Utah State University

DigitalCommons@USU

All Graduate Theses and Dissertations, Fall 2023 to Present

Graduate Studies

12-2023

Deep Learning With Effective Hierarchical Attention Mechanisms in Perception of Autonomous Vehicles

Qiuxiao Chen
Utah State University, anny.chen@usu.edu

Follow this and additional works at: https://digitalcommons.usu.edu/etd2023

Part of the Computer Sciences Commons

Recommended Citation

Chen, Qiuxiao, "Deep Learning With Effective Hierarchical Attention Mechanisms in Perception of Autonomous Vehicles" (2023). *All Graduate Theses and Dissertations, Fall 2023 to Present*. 58.
https://digitalcommons.usu.edu/etd2023/58

This Dissertation is brought to you for free and open access by the Graduate Studies at DigitalCommons@USU. It has been accepted for inclusion in All Graduate Theses and Dissertations, Fall 2023 to Present by an authorized administrator of DigitalCommons@USU. For more information, please contact digitalcommons@usu.edu.

UtahStateUniversity
MERRILL-CAZIER LIBRARY

DEEP LEARNING WITH EFFECTIVE HIERARCHICAL ATTENTION MECHANISMS IN PERCEPTION OF AUTONOMOUS VEHICLES

by

Qiuxiao Chen

A dissertation submitted in partial fulfillment
of the requirements for the degree

of

DOCTOR OF PHILOSOPHY

in

Computer Science

Approved:

Xiaojun Qi, Ph.D. Major Professor	Mahdi Nasrullah Al-Ameen, Ph.D. Committee Member
John Edwards, Ph.D. Committee Member	Shuhan Yuan, Ph.D. Committee Member
Jacob Gunther, Ph.D. Committee Member	D. Richard Cutler, Ph.D. Vice Provost of Graduate Studies

UTAH STATE UNIVERSITY
Logan, Utah

2023

Copyright © Qiuxiao Chen 2023

All Rights Reserved

ABSTRACT

Deep Learning with Effective Hierarchical Attention Mechanisms in Perception of Autonomous Vehicles

by

Qiuxiao Chen, Doctor of Philosophy

Utah State University, 2023

Major Professor: Xiaojun Qi, Ph.D.
Department: Computer Science

Sensing and interpreting information from the environment are essential for autonomous vehicles. To operate safely, autonomous vehicles must be able to effectively perceive a wide range of dynamic objects including other vehicles, pedestrians, and bicyclists, and static streets including pedestrian crossing, dividers, stop lines, and more.

In this dissertation, we aim to explore how to utilize the input information effectively in the 3D object detection task and map segmentation task with the help of hierarchical attention modules. Specifically, we introduce two networks utilizing hierarchical attention modules. One of these networks is for 3D object detection, and the other one is for map segmentation. Each of them utilizes the hierarchical attention modules and achieves comparable results with state-of-the-art methods on challenging benchmarks.

We name the 3D object detection network as Point Cloud Detection Network (PCDet), which utilizes LiDAR sensors to obtain the point cloud inputs with accurate depth information. To solve the problem of lacking multi-scale features and using the high-semantic features ineffectively, the proposed PCDet utilizes Hierarchical Double-branch Spatial Attention (HDSA)to capture high-semantic and fine-grained features at the same time. PCDet applies the Double-branch Spatial Attention (DSA) at the early stage and the late stage of

the network, which helps to use the high-semantic features at the beginning of the network and obtain the multi-scale features. However, HDSA does not consider global relational information. This limitation is solved by Hierarchical Residual Graph Convolutional Attention (HRGCA). PCDet applies the HRGCA module, which contains both graph and coordinate information, to not only effectively acquire the global information but also efficiently estimate contextual relationships of the global information in the 3D point cloud domain.

We name the map segmentation network as Multi-View Segmentation in Bird's-Eye-View (BEVSeg), which utilizes multiple camera sensors to obtain multi-view image inputs with dense semantic information. BEVSeg incorporates an Aligned BEV domain data Augmentation (ABA) module to augment the coherent BEV feature map and align its ground truths to address overfitting issues. It further incorporates the HDSA to effectively capture high-semantic and fine-grained features. It can also incorporate HRGCA to more accurately estimate global semantic relational features to address the ineffective high-semantic feature issues.

In general, the proposed HDSA is able to capture the high-level features and help utilize the high-level features effectively in both LiDAR-based 3D object detection and multiple camera-based map segmentation tasks, i.e. PCDet and BEVSeg. In addition, we propose a new effective HRGCA to further capture global relationships between different regions in the map segmentation task to improve the segmentation performance.

(78 pages)

PUBLIC ABSTRACT

Deep Learning with Effective Hierarchical Attention Mechanisms in Perception of Autonomous Vehicles

Qiuxiao Chen

Autonomous vehicles need to gather and understand information from their surroundings to drive safely. Just like how we look around and understand what's happening on the road, these vehicles need to see and make sense of dynamic objects like other cars, pedestrians, and cyclists, and static objects like crosswalks, road barriers, and stop lines.

In this dissertation, we aim to figure out better ways for computers to understand their surroundings in the 3D object detection task and map segmentation task. The 3D object detection task automatically spots objects in 3D (like cars or cyclists) and the map segmentation task automatically divides maps into different sections. To do this, we use attention modules to help the computer focus on important items. We create one network to find 3D objects such as cars on a highway, and one network to divide different parts of a map into different regions. Each of the networks utilizes the attention module and its hierarchical attention module to achieve comparable results with the best methods on challenging benchmarks.

We name the 3D object detection network as Point Cloud Detection Network (PCDet), which utilizes LiDAR sensors to obtain the point cloud inputs with accurate depth information. To solve the problem of lacking multi-scale features and using the high-semantic features ineffectively, the proposed PCDet utilizes Hierarchical Double-branch Spatial Attention (HDSA) to capture high-level and low-level features at the same time. PCDet applies the Double-branch Spatial Attention (DSA) at the early stage and the late stage of the network, which helps to use the high-level features at the beginning of the network and

obtain the multiple-scale features. However, HDSA does not consider global relational information. This limitation is solved by Hierarchical Residual Graph Convolutional Attention (HRGCA). PCDet applies the HRGCA module, which contains both graph and coordinate information, to not only effectively acquire the global information but also efficiently estimate contextual relationships of the global information in the 3D point cloud.

We name the map segmentation network as Multi-View Segmentation in Bird's-Eye-View (BEVSeg), which utilizes multiple cameras to obtain multi-view image inputs with plenty of colorful and textured information. The proposed BEVSeg aims to utilize high-level features effectively and solve the common overfitting problems in map segmentation tasks. Specifically, BEVSeg utilizes an Aligned BEV domain data Augmentation (ABA) module to flip, rotate, and scale the BEV feature map and repeat the same process on its ground truths to address overfitting issues. It further incorporates the hierarchical attention mechanisms, namely, HDSA and HRGCA, to effectively capture high-level and low-level features and to estimate global relationships between different regions in both the early stage and the late stage of the network, respectively.

In general, the proposed HDSA is able to capture the high-level features and help utilize the high-level features effectively in both LiDAR-based 3D object detection and multiple camera-based map segmentation tasks, i.e. PCDet and BEVSeg. In addition, we proposed a new effective HRGCA to further capture global relationships between different regions to improve both 3D object detection accuracy and map segmentation performance.

ACKNOWLEDGMENTS

I would like to begin by extending my profound gratitude to my advisor, Dr. Xiaojun Qi. Her unwavering support, expert guidance, and enduring patience have been instrumental throughout my Ph.D. journey. Her expertise has not only enriched my research but has also been pivotal in shaping my career.

I owe a debt of gratitude to the distinguished members of my dissertation committee: Dr. Jacob Gunther, Dr. John Edwards, Dr. Shuhan Yuan, Dr. Mahdi Nasrullah Al-Ameen, and Dr. Haitao Wang. Their invaluable feedback and continuous encouragement have been cornerstones of my doctoral journey.

On a more personal note, the support of my family has been immeasurable. My husband has been my anchor, offering unwavering support both in my academic pursuits and in our shared life journey. He has provided guidance, assistance, and invaluable support in my job search. My parents have always been the bedrock of my mental and emotional well-being, and I cannot thank them enough for their steadfast belief in me. I am also deeply thankful for the kindness and support of my parents-in-law, who have shown me nothing but love and encouragement.

To my friends who have helped me during my Ph.D. journey: Meng Xu, Kuan Huang, Dean Mathias, and Zengyan Zhang – your support have been indispensable. Additionally, I'd like to express my gratitude to those who made my time at Utah State University memorable and enjoyable: Qi Luo, Haixuan Guo, Peiyu Li, Mohammadreza Javanmardi, Jiyao Li, Xiankun Yan, Yiming Zhao, Fei Xu, and Amir Hossein Farzaneh. Your companionship and camaraderie have truly enriched my experience.

Qiuxiao Chen

CONTENTS

Page

ABSTRACT . . . iii

PUBLIC ABSTRACT . . . v

ACKNOWLEDGMENTS . . . vii

LIST OF TABLES . . . x

LIST OF FIGURES . . . xi

ACRONYMS . . . xiii

1 INTRODUCTION . . . 1
- 1.1 Sensors . . . 2
- 1.2 3D Object Detection . . . 4
- 1.3 Map Segmentation . . . 7
- 1.4 Attention . . . 7
- 1.5 Organization of Dissertation . . . 9

2 RELATED WORK . . . 11
- 2.1 Deep Learning . . . 11
- 2.2 Attention Mechanisms . . . 12
- 2.3 3D Object Detection . . . 12
  - 2.3.1 LiDAR-based 3D Object Detection . . . 13
  - 2.3.2 Multi-view 3D Object Detection . . . 14
- 2.4 Map Segmentation in BEV . . . 15
  - 2.4.1 Multi-view BEV Map Segmentation . . . 15
- 2.5 Proposed Method . . . 16

3 ATTENTION . . . 18
- 3.1 Double-branch Spatial Attention (DSA) and its Two Variants . . . 18
- 3.2 Design and Mathematical Formulation of DSA . . . 20
- 3.3 Residual Graph Convolutional Attention (RGCA) . . . 21
- 3.4 Hierarchical Attention Modules . . . 23

4 THE PROPOSED NETWORKS . . . 24
- 4.1 Proposed 3D Object Detection Network PCDet . . . 24
  - 4.1.1 PCDet Overall Structure . . . 25
- 4.2 Proposed Map Segmentation Network BEVSeg . . . 28
  - 4.2.1 Overall Architecture of BEVSeg . . . 30
  - 4.2.2 Basic Architectures . . . 31
  - 4.2.3 Proposed Modules in BEVSeg . . . 32

5 EXPERIMENTS 35
5.1 Datasets and Experimental Setup 35
5.2 Metrics 36
5.3 Results 37
5.3.1 3D Object Detection Results 37
5.3.2 Map Segmentation Results 46

6 CONCLUSIONS 56

REFERENCES 59

CURRICULUM VITAE 64

LIST OF TABLES

Table Page

5.1 Comparison of car detection results at three difficulty levels 37

5.2 Comparison of car detection results at three difficulty levels of different hierarchical attention-based small SECOND networks. 38

5.3 Comparison of AP(%) and FPS of the proposed PCDet_HDSA and PCDet_HRGCA (using a small SECOND network and using a large SECOND network as the backbone respectively) with AP(%) and FPS of ten peer one-stage voxel-based 3D object detectors on cars. 40

5.4 Comparison of AP (%) of the proposed PCDet_HDSA (using a small SECOND network and using a large SECOND network as the backbone) with AP(%) of three peer one-stage voxel-based 3D object detectors on cyclists. 42

5.5 Comparison of map segmentation results of different single attention-based networks. 46

5.6 Comparison of map segmentation results of different hierarchical attention-based networks. 47

5.7 Comparison of segmentation results of ten state-of-the-art methods for six classes on the nuScenes in terms of IoU. 49

LIST OF FIGURES

Figure Page

1.1 Illustrations of three common sensors utilized in 3D object detection and map segmentation: a) the raw outputs of multiple camera sensors (from the nuScenes dataset [1]) b) the raw output of the radar sensor (from NVIDIA) c) the raw output of the LiDAR sensor [2] 3

1.2 The block diagram of traditional feature-based object detection techniques . 4

1.3 The block diagram of deep learning-based 3D object detection techniques: (a) single-stage 3D object detection; (b) two-stage 3D object detection . . . 5

1.4 Illustrations of the LiDAR-based 3D object detection, multi-view 3D object detection, and multi-view map segmentation: a) 3D object detection results on LiDAR images; b) 3D object detection results on multiple images (adapted from BEVFormer [3]); c) Multi-view images and their BEV map segmentation results (adapted from BEVFusion [4]), where drivable area is shown in blue, land divider is shown in purple, walkway is shown in red, and crossing is shown in pink). 8

1.5 The block diagram of deep learning-based map segmentation 9

3.1 Structure of the Double-branch Spatial Attention. 19

3.2 Illustration of the Residual Graph Convolutional Attention 22

4.1 The overall architecture of the proposed PCDet. The upper part demonstrates its block diagram, where hierarchical attention modules reside in the blue-shaded block titled "2D Convolution with Attention". The lower part presents the flowchart of employing the hierarchical attention modules in the 2D backbone network, where Layer 1 represents the results from the attention module at the early stage and Layer 2 represents the results from the attention module at the late stage. The DSA and RGCA attention can be interchangeably applied at either the early stage or the late stage to build a different 3D object detection framework. 26

4.2 Illustration of BEVSeg's overall architecture: 1) image encoder extracts multi-camera features; 2) Image-to-BEV view transform converts multi-camera features to the BEV domain; 3) ABA shown in the light blue shaded block is the proposed geometry module that augments and aligns BEV features and ground truths; 4) hierarchical attention modules & BEV encoder shown in the dark blue shaded block is the proposed data-driven module that encodes BEV feature to obtain high-semantic information from different scales; 5) BEV SH shown in the light blue shaded block is proposed to replace the detection head in BEVDet and predict BEV binary maps for six categories. 30

4.3 Illustration of two categories in BEV map, where overlap regions are shown in red. Left to right: non-overlap between drivable area (brown) and walkway (purple); non-overlap between drivable area (brown) and ped crossing (purple); non-overlap between ped crossing (brown) and stop line (purple). 34

5.1 Three sample car detection results of baseline and six proposed networks (from top to bottom): SECOND, PCDet_DSA variant 1, PCDet_DSA variant 2, PCDet_DSA, PCDet_RGCA, PCDet_HDSA and PCDet_HRGCA. 43

5.2 Three sample cyclist detection results of baseline and six proposed networks (from top to bottom): SECOND, PCDet_DSA variant 1, PCDet_DSA variant 2, PCDet_DSA, PCDet_RGCA, PCDet_HDSA, and PCDet_HRGCA. 44

5.3 Illustration of four sample scenes in the daytime along with their ground truth and predicted segmentation results for six categories. For each scene, the first two rows present multi-view input images, the third row presents the ground truth for six categories, and the fourth to the eighth row respectively presents the predicted segmentation results of the baseline (variant A in Table 5.6), BEVSeg_DSA, BEVSeg_RGCA, BEVSeg_HDSA, and BEVSeg_HRGCA for six categories. 52

5.4 Illustration of four sample scenes at night along with their ground truth and predicted segmentation results for six categories. For each scene, the first two rows present multi-view input images, the third row presents the ground truth for six categories, and the fourth to the eighth row respectively presents the predicted segmentation results of the baseline (variant A in Table 5.6), BEVSeg_DSA, BEVSeg_RGCA, BEVSeg_HDSA, and BEVSeg_HRGCA for six categories. 53

5.5 Illustration of two BEV map ground truth (the first two rows) and one augmented BEV map ground truth generated by the ABA module (the last row) 54

5.6 Illustration of the ground truth of one scene and the segmentation results of baseline (BEVDet with SH) and variant B (baseline with ABA) shown from the top to the bottom. 54

5.7 Illustration of one sample scene of six views (first two rows), segmentation result of BEVDet with ABA and SH modules (third row), segmentation results of the proposed BEVSeg_HRGCA network (fourth row), and the ground-truth (fifth row). The inaccurate segmentation results are circled in red. . . 55

ACRONYMS

ABA	Aligned BEV domain data Augmentation
AI	Artificial Intelligence
AP	Average Precision
BEV	Bird's-Eye-View
BEVSeg	Multi-View Segmentation in Bird's-Eye-View
CNN	Convolutional Neural Networks
CVT	Cross-View Transformer
DNN	Deep Neural Network
DSA	Double-branch Spatial Attention
HDSA	Hierarchical Double-branch Spatial Attention
HOG	Histogram of Oriented Gradients
HRGCA	Hierarchical Residual Graph Convolutional Attention
IoU	Intersection-over-Union
LSS	Lift, Splat, Shoot
mAP	mean Average Precision
mIoU	mean Intersection-over-Union
ML	Machine Learning
NMS	Non-Maximum Suppression
OFT	Orthographic Feature Transform
PCDet	Point Cloud Detection Network
RGCA	Residual Graph Convolutional Attention
SE	Squeeze-and-Excitation
SGD	Stochastic Gradient Descent
SH	Segmentation Head
SIFT	Scale Invariant Feature Transform
SVM	Support Vector Machine
TA	Triple Attention

CHAPTER 1

INTRODUCTION

Autonomous vehicles, also known as self-driving cars or driverless cars, are vehicles that are capable of operating and navigating without human intervention. They have generated a lot of interest and excitement in recent years. The main reasons are their potential to significantly reduce accidents caused by human error, make transportation more convenient for people who cannot drive, and improve traffic flow. Driven by their immense potential, both academia and industry have shown a growing interest in the design and implementation of efficient and effective self-driving vehicle systems. Following the modern industrial principle (i.e., problem decomposition), academia and industry divide the self-driving vehicle system into four subsystems, including perception, prediction, planning, and control [5]. The perception subsystem senses and interprets information from the environment, which is essential for autonomous vehicles to operate safely and effectively. The prediction subsystem predicts the future states of the nearby vehicles based on current and past observations of the surrounding environment. The planning subsystem makes decisions about how to move through the world. The control subsystem controls maneuvers to track the planned trajectory autonomously.

In this dissertation, we focus on perception, which is one of the most important parts of autonomous vehicles. Specifically, autonomous vehicles must be able to perceive a wide range of dynamic objects (e.g., other vehicles, pedestrians, and cyclists) and static streets (e.g., drivable areas, pedestrian crossings, walkways, carpark areas, dividers, stop lines, and so on). To satisfy the requirements of autonomous vehicles, the perception utilizes a camera, radar, or LiDAR sensor to sense the surroundings and exploit 3D object detection, map segmentation, and object tracking techniques to interpret the environments. In this dissertation, we focus on designing a 3D objection technique to identify cars, pedestrians, and cyclists and designing a map segmentation technique to segment six crucial static

semantic classes.

