?id=QDdJhACyrlX>
17. Girgis, R., et al.: Latent variable nested set transformers & autobots. CoRR **abs/2104.00563** (2021). <https://arxiv.org/abs/2104.00563>
18. Li, F., et al.: Lite detr: an interleaved multi-scale encoder for efficient detr. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 18558–18567 (2023)
19. Li, Z., et al.: Bevformer: Learning bird’s-eye-view representation from multi-camera images via spatiotemporal transformers. In: European Conference on Computer Vision, pp. 1–18. Springer (2022)
20. Liang, M., et al.: Learning lane graph representations for motion forecasting. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16, pp. 541–556. Springer (2020)
21. Liu, M., et al.: Laformer: trajectory prediction for autonomous driving with lane-aware scene constraints. arXiv preprint [arXiv:2302.13933](https://arxiv.org/abs/2302.13933) (2023)
22. 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)
23. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: International Conference on Learning Representations (2019). <https://openreview.net/forum?id=Bkg6RiCqY7>
24. Luo, W., Park, C., Cornman, A., Sapp, B., Anguelov, D.: Jfp: joint future prediction with interactive multi-agent modeling for autonomous driving. In: Conference on Robot Learning, pp. 1457–1467. PMLR (2023)
25. Motional: nuscenets prediction challenge. <https://eval.ai/web/challenges/challenge-page/591/leaderboard/1659>. Accessed 14 Feb 2024
26. Ngiam, J., et al.: Scene transformer: a unified architecture for predicting future trajectories of multiple agents. In: International Conference on Learning Representations (2021)
27. Park, D., Ryu, H., Yang, Y., Cho, J., Kim, J., Yoon, K.J.: Leveraging future relationship reasoning for vehicle trajectory prediction. In: The Eleventh International Conference on Learning Representations (2023). <https://openreview.net/forum?id=CGBCTp2M6lA>
28. Roh, B., Shin, J., Shin, W., Kim, S.: Sparse DETR: efficient end-to-end object detection with learnable sparsity. CoRR **abs/2111.14330** (2021). <https://arxiv.org/abs/2111.14330>
29. Schäfer, M., Zhao, K., Bühren, M., Kummert, A.: Context-aware scene prediction network (caspsnet). In: 2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC), pp. 3970–3977. IEEE (2022)
30. Schäfer, M., Zhao, K., Kummert, A.: Caspsnet++: joint multi-agent motion prediction. arXiv preprint [arXiv:2308.07751](https://arxiv.org/abs/2308.07751) (2023)
31. Varadarajan, B., et al.: Multipath++: efficient information fusion and trajectory aggregation for behavior prediction. In: 2022 International Conference on Robotics and Automation (ICRA), pp. 7814–7821. IEEE (2022)
32. Vaswani, A., et al.: Attention is all you need. In: Advances in Neural Information Processing Systems, vol. 30 (2017)
33. Xia, Z., Pan, X., Song, S., Li, L.E., Huang, G.: Vision transformer with deformable attention. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4794–4803 (2022)

34. Yao, Z., Li, X., Lang, B., Chuah, M.C.: Goal-lbp: goal-based local behavior guided trajectory prediction for autonomous driving. *IEEE Trans. Intell. Transp. Syst.* (2023)
35. Zhou, Z., Wang, J., Li, Y.H., Huang, Y.K.: Query-centric trajectory prediction. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17863–17873 (2023)
36. Zhou, Z., Ye, L., Wang, J., Wu, K., Lu, K.: Hivt: hierarchical vector transformer for multi-agent motion prediction. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8823–8833 (2022)
37. Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J.: Deformable detr: deformable transformers for end-to-end object detection. In: *International Conference on Learning Representations* (2021). <https://openreview.net/forum?id=gZ9hCDWe6ke>



LF Tracy: A Unified Single-Pipeline Paradigm for Salient Object Detection in Light Field Cameras

Fei Teng¹, Jiaming Zhang², Jiawei Liu¹, Kunyu Peng², Xina Cheng³, Zhiyong Li¹, and Kailun Yang¹(✉)

¹ Hunan University, Changsha, China
kailun.yang@hnu.edu.cn

² Karlsruhe Institute of Technology, Karlsruhe, Germany

³ Xidian University, Xi'an, China

Abstract. Leveraging rich information is crucial for dense prediction tasks. Light field (LF) cameras are instrumental in this regard, as they allow data to be sampled from various perspectives. This capability provides valuable spatial, depth, and angular information, enhancing scene-parsing tasks. However, we have identified two overlooked issues for the LF salient object detection (SOD) task. (1): Previous approaches predominantly employ a customized two-stream design to discover the spatial and depth features within light field images. The network struggles to learn the implicit angular information between different images due to a lack of intra-network data connectivity. (2): Little research has been directed towards the data augmentation strategy for LF SOD. Research on inter-network data connectivity is scant. In this study, we propose an efficient paradigm (LF Tracy) to address those issues. This comprises a single-pipeline encoder paired with a highly efficient information aggregation (IA) module ($\sim 8M$ parameters) to establish an intra-network connection. Then, a simple yet effective data augmentation strategy called MixLD is designed to bridge the inter-network connections. Owing to this innovative paradigm, our model surpasses the existing state-of-the-art method through extensive experiments. Especially, LF Tracy demonstrates a 23% improvement over previous results on the latest large-scale PKU dataset. The source code is publicly available at: <https://github.com/FeiBryantkit/LF-Tracy>.

Keywords: Light field camera · Salient object detection · Neural network · Scene parsing

1 Introduction

The objective of SOD lies in mimicking human visual attention mechanisms to accurately identify the most conspicuous objects or regions in a variety of

F. Teng and J. Zhang—Equal contribution.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-78447-7_29.

visual contexts. In particular, SOD plays a dual role: it not only aids agents in discerning the most striking and important elements in visual scenarios but also plays a pivotal role in several downstream tasks, including object detection, segmentation, and other dense prediction tasks [2, 29].

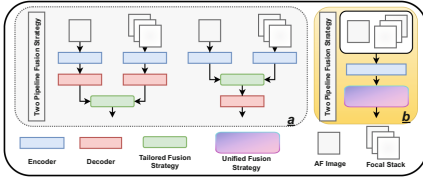


Fig. 1. Paradigms of LFSOD model. The conventional two-stream methods (a) and our single-pipeline method (b).

networks in more effectively learning scene features, LF cameras have been introduced [18]. LF camera is capable of capturing spatial, depth, and angular information. However, two significant challenges are neglected.

