PolarPoint-BEV: Bird-Eye-View Perception in Polar Points for Explainable End-to-End Autonomous Driving

Yuchao Feng, *Graduate Student Member, IEEE*, and Yuxiang Sun, *Member, IEEE*

***Abstract*—End-to-end autonomous driving has attracted great attentions in recent years. Compared to traditional modular methods, end-to-end methods are more scalable in complex traffic environments, but they lack explainability. Many methods have been proposed to increase the explainability for end-to-end autonomous driving, such as using semantic bird-eye-view (BEV) maps. BEV maps can explain the outputs of end-to-end methods by showing how the networks perceive and understand surrounding traffic environments. However, there are some limitations in traditional semantic BEV maps. For instance, all regions of traffic scenes are treated equally, but the fact is that the regions near the ego vehicle are normally more critical to vehicle safety. Moreover, traditional BEV maps represent traffic scenes in the fine-grained pixel-level mode, which leads to much computational cost. To address these issues, we introduce a novel lightweight BEV perception method, PolarPoint-BEV, which prioritizes the regions according to object distances to the ego vehicle. Furthermore, we propose an explainable end-to-end autonomous driving network to investigate the influence of our PolarPoint-BEV in terms of driving performance. Experimental results demonstrate that our PolarPoint-BEV improves both the driving capability and explainability of the network.**

***Index Terms*—End-to-end autonomous driving, BEV perception, explainable AI (XAI).**

I. Introduction

IN RECENT years, end-to-end autonomous driving has become increasingly popular. It takes as input raw sensory data and outputs waypoints or directly outputs control actions [1], [2], [3], [4], [5], [6]. The waypoints can be fed into low-level controllers, such as Proportional-Integral-Derivative or Model Predictive Control, to produce control actions. Compared to traditional methods that consist of various modules [7], such as localization [8], [9], [10], [11], perception [12], [13], [14], planning [15] and control [16], end-to-end methods could avoid accumulative errors from different modules and be more scalable to complex scenarios. Many research efforts [17], [18], [19], [20], [21], [22] have been made in end-to-end autonomous driving and notable research progress has been witnessed. However, end-to-end methods are usually not explainable since deep neural networks are black boxes. The lack of explainability may lead to unexpected errors and dangers, which hinders the wide deployment of end-to-end methods in real traffic environments. To provide explanations for end-to-end methods, many eXplainable Artificial Intelligence (XAI) techniques have been proposed, such as generating semantic bird-eye-view (BEV) maps of traffic scenes for explanations.

Semantic BEV map generation has recently emerged as a popular research topic in autonomous driving, because semantic BEV map is a straightforward data representation for downstream tasks, such as trajectory planning [23] and control [24]. Moreover, semantic BEV maps could show how autonomous driving networks perceive and understand surrounding traffic environments, which could be used as explanations for end-to-end driving. Recently, many research efforts [25], [26], [27], [28], [29], [30], [31], [32], [33] have been paid in this area.

Despite the success of the traditional semantic BEV map, it still has limitations. Firstly, all the regions in traffic scenes are treated equally in traditional semantic BEV maps. However, it is believed that objects in the regions closer to the ego vehicle are more safety-critical [34], [35]. In the traditional semantic BEV map, faraway regions that may already have lower impacts on immediate safety concerns, still receive the same level of attention and consideration as those close to the ego vehicle. The lack of discrimination for the regions with different distances may cause the methods to focus on unimportant regions, and hence miss critical information that is vital to vehicle safety. Secondly, traditional BEV maps describe traffic scenes in a dense grid-like data representation at the pixel level. This kind of pixel-level representation is considered effective for autonomous driving [25], [26], [27], [28], [29], [30]. However, compared to sparse representations, utilizing dense maps in deep neural networks incurs higher computational, communication, and memory costs. With limited computational resources available on vehicles, the increased computational cost may result in intolerable delays, thereby imposing potential safety hazards [36], [37], [38].

Manuscript received 26 December 2023; revised 10 January 2024; accepted 17 January 2024. Date of publication 1 February 2024; date of current version 30 June 2025. This work was supported in part by Guangdong Basic and Applied Basic Research Foundation under Grant 2022A1515010116, and in part by the National Natural Science Foundation of China under Grant 62003286. *(Corresponding author: Yuxiang Sun.)*

Yuchao Feng is with the Department of Mechanical Engineering, The Hong Kong Polytechnic University, Kowloon, Hong Kong (e-mail: yuchao.feng@connect.polyu.hk).

Yuxiang Sun is with the Department of Mechanical Engineering, City University of Hong Kong, Kowloon, Hong Kong (e-mail: yx.sun@cityu.edu.hk).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TIV.2024.3361093.

Digital Object Identifier 10.1109/TIV.2024.3361093

To provide a solution to the above issues, we propose a novel BEV perception method, named PolarPoint-BEV. In contrast to treating all regions of traffic scenes uniformly, our polar point BEV maps pay more attention to the regions that are near the ego vehicle. In addition, unlike traditional dense BEV maps, our polar point BEV map is a more lightweight way to represent traffic scenes since it is a sparse representation. To investigate whether the proposed PolarPoint-BEV can increase explainability and driving performance, an end-to-end autonomous driving network is proposed and named as eXplainable Planning (XPlan). XPlan is designed in a multi-task architecture to jointly predict control commands and polar point BEV maps as explanations. We evaluate the driving performance of our network in the CARLA simulator [39]. Experimental results show that both the driving performance and explainability of the end-to-end network are enhanced by our PolarPoint-BEV. The contributions of this work are summarized as follows:

- We propose a novel BEV perception method, PolarPoint-BEV, to address the limitations of the traditional BEV maps. Our code and dataset are publicly available.[1]
- We design a novel explainable end-to-end autonomous driving network, and evaluate it using the CARLA simulator.
- The experimental results show that our PolarPoint-BEV can improve the driving performance and explainability of the proposed network.

The remainder of this paper is structured as follows. Section II reviews the related work. Section III presents the details of our methods. Section IV discusses the experimental results. Conclusions and future work are drawn in the last section.

II. Related Work

A. End-to-End Autonomous Driving

In recent years, many end-to-end autonomous driving networks have been proposed and achieved notable progress. For example, Chen et al. [19] designed an end-to-end method that learns the driving policies from the experiences of all nearby vehicles, named the Learning from All Vehicles (LAV) network. The LAV network takes as input the multi-modal sensory information and outputs the future trajectories for all the detected vehicles. Wu et al. [20] proposed the Trajectory-guided Control Prediction (TCP) network that comprised a trajectory branch and a multi-step control branch. Chitta et al. [21] proposed a transformer-based network with the mechanism of integrating the image and LiDAR representations using self-attention. Hu et al. [22] designed a planning-oriented framework named the Unified Autonomous Driving (UniAD). In UniAD, the query design is proposed and utilized as the interface to connect all components of the system, by which the knowledge from the intermediate tasks could be exchanged and used to improve the planning.