1.1 Sensors

First, we introduce common sensors including cameras, radar, and LiDAR, utilized in 3D object detection and map segmentation to sense the surroundings. The raw outputs of the sensors are shown in Figure 1.1. Each sensor has its advantages and limitations. The choice of the sensor depends on the specific requirements of the application.

- Camera sensors use the lens to bring light to a fixed focal point and create a 2D high-resolution image of one view, which provides color and texture information of objects in a scene. To get the surrounding information, most people utilize multiple camera sensors with complete 360° coverage in autonomous vehicles. However, camera sensors have limitations in low-light conditions, and their depth perception capabilities are limited.

- Radar sensors use radio wave transmitters to measure the distance to objects and create a 3D representation of the environment, which operates well in all weather conditions and is resistant to interference from other sensors. However, radar has limitations in detecting small objects and cannot provide detailed information about the shape and texture of objects. In addition, the semantics of radar is so sparse that severely reduces the accuracy of semantic-oriented tasks.

- LiDAR sensors use laser beams to measure the distance to objects and create a 3D representation of the environment, which provides accurate depth information and detects objects in low-light conditions. Since one LiDAR sensor is able to get the surrounding information, most people utilize one LiDAR sensor in autonomous vehicles. However, LiDAR sensors are expensive and can be affected by weather conditions such as rain and fog. Furthermore, its sparse semantics lead to decreased accuracy on semantic-oriented tasks [4].

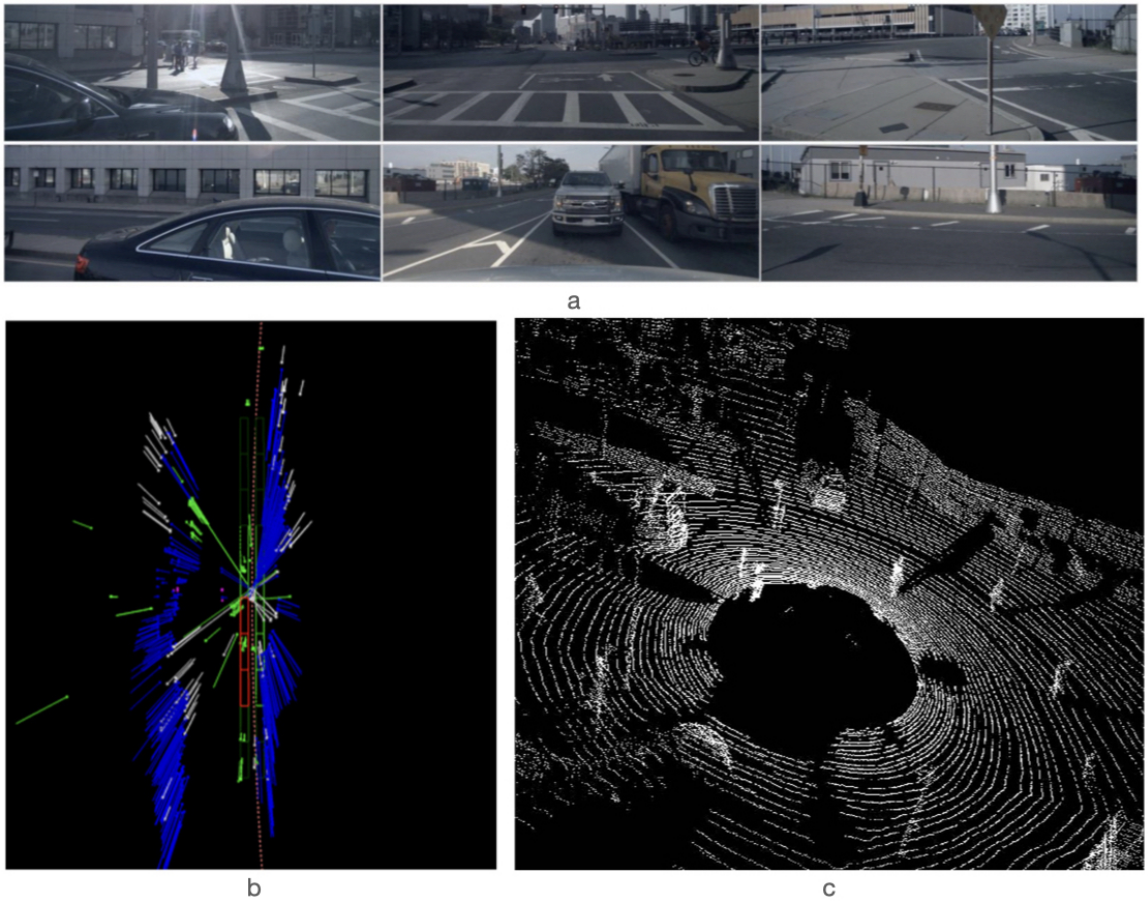

Fig. 1.1: Illustrations of three common sensors utilized in 3D object detection and map segmentation: a) the raw outputs of multiple camera sensors (from the nuScenes dataset [1]) b) the raw output of the radar sensor (from NVIDIA) c) the raw output of the LiDAR sensor [2]

1.2 3D Object Detection

3D object detection techniques help autonomous vehicles detect and recognize dynamic objects, such as other vehicles, pedestrians, and cyclists, in the 3D environment in terms of their shape (length, width, and height) and location (x, y, and z coordinates). 3D object detection is different from 2D object detection. 2D object detection focuses on identifying objects' 2D bounding boxes, while 3D object detection aims to estimate the objects' 3D positions and orientations in the physical world, which includes depth information. The development of 3D object detection has been a progressive journey, which could be roughly divided into two stages: traditional feature-based and deep learning-based 3D object detection.

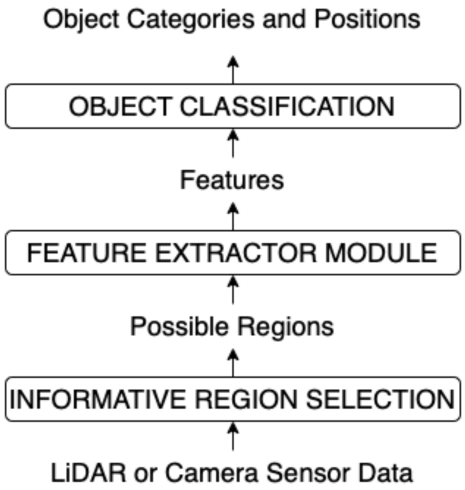

Fig. 1.2: The block diagram of traditional feature-based object detection techniques

Figure 1.2 demonstrates the block diagram of traditional feature-based object detection techniques, where informative region selection, feature extraction, and object classification are three major components. The first component, informative region selection, aims to find the locations of objects by graph-based segmentation [6], voxel-based clustering methods [7], and other methods. The second component, feature extraction, extracts the feature to get the important information by Scale Invariant Feature Transform (SIFT) [8], Histogram of

Oriented Gradients (HOG) [9], voxel's probabilistic features [7] or other traditional hand-engineered feature extraction methods. The last component, object classification, classifies the target objects from all possible categories, by Support Vector Machines (SVMs) [10], a mixture of bag-of-words classifiers [11], or other traditional algorithms.

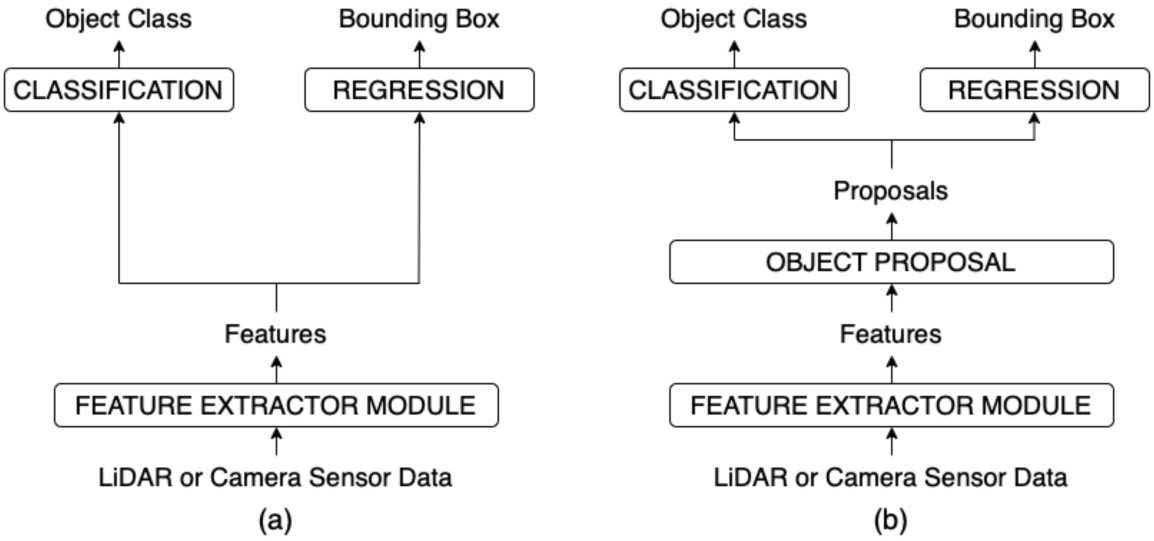

Fig. 1.3: The block diagram of deep learning-based 3D object detection techniques: (a) single-stage 3D object detection; (b) two-stage 3D object detection

Deep learning-based 3D object detection techniques can be classified into two categories: single-stage 3D object detection and two-stage 3D object detection. Figure 1.3 (a) demonstrates the block diagram of single-stage 3D object detection techniques, where feature extractor module, classification module, and regression module are three major components. The first component, deep learning feature extractor module, automatically extracts features from the input data by deep convolution feature extractor [12, 13], deep transformer extractor [14], or other methods. The second component, multiple layer classification, assigns a class label to each default box by different multiple layer classification heads [15]. The last component, regression, predicts the offsets (i.e., the difference in coordinates) between default bounding boxes and the ground-truth objects, allowing the detector to refine the locations of predicted bounding boxes by different multiple layer detection

heads [15].

Figure 1.3 (b) demonstrates the block diagram of two-stage 3D object detection techniques, where the feature extractor module, object proposal module, classification module, and regression module are four major components. The first component, the deep learning feature extractor module [16], is the same as the feature extractor of single-stage 3D object detection. The second component, object proposal module [17], generates a set of region proposals, which are potential bounding box locations likely to contain objects. The third component, multiple layer classification, assigns a class label to each region proposal by different multiple layer classification heads [18]. The last component, regression, predicts the offsets between region proposals and the ground-truth objects, by different multiple-layer detection heads [18].

Compared to the traditional models, the deep learning models' feature extractors have the ability to extract feature maps from input data automatically. This eliminates the requirement of manual feature extraction, allowing the models to capture more complex and abstract representations. In addition, the deep learning models enable end-to-end training, which eliminates the need for multiple steps and leads to improved integration and performance. The main difference is that the traditional pipeline optimizes every module individually, while the deep learning pipeline optimizes the whole network end-to-end. Therefore, in this dissertation, we focus on deep learning-based methods. Compared to the two-stage methods, the single-stage methods have a simpler and more straightforward architecture. Specifically, the single-stage methods predict object bounding boxes and class labels directly, while the two-stage methods need a separate region proposal stage. The simplicity of the single-stage methods leads to more efficient inference and faster speed, which makes single-stage methods more suitable for real-time applications, such as autonomous driving systems. Therefore, we focus on single-stage methods in this dissertation.

Nowadays, 3D object detection researchers tend to utilize the expensive LiDAR sensors [16,19] to gain accurate depth information, or utilize camera sensors [20,21] to decrease the cost of autonomous vehicles to obtain the color and texture information of objects in a

scene. Since radar cannot detect small objects nor provide detailed information about objects, researchers rarely use radar sensors these days. In the industry, there is still much discussion about which autonomous vehicle sensor—LiDAR or cameras—is more useful. Figure 1.4.a) presents one sample LiDAR-based 3D object detection result and Figure 1.4. b) presents one sample multi-view 3D object detection result.

1.3 Map Segmentation

Map segmentation helps the autonomous vehicles make a map of static streets, including drivable areas, dividers, walkways, carpark areas, stop lines, pedestrian crossings, and so on.

Different from 3D object detection, research in map segmentation is relatively new. The pioneering work of map segmentation [22] utilizes deep learning network structure, which lays the foundations of all Bird's-Eye-View (BEV) map segmentation methods to be deep learning-based. Figure 1.5 illustrates the block diagram of map segmentation techniques, where feature extractor and classification modules are the two major components. The first component, the feature extractor module, extracts high-level features from the input image through convolutional neural network architectures [23], transformer architectures [14], or other methods. The second component, pixel-wise classification, assigns a class label or generates a probability distribution for each pixel [4], indicating the segmentation class it belongs to.

Nowadays, radar and LiDAR sensors are rarely applied in map segmentation tasks since their sparse semantics lead to decreased accuracy on semantic-oriented tasks. Camera sensors are the most popular sensors used in map segmentation tasks thanks to their dense semantics [4]. Generally, map segmentation refers to BEV map segmentation. To get the complete BEV maps, multiple horizon cameras (e.g. six cameras) are utilized. Figure 1.4.c) presents one sample multi-view BEV map segmentation result.

1.4 Attention

To enhance the performance of models, attention mechanisms are applied in 3D object

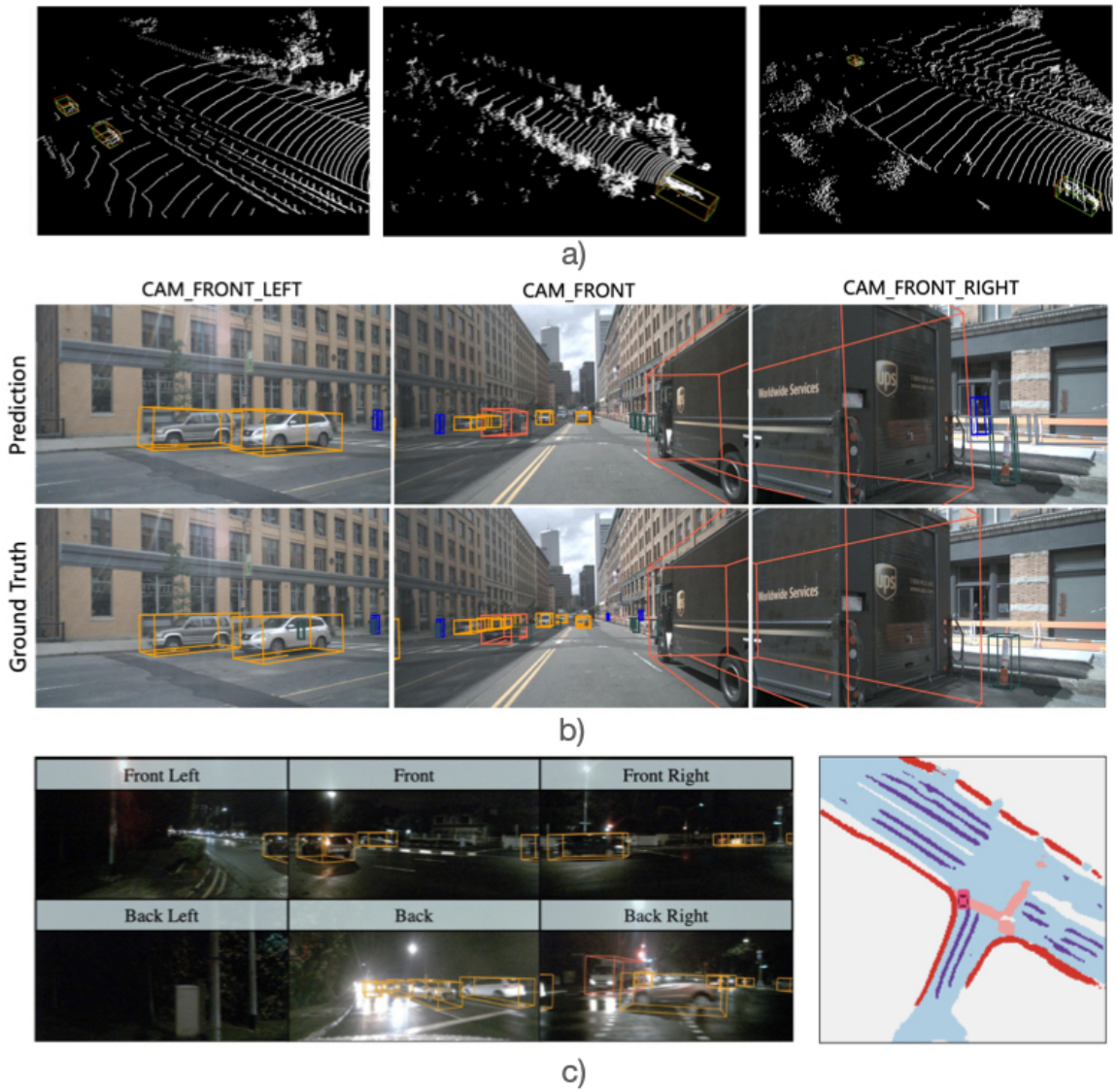

Fig. 1.4: Illustrations of the LiDAR-based 3D object detection, multi-view 3D object detection, and multi-view map segmentation: a) 3D object detection results on LiDAR images; b) 3D object detection results on multiple images (adapted from BEVFormer [3]); c) Multi-view images and their BEV map segmentation results (adapted from BEVFusion [4]), where drivable area is shown in blue, land divider is shown in purple, walkway is shown in red, and crossing is shown in pink).

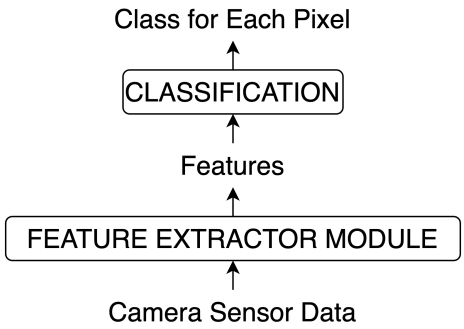

Fig. 1.5: The block diagram of deep learning-based map segmentation

detection and map segmentation, which help the models focus on relevant information and assign varying weights to different parts of the input data. Three kinds of attention mechanisms have been used.

Spatial attention is employed to selectively attend to specific points or pixels in the 3D point cloud data and images. By applying weights to different points or pixels, the models are able to focus on the important regions and filter out noises [24]. Channel attention is employed to emphasize important features or channels in the feature maps. By applying weights to different channels, the models are able to focus on the most informative features or channels and suppress redundant ones [25]. Graph attention is employed to capture dependencies and relationships between elements in point clouds or images. By applying weights to the edges between elements, the models are able to learn the useful relationships and contextual information between elements [26].

1.5 Organization of Dissertation

In this dissertation, we aim to explore how to effectively utilize accurate input information for 3D object detection and map segmentation without considering economic costs. Specifically, we explore how to utilize the accurate depth information captured by LiDAR sensors in 3D object detection tasks and the dense semantic information captured by camera sensors in map segmentation tasks. Chapter 2 provides the background knowledge of deep learning methods and attention mechanisms and introduces the related work in 3D object detection and BEV map segmentation. Chapter 3 describes the proposed attention

modules utilized in our methods. Chapter 4 presents the proposed 3D object detection and map segmentation frameworks. Chapter 5 presents the details of our experiments, including the experiment datasets, experiment setup, evaluation metrics, and experiment results. Chapter 6 concludes the dissertation and discusses future research directions.