One: Lacking Intra-network Data Connectivity. The existing datasets for LF cameras consist of post-processed All-Focused (AF) images and Focal Stacks (FS) [17, 22, 38, 41]. AF images are full of texture information. FS images refer to images that include angular and depth information. The asymmetric data construction enriches the geometric information captured by LF cameras.

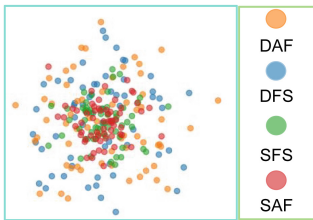


Fig. 2. Search space is visualized utilizing TSNE. “DAF” and “DFS” represent the feature maps of AF and FS in the dual pipeline method, while “SAF” and “SFS” represent the feature maps of AF and FS in the single pipeline method.

However, the implicit angular details cannot be directly utilized; they can only be obtained by exploring the latent relationships between images. While effectively utilizing depth and spatial information enhances the network’s ability to understand scenes, the current two-stream approach (Fig. 1(a)) neglects essential linkages among various images and disregards the angular information flow throughout the network, resulting in smaller searching space. As illustrated in Fig. 2, the high-dimensional data visualization (*i.e.*, TSNE) is conducted to demonstrate the search space of features. The search space of SFS (single-pipeline, focal stack) and SAF (single-pipeline, all-focused image) is significantly larger than that of the two-stream method.

Furthermore, in using a single-pipeline encoder, while different images can guide the network to learn angular features, merging the unfocused segments in AF image with all-focused data in FS images results in feature contamination within the feature space, significantly undermining the network’s discriminative capabilities. Hence, one of the key points of our work is “**how to leverage angular information while circumventing the alignment issues brought about by varying shooting viewpoints?**”.

Within the SOD community, the current 2D-based methods [6, 23] rely on the powerful feature extraction capabilities of Convolutional Neural Networks and Transformers, coupled with finely crafted decoders, to achieve impressive results. Meanwhile, a rich array of 3D methods [3, 27] have been introduced by utilizing depth or thermal information to boost the result. Given that information from various domains aids neural networks

Two: Lacking Inter-network Data Connectivity. Although researchers in the LF community enhance the understanding of scenes by introducing depth information (Focal Stack), existing works still adhere to the conventional RGB-D fusion structures [4, 5], employing common data augmentation (DA) strategies. Those methods isolatedly excavate the angular features and bury the relationship between different LF representations since there is no data interaction before the training process [33]. Therefore, another key point of our work is “**developing a novel DA strategy specifically for the LFSOD task to bridge a connection between various LF data sources before the training process**”. Figure 3 indicates a statistical result through the MixLD strategy. Before applying data augmentation, although a certain degree of data similarity between AF and FS can be observed from figures Fig. 3(a) and Fig. 3(b), there are still considerable differences in data within the range of [0, 100] pixels. However, after DA, by analyzing the distribution of the phase spectrum (Fig. 3(c)) and calculating its similarity with the central figure in the frequency domain (Fig. 3(f)), it can be seen that information has been aggregated.

In this work, we propose a novel paradigm (LF Tracy) to overcome the aforementioned challenges. Firstly, a single-pipeline framework in Fig. 1(b) is established to achieve the intra-network data connectivity. By learning different LF representations from a comprehensive perspective through a single backbone, our network can fully utilize the information from LF images rather than conducting separate feature extraction for LF representations. Furthermore, a simple yet IA model is performed within LF Tracy to effectively align and fuse the coupled features through the same backbone. Moreover, a simple data augmentation strategy called MixLD is introduced to establish inter-network data connectivity.

To demonstrate the efficiency of the proposed LF Tracy paradigm, comprehensive experiments are conducted on the large-scale PKU dataset [17], which comprises samples from both terrestrial and aquatic environments, and the LFSOD datasets [22, 38, 41]. By employing this paradigm (MixLD + Backbone + IA), our network achieved the state-of-the-art performance compared with previous works. Specifically, on the PKU dataset, our work achieved a 23% improvement in accuracy, fully validating the effectiveness of our network.

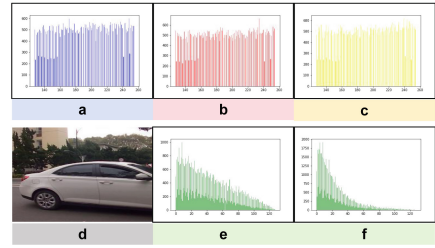


Fig. 3. A statistical result for the MixLD strategy. (d) indicates the AF image, and (a) and (b) illustrate the pixel distribution of the AF image and the FS image, respectively. (c) represents the central image after MixLD. (e) and (f) show the difference maps between the AF information and the original FS image, before and after MixLD. In (a), (b), and (c), the horizontal and vertical axes represent the pixel values and the number of pixels at those values, respectively. For (e) and (f), they represent the pixel value differences and the number of pixels at those difference values, respectively.

At a glance, we deliver the following contributions:

- We propose a single-stream SOD paradigm from scratch, bridging the inter-network and intra-network data connectivity.
- We have designed a low-parameter Information Aggregation (IA) Module that uncovers angular information while avoiding feature aliasing. Furthermore, we introduce a data augmentation strategy, namely MixLD, to establish inter-network data connectivity.
- An in-depth analysis is conducted to evaluate the performance of the single stream network under different hyper-parameters and module combinations.
- Our method achieves top performance on three LF datasets and one large-scale PKU dataset, which comprises over 10K images.

2 Related Work

Discovering and connecting the spatial, depth, and angular information of LF is essential for designing an efficient SOD neural network. Therefore, we will discuss the utilization of light field information from two aspects: **Intra-network Data Connectivity** in Sect. 2.1 and **Inter-network Data Connectivity** in Sect. 2.2. Lastly, preliminaries related to LF imaging are introduced in the appendix.

