A STUDY ON EXPLAINABLE END-TO-END AUTONOMOUS DRIVING

YUCHAO FENG

PhD

The Hong Kong Polytechnic University

2025

The Hong Kong Polytechnic University

Department of Mechanical Engineering

A Study on Explainable End-to-end Autonomous Driving

Yuchao Feng

A thesis submitted in partial fulfillment of the requirements for

the degree of Doctor of Philosophy

November 2024

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 acknowledgment has been made in the text.

Signature: _____

Name of Student: Yuchao Feng

Abstract

In recent years, end-to-end networks have emerged as a promising approach to achieving advanced autonomous driving in self-driving vehicles. Unlike modular pipelines, which divide autonomous driving into separate modules, this approach learns to drive by directly mapping raw sensory data to driving decisions (or control outputs). Compared to modular systems, end-to-end networks can avoid the accumulation of errors across different modules and are more scalable to complex scenarios. Despite these advantages, a major limitation of this approach is its lack of explainability. The outputs of end-to-end networks are generally not interpretable, making it difficult to understand why a specific input produces a given output. This limitation raises significant concerns about the safety and reliability of such systems, hindering their broader application and acceptance in real-world traffic environments.

Within this context, this study develops three methods to enhance the explainability of end-to-end autonomous driving networks. First, natural-language explanations are proposed to improve explainability. A novel explainable network, named the Natural-Language Explanation for Decision Making (NLE-DM), is designed to jointly

predict driving decisions and natural-language explanations. While natural-language explanations serve as an effective way to explain driving decisions, they often fall short of revealing the internal processes of the network. In contrast, visual explanations can provide insights into the network's inner workings. Therefore, to further enhance explainability, we propose combining natural-language and visual explanations as a multimodal approach. An explainable end-to-end network, named Multimodal Explainable Autonomous Driving (Multimodal-XAD), is designed to jointly predict driving decisions and multimodal environment descriptions. Finally, we revisit the concept of visual explanations and introduce an innovative Bird's-Eye-View (BEV) perception method, named PolarPoint-BEV. This method leverages a polar coordinate-based approach to better illustrate how the network perceives spatial relationships in the driving environment.

The three methods proposed in this study not only enhance the explainability of end-to-end networks but also address distinct key scientific problems in autonomous driving. For NLE-DM, the effect of natural-language explanations on driving decision prediction performance is investigated. The results demonstrate that the existence of natural-language explanations improves the accuracy of driving decision predictions. For Multimodal-XAD, the issue of error accumulation in downstream tasks of vision-based BEV perception is addressed by incorporating both context and local information before predicting driving decisions and environment descriptions. Experimental results show that combining context and local information enhances the

prediction performance of both tasks. For PolarPoint-BEV, the limitations of traditional BEV maps are identified and effectively addressed. Specifically, traditional BEV maps treat all regions equally, risking oversight of critical safety details, and use dense grids, resulting in high computational costs. To overcome these limitations, PolarPoint-BEV prioritizes regions closer to the ego vehicle, ensuring greater attention is given to critical areas while providing a more lightweight representation due to its sparse structure. To evaluate the impact of PolarPoint-BEV on explainability and driving performance, a multi-task end-to-end driving network, XPlan, is proposed to jointly predict control commands and polar point BEV maps.

Keywords: Autonomous Driving, end-to-end networks, Explainable AI (XAI), BEV Perception

Publications Arising from the Thesis

1. Yuchao Feng, Wei Hua, and Yuxiang Sun, "NLE-DM: Natural-Language Explanations for Decision Making of Autonomous Driving Based on Semantic Scene Understanding", *IEEE Transactions on Intelligent Transportation Systems (T-ITS)*, vol. 24, no. 9, pp. 9780-9791, 2023.

2. Yuchao Feng, Zhen Feng, Wei Hua and Yuxiang Sun, "Multimodal-XAD: Explainable Autonomous Driving Based on Multimodal Environment Descriptions", *IEEE Transactions on Intelligent Transportation Systems (T-ITS)*, vol. 25, no. 12, pp. 19469-19481, 2024.

3. Yuchao Feng, and Yuxiang Sun, "PolarPoint-BEV: Bird-eye-view Perception in Polar Points for Explainable End-to-end Autonomous Driving", *IEEE Transactions on Intelligent Vehicles (T-IV)*, early access, 2024.

Acknowledgments

I would like to express my gratitude to my supervisors, Dr. Sun Yuxiang and Dr. Chu Henry, for their unwavering support and guidance throughout my PhD journey. Dr. Sun has played a pivotal role in shaping the direction of my research. His insightful suggestions and expertise in our field have consistently guided me toward new ideas and deeper understanding. He was always willing to discuss research directions with me in detail, offering his profound knowledge and constructive feedback that helped to refine and focus my work. Moreover, Dr. Sun's patience and meticulous attention to detail were evident in every revision of my drafts. He took the time to provide thoughtful and precise critiques, helping me to develop not only as a researcher but also as a writer. His constant encouragement and trust in my abilities gave me the confidence to overcome challenges and strive for excellence in my research. I am deeply appreciative of his commitment to my academic growth, and I feel fortunate to have had the opportunity to work under his supervision. His mentorship has left a lasting impact, and I will carry the lessons I have learned from him into all my future endeavours.

I would also like to sincerely thank Dr. Chu for his invaluable support and guidance. His thoughtful feedback and advice, particularly during key stages of my Ph.D study, have been instrumental in helping me navigate challenges and stay on course. I am very grateful for his encouragement and contributions to my PhD journey.

I would like to thank my friends and labmates Xu Xin, Meng Shiyu, Ruan Jianyuan, Gao Shuang, Chen Keyu, Feng Zhen, Ma Weixin, Li Haotian, Tang Xuyang, Liu Zhuoyuan and Wang Yan, for their help along my educational journey. It is a rare fortune in life to have the opportunity to work together with these lovely friends.

I would like to express my appreciation to my girlfriend, for her unwavering support, understanding, and patience throughout my PhD journey. Her constant encouragement, both in moments of success and in times of difficulty, has been a tremendous source of motivation. She has been by my side through the highs and lows, offering her love and care, which have helped me stay focused and determined. I am incredibly grateful for her presence and all the sacrifices she has made to support me during this time.