CHAPTER 2

RELATED WORK

We briefly review representative works that are directly related to the proposed methods. Specifically, we review related works in deep learning, attention mechanisms, 3D object detection, and map segmentation in BEV.

2.1 Deep Learning

Deep learning is a subfield of Artificial Intelligence (AI) and Machine Learning (ML), both of which are based on artificial neural networks to perform cognition tasks. Artificial neural networks are inspired by the structure and function of biological brains, where information processing occurs through interconnected neurons. The "deep" in deep learning refers to the depth of the neural networks, which means the utilization of multiple layers of interconnected nodes, known as artificial neurons or units, in the network. Every layer processes the input data, extracting higher-level semantic features when the network becomes deeper. These multiple layers enable the deep learning model to learn complex representations of the input data and to solve more complex tasks in computer vision, machine translation, natural language processing, drug design, bioinformatics, autonomous vehicles, and so on.

There are four main types of learning in deep learning: supervised learning, semi-supervised learning, unsupervised learning, and reinforcement learning. Supervised learning methods learn from labeled data to make predictions or classify new instances. Unsupervised learning methods learn patterns, information, and structure from unlabeled data without explicit guidance. Semi-supervised learning falls between supervised and unsupervised learning, which is trained on a dataset that contains both labeled and unlabeled data. Reinforcement learning methods learn to make decisions by receiving rewards or penalties via interaction with the environment.

In the dissertation, we focus on supervised learning-based object detection and BEV map segmentation.

2.2 Attention Mechanisms

Attention mechanisms have recently been widely employed in the Deep Neural Network (DNN) to focus on important parts of the input data to capture sufficient outstanding features. Three kinds of representative attention mechanisms include channel attention [27–29], spatial attention [30–32], and graph attention [33–36].

Channel attention [27–29] usually utilizes average-pooling and max-pooling operations to aggregate spatial information of a feature map and utilizes a shared network composed of a 1-D convolution layer and a softmax layer to exploit the inter-channel relationship of features. For example, SENet [27] utilizes a Squeeze-and-Excitation (SE) block to capture channel-wise relationships and improve representation ability.

Spatial attention [31, 37] usually applies average-pooling and max-pooling operations along the channel axis and concatenates them to generate an efficient feature descriptor across the spatial domain to capture the inter-spatial relationship of features and select the most relevant spatial regions. Some spatial attentions tend to have a high time complexity, which limits their applications. For example, non-local network [31] first uses spatial attention to model non-local relationships, but it has a high time complexity.

Graph attention [33–36] understands attention from a graph learning perspective. For example, GloRe [34] utilizes graph convolutional networks to build attention mechanisms. It first collects N input features into M nodes by multiplying matrices and then learns an adjacency matrix of global interactions between nodes. Finally, the nodes distribute global information to input features. However, the low-dimension matrix projection might cause information loss.

2.3 3D Object Detection

The industry mainly utilizes LiDAR and camera sensors to solve the object detection task, due to their complementary strengths and ability to provide rich and accurate data

about the environment. Specifically, LiDAR sensors directly measure distances to objects by emitting laser pulses. This produces a dense and accurate 3D point cloud representation of the environment, which makes it well-suited for object detection and localization. On the other hand, cameras provide high-resolution RGB images that carry a lot of visual information about the objects and the scene. This information is useful for object recognition, classification, and contextual understanding. Depending on the number of cameras employed, 3D object detection is roughly divided into two categories including single-view 3D object detection and multi-view 3D object detection. Multi-view 3D object detection has better performance and provides more information for safe autonomous driving since single-view images are unable to convey perspectives of the immediate surroundings. Here, we provide the related work of LiDAR-based 3D object detection and multi-view 3D object detection.

2.3.1 LiDAR-based 3D Object Detection

The LiDAR 3D object detectors can be divided into voxel-based and point-based detectors based on data preprocessing methods. The voxel-based detectors transform the LiDAR point clouds to grid representations such as 3D voxels [13, 16, 19, 38]. We could utilize 3D Convolutional Neural Networks (CNN) to process the voxels effectively. The point-based detectors [17, 18, 39] directly process raw LiDAR point clouds to extract features. Generally speaking, the voxel-based detectors are more efficient, while the point-based detectors have more expensive computations [16].

Due to the importance of high efficiency for autonomous vehicles, we focus on voxel-based detectors [13, 19, 38] in this dissertation.

Here, we briefly review several influential works in voxel-based single-shot 3D object detection. VoxelNet [13] is a pioneer work in voxel-based 3D object detection. To improve the detection accuracy of LiDAR datasets, it divides a raw point cloud into equal 3D voxels and applies the voxel feature encoding layer to transform a group of points within each voxel into a feature representation. SECOND [38] improves VoxelNet by employing an improved sparse convolution method to increase both training and inference speed and significantly

reduce the detection time. It also introduces a new form of angle loss regression to improve the orientation estimation performance and a new data augmentation approach to enhance the convergence speed and performance. TANet [40] improves SECOND by utilizing a Triple Attention (TA) module, which consists of channel, point, and voxel-level soft attentions, to capture fine-grained features and increase module robustness. However, the lack of multi-scale features and ineffective use of high-semantic features hinder the performance of effective single-shot voxel-based detectors.

2.3.2 Multi-view 3D Object Detection

The input data of multi-view 3D object detection is multiple-view images taken from multiple cameras in the ego car. Compared to the traditional monocular-based 3D object detection and stereo-based 3D object detection, multi-view 3D object detection has the superiority of generating a BEV feature map containing all ego surrounding information. Multi-view 3D object detection could be roughly divided into two categories including transformer-based and CNN-based multi-view 3D object detection.

Recently, transformer-based BEV models have shown outstanding performance, which does not require Non-Maximum Suppression (NMS) processing. BEVFormer [3] generates BEV feature maps to contain useful environment information. To make full use of the BEV feature and introduce more information, BEVFormer proposes BEV query, spatial cross-attention, and temporal self-attention to combine temporary and spatial information. Detr3d [41] proposes a method that generates prediction directly in 3D space. Specifically, it processes multiple view input images to get 2D features and then utilizes the sparse 3D object query to index the 2D features. Finally, it projects 3D positions back to the multi-view image domain using the camera transformation matrices. Another state-of-the-art multi-view 3D object detector, PETR [21], proposes a new position embedding module called the 3D position embedding, which generates the position information of 3D coordinates and adds it into image features to get the 3D position-aware features. In that way, object query is able to process 3D position-aware features, perform end-to-end 3D object detection, and obtain 3D detection results directly.

CNN-based 3D object detectors also receive widespread attention due to their outstanding performance. BEVDet [20] substantially improves the existing modules' performance by proposing a data augmentation strategy and improving the NMS strategy. It gains a good balance between accuracy and efficiency. M^2BEV [42] proposes four designs including efficient BEV encoder design, dynamic box assignment, BEV center-based re-weighting, and large-scale 2D detection auxiliary supervision to further improve the detection results. Specifically, M^2BEV achieves state-of-the-art results in 3D object detection on the nuScenes dataset.

2.4 Map Segmentation in BEV

The industry mainly utilizes camera sensors to solve the map segmentation task. Radar and LiDAR sensors are rarely applied in map segmentation tasks since their sparse semantics lead to decreased accuracy on semantic-oriented tasks. Therefore, we only introduce the related work of the camera-based multi-view BEV map segmentation.

2.4.1 Multi-view BEV Map Segmentation

BEV map segmentation has two common backbones, deep CNN [5, 22, 42] and transformers [3, 43–45].

CNN-based BEV map segmentation utilizes intrinsic and extrinsic matrices of the cameras to translate multi-view autonomous vehicle images into the BEV features and applies CNN to further process the transformed BEV features and obtain the output maps. Lift, Splat, Shoot (LSS) [22] is a pioneer work of CNN-based BEV intermediate representation structure techniques. It builds a BEV transform network to yield better BEV segmentation results. Specifically, it utilizes CNN to get a feature map frustum in each view and combines all frustums into a unified BEV intermediate representation containing the geometric connection of each camera. The BEV representation is then further processed by CNN to get the final outputs. BEVerse [5] utilizes CNN to process multi-camera videos and get spatiotemporal BEV representations. The spatiotemporal BEV representations contain both spatial and temporal information, which are further processed by CNN-based decoders

to get the outputs. However, CNN-based BEV map segmentation networks require plenty of convolutional layers to gain semantic relationships of the whole BEV features due to the limited receptive field of each convolutional layer. In addition, they tend to have an overfitting issue and a lack of flexibility in the BEV feature domain [20].

On the other hand, transformer-based models employ transformers, which concentrate on local patches or part of the global information rather than the overall region relationship, to directly analyze multi-view image features. Cross-View Transformer (CVT) [43] is one representative transformer-based segmentation technique. It uses a camera-aware transformer together with the intrinsic and extrinsic matrices of cameras to obtain BEV segmentation from multi-view monocular images. Specifically, it uses position embeddings to encode input patches from different cameras and allow the transformer to gain implicit cross-view information. They pay attention to the local patches but ignore the importance of global information and high-semantic information. CoBEVT [44] is one of the state-of-the-art transformers that focuses on both local and global information. Specifically, CoBEVT designs a Fused AXial (FAX) attention module, which captures sparsely local and global spatial interactions across views and agents. However, it cannot obtain global semantic relationships using the sparsely sampled attention mechanism.

In summary, traditional BEV semantic segmentation does not consider contextual relationships of the global features or the high-semantic features at the early stage of the network. This leads to inaccurate segmentation, especially at the border of the objects and the regions. In addition, the overfitting issue is also a common problem in traditional BEV semantic segmentation.

2.5 Proposed Method

In this dissertation, we introduce two perception networks, which aim to simultaneously address the aforementioned drawbacks of their peers and improve the performance of their peers. One of these networks is for 3D object detection utilizing LiDAR sensors and the other one is for map segmentation utilizing camera sensors.

We name the 3D object detection network as Point Cloud Detection Network (PCDet),

which utilizes LiDAR sensors to obtain the point cloud inputs with accurate depth information. To solve the problem of lacking multi-scale features and using the high-semantic features ineffectively, the proposed PCDet individually utilizes Hierarchical Double-branch Spatial Attention (HDSA) and Hierarchical Residual Graph Convolutional Attention (HRGCA) to detect 3D objects. HDSA captures high-semantic and fine-grained features at the same time. HDSA applies the double-branch spatial attention at the early stage and the late stage of the network, which helps use the high-semantic features at the beginning of the network and obtains the multi-scale features. HRGCA is an effective attention mechanism, consisting of a spatial fully connected graph and a channel fully connected graph, to estimate global semantic relational features to solve the ineffective utilization of high-semantic features.

We name the map segmentation network as Multi-View Segmentation in Bird's-Eye-View (BEVSeg), which utilizes multiple camera sensors to obtain multi-view image inputs with dense semantic information. The proposed BEVSeg aims to utilize high-semantic features effectively and solve the common overfitting problems in map segmentation tasks. Specifically, BEVSeg individually incorporates the same attention mechanisms used in the PCDet, namely, HDSA and HRGCA, to capture high-semantic and fine-grained features at the early stage and the late stage of the network. In addition, BEVSeg incorporates an Aligned BEV domain data Augmentation (ABA) module to align the augmented object and segmentation ground truths and align the augmented BEV map and its augmented ground truths to address overfitting issues.

CHAPTER 3

ATTENTION

In this chapter, we introduce two effective attention mechanisms, namely, Double-branch Spatial Attention (DSA) and Residual Graph Convolutional Attention (RGCA), and their corresponding hierarchical structures HDSA and HRGCA. Both HDSA and HRGCA are utilized in our proposed 3D object detection framework and multi-view segmentation in the Bird's-Eye-View framework. The two attentions are proposed to utilize high-semantic features in an effective way. DSA enlarges the receptive field in spatial attention to get high-semantic information. RGCA utilizes graph convolution to estimate global semantic relational features. Because of the structure difference, both attentions have different advantages and disadvantages. DSA has low complexity and high flexibility. However, it lacks consideration of global relational information. RGCA is able to estimate global semantic relational information, which solves the limitation of DSA. However, it is not as flexible as DSA, due to its requirements of the graph space projection (i.e., the attention kernel size must be divisible by the node number).

3.1 Double-branch Spatial Attention (DSA) and its Two Variants

To capture high-semantic and fine-grained features at the same time, we propose a DSA module to automatically choose significant regions of the input features. Figure 3.1 illustrates the structure of DSA.

To simplify the complicated matrix calculation and reduce the computational complexity of spatial attention, we replace the matrix multiplication with a double-branch convolution. DSA consists of two control gate mask branches: one going through two convolutional layers and one going through one convolutional layer. Here, we use W to represent the 2D convolutional layer weights and the subscript of W to represent specific convolutional layers. Specifically, W_X, W_{XY}, and W_Y represent convolutional layer weights of the con-

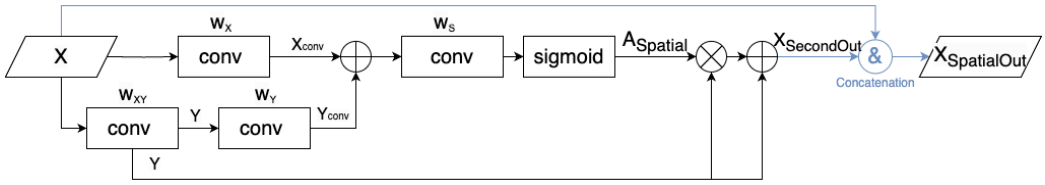

Fig. 3.1: Structure of the Double-branch Spatial Attention.

volutional layer connecting feature X and its convolved feature (i.e., X_{conv}), features X and Y, and feature Y and its convolved feature (i.e., Y_{conv}), respectively. For the input of the 2D convolutional networks (e.g., a given BEV feature map X), we use two branches of 1×1 convolutions W_X and W_{XY} to transform X into two new feature maps X_{conv} and Y with the same dimension, respectively. We then employ another $d \times d$ convolution W_Y to transform Y into Y_{conv} with the same dimensions and enlarge the receptive field at the same time. We combine X_{conv} and Y_{conv} via the elementwise addition and apply another 1×1 convolution W_S, which represents convolutional layer weights of the convolutional layer after the addition operation, to transform the combined feature map into a new feature map. The sigmoid operation is then employed on the new feature map to normalize it into a new weighted feature map $A_{Spatial}$. The feature map Y is elementwisely multiplied with $A_{Spatial}$, added with itself, and concatenated with input X to obtain the final weighted feature map $X_{SpatialOut}$.

The proposed DSA module allows the control gate to perform pixel-to-pixel modeling (i.e., voxel-wise addition or element-wise addition of local features) to make the focused and related resources be assigned to the most intrinsic and informative areas. In other words, the control gate branches function as a masking mechanism to recalibrate local features from multiple scales selectively strengthen valuable and informative areas, and suppress useless and non-informative features such as noise and background. As a result, the values in the attention mask represent the weights of corresponding pixels on the original feature maps of point clouds, which makes the attention mask more suitable for pixel-wise classification than global pooling. In summary, the proposed DSA module not only selects the most intrinsic and discriminative features toward the classification objective in the

feed-forward process but also prevents the updating of parameters with incorrect gradients during backpropagation [46]. It further makes our network more expressive, robust, and informative.

We also design two variant attentions for comparison.

1. Variant 1 intuitively applies attention to the original input feature map X to enhance target objects and filter out irrelevant areas in X. Its final weighted feature map is obtained by adding X and its multiplication with the weighted feature map $A_{Spatial}$, i.e. $X + A_{Spatial} \times X$.

2. Variant 2 applies the soft attention on the high-level feature map Y to enhance target objects, filter out irrelevant areas in Y, and learn more deformations of target objects. Its final weighted feature map is obtained by adding Y and its multiplication with $A_{Spatial}$, i.e. $Y + A_{Spatial} \times Y$.

The proposed DSA takes advantage of both variants to simultaneously consider both low and high-level feature maps, which contain rich and fine context information, by concatenating X with the final feature map obtained from variant 2.

3.2 Design and Mathematical Formulation of DSA

We treat X and Y_{conv} shown in Figure 3.1 as low-level features of layer L and high-level features of layer $L+2$ in the encoding stage, respectively. X_{conv} and Y are considered as high-level features of layer $L+1$. To improve the network sensitivity, we design our DSA block based on additive attention since it experimentally has higher accuracy than multiplicative attention [47]. To this end, we calculate the attention mask $A_{Spatial}$ at layer L by performing an elementwise addition operation between X_{conv} and Y_{conv} to learn critical features of objects. This attention mask $A_{Spatial}$ integrates the relationship between features from multiple scales or layers at different regions, focuses on useful regions, and indicates the significance of different regions. We then perform an elementwise multiplication operation between $A_{Spatial}$ and Y to identify relevant regions containing objects. Finally, we employ the addition [46] to retain original features, so the final output of the DSA block is defined

as $Y + Y \circ A_{Spatial}$, where $\circ$ represents the elementwise multiplication. The following equation summarizes the steps to compute the attention mask $A_{Spatial}$ at layer L, where concatenation is not involved:

$$A^L_{Spatial} = Sigmoid(W_s \star (W_X \star X^L + W_Y \star W_{XY} \star X^L))$$

Here, $X^L \in R^{H^L * W^L * Ch^L}$ are features at layer L with H, W, and Ch being its respective height, width, and channel number, $\star$ represents the conventional convolution operation, and W_X, W_Y, W_{XY}, and $W_S \in R^{Ch^L * Ch^L * k * k}$ are convolutional filters, whose kernel size is $k \times k$, used at different layers L to generate features at the next layers. The proposed DSA block functions as a feature selector to automatically augment useful structure features during the forward process by replacing X^L with the weighted attention features $X_{SpatialOut}$ concatenating with X^L. In summary, the final output of the variant 1, the variant 2, and the proposed DSA block are $X + X \circ A$, $Y + Y \circ A$, and concatenation of X and $Y + Y \circ A$, respectively.

3.3 Residual Graph Convolutional Attention (RGCA)

To utilize high-semantic features effectively, we propose an RGCA module to estimate global semantic relational features.

The RGCA module aims to learn the relationship between each region along both spatial and channel dimensions while maintaining the coordinate information. It consists of three sub-modules: graph space projection, RGC layers, and coordinate space reprojection. Figure 3.2 illustrates the structure of RGCA.

Graph Space Projection: To project coordinate features into the graph space, we utilize the stride convolution operation φ to convert input features $X \in \mathbb{R}^{C \times H \times W}$ to downsampled features $V_D \in \mathbb{R}^{C \times \frac{H}{d} \times \frac{W}{d}}$: $V_D = \varphi(X)$ where d represents the stride size. Since the filters have a kernel size of $d \times d$ and are not overlapping, the convolution operation φ is able to efficiently gather the complete feature information.