2.1 Intra-network Data Connectivity

The SOD task can be traced back to rule-based methodologies, which predominantly relied on visual attributes such as color, contrast, and spatial distribution to ascertain salient areas in images. In recent years, there has been a paradigm shift in the SOD community towards leveraging deep learning paradigms. Specifically, MENet [31] introduced iterative refinement and frequency decomposition mechanisms to improve detection accuracy. By utilizing transformer and multi-scale refinement architecture, Wang *et al.* [9] used high- and low-resolution images to achieve SOD. Furthermore, Zhang *et al.* [7] implemented SOD for panoramic images. Apart from those single-modality SOD networks, depth information is introduced to enhance performance, whereas multi-model fusion strategies [3, 8] are employed for RGB and thermal data.

For the SOD task of LF, Wang *et al.* [28] implemented a dual-pipeline neural network in the SOD community. Since then, the two-stream approach [21] for processing LF images has stood in a leading position in this field. Typically, this involves employing one backbone for processing AF images and another for FS images or the depth image extracted from LF sub-aperture images. Although the two-stream approach has seen considerable advancement in various tasks [36], it is typically applied to modalities that are isolated, such as depth and RGB images. For light field cameras, the depth, angular, and spatial information are embedded across different representations, *i.e.*, AF images and FS images. Processing these images in an isolated manner buries the angular features of light field cameras, and thus remains a sub-optimal method [33].

2.2 Inter-network Data Connectivity

Data augmentation (DA) has been thoroughly explored in various vision tasks such as image recognition, image classification, and semantic segmentation, proving effective in enhancing network performance and mitigating the issue of overfitting. The traditional data augmentation strategies can be roughly divided into five categories based on the adjusted purpose. 1) Flipping the image along its vertical and horizontal axis is a typical technique for increasing the diversity of data available for training. Furthermore, rotating an image at a certain angle is also a contributing factor. 2) Color jitter simulates images under different lighting and camera settings, enabling the trained model to better adapt to various scenarios. 3) Cutout [10] is introduced to drought or mismatch part of pixel-level information between neighboring pixels to increase the discrimination capability of the network. 4) Beyond deep learning, several works [11, 35] introduced machine learning-based strategies to boost the network capability. 5) Mixing-based methods [15, 33] leverage information from multiple images by generating blended input images.

Those methods demonstrate noticeable performance for the single image in the augmentation community. However, for light field cameras, the subtle angular information hidden within the interplay of multiple images cannot be captured through DA applied to individual images alone. Thus, establishing data connectivity across networks becomes crucial.

3 Methodology

This section introduces a comprehensive overview of our proposed paradigm, designed for the LFSOD task. Firstly, the framework’s architecture is meticulously expounded in Sect. 3.1. Additionally, in Sect. 3.2, we introduce a simple yet fusion module, which is pivotal for efficiently aggregating Light Field features. Last but not least, Sect. 3.3 delves into our innovative DA Strategy.

3.1 Proposed LF Tracy Framework

As shown in Fig. 4, the proposed network has two components: a four-stage encoder providing rich multi-dimensional information from different asymmetric data and the IA Module. The IA Module serves a dual purpose: 1) It overcomes the mismatching between the features established in-network connectivity through the same encoder block. 2) It can realign these features before sending them to the prediction head. The AFtention image I_{AF}^m and FSstention stack I_{FS}^m are described separately to provide a more intuitive description of the information flow and interaction process. The AFtention image and FSstention stack indicate the data source after MixLD. Furthermore, for simplicity, the following description is based on the stage one, which is the same for the other three stages. Especially, by applying the encoder block, the images are transferred into AF features ($F_{AF} \in \mathbb{R}^{(64 \times 64 \times 64)}$) and FS features ($F_{FS}^n \in \mathbb{R}^{(64 \times 64 \times 64)} | n \in [1, 12]$).

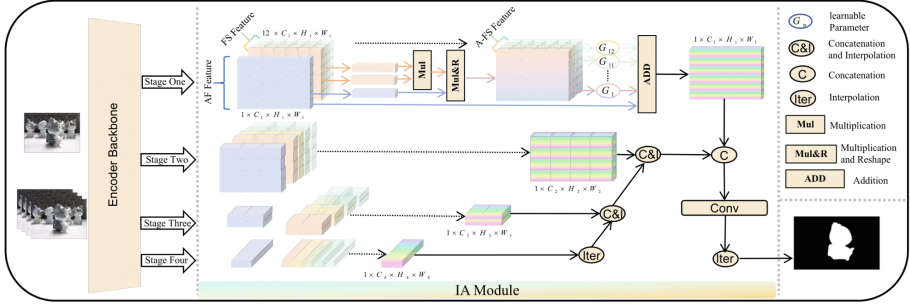


Fig. 4. Pipeline of LF Tracy network. FS and AS images are fed into the backbone for feature extraction. Multi-level features are then fed into the IA module (Sect. 3.3) for in-network data fusion and to predict the final result image.

After applying the IA Module, the 13 features are aggregated into one feature ($f_1 \in \mathbb{R}^{(64 \times 64 \times 64)}$), which contains all spatial, angular, and depth information. After four stages, there is a set of feature maps $\{f_l | l \in [1, 4]\}$ with channel dimension $\{64, 128, 320, 512\}$. Only f_1 is described in detail here, as the processes for the other dimensions are identical. Furthermore, at the training stage, to cooperate with the structure loss [12] calculation, f_l is also passed through one convolutional layer to compress channel information, as in Eq. (1).

$$f_{M_1} = \text{Conv}(64, 1)(f_l), \quad (1)$$

where $\text{Conv}(64, 1)(\cdot)$ indicates the convolutional layer with input channel 64 and output channel one. f_{M_1} denotes the feature after merging at the first stage. Furthermore, drawing upon the structure loss as outlined in [32], we have integrated the Tversky Loss [26] into our training process to improve supervision during training, specifically targeting a reduction in false positives and negatives.

