RESEARCH ON SEMANTIC BIRD-EYE-VIEW MAP PREDICTION FOR AUTONOMOUS DRIVING

GAO SHUANG

PhD

The Hong Kong Polytechnic University

This programme is jointly offered by

The Hong Kong Polytechnic University and

Harbin Institute of Technology

2025

The Hong Kong Polytechnic University

Department of Mechanical Engineering

Harbin Institute of Technology

Department of Control Science and Engineering

Research on Semantic Bird-Eye-View Map Prediction for Autonomous Driving

Gao Shuang

A thesis submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy

January 2025

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

_____ (Signed)

Gao Shuang (Name of student)

Abstract

Semantic Bird Eye View (BEV) map prediction is a widely utilized environmental perception technique in autonomous driving, converting forward-facing images captured by on-board cameras into top-down two-dimensional images. The semantic BEV map not only encompasses spatial positional information of the driving scene but also includes semantic information of different objects within the scene. This top-down representation facilitates the fusion of multiple sensor data, eliminates perspective distortions caused by camera imaging process, and intuitively conveys the spatial relationships between the autonomous vehicle and surrounding obstacles. This provides a solid perceptual foundation for downstream tasks in autonomous driving. Additionally, the embedded semantic information aids autonomous vehicles in achieving a high-level understanding of their environment. Due to its numerous advantages, semantic BEV map prediction has become a primary research focus in environmental perception.

The prevailing approach for generating semantic BEV map relies on deep learning models. This method is fundamentally data-driven, with its performance significantly influenced by the quality of the training datasets. However, generating semantic BEV map involves view transformation, making it difficult to label corresponding semantic BEV labels. Manual annotation can introduce significant

errors, leading to insufficient labeled samples and decreased label quality. The issue related to training samples present significant challenges for the prediction of semantic BEV map. Additionally, the limited field of view (FOV) of cameras constrains the expression of environmental information in semantic BEV map. This dissertation focuses on the prediction of semantic BEV map, exploring semi-supervised learning methods to ensure segmentation accuracy while reducing the dependency on labeled samples during training. To tackle the limitation in information expression, the dissertation explores sequential image fusion algorithms, using historical observations to enhance the information expression capability of semantic BEV map. The main research contributions are as follows:

To reduce the dependence of the semantic BEV map prediction network on labeled samples, this dissertation proposes a semi-supervised semantic BEV map prediction network based on contrastive learning. This network innovatively integrates view transformation with contrastive learning, avoiding the usage of complex data augmentation in traditional contrastive learning networks. The proposed network is capable of end-to-end training with both labeled and unlabeled samples, resulting in accurate and stable semantic BEV map.

To address the issue of limited environmental observation due to the restricted FOV of cameras, this dissertation proposes a full-view semantic BEV map prediction network based on equidistant sequence fusion. This network utilizes equidistant image sequences to expand the observation range of the environment. The proposed network is capable of generating accurate and clear semantic BEV map in full view with an explainable view transformation module.

To provide autonomous driving systems with perception of future scenes, this dissertation proposes a short-term future semantic BEV map prediction network

based on long short-term memory (LSTM). By predicting future scenes, this network enhances the perception system's ability to provide early warnings for autonomous vehicles, enabling timely reactions to the changes in driving environment.

Publications arising from the thesis

[1] **Shuang Gao**, Qiang Wang, Yuxiang Sun. S2G2: Semi-supervised semantic bird-eye-view grid-map generation using a monocular camera for autonomous driving[J]. *IEEE Robotics and Automation Letters*, 2022, 7.4: 11974–11981.

[2] **Shuang Gao**, Qiang Wang, and Yuxiang Sun. Boundary-aware Semantic Bird-Eye-View Map Generation based on Conditional Diffusion Models[J]. *IEEE Transactions on Circuits and Systems for Video Technology*, 2025.

[3] **Shuang Gao**, Qiang Wang, Yuxiang Sun. Seq-BEV: Semantic Bird-Eye-View Map Generation in Full View using Sequential Images for Autonomous Driving[J]. *IEEE Transactions on Intelligent Transportation Systems*, 2025.

[4] **Shuang Gao**, Qiang Wang, Yuxiang Sun. Obstacle-sensitive Semantic Bird-Eye-View Map Generation with Boundary-aware Loss for Autonomous driving[C]. *IEEE Intelligent Vehicles Symposium (IV)*, pp. 466–471, IEEE, 2024.

[5] **Shuang Gao**, Qiang Wang, David Navarro-Alarcon, Yuxiang Sun. Forecasting Semantic Bird-Eye-View Maps for Autonomous Driving[C]. *IEEE Intelligent Vehicles Symposium (IV)*, pp. 509–514, IEEE, 2024.

Acknowledgements

I would like to express my sincere gratitude to everyone who has supported and guided me throughout the journey of completing this dissertation. The advice, encouragement, and insights offered by mentors and colleagues have been invaluable, providing me with the direction and confidence needed to pursue this research. I am deeply thankful for the collective wisdom and support that have helped shape this work.

To begin with, I am deeply thankful to my supervisors, Dr. Yuxiang Sun and Dr. David NAVARRO-ALARCON for granting me the opportunity to join their labs and for their unwavering support throughout this research journey. Their continuous scientific guidance has been a source of inspiration, driving me to pursue my research with dedication and to accomplish my goals. I greatly appreciate their generous contributions of time, ideas, and funding, which made my Ph.D. experience both productive and enriching. I consider myself truly fortunate to have had the chance to learn under their mentorship.

I would also like to extend my heartfelt thanks to my partner chief supervisor, Prof. Qiang Wang, for his generous help and support. His insightful feedback and advice were invaluable, providing the guidance I needed at crucial moments.

I thank all group members for their kind support and assistance throughout my

time at PolyU. Your companionship makes my Ph.D. journey both enjoyable and unforgettable.

I am profoundly grateful to my family and friends for their love, patience, and understanding during this process. Their belief in me, along with their constant encouragement, provided the strength and motivation I needed to persevere through the challenges of this journey. Without their support, I might not have been able to see this journey through to the end.

Lastly, thanks also give to my own efforts and persistence. The Ph.D. journey has been full of hardships and challenges, but I am grateful for the personal growth it has brought me. I hope this experience will empower me with the courage to face the future with confidence.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background and Motivation

Autonomous driving is a cutting-edge technology that relies on advancements in computer science and artificial intelligence. The principal aim of this technology is to facilitate vehicles in autonomously executing safe driving operations and achieving destination without human manipulation. Effective deployment of autonomous driving systems is anticipated to enhance traffic safety, improve road efficiency, and optimize the utilization of road resources in an environmentally sustainable manner. Additionally, this technology is poised to create new opportunities for societal advancement.

However, owing to the considerable complexity and inherent risks of driving environments, the progression of autonomous driving technology is exhibiting a deceleration. Currently, the advancement of autonomous driving still necessitates further developments in automation, informatization, and intelligence. There are two different manners to achieve the complete autonomous driving, including the

end-to-end framework and the modular structure. As shown in fig. 1.1, the former employs the black-box feature of the deep learning technology, directly producing the control signal towards the vehicle according to the sensor inputs. The latter decomposes the autonomous driving task into three primary functional modules: environmental perception, decision-making and planning, and control execution. The environmental perception module utilizes driving scenario information collected by sensors (such as camera, LiDAR, etc) to accurately perceive the environment and fully understand objects, events, and states in the environment through various computer vision technologies such as detection, segmentation, and localizaiton. As the forefront task within the autonomous driving framework, the precise sensing capabilities of the environmental perception module are essential for ensuring the safety and reliability of subsequent tasks in autonomous driving.

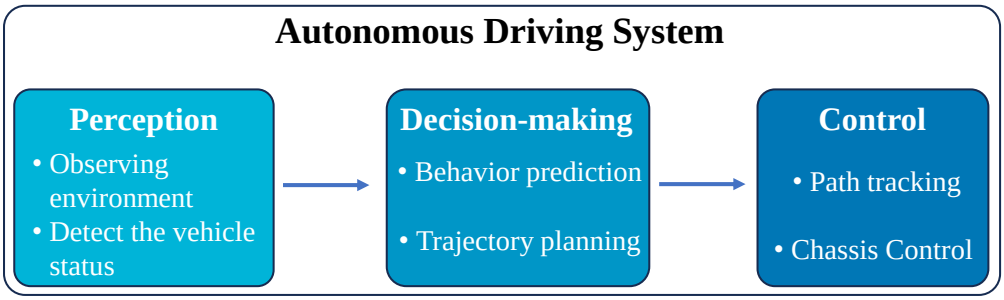


Figure 1.1: The modular structure of autonomous driving

To achieve the accurate environmental perception ability, autonomous vehicles are equipped with an increasing variety of sensors. LiDAR and camera are two commonly used sensors in the environmental perception module. LiDAR excels in collecting distance information, whereas cameras are able to capture color and texture details of the surrounding environment. Those two complement each other, enhancing the overall ability to perceive environmental information. How-

ever, LiDAR and cameras convey environmental data through point clouds and images, respectively. The different data formats complicate the autonomous driving system's environmental understanding. Consequently, developing a unified representation that integrates the diverse data forms from multiple sensors holds significant research importance.

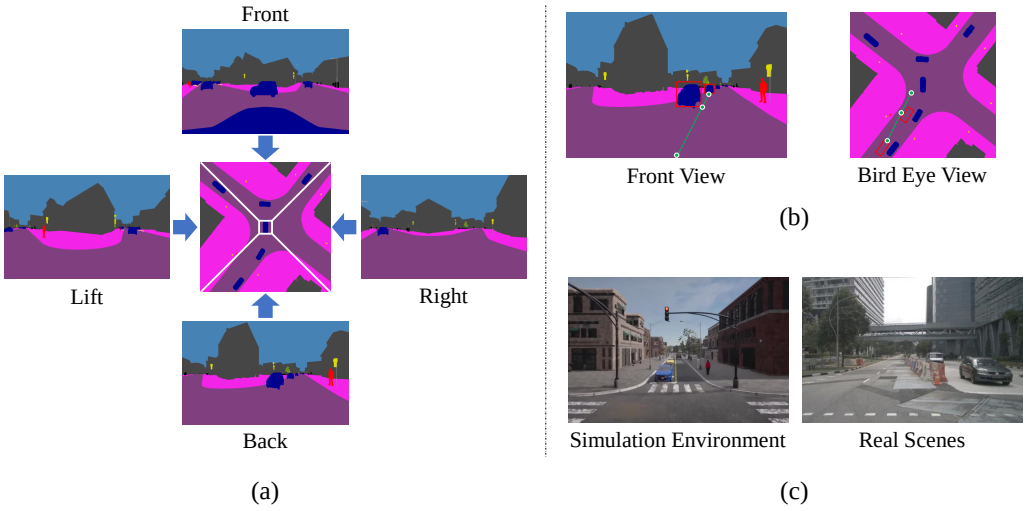


Figure 1.2: The advantages of semantic BEV map

The semantic bird-eye-view (BEV) map represents environmental information by categorizing it into specific semantic groups and illustrating their spatial distribution from a top-down perspective. One merit of semantic BEV map lies in its ability to express the information captured by variety sensors. Point clouds can be transformed into the bird-eye view through downwards projection, while the extensive details provided by images facilitate semantic extraction. Furthermore, compared to other environmental representations, the semantic BEV offers additional advantages. Fig. 1.2 summarizes those advantages: (1) The semantic BEV map can function as a medium for conveying environmental information

from various perspectives. To acquire comprehensive information surrounding the autonomous vehicle, multiple sensors of the same type are mounted around the vehicle's body. For instance, multiple cameras are positioned at different angles to achieve 360° field-of-view coverage. As illustrated in Fig. 1.2 (a), environmental information from diverse perspectives can be concatenated and integrated into a BEV map. (2) Semantic BEV map can mitigate scale and occlusion issues arising from imaging perspective principles. In real-world scenarios, the objects of identical size appear differently in images depending on their distance from the camera. The red boxes in Fig 1.2 (b) illustrate the varying sizes of vehicles in the front view, whereas the BEV map unifies their display size. Additionally, objects closer to the camera may obscure those farther away, resulting in information loss. The BEV map can also eliminate the occlusion. (3) The semantic bird's-eye view (BEV) map facilitates the execution of downstream tasks within the autonomous driving system. Accurate perception of the surrounding environment is essential for behavior prediction [1–3] and trajectory planning [4–6]. As illustrated by the green line in Fig 1.2 (b), the BEV map intuitively represents the distance and angle relationships between the ego-vehicle and other road objects. This provides a robust perceptual foundation for interactions between the ego-vehicle and the driving environment during autonomous driving. (4) The semantic BEV map can mitigate the discrepancies between real-world and simulation environments. Due to the complexity and high-risk of real-world driving, autonomous driving systems in the experimental stage often utilize simulators such as CARLA [7] for training and testing. However, as depicted in 1.2 (c), simulation environments lack the texture details and lightness variations present in real driving scenes. The semantic BEV map abstracts driving scenes into semantic representations, reduc-

ing the visual disparity between real and simulated environments, thereby making the training of autonomous driving more realistic.

Transforming the image from the front view into the bird-eye view involves computer vision tasks such as visual geometry transformation and semantic segmentation. Compared to LiDAR input, generating semantic BEV map from images poses greater challenges in research. Additionally, cameras are characterized by their low cost, minimal impact from weather, and easy for installation. Consequently, the primary focus of this dissertation is to develop an algorithm for generating semantic BEV map using monocular images as input.

1.2 Existing Research Questions

With the rapid advancement of deep learning techniques, semantic BEV map generating methodsare continually being developed. However, current approaches mainly concentrate on designing network architecture and enhancing performance, which are often tested under ideal experimental conditions. These methods overlook the challenges present in real-world environments, such as the issues related to learning samples and information representation. The former refers to insufficient training data and the latter is about the limitations of camera's field-of-view (FoV) and the constrained warning capability of single-frame images. The shortcomings and the areas worthy of further research in semantic BEV map prediction are summarized as follows:

(1) The perspective transformation involved in the semantic BEV map prediction poses challenges for labelling the semantic BEV ground truth, resulting in a scarcity of labeled samples in the training dataset. Typically, semantic ground

truth is created by assigning each pixel a semantic class based on the content of the RGB image, ensuring pixel-level alignment between the RGB image and its ground truth. However, for the semantic BEV map prediction task, the input and output are the front-view image and the BEV semantic map, respectively, which originate from different perspectives. The conversion between the front view and bird's-eye view disrupts the pixel correspondence between the input and output. As a result, considerable labor is required to estimate the positions of the variety road objects due to the view transformation during labelling. This estimation process introduces inaccuracies and errors, resulting in noisy semantic BEV ground truth. Those factors contribute to a scarcity of labeled samples and unsatisfactory quality in the training dataset. Existing reseach on semantic BEV map relies on fully supervised training methods, which depend on the quantity and quality of labeled samples. Therefore, exploring how to effectively use limited semantic BEV labels to train the network and generate reliable semantic BEV maps is a highly worthwhile research direction.

(2) The FoV of on-board cameras is generally restricted, spanning from 30° to 100°. As a result, when images captured by a single camera are converted to the bird's-eye view, effective visual information is concentrated in the conical area, while a significant portion of pixels outside this region remains uninformative. Additionally, due to the restricted FoV of camera, it is sometimes impossible to capture the road information on the left and right sides of the autonomous vehicle, leading to information loss and increased driving risk. Therefore, it is crucial to enhance the environmental information captured by the camera through advanced algorithms and design appropriate networks to obtain a full view semantic BEV map, ensuring safe driving.

(3) Current semantic BEV map prediction methods only depict the environmental content of the current frame, lacking the ability to predict future road conditions or obstacle positions. This limitation impairs the environmental warning capabilities for autonomous driving and hinders the detection of potential road hazards. In contrast, human drivers, due to the continuity of road scenes, can infer upcoming events and respond promptly by analyzing multiple consecutive frames. Hence, investigating the design of a semantic BEV map prediction network capable of predicting future scenarios holds significant research value.

1.3 Mean Contributions and Chapter Organization

This dissertation concentrates on the task of environment perception in autonomous driving, employing deep learning methods to investigate the prediction of semantic BEV map, and exploring various network structures to address problems encountered in real-world scenarios. The structure diagram of the dissertation is illustrated in Fig. 1.3. Chapter 1 provides an introduction. Chapters 2 reviews the previous research invovled in semantic BEV map prediction. Chapters 3 mainly addresses the issue related to learning samples. Chapters 4 and 5 aim to compensate for the deficiencies in information representation within semantic BEV maps. The specific organization of the chapters is as follows:

The primary obstacle in addressing sample-related challenges is the scarcity of labeled training samples. To mitigate this problem, Chapter 3 introduces a semi-supervised semantic BEV map prediction model based on contrastive learning. The objective is to explore methods to diminish the reliance of semantic BEV map prediction networks on labeled samples while maintaining the accuracy of

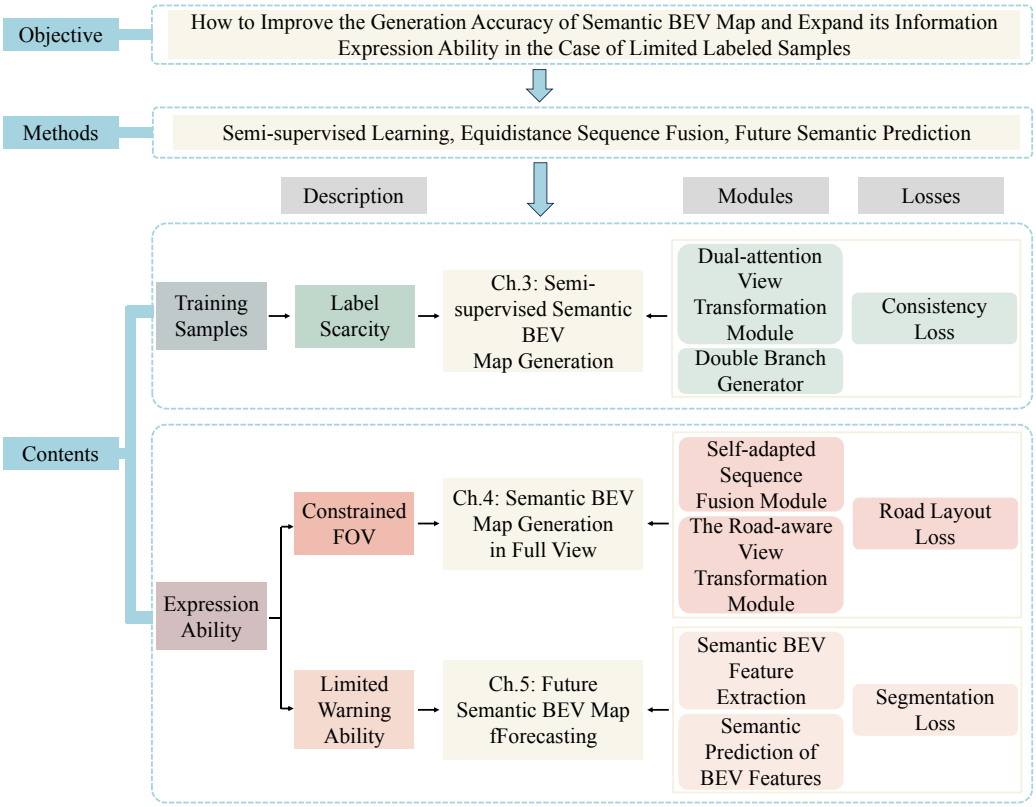


Figure 1.3: The organization of chapters

the generated semantic BEV map. To this end, Semi-supervised Semantic BEV Grid map Generation network (S2G2) is proposed. The proposed network utilizes a dual attention module to complete the perspective transformation from front view to bird's-eye view, and cooperates with a contrastive learning based dual branch semantic generator to enable the network to be trained under the joint supervision of labeled and unlabeled samples. The experimental findings indicate that the proposed dual attention module effectively facilitates view transformation. Additionally, the dual branch semantic generator decreases the necessity for labeled samples during training. The S2G2 network employs its inherent architecture, instead of relying on complex data augmentation techniques, to accomplish semi-

supervised learning.

Another type of problem discussed in this dissertation is the insufficient ability to express semantic BEV information. One reason for this problem is the limited FoV of the on-board camera, which results in restricted observation of environment and wasted space for semantic expression. To address this problem, Chapter 4 introduces a full-view semantic BEV map prediction network based on the self-adapted sequence fusion, namely Seq-BEV. The network introduces a novel approach by utilizing equidistant sequence images as input, enhancing the environmental information obtained from a single frame through a self-adaptive sequence fusion module. This module adjusts the degree of sequence fusion during the training process, resulting in the prediction of a full view semantic BEV map. Additionally, a road-aware view transformation module is developed, adhering to the principles of visual geometry to perform interpretable view transformation. Experimental results validate the effectiveness of the proposed method, demonstrating that Seq-BEV conveys semantic information more comprehensively and achieves more accurate semantic BEV segmentation compared to existing methods.

In order to further expand the information expression ability of semantic BEV map, Chapter 5 proposes a short-term forecasting network to generate semantic BEV map for the future scene based on Long Short Term Memory (LSTM). Existing semantic BEV map prediction networks take current environmental observations as input to generate corresponding BEV map. The current frame has limited warning ability for autonomous vehicles during driving. Deep learning models have powerful predictive reasoning abilities and can reasonably predict future scene changes based on historical observations. This chapter proposes a

short-term forecasting network that can generate a semantic BEV map of future scenes. The network extracts contextual and depth features from historical frames and then uses a future semantic prediction module based on LSTM to infer the semantic features of future scenes. To demonstrate the effectiveness of the proposed method, we established a set of baseline methods based on existing methods and compared them with our future forecasting network. The experimental results show that our proposed future forecasting network can effectively predict the semantic BEV map for unobserved road scenes, and the segmentation accuracy is higher than other methods.

Chapter 2

Literature Review

In this chapter, the previous work related to the semantic BEV map prediction is reviewed. The prediction of a semantic BEV map primarily involves two key issues: semantic prediction and view transformation. Typically, semantic segmentation tasks predict the semantic category of each pixel based on the content of front view images. Although the research objective of this dissertation is to generate semantic maps from the front view to the bird-eye view, the prediction of semantic BEV map still falls under the task of semantic segmentation. View transformation is a prominent research topic in the field of autonomous driving. In tasks related to vehicle-road semantic segmentation, object detection, and lane line detection, view transformation has been extensively explored. This dissertation categorizes the perception tasks in autonomous driving that involve view transformation as BEV perception research. The subsequent sections will review the predominant algorithms for semantic segmentation and BEV perception.

2.1 Related Work in Semantic Segmentation

Image segmentation, as a prerequisite task for image understanding and analysis, is an important part of the field of computer vision. Image segmentation can divide an image into a set of non overlapping regions based on its feature similarity, and assign different semantic categories to these regions. The collection of all semantic regions forms the entire image [8]. The research on image segmentation has a long history. Even before deep learning technology was applied to the field of images, researchers used edge detection [9, 10], Random walk [11–14], Traditional computer vision methods such as clustering [15–18] are used to segment images. At this stage, image segmentation techniques typically follow manually set segmentation rules, utilizing visual features such as grayscale, color, texture, or shape to divide the image into regions. However, practical application scenarios often have a high degree of complexity. The segmentation rules preset in traditional methods cannot make real-time changes and adjustments according to different situations, lack flexibility, and are difficult to deal with fine features in images, resulting in unsatisfactory segmentation results. The emergence of deep learning techniques [19–21] has made it possible to automatically explore deep features in images. Image segmentation instead utilizes abstract high-level semantic information in the image to partition regions, achieving more accurate and efficient segmentation while having stronger generalization ability. The development of deep learning has brought research on image segmentation into the stage based on neural network models.

Fully Convolutional Neural Network (FCN) [22] has pioneered the application of deep learning methods in the field of image semantic segmentation. The purpose

of semantic segmentation task is to convert a image into a multi region semantic mask. The input and output of this task are both images, and maintaining the spatial structure of the image is of great significance for understanding the contextual features of the image. Compared to previous convolutional neural network models, FCN abandons the Fully Connected Layer in the classification network [20, 23, 24] and adopts a fully convolutional structure for dense semantic classification for the first time, ensuring the integrity of the image space structure while breaking the limitation of using fixed size images as input. In addition, when upsampling image features, FCN uses multiple deconvolution layers instead of simple interpolation operations, achieving image size restoration through learning. Since then, semantic segmentation methods based on deep learning have rapidly developed on the basis of FCN, and their development direction is generally moving towards innovative convolutional neural network model structures, combining with other deep learning methods, and exploring new segmentation loss functions.

2.1.1 Application of Convolutional Neural Networks in Semantic Segmentation

Although FCN pioneered the use of fully convolutional operations in the field of semantic segmentation, the image resolution continuously decreases during the downsampling process, leading to blurred segmentation results. Subsequent researchers proposed the Encoder-Decoder structure, [25], where an encoder extracts high-level abstract features from the image while reducing its resolution, and a symmetrical decoder network restores the low-resolution image features into a semantic distribution map of the same size as the original image. Building on the

Encoder-Decoder structure, Badrinarayanan et al. [26] proposed SegNet and for the first time attempted to establish a connection between the encoder and decoder. The innovation lies in preserving the pooling indices from the max-pooling operations during feature encoding and using these indices in the corresponding decoder to recover feature positions. U-Net [27] further strengthened the information connection between the encoder and decoder. Inspired by the residual network [28], U-Net introduced skip connections between the encoder and decoder. This connection structure passes image features obtained after a corresponding number of encoder layers to the decoder, allowing the fusion of high-level abstract feature maps with shallow high-resolution features, thus facilitating information interaction during the upsampling and downsampling processes. Subsequently, similar skip connection structures were widely adopted in the semantic segmentation field. Milletari et al. [29] proposed V-Net for segmenting 3D medical images. Jégou et al. [30] added skip connections to DenseNet [31] to segment images with a deeper network structure. Networks such as G-FRNet [32], U-Net++ [33], and U-Net3+ [34] modified the skip connection structure to improve the combination of shallow features in the encoder and deep features in the decoder.