Finally, I would like to extend my heartfelt gratitude to my parents, whose unconditional love, support, and encouragement have been a constant source of strength throughout my life. Their belief in my abilities and their unwavering confidence in my potential has motivated me to persevere through the challenges and uncertainties of this long process. They have always been there for me, providing not only

emotional support but also the foundation that allowed me to pursue my academic goals. Without their guidance and sacrifices, this achievement would not have been possible. I am forever grateful for everything they have done for me.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

In recent years, end-to-end autonomous driving has gained significant attention in both academia and industry, resulting in numerous advancements [1–12]. Unlike traditional modular pipelines, where the driving task is divided into separate modules such as perception, prediction, tracking, planning, and control, end-to-end networks directly map raw sensory data—such as camera images, radar, and LiDAR point clouds—into driving decisions or control commands (e.g., steering, throttle, and braking). The end-to-end design offers several key advantages: i) In modular pipelines, information loss and feature mismatches between modules can occur, and errors in one module can propagate throughout the pipeline, degrading the overall performance of the autonomous driving system [5]. In contrast, end-to-end networks generate control commands directly from raw sensor inputs through a unified neural network, reducing information transfer loss, feature mismatches, and error accumu-

Figure 1.1: The robotaxi (left) and robovan (right) from Tesla company. The figures come from the official Tesla website..

lation, while simplifying system design and implementation. ii) modular pipelines operate with independent objectives, which can result in conflicting goals and suboptimal coordination between modules. In contrast, end-to-end networks align all components towards a single, cohesive objective [11], minimizing the risk of suboptimal performance due to conflicting goals, and improving efficiency and reliability. iii) end-to-end networks optimize the driving task holistically, capturing complex, nonlinear relationships between sensor inputs and network outputs. This adaptability allows end-to-end networks to better handle diverse driving conditions and scenarios, which is critical for safe navigation in complex and dynamic environments. iv) end-to-end networks facilitate efficient training and rapid iteration, improving generalization capabilities and reducing reliance on manually annotated data for specific driving scenarios. Given these advantages, many companies—such as Tesla, Huawei, Momenta, Motional, Xiaopeng, NIO, Zeekr, and Xiaomi—have adopted or are planning to implement end-to-end autonomous driving approaches. Fig. 1.1 shows the picture of Tesla's robotaxi and robovan.

Despite the benefits, a significant drawback of end-to-end networks is their lack of explainability [13–15]. First, explainability needs to be defined in the context of autonomous driving. There are several terms in the field of Explainable AI (XAI)—such as interpretability and transparency—that are similar to explainability and are often misused [16]. Interpretability refers to a passive and intrinsic property of a network, reflecting the degree to which its internal mechanisms or decision-making processes can be intuitively understood by humans. Transparency is conceptually similar to interpretability. Specifically, a network is considered transparent if it can be understood by humans. In contrast, explainability can be viewed as an active property of a network, encompassing externally directed methods, tools, or post hoc techniques designed to clarify its operations or outputs. Explainability is often associated with the concept of explanations serving as a bridge or communication layer between humans and the network, facilitating a clearer understanding of its processes and decisions [17]. Therefore, in the context of autonomous driving, explainability is defined as the network's ability to provide accurate and understandable explanations for its outputs, which is crucial for building trust and ensuring safety. Explainable end-to-end autonomous driving refers to a network that takes in raw sensor data, processes it through a deep learning architecture to generate driving commands, and provides clear explanations that help humans understand its decisions."

There are several reasons why explainability is essential: i) Trust: When users can comprehend how an autonomous system makes decisions, they are more likely to trust

and adopt these systems. ii) Safety: Explainability enables developers and testers to better assess and verify system behavior, improving overall safety. iii) Regulation: As autonomous technology evolves, regulatory agencies require transparency to ensure compliance with traffic laws and safety standards. Explainability offers a method for validating adherence to these regulations. iv) Accident analysis: In the event of an accident, explainability helps determine the cause and assigns responsibility when necessary.

There are two primary reasons why explainability in end-to-end networks is limited. First, these systems are based on deep neural networks, such as Convolutional Neural Networks (CNNs) [18], Recurrent Neural Networks (RNNs) [19], and transformers [20], which are inherently complex and often function as "black boxes" that are difficult for humans to interpret. Second, unlike modular pipelines, end-to-end approaches lack intermediate outputs (such as object detection, semantic segmentation or occupancy prediction) that provide human-understandable insights into how the system processes data. End-to-end networks output only driving decisions or control values, further reducing explainability. This lack of explainability makes it challenging to identify and correct errors, understand system behavior, and build trust in its decisions. As autonomous driving is a safety-critical application, the absence of transparency in end-to-end networks limits their wider application. To address this issue, several techniques have been developed to explain the outputs of autonomous driving systems, falling into two main categories: visual explanations [21–28] and

natural-language explanations [29–37].

Visual explanations [21–28] refer to methods and techniques that clarify how autonomous driving networks perceive, interpret, and make decisions by using visual representations. These methods visualize the inner workings of deep neural networks so that humans can better comprehend them. For instance, attention heatmaps can highlight the areas of visual input that influenced the decision-making process, allowing humans to understand which parts of the scene contributed to specific driving actions. Additionally, latent high-dimensional features can be decoded into meaningful visual representations, such as semantic segmentation or depth estimation, to improve the explainability of deep neural networks. Natural-language explanations [29–37], on the other hand, are textual descriptions that explain the network's actions, decisions, and reasoning in human-understandable language. They help users and developers comprehend how the network interprets its environment, makes driving choices, and responds to situations. These natural-language explanations are often more straightforward and easier to interpret than visual explanations. Moreover, with the rapid development of Large Language Models (LLMs), there is potential for using LLMs to generate text-rich, human-readable explanations for autonomous driving decisions [38].

However, despite efforts to improve explainability, several limitations remain. First, current methods for generating natural-language explanations often lack precision and fail to provide objective or comprehensive explanations. This reduces

their overall clarity and practical utility. Second, most explanation approaches rely on a single modality—either visual or natural-language explanations—each with its strengths and weaknesses. Visual explanations can be difficult to interpret for end users, while natural-language explanations do not reveal the internal workings of the network. Combining both modalities could offer a more balanced and interpretable solution. Finally, traditional BEV perception methods, although widely used to explain how the network understands the surrounding environment, face limitations in focusing on safety-critical regions and tend to be computationally expensive, reducing their efficiency for real-time applications.

To address these challenges, we proposed three approaches [39–41] to improve the explainability of end-to-end autonomous driving networks. Firstly, to address the limitation of existing natural-language explanations, a deep neural network (named NLE-DM [39]) is introduced to jointly predicts decision-making actions and natural-language explanations based on semantic scene understanding. This network provides explanations in two forms: the reasons for driving actions and descriptions of the ego vehicle's surrounding environment. Comparative experiments demonstrate that the proposed approach outperforms state-of-the-art (SOTA) methods on both publicly available datasets [29] and our own datasets, significantly improving both the prediction performance of decision-making actions and explainability. In addition, to overcome the limitations of unimodal explanations, a network (named Multimodal-XAD [40]) is proposed that generates explanations in multimodal formats, including

BEV maps and natural-language environment descriptions. By incorporating both context information from BEV perception and local information from semantic perception, the proposed network improves the prediction performance of both driving actions and environment descriptions. This combination enhances the safety and explainability of the autonomous driving network. Finally, to address the limitations of the traditional BEV method, a novel BEV perception method (named PolarPoint-BEV [41]) is proposed to focus on areas near the ego vehicle, which are more critical for safety. Unlike traditional dense BEV maps, PolarPoint-BEV provides a sparse representation of traffic scenes, improving computational efficiency. This approach is integrated into an end-to-end network called XPlan, which jointly predicts control commands and polar point BEV maps. Experiments in the CARLA simulator [42] demonstrate that PolarPoint-BEV enhances both driving performance and explainability. In summary, the main contributions of this thesis are as follows:

1. We propose NLE-DM, a novel explainable decision-making network based on semantic scene understanding for autonomous driving. In this network, both the natural-language reasons for driving actions and the descriptions for ego-vehicle's surrounding environment are applied to explain the decision-making actions. The superiority of the proposed network over other SOTA networks is demonstrated on both the publicly available dataset and our released datasets.