3.2 Information Aggregation: IA Module

To fuse the implicit angular, explicit spatial, and depth information from asymmetric data, we introduce a simple IA Module that follows a two-step interaction process. Firstly, given single feature ($F_{FS}^n | n \in [1, 12]$), the FS-guided Query and Key are generated through their respective convolutional layer. Through matrix multiplication, the attention map ($M \in \mathbb{R}^{(4096 \times 4096)}$) is obtained. The attention map integrates a broader context into the aggregation of local features and enhances the representative capability of the focus part. Furthermore, applying the third convolutional layer to F_{AF} , the AF image guided Value ($V_{AF} \in \mathbb{R}^{4096 \times 64}$) is generated, as in Eq. (2)–(5). Given that the SOD task is sensitive to hyper-parameters and module design, we adopt a dimension reduction method for Query and Key. For more details on dimension reduction, please refer to Sect. 5.2.

$$Q = Conv_q(C_{in}, C_{out}^*)(F_{FS}^n), \quad (2)$$

$$K = Conv_k(C_{in}, C_{out}^*)(F_{FS}^n), \quad (3)$$

$$M = Soft\{Mul(Q, K)\}, \quad (4)$$

$$V = Conv_v(C_{in}, C_{out})(F_{AF}). \quad (5)$$

The tokens ($T \in \mathbb{R}^{4096 \times 64}$), which contains information from certain focal images, is obtained by multiplication of attention map and Value, as in Eq. (6).

$$T = Mul(M, V). \quad (6)$$

After obtaining the tokens, the A-FS features \hat{F}_{FS}^n are generated by applying the reshape operation. Note that the number of images has remained unchanged until now. This operation aims to enhance the spatial information at the corresponding depth by guiding the information from the FS image and, with the help of AF features, establish a connection between the global AF and FS information. Secondly, we introduce a set of learnable parameters to calculate the contribution of different FS features. To further enhance the spatial context information, a submission is undertaken, and the final result f_1 is obtained as in Eq. (7).

$$f_1 = AF + \sum_{i=1}^{12} \sigma \times \hat{F}_{FS}^n. \quad (7)$$

Given multi-scale features $\{f_1, f_2, f_3, f_4\}$, interpolation and concatenation are conducted. Finally, by applying the convolutional layer following an interpolation, the mask f is compressed and sent to the prediction head.

3.3 Data Augmentation Strategy: MixLD

As depicted in Fig. 5, the primary objective of the specific data augmentation strategy for the LFSOD task is to amalgamate distinct representations inherent in light field camera, namely, AF image (I_{AF}), FS ($I_{FS}^n | n \in [1, 12]$), and implicit angular information. This strategy is methodically partitioned into two discrete phases, each targeting specific aspects of the integration process. Initially, a non-intrusive approach is employed to integrate angular and depth information into the composite AF image while preserving the integrity of spatial data dimensions. Specifically, the data augmentation strategy can be described as following steps:

Firstly: (FS2AF). Following the FS setting [25], one FS slice I_{FS}^n with dimension $\{3 \times 256 \times 256\}$ is randomly selected with a likelihood of 0.1. This FS image is then subjected to a pixel-level fusion process, meticulously blending it into the AF image representation, as shown in Eq. (8).

$$I_{AF}^m = \{\alpha \times I_{AF} + (1 - \alpha)\{Rand(I_{FS}^n)\}\}, \quad (8)$$

where α denotes the degree of blending and n indicates the quantities of focal images. I_{AF}^m indicates the AF image after blending, *i.e.*, AFttention image. In MixLD, $\alpha = 1$ indicates no blending and $\alpha = 0$ indicates that the AF Image is completely replaced. Only the AF image is altered during this process, while the FS images remain unchanged. Meanwhile, this procedure is not conducted for each interaction.

Secondly: (AF2FS). The AFttention image is integrated into all the FS images with a probability of 0.5, as in Eq. (9).

$$I_{F_n}^m = \{\beta \times I_{AF}^m + (1 - \beta) \times I_{FS}^n\}, \quad (9)$$

where β denotes also a super parameter for the degree of blending in stage two and n denotes the quantities of focal images. $I_{F_n}^m$ indicates the FS after blending *i.e.*, FSttention stack. This integration carried out with a fusion probability of 0.5 instead of 0.1, aims to make it more possible to enrich the FS with additional information. By blending the AF image into the FS images, each focal image retains its inherent depth information, gains implicit angular insights from the other focal image, and enhances its spatial geometric information from the AF image. Furthermore, the AFttention image I_{AF}^m and FSttention stack ($I_{F_n}^m | n \in [1, 12]$) are fed into the network. It is important to emphasize that both phases (FS2AF and AF2FS) of MixLD are conducted randomly. It is possible for data interaction to occur in only one phase, while the other remains non-interactive.

It is precisely through this form of blending that the neural network while learning the inherent AF and FS information, can break out of the conventional framework to learn implicit angular information. For detailed algorithms, please refer to the pseudocode presented in the Appendix.

4 Experiments

To effectively demonstrate the efficacy of the approach, we showcase the quantitative result and qualitative results on different datasets. Firstly, we introduce the experimental setup in Sect. 4.1. Secondly, in Sect. 4.2, we present a quantitative comparison with other methods. Thirdly, in Sect. 4.3, we showcase the visual results of the method, along with a visual comparison with previous approaches.

4.1 Implementation Details

Datasets: The experiments are conducted following the benchmark proposed by the PKU team [17]. The datasets involve traditional LFSOD datasets, which include LFSOD [22], DUT-LF [41], HFUT [38] and a large-scale PKU dataset [17].

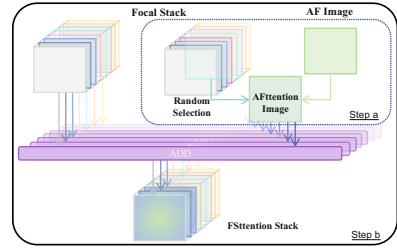


Fig. 5. Schematic illustration of our proposed MixLD strategy tailored for LFSOD. The strategy contains two independent steps (a and b), each of which is carried out randomly.

Table 1. Quantitative comparison with other methods in terms of MAE. The best result is highlighted in red. “Gain” indicates the improvement in our results compared to previous state-of-the-art methods. STSA₁, STSA₂, and STSA₃ represent the outcomes of the PKU Team [17] using different quantities of data for training. Although STSA₃ uses DUFT+HFUT+PKU-LF as a training set and outperforms other methods, our method still surpasses the STSA₃ network without expanding Training data. “Gain” denotes unavailable results.