Skip connections can be viewed as an innovative way to establish links between the encoder and decoder. Beyond this, improvements to the structure of either the encoder or decoder have also become a focal point in the research of neural network-based semantic segmentation methods. Encoders and decoders are composed of multiple stacked convolutional and pooling layers, utilizing this abstract structure to learn complex mapping relationships. Some works [35–39] leverage this characteristic, focusing on modifying the hierarchical structure of the network to achieve better segmentation results. Liu et al. [40], through ob-

servation and analysis of the FCN architecture and its segmentation performance, discovered that contextual information is crucial for comprehensively understanding semantic segmentation scenes. They proposed using global average pooling to provide global contextual priors for semantic segmentation. However, in complex scenarios, global average pooling tends to overlook detailed information and spatial correlations in the image, merging small areas of different objects into a single vector and causing blurred semantic segmentation. To enable the model to better capture multi-scale contextual information, Zhao et al. [41] designed the Spatial Pyramid Pooling module in the proposed PSPNet model. This module uses pooling kernels of different sizes to capture image features at various scales, aggregating contextual information from different regions to obtain both global and local priors. Similarly, Chen et al. [42–44], in their DeepLab series of networks, expanded the application of atrous convolution [45] by using different dilation rates to obtain receptive fields of varying sizes, thereby extracting features at different scales from the same feature map. The DeepLab V3+ network [44], also proposed by Chen et al., integrates features from different levels, significantly improving segmentation results. Lin et al. [46] applied the multi-scale concept to the input of the network, using images of different resolutions as inputs to obtain features at various scales.

2.1.2 Application of Other Neural Networks in Semantic Segmentation

With the development of deep learning, it has also been applied to a wider range of fields, and more and more network structures have emerged, such as Recur-

rent Neural Network (RNN) [47, 48], the self attention model (Transformer) [49–53] and the generation based model [54–57]. These models are gradually being applied to computer vision tasks, providing new ideas for the study of semantic segmentation. Among them, RNN is designed to handle sequence problems and is often used for medical image segmentation in visual tasks. ReSeg [58] attempts to use RNN to compensate for the FCN model's neglect of contextual information. Bai et al. [59] utilized the correlation ability of RNN for image sequences and integrated spatial and temporal information on the basis of FCN. The Long Short Term Memory (LSTM) [60, 61], as an improved version of RNN, introduces different gating signals when processing sequence information, selectively preserves sequence features, and achieves good results in processing sequence data. Subsequently, a large number of LSTM based methods [62–64] were used for medical image segmentation.

Unlike convolutional neural networks (CNNs) and recurrent neural networks (RNNs), the Transformer architecture is entirely composed of attention mechanisms. Transformers use this mechanism to focus on the intrinsic correlations of features, enabling the network to automatically identify meaningful parts of the data through training [65]. Zheng et al. [66] proposed the SETR network, extending the Vision Transformer (ViT) [67] concept to semantic segmentation. SETR abandons convolutional layers, with its architecture consisting entirely of attention modules, abstracting the semantic segmentation task into a sequence-to-sequence transformation process. Wang et al. [68] integrated the Transformer into a pyramid structure, enabling the Transformer to output image features at various resolutions. However, unlike tasks such as natural language processing, visual tasks use images as inputs, leading to a substantial number of network parameters. This results

in increased computational costs for Transformers in visual tasks. Semantic segmentation requires the network to output segmentation maps that match the input image size, further increasing computational demands. To address this issue, Chen et al. [69] proposed a hybrid convolutional-attention model, TransUNet. This network combines the strengths of UNet [27] and Transformers, using features from convolutional encoding layers as inputs to the Transformer. This approach leverages the Transformer's capability to model global context while preserving low-level visual features through convolution operations. Xie et al. [70] simplified the Transformer structure in SegFormer, introducing a Transformer network that does not rely on positional encoding and utilizes a lightweight encoder. The Swin Transformer [71] and Twins [72] focus on the design of the Transformer encoder.

In the field of artificial intelligence, generative models [73] and discriminative models [74] differ fundamentally in principle but hold equal research value. The aforementioned convolutional and other feedforward neural networks fall under the category of discriminative models. These models learn from the feature space of training data, modeling the decision boundaries between classes to achieve classification. In contrast, generative models focus on understanding and capturing the underlying patterns and distribution of the training data, enabling them to generate new data with similar characteristics to the original data. Luc et al. [75] successfully introduced adversarial generative models into the semantic segmentation domain. The authors designed a convolution-based semantic segmentation network and an adversarial network, where the adversarial network's role is to distinguish whether the semantic map is from ground truth data or the prediction of the semantic segmentation network. The adversarial and semantic segmentation networks compete against each other, ultimately training a well-performing

semantic segmentation network. Subsequently, other works [76–78] have leveraged adversarial generative approaches to successfully accomplish medical image segmentation tasks. Zhang et al. [79] proposed an end-to-end framework, VerseDiff-UNet, integrating denoising diffusion probabilistic mechanisms into the commonly used U-shaped structure in semantic segmentation, enhancing segmentation accuracy. EMIT-DIFF [80] incorporates text describing image content into the diffusion probabilistic model, using textual descriptions as conditional prompts to guide the segmentation process. Khani et al. [81] utilized Stable Diffusion [57] to construct a one-shot learning framework, SLiME, training with a small number of samples to achieve image segmentation capability.

2.2 Related Work in BEV Perception

Stable and safe autonomous driving relies on the accurate perception of the surrounding environment by autonomous vehicles. The surrounding environment includes static elements such as road layouts and dynamic elements such as other vehicles and pedestrians. This environmental information can be conveniently represented using bird's-eye view maps, which also facilitate the deployment of downstream tasks in autonomous driving, such as behavior prediction [2, 3] and trajectory planning [4, 5]. The outstanding performance of bird's-eye views in autonomous driving tasks has attracted increasing attention from researchers in the perception field. The images captured by cameras are two-dimensional frontal-view images in a perspective space. Mapping environmental information from the perspective space to the bird's-eye view has become a critical research issue. Inverse Perspective Mapping (IPM) projects frontal-view images onto the bird's-eye

plane through the homography matrix between the camera plane and the ground plane. However, the IPM method relies on strong assumptions, including the ground plane assumption and fixed camera extrinsics [82]. These conditions are difficult to meet in the real world, leading to significant distortions in the bird's-eye views obtained using the IPM method. Accurately mapping frontal-view information to the bird's-eye view requires knowing the depth information of each pixel in the frontal-view image, but the lack of depth information in monocular images poses challenges for image-based bird's-eye perception research. In recent years, various solutions have been proposed to address this issue in image-based bird's-eye perception methods. These methods can be broadly classified into end-to-end learning methods, Transformer-based methods, and depth estimation-based methods.

2.2.1 End-to-End Learning Methods

End-to-end learning methods implicitly address the viewpoint transformation issue in bird's-eye perception. These methods leverage the black-box nature of neural networks to train the network directly in an end-to-end manner, modeling the transformation process from perspective space to bird's-eye space in a data-driven way. The VED method proposed by Lu et al. [83] marked the beginning of end-to-end learning approaches. This network uses a variational autoencoder [55] to directly transform frontal color images into semantic bird's-eye views. The VED network significantly outperforms the IPM method in generating semantic bird's-eye views and is robust to changes in viewpoint directions. Cam2BEV [84] uses the semantic segmentation map of frontal-view images as in-

put. It employs a spatial transformer network [85] to simulate the IPM projection operation and uses a convolutional neural network to correct the transformed feature map, eliminating object distortion caused by the planar assumption. However, this method, which first predicts the semantic segmentation of the frontal-view image and then converts it to the bird's-eye view, tends to accumulate intermediate process errors. Therefore, subsequent bird's-eye perception methods mostly adopt end-to-end learning approaches, using frontal-view images as input and directly outputting semantic bird's-eye views. Pan et al. [86] proposed the VPN network, which simultaneously learns feature information from both depth maps and color images, combining these features to complete the viewpoint transformation. The VPN network also utilizes adversarial learning, employing a discriminator to distinguish between the outputs of the semantic bird's-eye view prediction network and the ground truth labels, thereby encouraging the network's output to resemble the real labels. MonoLayout [87] employs adversarial learning and uses two branches to predict static and dynamic categories separately. Discriminators are then added after the static and dynamic category segmentation heads to improve the network's ability to segment static and dynamic objects. Roddick et al. [88] proposed the PON network, which projects multi-scale features extracted from different network levels onto the bird's-eye plane. This network also adopts a semantic Bayesian occupancy grid framework, achieving the fusion of semantic information from multiple cameras and multiple time steps. Yang et al. [89] constructed a cyclic viewpoint transformation module, where two networks respectively transform frontal-view feature maps to the bird's-eye view and reconstruct the bird's-eye feature maps back to the frontal-view. This network is trained by minimizing the cyclic consistency loss between the original frontal-view features

and the reconstructed frontal-view features, ultimately decoding the bird's-eye feature maps to obtain semantic bird's-eye views.

2.2.2 Transformer-based Methods

Typically, Transformer structures [67] consist of self-attention modules and cross-attention modules, both of which are built upon three learned matrices: Query (Q), Key (K), and Value (V). The self-attention module learns these matrices layer by layer in the original feature domain for feature extraction and enhancement, while the cross-attention module uses the K and V matrices learned from the original feature domain and separately learns the Q matrix from the target domain to extract attention between different domains. In the context of bird's-eye perception, Transformer models typically use a combination of convolutional neural networks and self-attention to extract features from frontal-view images, and then employ cross-attention to generate bird's-eye view features. During the bird's-eye feature generation process, many Transformer-based bird's-eye perception methods draw inspiration from the object query approach in the cross-attention module of the DETR network [90]. This involves using the K and V matrices output by the self-attention module, while treating the Q matrix as a blank target feature template that is iteratively filled with bird's-eye view features during training. Chitta et al. [91] enhance the template filling process by incorporating temporal features along with spatial and semantic features of the bird's-eye view, predicting the corresponding semantic information from the frontal-view image content. Can et al. [92] proposed a Transformer-based road centerline regression network that extracts road network structures and detects vehicles in the bird's-eye view. This network itera-

tively learns static road structure features and dynamic vehicle features, using different segmentation and regression heads to predict vehicle semantics and extract road centerlines in the bird's-eye view. DETR3D [93] extends DETR by using object queries to establish connections between original 2D image features and 3D object detection tasks, directly identifying the center points of 3D bounding boxes and projecting them onto 2D feature maps using camera intrinsic and extrinsic matrices. Similarly, the CVT network by Zhou et al. [94] uses object queries to fill in semantic bird's-eye views, implicitly completing the viewpoint transformation. Liu et al. [95] proposed the PETR network, which refines DETR3D by using 3D position encoding to replace the original 3D-to-2D projection, addressing issues of coordinate errors and limited feature representation range. PETRv2 [96] further improves the 3D position encoder to model time series, leveraging information from previous frames to enhance 3D object detection performance. Saha et al. [97] noted the one-to-one correspondence between vertical pixel columns in frontal-view images and rays through the center in bird's-eye views, transforming the bird's-eye semantic prediction problem into a sequence-to-sequence transformation. By leveraging the Transformer's strengths in sequence processing, they used convolution to capture features around image columns, achieving good performance in semantic bird's-eye view prediction. BEVFormer [98] introduced spatial cross-attention and temporal self-attention modules, integrating multi-camera and multi-time-step image information. This network also utilized Deformable DETR [53] to accelerate the attention extraction process. Subsequently, Yang et al. [99] proposed BEVFormer v2, adding a frontal-view 3D detection head as auxiliary supervision in bird's-eye view detection tasks. The PersFormer network [100] applied Transformer principles for lane detection in bird's-eye views, initializing

bird's-eye features using a homography matrix derived from the IPM algorithm and optimizing features with a Transformer structure. Peng et al. [101] employed multi-scale self-attention modules and object query-based cross-attention modules to segment roads and predict vehicle positions within them.

2.2.3 Depth Estimation-based Methods

The prediction of semantic bird's-eye views involves viewpoint transformation. If the depth of each pixel in the image could be obtained, the distance between the observed objects and the camera could be accurately determined, resulting in precise bird's-eye information. However, most cameras used in autonomous driving are monocular and cannot inherently capture depth information. Consequently, some bird's-eye perception research focuses on explicit depth prediction to enhance perception accuracy. Lift-Splat-Shoot (LSS) [102] was the first work to integrate depth estimation into bird's-eye perception. This method pre-sets a set of depth values for each pixel and, through network training, learns the depth distribution probability for each pixel. Combined with image features, it projects the semantic bird's-eye view downward. The CaDDN network [103] utilizes the depth estimation concept from LSS to accomplish 3D object detection tasks. BEVDet [104] also employs a similar method for 3D object detection. However, since the LSS method requires predicting the depth distribution for each pixel, it consumes a significant amount of memory. To manage memory usage, M^2BEV [105] designed a weight-sharing feature encoder while assuming a uniform depth distribution for each pixel in the image. Unlike the LSS method, this network does not predict the depth distribution, thereby reducing the parameters the network needs to learn,

improving computational efficiency, and saving memory usage. Liu et al. [106] proposed BEVFusion, which uses both image and point cloud data as inputs to extract semantic features from images and geometric features from point clouds. In the image branch of the network, BEVFusion also adopts the LSS approach, assigning a set of depth values to each pixel in the image to lift the 2D image to 3D. During the projection of 3D features to the bird's-eye view, Liu et al. introduced an efficient bird's-eye pooling operation to optimize the grid association and feature aggregation in LSS, enhancing computational efficiency. MatrixVT [107] addressed the high computational cost of the Lift and Splat operations in the LSS method by proposing the concept of element extraction. This approach allows the network to focus on the main content of image features and corresponding depths, reducing the dimensions of image features and depth distributions and lightening the bird's-eye perception network.

Unlike the LSS method, other depth estimation-based approaches integrate existing monocular depth estimation networks to explicitly predict image depth, thereby enhancing the reliability of depth prediction and improving bird's-eye perception performance. BEVDepth [108] designed a depth network that takes frontal-view image features and camera intrinsics as input to predict the depth distribution of the scene. This network uses point cloud data to supervise depth prediction and subsequently feeds the depth features into a depth correction module constructed with convolutional layers, further enhancing the reliability of depth prediction.BEVStereo [109] builds upon BEVDepth by altering the depth prediction method. This approach estimates the stereo vision of the scene using the disparity between consecutive image frames.

Chapter 3

Semi-Supervised Semantic BEV Map Prediction

3.1 Motivation

Data representation for semantic environment perception is critical in autonomous driving. In recent years, semantic bird-eye-view (BEV) grid maps have attracted increasing attention in the robotics research community. Compared to the common data representation (i.e., front-view semantic segmentation images), semantic BEV grid maps are more straightforward to use. In semantic BEV maps, geometric relationships between ego-vehicle and obstacles are explicitly illustrated in a natural view. This advantage makes them more suitable for downstream tasks, such as motion planning [110–112], trajectory prediction, etc. Moreover, many networks of these downstream tasks are trained with visual images in simulation environments (e.g., CARLA). They suffer from the domain gap issue when transferred to real-world environments, since simulation environments lack real texture

details compared to real-world environments. Using semantic BEV grid maps instead of visual images could alleviate this issue, because semantic maps almost have the same style in both simulation and real-world environments.

Different from semantic segmentation algorithms that label front-view camera images pixel-wisely into front-view semantic maps, our task is more like a prediction process which generates semantic BEV maps from front-view camera images. To achieve this goal, some works [113, 114] first generate standard front-view semantic maps from front-view camera images, and then project the segmentation maps into the bird eye view using view transforming algorithms, such as the inverse perspective mapping (IPM) algorithm. However, the IPM algorithm suffers from the flat ground assumption [82], and the pipeline accumulates errors through the two steps. The issues make this stream of methods less generalizable. To address these issues, recent methods [83, 86, 89] resort to generating semantic BEV maps in an end-to-end manner, which could avoid using the IPM algorithm and alleviate the error propagation issue. However, most of the existing end-to-end methods adopt supervised learning to train their networks, which requires a large amount of ground-truth images to achieve acceptable results. The datasets with hand-labeled ground truth for semantic BEV grid maps are limited. In addition, manually labeling images is tedious and labor-intensive, and manually drawing semantic BEV maps according to the front-view camera images is difficult for humans.

To provide a solution to this problem, we propose a novel Semi-Supervised semantic BEV Grid-map Generation (S2G2) network, which requires only a small amount of labeled data and a large amount of unlabeled data to achieve superior performance. Our network is end-to-end. It consists of two major components:

view transformation from front-view to bird eye view, and semantic labeling on the bird eye view.

To the best of our knowledge, our network S2G2 is the first solution to generate semantic BEV maps in a semi-supervised manner. We implement multiple baselines to perform extensive comparative studies on a public dataset [83]. The results demonstrate our superiority. The contributions of this work are summarized as follows:

1. We propose S2G2, a novel semi-supervised semantic BEV prediction network that can be trained with unlabeled data.

2. We introduce a new dual-attention view transformation module to transform the front-view input into the bird-eye-view feature maps.

3. We create several semi-supervised baseline methods and compare our network with the baselines and the state-of-the-art supervised methods.

3.2 Background

3.2.1 Contrastive Learning VS Transfer Learning

The existing semi-supervised learning methods can be roughly divided into two categories, namely contrastive learning [115, 116] and transfer learning [117]. The network piplines of the two methods are shown in Fig. 3.1. The contrastive learning assumes that the fundamental nature of data remains unchanged despite the introduction of various perturbations. Based on this principle, the contrastive learning framework comprises two branches with identical network structures, referred

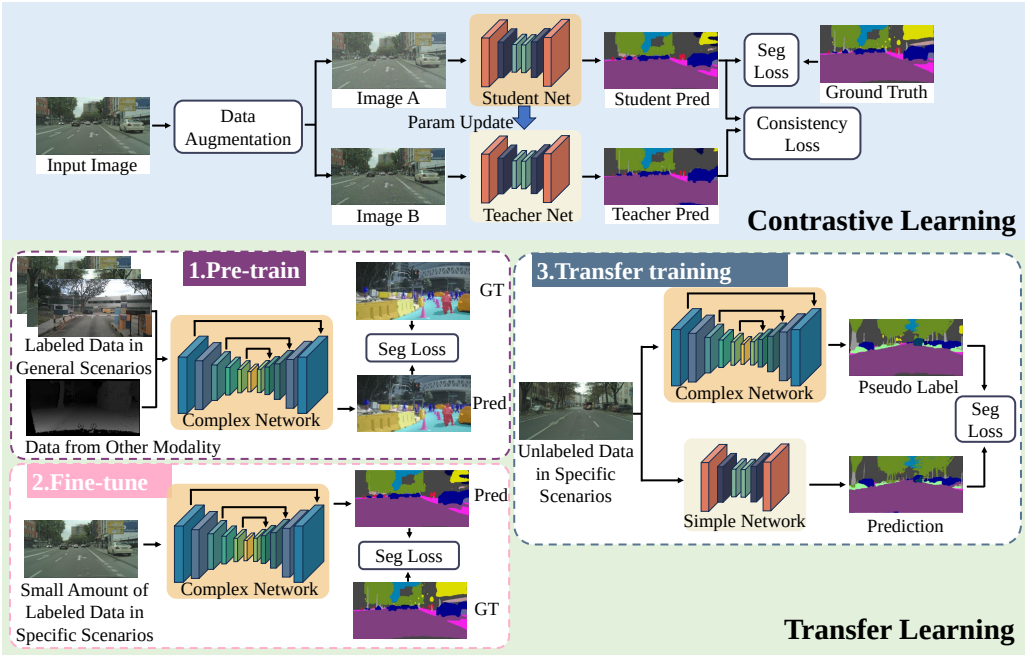


Figure 3.1: The pipelines of contrastive learning and transfer learning

to as student network and the teacher network, as depicted in the blue background in Fig. 3.1. Following random data augmentation on the image, two new images, A and B, are generated, each with different appearances but retaining the same essential content. These augmented images, A and B, are then fed into the student and teacher networks for result prediction. It is worth noting that only the student network is supervised using a small amount of ground truth labels to calculate the segmentation loss. The core idea of contrastive learning posits that when images with identical essential content are used as input, regardless of appearance changes, the two branch networks with the same structure should yield identical results. Consequently, the predictions of the student network and the teacher network should match, prompting the introduction of a consistency loss to supervise the output similarity of the two branch networks. Subsequently, gradients based

on segmentation loss and consistency loss are calculated, and error backpropagation is performed through the student network to update its parameters. In contrast, the teacher network adjusts its parameters based on those of the student network, without participating in the gradient backpropagation process. Through multiple iterations of training, the student network acquires the ability to learn from images under the supervision of a limited amount of ground truth labels. The parameter update mechanism between the student network and the teacher network enables the transfer of learning ability, allowing the teacher network to segment images without direct supervision.

The green background in Fig. 3.1 illustrates the pipeline of transfer learning. Transfer learning methods divide the semi-supervised learning into three distinct steps. The first step involves pre-training a complex network, characterized by a large structure and numerous parameters, using a substantial amount of relevant scene data or multimodal data (such as depth data, disparity images, etc.) in a fully supervised manner. By leveraging the complexity of the network and the substantial quantity of training data, these complex networks can achieve robust segmentation capabilities. The network is fine-tuned in the second step by a small amount of labeled data from the specific scenarios to futher adjust the parameters of the complex network and enhance its performance in specific tasks. The third step is to transfer the learning ability of complex networks to more compact and simple networks. During transfer training, the network reduces its dependence on labeled samples by utilizing the learning capabilities of the pre-trained complex network to perform semantic segmentation on images. The predicted results of the complex network are then used as pseudo labels to supervise the simpler network. Transfer learning methods heavily rely on the training of complex networks. In the

final step, the simple network depends entirely on the segmentation performance of the complex network. If the complex network is not adequately trained, it will not only waste training resources but also result in suboptimal performance of the simple network.

3.2.2 Mean Teacher Model

The Mean Teacher model [115] is developed for classification tasks and serves as a notable example of contrastive learning-based semi-supervised models. At the input stage, the Mean Teacher model employs various data augmentation techniques (η, η') on the image, preserving the underlying content to generate two visually distinct images. Following the contrastive learning approach, two networks with identical structures are utilized to classify the images with different appearances. Finally, a consistency loss J is applied to penalize the discrepancies between the outputs of the student network and the teacher network. The consistency loss is calculated using the Mean Square Error (MSE) and can be expressed by the following formula:

$$J(\theta) = \mathbb{E}_{x,\eta',\eta}[\|f(x,\theta',\eta') - f(x,\theta,\eta)\|^2], \tag{3.1}$$

with η and η' representing the data augmentation noise added to the images fed into the student network and the teacher network, respectively. θ, θ' denote the weight parameters of the student network and the teacher network. At the same time, the student network is supervised by a classification loss. After each training iteration, the student network updates its parameters under the supervision of consistency loss and classification loss.

To update the teacher network, the Mean Teacher model employs the concept of Ensemble Learning by first applying the Exponential Moving Average (EMA) to the parameters of the continuously updated student network and then allocating these averaged parameters to the teacher network, rather than sharing parameters with the student network as in previous approaches. Let θ_t represent the parameters of the student network after the t-th training iteration. The current teacher network parameters θ'_t are updated as follows:

$$\theta'_t = \alpha\theta'_{t-1} + (1-\alpha)\theta_t, \tag{3.2}$$

where α is the smoothing parameter of EMA. By using EMA as a strategy for updating the teacher network parameters, the teacher network can be considered as the averaged state of the student network after multiple consecutive updates. Each training session involves feeding small batches of data into the network and adjusting the network parameters accordingly, making it difficult to capture the characteristics of the entire dataset comprehensively. The parameter updates may occasionally deviate, and averaging the network parameters helps to broaden the network's observation range on the data, leading to improved classification results.

The Mean Teacher model suggests that to achieve better semi-supervised learning results, stronger data augmentation methods such as Mixup [118], Cutout [119], CutMix [120] should be applied to the input images. These techniques generate two images with higher discrimination for the student and teacher networks, enabling the network to gain a deeper understanding of the abstract essence of the images. However, data augmentation methods like Cutout can disrupt the spatial structure of images. Unlike the classification tasks handled by the Mean

Teacher model, generating semantic BEV map requires determining semantic relationships and performing view transformations based on the spatial associations in the front view image. Consequently, such data augmentation methods are not suitable for the task of generating semantic BEV map. Comparing to the single network structure, the contrastive learning-based semi-supervised method necessitates two identical networks, thereby increasing the scale of network parameters and consuming relatively high computational resources. To address this issues, a network that integrates view transformation and contrastive learning is proposed. This network can produce two different representations of the same image through its own internal module. These representations are then fed into a dual branch semantic generator, facilitating semi-supervised training using a small number of labeled samples and a large number of unlabeled samples.

3.3 The Proposed Network

3.3.1 The Overall Architecture

The motivation of this work is to generate semantic BEV grid maps from input front-view monocular images in an end-to-end manner. A contrastive learning-based semi-supervised learning framework with double branches is proposed and its network architecture is illustrated in Fig. 3.2. As we can see, our S2G2 takes as input both labeled and unlabeled front-view images. It consists of a feature extractor, a dual-attention view transformation (DVT) module, and a double branch generator (DBG). We employ the EfficientNet [121] as the backbone of the feature extractor to extract front-view features from the input images. This encoder

is shared for both passive and active branches. The network first extracts front-view feature maps $\mathcal{F}_{front}$ from the input front-view images, then transforms the viewpoint from front-view to BEV through the DVT module. Two distinct BEV feature maps, $\mathcal{F}_I$ and $\mathcal{F}_C$, originating from the same input image, denoted as the homologous features, are generated by the dual-attention block in the DVT module. The contrastive learning strategy adjusts the network parameters by minimizing the consistency loss between two identical networks. Therefore, the homologous features from the DVT module serve as the diversely augmented versions of one original image, which are the naturally suitable inputs for this semi-supervised scheme. With the semantic heads in both branches, a semantic BEV grid map can be generated.