B. Explainable Autonomous Driving

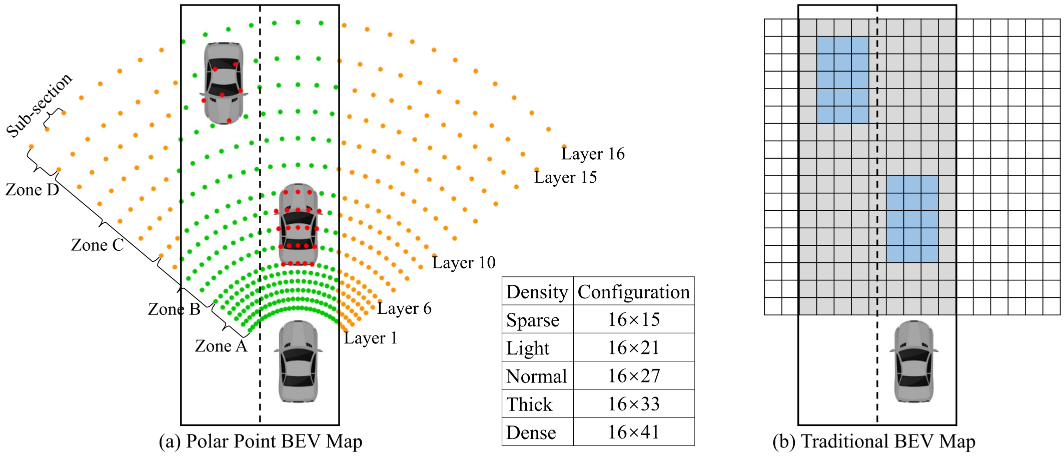
Given the crucial importance of explainability in autonomous driving, extensive research endeavours [40], [41], [42], [43], [44], [45], [46] have been undertaken in explainable autonomous driving in recent years. Kim et al. [40] designed an interpretable deep learning model for autonomous driving, utilizing visual attention heat maps to identify regions that have a causal impact on driving actions. Chen et al. [41] introduced an interpretable deep reinforcement learning method for end-to-end autonomous driving, in which the interpretable explanation is provided by generating a bird-view semantic mask. Xu et al. [42] proposed a multi-task network that jointly predicts driving actions and corresponding natural-language explanations. Teng et al. [43] introduced a two-stage autonomous driving model for complex traffic scenarios, named Hierarchical Interpretable Imitation Learning (HIIL). Within HIIL, traditional semantic BEV maps are utilized to explain the surrounding environments and failure cases of the ego vehicle. Renz et al. [44] proposed an explainable planning network for autonomous driving using a standard transformer architecture, in which the explanation is given by highlighting the objects in the scene that are relevant and crucial for the decision of the agent.

C. Traditional Semantic BEV Maps

The traditional semantic BEV maps have been widely applied in autonomous driving to increase explainability, and numerous downstream tasks are built upon accurate BEV perception. The BEV map could be divided into two categories: point cloud-based methods and visual image-based methods. For point cloud-based methods, radar or LiDAR sensors are employed to generate BEV maps. For example, Yang et al. [47] proposed a method to enhance the robust perception of dynamic objects in autonomous driving by combining radar and LiDAR. Kempen et al. [48] presented an end-to-end network for the task of occupancy grid mapping using LiDAR point clouds.

The visual image-based methods utilize images produced by visual cameras and convert visual information from the perspective viewpoint to BEV representation. Most visual image-based methods [25], [26], [27], [28], [29], [30] rasterize the BEV space along the Cartesian axes to obtain a uniformly distributed rectangular BEV map. For example, Philion et al. [25] proposed an end-to-end network to infer BEV representations of traffic scenes from arbitrary number of cameras. Pan et al. [27] proposed the View Parsing Network (VPN) for the task of the cross-view semantic segmentation. Liu et al. [30] proposed Position Embedding Transformation (PETR) for the 3D object detection by encoding the position information of 3D coordinates into image features and generating 3D position-aware features. To alleviate the foreshortening effect of camera imaging, some visual image-based methods [31], [32], [33] apply the Polar coordinate system to rasterize the BEV space. For example, Liu et al. [31] proposed to rasterize the BEV space both angularly and radially. Then, the associations of polar grids are modelled and rearranged to the array-like representation. Jiang et al. [32]

[1] [Online]. Available: https://github.com/lab-sun/PolarPoint-BEV



Density	Configuration
Sparse	16×15
Light	16×21
Normal	16×27
Thick	16×33
Dense	16×41

(a) Polar Point BEV Map

(b) Traditional BEV Map

Fig. 1. Schematic diagram of our polar point BEV map (Sub-fig. (a)) and the traditional BEV map (Sub-fig. (b)). In our polar point BEV map, the orange, red, green colors represent background, vehicle and road. The figure is best viewed in color.

proposed the Polar Transformer (PolarFormer) for 3D object detection in BEV, in which a cross-attention-based Polar detection head was designed to deal with irregular Polar grids.

D. Difference From Previous Works

Different from the aforementioned methods, we propose an explainable end-to-end network by jointly predicting the control commands and polar point BEV map. The closest work to ours is the network proposed by Liu et al. [31]. There are mainly two differences between the two works. Firstly, the BEV map predicted by [31] is uniformly distributed in the radial direction with the fine-grained pixel-level representation, which still has the limitations of the traditional BEV map. The lack of discrimination for the regions with different distances may cause their network to focus on unimportant regions and miss the critical details for vehicle safety. In contrast, our network can predict the polar point BEV map with varying radial intervals, allowing our network to pay more attention to the regions that are more critical for safety. Secondly, the network [31] only predicts the semantic BEV map, and the influence of the BEV map prediction on the driving performance is not investigated in the work. On the contrary, our work carefully investigates the influence of the PolarPoint-BEV on the driving performance of the autonomous agent by using the CARLA simulator.

III. The Proposed Network

A. The PolarPoint-BEV

As aforementioned, traditional BEV methods have some limitations. To overcome these limitations, our proposed PolarPoint-BEV employs the polar point BEV map to show how the network perceives and understands the surrounding environment, thereby explaining the output of the end-to-end autonomous driving network. Fig. 1 shows the comparison between our polar point BEV map and the traditional BEV map. The traditional BEV map describes the traffic scene by using a uniformly distributed rectangular grid map along the Cartesian axes. The polar point BEV map describes the traffic scene by applying a sequence of points scattered around the ego vehicle. Each point on the polar point BEV map has a semantic class depending on the object overlaid at that point. Specifically, there are 3 semantic classes (ranging from $\{0\}$ to $\{2\}$) for each point on the polar point BEV map. Different semantic classes are represented by different colors in Fig. 1. Here, $\{0\}$, $\{1\}$ and $\{2\}$ refer to the background, vehicle and road, which are represented by the orange points, red points and green points, respectively.

TABLE I
Details of Each Zone for the Polar Point BEV Map With Normal Configuration

Zone	Scope	Interval	Density
Zone A	Layer 1 to 6	$0.5m$	3.91 m^{-2}
Zone B	Layer 6 to 10	$1.0m$	1.21 m^{-2}
Zone C	Layer 10 to 15	$1.5m$	0.45 m^{-2}
Zone D	Layer 15 to 16	$2.0m$	0.42 m^{-2}

The position of each point on the polar point BEV map is described by polar coordinates. In the angular direction, the field of view (FOV) of the polar point BEV map is 100°, which is consistent with the horizontal FOV of the front-view camera. The FOV of the polar point BEV map is divided into sub-sections in the angular direction. Here, we test five different configurations regarding the number of sub-sections in the polar point BEV map, from 15 sub-sections to 41 sub-sections (as shown in Fig. 1). In the radial direction, the polar point BEV map contains 16 layers. Therefore, the polar point BEV map can be denoted as $P_i \in \{0, 1, 2\}^{16 \times n}$, where 16 is the number of layers in the radial direction, n is the number of sub-sections depending on the configuration of polar point BEV map.