Methods	Split NTU-60				Split NTU-120					STSA ₁	STSA ₂	STSA ₃	Ours Gain
	LFNet	SSF	D3Net	ATSA	UCNet	ESCNet	JLDCF	MEANet	GFRNet				
	[40]	[42]	[13]	[39]	[37]	[43]	[16]	[20]	[34]				
	TIP20	CVPR20	TNNLS21	ECCV21	TPAMI22	TIP22	TPAMI22	Neuc22	ICME23			[17]	
LFSD	.092	.067	.095	.068	.072	n.a.	.070	.077	.065	.067	.065	.062	.046 26%↑
HFUT	.096	.100	.091	.084	.090	.090	.075	.072	.072	.067	.072	.057	.056 2%↑
DUT-LF	.055	.050	.083	.041	.081	.061	.058	.031	.026	.033	.030	.027	.023 12%↑
PKU-LF	n.a.	.062	.067	.045	.070	n.a.	.049	n.a.	n.a.	.047	.042	.035	.027 23%↑

The images within the PKU dataset are sourced from terrestrial and aquatic environments. Two experiment strategies are conducted: **I**) training on DUT-LF + HFUT, ~1000 images, and evaluation on the whole LFSD dataset, the DUT-LF testing dataset, and HFUT testing dataset; **II**) training and testing on the PKU-LF dataset. PKU-LF dataset contains more than 10K images. For the ablation study, the experiments are based on experiment strategy one.

Setting Details: The image size for all the datasets is 256×256 . Each scene is structured to contain exactly 12 focal slices to meet specific coding requirements. This is achieved by strategically duplicating focal slices in the original order. Data augmentation is applied with Flipping, Cropping, Rotating, and MixLD for the training process. The blending parameter α, β are set into 0.5 and 0.5, respectively. The AdamW optimizer with a learning rate of $5e^{-5}$ and weight decay of $1e^{-4}$ is adapted for training. All the experiments are conducted on one A6000 GPU with a batch size of 6. The training epochs are limited to 300.

Evaluation Metrics: To analyze the results of different methods, we employ mean absolute error (MAE) [24] for a fair comparison. For F-measure (F_{β}^{mean}) [1], E-measure (S_{β}^{man}) [12], S-measure (S_{α}), we compare them with the previously best methods.

4.2 Quantitative Results

To verify the efficiency of the approach, we compare the designed network with existing methods. Table 1 shows that the best performance of the proposed approach significantly outperforms existing methods across the LFSD series dataset [22, 38, 41] and PKU dataset [17] on MAE. Due to the variability in performance across different evaluation metrics and datasets, we follow the benchmark provided by the PKU team [17].

The proposed method significantly surpasses this integrated benchmark. The network’s performance is most effectively proved, particularly with the large-scale and richly varied PKU dataset. By establishing the pre-network connectivity and the in-network connectivity of LF data, the network reconnects the intrinsic relationships between different light field camera images, achieving a 23% improvement in MAE compared with STSA₃. **It should be noted that the training dataset of STSA₃ is an extension dataset (DUT-LF + HFUT + PKU-LF). We used the PKU-LF dataset, and the network performance still exceeded by 23%.** In Table 2, we perform a comparison in terms of other evaluation criteria following PKU team [17]. While other networks may perform well in certain respects, LF Tracy still surpasses previous methods on a majority of metrics. This fully demonstrates the network’s superior comprehensive perception capabilities without being data-dependent.

4.3 Qualitative Results

It can be seen from Fig. 6 that the LF Tracy achieves outstanding accuracy across different scenarios by establishing intra-network and inter-network connectivity. Whether dealing with a single scene or complex scenarios, the network delivers excellent visualization results. Especially, for transparent backboards under varying lighting conditions, the network identifies the object through efficient information processing. Meanwhile, thin structures have always been a challenging issue in SOD tasks, yet the network has successfully identified both the necks of animals and the slender support poles of basketball hoops. Furthermore, the visual comparison results demonstrate the method’s superiority, as in Fig. 7. The proposed network accurately identifies the locations of objects, and notably, it precisely identifies challenging boundaries and lines. For the images in the middle row, the area with two pedestrians walking side by side is particularly challenging to discern. The varied colors and textures of their clothing present a significant challenge to the network. While other methods show numerous errors in this region, the proposed network achieves accurate identification.

Table 2. Quantitative comparison with other methods on different datasets in terms of F_{β}^{mean} , E_{β}^{mean} , and S_{α} . We conduct an unequal comparison by selecting the highest scores from previous works, *i.e.*, “PreV” and comparing them with our results.

Dataset	Metrics	PreV	Our	Gain
LFSD [22]	F_{β}^{mean}	.862 [4]	.896	3.9%↑
	E_{β}^{mean}	.902 [17]	.912	1.1%↑
	S_{α}	.864 [14]	.902	4.4%↑
HFUT [38]	F_{β}^{mean}	.771 [17]	.769	0.3%↓
	E_{β}^{mean}	.864 [17]	.865	0.1%↑
	S_{α}	.810 [17]	.833	0.1%↑
DUT-LF [41]	F_{β}^{mean}	.906 [17]	.936	3.3%↑
	E_{β}^{mean}	.954 [17]	.957	0.3%↑
	S_{α}	.911 [17]	.938	3.0%↑

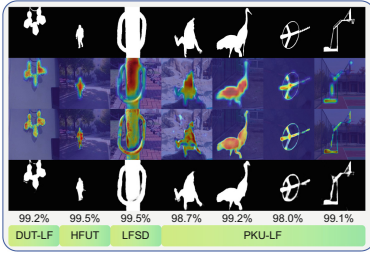


Fig. 6. Qualitative Result on four datasets. From top to bottom, the ground truth, AF image feature maps, decoder maps, and predicted masks are illustrated.



Fig. 7. Qualitative Comparison on three datasets. The difference maps between the visual results of various methods and the ground truth are displayed. Red pixels indicate pixels where the predicted results do not align with the ground truth. (Color figure online)

5 Ablation Studies

In this section, several ablation studies are conducted to showcase the process of designing the network from scratch. Firstly, in Sect. 5.1, the experiments are carried out to comprehensively examine the effects of various components incorporated in the methods. Section 5.2 showcases an in-depth analysis for the IA Module and FS Stack. Section 5.3 investigates the performance of different backbones for the SOD task. Section 5.4 demonstrated the in-depth analysis for MixLD.

5.1 Ablation Study for the Approach

In the experimental analysis, as shown in Table. 3, we ablated components of the approach to assess their contributions. The optimal performance achieved an MAE of 0.046. Firstly, eliminating the data augmentation strategy MixLD resulted in a performance decrease, and adapting CutMib [33] has few contributions to the performance. This indicates the necessity of MixLD to connect the different data before sending them into the network. After that, we ablate the core component of the network, the IA module, and the multi-scale features are directly fused. The MAE dramatically increased. The observed significant disparity of 0.286 highlights the effectiveness of the IA module. This module is integral for effectively realigning and managing the data imbalance across diverse sources. In particular, it is pivotal in reducing data mismatching between LF and AF images, facilitating more effective data integration, and improving accuracy with one stream encoder. Finally, without FS, the result is further reduced.