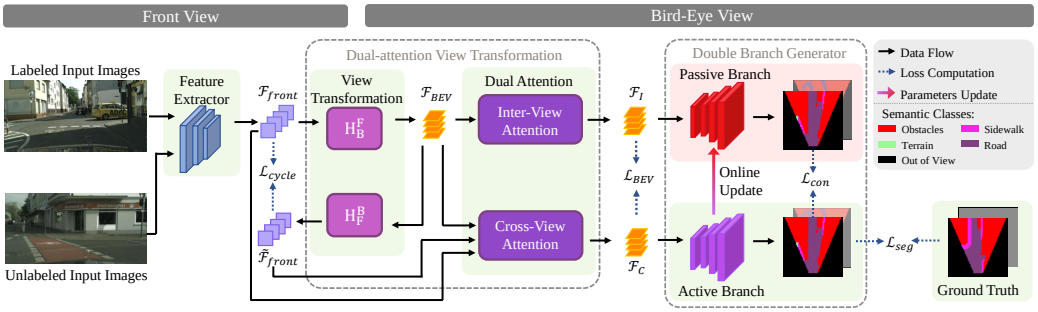


Figure 3.2: The overall architecture of the proposed S2G2

3.3.2 The Feature Extractor

A pre-trained CNN model, EfficientNet [121], is used as our feature extractor. Different from the existing contrastive learning methods [115], we employ a shared encoder to extract the low-level features from both labeled and unlabeled images rather than using a parallel architecture with two identical networks, which results

in a larger network model. With the increase of stages in EfficientNet, the receptive fields are enlarged, the backbone gradually reduces the feature-map resolution but increases the number of feature-map channels. The output feature map of the encoder is denote as $\mathcal{F}_{front}$. Since the EfficientNet has various variants, the number of channels of $\mathcal{F}_{front}$ can be different. Detailed channel numbers are display in Tab. 3.1

Table 3.1: The numbers of the channel of front-view feature map $\mathcal{F}_{front}$ after the feature extraction with different EfficientNet variants as the backbone, ranging from EfficientNet-B0 to EfficientNet-B7. EffNet is the short for EfficientNet.

	EffNet-B0	EffNet-B1	EffNet-B2	EffNet-B3
Channels	320	320	352	384
	EffNet-B4	EffNet-B5	EffNet-B6	EffNet-B7
Channels	448	512	576	640

3.3.3 The Dual-Attention View Transformation Module

To transform feature maps from front-view to BEV, we design the DVT module. According to the frontal features, this module predicts the corresponding feature maps in bird-eye view. The DVT module includes a view transformation block and a dual attention block. The former is designed to perform the view projection in a learning-based approach and the later will strengthen the transformed results.

3.3.3.1 View Transformation Block

Inspired by [122], the view transformation can be realized by training a transformation module H_B^F that transforms the feature maps from the front-view to BEV,

$\mathcal{F}_{BEV} = H_B^F(\mathcal{F}_{front})$. Another transformation function H_F^B is the inverse of H_B^F. H_F^B transforms the $\mathcal{F}_{BEV}$ back to the front view, $\tilde{\mathcal{F}}_{front} = H_F^B(\mathcal{F}_{BEV})$. To train the view transformation block, a cycle consistency loss is introduced here:

$$\mathcal{L}_{cycle} = ||H_F^B(H_B^F(\mathcal{F}_{front})) - \mathcal{F}_{front}||_1. \tag{3.3}$$

Minimizing $\mathcal{L}_{cycle}$ encourages the re-transformed front-view feature map $\tilde{\mathcal{F}}_{front}$ to be similar to the original one, $\mathcal{F}_{front}$. The input of the module H_B^F is the front-view feature map and a corresponding BEV feature map is the output. Here, we use double convolutional layers to fit the transformation module H_B^F and H_F^B. The convolutional operation focuses on the local features and preserves the spatial information. The designed double-layer convolution enlarges the receptive field layer by layer until covering the whole input feature, $\mathcal{F}_{front}$. This could allow our view transformation block considering both local and global information during the view transformation.

3.3.3.2 Dual Attention Block

Based on the work [89], we design a dual attention block to improve the view transformation results. We keep the cross-view attention part unchanged, following [89], which takes $\mathcal{F}_{front}$, $\mathcal{F}_{BEV}$ and $\tilde{\mathcal{F}}_{front}$ as inputs to infer the attention score between the front-view and BEV. The cross-view attention emphasizes the relationship between the two different views. However, the internal relationship within the generated BEV feature map is also worth noting. Since the convolution layer can be seen as a feature extractor, the feature maps produced from multi-layer convolutions already gathered different kinds of features, stacking in

the channel dimension. Moreover, the salient features are located differently in each feature layer at the spatial dimension. Therefore, we design an inter-view attention sub-block, which combines both channel and spatial attention to highlight the internal relationship of the BEV feature map, $\mathcal{F}_{BEV}$. The inter-view attention sub-block and cross-view attention sub-block together form the dual attention block and complement to each other.

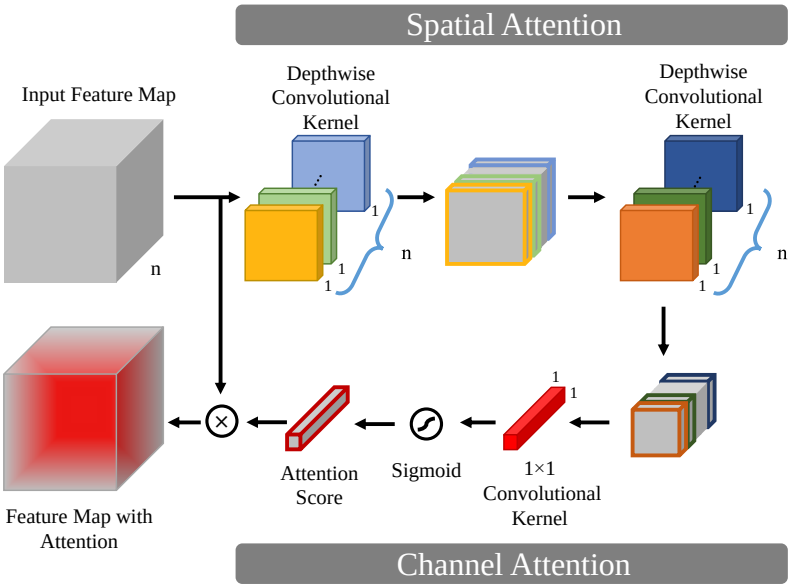


Figure 3.3: The structure of the inter-view attention block

Fig. 3.3 demonstrates the proposed inter-view attention sub-block. In this sub-block, we first perform n depthwise convolution kernels [123] on n-channel input feature map separately without changing the depth. This operation reduces the resolution of the feature map and the salient features located in the spatial dimension can be learned as training. We repeat depthwise convolution twice in our proposed S2G2. An 1×1 convolution is applied to the intermediate features sequentially, which exploits the channel relationship of the feature maps. Both types of convo-

lution are followed with a max-pooling operation. In short, the inter-view attention is computed as:

$$S_A = \text{sigmoid}\{C_1[C_d(\mathcal{F}_{in})]\}, \tag{3.4}$$

$$\mathcal{F}_{out} = \mathcal{F}_{in} \otimes S_A, \tag{3.5}$$

where C_d and C_1 denote the depthwise convolution and 1×1 convolution respectively. After two successive attention extraction, a Sigmoid function is applied to map the convolution output into the range from 0 to 1. Then, an attention score, S_A is produced. $\otimes$ represents element-wise multiplication. Through multiplication, the internal attention is spread into the input feature map.

In order to make the inter-view attention and cross-view attention complementary, we introduce a cross entropy loss function, $\mathcal{L}_{BEV}$, between the outputs of the separate attention block. The DVT module maintains the same dimension in the input and output feature maps, so it can be inserted into any existing network seamlessly.

3.3.4 The Double Branch Generator

Grounded on the contrastive learning strategy, we propose a double branch generator, which is composed of an active branch and a passive branch to implement semi-supervised learning. The main idea behind contrastive learning is that similar data are clustered together and different data are pushed away. This assumes that the network should generate consistent outputs, given similar inputs. In such a way, the unlabeled data can be utilized to boost the training process. Therefore, the performance of the contrastive learning-based semi-supervised methods relies

largely on the prediction of the homologous data. The existing contrastive learning approaches perform strong data augmentation combinations, as such Mixup [124], Cutout [119], and CutMix [120] to generate diverse versions of the same data. In our work, the output feature map is not aligned with the input image due to the view transformation task. Therefore, those data augmentation techniques that require the alignment between the input and output, do not apply to our network.

With the dual attention block, we get two outputs, $\mathcal{F}_I$ and $\mathcal{F}_C$. The two feature maps concentrate on the inter-view relationship and cross-view relationship, respectively. The two attention-included outputs originate from the same input but differ from each other. Therefore, we take them as inputs for the active branch and passive branch, naturally. To endow the network with the ability to output similar predictions for the similar inputs, we introduce a consistency loss, $\mathcal{L}_{con}$ that calculates the differences between the predictions from the active branch and the passive branch with mean squared error. The consistency loss can be written as:

$$\mathcal{L}_{con} = ||P_{act}(\mathcal{F}_c, \omega_a) - P_{pas}(\mathcal{F}_I, \omega_p)||_2, \tag{3.6}$$

where $P_{act}(\cdot)$ and $P_{pas}(\cdot)$ are the predictions from active branch and passive branch. The weights for the two branches are ω_a and ω_p, respectively.

At each training step, the active branch updates via the gradient descent from the weighted sum of the segmentation loss, $\mathcal{L}_{seg}$ and consistency loss, $\mathcal{L}_{con}$. We define the segmentation loss as the cross-entropy loss for the labeled images. The consistency loss is used for both labeled and unlabeled images. The weights in passive branch (ω_p) are updated with an Exponential Moving Average (EMA) strategy

instead of the gradient descent manner, which is formulated as:

$$\omega_p^i = \lambda\omega_p^{i-1} + (1-\lambda)\omega_a^i, \tag{3.7}$$

where the superscript i represents the i-*th* training step. λ is a hyperparameter for EMA decay, and it is set as 0.999 in our experiment.

3.3.5 Loss Function

We add losses from different modules together. We train our S2G2 in an end-to-end manner. The overall loss function is formulated as:

$$\mathcal{L} = \mathcal{L}_{seg} + \alpha\mathcal{L}_{cycle} + \beta\mathcal{L}_{BEV} + \gamma\mathcal{L}_{con}, \tag{3.8}$$

where $\mathcal{L}_{seg}$ is the major loss for our network. α, β, and γ are the weighted coefficient to balance each loss. In practice, we empirically set α, β both equal to 1, and let consistancy weight, γ, be adjusted in a self-adaption way. We will discuss the details of those hyperparameters in the experiment section.

3.4 Experimental Results and Discussions

3.4.1 The Dataset

We conducted our experiments using the dataset from MonoOccupancy [83]. This dataset includes 2600, 375 and 500 labeled images for training, validation and test. The dataset preserves the input images of the public dataset, Cityscapes [125]. But the authors create their own semantic BEV ground truth via semi-global match-

ing (SGM) method [126], using the disparity maps provided by Cityscapes. The ground truth contains 4 semantic classes, which are obstacles, sidewalk, terrain, and road. Fig. 3.4 illustrates several examples from the dataset. The first line shows front view RGB images, followed by manually annotated semantic BEV labels and rough labels generated using the SGM algorithm. The figure demonstrates that, compared to manually annotated labels, the rough labels exhibit some semantic omissions and category errors. However, due to the high cost of manual annotation, our experiments are conducted under weak supervision with these noisy rough labels. To evaluate the semi-supervised architecture, we build three groups with different ratios of unlabeled images. The ratios of the unlabeled images are 10%, 40% and 80%. Note that we use all the labeled images in the training set. The input images are normalized to 256×512 and the output size is 64×64.

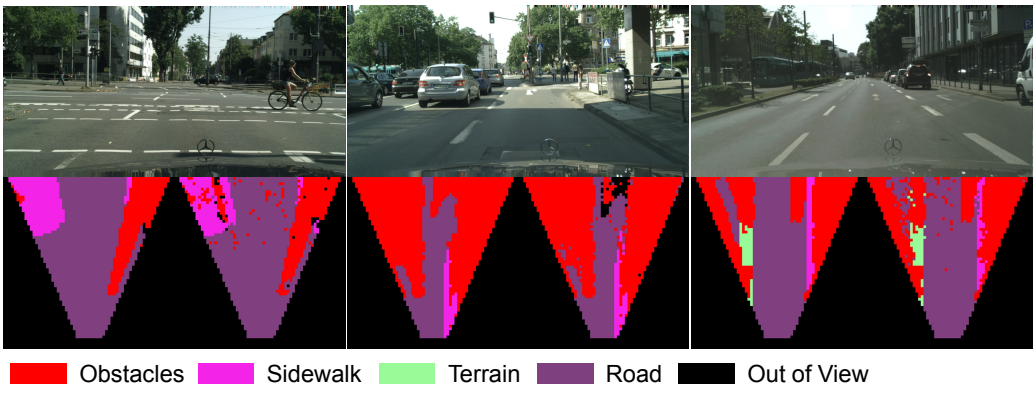


Figure 3.4: The examples from the training dataset

We randomly shuffle the input training data before feeding them to the network. Because the input image and generated semantic maps are in different perspectives, we apply random flipping and random brightness changing to perform the data augmentation, maintaining the relative position of the content in the images.

3.4.2 Training Details

The training is performed with an NVIDIA GeForce RTX 3060 GPU. Due to the limited memories of the GPU, we set the batch size to 4, which consists of 2 labeled images and 2 unlabeled images. We train our network for 200 epochs with the stochastic gradient descent (SGD) optimizer. The momentum and weight decay are set to 0.9 and 1×10^{-4}, respectively. We initialize the learning rate to 5×10^{-5} and adopt an exponential decay scheme to adjust the learning rate as training. The decay coefficient of the learning rate is 0.995.

Specifically, we adopt EfficientNet-B4 with the pre-trained weights as our backbone, the rest of our network parameters are randomly initialized. We employ a ramp-up tuning scheme to adjust the consistency weight, γ, following the practice of the Mean Teacher [115]. The ramp-up scheme ensures that γ increases to the set value gradually until the end of the ramp-up phase. Through an extensive ablation study, we set the consistency weight, γ, to 1 and the ramp-up step to 100. The selection details will be elaborated in the ablation study.

3.4.3 Ablation Study

We conduct several ablation experiments to verify the effectiveness of the structure and parameters used in our S2G2. To assess the performance of our S2G2, Mean Intersection over Union (mIoU) and Mean Average Precision (mAP) are adopted

as evaluation metrics:

$$mIoU = \frac{1}{N+1}\sum_{i=0}^{N}\frac{p_{ii}}{\sum_{j=0}^{N}p_{ij}+\sum_{j=0}^{N}p_{ji}-p_{ii}}, \tag{3.9}$$

$$mAP = \frac{1}{N+1}\sum_{i=0}^{N}\frac{p_{ii}}{\sum_{j=0}^{N}p_{ji}}, \tag{3.10}$$

Where N represents the number of semantic categories. We denote i as the label truth category and j as the predicted category. p_{ii} is the true example of category i, p_{ij} represents the total number of pixels belonging to category i in the truth label, and p_{ji} is the total number of pixels predicted to belong to category i.

According to the different amounts of the unlabeled images in the training set, we conduct our ablation experiments with 3 different sets, which respectively include 10%, 40%, and 80% of the unlabeled images.

3.4.3.1 Ablation on the Feature Extraction Module

In our S2G2, the implementation of semi-supervised learning depends on a double branch generator which enlarges the scale of network parameters, compared with the fully supervised network. Therefore, to make our network be efficient and effective in terms of training speed and memory usage, a compact and powerful backbone should be selected for the feature extraction module. EfficientNet is a network that focuses both on accuracy and efficiency. The EfficientNet family includes 8 variants, which are named as EfficientNet-B0 to EfficientNet-B7, respectively. Those variants are different from each other in the depth, width, and resolution.

In the ablation study, we first compare the performance of the proposed net-

work with different EfficientNet variants. In our network, we only keep the feature extraction part of the EfficientNet and remove the average pooling of the last layer, as well as the classification head. The modified EfficientNet variants produce a front-view feature map with a fixed size of 8×16 but with diverse numbers of channels. The different channels of the output feature map are listed in Tab. 3.1.

Tab. 3.2 displays the results of the ablation study on the different EfficientNet variants, including EfficientNet-B0 to EfficientNet-B7. The obvious raising trends can be seen when increasing the complexity of EfficientNet from B0 to B4. But after B4, the mIou and mAP of the prediction performance stay in a relatively stable range. To trade off performance and computation cost, we select EfficientNet-B4 as the backbone of our S2G2.

Table 3.2: The ablation study results (%) of the variants of the EfficientNet Family. According to the different amounts of the unlabeled images in the training set, we conduct our ablation study into 3 groups, which contain 10%, 40%, and 80% unlabeled images, respectively.

Variants	10%		40%		80%	
	mIou	mAP	mIou	mAP	mIou	mAP
EfficientNet-B0	0.5468	0.6410	0.5834	0.7006	0.5795	0.6941
EfficientNet-B1	0.5847	0.6745	0.5703	0.6635	0.5863	0.7066
EfficientNet-B2	0.5774	0.6766	0.5807	0.6902	0.5839	0.7020
EfficientNet-B3	0.5840	**0.7112**	0.5747	0.6784	0.5877	0.7099
EfficientNet-B4	**0.5894**	0.7003	0.5889	0.6956	**0.5879**	**0.7110**
EfficientNet-B5	0.5852	0.7044	**0.5890**	0.6954	0.5830	0.7098
EfficientNet-B6	0.5854	0.7046	0.5880	**0.7162**	0.5879	0.6952
EfficientNet-B7	0.5794	0.6959	0.5852	0.6937	0.5835	0.6928

3.4.3.2 Ablation on the Dual-Attention Block

To verify the effectiveness of the dual-attention block, we conduct two groups of tests with and without a certain attention block. We first only keep the inter-view attention block and discard the cross-view attention block. We term this variant as Only Inter-View Attention (OIVA). Then, we remove the inter-view attention block instead and get the Only Cross-View Attention (OCVA) variant. According to the results of the previous ablation study, our network gets the best performance with EfficientNet-B4. But EfficientNet-B7 is the most complicated variant with the most number of parameters, it should perform better in OIVA or OCVA. So, we chose the B4 and B7 variants as our feature extraction module to conduct this ablation study.

Moreover, we also exchange the input order to the Double Branch Generator module, which leads to two different structures. The first one takes the output feature from the inter-view attention, $\mathcal{F}_I$ as the input of passive branch and we denote this variant as S2G2-IPCA (Inter-view attention feature map for Passive branch and Cross-view attention feature map for Active branch). Note that S2G2-IPCA is the same as the proposed S2G2. For the second one, we let the feature map, $\mathcal{F}_C$ be the input of passive branch and term this as S2G2-CPIA (Cross-view attention feature map for Passive branch and Inter-view attention feature map for Active branch).

Tab. 3.3 demonstrates the effectiveness of the dual-attention block. From the table, the dual-attention that combines the inter-view and cross-view attention together, gets a superior performance against its counterparts. Tab. 3.4 shows the comparative results of different input orders to the active branch and passive

Table 3.3: The ablation study results (%) on dual-attention block. OIVA stands for the variant that only keeps the inter-view attention sub-block and OCVA means the module that only have the cross-view attention sub-block. B4 and B7 present the experiments are conducted with the EfficientNet-B4 and EfficientNet-B7 as their backbone.

Variants	10%		40%		80%	
	mIou	mAP	mIou	mAP	mIou	mAP
OIVA(B4)	0.5610	0.6632	0.5371	0.5750	0.5597	0.6725
OCVA(B4)	0.5277	0.6820	0.5372	0.6551	0.5605	0.6673
S2G2-B4	**0.5894**	**0.7003**	**0.5889**	**0.6956**	**0.5879**	**0.7110**
OIVA(B7)	0.5327	0.5957	0.5467	0.6368	0.5593	0.6550
OCVA(B7)	0.5425	0.4809	0.5308	0.5793	0.5471	0.6173
S2G2-B7	**0.5794**	**0.6959**	**0.5852**	**0.6937**	**0.5835**	**0.6928**

branch. The results indicate that the IPCA variant has a higher mIoU with 58.94%, compared with the CPIA variant, 56.95%. This is also true for the metric mAP.

Table 3.4: The ablation study results (%) on the different input orders to the finial double branch generator. S2G2-CPIA module feeds the cross-view attention feature map, $\mathcal{F}_C$, to the passive branch and the inter-view attention feature map, $\mathcal{F}_I$, to the active branch. S2G2-IPCA is the opposite version of S2G2-CPIA.

Variants	mIoU	mAP
S2G2-CPIA	0.5695	0.6657
S2G2-IPCA	**0.5894**	**0.7003**

3.4.3.3 Ablation on the Double Branch Generator Module

For the double branch generator module, we test different sets of parameters to check the impacts on the intensity of contrastive learning. Specifically, the consistency loss is linked to the prediction of similar outputs from the active branch and

passive branch. The learning effectiveness of the unlabeled images of the network is affected by the consistency loss-related hyperparameters, including consistency weight γ and the ramp-up step. Therefore, we set 6 different values for the consistency weight (0.05, 0.1, 0.5, 1, 5, 10) and 5 different ramp-up step (50, 75, 100, 125, 150) in this experiment.

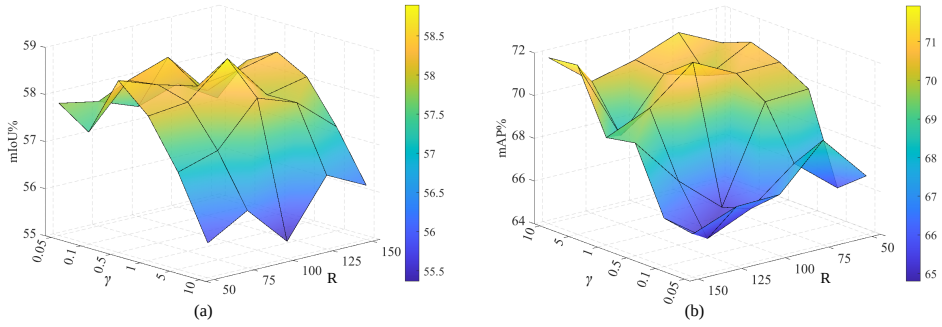


Figure 3.5: Impacts of the ramp-up steps (R) and the weighted coefficient of consistency loss (γ) on the mIoU and mAP.

The process of parameter tuning is presented in Fig. 3.5. We find that when the weight of the consistency loss equals to 1, and the ramp-up step is set as 100, the network produces the best performance. These two parameters impose an effect on how well the network can learn from the unlabeled data. We also find that a small consistency loss weight and a big ramp-up step can lead to insufficient contrastive learning due to less punishment towards the consistency loss. This means that the network could not predict the consistent outputs for the homologous features. But large weight value and too quick ramp-up may force the assimilation of the two branches, still resulting in a deficient learning capacity.

3.4.4 Comparative Results

3.4.4.1 Comparison with Baseline Methods

As our S2G2 is the first method that generates the semantic BEV grid map in a semi-supervised manner, we develop several baseline methods to perform the comparative experiments. Our target outputs are still in the image domain, so to form our baselines, we take advantage of the popular semantic segmentation methods, including U-Net [27], DeepLab V3+ [44], RTFNet [127], SegNet [26], HRNet [128]. Other than those semantic segmentation methods, we also take the MonoOccupancy [83] as consideration. However, the above methods are all trained in a supervised manner. In order to train the networks with the unlabeled images, we integrate the mentioned segmentation networks into the Mean Teacher [115] framework, which is originally designed for semi-supervised classification.

Specifically, we modify the aforementioned methods by adding an aspect-ratio changing layer and adjusting the output size of their decoders, because the input resolution (256×512) is not the same as that of the output (64×64). The Mean Teacher framework depends on the two identical networks to perform contrastive learning. So we duplicate the aforementioned methods as the two parallel networks in the Mean Teacher framework. To follow the idea of contrastive learning, we apply the random Gaussian noise to make the input image a pair of homologous similar ones. Then the noise-injected images are fed into the two networks of the Mean Teacher framework, respectively.

We report the quantitative comparative results for the baseline methods in Tab. 3.5. The results show that our proposed S2G2 achieves the best performance in terms of mIoU and mAP across all the networks. From the table, we can see

that the MonoOccupancy gets the second-best results. MonoOccupancy is also a semantic BEV grid map generator.

We conjecture the reason for the inferior performance of MonoOccupancy is the flattened operation in its supervised variational automatic encoder (VAE) structure, which converts the 2D feature map into 1D vector, dropping the spatial information. Although the generated semantic BEV grid map is a form of an image, the better performance of our proposed S2G2 and MonoOccupancy indicates that there are still great gaps laying between the tasks of semantic BEV grid map prediction and the classical semantic segmentation. Therefore the semantic BEV grid map prediction needs a specially designed structure.

Table 3.5: The comparative results (%) on the baseline methods. The various semantic segmentation methods are integrated into the mean teacher framework to perform semi-supervised learning. The random Gaussian noise is added to the input images before fed into the separate networks. The bold Font highlight the best results in each column. Our proposed S2G2 outperforms the others.

Methods	10%		40%		80%	
	mIoU	mAP	mIoU	mAP	mIoU	mAP
SegNet	0.5084	0.6138	0.5224	0.5205	0.5184	0.5561
U-Net	0.4413	0.5417	0.4554	0.5050	0.4535	0.4996
RTFNet	0.5256	0.6343	0.5346	0.5985	0.5258	0.5801
HRNet	0.5539	0.6670	0.5568	0.5970	0.5535	0.5938
DeepLab V3+	0.5144	0.6060	0.5145	0.5808	0.5024	0.5923
MonoOccupancy	0.5262	0.6787	0.5338	0.6575	0.5329	0.6148
S2G2 (ours)	**0.5894**	**0.7003**	**0.5889**	**0.6956**	**0.5879**	**0.7110**

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

We also evaluate the performances of our S2G2 together with some of the state-of-the-art supervised learning-based methods, including PYVA [89], PON [88], MonoLayout [87], and MonoOccupancy [83]. It can be seen from Tab. 3.6, testing on the dataset provided by MonoOccupancy, our proposed S2G2 outperforms all the previous networks with 58.86% in mIoU and 70.23% in mAP. The second best results were produced by MonoOccupancy. We attribute it to the fact that the other methods could not adapt well to the noisy ground truth provided by the training dataset.