2. We introduce Multimodal-XAD, a multimodal explainable network that leverages both BEV maps and natural-language environment descriptions to explain

driving actions. In this work, driving actions and natural-language environment descriptions are predicted based on both the context information from BEV perception and local information from semantic perception. The superiority of the proposed network over other SOTA networks is demonstrated on different datasets.

3. We develop PolarPoint-BEV, a lightweight BEV perception method that prioritizes critical regions near the ego vehicle. Furthermore, to evaluate the influence of the PolarPoint-BEV on the driving performance in an end-to-end network, 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 proposed PolarPoint-BEV can improve the driving performance and explainability of the proposed network.

4. The codes and datasets for the above works are publicly available, including NLE-DM[1], Multimodal-XAD[2] and PolarPoint-BEV[3].

The chapters of the thesis are organized as follows:

Chapter 1 shows the introduction and background of the research.

Chapter 2 gives a literature review of end-to-end autonomous driving networks, explainable AI techniques, explanation methods in autonomous driving, BEV per-

[1]NLE-DM: `https://github.com/lab-sun/NLE-DM`

[2]Multimodal-XAD: `https://github.com/lab-sun/Multimodal-XAD`

[3]PolarPoint-BEV: `https://github.com/lab-sun/PolarPoint-BEV`

ception methods and existing datasets of autonomous driving.

Chapter 3 introduces the proposed NLE-DM. The network architecture and training details are first presented. Then, comparative results are provided, demonstrating the superiority of NLE-DM. Finally, the relationship between decision-making actions and natural-language explanations is investigated in the ablation study.

Chapter 4 introduces the proposed Multimodal-XAD. The network architecture and training details are first presented. Then, comparative results are provided, demonstrating the superiority of Multimodal-XAD. Finally, the impact of combining context and local information on the prediction performance of driving actions and multimodal environment descriptions is investigated in the ablation study.

Chapter 5 introduces the proposed PolarPoint-BEV. The details of polar point BEV map, network architecture and training details are first presented. Then, comparative results are provided, demonstrating the superiority of PolarPoint-BEV. Finally, the prediction performance of polar point BEV maps with different configurations is investigated in the ablation study.

Chapter 6 presents the conclusions and suggestions for future research.

Chapter 2

Literature Review

2.1 End-to-end Autonomous Driving

In recent years, end-to-end autonomous driving networks have garnered significant attention in the field of self-driving technology. Unlike traditional modular pipelines, these end-to-end networks eliminate the problem of error propagation between different modules, making them more robust and scalable when handling the complexity of real-world driving scenarios. Their capacity to adapt to diverse and intricate environments makes them particularly well-suited for deployment in real-world autonomous driving applications.

Numerous end-to-end autonomous driving networks have been developed, demonstrating substantial advancements in this domain [1–12]. For instance, Prakash *et*

al. [4] introduced an innovative end-to-end framework that leverages transformer attention mechanisms to effectively combine image and LiDAR data. Chen *et al.* [6] introduced a method known as Learning from All Vehicles (LAV), an end-to-end network designed to derive driving policies by learning from the collective experiences of nearby vehicles. This network processes multi-modal sensory data and generates predictive trajectories for every vehicle it detects. Another notable example is the Trajectory-guided Control Prediction (TCP) network developed by Wu *et al.* [7]. TCP integrates two key components: a trajectory branch and a multi-step control branch, offering enhanced prediction and control capabilities by fusing information from these branches. Chitta *et al.* [10] proposed a transformer-based model to enhance the ability to process complex sensory inputs in an integrated manner. This architecture demonstrates the growing importance of advanced attention mechanisms in improving decision-making accuracy in autonomous systems. Hu *et al.* [11] introduced a planning-centric approach called Unified Autonomous Driving (UniAD), which features a novel query design. This design acts as a cohesive interface, allowing various components of the system to communicate effectively. Through this query-based mechanism, knowledge from intermediate tasks can be shared and utilized to refine the planning process, resulting in more effective driving strategies.

2.2 Explainable AI

Explainable Artificial Intelligence (XAI) seeks to equip humans with the ability to interpret and build appropriate levels of trust in the decisions and operations of AI systems [16, 43]. To date, a wide array of XAI methods have been developed and implemented across diverse models, serving various tasks. These approaches include transparent models [44, 45], local explanations [46, 47], simplification-based explanations [48], feature relevance-based interpretations [49], visual explanations [50–52], and architectural modifications [53–57], etc.

A transparent model, by definition, is inherently explainable without the need for additional interpretation techniques. Depending on the level of explainability, it can generally be classified into three types: simulatable models, decomposable models, and algorithmically transparent models [58]. Simulatable models are those simple enough for a human to fully understand, decomposable models allow for explanation by breaking down their components, and algorithmically transparent models enable users to grasp how the algorithm operates in practice. In local explanation techniques, the solution space of the model is segmented, allowing explanations to be generated for smaller, less complex subspaces. This process helps provide insights into the behavior of a model for specific predictions [16]. Explanation by simplification typically involves approximating complex models with simpler, more interpretable ones. A common approach here is the use of Local Interpretable Model-Agnostic Explanations (LIME) and its variations [59], which generate local linear models around the

predictions to help explain them. The core concept behind LIME is to focus on localized sections of the predictions, making them more understandable by creating a linear approximation for a specific instance. Feature relevance explanation methods go deeper by analyzing the internal mechanics of a model. They assign relevance scores to the input features based on their contribution to the output. This assigns weights to each input, reflecting how significant each feature is for predicting the target variable [16]. Visual explanation techniques are often used in conjunction with other XAI methods to provide a graphical representation of the behavior of models. These visual tools help users more intuitively grasp how a model operates by displaying which parts of the input data were most influential in the decision-making process. Architecture modification focuses on altering the structure of neural networks to enhance their explainability. This can be achieved using various techniques, such as modifying layers [53], combining models [54], using attention mechanisms [55, 56], or modifying the loss function [57]. For example, in [55], a global average pooling layer was added between the last convolutional layer and the fully connected layer. This architectural change enabled the creation of an attention map that highlights regions in an image associated with a specific object class, thus making the decision process more transparent.

2.3 Explanations in Autonomous Driving

Given the critical role of explainability in autonomous driving networks, a large number of research studies have emerged in recent years focusing on explainable autonomous driving methods. As mentioned earlier, there are two primary categories of explanation techniques employed in this area: visual explanations [21–28] and natural-language explanations [29–37]. Both approaches are applied to enhance the transparency of autonomous driving networks and increase user trust.