Table 3. The ablation study for LF Tracy.

Model	Our	w/o. MixLD	w. CutMib	w/o. IA	w/o. LF
MAE	.046	.052	.051	.332	.057

Parameters Analysis: The total Parameters of the designed LF Tray are $30M$. After removing the first stage in the IA module, the parameters decrease to $27M$. Furthermore, removing the entire IA module, the parameters fall into $24M$. With only $6M$ parameters, the network is capable of intra-network data connections and efficient feature fusion. **GFlops and FPS:** When processing 12 FS images, *i.e.*, handling a total of 13 light field images in a single training flow, the GFLOPs and FPS are 104.13 and 4.28, respectively. Without the IA module, these values are 84.8 GFLOPs and 4.73 FPS.

5.2 In-Depth Analysis for the IA Module and FS Stack

To demonstrate the contribution of the FS stack and the alignment and fusion capabilities of the IA module for asymmetric data, the ablation studies are conducted from three different aspects.

① **Focal Stack Images:** We compared the discrimination ability of the network with and without the IA module, using 2, 3, 5, and 12 FS images, respectively. As indicated in Table 4, without the IA module, continuously stacking FS images does not enhance the network’s capability; rather, it negatively impacts the network. With the addition of the IA module, the focused range and implicit angular information in the FS are utilized, increasing the network’s discrimination ability.

Table 4. An ablation study for the IA Module is conducted to evaluate its capabilities in terms of feature fusion and alignment.

Stack Size	2	3	5	12
w/o. IA	.137	.141	.205	.332
w. IA	.051	.051	.049	.046

② **Fusion strategy in IA module:** In Table 5, four different fusion strategies are compared. Firstly, we introduced an attention-based feature interaction process, accompanied by a set of learnable parameters, to achieve the fusion of information from different data sources. Then, we replaced this process with deformable cross attention [44]. Subsequently, we directly add the features point by point. Finally, we utilized cross-attention for feature interaction, directly adding the interacted feature maps. Although the point-by-point addition method has achieved significant results in semantic segmentation tasks, it does not work effectively for SOD tasks. Likewise, the method of deformable cross attention also did not surpass the method we proposed.

Table 5. The exploration of different fusion strategies: A&PD represents attention and dot product (with learnable parameters), DA represents deformable cross attention, ADD represents addition, and A&D represents attention and addition.

Strategy	A&PD	DA	ADD	A&D
MAE	.046	.056	.332	.048

Table 6. In the first step of the IA Module, the reduction rate for the dimensions of Query and Key is evaluated. We perform channel compression at different scales. The MAE and GFlops are reported. ‘R&Rate’ indicates Reduction Rate.

R&Rate	1	1/4	1/8	1/16
MAE	.050	.049	.046	.050

③ **Reduction Rate:** Last but not least, a set of experiments are conducted to deeply access the better hyper-parameters within IA module. Inspired by [19], the dimensions of the Query and Key are compressed in the IA module. Four different reduction rates are chosen. As shown in Table 6, over-reducing or under-reducing the channel can lead to performance degradation. The best option is to reduce the query and key dimensions to 1/8 of the original size.

5.3 Selection of Various Backbones

We conducted a series of experiments based on traditional datasets to assess the optimal feature extraction backbone. The PVTv2 [30] and the agent attention [19] are selected. To prevent pre-trained weights from causing an unfair comparison in the selection of backbones, we conducted experiments for 100 epochs without pre-trained weights. Table 7 shows that the agent attention is ineffective for the dataset, and the performance on the LFSOD dataset does not improve with the increase in the number of parameters. Due to this reason, we have chosen PVTv2 as the backbone.

Table 7. An ablation study for the selection of encoder backbone is conducted. B0, B1, B2, B4 indicate the backbone scales.

Backbone	B0	B1	B2	B4
PVTv2 [30]	.120	.097	.072	.087
AgentPVT [19]	.153	.137	.142	.145

5.4 In-Depth Analysis for MixLD

Interaction Probability Between Texture and Depth Information: In determining the optimal combination for incorporating depth information into AF images (FS2AF) and augmenting each FS image with texture information (AF2FS). Several experiments are designed with occurrence probabilities set at 0.1, 0.5, and 0.9. From the Table 8, it can be seen that: ① Assigning low occurrence probabilities (0.1) to both FS2AF and AF2FS minimally impacts the experimental outcomes, yet the performance metrics are analogous to those achieved with the CutMix augmentation technique. ② Excessive integration of

depth information into AF images (probability set at 0.9 for FS2AF) leads to a significant loss of spatial information, affecting the network’s performance. ③ While injecting spatial information into FS images improves the network’s ability to discriminate, excessive fusion can damage the valuable depth cues.

Table 8. Exploration for the occurrence probabilities of FS2AF and AF2FS.

Selection Rate		FS2AF		
		0.1	0.5	0.9
AF2FS	0.1	0.51	0.55	0.57
	0.5	0.46	0.49	0.54
	0.9	0.49	0.52	0.59

Table 9. Exploration of the blending rates in MixLD.

$\alpha = \beta$	0.1	0.3	0.5	0.7	0.9
MAE	.053	.049	.046	.073	.142

Blending Rate Analysis: To explore the optimal blending ratio of AF image and FS. We altered the parameter α in the first step, which involves blending one FS slice into AF images. Furthermore, in the second step, the parameter β is adjusted to merge the blended AF image into FS. Due to the various combinations of $\alpha - \beta$ pair, we only experimented with a few combinations based on $\alpha = \beta$. As shown in Table 9, the optimal outcome is achieved with a blending rate of 0.5. Notably, deviations from this ratio, either by increasing or decreasing the blending rate, result in a discernible decline in performance.

6 Conclusion

Contribution: In this paper, we present a unified single-stream method (LF Tracy) for salient object detection, bridging the inter-network and intra-network data connectivity. *First*, we have designed an efficient IA module. This module effectively addresses the feature mismatching of different LF representations. In combination with a single-pipeline encoder, it enables intra-network data connectivity. Uniquely, our study tests the network’s performance and achieves leading results on four distinct datasets. *Second*, we propose a data augmentation strategy for saliency object detection, specifically targeting inter-network connectivity. This method facilitates interaction among different channels of data, enhancing the network’s discriminative ability.