RGC Layers: We obtain downsampled features V_D after the graph space projection.

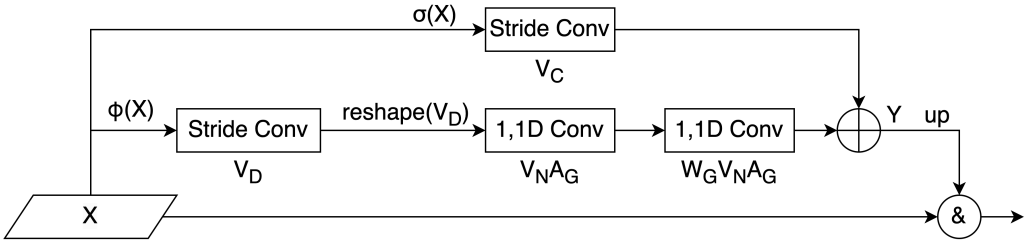

Fig. 3.2: Illustration of the Residual Graph Convolutional Attention

With the reshaping process, we transform V_D to graph node features $V_N \in \mathbb{R}^{C \times \frac{HW}{d^2}}$, where $\frac{HW}{d^2}$ represents the node number D_N and C represents the channel number for each node. In order to get the relationship between every region from both the spatial and channel dimensions, we build two completely linked graphs, namely the spatial graph and the channel graph, in the RGC module. The spatial graph learns the relationship between each node, where a node represents multi-channel node features and an edge represents the relationships between node features. The channel graph learns the interdependency between each channel, where a node represents node features along one channel and an edge represents the relationships between channels of each node feature. The two graphs contain the node adjacency matrix $A_G \in \mathbb{R}^{D_N \times D_N}$ and the channel-specific weights $W_G \in \mathbb{R}^{C \times C}$ for each node. They are used to compute interactive features $V_G \in \mathbb{R}^{C \times \frac{HW}{d^2}}$ of the RGC layer by

$$V_G = W_G \times V_N \times A_G \tag{3.1}$$

During the training process, A_G and W_G are randomly initialized and optimized along with other network parameters by the Stochastic Gradient Descent (SGD) method. It is worth noting that V_G has the same dimension as V_N. However, it not only captures the relationship of the nodes but also the relationship of the channels within each node. We further reshape V_G to Y_G to have the same dimension as V_D. To seamlessly combine the low-resolution coordinate features and graph features, we propose a downsample residual process to obtain combined features Y by

$$Y = Y_G + \sigma(X) \tag{3.2}$$

where σ represents a downsampling convolution operation to transform X to $V_C \in \mathbb{R}^{C \times \frac{H}{d} \times \frac{W}{d}}$ with the same dimension of Y_G.

Coordinate Space Reprojection: After the graph interaction, we utilize the upsampling operation to reproject Y back to the original coordinate space $\mathbb{R}^{C \times H \times W}$ to be consistent with the network architecture. Specifically, the bilinear interpolation $F_{bilinear}$ is adopted to upsample Y by d times to obtain residual graph features X_G:

$$X_G = F_{bilinear}(Y, scale = d) \tag{3.3}$$

After reprojection, the residual graph features X_G are finally concatenated with input features X to maintain both original information and processed information inputs as $(Concat(X, X_G))$. Figure 3.2 illustrates all the operations involved in the RGCA module.

3.4 Hierarchical Attention Modules

The two proposed attention modules could be applied in the early stage and the late stage of both 3D object detection and map segmentation tasks to form a hierarchical attention module, which accurately captures multi-scale features and effectively uses these features to better represent the objects. On the one hand, we utilize the HDSA module in PCDet, the 3D object detection network, to obtain the high-semantic features at the beginning of the network and obtain multi-scale features at the late stage of the network. In addition, we utilize the HDSA module in BEVSeg, the map segmentation network, to capture high-semantic and fine-grained features and utilize them effectively. In general, the HDSA enables the DNN to focus on useful areas and salient features. On the other hand, we utilize the HRGCA module in PCDet, the 3D object detection network, to obtain the global semantic relational 3D object features at both the early and late stages of the network. Besides, we utilize the HRGCA module in BEVSeg, the map segmentation network, to capture the global semantic relational BEV features from different scales. Generally speaking, the HRGCA provides the global semantic information for the DNN to highlight the useful areas and salient features.

CHAPTER 4

THE PROPOSED NETWORKS

In this chapter, we present the overall structure of the proposed 3D object detection network PCDet and the proposed map segmentation network BEVSeg. In addition, we introduce the application of the two proposed hierarchical attention modules HDSA and HRGCA to the two proposed networks.

4.1 Proposed 3D Object Detection Network PCDet

The proposed 3D object detection network, Point Cloud Detection Network (PCDet), utilizes LiDAR sensors to acquire input point clouds with precise depth data. PCDet individually uses the proposed Hierarchical Double-branch Spatial Attention (HDSA) and Hierarchical Residual Graph Convolutional Attention (HRGCA) to build two frameworks PCDet_HDSA and PCDet_HRGCA to address the issues of a lack of multi-scale features and the poor use of high-semantic data. High-semantic and fine-grained characteristics are simultaneously captured by HDSA. In order to leverage high-semantic characteristics at the beginning of the network and acquire multi-scale features, HDSA applies the double-branch spatial attention at both the early and late stages of the network. As a solution to the inefficient use of high-semantic features, HRGCA estimates global semantic relational characteristics using a spatial fully connected network and a channel fully connected graph.

The contributions of four versions of PCDet are summarized as follows:

- Proposing a 3D object framework PCDet that can easily incorporate different attention modules at different stages of the DNN to capture features at multiple scales and improve the detection accuracy of the SECOND network.
- Utilizing the features generated from the HDSA module to build PCDet_HDSA to learn and find the most important locations to focus on and filter out the irrelevant parts of the input point cloud.

- Incorporating the HRGCA module that contains both graph and coordinate information in the deep CNNs to build PCDet_HRGCA to not only effectively acquire the global information but also efficiently estimate contextual relationships of the global information in the 3D point cloud domain.

- Incorporating DSA and RGCA at either the early stage or the late stage to build PCDet_DSA+RGCA and PCDet_RGCA+DSA to respectively capture multi-scale high-semantic and fine-grained features and estimate global semantic relational characteristics to improve the detection accuracy.

- Improving the baseline network and achieving similar accuracy and inference speed compared with one-stage state-of-the-art systems on the KITTI validation dataset.

4.1.1 PCDet Overall Structure

Figure 4.1 shows the overall architecture of the proposed PCDet, which uses the widely used small SECOND network as its backbone to maintain the detection accuracy with a faster speed. SECOND is an effective 3D object detection system that achieves high accuracy at a fast speed. It first divides the input point cloud data into voxels of the same size for pre-processing. It then converts a certain number of points in each voxel into a vector of voxel features and coordinates to maintain geometric and spatial information. These vectors are next sent to 3D convolution blocks to expand their receptive fields. The 3D voxel features are reshaped into a BEV shape and sent to a 2D convolution block to obtain 2D features. Finally, the 2D features are put into box regression and classification branches to localize and classify detected objects, respectively.

To improve the efficiency and accuracy of SECOND, we use its small network as a backbone and modify this simple network structure from two perspectives. First, we cut 50% of the parameters of the last layer of 3D sparse convolutional layers to simplify the 3D convolutional block, speed up the training, and maintain efficiency. Second, we include the hierarchical attention modules in 2D convolutional blocks to improve the accuracy of SECOND. Specifically, we build the PCDet_HDSA framework by incorporating the HDSA

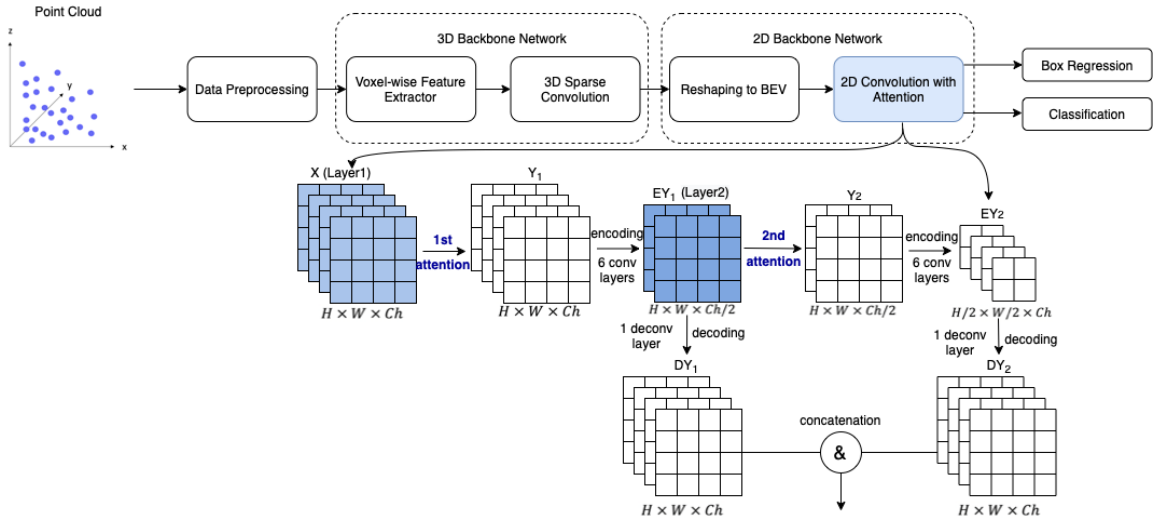

Fig. 4.1: The overall architecture of the proposed PCDet. The upper part demonstrates its block diagram, where hierarchical attention modules reside in the blue-shaded block titled "2D Convolution with Attention". The lower part presents the flowchart of employing the hierarchical attention modules in the 2D backbone network, where Layer 1 represents the results from the attention module at the early stage and Layer 2 represents the results from the attention module at the late stage. The DSA and RGCA attention can be interchangeably applied at either the early stage or the late stage to build a different 3D object detection framework.

module at both the early and late stages of the SECOND network, where the HDSA module learns the most important positions in the point cloud data, filters out the irrelevant parts, and combines feature maps of different scales to more accurately represent objects. We build the PCDet_HRGCA framework by incorporating the HRGCA module at both the early and late stages of the SECOND network, where the HRGCA module obtains the global relational information in the point cloud data, clusters the similar semantic regions, and reasons the relationships of different regions to get more connect information.

The section below the block diagram of Figure 4.1 presents the details of employing the proposed hierarchical attention modules at two places in the 2D backbone network, which consists of two layers of encoding and decoding blocks. The encoding block at each layer contains six convolutional layers and the decoding block at each layer contains one deconvolutional layer. We use X to denote SECOND's 2D BEV features, which is the input of the 2D backbone network. We first employ the attention module on X to calculate semantic relationships among voxels and obtain its weighted feature map Y_1 at the first layer. This weighted feature map Y_1 then goes through the encoding block of the first layer to obtain EY_1, whose channel number is reduced by half from Ch to $Ch/2$. EY_1 goes through two branches: one branch is to go through the decoding block of the first layer to obtain DY_1, which has the same dimension as X; the other branch is to employ the attention module on EY_1 to calculate semantic relationships among voxels and obtain its weighted feature map Y_2 at the second layer. This weighted feature map Y_2 goes through the encoding block of the second layer to obtain EY_2, whose channel is doubled from $Ch/2$ to Ch and whose height and width are reduced from H to $H/2$ and from W to $W/2$, respectively. EY_2 finally goes through the decoding block of the second layer to obtain DY_2, which has the same dimension as X. Lastly, DY_1 and DY_2 are concatenated together as the output of the 2D backbone network. This hierarchical structure combines features at different scales, enhances semantic information, and broadens the receptive field.

4.2 Proposed Map Segmentation Network BEVSeg

In this section, we propose a map segmentation framework, i.e., Multi-View Segmentation in Bird's-Eye-View (BEVSeg), to solve map segmentation tasks utilizing dense semantic information from multiple camera sensors. BEVSeg proposes three new modules, namely, data augmentation, Segmentation Head (SH), and a hierarchical attention module to address issues of overfitting, misalignment, overlapping, the lack of multi-scale features, and the poor usage of high-semantic data. Specifically, the proposed data augmentation, i.e., Aligned BEV domain data Augmentation (ABA), addresses overfitting and misalignment issues. The proposed SH individually segments each category, which solves the overlapping issues in map segmentation. Two proposed hierarchical attention modules, HDSA and HRGCA, address the issues of the lack of multi-scale features and the poor usage of high-semantic data. However, BEVSeg with different attention mechanisms leads to different advantages and disadvantages. For example, BEVSeg_HDSA, which incorporates DSA at both the early and late stages of the network, is efficient and flexible because of the structure of DSA. However, it lacks consideration of global relational information. BEVSeg_HRGCA, which incorporates RGCA at both the early and late stages of the network, solves the limitation of BEVSeg_HDSA by using an HRGCA module to estimate global semantic relational information. However, it is not as flexible as BEVSeg_HDSA due to its requirements that the attention kernel size must be divisible by the node number.

The contributions of four versions of BEVSeg are as follows:

- Proposing a new network architecture BEVSeg to perform semantic segmentation of a scene with multi-view images and achieve state-of-the-art results.
- Incorporating ABA in the geometry module to augment the coherent BEV map, align the augmented object and segmentation ground truths, and align the augmented BEV map and its augmented ground truths to address overfitting and misalignment issues.
- Extending the SH to individually process each semantic category to address the possible overlapping among semantic categories.

- Incorporating low-complexity HDSA in the data-driven module to build BEVSeg_HDSA to learn multi-scale BEV features flexibly by enlarging the feature receptive field and learning interest regions.

- Incorporating the HRGCA module in the data-driven module to build BEVSeg_HRGCA to gather the global semantic relationship from different scales.

- Incorporating DSA and RGCA in the data-driven module at either the early stage or the late stage to build BEVSeg_DSA+RGCA and BEVSeg_RGCA+DSA to respectively capture multi-scale high-semantic and fine-grained features and estimate global semantic relational characteristics to improve the segmentation accuracy.

- Improving the baseline network in terms of segmentation accuracy for six major semantic categories.

BEVSeg_HRGCA improves BEVSeg_HDSA by utilizing an effective HRGCA attention mechanism to obtain the global information at the early and late stages of the network. Its contributions are as follows:

- Proposing a novel RGCA module consisting of two interconnected graphs (i.e., the spatial graph and the channel graph), where the spatial graph extracts spatial information between each node and the channel graph extracts channel information within each node. The RGC module employs a downsample residual process to enhance the coordinate feature reuse to maintain the global information. It also employs a non-overlapping graph space projection to efficiently project the complete BEV information into graph space.

- Incorporating the RGCA module that contains both graph and coordinate information in the deep CNNs to enable BEV semantic segmentation networks to not only effectively acquire the global information but also efficiently estimate contextual relationships of the global information in the BEV map domain produced by multi-view images.

In Section 4.2.1, we introduce the overall architecture of BEVSeg. In Section 4.2.2, we introduce the basic modules of BEVSeg. In Section 4.2.3, we introduce three proposed modules in BEVSeg. The structure of hierarchical attention modules is introduced in Chapter 3. Interested readers may refer to Chapter 3 for more details.

4.2.1 Overall Architecture of BEVSeg

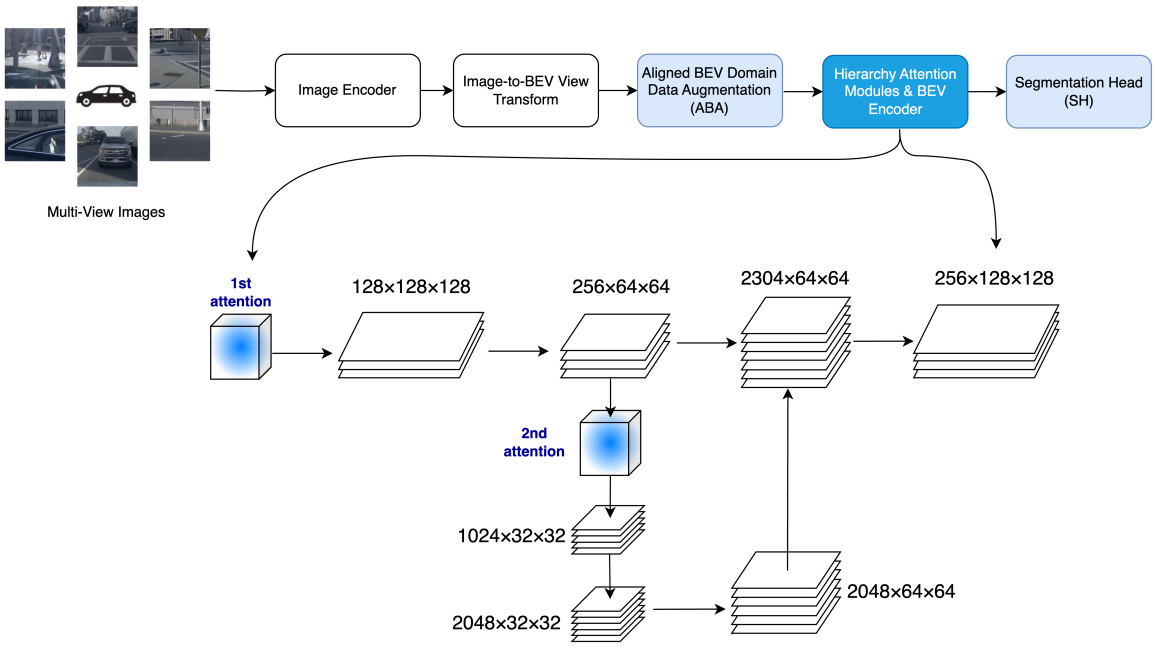

Fig. 4.2: Illustration of BEVSeg's overall architecture: 1) image encoder extracts multi-camera features; 2) Image-to-BEV view transform converts multi-camera features to the BEV domain; 3) ABA shown in the light blue shaded block is the proposed geometry module that augments and aligns BEV features and ground truths; 4) hierarchical attention modules & BEV encoder shown in the dark blue shaded block is the proposed data-driven module that encodes BEV feature to obtain high-semantic information from different scales; 5) BEV SH shown in the light blue shaded block is proposed to replace the detection head in BEVDet and predict BEV binary maps for six categories.

We propose a new network architecture BEVSeg, which uses multi-view camera images to produce better segmentation results in a BEV map. Figure 4.2 illustrates the overall architecture of the proposed BEVSeg framework, which consists of five modules including image encoder, image-to-BEV view transform, Aligned BEV domain data Augmentation

(ABA), Hierarchical Attention modules & BEV encoder, and BEV map Segmentation Head (SH). Image encoder and Image-to-BEV view transform are directly adopted from the conventional BEV network [20], which is introduced in Section 4.2.2. ABA (the geometry module) is the proposed module, which is shown in the light blue shaded block. Hierarchical Attention modules combining DSA and RGCA in the data-driven module at the early stage or the late stage of the network are the proposed modules shown in the dark blue shaded block. The lower part presents the flowchart of employing the hierarchical attention modules in the BEV encoder. SH is the last proposed module, which is shown in the light blue shaded block. All the proposed modules are introduced in Section 4.2.3.