Visual explanations are typically generated by revealing the internal processes of the network through visual representations. For instance, Kim *et al.* [22] utilized visual attention heatmaps to highlight areas of an image that significantly influence driving decisions, allowing users to understand which regions of the input data are pivotal in the reasoning of the network. Similarly, Chen *et al.* [23] proposed a method within deep reinforcement learning for end-to-end autonomous driving, providing interpretability by generating BEV semantic masks that visually describe the environment. Another noteworthy approach is presented by Teng *et al.* [25], who developed the Hierarchical Interpretable Imitation Learning (HIIL) model designed for complex driving scenarios. In HIIL, traditional semantic BEV maps are employed to explain the surrounding environment and identify failure cases involving the ego vehicle. Renz *et al.* [26] introduced PlanT, an explainable planning transformer. By extracting and visualizing attention weights, PlanT identifies objects that are crucial for the agent's decision-making process, thereby improving transparency and explainability.

In addition to visual-based techniques, natural-language explanations are gaining attraction within explainable autonomous driving networks [29–37]. One advantage of natural-language explanations is their intuitive clarity, making them more accessible to end users. Moreover, natural-language explanations hold the potential to be integrated with Large Language Models (LLMs), leveraging their commonsense reasoning abilities to enhance both interpretability and the generalization capabilities of autonomous driving systems [38]. This combination could enable models to provide more comprehensive and understandable explanations for their decisions. Several notable contributions have been made in the area of natural-language explanations. Xu *et al.* [29] introduced a multi-task model that predicts driving actions and generates corresponding natural-language explanations by combining reasoning about action-inducing objects and global scene understanding. Meanwhile, Dong *et al.* [31] developed an explainable end-to-end model based on the Transformer architecture, which maps visual inputs to driving actions while producing natural-language justifications. Additionally, Ben-Younes *et al.* [36] proposed the BEhavior Explanation with Fusion (BEEF) model, an explainable autonomous driving framework. The key innovation in BEEF is the fusion of multi-level features to simultaneously predict both vehicle trajectories and natural-language explanations, offering a more holistic understanding of the decision-making process in autonomous driving.

2.4 BEV Perception

The BEV maps have long been an essential tool in autonomous driving, primarily for enhancing the explainability of system behaviors. Moreover, the BEV maps serve as a foundation for numerous downstream tasks, which depend heavily on precise BEV perception. BEV perception approaches can be generally divided into three categories: point cloud-based methods [60–65], visual image-based methods [66–86], and multimodal methods [87–91].

Point cloud-based methods rely on radar or LiDAR sensors to generate BEV maps. Sless *et al.* [60], for example, introduced a learnable inverse sensor model that transforms sparse and noisy radar data into binary occupancy grid maps using a data-driven approach. Yang *et al.* [61] proposed a method that improves the perception of dynamic objects by integrating radar and LiDAR data, enhancing robustness in challenging autonomous driving scenarios. Similarly, Kempen *et al.* [64] developed a novel network that focuses on generating occupancy grid maps using LiDAR point clouds, thereby enabling accurate spatial understanding for autonomous navigation.

Multimodal methods, in contrast, integrate multiple sensor inputs—such as camera, LiDAR, and radar data—into a unified BEV perception system. Liang *et al.* [88] presented a straightforward yet effective network that encodes raw data from both cameras and LiDAR sensors into the same BEV space, ensuring that the system leverages complementary features from each sensor type. Building on this, Liu *et*

al. [89] proposed a multi-task, multi-sensor fusion framework that unifies multimodal features in a shared BEV representation space, preserving both geometric and semantic information. Man *et al.* [90] introduced a novel BEV learning framework capable of unifying a variety of sensors—including cameras, LiDAR, and radar—under direct BEV supervision in an end-to-end manner, increasing the efficiency of sensor fusion.

Vision-based methods, which use RGB images from visual cameras, require transforming perspective view (PV) inputs into BEV format. While these methods tend to be more cost-effective compared to point cloud-based and multimodal approaches, they often suffer from lower semantic perception performance [92]. Kim *et al.* [66], for instance, employed inverse perspective mapping to estimate distances from monocular images by assuming that all image pixels are on the ground plane, but this assumption reduces height discrimination accuracy. To address this limitation, Philion *et al.* [70] introduced a network that can infer BEV representations from multiple camera inputs, offering a more flexible and accurate solution for complex driving environments. Pan *et al.* [75] developed the View Parsing Network (VPN) for cross-view semantic segmentation, parsing first-person view observations into BEV semantic maps. This was further refined by Zhou *et al.* [78], who introduced Cross-View Transformers (CVT), an efficient attention-based model designed to perform map-view semantic segmentation using multiple camera inputs. Liu *et al.* [81] proposed the Position Embedding Transformation (PETR) for 3D object detection, which encodes 3D positional information directly into image features, generating position-aware representations that

improve detection performance.

Most visual image-based methods rasterize the BEV space along Cartesian coordinates to create uniformly distributed rectangular maps. However, to mitigate the foreshortening effect inherent in camera imaging, some vision-based approaches [84–86] apply the Polar coordinate system when rasterizing BEV spaces. For example, Liu *et al.* [84] introduced a method that rasterizes BEV spaces angularly and radially, then rearranges and maps the relationships between polar grids to form an array-like representation. In a similar way, Jiang *et al.* [85] proposed PolarFormer, a 3D object detection framework in BEV that features a cross-attention-based Polar detection head designed to handle the irregular structure of polar grids.

2.5 Datasets in Autonomous Driving

Autonomous driving heavily relies on datasets to develop, test, and validate algorithms before deploying them on public roads. A variety of autonomous driving datasets have been created so far [29, 93–98], containing both vision-based data and information from multiple sensors, such as GPS, radar, LiDAR, and IMU data. One of the most well-known datasets, KITTI [93], focuses on tasks like stereo vision, optical flow, visual odometry, and 3D object detection. CityScapes [94], on the other hand, is designed for evaluating semantic urban scene understanding, with high-quality annotations provided for 5,000 frames and an additional 20,000 weakly annotated

frames. Apolloscape [95] offers a rich dataset with 144,000 frames collected from four regions in China under varying times of day and weather conditions. nuScenes [96] contains a vast amount of data, including 1.4 million camera images, 390,000 LiDAR sweeps, 1.4 million radar sweeps, and 1.4 million object bounding boxes across 1,000 scenes. BDD100K [97], one of the largest video datasets for autonomous driving, comprises 100,000 videos and supports a range of tasks, including object detection, semantic segmentation, and lane detection.

Despite their size and diversity, these datasets are still limited in scope, particularly when it comes to corner cases and long-tail scenarios that may not be adequately represented. To address these gaps, synthetic datasets generated via simulators [42, 99–102] and world models [103, 104] have emerged as low-cost alternatives for producing specific scenarios. For instance, The CARLA simulator [42] is an open-source, high-fidelity autonomous driving simulator that uses photorealistic urban and rural environments to train, test, and validate autonomous driving networks in diverse driving conditions. DriveDreamer [103] can generate high-quality driving videos depicting realistic traffic scenes, while also simulating reasonable driving policies. However, whether the datasets are derived from real-world driving or generated synthetically, they often fall short in their applicability to training explainable autonomous driving networks.