Moreover, we compared the performance of the MonoOccupancy in the mean teacher framework and the original one. We find that the results of the former one are inferior to the latter. The reason for this case may be that the semi-supervised semantic prediction requires strong perturbations to produce a qualified homologous similar input pair. We refer readers to [116] for more details.

Table 3.6: The comparative results (%) on the test dataset from [83]. All the comparative methods predict the semantic BEV map in a supervised manner. The table shows that our semi-supervised approach achieves the best performance.

Methods	mIoU	mAP
PYVA [89] (CVPR 2021)	0.5066	0.6219
PON [88] (CVPR 2020)	0.4883	0.6332
MonoLayout [87] (WACV 2020)	0.5307	0.6776
MonoOccupancy [83] (RA-L 2019)	0.5786	0.6513
S2G2 (ours)	**0.5886**	**0.7023**

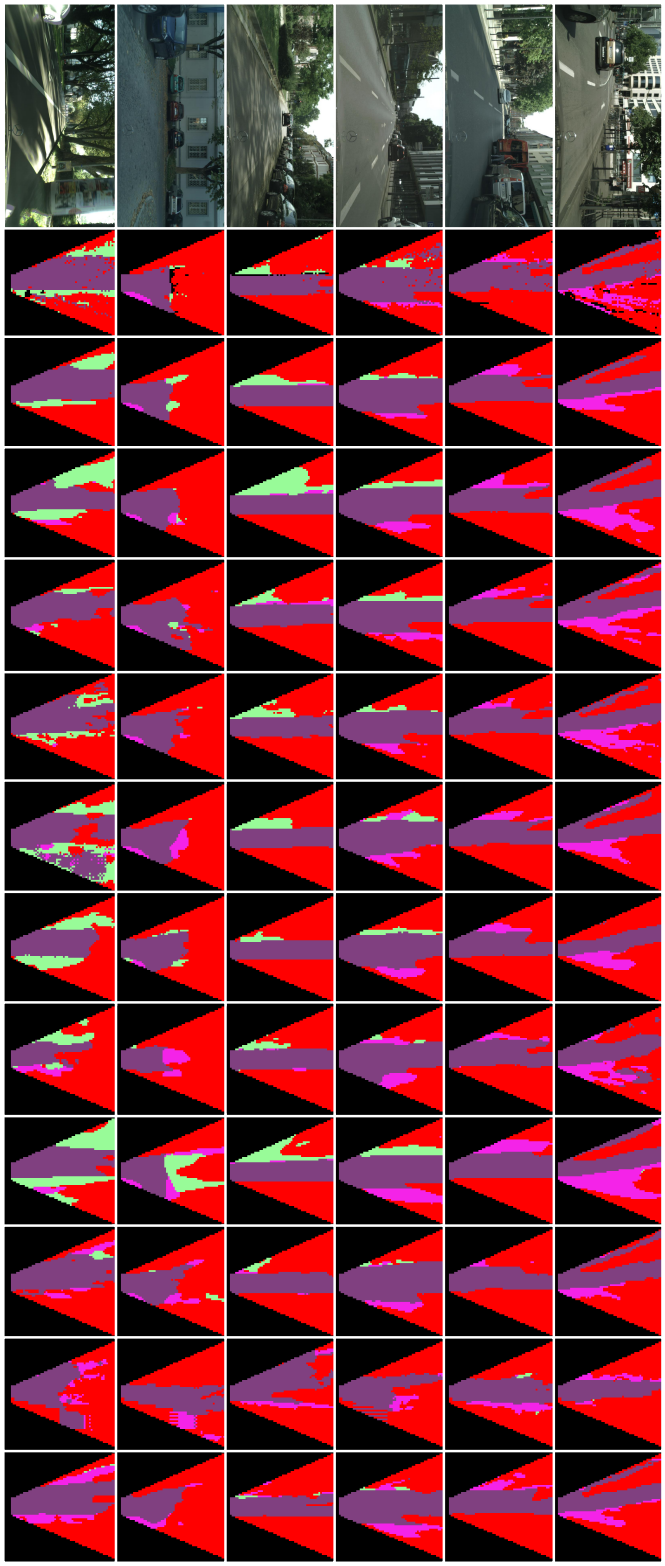


Figure 3.6: Example qualitative performances for the semantic BEV grid-map prediction networks.

3.4.4.3 The Qualitative Demonstrations

Sample qualitative semantic BEV prediction results are shown in Fig. 3.6. In general, our S2G2 generates a more precise and clear semantic BEV grid map. Note that the ground truth contains noise since they are produced by the SGM method. Even so, our S2G2 can still generate the compelling semantic BEV map. According to the results, we can see that our S2G2 is more sensitive to obstacles compared to the other methods, which is crucial for safe navigation. The last two rows display the more complicated driving environments due to the unusual road conditions and the uneven sunlight. Most other methods fail to predict the correct semantic classes, but our S2G2 still provides relatively clear and accurate semantic boundaries.

3.5 Summary of This Chapter

A semantic BEV map provides an intuitive representation of driving scene information. Existing methods typically train networks in a fully supervised manner, with performance heavily dependent on the quantity and quality of labeled samples in the training set. Due to the high cost and difficulty of annotating semantic BEV labels, existing datasets often lack sufficient reliable labeled samples, hindering the development of semantic BEV map prediction networks. To address this issue, this chapter proposes a semi-supervised semantic BEV map prediction network, termed the S2G2 network.

The effectiveness of the proposed S2G2 network was validated on the public Cityscapes dataset, utilizing mean Intersection over Union (mIoU) and mean Average Precision (mAP) as evaluation metrics. Extensive ablation experiments

were conducted to demonstrate the selection of hyperparameters and the design of the network structure. Additionally, to verify the semi-supervised capability, we established semi-supervised baseline methods using existing semantic segmentation networks. Our proposed method was compared with these baseline methods under identical experimental conditions, demonstrating superior performance in semi-supervised learning. Finally, compared to existing state-of-the-art semantic BEV map prediction methods, the S2G2 network achieved the highest segmentation accuracy in mIoU and mAP metrics, and effectively handled extreme lighting conditions and unusual road structures.

Although the proposed S2G2 network outperforms other existing methods in semantic BEV map prediction, it is still constrained by the narrow field of view of the camera, resulting in the generated BEV semantics being limited to a conical region. Therefore, in future research, we will explore improvements to the network structure to enable the prediction of a full-view semantic BEV map.

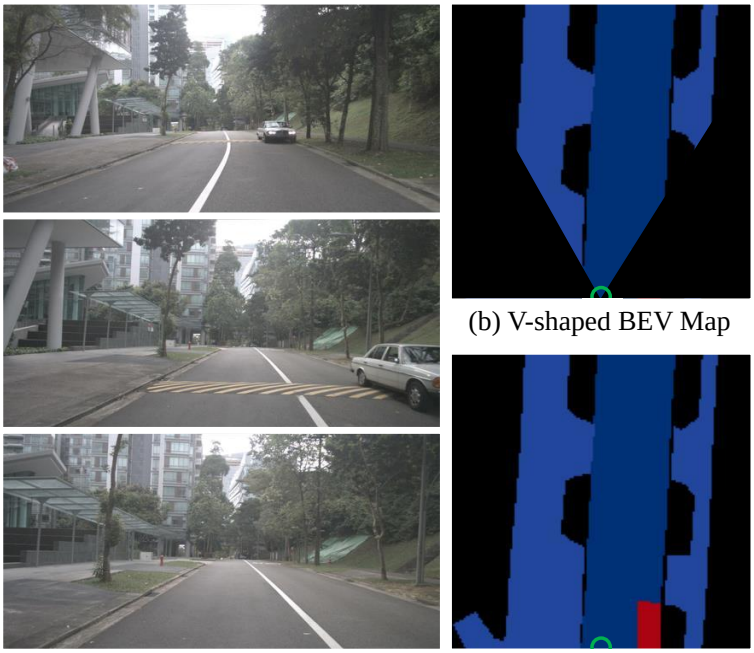
Chapter 4

Semantic BEV Map Prediction in Full View

4.1 Motivation

Semantic scene understanding is a fundamental component for autonomous driving. Suitable formats of data representation for semantic scene understanding could facilitate downstream tasks, such as motion planning [111, 112, 129] and trajectory prediction [130, 131]. Compared with semantic segmentation in front view, semantic maps in bird-eye-view (BEV) are more appropriate for the downstream tasks in autonomous driving due to the following reasons: 1) the distances and geometric relationships between the ego-vehicle and other road users can be explicitly indicated; 2) semantic BEV maps have no distortions that appear on front-view semantic segmentation maps. For example, the same object keeps the same size no matter how far the object is from the camera; 3) semantic BEV maps are high-level abstractions of the surrounding environment. So, using such maps

to train control networks for autonomous driving in simulation environments, like CARLA [7], could alleviate the domain gap issue when deploying the networks in the real world.



(a) Equidistant Sequence Images (b) V-shaped BEV Map (c) Full BEV Map

Figure 4.1: Comparison between V-shaped semantic BEV map and full BEV map.

To generate semantic BEV maps using front-view images, traditional methods involve a number of algorithms, such as geometric projection and semantic segmentation. Recently, deep learning-based end-to-end methods have shown great potential [83, 86, 132]. However, most existing methods that take as input the front-view image from a single camera suffer from the limited Field of View (FOV) issue. As illustrated in Fig. 4.1, due to the lack of visual information that is outside the FOV of the front-facing camera, the BEV map generated from a single front-view image is limited by the camera FOV, leading to a V-shaped BEV map. To get a full-view map, some work [88, 99, 133, 134] uses images from multiple

cameras around the ego-vehicle. However, those methods usually need to simultaneously process 6 or more images, which increases time and computing costs. There are also some attempts [98, 135] using the images from a temporal sequence to get more views of the environment. But in practice, the ego-vehicle could slow down or stop for a while on roads to wait for pedestrians or traffic lights. In such cases, the camera may repeatedly capture redundant images for the same scene, and existing methods could fail to get a larger view or full view when receiving these redundant images.

To provide a solution to this issue, we propose a novel network, Seq-BEV, which takes as input the equidistant sequential images and directly outputs semantic BEV maps in full view (i.e., $180°$ field-of-view). The equidistant sequential images refer to the images sampled at uniform distance intervals rather than time intervals. Our network is end-to-end trainable. It is composed of a sequence fusion module and a road-aware view transformation module. The former fuses the equidistant sequential images during the feature extraction process, and the latter transforms the front-view features into BEV with an attention mechanism.

To the best of our knowledge, our network is the first solution to use equidistant sequential images to get a semantic BEV map in full view. To train and evaluate the proposed network, we create a dataset with semantic BEV map labels in full view from the nuScenes dataset [136]. The experimental results demonstrate our effectiveness and superiority. The contributions of this work are summarized as follows:

1. We propose a novel semantic BEV map prediction network that takes as input a set of equidistant sequential images sampled at uniform distance

intervals and outputs a semantic BEV map in full view.

2. We design a new self-adapted sequence fusion module to fuse the features from different images, which provides complementary information to get more views.

3. We provide a new method for view transformation by first extracting the attention of road planes and then projecting attention-based features to BEV.

4. We create a dataset with semantic ground truth labels in full view from the nuScenes dataset to train and test our method. The code and dataset will be available upon acceptance of this paper.

4.2 Background

The proposed Seq-BEV network primarily employs the Inverse Perspective Mapping (IPM) algorithm to perform view transformations. To facilitate a better understanding of our proposed network, we will now introduce the fundamental concepts of the homography matrix and the IPM algorithm.

4.2.1 Homography Matrix

Homography is a crucial concept in the field of computer vision, used to describe the relationship between images captured by a camera viewing the same planar object from different positions. As illustrated in Fig. 4.2, the camera captures images of the same object from perspectives A and B. Due to the differing viewpoints, the same object produces distinct projections on image planes A and B. There is a specific correlation between these two images, such that through homography

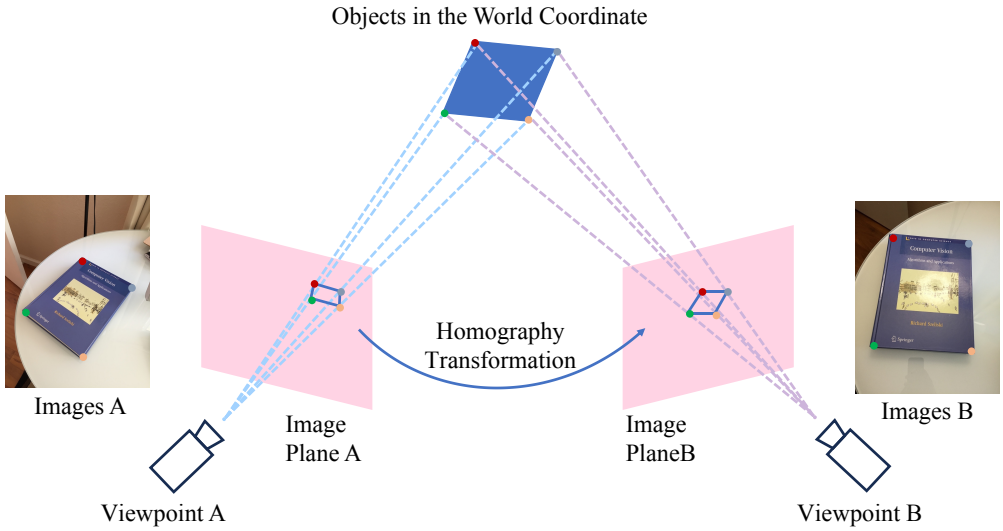


Figure 4.2: An example for the homography matrix between two images

transformation, points on the same plane in one image can be mapped to the corresponding points in another image. This transformation process is described by the homography matrix H:

$$H = \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \tag{4.1}$$

Eq. 4.2 represents the process of mapping the pixel $p_1(u_1, v_1)$ in image A to its corresponding point $p_2(u_2, v_2)$ in image B using a homography matrix:

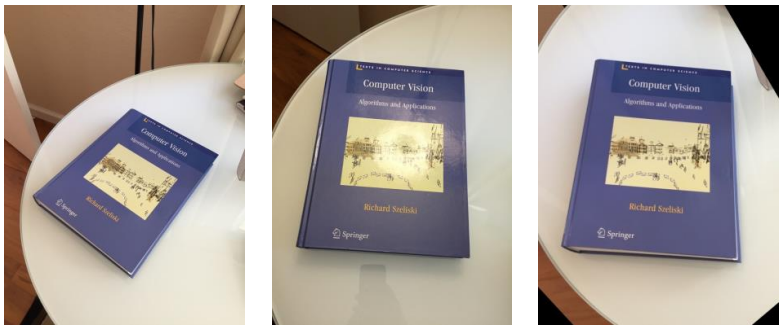
$$\begin{bmatrix} u_2 \\ v_2 \\ 1 \end{bmatrix} = H \begin{bmatrix} u_1 \\ v_1 \\ 1 \end{bmatrix} = \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} u_1 \\ v_1 \\ 1 \end{bmatrix} = \begin{bmatrix} h_1u_1 + h_2v_1 + h_3 \\ h_4u_1 + h_5v_1 + h_6 \\ h_7u_1 + h_8v_1 + h_9 \end{bmatrix} \tag{4.2}$$

The pixel p_1 and p_2 are represented in Homogeneous coordinates. Typically, the homography matrix is scaled by a non-zero factor, such that the last element h_9 in matrix H is equal to 1. Then, according to the third row in the matrix, this non-zero factor is eliminated, resulting in:

$$u_2 = \frac{h_1 u_1 + h_2 v_1 + h_3}{h_7 u_1 + h_8 v_1 + h_9} \tag{4.3}$$

$$v_2 = \frac{h_4 u_1 + h_5 v_1 + h_6}{h_7 u_1 + h_8 v_1 + h_9} \tag{4.4}$$

The corresponding points in the two images is a set of matching points. According to Eq. 4.3 and 4.4, each set of matching points can construct two constraint terms for solving the homography matrix H. The homography matrix H is a 3×3 matrix, but since $h_9 = 1$, it only has 8 degrees of freedom. Therefore, we need four sets of non collinear matching points to solve the homography matrix between two planes.



(a) (b) (c)

Figure 4.3: An example for the homography transform: (a) is the source image; (b) shows the target image; (c) is the image after the homography transform.

As shown in Fig. 4.2, we have marked four different sets of matching points in images A and B, with the same color. Using these four sets of points, we can

solve for the homography matrix between the planes where these two books are located. Through this matrix, we can calculate the corresponding position of each pixel in image A from perspective B. The images before and after homography transformation are shown in Fig. 4.3. Among them, (a) displays image A, as the source image before transformation; (b) illustrates the target image B. We use homography matrix to transform the perspective of image A to that of image B, and (c) is the transformed image in the target perspective.

4.2.2 Inverse Perspective Mapping

Due to the perspective effects, the parallel lines in the real world appear to intersect with each other in a perspective image. IPM algorithm is a method for eliminating these perspective distortions. By utilizing a homography matrix, it transforms an image from a front view to a bird's-eye view.

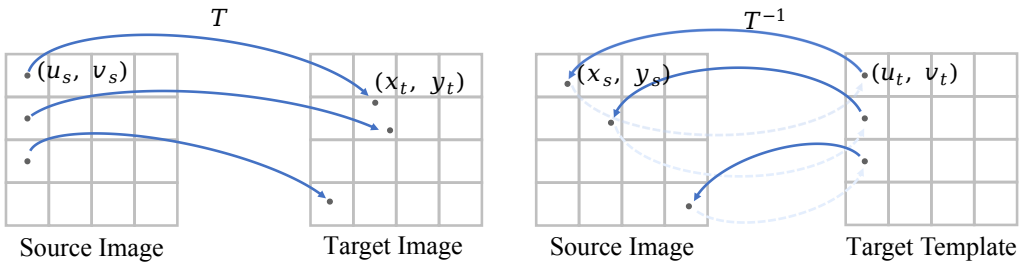


Figure 4.4: Forward mapping and inverse mapping

The transformation process from the source image to the target image depicted in Sec. 4.2.1 is known as forward mapping, as illustrated in Fig. 4.4 (a). Different from this forward process, the IPM algorithm first calculate the inverse transformation matrix from the target space to the source. To distinguish it from the forward

transformation matrix, we use T^{-1} to represent the inverse transformation matrix. By using T^{-1}, the corresponding position (x_s, y_s) of the pixel (u_t, v_t) in the image template from the target perspective can be found in the source image. Finally, interpolation is performed on the source image based on the pixel position (x_s, y_s) to obtain the target pixel value. Similar to obtaining the forward transformation matrix, the inverse transformation matrix can also be calculated using four sets of corresponding points.

The inverse transformation begins from the target perspective and calculates the source image position according to each pixel in the target template:

$$(x_s, y_s) = T^{-1}(u_t, v_t), \tag{4.5}$$

This reverse calculation can prevent issues such as holes and overlaps that occur in forward transformation, ensuring that each pixel in the target image template finds its corresponding pixel value in the source image.

(a) Front View Image (b) BEV Image After IPM

Figure 4.5: The front-view image of the road and that in bird-eye view after IPM

Similar to homography transformation, IPM is also constrained by the plane assumption. This means that IPM can only accurately transform pixels on the same plane to another perspective, leading to severe deformation in non-planar scenes.

As illustrated in Fig. 4.5, we perform IPM on an image captured by the front camera, mapping the road plane in the front image to a bird's-eye view. The blue line segments in the figure, which intersect at a distance in the forward view, are restored to their original parallel state in the transformed bird's-eye view. However, objects such as cars that do not lie on the road surface (indicated by the red area) undergo significant deformation in the bird's-eye view due to the flat surface assumption.

4.3 The Proposed Network

4.3.1 The Overall Architecture

Our motivation is to generate semantic BEV maps in full view using sequential images that are sampled at equal distances. The structure of our network is illustrated in Fig. 4.6. As we can see, our network has two streams that respectively take as input single and sequential images at the same time. The two kinds of inputs are respectively fed into the spatial and sequential encoders, where the low-level features and high-level features are extracted at different stages of the encoder.

To complement the vision information from the equidistant sequential images and get the full view, we design a self-adapted sequence fusion module in the sequential encoder. This is a parameter-free module that can directly manipulate the feature tensor. The spatial low-level feature and the sequential low-level feature are fused into the S&S fusion feature via convolution operation. The S&S refers to spatial and sequential. The road-aware view transformation module takes as input the S&S fusion feature and computes the road layout attention under explicit

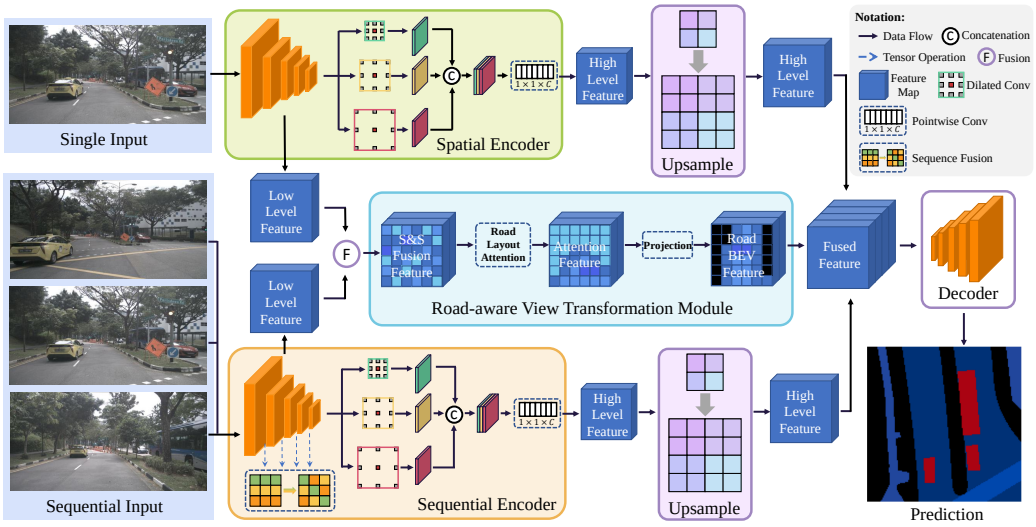


Figure 4.6: The overall architecture of our proposed Seq-BEV network

supervision before projecting it into the road BEV feature. We concatenate the high-level features and the road BEV feature, producing the fused feature, and then send it to the decoder to get the semantic BEV map in full view.

4.3.2 The Two-stream Encoder

We design a two-stream encoder to extract the spatial and sequential features, respectively. In such a way, the spatial integrity of the images can be preserved when performing the equidistant sequence fusion. The structures of the spatial and sequential encoders are similar. The only difference is that there is a sequence fusion module in the sequential encoder. So, in the following text, we do not particularly distinguish the spatial encoder from the sequential encoder and briefly term both of them as encoder.

We use the DeepLab V3+ [44] to extract the features and choose MobileNet V2 [137] as the backbone. MobileNet V2 is a lightweight network that requires relatively

few computation resources. In the encoder, we take out the low-level features $\mathcal{F}_L$ from the low stage of the backbone and the high-level features $\mathcal{F}_H$ from the high stage of the backbone. The low-level features keep the higher resolution and richer spatial information but contain relatively less semantic information. The spatial information covered in the low-level features, such as geometrical structure, could be helpful for our road layout attention extraction.

In contrast, the high-level features encode more semantic information, and these high-level features could be invariant in scale because they pass through the multi-scale dilated convolutions. In the front-view image, the same-sized objects may have various scales due to the different distances from the camera. So, the high-level features would be more suitable for sensing objects on roads. It should be noted that the low- and high-level features are extracted from the $3rd$ and $17th$ layers of the backbone, respectively. The $3rd$ layer corresponds to the one following the first downsampling operation, while the $17th$ layer represents the final layer of the feature extractor.

4.3.3 The Sequence Fusion Module

We design a self-adapted sequence fusion module to fuse the complementary information from the equidistant sequential images and get a semantic BEV map in full view. This module fuses the sequential features by applying varying degrees of grouping and shifting operations, based on the number of training iterations. The input and output sizes (i.e., channel and resolution) of the self-adapted sequence fusion module are the same, thus, it can be inserted into existing networks seamlessly.

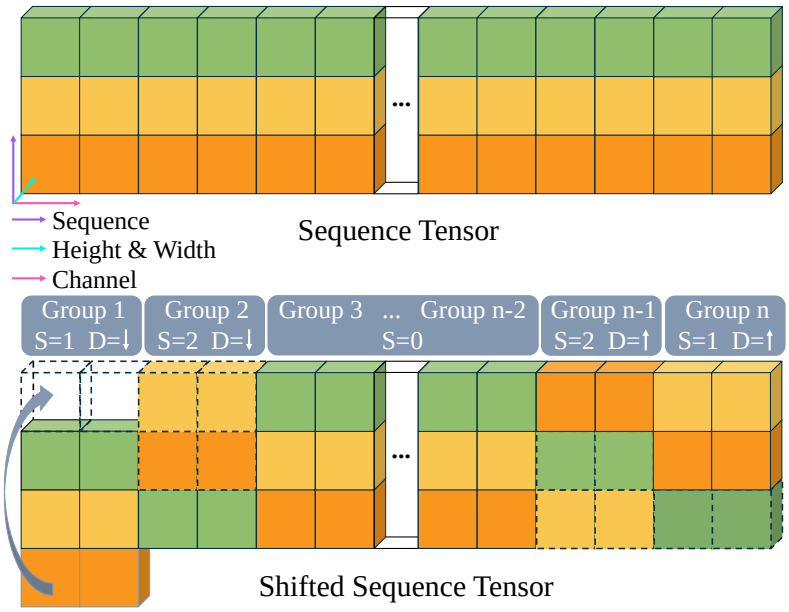


Figure 4.7: The demonstration of the sequence fusion module

The self-adapted sequence fusion module is shown in Fig. 4.7. The sequential feature is a 4 dimension tensor, which is denoted as $\mathcal{F}_{seq} \in \mathbb{R}^{B\times S\times C\times H\times W}$, where B, S, C, H, W are the batch size, numbers of the images in the sequence, number of channels, height, and width, respectively. The example provided shows a sequence of three frames, with different colors representing each one. The fusing operation is inspired by TSM [138]. The channel dimension is divided into groups, each containing a subset of the image features. As shown in Fig. 4.7, we assign two channels per group for demonstration. The groups are then shifted according to a defined principle to fuse the features across frames. However, TSM relies on a fixed-size grouping (e.g., $1/8$ of the channel dimension), where the group size is a pre-defined parameter. Selecting an inappropriate group size can result in suboptimal outcomes.