The 16 layers in the radial direction could be divided into four zones, including Zone A, Zone B, Zone C and Zone D. As summarized in Table I, each zone has different intervals between layers. Based on the observation the regions that are closer to the ego vehicle are more likely to be critical to safety, Zone A and Zone B have relatively small intervals, and Zone C and Zone D have relatively large intervals (Zone A $<$ Zone B $<$ Zone C $<$ Zone D). Here, we define the density of the polar point BEV map as the ratio between the number of semantic

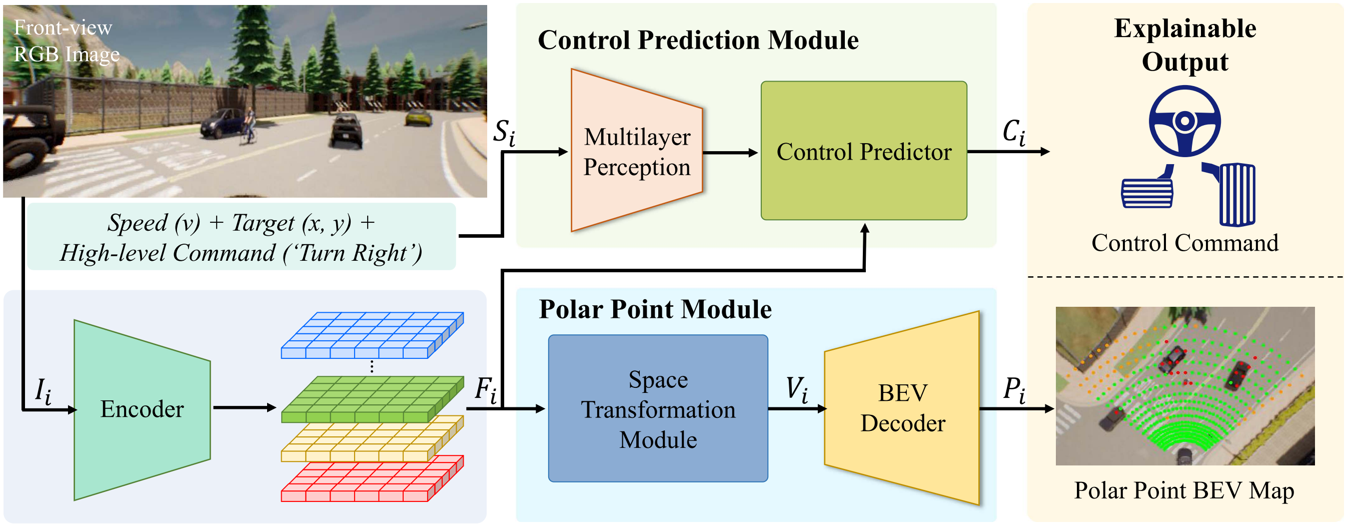


Fig. 2. Structure of our proposed XPlan network. This explainable end-to-end network takes as input the front-view RGB image as well as navigation information, and outputs control commands along with polar point BEV map as the explanations. In the polar point BEV map, the orange, red, green colors represent background, vehicle and road. The figure is best viewed in color.

points and the area of the region. Table I shows the density for the different zones in the polar point BEV map with normal configuration. In this case, the density of Zone A is about 9 times larger than the density of Zone D. In contrast, the semantic points in traditional semantic BEV maps are uniformly distributed to depict the traffic scene. Therefore, instead of treating all regions of the traffic scene equally, the polar point BEV map places more focus on the regions closer to the ego vehicle.

B. The Network Structure

Fig. 2 shows the structure of our proposed XPlan network for autonomous driving. The XPlan network mainly consists of three components: encoder, control prediction (C-P) module, and Polar-Point (P-P) module. The network takes as input the front-view RGB image I_i and the navigation information S_i to output the control commands C_i along with the polar point BEV map P_i as the explanations. The $I_i \in \mathbb{R}^{\mathrm{h}\times\mathrm{w}\times\mathrm{c}}$ denotes the front-view RGB image, where h, w and c respectively represent height, width and number of channels for the image. The navigation information is denoted as S_i, which contains the current speed v, the future target (x, y) and the high-level command. The control command C_i contains values for the steer, throttle, and brake. Therefore, the whole process of our XPlan network can be described as follows:

$$\text{XPlan: } (I_i, S_i) \rightarrow \left(C_i, P_i \in \{0, 1, 2\}^{16\times n}\right). \tag{1}$$

The ResNet-34 is adopted as the encoder to extract the feature maps F_i of the input images. Then, the feature maps are fed into the C-P module and P-P module, respectively. The C-P module is designed based on the Trajectory-guided Control Prediction (TCP) network [20], we refer readers to [20] for more details of the TCP network. The C-P module takes as input the feature maps F_i and navigation information S_i and outputs the control commands C_i. Therefore, the process of the C-P module is described as $(F_i, S_i) \rightarrow C_i$.

The P-P module is designed to generate the polar point BEV map, P_i of the traffic scene, to explain the control commands of the network. The P-P module is divided into two modules: the space transformation module and the BEV decoder. In the space transformation module, the feature maps of the front-view space are flattened and fed into the Multilayer Perceptron (MLP) to learn the relations between the front-view space and the BEV space. By using the MLP, the perspective of feature maps from the front-view space is transformed into the BEV space. Then, the flattened feature maps of the BEV space are reshaped to the original shape of feature maps and fed into the BEV decoder to output the polar point BEV map, which is the sequence of semantic points. The process of the P-P module is described as: $F_i \rightarrow P_i$.

There are 5 different configurations for the polar point BEV map, ranging from sparse to dense configurations. Therefore, the XPlan network that predicts the normal polar point BEV map (16 × 27) is noted as XPlan-N. Similarly, the XPlan network that predicts the sparse/light/thick/dense polar point BEV map is noted as XPlan-S/L/T/D.

C. Dataset and Training Details

The dataset is collected in the CARLA simulator by using the randomly generated routes under various weather and lighting conditions, including ClearNoon, CloudyNoon, WetNoon, WetCloudyNoon, SoftRainNoon, MidRainyNoon, HardRainNoon, ClearSunset, CloudySunset, WetSunset, WetCloudySunset, MidRainSunset, HardRainSunset, SoftRainSunset, ClearNight, CloudyNight, WetNight, WetCloudyNight, SoftRainNight, MidRainyNight, and HardRainNight. The dataset is generated by utilizing Roach [49] as the expert, which possesses privileged information of the traffic scene and various elements, such as roads, vehicles, pedestrians, traffic lights, etc. The dataset contains more than 92 k data batches from the 6 public towns (Town01, Town03, Town04, Town06, Town07 and Town10) offered by the CARLA simulator, which includes the scenarios of the small town, quiet rural community, and inner-city environment. Each data batch includes the front-view RGB image, current speed, future target, high-level command, ground truths of waypoints, control values and the BEV map, etc. The proposed XPlan network is trained based on the collected dataset. The evaluation of the XPlan network is based on the routes from [19], These routes contain traffic scenes from 2 towns (Town02 and Town05) from the CARLA simulator, which includes the scenarios of the small town and urban environment.

All the networks are trained with NVIDIA GeForce RTX 3090 GPU. The inference speeds of the networks are tested with NVIDIA GeForce RTX 3060 GPU. In the first stage, we pre-train the C-P module using the dataset collected from [20]. Then, we train and evaluate the whole XPlan network based on our dataset in the second stage. The Adam optimizer is applied with the initial learning rate of 1×10^{-4} and weight decay of 1×10^{-7}. A multi-task loss function is designed for the XPlan network, which is calculated as $\mathcal{L}_{total} = \mathcal{L}_{ctrl} + \lambda_1 \mathcal{L}_{bev}$. Here, $\mathcal{L}_{total}$ is the total loss, $\mathcal{L}_{ctrl}$ is the loss for the prediction of control commands, $\mathcal{L}_{bev}$ is the loss for the polar point BEV map generation. λ_1 denotes the weighting coefficient that determines the importance between the control command prediction and polar point BEV map generation. For the polar point BEV map, each semantic point has 3 classes, so $\mathcal{L}_{bev}$ could be calculated as $\mathcal{L}_{bev} = \mathcal{L}_{vehicle} + \lambda_2 \mathcal{L}_{road} + \lambda_3 \mathcal{L}_{backg}$. Here, the $\mathcal{L}_{vehicle}$, $\mathcal{L}_{road}$ and $\mathcal{L}_{backg}$ are the cross entropy loss for the vehicle, road and background, respectively. λ_2 and λ_3 refer to the weight coefficients that determine the importance between different classes in the polar point BEV map.