Limitation and Further Work: The task of salient object detection is sensitive to the choice of backbone, which sets it apart from other dense prediction tasks, such as semantic segmentation. Establishing a unified pixel-wise prediction framework is challenging and requires investigation in future work.

Acknowledgment. This work was supported in part by the National Natural Science Foundation of China (No. 62473139), in part by Helmholtz Association of German Research Centers, in part by the MWK through the Cooperative Graduate School Accessibility through AI-based Assistive Technology (KATE) under Grant BW6-03, and in part by Hangzhou SurImage Technology Co. Ltd.

References

1. Achanta, R., Hemami, S., Estrada, F., Susstrunk, S.: Frequency-tuned salient region detection. In: Proceedings of the CVPR (2009)
2. Borji, A., Itti, L.: State-of-the-art in visual attention modeling. *IEEE Trans. Pattern Anal. Mach. Intell.* (2013)
3. Chen, G., et al.: Modality-induced transfer-fusion network for RGB-D and RGB-T salient object detection. *IEEE Trans. Circuits Syst. Video Technol.* (2022)
4. Chen, G., et al.: Fusion-embedding Siamese network for light field salient object detection. *IEEE Trans. Multimed.* (2023)
5. Chen, Y., Li, G., An, P., Liu, Z., Huang, X., Wu, Q.: Light field salient object detection with sparse views via complementary and discriminative interaction network. *IEEE Trans. Circuits Syst. Video Technol.* (2023)
6. Chen, Z., Xu, Q., Cong, R., Huang, Q.: Global context-aware progressive aggregation network for salient object detection. In: Proceedings of the AAAI (2020)
7. Cong, R., Huang, K., Lei, J., Zhao, Y., Huang, Q., Kwong, S.: Multi-projection fusion and refinement network for salient object detection in 360° omnidirectional image. *IEEE Trans. Neural Netw. Learn. Syst.* (2023)
8. Cong, R., et al.: Does thermal really always matter for RGB-T salient object detection? *IEEE Trans. Multimed.* (2023)
9. Deng, X., Zhang, P., Liu, W., Lu, H.: Recurrent multi-scale transformer for high-resolution salient object detection. In: Proceedings of the MM (2023)
10. DeVries, T., Taylor, G.W.: Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552* (2017)
11. Ekin, B., Dandelion, V., Le, Q.V.: AutoAugment: learning augmentation policies from data. In: Proceedings of the CVPR (2019)
12. Fan, D.P., Cheng, M.M., Liu, Y., Li, T., Borji, A.: Structure-measure: a new way to evaluate foreground maps. In: Proceedings of the ICCV (2017)
13. Fan, D.P., Lin, Z., Zhang, Z., Zhu, M., Cheng, M.M.: Rethinking RGB-D salient object detection: models, data sets, and large-scale benchmarks. *IEEE Trans. Neural Netw. Learn. Syst.* (2021)
14. Fan, D.P., Zhai, Y., Borji, A., Yang, J., Shao, L.: BBS-Net: RGB-D salient object detection with a bifurcated backbone strategy network. In: Proceedings of the ECCV (2020)
15. Florea, C., Vertan, C., Florea, L.: SoftClusterMix: learning soft boundaries for empirical risk minimization. *Neural Comput. Appl.* (2023)
16. Fu, K., Fan, D.P., Ji, G.P., Zhao, Q., Shen, J., Zhu, C.: Siamese network for RGB-D salient object detection and beyond. *IEEE Trans. Pattern Anal. Mach. Intell.* (2022)
17. Gao, W., Fan, S., Li, G., Lin, W.: A thorough benchmark and a new model for light field saliency detection. *IEEE Trans. Pattern Anal. Mach. Intell.* (2023)
18. Georgiev, T., Intwala, C.: Light field camera design for integral view photography. Adobe System Inc., Technical Report (2006)

19. Han, D., Ye, T., Han, Y., Xia, Z., Song, S., Huang, G.: Agent attention: on the integration of softmax and linear attention. In: Proceedings of the ECCV (2024)
20. Jiang, Y., Zhang, W., Fu, K., Zhao, Q.: MEANet: multi-modal edge-aware network for light field salient object detection. *Neurocomputing* (2022)
21. Jing, D., Zhang, S., Cong, R., Lin, Y.: Occlusion-aware bi-directional guided network for light field salient object detection. In: Proceedings of the MM (2021)
22. Li, N., Ye, J., Ji, Y., Ling, H., Yu, J.: Saliency detection on light field. In: Proceedings of the CVPR (2014)
23. Pang, Y., Zhao, X., Zhang, L., Lu, H.: Multi-scale interactive network for salient object detection. In: Proceedings of the CVPR (2020)
24. Perazzi, F., Krähenbühl, P., Pritch, Y., Hornung, A.: Saliency filters: contrast based filtering for salient region detection. In: Proceedings of the CVPR (2012)
25. Piao, Y., Jiang, Y., Zhang, M., Wang, J., Lu, H.: PANet: patch-aware network for light field salient object detection. *IEEE Trans. Cybern.* (2023)
26. Salehi, S., Erdogmus, D., Gholipour, A.: Tversky loss function for image segmentation using 3D fully convolutional deep networks. In: Proceedings of the MLMI@MICCAI (2017)
27. Sun, F., Ren, P., Yin, B., Wang, F., Li, H.: CATNet: a cascaded and aggregated transformer network for RGB-D salient object detection. *IEEE Trans. Multimed.* (2023)
28. Wang, T., Piao, Y., Li, X., Lu, H.: Deep learning for light field saliency detection. In: Proceedings of the ICCV (2019)
29. Wang, W., Shen, J., Porikli, F.: Saliency-aware geodesic video object segmentation. In: Proceedings of the CVPR (2015)
30. Wang, W., et al.: PVT v2: improved baselines with pyramid vision transformer. *Comput. Vis. Media* (2022)
31. Wang, Y., Wang, R., Fan, X., Wang, T., He, X.: Pixels, regions, and objects: multiple enhancement for salient object detection. In: Proceedings of the CVPR (2023)
32. Wei, J., Wang, S., Huang, Q.: F³Net: fusion, feedback and focus for salient object detection. In: Proceedings of the AAAI (2020)
33. Xiao, Z., Liu, Y., Gao, R., Xiong, Z.: CutMIB: boosting light field super-resolution via multi-view image blending. In: Proceedings of the CVPR (2023)
34. Yuan, B., Jiang, Y., Fu, K., Zhao, Q.: Guided focal stack refinement network for light field salient object detection. In: Proceedings of the ICME (2023)
35. Zhang, C., Li, X., Zhang, Z., Cui, J., Yang, B.: BO-Aug: learning data augmentation policies via Bayesian optimization. *Appl. Intell.* (2023)
36. Zhang, J., et al.: Delivering arbitrary-modal semantic segmentation. In: Proceedings of the CVPR (2023)
37. Zhang, J., et al.: Uncertainty inspired RGB-D saliency detection. *IEEE Trans. Pattern Anal. Mach. Intell.* (2021)
38. Zhang, J., Wang, M., Lin, L., Yang, X., Gao, J., Rui, Y.: Saliency detection on light field: a multi-cue approach. *ACM Trans. Multimed. Comput. Commun. Appl. (TOMM)* (2017)
39. Zhang, M., Fei, S.X., Liu, J., Xu, S., Piao, Y., Lu, H.: Asymmetric two-stream architecture for accurate RGB-D saliency detection. In: Proceedings of the ECCV (2020)
40. Zhang, M., et al.: LFNet: light field fusion network for salient object detection. *IEEE Trans. Image Process.* (2020)
41. Zhang, M., Li, J., Wei, J., Piao, Y., Lu, H.: Memory-oriented decoder for light field salient object detection. In: Proceedings of the NeurIPS (2019)