During experimentation, we observed that as the network training progresses,

its feature extraction capacity improves, and a larger shift portion becomes necessary for more effective fusion. In response to this observation, we introduced a self-adaptive mechanism to the sequence fusion process. First, we divide the channel dimension into n groups $\{G^1, G^2, ..., G^n\}$. The proposed self-adapted fusion method dynamically adjusts the number of channels within each group according to the training iterations. The number of channels in each group is determined by the following calculation:

$$\alpha = 1/\min\{\lfloor C/N \rfloor, \max(\lfloor \lambda I_t / I_c \rfloor, 1)\}, \tag{4.6}$$

$$m = \alpha C/N, \tag{4.7}$$

where α is the grouping coefficient, which is a dynamic parameter intended to adjust the channel number in a group according to the current number of iterations. N and C are the minimum number of groups and the total channel number of the current feature tensor, respectively. The total number and the current number of the iterations are denoted as I_t and I_c, respectively. λ is a parameter that indicates the reliability of the network feature extraction capability. The larger λ is, the fewer channels are assigned to each group. Here, we set λ as 0.5 empirically. $\lfloor \cdot \rfloor$ represents the round down operation. m is the current channel number in a group.

As illustrated in Fig. 4.7, we move the individual group with different strides in either up or down directions after figuring out m. This module does not consume extra computation costs because the self-adapted grouping and shifting operations need no learnable parameters. We conduct the sequence fusion at the $2nd$, $4th$, $7th$, and $14th$ layer of the backbone network before the downsampling operations. It is worth noting that the shifting operation exchanges the channels among different

frames, breaking the feature completeness of an individual frame in terms of the spatial dimension. So, we employ a spatial encoder to extract the feature map from a single image for better spatial modeling.

4.3.4 The Road-aware View Transformation Module

Most current view transformation approaches employ data-driven methods to generate BEV maps, relying on the complex mapping relationships learned by deep neural networks. However, this process often lacks interpretability. To achieve a more reliable and explainable view transformation, we project front-view features onto the BEV plane using a learnable homography transformation, which is applied after extracting attention from the road surface. The use of a learnable homography enhances the interpretability of the transformation while incorporating road-aware features helps mitigate the distortion commonly associated with the flat-ground assumption[82].

To acquire an accurate road layout during the view transformation, we conduct a road attention extraction in an auxiliary supervision manner before projecting the front view feature into the bird-eye view. As shown in Fig. 4.8, the road-aware view transformation module consists of the auxiliary road attention extractor and the learning-based Spatial Transformer Network (STN).

The auxiliary road attention extractor employs SENet [139] to emphasize the informative components in the feature map. The input feature $\mathcal{F}_{in}$, which is produced by the low-level encoders, is fed into the extractor, and through a squeeze operation compresses the global spatial information into a $1 \times 1 \times C$ feature $\mathcal{F}_{sq}$, where C refers to the number of channels. The following excitation operation

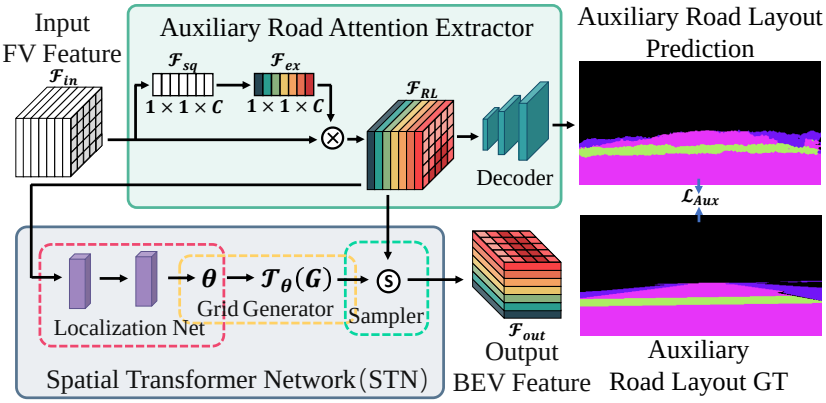


Figure 4.8: The pipeline of the road-aware view transformation module

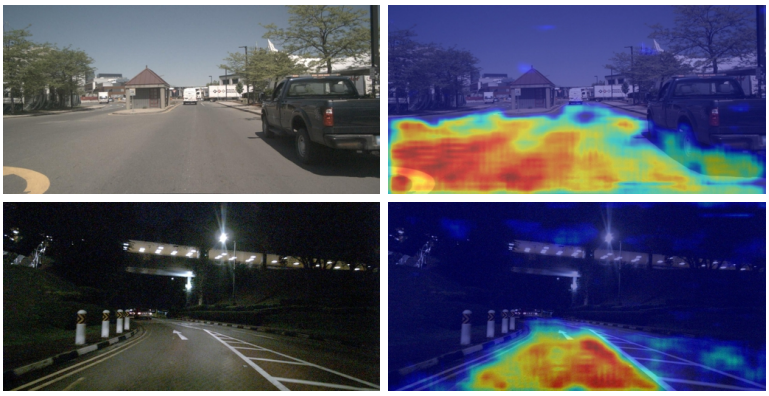


Figure 4.9: Qualitative demonstrations of the attention extracted by the road attention extractor

captures the channel-wise relations in $\mathcal{F}_{sq}$ via two fully connected (FC) layers and generates $\mathcal{F}_{ex}$. $\mathcal{F}_{ex}$ can be seen as a set of channel weights that indicates the salient features with a high score. Finally, the informative components are selected by multiplying $\mathcal{F}_{in}$ and $\mathcal{F}_{ex}$. The above steps can be formulated as:

$$\mathcal{F}_{sq} = \frac{1}{H' \times W'} \sum_{u=1}^{H'} \sum_{v=1}^{W'} \mathcal{F}_{in}(u, v), \tag{4.8}$$

$$\mathcal{F}_{ex} = FC(\mathcal{F}_{sq}, \mathbf{W}), \tag{4.9}$$

$$\mathcal{F}_{RL} = \mathcal{F}_{in} \otimes \mathcal{F}_{ex}, \tag{4.10}$$

where H' and W' are the height and width of the input feature map $\mathcal{F}_{in}$. The fully-connected operation is denoted as $FC(\cdot)$ and $\mathbf{W}$ is the learnable parameter. $\otimes$ represents element-wise multiplication. Fig. 4.9 qualitatively demonstrates sample salient features extracted by the attention mechanism. To get a more reliable road layout segmentation, we conduct this attention extraction under the auxiliary road layout supervision.

The view transformation is implemented on the feature map $\mathcal{F}_{RL}$, which encodes the road plane attention. The STN [85] is employed to regress a 3×3 projection matrix θ via the localization net. Then, with the projection matrix, the grid generator creates a sampling grid before sending it to the sampler. The sampler samples $\mathcal{F}_{RL}$ at the sampling grid points. We refer readers to STN [85] for more details. Usually, the geometric projection methods suffer from the flat ground assumption [82], leading to distortions for objects above roads or distortions for roads that are not flat. But our proposed view transformation method focuses on the road plane through the attention mechanism before the projection, which alleviates the limitation of the flat plane assumption.

4.3.5 Loss Functions

We use two losses in this work. One is the auxiliary loss $\mathcal{L}_{Aux}$, which supervises the road attention extractor. The other is the BEV loss $\mathcal{L}_{BEV}$, which enables the network to produce semantic BEV maps. Due to the class imbalance issue in the dataset, the Focal Loss [140] is used as $\mathcal{L}_{Aux}$ and $\mathcal{L}_{BEV}$. We train our network in

an end-to-end manner. The overall loss function is formulated as:

$$\mathcal{L} = \mathcal{L}_{BEV} + \gamma \mathcal{L}_{Aux}, \tag{4.11}$$

where γ is the weighting parameter to balance the two losses. We empirically set γ as 1.

4.4 Experimnetal Results and Discussions

4.4.1 The Dataset

The semantic BEV data used in this chapter is created based on the publicly available autonomous driving dataset, nuScenes [136]. We present an image example of the dataset we used in Fig. 4.10. The nuScenes dataset collected 850 road scene segments in different areas of Singapore and Boston, each lasting approximately 20 seconds and containing around 30 frames of images. The dataset also provides different forms of ground truth, including: (1) the 3D bounding box for point cloud recognition tasks, through which we can obtain the position information, size, and category information of target objects in the driving scene (as shown in Fig. 4.10 (b)); (2) The semantic segmentation map of the front view image, which divides the image into different semantic regions based on the categories of objects in the image; (3) The High-Definition (HD) map covering the entire collection scene, consists of basic units of different shapes' polygons, representing the road surface semantics. We refer to these polygons containing semantics as road semantic masks. Multiple different types of semantic masks are stacked and concatenated to form a high-precision map of the entire scene (as shown in Fig. 4.10 (c))

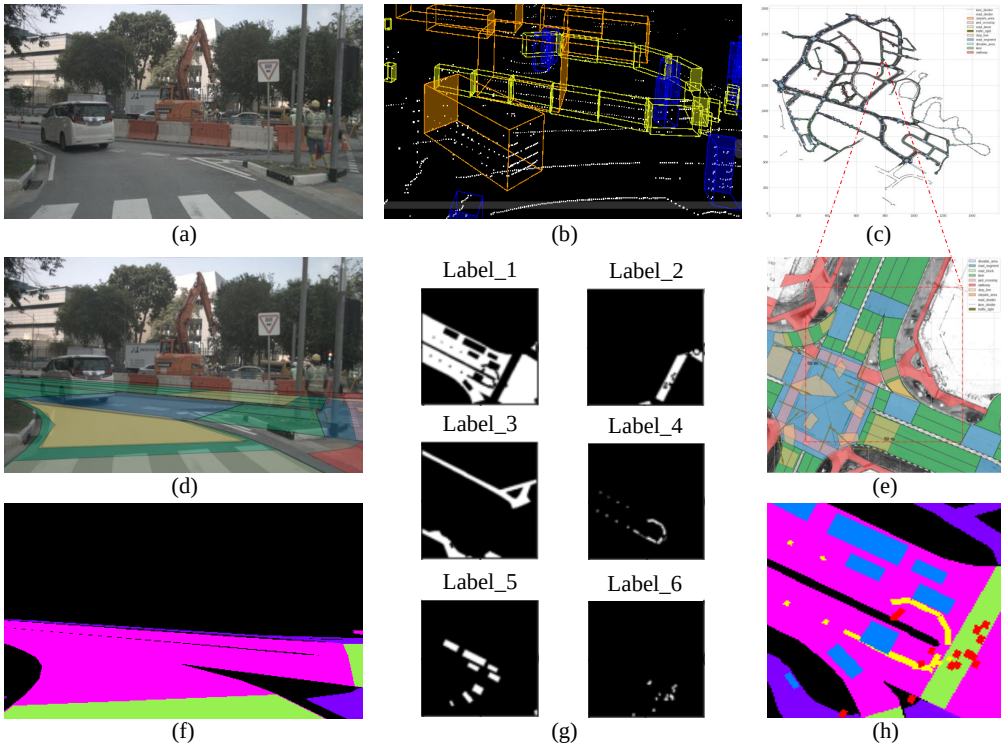


Figure 4.10: Examples from the dataset: (a) original front-view image; (b) The 3D bounding box for the point cloud detection; (c) The HD map for the whole city; (d) The road semantics projected in the front-view image; (e) The map patch for the current scene; (f) The road semantic segmentation in the front view; (g) The different types of BEV semantic masks; (h) The semantic BEV label.

To creating semantic BEV labels, we first identify the area around the autonomous vehicle in the HD map based on the ego-pose information. We then extract and frame the relevant map segments around the autonomous vehicle. This process is shown in Fig. 4.10 (e). The angle of the selected part is adjusted to ensure that the vehicle's front direction points north. Next, road masks for the required semantic categories are selected from these map segments. Overlaps between semantic masks are removed, and semantic IDs are assigned to different semantic categories, as shown in Fig. 4.10 (g). This process yields a semantic map of the road from a bird's-eye view. Subsequently, 3D bounding boxes are pro-

jected onto the semantic map, and new semantic IDs are assigned to road objects. This results in the semantic BEV labels depicted in Fig. 4.10 (h). The semantic labels used in this chapter encompass seven categories: background, drivable area, sidewalk, pedestrian crossing, vehicle, obstacle, and pedestrian.

The road semantic labels in the front view is also utilized in the proposed methods as auxiliary supervision for road attention extracion. Therefore, we projected the road semantic masks from the HD map onto the front view image to obtain the image shown in Fig. 4.10 (d). By encoding different semantics, we generated the semantic labels of the road plane in the front view image, as illustrated in Fig. 4.10 (f). The road semantic labels used in this chapter include four categories: background class, drivable area, sidewalk, and pedestrian crossing.

Compared to other semantic BEV labels, the dataset created in this chapter overcomes the limitations of v-shaped view. Additionally, the original dataset organizes data using database tables, assigning unique tokens to each sample for retrieval. We have modified the data organization by storing the images used for training, validation, and testing in their respective folders. During training and testing, the images are read directly from these folders as inputs to the network, thereby reducing data reading time. It should be noted that in the process of creating semantic labels for front-view road surfaces, semantic fragments from HD maps are projected onto front-view images. If the road surface is uneven, this can lead to inaccurate semantic projection. Therefore, we manually removed scenes with obvious projection errors, and divided the entire dataset into 548 training sets, 150 validation sets, and 148 testing sets. To achieve equal-distance sampling, three images with equal intervals are sampled using the ego-pose information provided by the dataset to construct a sequence. We set three dataset groups with different

distance intervals for the experiment. Specifically, each group with the sequence comprises three images acquired at distances of 10, 20, and 30 meters, respectively, or at angular displacements exceeding $30°$. The input images are normalized to the resolution of 256×512 pixels, and the size of the output semantic labels is 150×150 pixels, where each pixel encodes a semantic area of 0.2×0.2 square meters.

4.4.2 Training Details

We train our network with NVIDIA GeForce RTX 3090. To balance the memory consumption and the time cost, we set the batch size to 8. We train our network for 50 epochs using the AdamW optimizer [141]. We initialize the learning rate as 5×10^{-4} and adopt the cosine annealing scheme [142] to adjust the learning rate during training. The warm-up strategy is employed for the learning rate adjustment. This strategy gradually increases the learning rate until the preset epoch for warm up ends (the $20th$ epoch in our network), and then decreases the learning rate according to the decay scheme. The momentum and weight decay are set to 0.9 and 5×10^{-4}, respectively. MobileNet V2 is used as our backbone and initialized with the pre-trained weight. The rest of our network parameters are initialized randomly. The sequence fusion is inserted into the $2nd$, $4th$, $7th$, and $14th$ block of the backbone, and the minimum number of groups for the self-adapt grouping N is set to 24.

4.4.3 Ablation Study

To verify the effectiveness of our network structure and to choose appropriate parameters for our network, we conduct several ablation experiments. The mean Intersection over Union (mIoU) and the mean Average Precision (mAP) are employed to quantitatively evaluate the performance of our network.

4.4.3.1 Ablation on Backbone

We compare the performance of our network with different backbones in the encoders, including MobileNet V2 [137], MobileNet V3 [143], ResNet family [28], and Xception [123]. Similar to our proposed Seq-BEV, we modify each backbone to get the low-level feature and the high-level feature and also insert the multiple sequence fusion modules into the backbones.

Fig 4.11 demonstrates the results, which shows the trade-off between the network performance and the number of parameters. The network runtime is assessed in terms of Frames Per Second (FPS) on the RTX 3090 GPU and represented visually by the hollow purple circle, whose area is inversely proportional to the number of network parameters. It is observed that MobileNet V2, which has the fewest parameters, achieves a frame rate of 28.04 FPS while delivering satisfactory performance. So, MobileNet V2 is chosen as our backbone.

4.4.3.2 Ablation on Different Variants

This ablation study is divided into two groups, which takes as input single (SGL) image and equidistant sequential (SEQ) images, respectively. Note that the self-adapted sequence fusion module is removed from the SGL group. For the first

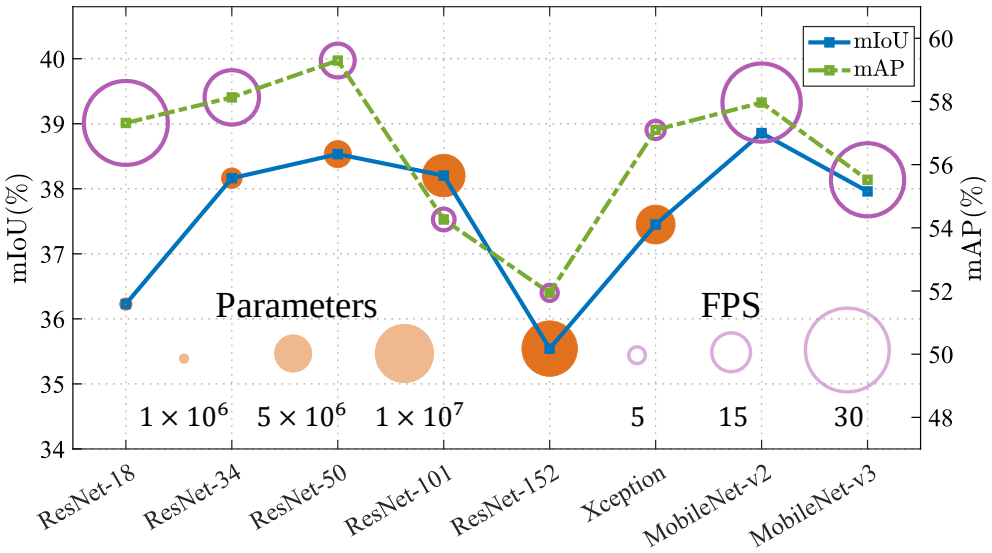


Figure 4.11: Impacts caused by backbone selection in terms of mIoU and mAP. The blue solid line indicates the results measured by mIoU, while the green dotted line corresponds to the mAP measurements. Additionally, the area of the solid orange circle reflects the number of parameters within the network for various backbone architectures. The area of the hollow purple circle represents the FPS performance of each respective backbone. The figure is best viewed in color.

group, we employ the semantic segmentation network DeepLab V3+ [44] as the baseline. For the second group, we add the plain STN to the baseline as our view transformation module. We term this variant as PLVT. For the third group, we integrate the road-attention extractor with the view transformation to test the performance of the road-aware mechanism. We name this road-aware variant RAVT.

Tab. 4.1 displays the results. We can see that our proposed Seq-BEV (the SEQ-RAVT variant), which contains the sequential input fusion module and the road-aware view transformation module, achieves the best performance in terms of both mIoU and mAP. The data in the table leads to the conclusion that all the variants that take as input the sequential images get a superior performance against

Table 4.1: The ablation study results (%) on different variants. There are two groups of tests, whose input is the single image (SGL) and the sequential images (SEQ) respectively. In each group, we compare the results from the three variants, which are the semantic segmentation baseline method, the plain view transformation variant, and the road-aware one. Those variants are denoted as Baseline, PLVT, and RAVT in this table.

Variants	Background		Drivable Area		Ped. Crossing		WalkWay		Obstacle		Vehicle		Pedestrain		mIoU	mAP
	IoU	AP	IoU	AP	IoU	AP	IoU	AP	IoU	AP	IoU	AP	IoU	AP		
SGL-Baseline	56.36	69.67	64.67	76.91	16.75	34.73	31.54	52.64	10.67	23.27	15.85	36.18	**1.08**	**4.55**	28.13	42.56
SGL-PLVT	56.84	70.22	65.38	**77.47**	18.04	41.36	32.27	**53.85**	**11.26**	**36.24**	**16.52**	33.38	0.94	3.89	**28.75**	45.20
SGL-RAVT	**57.17**	**71.40**	**66.16**	77.28	**18.50**	**45.04**	**32.69**	52.33	11.09	36.13	15.07	**36.90**	0.43	2.71	28.73	**45.97**
SEQ-Baseline	62.96	73.98	70.86	82.51	26.28	57.08	39.31	58.75	16.79	39.05	17.73	40.69	0.51	4.57	33.49	50.95
SEQ-PLVT	62.36	74.45	70.82	81.25	27.02	56.37	39.28	60.13	17.74	40.68	17.68	42.94	0.81	3.49	33.67	51.33
SEQ-RAVT(Ours)	**66.59**	**76.85**	**74.89**	**85.19**	**35.76**	**65.27**	**43.73**	**64.72**	**22.27**	**52.43**	**27.63**	**54.32**	**1.12**	**7.24**	**38.86**	**57.96**

their counterparts. Comparing PLVT and RAVT, we can see that the prediction performance increases due to the incorporation of road attention.

4.4.3.3 Ablation on Sequence Fusion Module

We fuse the information of different images from a sequence by shifting the sequential channel in a feature map. Since this fusion module directly manipulates the feature tensor without any learnable parameters, it can be flexibly inserted into any position of a CNN. So, the insert position of this sequence fusion module needs to be chosen.

In our Seq-BEV, we select MobileNet V2 as our backbone network. The MobileNet V2 is stacked by the inverted residual blocks, which consist of an expansion layer, a depthwise layer, and a projection layer. In this ablation study, we test the performance of the Seq-BEV with the sequence fusion module inserted before different layers.

In addition, we design a self-adapted mechanism to group different channel numbers in the sequence dimension according to the training process. Here, we also compare the performance of our proposed self-adapted grouping strategy with that of the fixed grouping method. We set 3 fixed groups, dividing the sequence channel into 8, 16, and 24 groups, respectively.

Tab. 4.2 displays the results. From the table, we can see that the self-adapted grouping strategy improves the network performance in terms of mIoU. Moreover, the network achieves the best results when we insert the sequence fusion module before the depthwise layer. We conjecture the reason for this superior performance is the expansion operation in the expansion layer, which extends the dimension of the feature map. It can be seen as a process of data decompression. Thus, the

Table 4.2: The ablation study results (%) on the sequence fusion module. We test the networks with different insertion positions and the sequence channel grouping schemes at the same time. The sequence fusion module is inserted before the expansion layer, depthwise layer, and projection layer, respectively. To test the performance of the designed self-adapted grouping mechanism, we compare it with 3 fixed groups of the sequence channel, including 8, 16, and 24 groups.

Grouping	Expansion Layer		Depthwise Layer		Projection Layer	
	mIoU	mAP	mIoU	mAP	mIoU	mAP
8-Groups	36.43	56.01	37.42	57.60	37.24	57.41
16-Groups	37.30	57.70	37.87	58.07	37.81	57.73
24-Groups	37.63	58.33	38.08	58.72	37.97	57.40
Self-adapted	37.75	56.93	38.86	57.96	38.58	57.44

feature map produced by the expansion layer could provide enough information for the sequence fusion.

4.4.3.4 Ablation on Network Structure

In this ablation study, we first conduct experiments to determine the best way to combine the road BEV feature and the high-level features. Then, we adjust the input feature map of the road-aware view transformation module to choose the most effective Seq-BEV structure. In addition, an ablation study is conducted to determine the optimal loss weighting factor, denoted as γ, for appropriately balancing the BEV loss, $\mathcal{L}_{BEV}$, and the auxiliary loss, $\mathcal{L}_{Aux}$.

Element-wise addition and concatenation are two common ways to combine the separate features. We report the results of the network that adopts the two combination methods in Tab. 4.4. Note that in order to keep the feature dimension unchanged, a convolution layer is applied after the concatenation operation. According to the results from Tab.4.4, the concatenation method gets the best per-

Table 4.3: The ablation study results (%) on the loss weight factors γ. We set it to 0.5, 0.1, 1.0, 1.5, 2.0, and 10.0.

Weight	0.1	0.5	1.0	1.5	2.0	10
mIoU	38.38	38.50	**38.86**	38.33	38.53	38.01
mAP	58.82	59.72	57.96	58.31	57.53	**59.75**

formance in terms of mIoU and mAP. We also note that the element-wise addition method performs better in the segmentation of small objects on the road, like obstacles, vehicles, and pedestrians. The reason for this case may be that compared with the obvious road feature, those small object features become negligible ones during the convolution operation in the concatenation method, leading to inferior performance.

We use the low-level feature as the input of the road-aware view transformation module. The low-level feature maps preserve the high resolution and hence encode rich spatial information, such as geometrical structure, which may be suitable for road layout extraction. To verify this idea, we change the input of the road-aware view transformation module to the high-level feature or both the low-level and high-level features. Tab. 4.5 shows the experiment results. We can see that the road-aware view transformation module conducted on the low-level feature has the higher mIoU with 38.86%, compared with the others. This is also true for the metric mAP. This result validates the applicability of low-level features to road attention extraction.

The entire network is trained under the supervision of both the BEV loss, $\mathcal{L}_{BEV}$, and an auxiliary loss, $\mathcal{L}_{Aux}$. A suitable weighting factor can balance the influence of these losses on the training process. Tab. 4.3 presents the network's performance across varying loss weights, based on which we assign a value of 1.0

to this factor.

4.4.3.5 Ablation on the Distance Intervals

The distance sequence employed in our network is designed to address blind spots resulting from the limited field of view. These distance intervals can be adjusted as long as the sequential images capture environmental details beyond the frame's visual range. To evaluate the effectiveness of the sequence fusion and determine the optimal distance configuration for processing inputs, we conducted an ablation study on various distance intervals. In this experiment, we utilized intervals of 10, 20, and 30 meters.

The experiment results are displayed in Tab. 4.6, demonstrating that the network achieves the highest performance when processing images at 10-meter intervals. These findings suggest that increasing the distance between images may lead to a decline in the accuracy of generating the full-view semantic BEV map, as larger intervals fail to capture the necessary environmental information in blind spots.