IV. Experimental Results and Discussions

A. Evaluation Metrics

The driving performance and the prediction performance of the polar point BEV map of our XPlan network are evaluated in the CARLA simulator. So, there are two objectives for the autonomous agent in the CARLA simulator: 1) to reach the designated destination safely and efficiently in a predefined path; and 2) to accurately predict the polar point BEV map of the traffic scene to explain the control commands. To evaluate the driving performance of our XPlan network, different metrics have been proposed in the CARLA simulator, including the 3 main metrics of the route completion, infraction score and driving score. The route completion refers to the percentage of the route completed by the agent. The infraction score shows the occurrences of rule violations throughout the route, such as infractions involving pedestrians, vehicles, road layouts, red lights, etc. The driving score is defined as the product of route completion and infraction score. Besides these 3 main metrics, the other metrics that target the specific performance are also calculated in this work, such as vehicle/layout collisions, red light/off-road infractions and agent blocked.

To evaluate the overall prediction performance of the polar point BEV map, the Intersection-over-Union (IoU) and the F1 score for the road, vehicle and background are calculated. However, for the calculation of the IoU and F1 score, all the points of the BEV map are treated equally with the same weight. Based on the principle that the region near the ego vehicle is more likely to be critical for safety, we propose a new metric named the weighted Intersection-over-Union (wIoU). The wIoU is calculated as:

$$\mathrm{wIoU} = \sum_{z=A}^{D} \left(\mathrm{mIoU}_z \times N_z\right), \quad N_z = L_z^{-1} / \sum_{z=A}^{D} \left(L_z^{-1}\right), \quad (2)$$

where, the mIoU_z is the mean IoU for each zone (from Zone A to Zone D) in the BEV map. The N_z is the normalized weight for each zone. The L_z^{-1} represents the reciprocal of the distance for each zone to the ego vehicle. In this way, the zone that is closer to the ego vehicle has a larger weight in the calculation of the wIoU.

B. Comparative Results

In this section, we first investigate the prediction performance of the polar point and traditional BEV maps. Here, we propose an end-to-end network to jointly predict the control commands of the agent and the corresponding traditional BEV map. The network is named the Planning-BEV (Plan-B) network. For the Plan-B network, the encoder and C-P module remain the same as the XPlan network. However, the P-P module is replaced with the traditional BEV generation module, which is designed based on the View Parsing Network (VPN) [27]. Both the XPlan and Plan-B networks are trained for 60 epochs with the same pre-trained weight for the C-P module. The weighting coefficients (λ_2 and λ_3 in the loss function) of the XPlan and Plan-B are the same.

Table II shows the comparative results of overall prediction performance for the polar point and traditional BEV maps. Each network is tested for 3 runs and we report the mean and standard deviations for both IoU and F1 score. As shown in Table II, both the mIoU and the overall F1 score of the polar point BEV map are close to the traditional BEV map, indicating that the overall prediction performance of the polar point and traditional BEV maps is at the same level. However, for vehicle and road, the IoU and F1 scores of the polar point BEV map are both higher than those of the traditional BEV map.

Table III displays the comparative results in terms of mIoU for different zones, and wIoU for the polar point and traditional BEV maps. In Zone A and Zone B, the prediction performance of the polar point BEV map is better than the traditional BEV map. On the contrary, the prediction performance of the traditional BEV map is better than the polar point BEV map in Zone D. In Zone C, the prediction performance of the polar point and traditional BEV maps is at the same level. For wIoU, the polar point BEV map is about 4% higher than the traditional BEV map. These results indicate that although the overall prediction performance of the polar point and traditional BEV maps is at the same level, the polar point BEV map has better prediction performance than the traditional BEV map in regions close to the ego vehicle.

Fig. 3 shows examples of the polar point and traditional BEV maps under different weather and lighting conditions, including SoftRainDawn, ClearNoon, CloudySunset and HardRainNight.

TABLE II
COMPARATIVE RESULTS OF THE OVERALL PREDICTION PERFORMANCE FOR THE POLAR POINT AND TRADITIONAL BEV MAPS

Network	IoU (%)				F1 Score			
	Vehicle	Road	Background	Mean	Vehicle	Road	Background	Overall
Plan-B	54.83±0.55	87.27±0.15	**93.07±0.15**	78.37±0.15	0.71±0.01	0.93±0.00	**0.96±0.00**	**0.95±0.00**
XPlan-N	**61.03±1.37**	**91.77±0.49**	86.93±1.08	**79.93±0.86**	**0.76±0.01**	**0.96±0.01**	0.93±0.01	0.94±0.01

The mean and standard deviations are calculated over 3 runs. The best results are highlighted in bold font.

TABLE III
COMPARATIVE RESULTS OF THE MIOU OF DIFFERENT ZONES AND THE WIOU FOR THE POLAR POINT AND TRADITIONAL BEV MAPS

Network	Zone A (%)	Zone B (%)	Zone C (%)	Zone D (%)	wIoU (%)
Plan-B	69.10±0.46	73.97±0.06	**81.90±0.30**	**89.00±0.35**	73.50±0.20
XPlan-N	**73.93±1.10**	**83.47±0.80**	81.57±1.29	69.13±0.81	**76.50±0.78**

The mean and standard deviations are calculated over 3 runs. The best results are highlighted in bold font.