To address the need for developing and testing explainable autonomous driving networks, several explainable datasets have been introduced [29, 30, 105, 106]. For ex-

ample, the BDD-X dataset [30] includes annotations for driving actions (e.g., "the car slows down") along with corresponding natural-language explanations (e.g., "because it is about to merge onto a busy highway"). Similarly, the BDD-OIA dataset [29] contains approximately 23,000 front-view images sourced from BDD100K, where each image is annotated with driving actions and their respective natural-language justifications. Although these explainable datasets contribute significantly to the field, they are primarily focused on natural-language explanations. To further enhance the level of explainability, a combination of both visual and natural-language explanations could offer a more comprehensive understanding of autonomous driving network outputs.

Chapter 3

NLE-DM: Natural-Language Explanations for Decision Making of Autonomous Driving

3.1 Introduction

Autonomous driving presents a promising solution to reducing road accidents and enhancing traffic safety, which has sparked considerable interest in both robotics and computer vision communities in recent years. According to a report from the American National Highway Traffic Safety Administration (NHTSA), around 94% of traffic accidents are caused by human-related factors such as distractions and violations of traffic regulations [107]. By removing the potential for human error, autonomous

driving systems have the potential to drastically improve driving safety. Despite extensive research progress over the past decade, the technology is still not reach this expectation. Conventional methods have not yet achieved significant breakthroughs. On the other hand, the rise of deep learning has led to groundbreaking advancements across various research fields. By leveraging deep learning, the field of autonomous driving has advanced considerably. Deep neural networks have been successfully applied to a broad range of autonomous driving tasks, such as object detection [108], semantic scene interpretation [109,110], localization [111], motion planning [112,113], trajectory forecasting [114,115], vehicle control [116,117], and decision making [118].

The decision-making process in autonomous driving entails choosing a specific control action (e.g., driving straight, turning left, turning right or stop to avoid collision) based on both the state of the ego-vehicle and its surrounding context [119]. As a fundamental component of autonomous driving, decision making plays a crucial role in ensuring the vehicle's safe operation. Various decision-making strategies have been introduced, which can broadly be divided into traditional techniques and deep learning-based approaches. Traditional methods include rule-based, optimization-driven, and probabilistic frameworks. However, due to the complexity and dynamic nature of real-world traffic scenarios, classical approaches often struggle to perform well in such environments. In contrast, deep learning-based methods have shown more robust results in managing these challenges [119,120].

Although deep learning-based techniques have demonstrated impressive perfor-

mance, they suffer from a critical limitation: their lack of explainability. One of the main reasons is that deep neural networks function as "black boxes", making it difficult to decipher how or why a particular driving decision was made based on a given input. This opacity poses a challenge when deploying such systems in real-world applications, as the unpredictability and complexity of real environments make it unsafe to blindly trust the model's outputs without understanding the rationale behind its actions. This is particularly true in unforeseen or highly variable environments, where the model's decisions may become unreliable without a clear explanation.

To tackle this problem, we propose a novel deep neural network that not only predicts decision-making actions but also provides natural-language explanations based on semantic scene understanding. Specifically, two types of explanations are generated: reasons for the selected driving actions and descriptions of the surrounding environment relevant to the ego-vehicle. To train and test the proposed network, a large-scale dataset have annotated by consisting of $10,000$ images from the BDD-OIA dataset [29], labeled with 4 distinct driving actions and 6 environmental descriptions. In addition, to further assess the network's ability to generalize to new driving scene, a subset of $1,500$ frames from the nuScenes dataset [96], have been selected and labeled with 4 driving actions and corresponding natural-language explanations. Experimental results from both publicly available datasets [29] and our own datasets demonstrate the effectiveness of the proposed model. The proposed network significantly improves both the accuracy of predictions and the explainability.

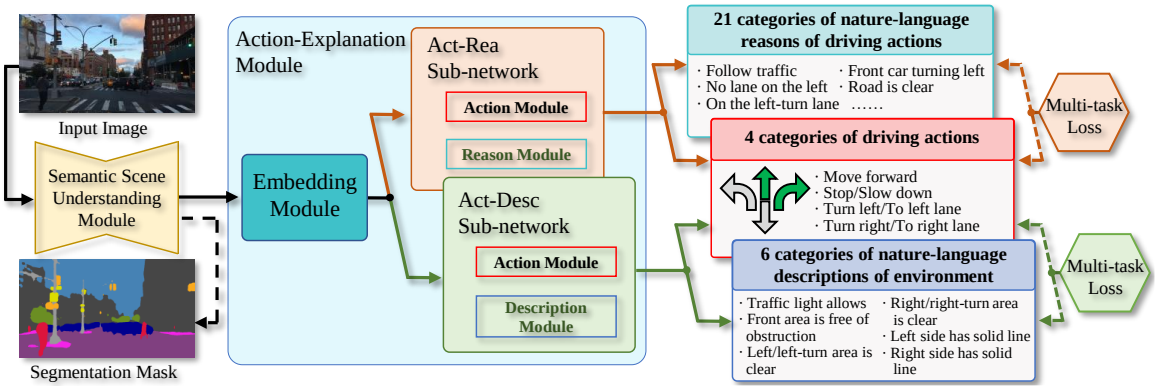


Figure 3.1: The architecture of the proposed NLE-DM network. The action-explanation module takes the feature maps from the semantic scene understanding module as input, and outputs the decision-making actions and the corresponding natural-language explanations.

3.2 Methodology

3.2.1 The Network Architecture

Fig. 3.1 shows the structure of the proposed NLE-DM model, which is composed of two main modules: the semantic scene understanding module and the action-explanation module. The network processes front-view images as input and generates both driving decisions and corresponding natural-language explanations.

Let the frame set be defined as $\mathbf{X} = \{X_1, \ldots, X_i, \ldots, X_N\}$, where each frame $X_i \in \mathbb{R}^{\mathrm{h}\times\mathrm{w}\times\mathrm{c}}$ represents an image with height h, width w, and c channels. The total number of frames is represented by N. The driving decisions are grouped into four categories: "move forward", "stop/slow down", "turn left/change to left lane" and "turn right/change to right lane". For the explanations, two kinds of natural-

language descriptions are provided: reasons for driving actions and descriptions of the surrounding environment. Specifically, the reasons for actions are divided into 21 distinct categories, while environmental conditions are categorized into 6 groups. The action reasons are adapted from the approach introduced in [29], and the complete list of 21 categories is available in Table 3.2. The environmental descriptions include categories of: "traffic light allows", "front area is free of obstruction", "left/left-turn area is clear", "right/right-turn area is clear", "left side has solid line" and "right side has solid line" as detailed in Table 3.5. The overall behavior of the NLE-DM network can be expressed through the following representations:

$$\begin{gathered} \mathbf{X} \rightarrow (A, R) \in \{0,1\}^4 \times \{0,1\}^{21}, \\ \text{or} \\ \mathbf{X} \rightarrow (A, D) \in \{0,1\}^4 \times \{0,1\}^6, \end{gathered} \tag{3.1}$$

where A represents driving actions, R refers to the reasons for those actions, and D is the descriptions of the surrounding environment.