4.4.4 Comparative Results

4.4.4.1 The Quantitative Results

We evaluate the performance of our Seq-BEV with some state-of-the-art semantic BEV prediction methods, including Cross-view Transformation (CVT) [89], Variational Encoder-Decoder Networks (VED) [83], MonoLayout [87], and View Parsing Network (VPN) [86]. Those methods originally take a single image as input. In order to compare them with our Seq-BEV, we modify those networks

Table 4.4: The ablation study results (%) on the combination method of the road BEV feature and the high-level feature. We apply element-wise addition and concatenation to combine the two features, respectively. In order to maintain the same channel size of the feature map produced by the separate methods, we use the convolution layer after the concatenate operation.

Combine Method	Background		Drivable Area		Ped. Crossing		WalkWay		Obstacle		Vehicle		Pedestrain		mIoU	mAP
	IoU	AP	IoU	AP	IoU	AP	IoU	AP	IoU	AP	IoU	AP	IoU	AP		
Element-wise Add	65.21	75.48	73.83	83.41	30.07	64.89	42.38	61.86	17.36	**55.73**	17.87	**55.57**	0.25	**8.59**	35.28	57.93
Concat & Conv	**66.59**	**76.58**	**74.89**	**85.19**	**35.76**	**65.27**	**43.73**	**64.72**	**22.27**	52.43	**27.63**	54.32	**1.12**	7.24	**38.86**	**57.96**

Table 4.5: The ablation study results (%) on the input feature of the road-aware view transformation module. We conduct this experiment by sending the high-level feature, low-level feature, and both of them to the road layout attention extractor. The network that takes as input the low-level feature gets the best performance, which implies that the low-level feature encodes rich spatial information and is suitable for this task.

Input Feature	Background		Drivable Area		Ped. Crossing		WalkWay		Obstacle		Vehicle		Pedestrain		mIoU	mAP
	IoU	AP	IoU	AP	IoU	AP	IoU	AP	IoU	AP	IoU	AP	IoU	AP		
High-level	65.66	74.88	74.23	**85.62**	34.74	64.69	42.69	62.60	18.64	49.92	25.33	53.78	0.72	**9.35**	37.43	57.26
Low-level (ours)	**66.59**	76.58	**74.89**	85.19	**35.76**	**65.27**	**43.73**	**64.72**	**22.27**	**52.43**	**27.63**	**54.32**	**1.12**	7.24	**38.86**	**57.96**
Both	65.78	**77.21**	74.38	84.31	34.28	62.00	43.40	63.34	19.50	49.98	27.62	52.82	0.96	7.48	37.99	56.73

Table 4.6: The ablation study results (%) on the distance intervals. Seq-BEV network processes the images at specific distance intervals to capture environmental details beyond the frame's visual range. In this analysis, distance intervals of 10, 20, and 30 meters is used to identify the optimal configuration.

Distance Interval	Background		Drivable Area		Ped. Crossing		WalkWay		Obstacle		Vehicle		Pedestrain		mIoU	mAP
	IoU	AP	IoU	AP	IoU	AP	IoU	AP	IoU	AP	IoU	AP	IoU	AP		
10-meter	**66.59**	**76.85**	**74.89**	**85.19**	**35.76**	**65.27**	**43.73**	**64.72**	**22.27**	**52.43**	**27.63**	**54.32**	**1.12**	**7.24**	**38.86**	**57.96**
20-meter	65.15	75.40	73.48	84.34	33.00	62.10	40.23	61.01	16.03	41.91	25.20	51.64	0.52	4.68	36.23	54.44
30-meter	63.12	74.31	71.45	81.99	27.64	63.96	36.76	57.24	10.49	50.76	22.05	52.43	0.44	6.84	33.14	55.36

and enable them to predict the full BEV map with sequential images as well. To maintain the original network structure, we keep the original network unchanged to extract the spatial feature while we duplicate the encoders of those networks to fuse the sequential feature. Then, we feed the spatial and temporal features together into the decoder for the semantic BEV map prediction.In addition, we also compare our Seq-BEV with some state-of-the-art BEV detection networks, such as BEVFormer [98], BEVdepth [108] and MatrixVT [144], by changing the detection head into the segmentation one. Keeping the original input settings as the same, the temporal sequential images from 6 vehicle-surrounding cameras are fed into those networks. However, the multi-view inputs slow down the training process. To trade off the computing resources and the training efficiency, we only use the BEVFormer-tiny for the comparison.

The results are shown in Tab. 4.7. Testing with our own full BEV semantic map, the proposed Seq-BEV achieves 38.86% in mIoU and 57.69% in mAP, outperforming all the other methods for most categories. We find that our method is more effective in segmenting small objects, especially for the pedestrian category. It can also be seen that the performance of the original networks is better than the modified networks that take as input sequential images. The reason behind this result may be that the fusion of sequential information requires a special design to get better performance.

4.4.4.2 The Qualitative Demonstrations

Fig. 4.12 shows sample qualitative demonstrations. Due to space limitation, we only displayed the semantic BEV maps generated by the networks that achieved better quantitative results. We can see that our Seq-BEV produces a more accu-

Table 4.7: The comparative results (%) compared with the state-of-the-art methods. We conducted two groups of tests. One is the original network, which takes as input a single image. The other is the modified network, which takes as input sequential images. We use SGL and SEQ to distinguish these two groups. Some BEV-based detection methods are also compared by adding the segmentation head.

Methods	Background		Drivable Area		Ped. Crossing		WalkWay		Obstacle		Vehicle		Pedestrain		mIoU	mAP
	IoU	AP	IoU	AP	IoU	AP	IoU	AP	IoU	AP	IoU	AP	IoU	AP		
SGL-CVT	**67.27**	77.85	72.98	85.15	28.80	53.25	**46.69**	64.43	7.65	35.59	26.11	42.59	0	0	33.91	49.05
SGL-VED	65.91	76.56	74.08	83.98	33.02	57.42	44.30	**65.03**	12.61	49.81	16.82	42.41	0	0	35.25	53.60
SGL-MonoLayout	53.84	64.43	62.58	78.15	2.82	16.97	26.99	43.88	0.82	1.58	8.60	23.56	0	0	22.13	32.65
SGL-VPN	63.35	71.74	71.97	83.75	22.93	55.24	38.19	64.56	14.12	41.40	21.06	44.91	0	0	33.09	51.65
SEQ-CVT	60.84	71.96	68.07	80.62	13.89	29.99	36.14	55.15	1.55	40.10	12.06	35.60	0	0	27.51	44.77
SEQ-VED	61.73	73.68	69.03	79.93	16.19	36.99	38.31	58.63	10.18	36.64	11.40	35.77	0.08	2.17	29.56	46.26
SEQ-MonoLayout	53.40	63.06	62.23	78.37	2.85	19.44	25.79	43.46	0.04	4.10	8.40	24.49	0	0	21.82	33.28
SEQ-VPN	58.70	70.70	67.32	75.65	2.46	55.69	33.42	64.05	1.80	44.60	9.10	48.14	0	0	24.69	51.26
BEVFormer	50.73	66.72	63.45	74.05	17.11	43.39	26.5	45.23	5.5	21.7	14.17	37.56	0.02	0.2	25.35	41.26
BEVDepth	63.85	**77.87**	**75.53**	81.48	**36.71**	63.68	23.9	47.38	0	0	0.41	16.74	0	0	28.63	47.86
MatrixVT	54.07	61.37	63.71	80.20	24.67	57.41	27.67	48.33	6.76	33.12	6.13	33.67	0	0	26.14	44.87
SEQ-BEV (ours)	66.59	76.85	74.89	**85.19**	35.76	**65.27**	43.73	64.72	**22.27**	**52.43**	**27.63**	**54.32**	**1.12**	**7.24**	**38.86**	**57.96**

rate semantic full-BEV map. From the first two rows, compared with the other methods, our Seq-BEV is more sensitive to small objects, such as obstacles or pedestrians on the road. However, the predicted position of the small objects is not perfect because these categories account for a small proportion of the dataset. Moreover, the size of the vehicle predicted by Seq-BEV is closer to the real one. We credit this to the multi-scale dilated convolutions, which are used to process the high-level feature and make it invariant in scale. The last row demonstrates the semantic BEV prediction at nighttime, which indicates that Seq-BEV can still generate a clear and precise result under dark illumination conditions.

4.5 Summary of This Chapter

Semantic BEV map is an environmental representation method for autonomous driving environment information. However, existing methods use single-frame front-view images as network inputs, and due to the limited FoV of the on-board camera, these methods can only predict V-shaped semantic BEV maps, resulting in incomplete observation of the driving environment. To address this issue, this chapter proposes the Seq-BEV network, a full-view semantic BEV map prediction network based on equidistant sequence fusion.

To complete the content outside the camera FoV using multiple images, a self-adapted sequence fusion module was designed in the proposed Seq-BEV network. This module allows the network to adjust the degree of fusion between different images based on the number of training iterations. Additionally, to achieve the transformation from a front view to a bird's-eye view, a road-aware view transformation module has been introduced. This module utilizes an attention mechanism

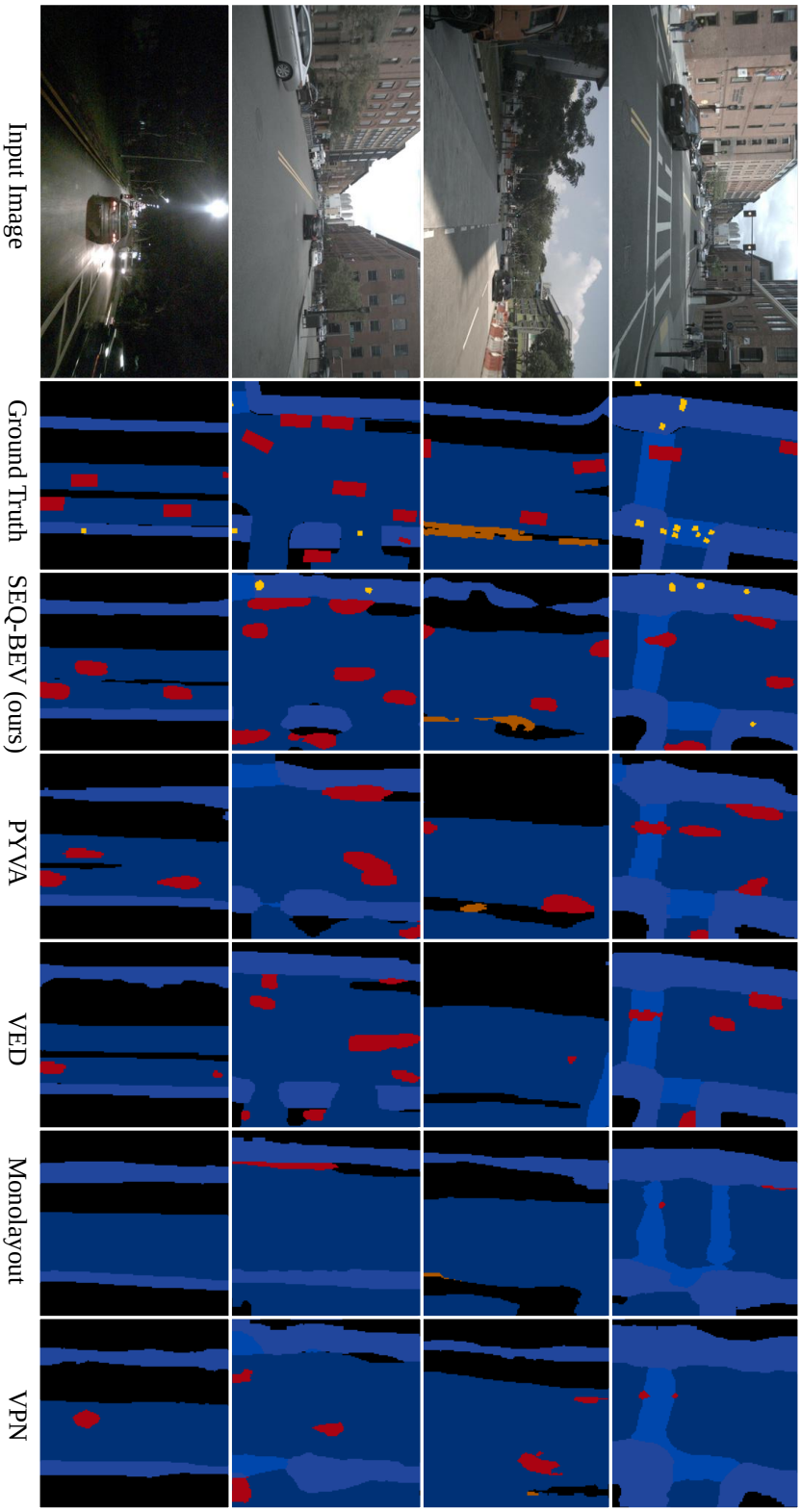


Figure 4.12: Sample qualitative results for the full BEV semantic map prediction networks

to extract road layout features in the driving scene and then inputs these features into a learning-based spatial transformer network, projecting them onto a bird's-eye view. This approach overcomes the limitations of traditional inverse perspective transformation, which assumes a flat plane, and provides interpretability for the neural network during the view transformation process. By integrating the feature encoder, self-adapted sequence fusion module, and road-aware view transformation module, the Seq-BEV network is able to generate full-view semantic BEV maps.

In addition, this chapter presents a dataset created for generating semantic BEV maps, based on the existing autonomous driving dataset nuScenes, providing full-view semantic BEV labels and road semantic labels. The performance of the proposed Seq-BEV network was validated on this dataset using mIoU and mAP as evaluation metrics. Multiple ablation experiments were designed to verify the effectiveness of the proposed network structure. Comparative experiments were also conducted with the most advanced BEV perception methods, including semantic BEV map prediction networks and BEV object detection networks. For the object detection networks, we replaced the detection head with a segmentation head to compare them with our proposed method under the same experimental conditions. Qualitative and quantitative results from various experiments are provided in the experimental section, demonstrating the superiority of the Seq-BEV network through comparative analysis.

Chapter 5

Future Semantic BEV Map Forecasting

5.1 Motivation

Semantic forecasting aims to segment future frames pixel-wisely from previous observations. It is important for semantic environment understanding, which is a fundamental capability of autonomous vehicles [2, 145]. Semantic forecasting could facilitate the intelligent decision-making process [146, 147] by predicting the possible position of the other road agents and the road layout, enabling self-driving cars to avoid obstacles. The semantic bird-eye-view (BEV) map is an ideal format for such task because the BEV map is more flexible in representing the dynamically changing environment. The relative distance between the self-driving car and other agents can be explicitly illustrated. Compared with the front-view images, the BEV map could eliminate the foreshortening due to the perspective projection. Besides the advantages of representation, the BEV map provides a

uniform coordinate to fuse the observation information from different modality inputs. This is in line with the development of autonomous driving, where an increasing number and variety of sensors are equipped for self-driving cars.

The conventional semantic segmentation tasks predict the semantic class for each pixel according to the observation. The semantic segmentation pays attention to the task under the front view. In contrast, the key point of the semantic BEV map prediction is to predict the cross-view semantic position for the objects observed by the front-faced cameras. Recent works [83, 86, 89, 132] have achieved satisfactory results with deep neural networks. However, semantic forecasting is required to predict the semantic distribution for the unobserved frame according to the previous frames. Most existing works focus on front-view semantic forecasting for the future frames or semantic BEV map prediction for the current frame separately. Few attempts solve those two problems within a whole framework. In this work, we aim to forecast the short-term future semantic segmentation in the form of the BEV map. The most related work is proposed by Hoyer *et al.* [148] in 2019. However, they conduct the semantic forecasting in two steps. They generate the semantic segmentation using the off-the-shelf method, DeepLabV3 [43], then transform the semantic information into the bird-eye view in the second step. Such two-step manipulation suffers error accumulation, resulting in inferior performance.

Different from the previous method, we propose an end-to-end semantic forecasting network to predict the semantic BEV map for future frames. We extract the front-view feature from the previously observed images and then predicting the depth distribution with LSS [102]. We propose a dual-forecasting module for semantic forecasting, in which the context and depth of the unobserved frames can

be predicted together. To the best of our knowledge, this is the first network that forecasts the semantic BEV map in an end-to-end manner. The contributions of this work are summarized as follows:

1. We propose an end-to-end framework to forecast semantic BEV map for future frames.

2. We design a depth-context forecasting module to predict and fuse the future depth and context features.

3. We create a group of baseline methods based on the existing semantic BEV map prediction networks and compare the performance of our network with those baselines.

5.2 Background

The previous future forecasting methods can be divided into two categories as illustrated in Fig. 5.1. Early semantics-to-semantics (S2S) methods [149, 150] predict future semantic information with semantic segmentation from the past as the input to the network. Those S2S networks separate the semantic segmentation and forecasting into two tasks rather than predict the future semantics in an end-to-end manner. Recently, feature-to-feature (F2F) forecasting has drawn attention in the forecasting research field. The methods [151, 152] adopt such approach to extract the feature from the origin RBG images and recover the feature maps to the semantic map.

Compared to the S2S method, the F2F strategy directly learns and infers information about future scenes from image features. However, both approaches

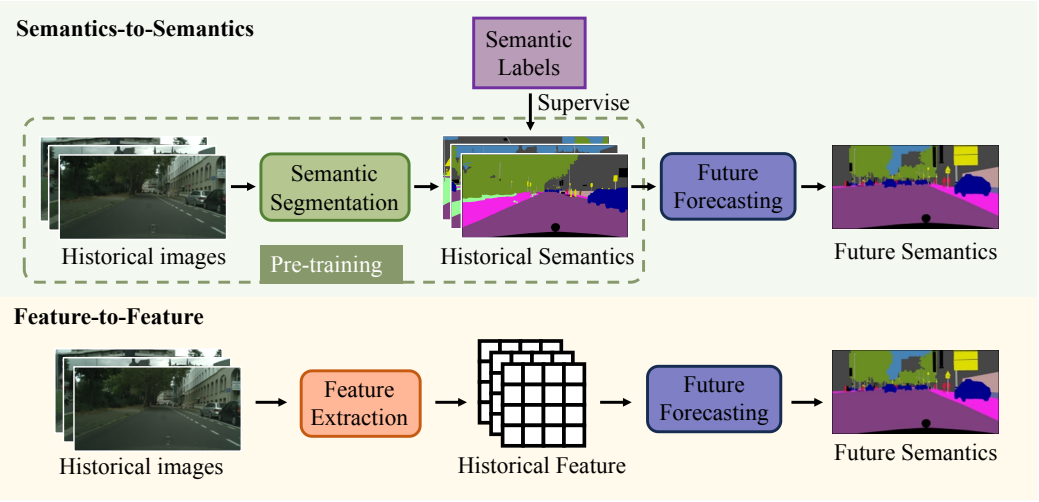


Figure 5.1: The semantics-to-semantics framework VS feature-to-feature framework

focus on future semantic prediction from the same perspective as the historical inputs. In contrast, this chapter aims to represent environmental information at future moments as a semantic BEV map, involving cross-view semantic prediction. Specifically, cross-view semantic prediction necessitates view transformation and the fusion of information from both front and bird's-eye views, which not only increases computational complexity but also places higher demands on the model's generalization capability.

5.3 The Proposed Network

5.3.1 The Overall Architecture

Fig. 5.2 shows the overall architecture of our proposed network. The proposed network mainly consists of a feature extractor, a depth-context forecasting module, and a frustum grid generator. In the first part of our network, the EfficientNet [121]

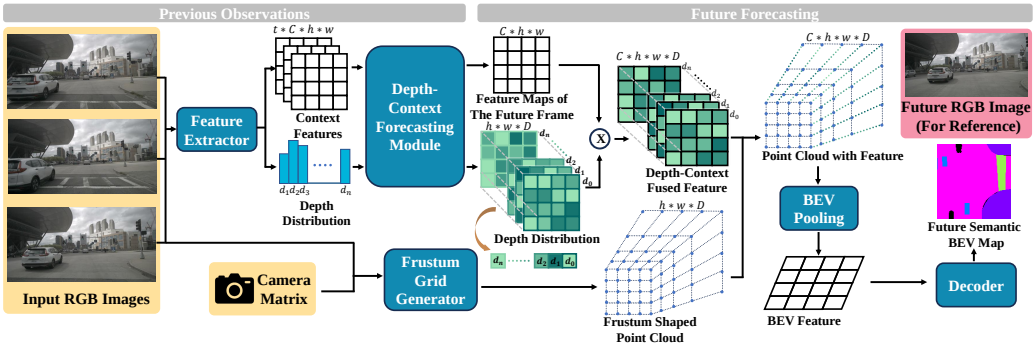


Figure 5.2: The overall architecture of the proposed semantic forecasting network

is adopted as the backbone network. We employ the backbone network to extract the front-view feature from the observed RGB images rather than directly using the semantic maps as the S2S methods. Then, the depth and context feature for the future frame is predicted based on the past frames' feature maps within the depth-context forecasting module. At the same time, the RGB images from the past frames and the camera matrix are fed into the frustum grid generator to get the frustum-shaped point cloud. Each point in this point cloud corresponds to a pixel of the given image at various depths. After getting the point cloud, we assign the obtained depth-context feature to each point and project those points into the BEV plane. Through the semantic head at the end of the proposed network, a semantic BEV map for the future frame can be generated.

5.3.2 BEV Feature Map Prediction

In this work, the images from the front-faced camera and the extrinsic and intrinsic matrix are taken as input to the whole network. Let $I_t \in \mathbb{R}^{3\times H\times W}$ denote the input front-view images from the past t frames. Those images are fed into a pre-trained CNN model, EifficientNet, to get the individual feature, $\mathcal{F}_{front}$. The dimension of

$\mathcal{F}_{front}$ is $B \times t \times C \times h \times w$, where B, t, C, h, w stand for the batch size, numbers of the past input, channel size, the height and width of the extracted feature maps. Given those feature maps, our semantic forecasting network can predict the semantic BEV maps for the future frame, F_{t+m}, where m denotes the timestamp of the future.

After getting the past t frames' feature maps, we transfer perspective from the front view to BEV. To this end, the feature maps in BEV space are generated by first lifting the 2D front-view images, I_t, into the 3D point cloud, P_t. Because the input images are from a monocular camera, the depth estimation for a single image seems like an ill-posed problem without any other input. Taking the camera extrinsic matrix, and intrinsic matrix as the input, each pixel in the image can be projected into the world coordinate, but the individual depth is not sure, which is formulated as:

$$z_c \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = K \cdot \begin{bmatrix} R & T \\ 0_{1\times 3} & 1 \end{bmatrix} \begin{pmatrix} x_w \\ y_w \\ z_w \\ 1 \end{pmatrix} \tag{5.1}$$

where $(u, v, 1)$ is the coordinate of an image pixel p, represented in the form of homogeneous coordinates. K denoted the camera intrinsic matrix. R and T are the rotation and translation matrix, describing the camera's motion pose. $(x_w, y_w, z_w, 1)$ is the world coordinate of a point P_w, corresponding to pixel p. z_c is the distance between the real-world point P_w and the camera, namely the depth of the pixel p. Note that the depth z_c is uncertain for a monocular image.

To get the corresponding depths of the pixels in the given monocular image, we base on [102], assigning n possible depths to each pixel in the I_t. The possible

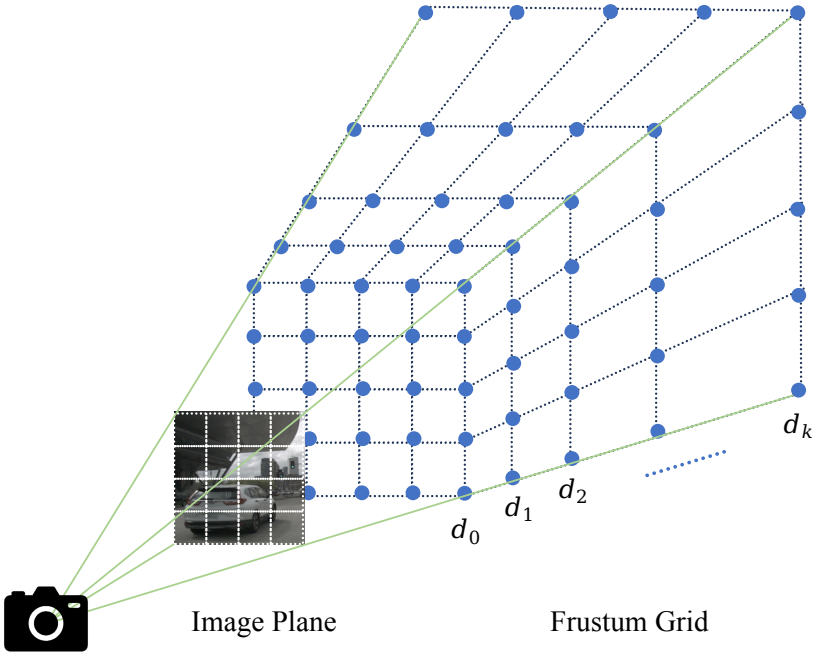


Figure 5.3: The illustration of frustum-shaped point cloud

depth $D = \{d_0, d_1, ..., d_n\}$ is a set of equidistant discrete values. Thus, the 2D image can be projected into a frustum-shaped point cloud with depth D, as illustrated in Fig. 5.3. The depth distribution probability α is predicted, which can be considered as the confidence score at the different depths.

5.3.3 Future Semantic Forecasting

After obtaining the contextual features and depth distribution probabilities of historical images, we designed a depth-context forecasting module to predict future scenes. Based on the concept of convLSTM, a depth-context forecasting module is designed as shown in Figure 5-4. This module combines convolutional neural networks and long short-term memory networks, which can capture spatial features of images while preserving temporal features in sequence data. We use this

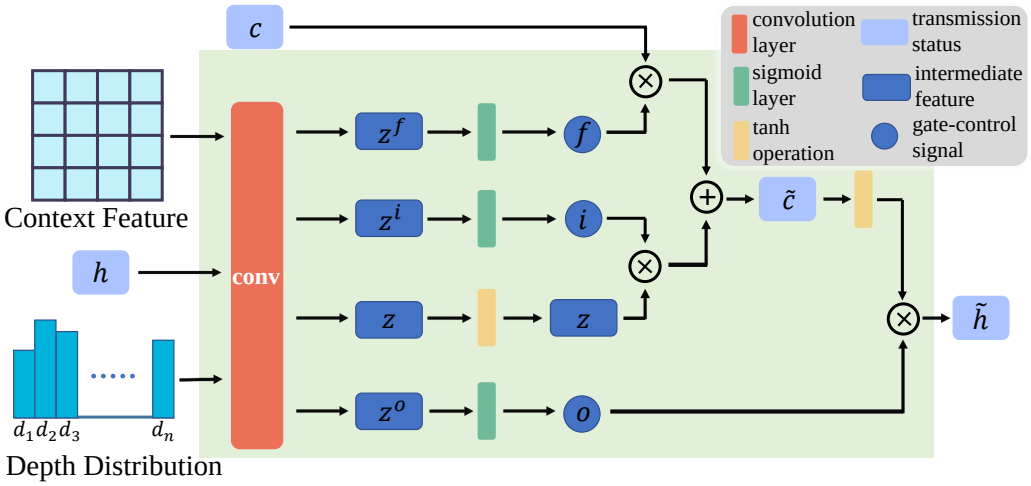


Figure 5.4: The structure of the proposed depth-context forecasting module

module to process historical image sequences to predict contextual features and depth of future scenes.