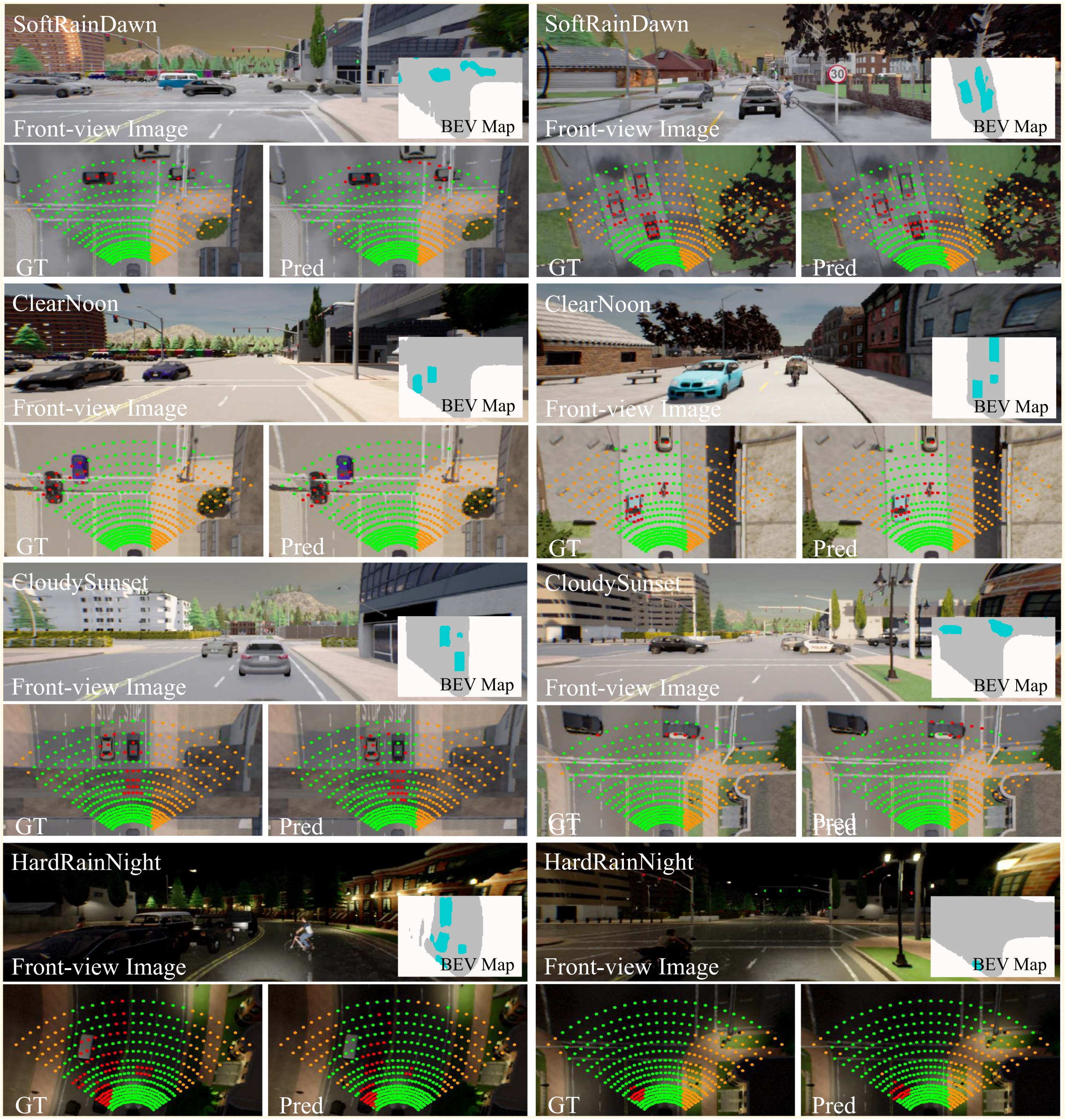


Fig. 3. Sample qualitative results of the polar point and traditional BEV maps. GT and Pred refer to ground truth and prediction. In the polar point BEV maps, the points with orange, red and green colors respectively represent the background, vehicle and road. The figure is best viewed in color.

TABLE IV
COMPUTATIONAL COMPLEXITY FOR DIFFERENT NETWORKS

Network	Param	MACs	FPS
TCP	25.77M	17.09G	137.88
Plan-B	38.58M	27.21G	90.08
XPlan-N	27.57M	17.58G	132.66

The inference speed is tested using an NVIDIA GEFORCE RTX 3060 GPU.

As shown in Fig. 3, both the polar point and traditional BEV maps could accurately describe the traffic scenes and show how the network perceives and understands the surrounding traffic environment. Therefore, these two methods both provide effective explanations for the control commands generated by the end-to-end networks.

Table IV shows the computational complexity for different networks. The TCP network could be considered as the XPlan/Plan-B network without the P-P/BEV module. Compared with the TCP network, the computational complexity of the XPlan-N network is slightly increased. The number of parameters (Param) and multiply–accumulate operations (MACs) in the XPlan-N are about 7% and 3% higher than the TCP network, respectively. The frames-per-second (FPS) for the inference of the XPlan-N network is about 4% lower than the TCP network. However, the Plan-B network experiences a notable increase in computational complexity when compared with the TCP network. The Param and MACs in the Plan-B network are about 50% and 60% higher than the TCP network, respectively. The inference FPS of the Plan-B network is about 35% lower than the TCP network. Therefore, we could conclude that the PolarPoint-BEV is a lightweight and efficient method to describe the traffic scene and explain the control commands of autonomous agents. By adopting the polar point BEV map, the XPlan network is able to reduce the computational cost associated with the BEV generation. The polar point BEV map balances efficiency and accuracy, making it an efficient solution for autonomous driving systems where computing resources and real-time performance are of great importance.

We also investigate the influence of the polar point and traditional BEV maps on the driving performance of the end-to-end network. Table V shows the comparative results of the driving performance for different networks. It is worth noting that the results of the LAV network are from the work [19], which is trained based on the dataset with 186 K data from 4 towns and tested on the same routes as other networks listed in Table V. As shown in Table V, both the XPlan-N and Plan-B networks present better driving performance when compared with the LAV and TCP networks. Considering the fact that the TCP network could be considered as the XPlan/Plan-B network without the P-P/BEV module, it is safe to say that the PolarPoint-BEV/BEV generation has a positive impact on the driving performance of the XPlan/Plan-B network. Furthermore, the driving performance of the XPlan-N network is better than the Plan-B network. For the mean values of the three main metrics, the driving score of the XPlan-N network is about 9% higher than the Plan-B network; the route completion of the XPlan-N network is about 2% higher than the Plan-B network; the infraction score of XPlan-N network is about 4% higher than the Plan-B network. We conjecture the reasons why the driving performance of the XPlan-N network is better than the Plan-B network as follows:

1) The polar point BEV map applies the mechanism of paying more attention to the regions that are near the ego vehicle. Furthermore, the comparative results show that the polar point BEV map has better prediction performance than the traditional BEV map in regions close to the ego vehicle. Based on the observation that the regions close to the ego vehicle are more likely to be critical to vehicle safety, we believe that the polar point BEV map is able to enhance the system capability to perceive and respond to potential dangers, resulting in safer and more reliable autonomous driving performance.
2) Even though the overall prediction performance of the polar point and traditional BEV maps is at the same level, the polar point BEV map processes better prediction performance for the classes of vehicle and road. Compared with the background, the vehicle and road are more relevant and crucial for the safety of the autonomous agent. So, the better prediction performance of the vehicle and road leads to the better driving performance of the autonomous agent.

Considering the domain gap between the synthetic environments and real-world environments, we also investigate the prediction performance of the PolarPoint-BEV method on the nuScenes [50] dataset. Table VI shows the comparative results of overall prediction performance for the polar point and traditional BEV maps that are predicted by different networks. The polar point BEV map is predicted by using the XPlan-N network without the C-P module. As shown in Table VI, both the mIoU and the overall F1 score of the polar point BEV map are close to the traditional BEV map. This indicates that the overall prediction performance of the polar point and traditional BEV maps is at the same level on the nuScenes dataset.

The mIoU of different zones and the wIoU for the polar point and traditional BEV maps on the nuScenes dataset are shown in Table VII. Similar to the testing results in the Carla simulator, the prediction performance of the polar point BEV map is better than the traditional BEV map in Zone A and Zone B, and worse than the traditional BEV map in Zone D. The wIoU of the polar point BEV map is higher than the traditional BEV map on the nuScenes dataset. These results validate that the polar point BEV map has better prediction performance than the traditional BEV map in regions close to the ego vehicle. Table VII also shows the computational complexity for different networks. The computational complexity of the XPlan-N network (without the C-P module) is much lower than other networks. This result demonstrates that the PolarPoint-BEV is a lightweight and efficient method to describe the traffic scene in real-world environments.

C. Ablation Study

In the ablation study, we first investigate the prediction performance of the polar point BEV maps with different configurations, and the influence of different polar point BEV maps

TABLE V
COMPARATIVE RESULTS OF THE DRIVING PERFORMANCE FOR DIFFERENT NETWORKS

Networks	Driving Score	Route Completion	Infraction Score	Vehicle Collisions	Layout Collisions	Red Light Infractions	Off-road Infractions	Agent Blocked
LAV [19]	45.20±6.35	*91.55±5.61*	0.49±0.06	0.92±0.42	0.33±0.50	0.28±0.28	0.27±0.01	0.01±0.02
TCP [20]	53.10±2.18	78.66±3.86	**0.67±0.01**	0.09±0.03	0.15±0.02	0.01±0.02	0.05±0.02	0.16±0.04
Plan-B	*55.19±1.48*	90.34±4.41	0.63±0.03	0.11±0.03	0.00±0.00	0.03±0.03	0.02±0.01	0.03±0.03
XPlan-N	**60.41±3.31**	**92.62±0.96**	*0.66±0.04*	0.07±0.03	0.03±0.01	0.03±0.01	0.04±0.01	0.05±0.01

The mean and standard deviations are calculated over 3 runs. The best and second-best results for the three main metrics are highlighted in bold font and italic font.

TABLE VI
COMPARATIVE RESULTS OF THE OVERALL PREDICTION PERFORMANCE FOR THE POLAR POINT AND TRADITIONAL BEV MAPS ON THE NUSCENES [50] DATASET

Network	IoU (%)					F1 Score				
	Vehicle	Road	Divider	Background	Mean	Vehicle	Road	Divider	Background	Overall
VED [26]	35.5	76.6	29.8	**92.1**	58.5	0.52	0.87	0.46	**0.96**	**0.91**
VPN [27]	37.7	77.8	29.7	91.8	59.2	0.55	0.87	0.46	**0.96**	**0.91**
PON [28]	38.6	75.5	**34.7**	92.0	60.2	0.56	0.86	**0.52**	**0.96**	0.90
XPlan-N	**52.7**	**85.9**	27.3	75.4	**60.3**	**0.69**	**0.92**	0.43	0.86	0.88

The polar point BEV map is predicted by the XPlan-N network without the C-P module.
The best results are highlighted in bold font.