BEVSeg projects camera features into the unified BEV intermediate representation via ABA in the geometry module. It also utilizes the combination of DSA and RGCA at the early stage or the late stage of the network to construct a hierarchical attention module to process the BEV features. BEVSeg_HDSA, which utilizes DSA at both early and late stages, learns low-complexity HDSA in the data-driven module to optimize the implicit geometric information in BEV feature maps and learn multi-scale BEV features flexibly via enlarged feature respective field and learned interest regions. HDSA utilizes the DSAs (mentioned in Section 3.1, Section 3.2) in the hierarchy. However, HDSA has one limitation it does not consider the global relational information, which will be solved by HRGCA. BEVSeg_HRGCA, which utilizes RGCA at both early and late stages, utilizes multi-view image inputs to generate graph-based BEV features and produce BEV segmentation results. In different scales, the HRGCA module effectively combines the region interaction information and the BEV coordinate information to increase segmentation accuracy, which solves HDSA's problem. HRGCA utilizes the RGCA modules (mentioned in Section 3.3) in the hierarchy. The RGCA module estimates relationships between each region in spatial and channel dimensions while maintaining the coordinate information using its residuals.

4.2.2 Basic Architectures

We build our network, BEVSeg, based on the modular 3D object detector, BEVDet [20], a multi-view 3D object detector consisting of a simpler modular structure compared with

other networks. In this section, we introduce the three basic modules used in our map segmentation network:

Image Encoder: The image encoder aims to efficiently represent multi-view camera images to facilitate the later learning process. It usually uses a backbone to convert multi-view camera images to multi-scale feature maps containing both high-level semantic information and low-level texture information. It then uses a neck structure to aggregate feature maps of various resolutions to obtain a compact representation of multi-review camera images. In the proposed system, we use the most recent SwinTransformer [14] as our backbone due to its outstanding multi-task performance and use the traditional LSS-based neck structure [22] as our neck.

Image-to-BEV View Transform: The image-to-BEV view transform [20,22] aims to project multi-view image features into the BEV domain to create BEV map features, which are a coherent representation of the surrounding environment at the same ego-direction. It also predicts the depth of each view and renders the unified 3D point cloud, which can be further utilized in later modules to achieve better segmentation results.

BEV Encoder: The BEV encoder aims to further process BEV features to facilitate segmentation by combining high-level semantic and low-level texture information in a pyramid structure. Similar to the image encoder, the BEV encoder consists of a backbone and a neck, which are constructed by traditional residual blocks [23].

4.2.3 Proposed Modules in BEVSeg

In this section, we introduce the proposed modules of BEVSeg, which are uniquely incorporated to improve segmentation accuracy. The structure of hierarchical attention modules is introduced in Chapter 3.

Aligned BEV Domain Data Augmentation (ABA)

In order to improve network performance and address the overfitting and misalignment problem in the BEV domain, we propose an ABA technique to simultaneously augment and align BEV feature maps and the ground truths, including 3D object and segmentation

ground truths. By aligning three augmented counterparts, ABA ensures that most 3D object ground truths are located in the drivable area (e.g. the 3D vehicles are in the drivable area), which increases the model interpretability. Specifically, we apply rotation, flipping, and scaling operations on the BEV feature maps and their corresponding ground truths. Unlike the augmentation in BEVFormer, BEVFusion, CVT, and BEVDet, our module preserves semantic alignment between the augmented object and segmentation ground truths and semantic alignment between the augmented BEV map and its augmented ground truths.

Given 3×3 rotation and flipping transformation matrices T_{Rot} and T_{Flip} and the scaling transformation parameter S, the augmented BEV feature map $M_{AUG-BEV}$ is generated by:

$$M_{AUG-BEV} = S \times T_{Flip} \times T_{Rot} \times M_{BEV} \tag{4.1}$$

where M_{BEV} is the BEV feature map.

In order to align ground truth with augmented BEV feature maps, the corresponding rotation angle is computed by the Euler angle formula [48]:

$$angle = \arctan\left(\frac{T_{Rot}^{-1}(2,1)}{T_{Rot}^{-1}(1,1)}\right) \tag{4.2}$$

where $T_{Rot}^{-1}(x,y)$ denotes the element at the coordinate of (x,y) of the inverse rotation matrix. The augmented ground truth segmentation results G_{AUG} are estimated by:

$$G_{AUG} = S \times T_{Flip} \times G(loc, angle, size) \tag{4.3}$$

where $G(loc, angle, size)$ is the ground truth segmentation result rotated by $angle$ at location loc with a size of $size$, which is a predefined value directly adopted from BEVFusion [4]. We adopt the data augmentation idea in [20] to estimate the augmented 3D object ground truths. We finally align the augmented BEV feature map, augmented segmentation ground truths, and augmented 3D object ground truths.

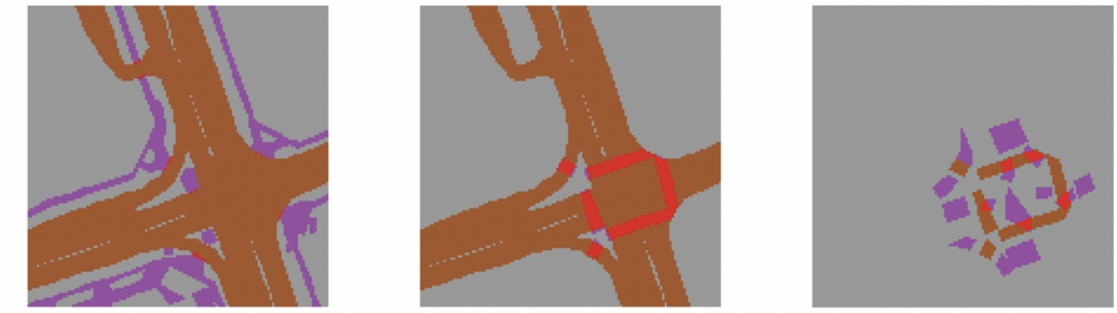

Fig. 4.3: Illustration of two categories in BEV map, where overlap regions are shown in red. Left to right: non-overlap between drivable area (brown) and walkway (purple); non-overlap between drivable area (brown) and ped crossing (purple); non-overlap between ped crossing (brown) and stop line (purple).

BEV Map Segmentation Head (SH)

For multi-view images, one category may cross over into another category in the BEV representation. Figure 4.3 illustrates three scenarios, where areas in red represent the overlapping of two categories. To address the aforementioned overlapping issue, we utilize one binary mask to store segmentation results for one category. Since there are six major 3D object categories, we generate six binary masks to store their respective segmentation results.

The structure of the proposed BEV SH contains eight 3×3 convolutions and one 1×1 convolution due to the unified BEV representation. To further reduce the number of parameters, we reduce output channels by half. The BEV SH result is $R^{H_s \times W_s \times N_t}$, where H_s and W_s are the height and width of the segmentation mask and N_t is the total number of categories. The loss is the sigmoid focal loss [49].

CHAPTER 5

EXPERIMENTS

In this chapter, we first introduce the datasets utilized in our two tasks. Then we introduce the metrics utilized to evaluate our networks. In the last section, we introduce the performance, ablation study, and qualitative results of our networks.

5.1 Datasets and Experimental Setup

KITTI and nuScenes are both large-scale datasets used in the field of computer vision and autonomous vehicles, which play an important role in advancing research nowadays. We utilize the KITTI dataset in our LiDAR-based 3D object detection task. Since the KITTI dataset does not have multi-view images, we utilize another popular autonomous vehicle dataset, the nuScenes dataset, in our multiple camera-based map segmentation task.

KITTI dataset [50] is a novel challenging real-world computer vision benchmark captured by driving around the mid-size city Karlsruhe, its rural areas, and its highways. The dataset contains images, videos, 3D point clouds, and their Global Positioning System (GPS) locations. In this research, we focus on the KITTI 3D point cloud dataset, which has 7,481 training and 7,518 testing point clouds in three categories (e.g., cars, pedestrians, and cyclists). Each category has point clouds with three difficulty levels including easy, moderate, and hard based on bounding box height, occlusion, and truncation levels. The height of the bounding box of objects at easy, moderate, and hard difficulty levels is at least 40, 25, and 25 pixels, respectively. The occlusion of the objects at easy, moderate, and hard difficulty levels is fully visible, partly occluded, and hard to see, respectively. The truncated percentage of objects at easy, moderate, and hard difficulty levels is at most 15%, 30%, and 50%, respectively. Objects that do not satisfy the above requirements (e.g., 6,473 cars, 170 pedestrians, and 165 cyclists) are not used for training and validation. In total, the KITTI dataset contains 17,823 easy objects including 13,067 cars, 3,694 pedestrians, and 1,062

cyclists, 9,547 moderate objects including 8,602 cars, 563 pedestrians, and 382 cyclists, and 678 hard objects including 600 cars, 60 pedestrians, and 18 cyclists. We divide the training data into training and validation split with 3,712 and 3,769 point clouds, respectively.

NuScenes [1] is the most recent and popular benchmark for 3D object detection, tracking, and BEV semantic segmentation in autonomous driving. It is an extensive outdoor dataset consisting of 1,000 driving scenes collected in Boston Seaport and Singapore's One North, Queenstown, and Holland Village districts. Each scene is 20 seconds long and contains a LiDAR scan and RGB images from six horizon monocular cameras. Each scene is also labeled with semantic mask annotations for 11 semantic classes and additional bitmaps. We utilize the ego-motion measurements to produce the fixed-size ground truths of the corresponding area in the same ego direction. We also use BEVFusion's experimental setup to choose six crucial BEV semantic classes for all evaluations. The training, validation, and testing splits of the nuScenes dataset contain 700, 150, and 150 scenes, respectively.

5.2 Metrics

Average Precision (AP) and mean Average Precision (mAP) are the most popular metrics used to evaluate 3D object detection models. AP is calculated as the weighted mean of precision at each threshold. mAP is the average of AP of all classes. The higher AP and mAP values indicate better 3D object detection performance. We employ AP over 11 recall positions as the metric to evaluate the 3D object detection results in the validation split. Different IoU thresholds are empirically determined by other researchers to compute AP. An IoU of 0.7 is commonly used for cars and an IoU of 0.5 is commonly used for cyclists and pedestrians. The leaderboard rank is based on the results of the dataset at the moderate level.

Intersection-over-Union (IoU) and mean IoU (mIoU) are the two most common metrics to evaluate how well a map segmentation model performs. IoU is calculated by dividing the overlap between the predicted and ground truth annotation by the union of these. mIoU is the average of IoU of all classes. Higher IoU and mIoU values indicate better map segmentation performance. We use these two metrics to evaluate the performance of all

Attention	3D			BEV (2D)		
	Easy	Moderate	Hard	Easy	Moderate	Hard
no attention	85.5	75.04	68.78	89.79	86.2	79.55
N-L	87.4	78.11	**75.39**	91.57	87.82	**86.86**
SE	87.83	78.17	74.67	91.74	87.72	85.66
GCNet	88.16	78.14	74.6	**92.61**	87.91	86.14
TA	87.57	78.1	73.97	92.1	87.83	85.74
DSA Variant 1 (ours)	87.45	77.30	73.24	91.53	87.48	84.97
DSA Variant 2 (ours)	**88.59**	77.88	73.35	92.15	87.55	84.92
DSA (ours)	87.84	78.26	73.35	91.57	87.86	85.12
RGCA (ours)	88.46	**78.32**	74.97	92.2	**87.94**	85.17

Table 5.1: Comparison of car detection results at three difficulty levels of different attention-based small SECOND networks.

compared methods. We first calculate different IoU scores using thresholds ranging from 0.35 to 0.65 with a stepsize of 0.05. We then report the highest IoU score for each of the six crucial semantic classes (e.g., drivable area, ped crossing, walkway, stop line, carpark area, and divider). At last, we use the mIoU of six semantic classes as the major ranking metric to evaluate the overall segmentation performance of each method [20].

5.3 Results

We report the experimental results of the 3D object detection and map segmentation in Section 5.3.1 and Section 5.3.2

5.3.1 3D Object Detection Results

The proposed DSA and RGCA attention can be easily incorporated in any stage of the object detection baseline SECOND. Table 5.1 lists 3D and 2D car detection results in terms of AP at three difficulty levels (easy, moderate, and hard) of the baseline SECOND (without any attention), PCDet_DSA and its two variants, PCDet_RGCA, and baseline SECOND with four commonly used attention modules (e.g., Non-Local, SE, GCNet, and TA). To use the smallest computer resources, we use a small SECOND network as a backbone for all networks and implement all attention modules at the early stage. Specifically, N-L represents the Non-Local attention module [31], SE represents the Squeeze-and-Excitation

Attention	3D			BEV (2D)		
	Easy	Moderate	Hard	Easy	Moderate	Hard
no attention	85.5	75.04	68.78	89.79	86.2	79.55
DSA	87.84	78.26	73.35	91.57	87.86	85.12
RGCA	**88.46**	78.32	74.97	**92.2**	87.94	85.17
HDSA	88.36	78.49	**76.05**	91.04	87.86	86.1
DSA+RGCA	87.76	78.56	75.69	91.59	87.73	**86.90**
RGCA+DSA	88.19	78.76	75.15	91.97	**88.08**	86.41
HRGCA	88.16	**78.92**	75.68	92.04	88.04	86.77

Table 5.2: Comparison of car detection results at three difficulty levels of different hierarchical attention-based small SECOND networks.

attention module [27], GCNet represents Global Context attention module [51] and TA represents Triplet Attention module [52]. In the KITTI dataset, leaderboards are ranked based on the 3D object detection results at the moderate difficulty level. Table 5.1 shows that the proposed DSA and RGCA modules respectively improve the baseline network SECOND by 3.22% and 3.28% for cars of the moderate difficulty level. They also achieve better 3D car detection accuracy at the moderate difficulty level than all four commonly used attention modules. The DSA module outperforms its two variants in detecting 3D cars of the moderate difficulty level. In general, our RGCA-based SECOND network achieves the best 3D car detection accuracy and our DSA-based SECOND network achieves the second best 3D car detection accuracy. As a result, DSA and RGCA are utilized in the hierarchical structure for further experiments.

In order to demonstrate the effectiveness of the hierarchical structure, we compare the baseline (e.g., a small SECOND), the proposed DSA at the early stage with a small SECOND (e.g., PCDet_DSA), the proposed RGCA at the early stage with a small SECOND (e.g., PCDet_RGCA), and four hierarchical attention modules with a small SECOND (e.g., PCDet_DSA+RGCA, PCDet_HDSA, PCDet_RGCA+DSA, and PCDet_HRGCA) built by the four combinations of DSA and RGCA. Table 5.2 shows that RGCA performs better than DSA at the early stage and all hierarchical attention-based networks outperform single-attention networks (e.g., PCDet_DSA and PCDet_RGCA). Specifically, the proposed PCDet_HDSA, PCDet_DSA+RGCA, PCDet_RGCA+DSA, and PCDet_HRGCA respec-

tively improve SECOND by 3.45%, 3.52%, 3.72%, and 3.88%. They respectively improve PCDet_DSA by 0.23%, 0.3%, 0.5%, and 0.66% and improve PCDet_RGCA by 0.17%, 0.24%, 0.44%, and 0.6%. We conclude that a hierarchical attention structure is effective for the 3D object detection task. Comparing the four hierarchical attention-based SECOND networks, we observe that PCDet_HRGCA performs the best and PCDet_HDSA performs the worst in detecting 3D cars at a moderate difficulty level. Replacing the DSA with RGCA at the early stage or the late stage of the network improves the car detection accuracy. For example, PCDet_DSA+RGCA performs better than PCDet_HDSA and PCDet_HRGCA performs better than PCDet_RGCA+DSA. On the other hand, PCDet_DSA+RGCA and PCDet_RGCA+DSA achieve similar 3D car detection accuracy.

Table 5.3 lists the AP of the proposed PCDet_HDSA with a small SECOND network, the proposed PCDet_HDSA with a large SECOND network, the proposed PCDet_HRGCA with a small SECOND network, the proposed PCDet_HRGCA with a large SECOND network, and ten peer one-stage voxel-based 3D object detectors, namely, SECOND with a small network, SECOND with a large network, TANet [40], Voxel-FPN [19], SA-SSD [53], SE-SSD [54], CenterNet3D-SL1 [55], Pointpillars [56], SCNet [57], and AFDet [58], on the KITTI car validation dataset. It shows that PCDet_HDSA with a large SECOND network achieves better car detection results than PCDet_HDSA with a small SECOND network at three difficulty levels. It ranks the best in detecting cars at easy and hard levels and fifth in detecting cars at the moderate level. PCDet_HRGCA with a large SECOND network achieves better car detection results than PCDet_HRGCA with a small SECOND network at moderate and hard difficulty levels. It ranks third in detecting cars at the moderate level. In Table 5.3, we compare the performance of PCDet_HDSA with peer one-stage voxel-based 3D object detectors due to its simple structure and fast speed. We compare the performance of PCDet_HRGCA with peer one-stage voxel-based 3D object detectors due to its good performance at the moderate level. We expect other hierarchical structures on both small and large SECOND networks (e.g., PCDet_DSA+RGCA, and PCDet_RGCA+DSA) will achieve better 3D car detection results than PCDet_HDSA on both small and large SECOND net-

works, and worse 3D car detection results than PCDet_HRGCA on both small and large SECOND networks at the moderate level.

In the following, we will compare the car detection performance of the simple and fast network PCDet_HDSA and several detectors in terms of detection accuracy, detection speed, and ablation studies.

3D Detector	Easy	Moderate	Hard	FPS
SECOND (small network)	85.5	75.04	68.78	40
SECOND (large network)	87.43	76.48	69.1	25
TANet	88.21	77.85	75.62	29
Voxel-FPN	88.27	77.86	75.84	***50***
SA-SSD	***88.75***	***79.79***	74.16	25
SE-SSD	N/A	**85.71**	N/A	32
CenterNet3D-SL1	87.92	76.84	75.74	25
Pointpillars	86.13	77.03	72.43	**62**
SCNet	87.83	77.77	75.97	25
AFDet	85.68	75.57	69.31	N/A
PCDet_HDSA (Proposed, small network)	88.36	78.49	***76.05***	40
PCDet_HDSA (Proposed, large network)	**88.98**	78.77	**77.27**	25
PCDet_HRGCA (Proposed, small network)	88.16	78.92	75.68	38
PCDet_HRGCA (Proposed, large network)	87.98	79.05	75.82	23

Table 5.3: Comparison of AP(%) and FPS of the proposed PCDet_HDSA and PCDet_HRGCA (using a small SECOND network and using a large SECOND network as the backbone respectively) with AP(%) and FPS of ten peer one-stage voxel-based 3D object detectors on cars.