To predict the next state, we perform convolution on the contextual features and depth distribution separately, and then apply the Sigmoid activation function to obtain a set of gate signals with values in the [0,1] interval:

$$g = \text{Sigmoid}\Big\{\text{Conv}\big[\boldsymbol{W}, \text{Cat}(x^i, h^{i-1})\big]\Big\}, \tag{5.2}$$

where, g uniformly represents the different gate control signals calculated, including forget f, information i, and output o. These gating signals control the information transmitted to the next block. $\boldsymbol{W}$ is the weight matrix of different gating signals, and x^i is the i-th input of the module. H^{i-1} is the state output from the previous prediction block, which we initialize to 0 at the beginning of training.

After performing state prediction on the context feature sequence and depth distribution sequence of historical images, the spatial feature F and depth distri-

bution probability α about the future scene are obtained. As mentioned above, the probability of depth distribution can be seen as the confidence score of pixels appearing in different spatial positions. Therefore, after multiplying the spatial features with the spatial position confidence, we can obtain the feature distribution F_{3d} in three-dimensional space:

$$F_{3d} = \alpha \otimes F, \tag{5.3}$$

we assign the F_{3d} to a frustum spatial grid according to its spatial position, and use the sum pooling to produce the BEV feature maps by projecting.

5.3.4 The Semantic BEV Head

After getting the BEV feature map, a semantic BEV head is introduced to generate the semantic BEV map. This module first conducts the feature learning from the BEV space by a structure that contains the first three layers of the ResNet18 [28]. The size of the BEV feature map shrinks after those layers, and then the upsampling operation is used to recover the output size of the semantic BEV map.

During training, we combine Focal loss and Dice loss to calculate the difference between the future semantic BEV map predicted by the network and its ground truth labels, and update the network parameters through backpropagation gradient:

$$L = L_{focal} + \gamma L_{dice}, \tag{5.4}$$

where, γ is the loss weight used to balance the impact of two types of losses on network training. Through experiments, we set γ to 1.

5.4 Experimnetal Results and Discussions

5.4.1 The Dataset

In our experiments, we use a public autonomous driving dataset, nuScenes [136], to evaluate the performance of our network for semantic BEV map forecasting. There are 850 annotated scenes in the nuScenes dataset. The annotations include the 3D object bounding boxes, the high-definition (HD) maps, and the camera matrix for every frame. Using those annotations, we create the sequential input images for semantic forecasting and the future semantic BEV map as the ground truth labels. We annotate the 7 semantic classes, including the background, drivable area, pedestrian crossing, walkway, obstacle, vehicle and pedestrian. Note that some ground truth labels may not be properly generated due to the limitation of the flat ground assumption. To train the network, we randomly split the whole dataset into 548 training sets, 150 validation sets, and 148 test sets, excluding the sets that contain incorrect semantic BEV labels. For the input sequence, we choose the 3 consecutive frames as the input and take the $4th$ or $6th$ frame for future forecasting. The size of the input images is 256×512, and the output future semantic BEV map contains 150×150 grids, whose resolution is 0.2 m.

5.4.2 Training Details

Our proposed network is implemented on an NVIDIA GeForce RTX 3090 (24 GB RAN) graphics card. Taking the computation cost and the time consumption into consideration, we set the batch size to 16. We train our network for 30 epochs with the Adam optimizer. The initial learning rate is 1×10^{-4} and the weight decay rate

is 1×10^{-5}.

5.4.3 Ablation Study

We conduct ablation studies to verify the effectiveness of the proposed network. In our experiments, the mean Intersection over Union (mIoU) and the mean Average Precision (mAP) are used as the evaluation metrics to assess the network performance.

5.4.3.1 Ablation on the Backbone Network

Since we chose the F2F strategy to forecast the future semantics, it is important to select a powerful backbone network to extract the front-view features from the previously observed images. EfficientNet [121] is known for its accuracy and efficiency. The EfficientNet includes 8 variants, whose structures mainly differ in depth, channel and width. The names of the different variants range from Efficient-B0 to Efficient-B7. This ablation study compares the performance of the network equipped with different EfficientNet variants.

We report the ablation study results in Tab. 5.1. The correct predictions of the road layout and the objects on the road are both critical for autonomous driving. For the convenience of comparison, we divide the 7 semantic classes into the static and dynamic categories. The former includes the background, drivable area, pedestrian crossing, and walkway; the latter includes the obstacle, vehicle and pedestrian. The table shows an obvious rising trend in mIoU and mAP when the backbone changes from a simple structure to a complex one. Therefore, we chose EfficientNet-B7 as our backbone for the proposed network.

Table 5.1: The ablation study results (%) of the variants of the EfficientNet Family. Eff is the short for the EfficientNet. The seven semantic classes are divided into static and dynamic categories, and the mIoU and mAP for those two categories, as well as the mean results across the seven classes, are reported respectively. The best results are highlighted in bold font.

Variants	Statics		Dynamics		mIoU	mAP
	mIoU	mAP	mIoU	mAP		
Eff-B0	36.55	62.17	8.21	24.76	27.85	46.14
Eff-B1	42.77	61.62	7.71	28.14	27.74	47.27
Eff-B2	42.61	60.01	8.26	26.41	27.89	45.61
Eff-B3	42.68	60.70	8.48	25.71	28.02	45.71
Eff-B4	42.94	62.51	8.26	25.46	28.08	46.63
Eff-B5	**43.48**	**63.70**	7.36	25.98	28.00	47.54
Eff-B6	43.03	62.01	8.30	23.59	28.14	45.54
Eff-B7	43.22	63.22	**8.61**	**28.45**	**28.39**	**48.32**

5.4.3.2 Ablation on the Semantic Forecasting

In this section, we compare the forecasting performance of the network with different input and output conditions. This ablation study is separated into two groups, which predict the semantic BEV map for the $1st$ and $3rd$ future frame, respectively. Furthermore, we also set different numbers of the previously observed frames as input for each group. Specifically, the 1, 3, and 5 past front-view images are fed into the network to forecast the future one or three frames.

Tab. 5.2 displays the results of this ablation study. We find that the network forecasting the future one frame performed best when taking as input 3 past observations, while worse having 5 inputs. The situation is changed for forecasting the $3rd$ future frame. The table shows that the best performance can be achieved by taking 5 past frames. We conjecture the reason for this change in the results is the

Table 5.2: The ablation study results (%) of the semantic forecasting. The experiment is separated into two groups, forecasting the $1st$ and $3rd$ future frame, respectively. To further verify the semantic forecasting ability, we set three different inputs for each group. I_n stands for the number of previously observed frames, and O_f indicates which frame is predicted in the future. The best results are highlighted in bold font for forecasting $1st$ and $3rd$ future frame, respectively.

I_n	O_f	Statics		Dynamics		mIoU	mAP
		mIoU	mAP	mIoU	mAP		
1	$1st$	41.09	**63.96**	6.71	26.48	26.36	47.90
3	$1st$	**43.22**	63.22	**8.61**	**28.45**	**28.39**	**48.32**
5	$1st$	41.61	61.44	7.37	26.03	26.93	46.27
1	$3rd$	36.74	56.64	5.07	21.89	23.10	41.75
3	$3rd$	38.24	56.22	6.03	24.36	24.43	42.57
5	$3rd$	**38.84**	**58.00**	**7.22**	**26.94**	**25.29**	**44.69**

information redundancy due to the increasing numbers of input. Using too many past frames to forecast the near future frame is redundant, whereas longer-term future prediction requires more past information to perform better.

5.4.4 Comparative Study

As the proposed network is the first method to forecast the semantic BEV map in the F2F manner, we create several baseline methods to perform the comparative experiments. The networks we chose for baseline comparison are specific to the semantic BEV map prediction task. The networks include VPN [86], VED [83], PYVA [89]. All those networks can only predict the current semantics without the forecasting ability. So, to test the semantic forecasting performance of the proposed module, we integrate the feature forecasting module into those networks. The feature forecasting module is inserted behind the feature extractors to keep

the original network structures unchanged. In this experiment, we also conduct the test with different input and output conditions to compare the precision of semantic forecasting. Meanwhile, we use the original network structures to predict the semantic BEV map for the next future frame with the 1 frame input as a baseline.

5.4.4.1 The Quantitative Results

The comparative results are shown in Tab. 5.3. Taking the three previously observed frames, the proposed network achieves the best forecasting performance, with 28.39% in mIoU and 48.32% in mAP. From the table, we can see that all the networks inserted with the forecasting module get better forecasting results compared with the origin structure (marked with $1/1$ for the I_n/O_n term). This verifies the effectiveness of our semantic forecasting module. In addition, the table shows that our network performs best in predicting small dynamic objects, such as obstacles, vehicles, and pedestrians, illustrating the superiority of our network.

5.4.4.2 The Qualitative Demonstrations

Some sample qualitative results are shown in Fig. 5.5. The networks take three previous frames as input and forecast the next future semantic BEV map. In general, our network achieves the best performance for the semantic forecasting task. Compared with the other networks, Our network is sensitive to small objects like obstacles and pedestrians (labeled in brown and yellow, respectively).

Table 5.3: The comparative results (%) with the baseline methods. We conduct different groups of experiments to test the performance of the selected network with the proposed semantic forecasting module. Each network takes as input 1 or 3 past frames and forecasts the next frame or the $3rd$ frame in the future. I_n and O_n represent the numbers of the input images and which frame is predicted in the future, respectively. We bold the best results according to the different input-output conditions for each method.

Network	I_n / O_n	Background		Drivable Area		Ped. Crossing		WalkWay		Obstacle		Vehicle		Pedestrian		mIoU	mAP
		IoU	AP	IoU	AP	IoU	AP	IoU	AP	IoU	AP	IoU	AP	IoU	AP		
VPN	1/1	53.05	61.17	61.12	**80.92**	11.24	**71.84**	28.93	39.91	0.54	**34.49**	2.59	51.07	0.0	0.0	22.50	**48.48**
	3/1	**56.21**	**71.06**	**67.65**	73.87	**22.58**	67.24	**29.31**	**63.10**	0.0	0.0	7.66	**38.84**	0.0	0.0	**26.20**	44.87
	3/3	52.38	62.21	62.27	76.69	19.35	56.57	27.18	47.26	**2.75**	32.10	**8.22**	31.41	0.0	0.0	24.59	43.75
VED	1/1	56.32	**71.99**	67.36	74.40	27.31	59.74	33.57	60.22	0.0	0.0	3.60	53.80	0.0	0.0	26.88	45.74
	3/1	**61.06**	69.56	**70.88**	**78.63**	**34.49**	**63.61**	**36.13**	**63.91**	0.95	0.0	**6.67**	**51.04**	0.0	0.0	**27.63**	46.69
	3/3	54.48	66.75	65.41	75.36	22.66	61.08	31.47	56.15	**2.20**	**35.60**	4.84	42.62	0.0	0.0	25.87	**48.22**
PYVA	1/1	54.33	**70.31**	66.20	75.56	28.27	55.12	31.64	51.18	3.08	36.81	7.60	36.90	0.0	0.0	27.3	42.91
	3/1	**58.53**	68.86	**68.98**	**80.36**	**29.87**	**61.27**	**33.79**	**53.98**	0.0	0.0	6.74	**42.14**	0.0	0.0	**28.27**	**43.80**
	3/3	54.66	67.73	65.43	76.41	25.03	53.06	30.30	50.60	**5.21**	**36.65**	6.92	32.09	0.0	0.0	26.79	45.22
OURS	1/1	50.31	**62.99**	**62.94**	72.71	25.15	58.50	25.97	**61.65**	10.17	41.48	9.87	**37.33**	0.10	0.62	26.36	47.90
	3/1	**51.45**	62.31	62.87	**77.12**	**27.57**	**59.62**	**30.98**	53.84	**13.64**	**53.42**	**11.71**	29.15	0.48	2.78	**28.39**	**48.32**
	3/3	50.27	61.95	61.49	74.90	25.61	54.18	29.07	54.74	11.42	49.89	9.94	24.85	**0.74**	**3.35**	26.93	46.27

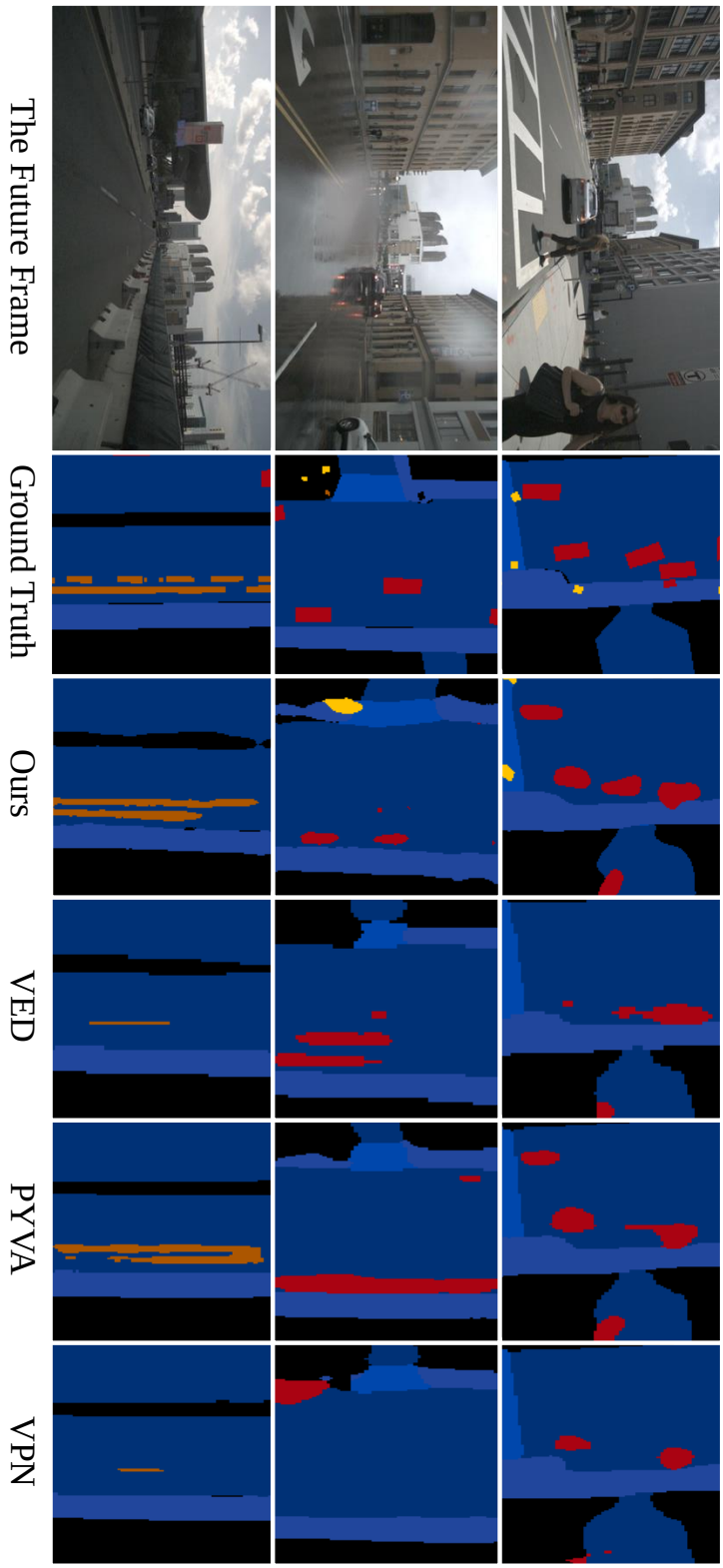


Figure 5.5: Sample qualitative demonstrations for the semantic BEV map forecasting networks

5.5 Summary of This Chapter

The semantic BEV map offers a comprehensive representation of the environmental information surrounding autonomous vehicles. However, existing methods only generate a semantic BEV map for the current frame. This limitation restricts the observation range of the environment in a single frame, thereby limiting the perception system's warning capabilities for autonomous vehicles in special situations. To address this issue, this chapter proposes a network for forecasting future semantic BEV maps.

Multiple frames of historical images are used as input to generate a semantic BEV map for future scenes through the predictive capability of the proposed network. we introduce a depth-context forecasting module that leverages the concept of convolutional long short-term memory to simultaneously learn the spatial and temporal features of sequence images. In our proposed future semantic forecasting network, we first extract context features from historical observations and learn depth distribution probabilities. The proposed depth-context forecasting module is then employed to predict the features and depth distribution of future scenes.

In the experimental phase, we combined multiple historical images into an input sequence using the publicly available autonomous driving dataset nuScenes and created semantic BEV labels for future scenes. We validated the effectiveness of the proposed method on the constructed dataset using mIoU and mAP as evaluation metrics. Through ablation experiments, we determined the optimal parameters and network structure. Additionally, to compare with existing semantic BEV map prediction methods, we established a set of baseline methods to forecast future semantic BEV maps. The experimental results demonstrate that the

proposed method achieves superior performance in future prediction, surpassing existing baseline methods with mIoU and mAP scores of 28.39% and 48.32%, respectively.

Although the proposed method demonstrates excellent capability in generating future semantic BEV maps, this study did not account for the impact of dynamic and static objects in the scene on future semantic predictions. Consequently, future work will involve subdividing the objects in the scene and utilizing the attributes of different objects as prior information for future scene prediction to enhance the accuracy of the predictions.

Chapter 6

Conclusion

Environmental perception is a primary task in autonomous driving systems, providing crucial information about driving scenes. The semantic BEV map has become a mainstream representation in environmental perception tasks due to its inherent advantages. Therefore, research on semantic BEV map prediction is highly significant for the advancement of autonomous driving. This dissertation focuses on an in-depth study of semantic BEV map prediction tasks. Deep learning networks are commonly employed in generating semantic BEV maps, which is a data-driven approach heavily reliant on training datasets. However, due to the transformation between the bird's-eye view and the front view of the original collected images, labeling semantic BEV ground truth presents many challenges. These challenges result in an insufficient number of labeled images and labeling noise in the dataset. To effectively address this issue, this dissertation first investigates a semantic BEV map prediction method under a semi-supervised framework, which alleviates the network training process's dependence on labeled samples while ensuring the quality of the semantic BEV map prediction. The ongoing advancement

of autonomous driving systems necessitates enhanced capabilities for environmental perception systems. Current semantic BEV maps still exhibit limitations in their ability to convey comprehensive environmental information. To address this, this dissertation investigates a method for generating full-view semantic BEV maps using equidistant sequences. By incorporating historical environmental observations, the semantic BEV map extends its scope and predicts future scene changes, thereby improving the perception system's warning capabilities for autonomous vehicles. The primary research contributions of this dissertation are as follows:

(1) A semi-supervised semantic BEV map prediction network (S2G2 network) leveraging contrastive learning is proposed to mitigate the reliance on labeled samples. The S2G2 network incorporates the contrastive learning framework into the task of semantic BEV map prediction, facilitating training with both labeled and unlabeled data. The network comprises a dual-attention view transformation module and a dual-branch generator. The dual-attention view transformation module employs attention mechanisms to extract inter-view and cross-view attention for front-view and BEV features, generating two distinct BEV feature maps from the same front-view input image and treating them as homologous similar features. The dual-branch generator applies contrastive learning by inputting homologous similar features into two identical network branches, achieving semi-supervised learning through penalizing the consistency loss between the dual branch network outputs. Experiments conducted on the publicly available Cityscapes autonomous driving dataset demonstrate that the S2G2 network outperforms existing semantic BEV map prediction networks, achieving superior results in semantic BEV map prediction. The network improved semantic segmentation accuracy by approximately 1%, reaching 58.86% mIoU and 70.23% mAP, and exhibited enhanced

robustness to atypical lighting conditions.

(2) The Seq-BEV network, based on equidistant sequence fusion, is proposed to address the issue of limited environmental observation due to constrained camera field of view. To mitigate the failure of temporal series during driving, this network innovatively utilizes equidistant sequence images to enhance the environmental observation range of a single frame. The Seq-BEV network comprises a two-stream encoder, a self-adapted sequence fusion module, and a road-aware view transformation module. The two-stream encoder extracts sequence features and spatial features separately, preserving the integrity of spatial features during sequence fusion. The self-adapted sequence fusion module adjusts the degree of fusion of different features from sequence images according to the training progress. The road-aware view transformation module first extracts feature maps containing road attention under auxiliary supervision, subsequently employing a learnable spatial transformation network to project front-view road features onto a bird's-eye view. This module is designed based on principles of visual geometry, which alleviate the limitations of the flat plane assumption in inverse perspective mapping and provide explainability for view transformation. Using the publicly available nuScenes autonomous driving dataset, this dissertation creates full-view semantic BEV labels and semantic road layout labels. Experimental results on this dataset demonstrate that the Seq-BEV network effectively extends the expression range of the semantic BEV map, resulting in a full-view semantic BEV map. Compared to existing semantic BEV map prediction methods, Seq-BEV improved semantic segmentation accuracy from 35.25% to 38.86% as measured by mIoU.

(3) A future semantic BEV map forecasting network utilizing ConvLSTM is proposed to equip autonomous driving systems with short-term future scene per-

ception. The network employs the F2F strategy to forecast future scene information, first extracting front-view context features and the depth distribution from the historical observations. Additionally, a depth-context forecasting module is designed, which uses the features and depths from historical images as input sequences to predict the features and depth distribution of future scenes. The combination of depth distribution vectors and scene feature maps yields the feature distribution in three-dimensional space. Finally, through BEV pooling, it is projected onto the BEV plane and decoded by a semantic encoder to obtain a semantic BEV map for the future scene. Experimental evaluations on a publicly available autonomous driving dataset demonstrate that when three frames of historical images are input into the proposed network, it effectively generates a semantic BEV map for the subsequent one or three frames.

The key innovations of this dissertation are summarized as follows:

(1) A semi-supervised semantic BEV map prediction network leveraging contrastive learning has been proposed to reduce the reliance on labeled data during training without the need for complex data augmentation techniques. A dual-attention view transformation module was developed to convert front-view features into a bird's-eye view while generating a set of homologous similar features as inputs for a contrastive learning-based dual-branch semantic generator.

(2) A full-view semantic BEV map prediction network based on equidistant sequence fusion has been proposed. This network employs the training process to control the fusion degree of sequence images and uses an explainable network for view transformation. By innovatively using equidistant sequence images, the network expands the observation range of a single frame image of the environment. Additionally, a self-adapted sequence fusion module and a road-aware view trans-

formation module are introduced.

(3) A future semantic BEV forecasting network based on ConvLSTM has been proposed to accurately predict future scenarios and enhance the warning capabilities of the perception system in autonomous vehicles. By employing a depth-context forecasting module, historical environmental features are utilized as input to infer a semantic BEV map for future moments, thereby enabling vehicles to respond more promptly to forthcoming environmental changes.

In summary, this article has conducted an in-depth study on the prediction method of semantic BEV map in autonomous driving, yielding significant research findings. Nevertheless, there remain several directions for further exploration in future research, as outlined below:

(1) This dissertation addresses the issue of training samples and enhances the information expression capability of semantic BEV maps. These challenges may arise in various real-world scenarios. To adapt flexibly to changing conditions in real-world applications, a unified network framework should be developed to integrate the various capabilities of the network proposed in this dissertation. Depending on the specific task requirements, appropriate forms of semantic BEV maps should be generated.

(2) Dynamic and static objects in a scene possess distinct attributes, necessitating an exploration of the impact of prior information, such as the motion state and behavioral characteristics of different objects, on semantic BEV map prediction. The advent of visual language models offers new perspectives for this research. These models excel in understanding and generating natural language and images, and can enhance semantic BEV map prediction networks by providing a more accurate understanding of driving scene content.

(3) In complex environments such as rain, snow, and low-light conditions, visual sensors often face significant challenges in collecting environmental data. Investigating the use of multimodal fusion for semantic BEV map prediction is crucial. This approach involves integrating data from sources like laser radar point clouds, which provide accurate distance information, with image data that offers texture and color information. By combining these modalities, more reliable driving scene information can be obtained, thereby enhancing the autonomous driving system's ability to comprehend the complex scenes.

(4) In the process of generating semantic BEV maps, height information of the scene is often neglected, with only two-dimensional plane information being retained. However, height information is crucial for distinguishing between ground obstacles and suspended obstacles. Therefore, investigating the impact of height information on semantic BEV map prediction can not only enhance the accuracy of obstacle detection but also significantly improve the environmental perception capabilities of autonomous driving systems.

Chapter 7

References

[1] Görkay Aydemir, Adil Kaan Akan, and Fatma Güney. "Adapt: Efficient multi-agent trajectory prediction with adaptation". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 8295–8305.

[2] Siyu Teng et al. "Motion planning for autonomous driving: The state of the art and future perspectives". In: *IEEE Transactions on Intelligent Vehicles* 8.6 (2023), pp. 3692–3711.

[3] Junru Gu et al. "Vip3d: End-to-end visual trajectory prediction via 3d agent queries". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 5496–5506.

[4] Lukas Neumann and Andrea Vedaldi. "Pedestrian and ego-vehicle trajectory prediction from monocular camera". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 10204–10212.

[5] Ye Yuan et al. “Agentformer: Agent-aware transformers for socio-temporal multi-agent forecasting”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 9813–9823.