TABLE VII
COMPARATIVE RESULTS OF THE PREDICTION PERFORMANCE (MIOU OF DIFFERENT ZONES AND THE WIOU) AND THE COMPUTATIONAL COMPLEXITY FOR DIFFERENT NETWORKS ON THE NUSCENES [50] DATASET

Network	Prediction Performance					Complexity		
	Zone A (%)	Zone B (%)	Zone C (%)	Zone D (%)	wIoU (%)	Param	MACs	FPS
VED [26]	53.9	54.6	60.2	67.1	56.0	45.59	159.09	59.61
VPN [27]	54.2	54.8	61.3	69.5	56.6	37.15	43.59	79.11
PON [28]	49.6	57.5	**68.3**	**72.8**	55.7	37.94	62.10	32.44
XPlan-N	**54.8**	**63.5**	65.0	59.7	**58.4**	**26.14**	**28.24**	**109.72**

The polar point BEV map is predicted by the XPlan-N network without the C-P module.
The inference speed is tested using an NVIDIA GEFORCE RTX 3060 GPU. The best results are highlighted in bold font.

TABLE VIII
COMPUTATIONAL COMPLEXITY FOR THE XPLAN NETWORKS WITH DIFFERENT CONFIGURATIONS OF THE POLAR POINT BEV MAP

Configuration	Param	MACs	FPS
XPlan-S	27.57M	17.55G	134.33
XPlan-L	27.57M	17.56G	134.09
XPlan-N	27.57M	17.58G	132.66
XPlan-T	27.57M	17.59G	131.62
XPlan-D	27.57M	17.61G	130.52

The inference speed is tested using an NVIDIA GEFORCE RTX 3060 GPU.

on driving performance. Here, 5 different configurations of the polar point BEV map are considered, with the semantic point number ranging from 16×15 to 16×41. Table VIII shows the computational complexity of the XPlan networks with different configurations of the polar point BEV map. It shows that these XPlan networks exhibit very close computational costs, and more importantly, they all offer lightweight ways to describe the traffic scenes and explain the control commands.

The left figures of Fig. 4 summarize the prediction performance for the polar point BEV maps with different configurations. As shown in Fig. 4(a), the polar point BEV map with normal configuration exhibits the highest mIoU, which is about 3% higher than the lowest mIoU from the polar point BEV map with sparse configuration. For the F1 score shown in Fig. 4(b), the highest overall F1 score from the polar point BEV map with normal configuration is about 3% higher than the lowest value from the polar point BEV map with light configuration. These results show that the prediction performance of different polar point BEV maps is at the same level, which demonstrates the robustness and reliability of our PolarPoint-BEV.

Table IX summarizes the mIoU of different zones and the wIoU for polar point BEV maps with different configurations. The polar point BEV map with normal configuration exhibits the highest mIoU in all different zones. From Zone A to Zone D, the mIoU of the polar point BEV map with normal configuration is about 4%, 5%, 4% and 7% higher than the lowest value, respectively. The wIoU of the polar point BEV map with normal configuration is about 4% higher than the lowest value. These results validate the robustness and reliability of our proposed PolarPoint-BEV.

The influence of different polar point BEV maps on the driving performance of the XPlan network is also investigated. The right figures of Fig. 4 show the driving performance of the XPlan networks with different configurations of the polar point BEV

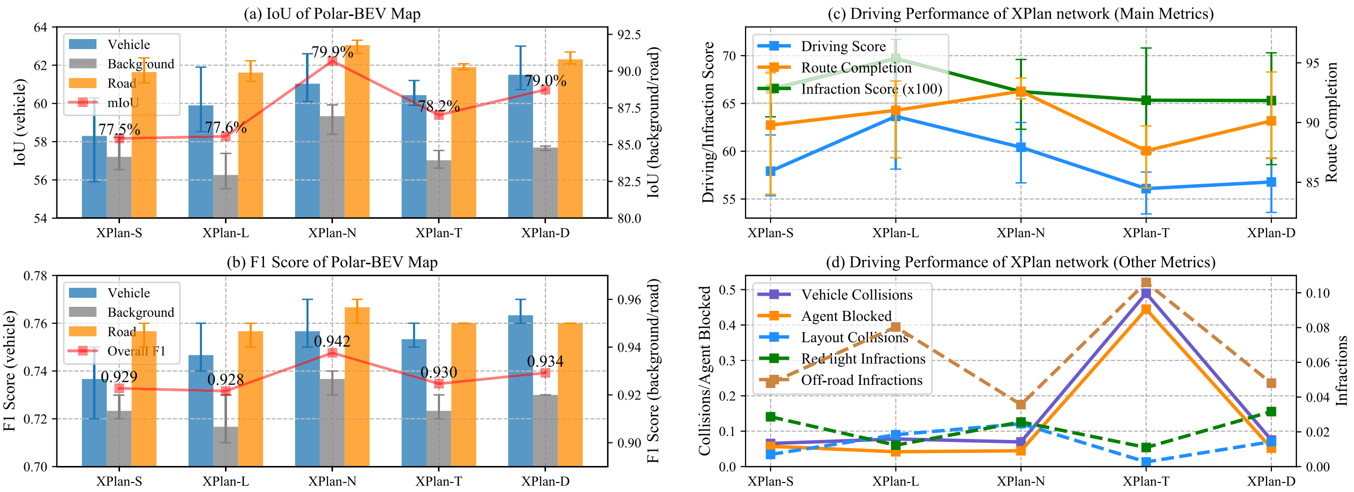


Fig. 4. Ablation study results of the prediction performance of XPlan networks with different configurations of polar point BEV map. (a) and (b) show the prediction performance of the polar point BEV maps with different configurations. (c) and (d) show the driving performance of the XPlan networks with different configurations of the polar point BEV map. The figure is best viewed in color.

TABLE IX
Ablation Study Results of the mIoU of Different Zones and the wIoU for Polar Point BEV Maps With Different Configurations

Network	Zone A (%)	Zone B (%)	Zone C (%)	Zone D (%)	wIoU (%)
XPlan-S	71.17±1.56	79.47±1.44	80.13±1.45	68.17±0.49	73.80±1.40
XPlan-L	71.67±1.07	79.87±1.42	78.33±2.10	66.97±2.52	73.82±1.23
XPlan-N	**73.93±1.10**	**83.47±0.80**	**81.57±1.29**	**69.13±0.81**	**76.50±0.78**
XPlan-T	72.83±0.40	80.73±0.76	79.43±1.01	64.77±1.33	74.61±0.55
XPlan-D	72.73±0.47	81.97±0.93	80.90±0.70	65.53±1.70	75.06±0.30

The mean and standard deviations are calculated over 3 runs. The best results are highlighted in bold font.

TABLE X
Ablation Study Results of the Driving Performance for the XPlan-N Network With Different Configurations of the C-P Module

Network	Driving Score	Route Completion	Infraction Score
XPlan-N (fixed)	57.33±0.80	86.74±4.96	0.66±0.05
XPlan-N (not fixed)	60.41±3.31	92.62±0.96	0.66±0.04

The mean and standard deviations are calculated over 3 runs.

map. Each XPlan network is tested three times. As shown in Fig. 4(c), the XPlan-L network exhibits the highest driving score and infraction score. The driving score of the XPlan-L network is about 13.5% higher than the lowest value from the XPlan-T network. The infraction score from the XPlan-L network is about 6.7% higher than the lowest value from the XPlan-D network. For the route completion, the XPlan-N network exhibits the highest driving score, which is about 5.7% higher than the lowest value from the XPlan-T network.