Detection Accuracy Comparison: Table 5.3 shows that SE-SSD, SA-SSD, PCDet_HDSA with a large SECOND network, and PCDet_HDSA with a small SECOND network rank the top four 3D car detectors at the moderate level with detection rates of 85.71%, 79.79%, 78.77%, and 78.49%, respectively. PCDet_HDSA with a large SECOND network has the highest 3D car detection rate of 88.98% and 77.27% for easy and hard levels, respectively. It improves the second best car detectors SA-SSD at the easy level by 0.26% and PCDet_HDSA with a small SECOND network at the hard level by 1.55%. However, SE-SSD has a more complex training process than PCDet_HDSA. Its training process iteratively updates the teacher and student SSDs, while PCDet_HDSA's training process is straightforward. SA-

SSD has a more complex network structure than PCDet_HDSA since it maintains an auxiliary network and involves a partial-sensitive deformation operation. Overall, the proposed PCDet_HDSA with a large SECOND network achieves the best detection accuracy for cars at easy and hard levels and the third best detection accuracy for cars at the moderate level when compared with ten state-of-the-art networks.

Detection Speed Comparison: Table 5.3 shows that Pointpillars, Voxel-FPN, and PCDet_HDSA with a small SECOND network are the three fastest detectors with an inference speed of 62, 50, and 40 FPS, respectively. Pointpillars treat voxels in the same (x, y) coordinates as a whole to accelerate speed. Voxel-FPN uses the multi-scale voxel features fusion module to accelerate speed. However, both lead to information loss, which degrades their detection accuracy. PCDet_HDSA with a small SECOND network improves the detection accuracy of Pointpillars by 2.59%, 1.90%, and 5.00% and Voxel-FPN by 0.10%, 0.81%, and 0.28% for cars at easy, moderate, and hard levels, respectively. Overall, PCDet_HDSA with a small SECOND network is an excellent trade-off between performance and efficiency.

Ablation Studies: In Table 5.3, PCDet_HDSA improves the detection accuracy of its corresponding baseline SECOND. Specifically, PCDet_HDSA with a large SECOND network improves the large SECOND by 1.77%, 2.99%, and 11.82% and PCDet_HDSA with a small SECOND network improves the small SECOND by 3.35%, 4.60%, and 10.57% to detect cars at easy, moderate, and hard levels, respectively. The improvement is mainly achieved by the HDSA model. First, HDSA considers input as low-level features and processes them with convolutional layers to learn high-level features for abstract semantics. Second, it combines input with high-level convolved features to capture both abstract semantics and detailed information to represent an object. PCDet_HDSA and SECOND have a similar network structure and the same settings including a learning rate of 0.003, the Adam one-cycle optimizer, and the loss function of SmoothL_1. However, PCDet_HDSA's network parameters (e.g., 4.5 and 9.6 million respectively for the small and large SECOND network) are 18.4% and 24.0% less than their baseline SECOND (e.g., 5.33 and 11.9 million respectively for the small and large network) due to the removal of 50% of the parameters of

the last 3D convolutional layer. As a result, PCDet_HDSA is a little faster than SECOND.

Network	Easy	Moderate	Hard
VoxelNet	67.17	47.65	45.11
TANet	**85.98**	64.95	60.40
Voxel-FPN	68.77	61.86	56.40
PCDet_HDSA (Proposed, small network)	79.58	***67.28***	***63.35***
PCDet_HDSA (Proposed, large network)	***83.59***	**68.46**	**63.66**

Table 5.4: Comparison of AP (%) of the proposed PCDet_HDSA (using a small SECOND network and using a large SECOND network as the backbone) with AP(%) of three peer one-stage voxel-based 3D object detectors on cyclists.

Table 5.4 lists the AP of PCDet_HDSA with a small SECOND network, PCDet_HDSA with a large SECOND network, and three peer 3D object detectors (e.g., VoxelNet [13], TANet [40], and Voxel-FPN [19]) on the KITTI cyclist validation dataset. Since seven peer detectors listed in Table 5.3 do not provide the cyclist AP on the KITTI validation dataset, we only compare the cyclist detection results of three peer systems in Table 5.4. This table shows that PCDet_HDSA with a large SECOND network achieves the best performance on cyclists at moderate and hard levels and the second best performance at the easy level.

In summary, our extensive experimental results as shown in Table 5.1, and Table 5.2 demonstrate the following:

1. applying attention to a network tends to lead a better detection accuracy than a network without attention;

2. concatenating the original input with the attention-based weighted features tends to lead a better detection accuracy than directly using the attention-based weighted features;

3. RGCA tends to achieve better detection results than DSA.

4. hierarchically applying attention mechanisms at the early and late stages of a network leads to a better detection accuracy than applying attention at early or late stages of a network;

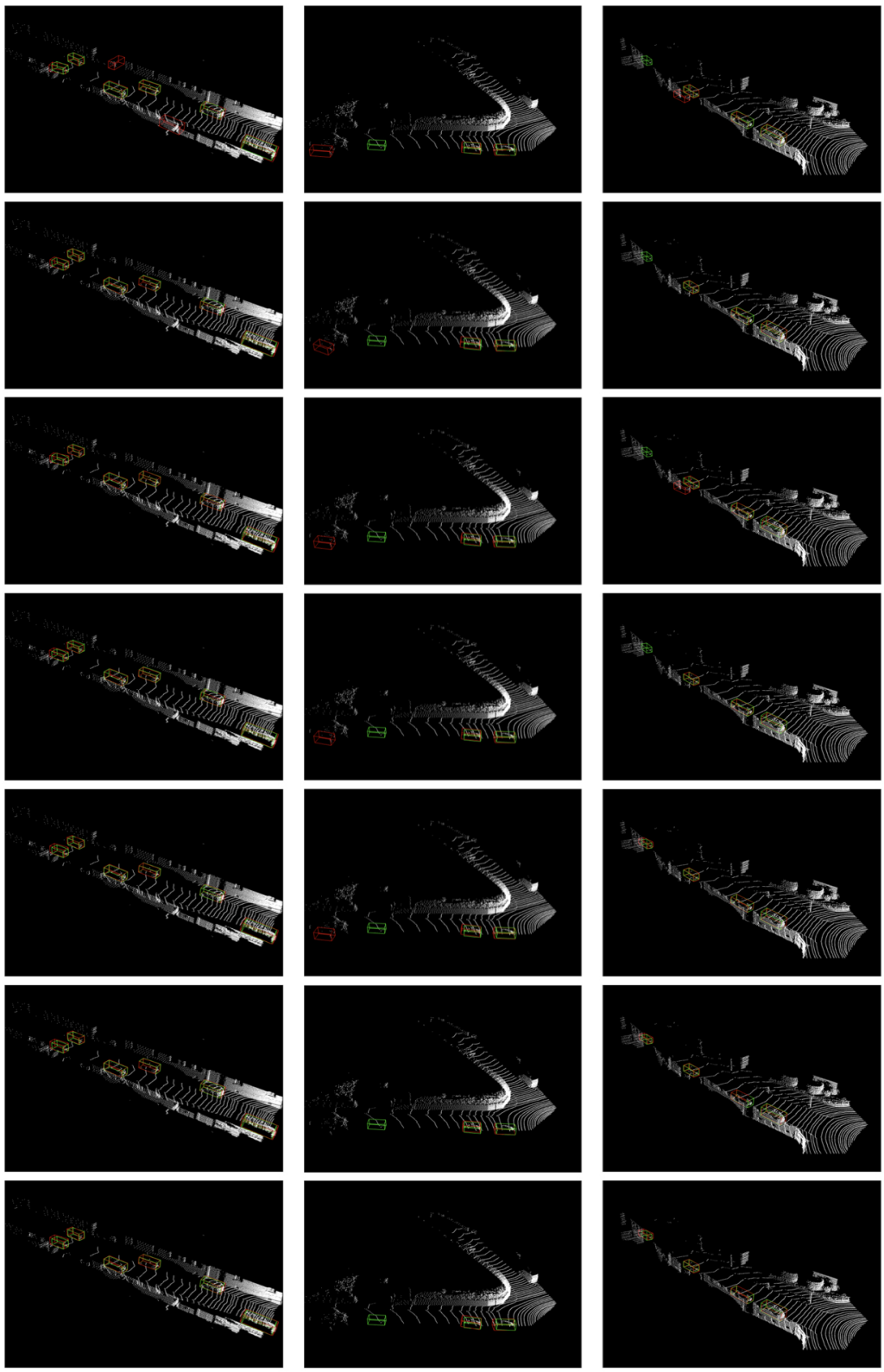

Fig. 5.1: Three sample car detection results of baseline and six proposed networks (from top to bottom): SECOND, PCDet_DSA variant 1, PCDet_DSA variant 2, PCDet_DSA, PCDet_RGCA, PCDet_HDSA and PCDet_HRGCA.

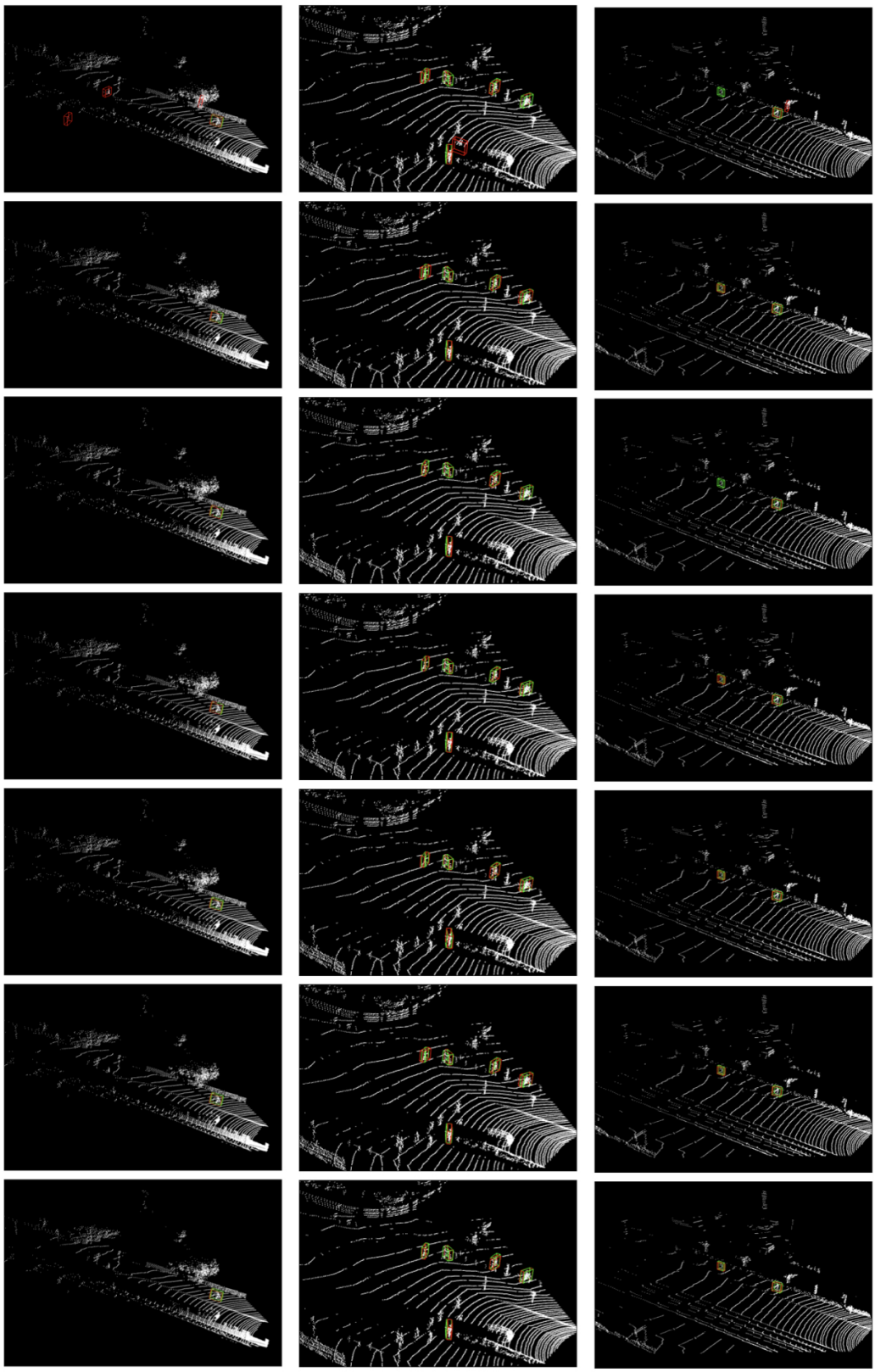

Fig. 5.2: Three sample cyclist detection results of baseline and six proposed networks (from top to bottom): SECOND, PCDet_DSA variant 1, PCDet_DSA variant 2, PCDet_DSA, PCDet_RGCA, PCDet_HDSA, and PCDet_HRGCA.

Qualitative Results: Figure 5.1 and Figure 5.2 demonstrate three sample 3D car detection results and three sample 3D cyclist detection results of the baseline (SECOND), PCDet_DSA Variant 1, PCDet_DSA Variant 2, PCDet_DSA, PCDet_RGCA, PCDet_HDSA, and PCDet_HRGCA, respectively, where ground truths are shown in green bounding boxes and predicted results are shown in red bounding boxes. In Figure 5.1, the first column presents a scenario in which SECOND detects eight cars with two of them being a false positive. PCDet_DSA, its two variants, PCDet_RGCA, PCDet_HDSA, and PCDet_HRGCA accurately detect six true cars without any false positives. The second column presents a scenario in which SECOND, PCDet_DSA, its two variants, and PCDet_RGCA detect two true cars and one false positive car and fail to detect one true car. PCDet_HDSA and PCDet_HRGCA obtain the same predicted results as SECOND except that they do not have false positives. The last column presents a scenario in which SECOND and PCDet_DSA's Variant 2 detect three true cars and one false positive car and fail to detect one true car. PCDet_DSA and its first variant obtain the same detection results as SECOND except that they do not have false positives. PCDet_RGCA, PCDet_HDSA, and PCDet_HRGCA correctly detect four true cars only. In general, PCDet_HRGCA has the best performance among all the compared methods, which matches our previous quantitative results.

In Figure 5.2, the first column shows that SECOND detects four cyclists, while three of them are false positives. PCDet_DSA, its two variants, PCDet_RGCA, PCDet_HDSA, and PCDet_HRGCA detect one cyclist target object precisely. The second column shows that SECOND detects six cyclists, while one of them is a false positive. PCDet_DSA, its two variants, PCDet_RGCA, PCDet_HDSA, and PCDet_HRGCA precisely detect five cyclist target objects. The last column shows that SECOND detects one true cyclist and one false positive cyclist and fails to detect one true cyclist. PCDet_DSA variant 2 obtains the same detection results as SECOND except that it does not have a false positive. PCDet_DSA Variant 1, PCDet_DSA, PCDet_RGCA, PCDet_HDSA, and PCDet_HRGCA correctly detect two true cyclists only. In general, Figure 5.2 shows the proposed single attention-based networks and the proposed hierarchical attention-based networks achieve great 3D cyclist

variant	attention	driv.	ped	walkway	stop	carpark	divider	mIoU
A		80.5	54.0	58.2	47.7	52.3	51.3	57.3
B	N-L	80.4	53.0	57.8	46.5	52.5	50.6	56.8
C	SE	81.0	54.0	58.7	48.2	50.0	51.8	57.3
D	GCNet	80.9	54.5	58.5	48.3	50.7	51.9	57.5
E	TA	81.4	54.8	59.4	49.1	50.7	52.5	58.0
F	DSA	81.5	56.2	60.1	51.1	51.5	**54.0**	59.0
G	RGCA	**81.7**	**57.1**	**60.5**	**51.7**	**53.8**	53.5	**59.7**

Table 5.5: Comparison of map segmentation results of different single attention-based networks.

detection performance, while the baseline SECOND achieves moderate 3D cyclist detection accuracy.

In general, the qualitative results show that the proposed PCDet_DSA, its two variants, and the proposed PCDet_RGCA outperform their baseline SECOND in detecting 3D cars and cyclists. Furthermore, the hierarchical attention-based networks PCDet_HDSA and PCDet_HRGCA outperform single attention-based networks including PCDet_DSA and PCDet_RGCA in detecting 3D cars and cyclists.

5.3.2 Map Segmentation Results

In this subsection, we evaluate the proposed map segmentation network BEVSeg on the NuScenes dataset [1].

Single Attention Comparison: Table 5.5 compares the performance of the baseline, the two proposed attention modules DSA and RGCA, and four commonly used attention modules in terms of IoU in each category and mIoU. Variant A is the baseline network, i.e., BEVDet [20] network incorporated with the proposed SH and ABA modules. Variants B, C, D, and E are the baseline network adding the commonly used attentions including N-L [31], SE [27], GCNet [51], and TA [52], respectively. Variants F and G are the baseline network that is added with the proposed attention mechanisms DSA and RGCA, respectively. In order to have a fair comparison, all the attention modules are added at the early stage. Table 5.5 shows that variant B decreases the baseline network by 0.5% in mIoU. Variant C and the baseline network have the same segmentation accuracy in terms of mIoU. Although

variant	augment	attention	driv.	ped	walkway	stop	carpark	divider	mean
A			67.0	32.8	36.9	29.1	31.9	31.2	38.2
B	✓		80.5	54.0	58.2	47.7	52.3	51.3	57.3
C	✓	DSA	81.5	56.2	60.1	51.1	51.5	54.0	59.0
D	✓	RGCA (4 × 4 nodes)	81.4	56.1	59.6	50.8	53.7	53.0	59.1
E	✓	RGCA	81.7	57.1	60.5	51.7	53.8	53.5	59.7
F	✓	HDSA	81.9	57.2	60.7	52.9	**54.5**	54.1	60.2
G	✓	RGCA + DSA	82.0	58.1	61.0	53.4	53.1	54.1	60.3
H	✓	DSA + RGCA	82.8	57.6	61.2	52.9	**54.5**	54.0	60.5
I	✓	HRGCA	**83.4**	**58.7**	**62.6**	**54.5**	51.4	**55.2**	**61.0**

Table 5.6: Comparison of map segmentation results of different hierarchical attention-based networks.

N-L and SE modules have great performance in the single-view image domain, they do not seem to be suited to extract information in the BEV domain. Variants D and E respectively improve the baseline network by 0.2% and 0.7% in mIoU. However, our proposed DSA and RGCA modules (variant F and variant G) significantly outperform the baseline network with higher mIoUs of 1.7% and 2.4%, respectively. In general, our proposed attention modules perform better than current commonly used attention modules.