[6] Yanjun Huang et al. “A survey on trajectory-prediction methods for autonomous driving”. In: *IEEE Transactions on Intelligent Vehicles* 7.3 (2022), pp. 652–674.

[7] Alexey Dosovitskiy et al. “CARLA: An open urban driving simulator”. In: *Conference on robot learning*. PMLR. 2017, pp. 1–16.

[8] Robert M Haralick and Linda G Shapiro. *Computer and robot vision*. Vol. 1. Addison-wesley Reading, MA, 1992.

[9] Shawn Lankton and Allen Tannenbaum. “Localizing region-based active contours”. In: *IEEE transactions on image processing* 17.11 (2008), pp. 2029–2039.

[10] Daniel Freedman and Tao Zhang. “Interactive graph cut based segmentation with shape priors”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Piscataway, NJ: IEEE, 2005, pp. 755–762.

[11] Leo Grady. “Random walks for image segmentation”. In: *IEEE transactions on pattern analysis and machine intelligence* 28.11 (2006), pp. 1768–1783.

[12] Wenxian Yang et al. “User-friendly interactive image segmentation through unified combinatorial user inputs”. In: *IEEE Transactions on Image Processing* 19.9 (2010), pp. 2470–2479.

[13] Yu-Kun Lai et al. “Fast mesh segmentation using random walks”. In: *Proceedings of the 2008 ACM symposium on Solid and physical modeling*. New York, NY: Association for Computing Machinery, 2008, pp. 183–191.

[14] Juyong Zhang et al. “Mesh snapping: Robust interactive mesh cutting using fast geodesic curvature flow”. In: *Computer graphics forum*. Vol. 29. 2. Wiley Online Library. Oxford, UK: Blackwell Publishing Ltd, 2010, pp. 517–526.

[15] Dorin Comaniciu and Peter Meer. “Mean shift: A robust approach toward feature space analysis”. In: *IEEE Transactions on pattern analysis and machine intelligence* 24.5 (2002), pp. 603–619.

[16] Keh-Shih Chuang et al. “Fuzzy c-means clustering with spatial information for image segmentation”. In: *computerized medical imaging and graphics* 30.1 (2006), pp. 9–15.

[17] Radhakrishna Achanta et al. “SLIC superpixels compared to state-of-the-art superpixel methods”. In: *IEEE transactions on pattern analysis and machine intelligence* 34.11 (2012), pp. 2274–2282.

[18] Zhengqin Li and Jiansheng Chen. “Superpixel segmentation using linear spectral clustering”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Piscataway, NJ: IEEE, 2015, pp. 1356–1363.

[19] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. “Deep learning”. In: *nature* 521.7553 (2015), pp. 436–444.

[20] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. “Imagenet classification with deep convolutional neural networks”. In: *Advances in neural information processing systems* 25 (2012).

[21] Christian Szegedy et al. “Going deeper with convolutions”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. Piscataway, NJ: IEEE, 2015, pp. 1–9.

[22] Jonathan Long, Evan Shelhamer, and Trevor Darrell. “Fully convolutional networks for semantic segmentation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. Piscataway, NJ: IEEE, 2015, pp. 3431–3440.

[23] Karen Simonyan and Andrew Zisserman. “Very deep convolutional networks for large-scale image recognition”. In: *arXiv preprint arXiv:1409.1556* (2014).

[24] Camille Couprie et al. “Indoor semantic segmentation using depth information”. In: *arXiv preprint arXiv:1301.3572* (2013).

[25] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. “Learning deconvolution network for semantic segmentation”. In: *Proceedings of the IEEE international conference on computer vision*. Piscataway, NJ: IEEE, 2015, pp. 1520–1528.

[26] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. “Segnet: A deep convolutional encoder-decoder architecture for image segmentation”. In: *IEEE transactions on pattern analysis and machine intelligence* 39.12 (2017), pp. 2481–2495.

[27] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. “U-net: Convolutional networks for biomedical image segmentation”. In: *International Conference on Medical image computing and computer-assisted intervention*. Springer. 2015, pp. 234–241.

[28] Kaiming He et al. “Deep residual learning for image recognition”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[29] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. “V-net: Fully convolutional neural networks for volumetric medical image segmentation”. In: *2016 fourth international conference on 3D vision (3DV)*. Ieee. Piscataway, NJ: IEEE, 2016, pp. 565–571.

[30] Simon Jégou et al. “The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*. Piscataway, NJ: IEEE, 2017, pp. 11–19.

[31] Gao Huang et al. “Densely connected convolutional networks”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. Piscataway, NJ: IEEE, 2017, pp. 4700–4708.

[32] Md Amirul Islam et al. “Gated feedback refinement network for dense image labeling”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. Piscataway, NJ: IEEE, 2017, pp. 3751–3759.

[33] Zongwei Zhou et al. “Unet++: Redesigning skip connections to exploit multiscale features in image segmentation”. In: *IEEE transactions on medical imaging* 39.6 (2019), pp. 1856–1867.

[34] Huimin Huang et al. “Unet 3+: A full-scale connected unet for medical image segmentation”. In: *ICASSP 2020-2020 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE. Piscataway, NJ: IEEE, 2020, pp. 1055–1059.

[35] Jun Fu et al. “Stacked deconvolutional network for semantic segmentation”. In: *IEEE Transactions on Image Processing* (2019).

[36] Chao Peng et al. “Large kernel matters–improve semantic segmentation by global convolutional network”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. Piscataway, NJ: IEEE, 2017, pp. 4353–4361.

[37] Tobias Pohlen et al. “Full-resolution residual networks for semantic segmentation in street scenes”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. Piscataway, NJ: IEEE, 2017, pp. 4151–4160.

[38] Zbigniew Wojna et al. “The devil is in the decoder: Classification, regression and gans”. In: *International Journal of Computer Vision* 127 (2019), pp. 1694–1706.

[39] Zhenli Zhang et al. “Exfuse: Enhancing feature fusion for semantic segmentation”. In: *Proceedings of the European conference on computer vision (ECCV)*. Berlin□German: Springer, 2018, pp. 269–284.

[40] Wei Liu, Andrew Rabinovich, and Alexander C Berg. “Parsenet: Looking wider to see better”. In: *arXiv preprint arXiv:1506.04579* (2015).

[41] Hengshuang Zhao et al. “Pyramid scene parsing network”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. Piscataway, NJ: IEEE, 2017, pp. 2881–2890.

[42] Liang-Chieh Chen et al. “Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs”. In: *IEEE transactions on pattern analysis and machine intelligence* 40.4 (2017), pp. 834–848.

[43] Liang-Chieh Chen et al. “Rethinking atrous convolution for semantic image segmentation”. In: *arXiv preprint arXiv:1706.05587* (2017).

[44] Liang-Chieh Chen et al. “Encoder-decoder with atrous separable convolution for semantic image segmentation”. In: *Proceedings of the European conference on computer vision (ECCV)*. 2018, pp. 801–818.

[45] Liang-Chieh Chen et al. “Semantic image segmentation with deep convolutional nets and fully connected crfs”. In: *arXiv preprint arXiv:1412.7062* (2014).

[46] Guosheng Lin et al. “Refinenet: Multi-path refinement networks for high-resolution semantic segmentation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. Piscataway, NJ: IEEE, 2017, pp. 1925–1934.

[47] Alex Graves. “Generating sequences with recurrent neural networks”. In: *arXiv preprint arXiv:1308.0850* (2013).

[48] Nal Kalchbrenner, Ivo Danihelka, and Alex Graves. “Grid long short-term memory”. In: *arXiv preprint arXiv:1507.01526* (2015).

[49] Ashish Vaswani et al. “Attention is all you need”. In: *Advances in neural information processing systems* 30 (2017).

[50] Jacob Devlin et al. “Bert: Pre-training of deep bidirectional transformers for language understanding”. In: *arXiv preprint arXiv:1810.04805* (2018).

[51] Tom Brown et al. “Language models are few-shot learners”. In: *Advances in neural information processing systems* 33 (2020), pp. 1877–1901.

[52] Mark Chen et al. “Generative pretraining from pixels”. In: *International conference on machine learning*. PMLR. New York, NY: PMLR, 2020, pp. 1691–1703.

[53] Xizhou Zhu et al. “Deformable detr: Deformable transformers for end-to-end object detection”. In: *arXiv preprint arXiv:2010.04159* (2020).

[54] Ian Goodfellow et al. “Generative adversarial networks”. In: *Communications of the ACM* 63.11 (2020), pp. 139–144.

[55] Diederik P Kingma and Max Welling. “Auto-encoding variational bayes”. In: *arXiv preprint arXiv:1312.6114* (2013).

[56] Yuri Burda, Roger Grosse, and Ruslan Salakhutdinov. “Importance weighted autoencoders”. In: *arXiv preprint arXiv:1509.00519* (2015).

[57] Robin Rombach et al. “High-resolution image synthesis with latent diffusion models”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. Piscataway, NJ: IEEE, 2022, pp. 10684–10695.

[58] Francesco Visin et al. “Reseg: A recurrent neural network-based model for semantic segmentation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*. Piscataway, NJ: IEEE, 2016, pp. 41–48.

[59] Wenjia Bai et al. “Recurrent neural networks for aortic image sequence segmentation with sparse annotations”. In: *Medical Image Computing and Computer Assisted Intervention–MICCAI 2018: 21st International Conference, Granada, Spain, September 16-20, 2018, Proceedings, Part IV 11*. Springer. Berlin German: Springer, 2018, pp. 586–594.

[60] Xingjian Shi et al. “Convolutional LSTM network: A machine learning approach for precipitation nowcasting”. In: *Advances in neural information processing systems* 28 (2015).

[61] Yong Yu et al. “A review of recurrent neural networks: LSTM cells and network architectures”. In: *Neural computation* 31.7 (2019), pp. 1235–1270.

[62] Md Zahangir Alom et al. “Recurrent residual U-Net for medical image segmentation”. In: *Journal of Medical Imaging* 6.1 (2019), pp. 014006–014006.

[63] Arunava Chakravarty and Jayanthi Sivaswamy. “RACE-net: a recurrent neural network for biomedical image segmentation”. In: *IEEE journal of biomedical and health informatics* 23.3 (2018), pp. 1151–1162.

[64] Kelvin KL Wong et al. “Brain image segmentation of the corpus callosum by combining Bi-Directional Convolutional LSTM and U-Net using multi-

slice CT and MRI". In: *Computer Methods and Programs in Biomedicine* 238 (2023), p. 107602.

[65] Kai Han et al. "A survey on vision transformer". In: *IEEE transactions on pattern analysis and machine intelligence* 45.1 (2022), pp. 87–110.

[66] Sixiao Zheng et al. "Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. Piscataway, NJ: IEEE, 2021, pp. 6881–6890.

[67] Alexey Dosovitskiy et al. "An image is worth 16x16 words: Transformers for image recognition at scale". In: *arXiv preprint arXiv:2010.11929* (2020).

[68] Wenhai Wang et al. "Pyramid vision transformer: A versatile backbone for dense prediction without convolutions". In: *Proceedings of the IEEE/CVF international conference on computer vision*. Piscataway, NJ: IEEE, 2021, pp. 568–578.

[69] Jieneng Chen et al. "Transunet: Transformers make strong encoders for medical image segmentation". In: *arXiv preprint arXiv:2102.04306* (2021).

[70] Enze Xie et al. "SegFormer: Simple and efficient design for semantic segmentation with transformers". In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 12077–12090.

[71] Ze Liu et al. "Swin transformer: Hierarchical vision transformer using shifted windows". In: *Proceedings of the IEEE/CVF international conference on computer vision*. Piscataway, NJ: IEEE, 2021, pp. 10012–10022.

[72] Xiangxiang Chu et al. “Twins: Revisiting spatial attention design in vision transformers”. In: *arXiv preprint arXiv:2104.13840* 2.3 (2021).

[73] GM Harshvardhan et al. “A comprehensive survey and analysis of generative models in machine learning”. In: *Computer Science Review* 38 (2020), p. 100285.

[74] Zhuowen Tu. “Probabilistic boosting-tree: Learning discriminative models for classification, recognition, and clustering”. In: *Tenth IEEE International Conference on Computer Vision (ICCV’05) Volume 1*. Vol. 2. IEEE. Piscataway, NJ: IEEE, 2005, pp. 1589–1596.

[75] Pauline Luc et al. “Semantic segmentation using adversarial networks”. In: *arXiv preprint arXiv:1611.08408* (2016).

[76] Mina Rezaei et al. “A conditional adversarial network for semantic segmentation of brain tumor”. In: *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries: Third International Workshop, BrainLes 2017, Held in Conjunction with MICCAI 2017, Quebec City, QC, Canada, September 14, 2017, Revised Selected Papers 3*. Springer. Berlin German: Springer, 2018, pp. 241–252.

[77] Naji Khosravan et al. “Pan: Projective adversarial network for medical image segmentation”. In: *Medical Image Computing and Computer Assisted Intervention–MICCAI 2019: 22nd International Conference, Shenzhen, China, October 13–17, 2019, Proceedings, Part VI 22*. Springer. Berlin German: Springer, 2019, pp. 68–76.

[78] Yuan Xue et al. “Segan: Adversarial network with multi-scale l 1 loss for medical image segmentation”. In: *Neuroinformatics* 16 (2018), pp. 383–392.

[79] Zhiqing Zhang et al. “Introducing Shape Prior Module in Diffusion Model for Medical Image Segmentation”. In: *arXiv preprint arXiv:2309.05929* (2023).

[80] Zheyuan Zhang et al. “EMIT-Diff: Enhancing Medical Image Segmentation via Text-Guided Diffusion Model”. In: *arXiv preprint arXiv:2310.12868* (2023).

[81] Aliasghar Khani et al. “Slime: Segment like me”. In: *arXiv preprint arXiv:2309.03179* (2023).

[82] Yiyang Zhou et al. “Automatic Construction of Lane-level HD Maps for Urban Scenes”. In: *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2021, pp. 6649–6656.

[83] Chenyang Lu, Marinus Jacobus Gerardus van de Molengraft, and Gijs Dubbelman. “Monocular semantic occupancy grid mapping with convolutional variational encoder–decoder networks”. In: *IEEE Robotics and Automation Letters* 4.2 (2019), pp. 445–452.

[84] Lennart Reiher, Bastian Lampe, and Lutz Eckstein. “A sim2real deep learning approach for the transformation of images from multiple vehicle-mounted cameras to a semantically segmented image in bird’s eye view”. In: *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*. IEEE. Piscataway, NJ: IEEE, 2020, pp. 1–7.

[85] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. “Spatial transformer networks”. In: *Advances in neural information processing systems* 28 (2015).

[86] Bowen Pan et al. “Cross-view semantic segmentation for sensing surroundings”. In: *IEEE Robotics and Automation Letters* 5.3 (2020), pp. 4867–4873.

[87] Kaustubh Mani et al. “MonoLayout: Amodal scene layout from a single image”. In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2020, pp. 1689–1697.

[88] Thomas Roddick and Roberto Cipolla. “Predicting semantic map representations from images using pyramid occupancy networks”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, pp. 11138–11147.

[89] Weixiang Yang et al. “Projecting Your View Attentively: Monocular Road Scene Layout Estimation via Cross-View Transformation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 15536–15545.

[90] Nicolas Carion et al. “End-to-end object detection with transformers”. In: *European conference on computer vision*. Springer. Berlin German: Springer, 2020, pp. 213–229.

[91] Kashyap Chitta, Aditya Prakash, and Andreas Geiger. “Neat: Neural attention fields for end-to-end autonomous driving”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. Piscataway, NJ: IEEE, 2021, pp. 15793–15803.

[92] Yigit Baran Can et al. “Structured bird’s-eye-view traffic scene understanding from onboard images”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. Piscataway, NJ: IEEE, 2021, pp. 15661–15670.

[93] Yue Wang et al. “Detr3d: 3d object detection from multi-view images via 3d-to-2d queries”. In: *Conference on Robot Learning*. PMLR. New York, NY: PMLR, 2022, pp. 180–191.

[94] Brady Zhou and Philipp Krähenbühl. “Cross-view transformers for real-time map-view semantic segmentation”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. Piscataway, NJ: IEEE, 2022, pp. 13760–13769.

[95] Yingfei Liu et al. “Petr: Position embedding transformation for multi-view 3d object detection”. In: *European Conference on Computer Vision*. Springer. Berlin German: Springer, 2022, pp. 531–548.

[96] Yingfei Liu et al. “Petrv2: A unified framework for 3d perception from multi-camera images”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. Piscataway, NJ: IEEE, 2023, pp. 3262–3272.

[97] Avishkar Saha et al. “Translating images into maps”. In: *2022 International conference on robotics and automation (ICRA)*. IEEE. Piscataway, NJ: IEEE, 2022, pp. 9200–9206.

[98] Zhiqi Li et al. “Bevformer: Learning bird’s-eye-view representation from multi-camera images via spatiotemporal transformers”. In: *European conference on computer vision*. Springer. 2022, pp. 1–18.

[99] Chenyu Yang et al. “BEVFormer v2: Adapting Modern Image Backbones to Bird’s-Eye-View Recognition via Perspective Supervision”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 17830–17839.

[100] Li Chen et al. “Persformer: 3d lane detection via perspective transformer and the openlane benchmark”. In: *European Conference on Computer Vision*. Springer. Berlin German: Springer, 2022, pp. 550–567.

[101] Lang Peng et al. “Bevsegformer: Bird’s eye view semantic segmentation from arbitrary camera rigs”. In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. Piscataway, NJ: IEEE, 2023, pp. 5935–5943.

[102] Jonah Philion and Sanja Fidler. “Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d”. In: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*. Springer. 2020, pp. 194–210.

[103] Cody Reading et al. “Categorical depth distribution network for monocular 3d object detection”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. Piscataway, NJ: IEEE, 2021, pp. 8555–8564.

[104] Junjie Huang et al. “Bevdet: High-performance multi-camera 3d object detection in bird-eye-view”. In: *arXiv preprint arXiv:2112.11790* (2021).

[105] Enze Xie et al. “M^2 BEV: Multi-Camera Joint 3D Detection and Segmentation with Unified Birds-Eye View Representation”. In: *arXiv preprint arXiv:2204.05088* (2022).

[106] Zhijian Liu et al. “Bevfusion: Multi-task multi-sensor fusion with unified bird’s-eye view representation”. In: *2023 IEEE international conference on robotics and automation (ICRA)*. IEEE. Piscataway, NJ: IEEE, 2023, pp. 2774–2781.

[107] Hongyu Zhou et al. “Matrixvt: Efficient multi-camera to bev transformation for 3d perception”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. Piscataway, NJ: IEEE, 2023, pp. 8548–8557.

[108] Yinhao Li et al. “Bevdepth: Acquisition of reliable depth for multi-view 3d object detection”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 37. 2. 2023, pp. 1477–1485.

[109] Yinhao Li et al. “Bevstereo: Enhancing depth estimation in multi-view 3d object detection with temporal stereo”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 37. 2. Menlo Park, CA: Association for the Advancement of Artificial Intelligence, 2023, pp. 1486–1494.

[110] Peide Cai et al. “DiGNet: Learning Scalable Self-Driving Policies for Generic Traffic Scenarios with Graph Neural Networks”. In: *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2020, pp. 8979–8984.

[111] Hengli Wang et al. “Learning interpretable end-to-end vision-based motion planning for autonomous driving with optical flow distillation”. In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2021, pp. 13731–13737.

[112] Hengli Wang et al. “End-to-End Interactive Prediction and Planning With Optical Flow Distillation for Autonomous Driving”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*. June 2021, pp. 2229–2238.

[113] Özgür Erkent, Christian Wolf, and Christian Laugier. “Semantic grid estimation with occupancy grids and semantic segmentation networks”. In: *International Conference on Control, Automation, Robotics and Vision (ICARCV)*. IEEE. 2018, pp. 1051–1056.

[114] Sven Richter et al. “Semantic Evidential Grid Mapping based on Stereo Vision”. In: *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*. IEEE. 2020, pp. 179–184.

[115] Antti Tarvainen and Harri Valpola. “Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results”. In: *Advances in neural information processing systems* 30 (2017).

[116] Geoff French et al. “Semi-supervised semantic segmentation needs strong, varied perturbations”. In: *arXiv preprint arXiv:1906.01916* (2019).

[117] Yuliang Zou et al. “Pseudoseg: Designing pseudo labels for semantic segmentation”. In: *arXiv preprint arXiv:2010.09713* (2020).

[118] Hongyi Zhang et al. “mixup: Beyond empirical risk minimization”. In: *arXiv preprint arXiv:1710.09412* (2017).

[119] Terrance DeVries and Graham W Taylor. “Improved regularization of convolutional neural networks with cutout”. In: *arXiv preprint arXiv:1708.04552* (2017).

[120] Sangdoo Yun et al. “Cutmix: Regularization strategy to train strong classifiers with localizable features”. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2019, pp. 6023–6032.

[121] Mingxing Tan and Quoc Le. “Efficientnet: Rethinking model scaling for convolutional neural networks”. In: *International Conference on Machine Learning*. PMLR. 2019, pp. 6105–6114.

[122] Jun-Yan Zhu et al. “Unpaired image-to-image translation using cycle-consistent adversarial networks”. In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 2223–2232.

[123] François Chollet. “Xception: Deep learning with depthwise separable convolutions”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 1251–1258.

[124] Hongyi Zhang et al. “mixup: Beyond Empirical Risk Minimization”. In: *International Conference on Learning Representations*. 2018.

[125] Marius Cordts et al. “The Cityscapes Dataset for Semantic Urban Scene Understanding”. In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016.

[126] Heiko Hirschmuller. “Stereo processing by semiglobal matching and mutual information”. In: *IEEE Transactions on pattern analysis and machine intelligence* 30.2 (2007), pp. 328–341.

[127] Yuxiang Sun, Weixun Zuo, and Ming Liu. “Rtfnet: Rgb-thermal fusion network for semantic segmentation of urban scenes”. In: *IEEE Robotics and Automation Letters* 4.3 (2019), pp. 2576–2583.

[128] Ke Sun et al. “High-resolution representations for labeling pixels and regions”. In: *arXiv preprint arXiv:1904.04514* (2019).

[129] Kaouther Messaoud et al. “Trajectory prediction for autonomous driving based on multi-head attention with joint agent-map representation”. In: *2021 IEEE Intelligent Vehicles Symposium (IV)*. IEEE. 2021, pp. 165–170.

[130] Shengchao Hu et al. “St-p3: End-to-end vision-based autonomous driving via spatial-temporal feature learning”. In: *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXVIII*. Springer. 2022, pp. 533–549.

[131] Hao Dong et al. “SuperFusion: Multilevel LiDAR-Camera Fusion for Long-Range HD Map Generation and Prediction”. In: *arXiv preprint arXiv:2211.15656* (2022).

[132] Shuang Gao, Qiang Wang, and Yuxiang Sun. “S2G2: Semi-supervised semantic bird-eye-view grid-map generation using a monocular camera for autonomous driving”. In: *IEEE Robotics and Automation Letters* 7.4 (2022), pp. 11974–11981.

[133] Lang Peng et al. “BEVSegFormer: Bird’s Eye View Semantic Segmentation From Arbitrary Camera Rigs”. In: *arXiv preprint arXiv:2203.04050* (2022).

[134] Yanqin Jiang et al. “Polarformer: Multi-camera 3d object detection with polar transformers”. In: *arXiv preprint arXiv:2206.15398* (2022).

[135] Chenyu Yang et al. “BEVFormer v2: Adapting Modern Image Backbones to Bird’s-Eye-View Recognition via Perspective Supervision”. In: *arXiv preprint arXiv:2211.10439* (2022).

[136] Holger Caesar et al. “nuscenes: A multimodal dataset for autonomous driving”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 11621–11631.

[137] Mark Sandler et al. “Mobilenetv2: Inverted residuals and linear bottlenecks”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 4510–4520.

[138] Ji Lin, Chuang Gan, and Song Han. “Tsm: Temporal shift module for efficient video understanding”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 7083–7093.

[139] Jie Hu, Li Shen, and Gang Sun. “Squeeze-and-excitation networks”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 7132–7141.

[140] Tsung-Yi Lin et al. “Focal loss for dense object detection”. In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 2980–2988.

[141] Ilya Loshchilov and Frank Hutter. “Decoupled weight decay regularization”. In: *arXiv preprint arXiv:1711.05101* (2017).

[142] Ilya Loshchilov and Frank Hutter. “Sgdr: Stochastic gradient descent with warm restarts”. In: *arXiv preprint arXiv:1608.03983* (2016).

[143] Andrew Howard et al. “Searching for mobilenetv3”. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2019, pp. 1314–1324.

[144] Hongyu Zhou et al. “MatrixVT: Efficient Multi-Camera to BEV Transformation for 3D Perception”. In: *arXiv preprint arXiv:2211.10593* (2022).

[145] Weixin Ma, Shoudong Huang, and Yuxiang Sun. “Triplet-Graph: Global Metric Localization Based on Semantic Triplet Graph for Autonomous Vehicles”. In: *IEEE Robotics and Automation Letters* 9.4 (2024), pp. 3155–3162.

[146] Lu Xiong et al. “Integrated Decision Making and Planning Based on Feasible Region Construction for Autonomous Vehicles Considering Prediction Uncertainty”. In: *IEEE Transactions on Intelligent Vehicles* (2023).

[147] Lingguang Wang, Carlos Fernandez, and Christoph Stiller. “High-level decision making for automated highway driving via behavior cloning”. In: *IEEE Transactions on Intelligent Vehicles* 8.1 (2022), pp. 923–935.

[148] Lukas Hoyer et al. “Short-term prediction and multi-camera fusion on semantic grids”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*. 2019, pp. 0–0.

[149] Apratim Bhattacharyya, Mario Fritz, and Bernt Schiele. “Bayesian prediction of future street scenes using synthetic likelihoods”. In: *arXiv preprint arXiv:1810.00746* (2018).

[150] Mrigank Rochan et al. “Future semantic segmentation with convolutional lstm”. In: *arXiv preprint arXiv:1807.07946* (2018).

[151] Jian-Fang Hu et al. “Apanet: Auto-path aggregation for future instance segmentation prediction”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44.7 (2021), pp. 3386–3403.

[152] Zihang Lin et al. “Predictive feature learning for future segmentation prediction”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 7365–7374.