As aforementioned, the parameters of the C-P module are not fixed during the second stage of the training process. To investigate the influence of fixing the parameters of the C-P module on the driving performance of the proposed network, we perform a comparative experiment between the XPlan-N network with the C-P module fixed and the XPlan-N network with the C-P module not fixed. These two networks are both trained for 60 epochs with the same pre-trained weight for the C-P module. As shown in Table X, the driving performance of the XPlan-N network with the C-P module not fixed is better than the XPlan-N network with the C-P module fixed. These results show that not fixing the parameters of the C-P module in the second stage of training could improve the driving performance of the proposed network.

D. Limitations

Despite the superiority of the proposed PolarPoint-BEV method and the XPlan network, they still have some limitations. Firstly, the proposed polar point BEV map contains only 3 classes. To better describe the traffic scene, more classes, such as road dividers, road signs, pedestrians and other objects, should be considered. This could enable a more comprehensive and detailed description of the traffic scene, which may increase the explainability of the end-to-end autonomous driving systems. Secondly, the interval between layers in the polar point BEV map may not be optimal, particularly when dealing with some corner cases of diverse objects with different sizes and shapes. To better optimize the interval between layers in the polar point BEV map, different configurations of layer interval could be tried to identify the most effective polar point BEV map configuration.

V. Conclusions and Future Work

In this work, we proposed a novel BEV perception method, PolarPoint-BEV, to address the limitations of traditional BEV methods in explainable end-to-end autonomous driving. Based on the observations that the region near the ego vehicle is more likely to be critical for safety, and the fine-grained pixel-level mode of the traditional BEV map may be unnecessary, we proposed the polar point BEV map that uses a sequence of semantic points scattered around the ego vehicle to describe the traffic scene. To investigate the influence of the polar point BEV map on the driving performance in an end-to-end system, a multi-task explainable network is designed to jointly predict the control commands and the polar point BEV maps. The experimental results show that the application of the PolarPoint-BEV improves both the driving performance and explainability of the network.

Regarding future work, there are many interesting directions related to our PolarPoint-BEV. For example, one promising research direction could be how to realize the downstream tasks, such as trajectory planning, on the polar point BEV maps. With the polar point BEV maps, researchers can explore innovative strategies to enhance trajectory planning algorithms, enabling more precise and efficient navigation in complex urban scenarios. In addition, considering the fact that the format of the polar point BEV map is aligned with the point clouds produced by radar or LiDAR sensors, it would be an interesting direction to investigate how to perceive the traffic scene in the multi-modal mode based on the polar point BEV map and point clouds. Finally, it would be interesting to investigate the feasibility of deploying our PolarPoint-BEV in real-world dynamic scenarios and test the runtime speed on edge computing devices, which is important in real-world applications.

REFERENCES

[1] P. S. Chib and P. Singh, "Recent advancements in end-to-end autonomous driving using deep learning: A survey," *IEEE Trans. Intell. Veh.*, early access, doi: 10.1109/TIV.2023.3318070.

[2] P. Cai, H. Wang, Y. Sun, and M. Liu, "DQ-GAT: Towards safe and efficient autonomous driving with deep Q-learning and graph attention networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 11, pp. 21102–21112, Nov. 2022.

[3] L. Chen et al., "Milestones in autonomous driving and intelligent vehicles: Survey of surveys," *IEEE Trans. Intell. Veh.*, vol. 8, no. 2, pp. 1046–1056, Feb. 2023.

[4] P. Cai, S. Wang, Y. Sun, and M. Liu, "Probabilistic end-to-end vehicle navigation in complex dynamic environments with multimodal sensor fusion," *IEEE Robot. Automat. Lett.*, vol. 5, no. 3, pp. 4218–4224, Jul. 2020.

[5] S. Teng et al., "Motion planning for autonomous driving: The state of the art and future perspectives," *IEEE Trans. Intell. Veh.*, vol. 8, no. 6, pp. 3692–3711, Jun. 2023.

[6] P. Cai, X. Mei, L. Tai, Y. Sun, and M. Liu, "High-speed autonomous drifting with deep reinforcement learning," *IEEE Robot. Automat. Lett.*, vol. 5, no. 2, pp. 1247–1254, Apr. 2020.

[7] B. Paden, M. Čáp, S. Z. Yong, D. Yershov, and E. Frazzoli, "A survey of motion planning and control techniques for self-driving urban vehicles," *IEEE Trans. Intell. Veh.*, vol. 1, no. 1, pp. 33–55, Mar. 2016.

[8] W. Ma, S. Huang, and Y. Sun, "Triplet-graph: Global metric localization based on semantic triplet graph for autonomous vehicles," *IEEE Robot. Automat. Lett.*, early access, doi: 10.1109/LRA.2024.3358752.

[9] G. Bresson, Z. Alsayed, L. Yu, and S. Glaser, "Simultaneous localization and mapping: A survey of current trends in autonomous driving," *IEEE Trans. Intell. Veh.*, vol. 2, no. 3, pp. 194–220, Sep. 2017.

[10] S. Lowry et al., "Visual place recognition: A survey," *IEEE Trans. Robot.*, vol. 32, no. 1, pp. 1–19, Feb. 2016.

[11] M. U. M. Bhutta, Y. Sun, D. Lau, and M. Liu, "Why-so-deep: Towards boosting previously trained models for visual place recognition," *IEEE Robot. Automat. Lett.*, vol. 7, no. 2, pp. 1824–1831, Apr. 2022.

[12] Z. Feng, Y. Guo, and Y. Sun, "CEKD: Cross-modal edge-privileged knowledge distillation for semantic scene understanding using only thermal images," *IEEE Robot. Automat. Lett.*, vol. 8, no. 4, pp. 2205–2212, Apr. 2023.

[13] Y. Sun, W. Zuo, and M. Liu, "RTFNet: RGB-thermal fusion network for semantic segmentation of urban scenes," *IEEE Robot. Automat. Lett.*, vol. 4, no. 3, pp. 2576–2583, Jul. 2019.

[14] S. Gao, Q. Wang, and Y. Sun, "S2G2: Semi-supervised semantic bird-eye-view grid-map generation using a monocular camera for autonomous driving," *IEEE Robot. Automat. Lett.*, vol. 7, no. 4, pp. 11974–11981, Oct. 2022.

[15] B. Li, Y. Ouyang, L. Li, and Y. Zhang, "Autonomous driving on curvy roads without reliance on frenet frame: A cartesian-based trajectory planning method," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 9, pp. 15729–15741, Sep. 2022.

[16] F. Tian, Z. Li, F.-Y. Wang, and L. Li, "Parallel learning-based steering control for autonomous driving," *IEEE Trans. Intell. Veh.*, vol. 8, no. 1, pp. 379–389, Jan. 2023.

[17] P. Cai, Y. Sun, H. Wang, and M. Liu, "VTGNet: A vision-based trajectory generation network for autonomous vehicles in urban environments," *IEEE Trans. Intell. Veh.*, vol. 6, no. 3, pp. 419–429, Sep. 2021.

[18] Z. Huang, J. Wu, and C. Lv, "Efficient deep reinforcement learning with imitative expert priors for autonomous driving," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 10, pp. 7391–7403, Oct. 2023.

[19] D. Chen and P. Krähenbühl, "Learning from all vehicles," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, New Orleans, LA, USA, 2022, pp. 17222–17231.

[20] P. Wu, X. Jia, L. Chen, J. Yan, H. Li, and Y. Qiao, "Trajectory-guided control prediction for end-to-end autonomous driving: A simple yet strong baseline," in *Proc. Adv. Neural Inf. Process. Syst.*, New Orleans, LA, USA, 2022, pp. 6119–6132.

[21] K. Chitta, A. Prakash, B. Jaeger, Z. Yu, K. Renz, and A. Geiger, "Transfuser: Imitation with transformer-based sensor fusion for autonomous driving," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 11, pp. 12878–12895, Nov. 2023.