Hierarchical Attention Comparison: Table 5.6 compares the performance of the baseline, the baseline with the ABA module, the two proposed attention modules DSA and RGCA with the ABA module, and four hierarchical attention modules (e.g., HDSA, DSA+RGCA, RGCA+DSA, and HRGCA) with the ABA module in terms of IoU in each category and mIoU. Variant A is the baseline network, i.e., BEVDet [20] network incorporated with the proposed SH module. Variant B is the baseline network adding the proposed SH and ABA modules. Variants C, D, and E are variant B adding the proposed attentions DSA, RGCA with 16 graph nodes, and RGCA with 64 graph nodes (default). The feature size in RGCA needs to be divisible by the node number in RGCA. Since the feature size is 128 × 128, the node number could be 1 × 1, 2 × 2, 4 × 4, and 8 × 8. We choose 4 × 4 and 8 × 8 nodes to obtain more relational information in more detail. Variants F, G, H, and I are variant B adding the hierarchical attention module built by four combinations of DSA and RGCA including HDSA, RGCA+DSA, DSA+RGCA, and HRGCA, respectively. We observe the following from Table 5.6:

1. ABA is the most contributing module since augmented geometry information improves

the variety of ground truth segmentation maps and avoids the overfitting issue, which leads to the significant performance boost of a higher mIoU of 19.1% from variant A to variant B.

2. DSA helps to find the inherent structure of augmented geometry information to highlight relevant regions to gather more usable information, which leads to a segmentation accuracy improvement of a higher mIoU of 1.7% and 1.2% from variant B to variant C and from variant C to variant F, respectively.

3. The graph attention structure RGCA works better than DSA by helping to gather useful information at both textural and semantic levels, which leads to a higher mIoU of 0.7% from variant C to variant E.

4. Increasing the graph node numbers provides more relational information, which leads to a better segmentation accuracy of a higher mIoU of 0.6% from variant D to variant E.

5. The hierarchical homogenous attention (i.e., the same attention applied at the early and the late stages) helps to get more high-level semantic information, which leads to an improved segmentation accuracy of 1.2% from variant C to variant F and an improved segmentation accuracy of 0.8% from variant E to variant I.

6. The hierarchical heterogeneous attention (i.e., one attention applied at the early stage and another attention applied at the late stage) helps to get more high-level semantic information, which leads to an improved segmentation accuracy of 0.6% and 1.5% from variant E to variant G and from variant C to variant H, respectively.

In summary, the proposed hierarchical attention-based networks achieve better segmentation accuracy than the proposed single attention-based networks. Specifically, among the four hierarchical attention-based networks, BEVSeg_HRGCA network (variant I) achieves the best segmentation accuracy of mIoU of 61.0% and BEVSeg_HDSA network (variant F) achieves the worst segmentation accuracy of mIoU of 60.2%. Among the single attention-based networks, BEVSeg_RGCA improves BEVSeg_DSA by a higher mIoU of 0.7% due to

Networks	Driv.	Ped	Walkway	Stop	Carpark	Divider	mIoU
OFT [59]	74.0	35.3	45.9	27.5	35.9	33.9	42.1
LSS [22]	75.4	38.8	46.3	30.3	39.1	36.5	44.4
CVT [43]	74.3	36.8	39.9	25.8	35.0	29.4	40.2
M^{2}BEV [42]	77.2	-	-	-	-	40.5	-
BEVFormer [3]	80.7	-	-	-	-	21.3	-
BEVFusion [4]	81.7	54.8	58.4	47.4	50.7	46.4	56.6
BEVSeg_HDSA (Ours)	81.9	57.2	60.7	52.9	**54.5**	54.1	60.2
BEVSeg_DSA+RGCA (Ours)	82.8	57.6	61.2	52.9	54.5	54.0	60.5
BEVSeg_RGCA+DSA (Ours)	82.0	58.1	61.0	53.4	53.1	54.1	60.3
BEVSeg_HRGCA (ours)	**83.4**	**58.7**	**62.6**	**54.5**	51.4	**55.2**	**61.0**

Table 5.7: Comparison of segmentation results of ten state-of-the-art methods for six classes on the nuScenes in terms of IoU.

its superior ability to extract useful semantic information and capture global relationships between different regions at different scales.

We compare the proposed BEVSeg_HDSA, BEVSeg_DSA+RGCA, BEVSeg_RGCA+DSA, and BEVSeg_HRGCA methods with six state-of-the-art BEV segmentation methods including Orthographic Feature Transform (OFT) [59], LSS [22], CVT [43], Multi-Camera Joint 3D Detection and Segmentation with Unified Bird's-Eye View Representation (M^{2}BEV) [42], BEVFormer [3], and BEVFusion [4]. To the best of our knowledge, BEVFusion [4] achieves the best BEV segmentation performance in terms of mIoU. Table 5.7 lists the IoU scores of all ten compared methods for each of the six semantic categories and mIoU for all six categories. However, M^{2}BEV and BEVFormer only report their segmentation results on drivable roads and dividers. To achieve a fair comparison, we report segmentation results of the multi-task network (e.g. M^{2}BEV) and report segmentation results of the temporal-model-based network (e.g., BEVFormer) at a single timestamp. We also report segmentation results of the multi-sensor-model-based network (e.g., BEVFusion) using images captured by cameras and report segmentation results of some early methods like OFT, LSS, and CVT on the nuScene dataset by copying IoU values from the published results in [4]. Table 5.7 clearly shows that BEVSeg_HRGCA and BEVSeg_HDSA improve the mIoU of BEVFusion by 4.4% and 3.6% for six semantic classes, respectively. Specifically, BEVSeg_HRGCA achieves a higher IoU of at least 1.7%, 3.9%, 4.2%, 7.1%, 0.7%, and 8.8% than BEVFusion for semantic classes of drivable area, ped-crossing, walkway, stop-line, carpark-area,

and divider, respectively. BEVSeg_HDSA achieves a higher IoU of 0.2%, 2.4%, 2.3%, 5.5%, 3.8%, and 7.7% than BEVFusion on drivable area, ped crossing, walkway, stop line, carpark area, and divider, respectively. We conclude that BEVSeg_HRGCA outperforms the peer methods on all six semantic classes containing large or small regions and achieves a higher mIoU of 0.8% than BEVSeg_HDSA for all six semantic classes. BEVSeg_HDSA also performs better than the leading method, BEVFusion, especially when the category contains small and delicate regions. In summary, both BEVSeg_HRGCA and BEVSeg_HDSA deliver cutting-edge accuracy across most semantic classes, while BEVSeg_HRGCA achieves the best performance.

Qualitative Results: Figure 5.3 presents four sample scenes in the daytime along with their ground truth and predicted segmentation results for six categories. Each scene has six multi-view input images, where the first two rows present input images from front and back views in the directions of left, center, and right. The third row presents the ground truth for six categories including drivable area, ped crossing, walkway, stop line, parking area, and divider, as shown from the left to the right. The fourth row to the eighth row respectively present the predicted segmentation results of the baseline (BEVDet with SH only), BEVSeg_DSA, BEVSeg_RGCA, BEVSeg_HDSA, and BEVSeg_HRGCA for the six categories in the same order. Figure 5.3 (A) and (B) demonstrate all the compared methods perform well when the vehicle drives along the road in the daytime. Figure 5.3 (C) and (D) show that the baseline (no ABA module) performs much worse than other compared methods with the ABA module when the vehicle makes a turn in the daytime.

Figure 5.4 presents four sample scenes at night along with their ground truth and predicted segmentation results for six categories. Its layout is the same as the layout of Figure 5.3. Figure 5.4 (A), (B), (C), and (D) show that no methods perform well at night and the baseline (no ABA module) performs the worst. In general, the night scenes are difficult to segment due to their dark background.

To demonstrate the effectiveness of the ABA module, we present the ground truths of two scenes and the augmented ground truth in Figure 5.5 and demonstrate segmentation

results of baseline (BEVDet with SH only) and baseline adding the ABA module for one scene from the side-forward view in Figure 5.6. In Figure 5.5, we show the front-view input image in the first row, the side-view input image in the second row, and the augmented ground truth results in the third row. We observe that augmented ground truth results are significantly different from the two ground truths. This indicates that our ABA module is able to generate a wider range of "ground truth" images in the same domain and help the network learn the side-forward images more effectively. In Figure 5.6, we show the ground truth of six semantic classes on the 1st row and their corresponding segmentation results of the baseline and the ABA-enabled baseline on the next two rows. We observe that the ABA-enabled baseline produces segmentation results that have a higher similarity to the ground truth than the baseline without the ABA module, which seems to produce distorted segmentation results. Both the augmented ground truth results in Figure 5.5 and the segmentation results in Figure 5.6 produced by the ABA-enabled network show ABA plays an important role in improving the network's performance on side-forward scenes.

Figure 5.7 presents the qualitative segmentation results of a traditional deep CNN-based segmentation network without the proposed HRGCA module (i.e., BEVDet with ABA and SH modules) and the proposed BEVSeg_HRGCA network on one sample scene. We observe that the traditional deep CNN-based segmentation network cannot accurately segment the border of a delicate region as circled in red and the BEVSeg_HRGCA network produces more accurate segmentation results due to its accurate estimation of global contextual relationships.

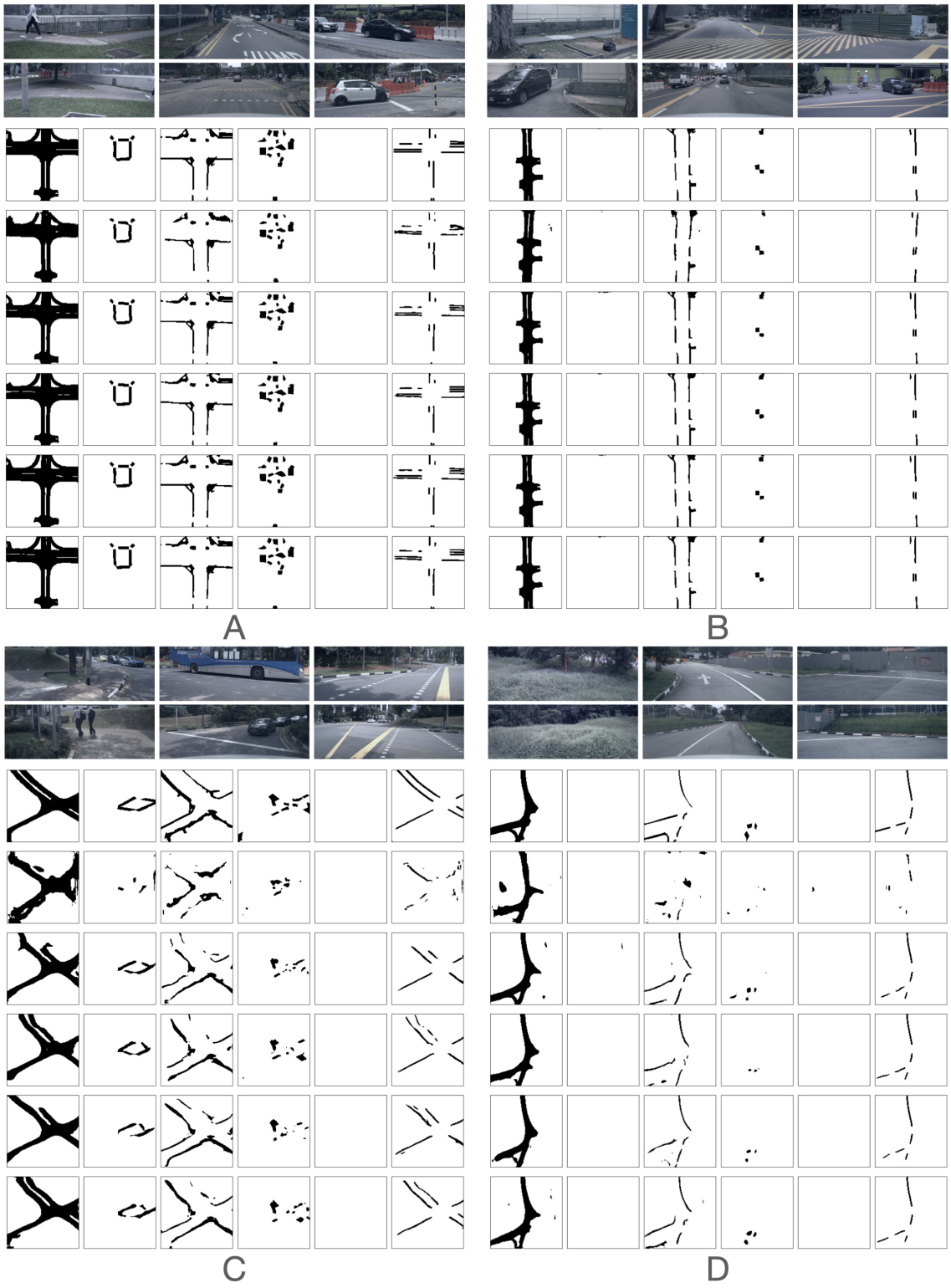

Fig. 5.3: Illustration of four sample scenes in the daytime along with their ground truth and predicted segmentation results for six categories. For each scene, the first two rows present multi-view input images, the third row presents the ground truth for six categories, and the fourth to the eighth row respectively presents the predicted segmentation results of the baseline (variant A in Table 5.6), BEVSeg_DSA, BEVSeg_RGCA, BEVSeg_HDSA, and BEVSeg_HRGCA for six categories.

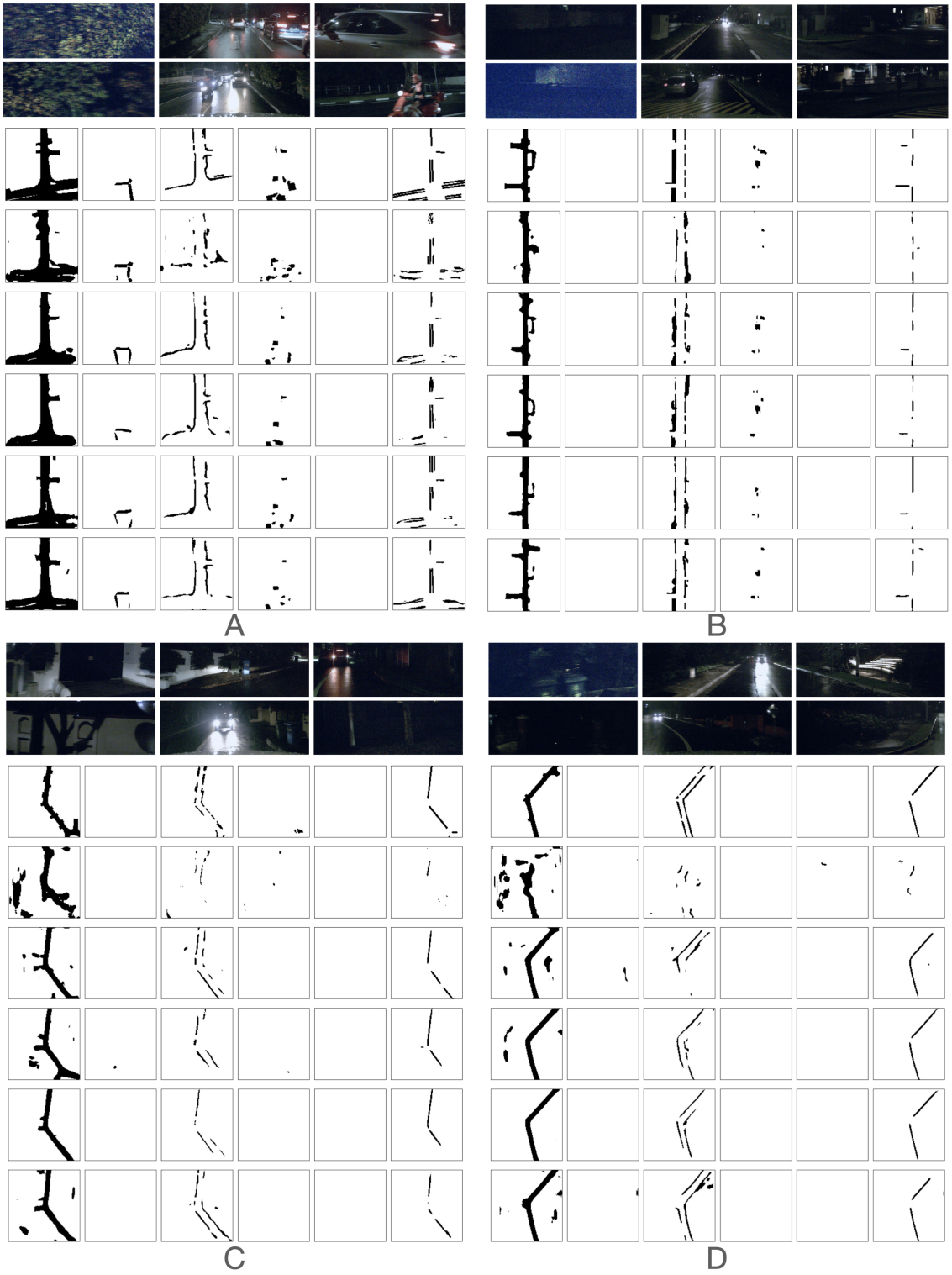

Fig. 5.4: Illustration of four sample scenes at night along with their ground truth and predicted segmentation results for six categories. For each scene, the first two rows present multi-view input images, the third row presents the ground truth for six categories, and the fourth to the eighth row respectively presents the predicted segmentation results of the baseline (variant A in Table 5.6), BEVSeg_DSA, BEVSeg_RGCA, BEVSeg_HDSA, and BEVSeg_HRGCA for six categories.