The semantic scene understanding (S-S) module is designed based on the DeepLabv3 semantic segmentation network [121]. More technical details regarding DeepLabv3 can be found in [121]. This module extracts a semantic feature map $M_i \in \mathbb{R}^{\mathrm{h}\times\mathrm{w}\times\mathrm{n}}$ from each input image X_i, where h and w are the spatial dimensions, and n is the number of semantic classes. The process is formalized as:

$$\text{S-S Module: } X_i \rightarrow M_i \in \mathbb{R}^{\mathrm{h}\times\mathrm{w}\times\mathrm{n}}, \quad 1 \leq i \leq N, \tag{3.2}$$

where N is the number of frames in the frame set $\mathbf{X}$.

The action-explanation (A-E) module is composed of three primary components: the embedding module, the Act-Rea sub-network, and the Act-Desc sub-network. First, the semantic feature maps generated by the S-S module are passed into the embedding module, where the resolution and dimensionality are reduced. The resulting embedding feature map has a size of $64 \times 18 \times 32$, where 64 is the number of channels, and 18×32 represents the downscaled resolution. For reference, the original resolution of the input image is 720×1280 with 3 channels (RGB format). Once the feature map is embedded, it is flattened and passed into the Act-Rea and Act-Desc sub-networks, which are responsible for predicting the driving decisions and producing the corresponding natural language explanations. This process is represented as $M_i \rightarrow V_i$, where V_i is the resulting flattened feature vector.

Both the reasons for the driving decisions and the environmental descriptions aim to explain the driving actions, and thus, the A-E module is designed to output both actions and their respective natural-language explanations. The Act-Rea sub-network includes two branches: one that predicts driving actions and another that generates natural-language reasons for those actions. The Act-Rea sub-network's functionality is expressed as follows:

$$\text{Act-Rea: } V_i \rightarrow (A, R) \in \{0,1\}^4 \times \{0,1\}^{21}, \quad 1 \leq i \leq N, \tag{3.3}$$

where N is the total number of frames in the sequence $\mathbf{X}$. Similarly, the Act-Desc sub-network includes both an action-prediction branch and an environmental description branch, enabling it to provide natural-language descriptions of the surrounding

environment in addition to the predicted driving actions. The Act-Desc sub-network process is described as:

$$\text{Act-Desc: } V_i \rightarrow (A, D) \in \{0,1\}^4 \times \{0,1\}^6, \quad 1 \leq i \leq N, \tag{3.4}$$

where N refers to the total number of frames in the set $\mathbf{X}$.

3.2.2 Training Details

We begin by pre-training the S-S module on the BDD10K dataset, which is a subset of the larger BDD100K dataset [97]. This pre-training process equips the S-S module with the ability to perform pixel-wise semantic scene understanding. Subsequently, the entire network is trained using the pre-trained weights. For the Act-Rea sub-network, the training and evaluation are conducted on the BDD-OIA dataset [29], while the Act-Desc sub-network is trained on a newly created dataset, named *BDD Actions and Descriptions* (BDD-AD). The BDD-AD dataset comprises images selected from BDD-OIA, annotated with driving actions and natural-language descriptions of the ego-vehicle's surrounding environment. Further details on the BDD-AD dataset can be found in the following section. To further assess the performance and generalization capability of the proposed network, both the Act-Rea and Act-Desc sub-networks are evaluated on $1,500$ frames from the nuScenes dataset [96], annotated with driving actions and associated natural-language explanations, which include both reasons and descriptions.

The stochastic gradient descent (SGD) optimizer is utilized with an initial learning rate of 0.001, momentum set to 0.9, and a weight decay of 1×10^{-4}. It should be noted that the BDD10K and BDD-OIA datasets are not fully overlapping. Therefore, to ensure adaptability to new scenes, the weights of the S-S module, though pre-trained on BDD10K, are allowed to be fine-tuned during training with BDD-OIA or BDD-AD. The proposed network is optimized using a multi-task loss function, which is defined as follows:

$$\mathcal{L}_{total} = \mathcal{L}_{act} + \lambda \mathcal{L}_{rea}, \tag{3.5}$$

$$\mathcal{L}_{total} = \mathcal{L}_{act} + \lambda \mathcal{L}_{desc}, \tag{3.6}$$

where $\mathcal{L}_{total}$ represents the total loss, $\mathcal{L}_{act}$, $\mathcal{L}_{rea}$, and $\mathcal{L}_{desc}$ correspond to the binary cross-entropy losses for action, reason, and description predictions, respectively. The loss function in equation (3.5) is applied to the Act-Rea sub-network, and equation (3.6) is applied to the Act-Desc sub-network. The parameter λ controls the balance between the importance of the decision-making actions and the accompanying natural-language explanations.

In the case of the Act-Rea sub-network, we adopt the 4 categories of driving actions and the 21 categories of natural-language reasons utilized in [29]. It is important to note that, in [29], the 21 categories of natural-language reasons are termed as "explanations". The two loss terms, $\mathcal{L}_{act}$ and $\mathcal{L}_{rea}$, are computed as follows: $\mathcal{L}_{act} = \sum_{i=1}^{4} \mathcal{L}[\hat{A}_i, A_i]$ and $\mathcal{L}_{rea} = \sum_{j=1}^{21} \mathcal{L}[\hat{R}_j, R_j]$, where $\hat{A}_i$ and A_i are the predicted and ground-truth driving actions, respectively, while $\hat{R}_j$ and R_j denote the

predicted and true reasons. For the Act-Desc sub-network, the 4 driving actions are also predicted, but the network predicts 6 categories of natural-language descriptions instead. Therefore, in addition to $\mathcal{L}_{act}$, the description loss $\mathcal{L}_{desc}$ is calculated as: $\mathcal{L}_{desc} = \sum_{k=1}^{6} \mathcal{L}[\hat{D}_k, D_k]$, where $\hat{D}_k$ and D_k represent the predicted and ground-truth descriptions, respectively.