[22] Y. Hu et al., "Planning-oriented autonomous driving," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Vancouver, Canada, 2023, pp. 17853–17862.

[23] H. Wang, P. Cai, Y. Sun, L. Wang, and M. Liu, "Learning interpretable end-to-end vision-based motion planning for autonomous driving with optical flow distillation," in *Proc. IEEE Int. Conf. Robot. Automat.*, Xi'an, China, 2021, pp. 13731–13737.

[24] S. Hu, L. Chen, P. Wu, H. Li, J. Yan, and D. Tao, "ST-P3: End-to-end vision-based autonomous driving via spatial-temporal feature learning," in *Proc. Eur. Conf. Comput. Vis.*, Tel Aviv, Israel, 2022, pp. 533–549.

[25] J. Philion and S. Fidler, "Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3D," in *Proc. Eur. Conf. Comput. Vis.*, Glasgow, U.K., 2020, pp. 194–210.

[26] C. Lu, M. J. G. van de Molengraft, and G. Dubbelman, "Monocular semantic occupancy grid mapping with convolutional variational encoder–decoder networks," *IEEE Robot. Automat. Lett.*, vol. 4, no. 2, pp. 445–452, Apr. 2019.

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

[28] T. Roddick and R. Cipolla, "Predicting semantic map representations from images using pyramid occupancy networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Seattle, WA, USA, 2020, pp. 11138–11147.

[29] Z. Li et al., "Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers," in *Proc. Eur. Conf. Comput. Vis.*, Tel Aviv, Israel, 2022, pp. 1–18.

[30] Y. Liu, T. Wang, X. Zhang, and J. Sun, "PETR: Position embedding transformation for multi-view 3D object detection," in *Proc. Eur. Conf. Comput. Vis.*, Tel Aviv, Israel, 2022, pp. 531–548.

[31] Z. Liu et al., "Vision-based uneven bev representation learning with polar rasterization and surface estimation," in *Proc. 6th Conf. Robot Learn.*, Atlanta, GA, USA, 2023, pp. 437–446.

[32] Y. Jiang et al., "Polarformer: Multi-camera 3D object detection with polar transformer," in *Proc. AAAI Conf. Artif. Intell.*, 2023, pp. 1042–1050.

[33] H. Yang, X. Bai, X. Zhu, and Y. Ma, "One training for multiple deployments: Polar-based adaptive BEV perception for autonomous driving," 2023, *arXiv:2304.00525*.

[34] R. Nahata, D. Omeiza, R. Howard, and L. Kunze, "Assessing and explaining collision risk in dynamic environments for autonomous driving safety," in *Proc. IEEE Int. Intell. Transp. Syst. Conf.*, Indianapolis, IN, USA, 2021, pp. 223–230.

[35] A. Bansal, J. Singh, M. Verucchi, M. Caccamo, and L. Sha, “Risk ranked recall: Collision safety metric for object detection systems in autonomous vehicles,” in *Proc. 10th Mediterranean Conf. Embedded Comput.*, MECO, Budva, Montenegro, 2021, pp. 1–4.

[36] M. Cui, S. Zhong, B. Li, X. Chen, and K. Huang, “Offloading autonomous driving services via edge computing,” *IEEE Internet Things J.*, vol. 7, no. 10, pp. 10535–10547, Oct. 2020.

[37] Y. Wang, P. H. Chan, and V. Donzella, “Semantic-aware video compression for automotive cameras,” *IEEE Trans. Intell. Veh.*, vol. 8, no. 6, pp. 3712–3722, Jun. 2023.

[38] R. Ke et al., “Lightweight edge intelligence empowered near-crash detection towards real-time vehicle event logging,” *IEEE Trans. Intell. Veh.*, vol. 8, no. 4, pp. 2737–2747, Apr. 2023.

[39] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, “CARLA: An open urban driving simulator,” in *Proc. 1st Annu. Conf. Robot Learn.*, Mountain View, CA, USA, 2017, pp. 1–16.

[40] J. Kim and J. Canny, “Interpretable learning for self-driving cars by visualizing causal attention,” in *Proc. IEEE Int. Conf. Comput. Vis.*, Venice, Italy, 2017, pp. 2942–2950.

[41] J. Chen, S. E. Li, and M. Tomizuka, “Interpretable end-to-end urban autonomous driving with latent deep reinforcement learning,” *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 6, pp. 5068–5078, Jun. 2022.

[42] Y. Xu et al., “Explainable object-induced action decision for autonomous vehicles,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Seattle, WA, USA, 2020, pp. 9523–9532.

[43] S. Teng, L. Chen, Y. Ai, Y. Zhou, Z. Xuanyuan, and X. Hu, “Hierarchical interpretable imitation learning for end-to-end autonomous driving,” *IEEE Trans. Intell. Veh.*, vol. 8, no. 1, pp. 673–683, Jan. 2023.

[44] K. Renz, K. Chitta, O.-B. Mercea, A. Koepke, Z. Akata, and A. Geiger, “PlanT: Explainable planning transformers via object-level representations,” in *Proc. 6th Annu. Conf. Robot Learn.*, 2022, pp. 1–12.

[45] Y. Feng, W. Hua, and Y. Sun, “NLE-DM: Natural-language explanations for decision making of autonomous driving based on semantic scene understanding,” *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 9, pp. 9780–9791, Sep. 2023.

[46] T. Hickling, N. Aouf, and P. Spencer, “Robust adversarial attacks detection based on explainable deep reinforcement learning for UAV guidance and planning,” *IEEE Trans. Intell. Veh.*, vol. 8, no. 10, pp. 4381–4394, Oct. 2023.

[47] B. Yang, R. Guo, M. Liang, S. Casas, and R. Urtasun, “Radarnet:Exploiting radar for robust perception of dynamic objects,” in *Proc. Eur. Conf. Comput. Vis.*, Glasgow, U.K., 2020, pp. 496–512.

[48] R. Van Kempen, B. Lampe, T. Woopen, and L. Eckstein, “A simulation-based end-to-end learning framework for evidential occupancy grid mapping,” in *Proc. IEEE Intell. Veh. Symp.*, Nagoya, Japan, 2021, pp. 934–939.

[49] Z. Zhang, A. Liniger, D. Dai, F. Yu, and L. Van Gool, “End-to-end urban driving by imitating a reinforcement learning coach,” in *Proc. IEEE Int. Conf. Comput. Vis.*, Montreal, Canada, 2021, pp. 15222–15232.

[50] H. Caesar et al., “nuScenes: A multimodal dataset for autonomous driving,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Seattle, WA, USA, 2020, pp. 11621–11631.

Yuchao Feng (Graduate Student Member, IEEE) received the bachelor’s degree from the Taiyuan University of Technology, China, in 2016, and the master’s degree from the University of Science and Technology of China, Hefei, China, in 2019. He is currently working toward the Ph.D. degree with the Department of Mechanical Engineering, The Hong Kong Polytechnic University, Hong Kong SAR, China. His research interests include autonomous driving, explainable artificial intelligence, computer vision, and deep learning.

Yuxiang Sun (Member, IEEE) received the bachelor’s degree from the Hefei University of Technology, Hefei, China, in 2009, the master’s degree from the University of Science and Technology of China, Hefei, China, in 2012, and the Ph.D. degree from The Chinese University of Hong Kong, Shatin, Hong Kong, in 2017. He is currently an Assistant Professor with the Department of Mechanical Engineering, City University of Hong Kong, Hong Kong. His research interests include robotics and AI, autonomous systems, mobile robots, autonomous driving, robotic perception and control, and autonomous navigation. He is also an Associate Editor for IEEE TRANSACTIONS ON INTELLIGENT VEHICLES, IEEE ROBOTICS AND AUTOMATION LETTERS, IEEE International Conference on Robotics and Automation, and IEEE/RSJ International Conference on Intelligent Robots and Systems.