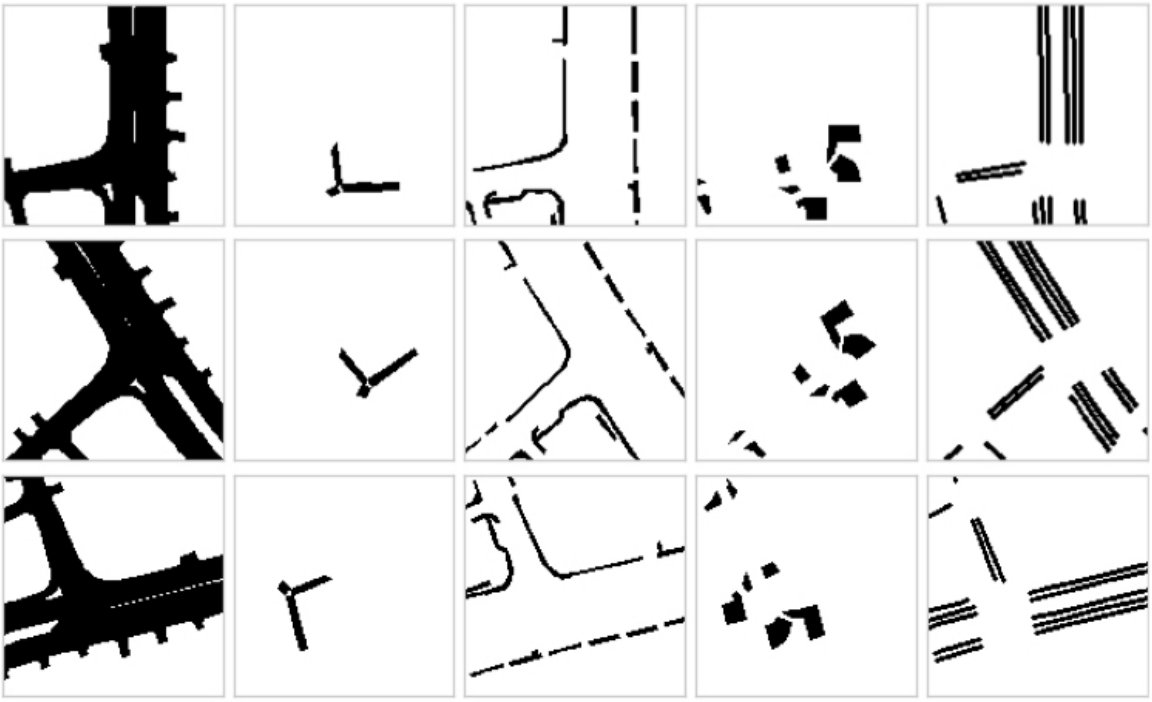

Fig. 5.5: Illustration of two BEV map ground truth (the first two rows) and one augmented BEV map ground truth generated by the ABA module (the last row)

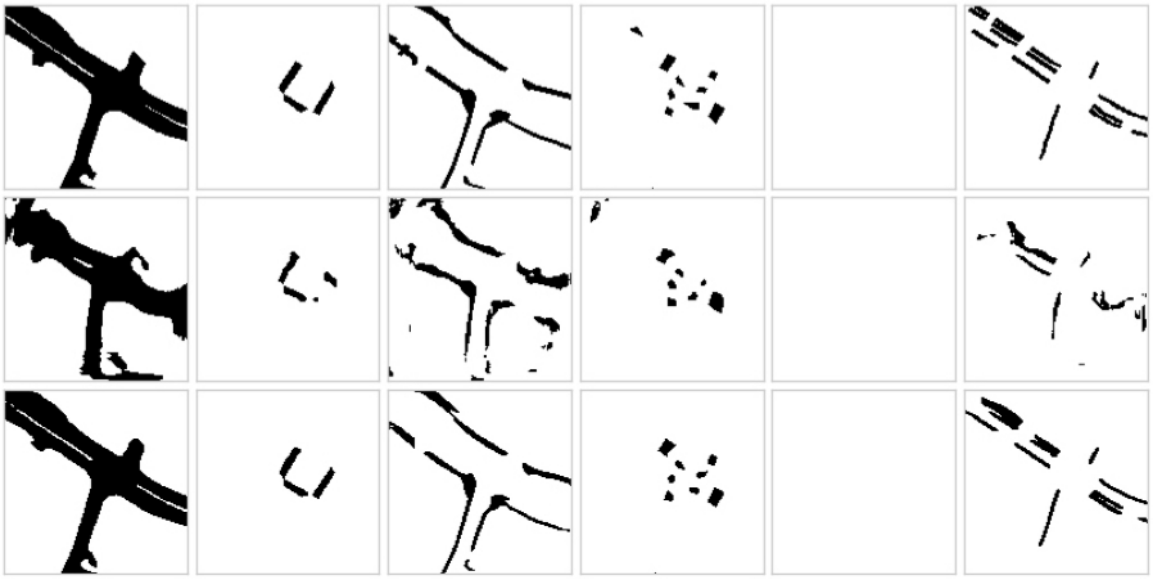

Fig. 5.6: Illustration of the ground truth of one scene and the segmentation results of baseline (BEVDet with SH) and variant B (baseline with ABA) shown from the top to the bottom.

Fig. 5.7: Illustration of one sample scene of six views (first two rows), segmentation result of BEVDet with ABA and SH modules (third row), segmentation results of the proposed BEVSeg_HRGCA network (fourth row), and the ground-truth (fifth row). The inaccurate segmentation results are circled in red.

CHAPTER 6

CONCLUSIONS

In this dissertation, we introduce two perception networks. One of these networks is for 3D object detection utilizing LiDAR sensors, and the other one is for map segmentation utilizing camera sensors. We compare their performance with state-of-the-art methods. Specifically, we summarize the strategy and performance of each proposed method as follows:

- We propose a PCDet network for 3D object detection utilizing LiDAR sensors with two hierarchical attention modules, i.e., HDSA and HRGCA. PCDet_HDSA incorporates multi-resolution features, focuses on the important locations, and filters out the irrelevant parts to improve detection accuracy. HDSA improves the baseline network and achieves similar accuracy and inference speed compared with one-stage state-of-the-art systems on the KITTI dataset. PCDet_HRGCA incorporates multi-resolution features and captures global semantic relational features to improve detection accuracy. PCDet_HRGCA improves the baseline network and achieves better performance than PCDet_HSA on the KITTI dataset.

- We propose a novel BEVSeg network for map segmentation utilizing camera sensors with two hierarchical attention modules, i.e., HDSA and HRGCA. BEVSeg incorporates ABA to augment the BEV feature map and the segmentation ground truths correspondingly, which solves the overfitting issue. BEVSeg_HDSA enlarges the receptive field in spatial attention to get high-semantic information in different scales. BEVSeg_HRGCA combines graph and coordinate information in the deep CNNs to effectively estimate global contextual relationships. Specifically, the RGCA module consists of the spatial graph to extract spatial information between nodes and the channel graph to extract channel information within each node. BEVSeg_HDSA and

BEVSeg_HRGCA networks outperform six state-of-the-art methods on the nuScenes dataset.

The contributions of the PCDet network include:

- Proposing a 3D object framework PCDet that can easily incorporate different attention modules at different stages of the DNN to capture features at multiple scales and improve the detection accuracy of the SECOND network.
- Utilizing the features generated from the HDSA module to build PCDet_HDSA to learn and find the most important locations to focus on and filter out the irrelevant parts of the input point cloud.
- Incorporating the HRGCA module that contains both graph and coordinate information in the deep CNNs to build PCDet_HRGCA to not only effectively acquire the global information but also efficiently estimate contextual relationships of the global information in the 3D point cloud domain.
- Incorporating DSA and RGCA at either the early stage or the late stage to build PCDet_DSA+RGCA and PCDet_RGCA+DSA to respectively capture multi-scale high-semantic and fine-grained features and estimate global semantic relational characteristics to improve the detection accuracy.
- Improving the baseline network and achieving similar accuracy and inference speed compared with one-stage state-of-the-art systems on the KITTI validation dataset.

The contributions of the BEVSeg network include:

- Proposing a new network architecture BEVSeg to perform semantic segmentation of a scene with multi-view images and achieve state-of-the-art results.
- Incorporating ABA in the geometry module to augment the coherent BEV map, align the augmented object and segmentation ground truths, and align the augmented BEV map and its augmented ground truths to address overfitting and misalignment issues.

- Extending the SH to individually process each semantic category to address the possible overlapping among semantic categories.
- Incorporating low-complexity HDSA in the data-driven module to build BEVSeg_HDSA to learn multi-scale BEV features flexibly by enlarging the feature receptive field and learning interest regions.
- Incorporating the HRGCA module in the data-driven module to build BEVSeg_HRGCA to gather the global semantic relationship from different scales.
- Incorporating DSA and RGCA in the data-driven module at either the early stage or the late stage to build BEVSeg_DSA+RGCA and BEVSeg_RGCA+DSA to respectively capture multi-scale high-semantic and fine-grained features and estimate global semantic relational characteristics to improve the segmentation accuracy.
- Improving the baseline network in terms of segmentation accuracy for six major semantic categories.

In the future, we will test the two proposed approaches on bigger autonomous driving datasets and investigate other structures to enhance their generalizability. Additionally, in order to test the proposed HDSA and HRGCA modules' effectiveness and discover fresh ideas for improvement, we will compare them to more commonly used spatial, channel, and graph attention modules.

REFERENCES

[1] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 621–11 631.

[2] Y. Maalej, S. Sorour, A. Abdel-Rahim, and M. Guizani, "Tracking 3d lidar point clouds using extended kalman filters in kitti driving sequences," in *2018 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2018, pp. 1–6.

[3] Z. Li, W. Wang, H. Li, E. Xie, C. Sima, T. Lu, Q. Yu, and J. Dai, "Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers," *arXiv preprint arXiv:2203.17270*, 2022.

[4] Z. Liu, H. Tang, A. Amini, X. Yang, H. Mao, D. Rus, and S. Han, "Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation," *arXiv preprint arXiv:2205.13542*, 2022.

[5] Y. Zhang, Z. Zhu, W. Zheng, J. Huang, G. Huang, J. Zhou, and J. Lu, "Beverse: Unified perception and prediction in birds-eye-view for vision-centric autonomous driving," *arXiv preprint arXiv:2205.09743*, 2022.

[6] D. Z. Wang, I. Posner, and P. Newman, "What could move? finding cars, pedestrians and bicyclists in 3d laser data," in *2012 IEEE International Conference on Robotics and Automation*. IEEE, 2012, pp. 4038–4044.

[7] A. Azim and O. Aycard, "Layer-based supervised classification of moving objects in outdoor dynamic environment using 3d laser scanner," in *2014 IEEE Intelligent Vehicles Symposium Proceedings*. IEEE, 2014, pp. 1408–1414.

[8] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the seventh IEEE international conference on computer vision*, vol. 2. Ieee, 1999, pp. 1150–1157.

[9] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, vol. 1. Ieee, 2005, pp. 886–893.

[10] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, pp. 273–297, 1995.

[11] J. Behley, V. Steinhage, and A. B. Cremers, "Laser-based segment classification using a mixture of bag-of-words," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 4195–4200.

[12] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *Advances in neural information processing systems*, vol. 30, 2017.

[13] Y. Zhou and O. Tuzel, "Voxelnet: End-to-end learning for point cloud based 3d object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4490–4499.

[14] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 10 012–10 022.

[15] Z. Yang, Y. Sun, S. Liu, and J. Jia, "3dssd: Point-based 3d single stage object detector," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 040–11 048.

[16] S. Shi, C. Guo, L. Jiang, Z. Wang, J. Shi, X. Wang, and H. Li, "Pv-rcnn: Point-voxel feature set abstraction for 3d object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 10 529–10 538.

[17] C. R. Qi, W. Liu, C. Wu, H. Su, and L. J. Guibas, "Frustum pointnets for 3d object detection from rgb-d data," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 918–927.

[18] S. Shi, X. Wang, and H. Li, "Pointrcnn: 3d object proposal generation and detection from point cloud," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 770–779.

[19] H. Kuang, B. Wang, J. An, M. Zhang, and Z. Zhang, "Voxel-fpn: Multi-scale voxel feature aggregation for 3d object detection from lidar point clouds," *Sensors*, vol. 20, no. 3, p. 704, 2020.

[20] J. Huang, G. Huang, Z. Zhu, and D. Du, "Bevdet: High-performance multi-camera 3d object detection in bird-eye-view," *arXiv preprint arXiv:2112.11790*, 2021.

[21] Y. Liu, T. Wang, X. Zhang, and J. Sun, "Petr: Position embedding transformation for multi-view 3d object detection," in *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVII*. Springer, 2022, pp. 531–548.

[22] J. Philion and S. Fidler, "Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d," in *European Conference on Computer Vision*. Springer, 2020, pp. 194–210.

[23] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[24] S. Deng, Z. Liang, L. Sun, and K. Jia, "Vista: Boosting 3d object detection via dual cross-view spatial attention," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 8448–8457.

[25] H. Lu, X. Chen, G. Zhang, Q. Zhou, Y. Ma, and Y. Zhao, "Scanet: Spatial-channel attention network for 3d object detection," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 1992–1996.

[26] W. Shi and R. Rajkumar, "Point-gnn: Graph neural network for 3d object detection in a point cloud," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 1711–1719.

[27] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141.

[28] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "Supplementary material for 'eca-net: Efficient channel attention for deep convolutional neural networks," in *Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, IEEE, Seattle, WA, USA*, 2020, pp. 13–19.

[29] H. Zhang, K. Dana, J. Shi, Z. Zhang, X. Wang, A. Tyagi, and A. Agrawal, "Context encoding for semantic segmentation," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2018, pp. 7151–7160.

[30] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *European conference on computer vision*. Springer, 2020, pp. 213–229.

[31] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7794–7803.

[32] K. Gregor, I. Danihelka, A. Graves, D. Rezende, and D. Wierstra, "Draw: A recurrent neural network for image generation," in *International conference on machine learning*. PMLR, 2015, pp. 1462–1471.

[33] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu, "Ccnet: Criss-cross attention for semantic segmentation," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 603–612.

[34] Y. Chen, M. Rohrbach, Z. Yan, Y. Shuicheng, J. Feng, and Y. Kalantidis, "Graph-based global reasoning networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 433–442.

[35] S. Zhang, X. He, and S. Yan, "Latentgnn: Learning efficient non-local relations for visual recognition," in *International Conference on Machine Learning*. PMLR, 2019, pp. 7374–7383.

[36] I. O. Tolstikhin, N. Houlsby, A. Kolesnikov, L. Beyer, X. Zhai, T. Unterthiner, J. Yung, A. Steiner, D. Keysers, J. Uszkoreit *et al.*, "Mlp-mixer: An all-mlp architecture for vision," *Advances in neural information processing systems*, vol. 34, pp. 24 261–24 272, 2021.

[37] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *European conference on computer vision*. Springer, 2020, pp. 213–229.

[38] Y. Yan, Y. Mao, and B. Li, "Second: Sparsely embedded convolutional detection," *Sensors*, vol. 18, no. 10, p. 3337, 2018.

[39] Z. Wang and K. Jia, "Frustum convnet: Sliding frustums to aggregate local point-wise features for amodal 3d object detection," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 1742–1749.

[40] Z. Liu, X. Zhao, T. Huang, R. Hu, Y. Zhou, and X. Bai, "Tanet: Robust 3d object detection from point clouds with triple attention," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 07, 2020, pp. 11 677–11 684.

[41] Y. Wang, V. C. Guizilini, T. Zhang, Y. Wang, H. Zhao, and J. Solomon, "Detr3d: 3d object detection from multi-view images via 3d-to-2d queries," in *Conference on Robot Learning*. PMLR, 2022, pp. 180–191.

[42] E. Xie, Z. Yu, D. Zhou, J. Philion, A. Anandkumar, S. Fidler, P. Luo, and J. M. Alvarez, "Mˆ 2bev: Multi-camera joint 3d detection and segmentation with unified birds-eye view representation," *arXiv preprint arXiv:2204.05088*, 2022.

[43] B. Zhou and P. Krähenbühl, "Cross-view transformers for real-time map-view semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 13 760–13 769.

[44] R. Xu, Z. Tu, H. Xiang, W. Shao, B. Zhou, and J. Ma, "Cobevt: Cooperative bird's eye view semantic segmentation with sparse transformers," *arXiv preprint arXiv:2207.02202*, 2022.

[45] B. Pan, J. Sun, H. Y. T. Leung, A. Andonian, and B. Zhou, "Cross-view semantic segmentation for sensing surroundings," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 4867–4873, 2020.

[46] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, and X. Tang, "Residual attention network for image classification," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017, pp. 3156–3164.

[47] O. Oktay, J. Schlemper, L. L. Folgoc, M. Lee, M. Heinrich, K. Misawa, K. Mori, S. McDonagh, N. Y. Hammerla, B. Kainz *et al.*, "Attention u-net: Learning where to look for the pancreas," *arXiv preprint arXiv:1804.03999*, 2018.

[48] D. Eberly, "Euler angle formulas," *Geometric Tools, LLC, Technical Report*, pp. 1–18, 2008.

[49] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

[50] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 3354–3361.

[51] Y. Cao, J. Xu, S. Lin, F. Wei, and H. Hu, "Gcnet: Non-local networks meet squeeze-excitation networks and beyond," in *Proceedings of the IEEE/CVF international conference on computer vision workshops*, 2019, pp. 0–0.

[52] D. Misra, T. Nalamada, A. U. Arasanipalai, and Q. Hou, "Rotate to attend: Convolutional triplet attention module," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2021, pp. 3139–3148.

[53] C. He, H. Zeng, J. Huang, X.-S. Hua, and L. Zhang, "Structure aware single-stage 3d object detection from point cloud," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11 873–11 882.

[54] W. Zheng, W. Tang, L. Jiang, and C.-W. Fu, "Se-ssd: Self-ensembling single-stage object detector from point cloud," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2021, pp. 14 494–14 503.

[55] G. Wang, J. Wu, B. Tian, S. Teng, L. Chen, and D. Cao, "Centernet3d: An anchor free object detector for point cloud," *arXiv preprint arXiv:2007.07214*, 2020.

[56] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "Pointpillars: Fast encoders for object detection from point clouds," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 12 697–12 705.

[57] Z. Wang, H. Fu, L. Wang, L. Xiao, and B. Dai, "Scnet: Subdivision coding network for object detection based on 3d point cloud," *IEEE Access*, vol. 7, pp. 120 449–120 462, 2019.

[58] R. Ge, Z. Ding, Y. Hu, Y. Wang, S. Chen, L. Huang, and Y. Li, "Afdet: Anchor free one stage 3d object detection," *arXiv preprint arXiv:2006.12671*, 2020.

[59] T. Roddick, A. Kendall, and R. Cipolla, "Orthographic feature transform for monocular 3d object detection," *arXiv preprint arXiv:1811.08188*, 2018.

CURRICULUM VITAE

Qiuxiao Chen

Education

- Ph.D., Computer Science, Utah State University, Logan, Utah, US, Adviser: Dr. Xiaojun Qi, October 2023.
- B.S., Electrical Engineering, Zhejiang University, Zhejiang, China. June 2018.

Research Interests

- Computer Vision
- Deep Learning
- Autonomous Driving

Published Conference Papers

- Qiuxiao Chen, Hung-Shuo Tai, Pengfei Li, Ke Wang, Xiaojun Qi. BEVSeg: Geometry and Data-Driven based Multi-View Segmentation in Bird's-Eye-View. In the 14th International Conference on Computer Vision Systems (ICVS 2023), 2023
- Chen, Qiuxiao, Xiaojun Qi, and Ziqi Song. Real-time Hierarchical Soft Attention-based 3D Object Detection in Point Clouds. In 2022 26th International Conference on Pattern Recognition (ICPR). IEEE, 2022.
- Qiuxiao Chen, Pengfei Li, Meng Xu, Xiaojun Qi. Sparse Activation Maps for Interpreting 3D Object Detection. Safe Artificial Intelligence for Automated Driving (Computer Vision and Pattern Recognition Workshop), 2021, Best Paper Finalist

- Meng Xu, Kuan Huang, Qiuxiao Chen, Xiaojun Qi. MSSA-Net: Multi-Scale Self-Attention Network for Breast Ultrasound Image Segmentation. International Symposium on Biomedical Imaging (ISBI), 2021, Oral