3.3 Experimental Results and Discussions

3.3.1 Evaluation Metrics

To quantitatively assess the prediction performance of the decision-making actions, the reasons behind those actions, and the descriptions of the surrounding environment, the standard F1 score is utilized as the evaluation metric. Two variants of the F1 score are employed: $\mathrm{F1}_{\mathrm{oval}}$ (overall F1 score) and $\mathrm{F1}_{\mathrm{m}}$ (mean F1 score). The overall F1 score, $\mathrm{F1}_{\mathrm{oval}}$, is computed as follows:

$$\mathrm{F1}_{\mathrm{oval}}^{\mathrm{act}} = \frac{1}{N} \sum_{i=1}^{N} \mathrm{F1}(\hat{A}_i, A_i), \tag{3.7}$$

$$\mathrm{F1}_{\mathrm{oval}}^{\mathrm{rea}} = \frac{1}{M} \sum_{j=1}^{M} \mathrm{F1}(\hat{R}_j, R_j), \tag{3.8}$$

$$\mathrm{F1}_{\mathrm{oval}}^{\mathrm{desc}} = \frac{1}{Q} \sum_{k=1}^{Q} \mathrm{F1}(\hat{D}_k, D_k), \tag{3.9}$$

In the equations above, $\mathrm{F1}_{\mathrm{oval}}^{\mathrm{act}}$, $\mathrm{F1}_{\mathrm{oval}}^{\mathrm{rea}}$, and $\mathrm{F1}_{\mathrm{oval}}^{\mathrm{desc}}$ represent the overall F1 scores for action predictions, reason predictions, and description predictions, respectively. The

variables N, M, and Q denote the total number of action predictions, reason predictions, and description predictions, respectively.

Given the imbalance in the BDD-OIA and BDD-AD datasets, the mean F1 score is also computed, $\mathrm{F1_m}$, to mitigate the impact of class imbalance. The mean F1 score is defined as follows:

$$\mathrm{F1}_{\mathrm{m}}^{\mathrm{act}} = \frac{1}{4}(\mathrm{F1_F} + \mathrm{F1_S} + \mathrm{F1_L} + \mathrm{F1_R}), \tag{3.10}$$

$$\mathrm{F1}_{\mathrm{m}}^{\mathrm{rea}} = \frac{1}{21}\sum_{j=1}^{21} \mathrm{F1}_{\mathrm{j}}^{\mathrm{rea}}, \tag{3.11}$$

$$\mathrm{F1}_{\mathrm{m}}^{\mathrm{desc}} = \frac{1}{6}\sum_{k=1}^{6} \mathrm{F1}_{\mathrm{k}}^{\mathrm{desc}}, \tag{3.12}$$

In these equations, $\mathrm{F1}_{\mathrm{m}}^{\mathrm{act}}$, $\mathrm{F1}_{\mathrm{m}}^{\mathrm{rea}}$, and $\mathrm{F1}_{\mathrm{m}}^{\mathrm{desc}}$ are the mean F1 scores for action, reason, and description predictions, respectively. For action prediction, $\mathrm{F1_F}$, $\mathrm{F1_S}$, $\mathrm{F1_L}$, and $\mathrm{F1_R}$ represent the F1 scores for predicting "move forward", "stop/slow down", "turn left/change to left lane" and "turn right/change to right lane" respectively. Similarly, $\mathrm{F1}_{\mathrm{j}}^{\mathrm{rea}}$ and $\mathrm{F1}_{\mathrm{k}}^{\mathrm{desc}}$ refer to the F1 scores for each reason and description category.

3.3.2 Jointly Predicting Actions and Reasons

We first present and analyze the experimental results for the Act-Rea sub-network, which jointly predicts driving actions and their corresponding natural-language reasons. Tab. 3.1 compares the quantitative performance of Act-Rea sub-network against several other models, including those presented in [29, 32–34, 122]. As men-

Table 3.1: Comparative results of the prediction performance for different networks. The F/S/L/R refer to "move forward", "stop/slow down", "turn left/change to left lane" and "turn right/change to right lane" respectively. The best and second-best results are highlighted in bold font and italic font.

Methods	F	S	L	R	$F1_{m}^{act}$	$F1_{oval}^{act}$	$F1_{m}^{rea}$	$F1_{oval}^{rea}$
Act-Rea ($\lambda = 1.0$)	0.827	0.760	0.651	0.653	**0.723**	*0.733*	0.312	0.517
Act-Rea ($\lambda = 2.0$)	0.813	0.768	0.649	0.643	*0.718*	0.728	**0.350**	**0.546**
OIA [29]	0.829	0.781	0.630	0.634	*0.718*	**0.734**	0.208	0.422
Local selector [122]	0.810	0.762	0.600	0.624	0.699	0.711	0.196	0.406
C-SENN [33]	0.772	0.744	0.469	0.486	0.618	–	–	–
CBM [34]	0.795	0.732	0.483	0.431	0.610	0.661	0.292	0.412
CBM-AUC [32]	0.803	0.751	0.551	0.525	0.658	0.704	*0.342*	*0.522*

tioned earlier, the parameter λ in the loss function (3.5) controls the relative importance between action predictions and reason explanations. The effect of λ on performance is further explored in the ablation study. The OIA network [29] combines object reasoning with global scene analysis to emphasize objects that induce actions. Wang *et al.* [122], as modified by Xu *et al.* [29], proposed a local selector network that predicts actions and reasons, serving as a variant of OIA that focuses only on local object reasoning. The contrastive self-explaining neural network (C-SENN) [33] integrates contrastive and concept learning to enhance the accuracy and explainability of driving actions. Meanwhile, the concept bottleneck model (CBM) [34], developed

Figure 3.2: Sample comparative results of action and reason predictions for the OIA network and the proposed network. The GT and Pre refer to the ground truth and prediction of reason predictions.

by Kon *et al.* [34] and later modified by Sawada *et al.* [32], jointly predicts actions and reasons. An extension of CBM, the CBM with additional unsupervised concepts (CBM-AUC) [32], incorporates both supervised and unsupervised concepts to further boost performance.

As displayed in Tab. 3.1, Act-Rea sub-network (with $\lambda = 1.0$ and $\lambda = 2.0$) achieves action prediction performance comparable to that of the OIA network, outperforming the remaining models. In terms of reason prediction, the proposed network ($\lambda = 1.0$ and $\lambda = 2.0$) and CBM-AUC demonstrate similar results, both surpassing the other methods. These comparative results clearly highlight the superior perfor-

mance of Act-Rea sub-network for predicting both driving actions and the associated reasons.

Additionally, the qualitative analysis further validates the superiority of the proposed model, as illustrated in Fig. 3.2. To ensure fairness, identical examples from the OIA paper [29] have been selected. In Fig. 3.2, although the predicted driving actions of the Act-Rea sub-network ($\lambda = 1.0$) and the OIA network are identical, the accuracy of reason predictions differs. For the OIA network, the true positive ratios for reason prediction in the four examples are 100%, 33.3%, 50%, and 66.7%, respectively. In contrast, Act-Rea sub-network achieves 100%, 50%, 75%, and 100% for the same examples.

We hypothesize that the improved performance of the proposed network over OIA can be attributed to the following reasons:

- The proposed model leverages atrous spatial pyramid pooling (ASPP) [121] to capture multi-scale features, whereas the OIA network only integrates global and local features. In complex driving environments, where objects vary in size, the absence of multi-scale perception in OIA can lead to errors in identifying objects across scales, resulting in incorrect reason predictions.

- The OIA network employs object detection (Faster R-CNN) to generate feature maps and identify action-inducing objects. In contrast, the proposed model utilizes semantic segmentation (DeepLabv3), which provides finer, pixel-level

Table 3.2: The prediction performance of the natural-language reasons.