42. Zhang, M., Ren, W., Piao, Y., Rong, Z., Lu, H.: Select, supplement and focus for RGB-D saliency detection. In: Proceedings of the CVPR (2020)
43. Zhang, M., Xu, S., Piao, Y., Lu, H.: Exploring spatial correlation for light field saliency detection: expansion from a single view. *IEEE Trans. Image Process.* (2022)
44. Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J.: Deformable DETR: deformable transformers for end-to-end object detection. In: Proceedings of the ICLR (2021)

Author Index

A

Amarbayasgalan, Tsatsral 156
Antonacopoulos, Apostolos 204
Audigier, Romaric 109

B

Babazaki, Yasunori 141
Balboni, Beatrice 269
Begarani, Filippo 269
Bolelli, Federico 187, 269

C

Chamiti, Tzoulis 371
Chang, Jun 362
Chen, Yibo 30
Chen, Yishuo 236
Cheng, Xina 435
Cheng, Zhengyun 405
Choi, Hong Seok 253
Choi, Jin Young 253

D

D'Alessandro, Tiziana 77
Dai, Shun 284
De Stefano, Claudio 77
Dong, Yan 45

F

Feng, Weng 362
Fontanella, Francesco 77
Fu, Sin-Yu 125

G

Gao, Ge 362
Gao, Guangshuai 45
Ghosal, Palash 220
Ghose, Shuvojit 173
Grana, Costantino 187, 269
Guo, Xinyu 236

H

Han, David K. 299
He, Jiasheng 236
Hosoi, Toshinori 93
Hu, Zhiyuan 30
Huang, Le 30
Huang, Yi-Chao 330
Huynh, Nguyen Truong Thinh 16
Hyypä, Juha 387
Hyyti, Heikki 387

J

Jeong, Chi Yoon 156
Jeong, Seong-Gyun 346

K

Kannala, Juho 387
Kim, Hayoung 346
Kim, Jinkyu 346
Kim, Mooseop 156
Kukko, Antero 387

L

Lee, Hwijun 253
Lee, Yoonji 253
Li, Chunlei 45
Li, Manyi 173
Li, Ming 362
Li, Yang 362
Li, Zhiyong 435
Liao, Powei 61
Lin, Huei-Yung 330
Lin, I.-Chen 125
Liu, Hongjuan 284
Liu, Jiaqi 284
Liu, Jiawei 435
Liu, Xiaobin 236
Lu, Jun 315
Lumetti, Luca 269

M

Maanpää, Jyri 387
 Maglo, Adrien 109
 Mai, Xuan Toan 16
 Manninen, Petri 387
 Marchesini, Kevin 269
 Marichal, Henry 1
 Meisen, Tobias 420
 Melekhov, Iaroslav 387
 Moon, Seokha 346

N

Nagase, Yasuto 141
 Nakano, Gaku 61
 Namiki, Shigeaki 93

O

Ogawa, Takuya 93

P

Pal, Umapada 204, 220
 Palaiahnakote, Shivakumara 204, 220
 Passalis, Nikolaos 371
 Passarella, Diego 1
 Patiño, Diego 299
 Peng, Kunyu 435
 Pesonen, Julius 387
 Pham, Van Linh 16
 Purkayastha, Kunal 220

Q

Qian, Yiming 173

R

Ramachandra, Raghavendra 204
 Randall, Gregory 1
 Rosati, Gabriele 269
 Rouhi, Amirreza 299
 Roy, Ayush 204

S

Santi, Stefano 187
 Sarkar, Shashwat 220
 Sartori, Federica 269
 Schaefer, Maximilian 420

Scotto di Freca, Alessandra 77

Shibata, Takashi 141

T

Tefas, Anastasios 371
 Teng, Fei 435
 Tian, Xuetao 284
 Tran, Tuan Anh 16

U

Um, Daeho 253

V

Vescovi, Luca 269
 Vezzali, Enrico 187

W

Wang, Boran 236
 Wang, Changhao 405
 Wang, Jinzhong 284
 Wang, Yang 173
 Wei, Minghong 45

X

Xiang, Haobin 45
 Xu, Yang 315

Y

Yachida, Shoji 93
 Yadav, Harsh 420
 Yang, Kailun 435
 Yeon, Kyuhwan 346
 Yokoyama, Keiko 93
 Yoon, Hajung 253
 Yuan, Jing 236

Z

Zeng, Haorui 284
 Zhang, Guanwen 405
 Zhang, Jiaming 435
 Zhang, Jianming 30
 Zhang, Xiuwei 284
 Zhang, Yanning 284
 Zhao, Kun 420
 Zhou, Wei 405
 Zhu, Wenbin 236
 Zhuo, Tao 284