Action Category	Reason Category	F1 Score
Move forward	follow traffic	0.645
	the road is clear	0.447
	the traffic light is green	0.528
Stop/Slow down	obstacle: car	0.599
	obstacle: person/pedestrian	0.440
	obstacle: rider	0.000
	obstacle: others	0.000
	the traffic light	0.768
	the traffic sign	0.000
Turn left	front car turning left	0.000
	on the left-turn lane	0.000
	traffic light allows	0.000
Turn right	front car turning right	0.000
	on the right-turn lane	0.053
	traffic light allows	0.000
Can't change to left lane	obstacles on the left lane	0.585
	no lane on the left	0.472
	solid line on the left	0.474
Can't change to right lane	obstacles on the right lane	0.624
	no lane on the right	0.474
	solid line on the right	0.442

details from the scene. Semantic segmentation offers a more detailed understanding of the environment, as it labels objects at the pixel level, whereas object detection works at a coarser bounding-box level.

We also examine the prediction performance of the proposed model with respect to natural-language reasons. Tab. 3.2 reports the F1 scores for each reason category predicted by the Act-Rea sub-network ($\lambda = 1.0$). Unlike the action predictions, the reason prediction results show some bias. For certain reason categories, the F1 scores are zero, indicating the network's inability to predict those reasons. We believe the poor performance in some cases can be explained by the following factors:

- The network is pre-trained on the BDD10K dataset for semantic segmentation, and poor segmentation performance for specific object classes could lead to unsatisfactory reason predictions. For instance, the intersection over union (IoU) for the "rider" class is only 13.4% (see Tab. 3.3), which may prevent the A-E module from recognizing the "rider" correctly, and thus fail to associate the reason "obstacle: rider" with the action "Stop/Slow down." Similarly, the IoU for "obstacle: others" is poor, as evidenced by IoUs of 0.0% for trains, 36.3% for motorcycles, and 36.2% for bicycles.

- Some reasons are inherently abstract or ambiguous, which can lead to incorrect predictions. For example, even though the IoU for the "car" class is relatively high (89.5%), the network still struggles to predict reasons such as "front car

Table 3.3: The predicted IOU (%) for each class on the BDD10K dataset.

Class	road	sidewalk	building	wall	fence	pole	light	sign	vegetation
IoU	94.2	64.2	84.7	41.6	52.1	36.9	46.4	47.2	85.7
Class	sky	person	rider	car	truck	bus	train	motocycle	bicycle
IoU	94.5	59.8	13.4	89.5	57.1	78.5	0.00	36.3	36.2

Table 3.4: Comparative results on BDD-OIA and nu-AR for the prediction performance of the Act-Rea sub-network.

Test Set	$F1_{m}^{act}$	$F1_{oval}^{act}$	$F1_{m}^{rea}$	$F1_{oval}^{rea}$
BDD-OIA	0.723	0.733	0.312	0.517
nu-AR	0.688	0.722	0.308	0.499

turning left/right" as it cannot easily infer the direction of the front car's movement.

To further evaluate the generalization ability of the Act-Rea sub-network, we tested it on a new dataset of $1,500$ images selected from the nuScenes dataset [96], which is labeled with 4 driving actions and 21 reasons. This dataset, named *nuScenes Actions and Reasons* (nu-AR), uses the same action and reason categories as the BDD-OIA dataset [29]. We then evaluated the proposed model, pre-trained on BDD-OIA, on the nu-AR dataset. Tab. 3.4 presents the comparative results. As shown, the prediction performance of Act-Rea on BDD-OIA is slightly better than on nu-AR.

For $F1_{m}^{act}$ and $F1_{oval}^{rea}$, the results on BDD-OIA are around 5% higher than those on nu-AR. Similarly, for $F1_{oval}^{act}$ and $F1_{m}^{rea}$, the performance on BDD-OIA is about 1% higher than on nu-AR. These findings confirm the strong generalization capability of Act-Rea sub-network.

3.3.3 Jointly Predicting Actions and Descriptions

To further enhance the explainability of decision-making processes, a new approach is introduced to leverage natural-language descriptions of the ego-vehicle's surrounding environment to explain its decision-making actions. As depicted in Fig. 3.3, this method employs comprehensive descriptions of the surroundings, focusing on categories such as "traffic light allows", "front area is free of obstruction", "left/left-turn area is clear", "right/right-turn area is clear", "left side has solid line" and "right side has solid line". The "left/left-turn area is clear" category encompasses both "left area of the ego-vehicle is clear" and "left-turn area of crossroads is clear", while the "right/right-turn area is clear" description includes both "right area of the ego-vehicle is clear" and "right-turn area of crossroads is clear". Since the relative occurrence of "left/right-turn area of crossroads is clear" is low, the "left/right area of ego-vehicle is clear" and "left/right-turn area of crossroads is clear" are merged into a single category, namely "left/left-turn (right/right-turn) area is clear", to avoid uneven data distribution. Compared to the natural-language reasons for driving actions, the descriptions of the surrounding environment are more direct and objective.

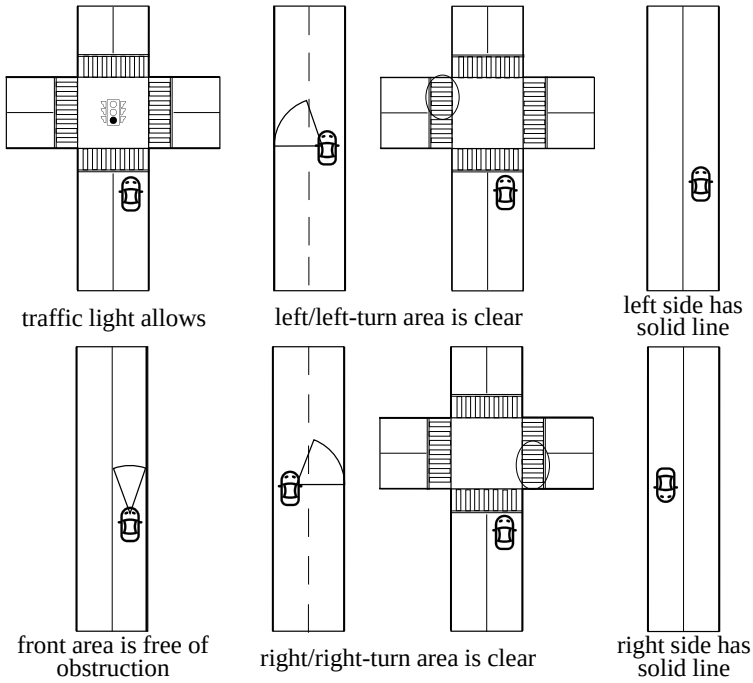


Figure 3.3: The schematic diagram for surrounding environment descriptions of the ego-vehicle.

To utilize these environmental descriptions in explaining decision-making, the Act-Desc sub-network jointly predicts both the driving actions and natural-language environment descriptions. Consequently, the descriptions serve as explanations for the actions taken. For instance, if the natural-language descriptions indicate "traffic light allows" and "front area is free of obstruction", the action "move forward" can be clearly justified.

To train the Act-Desc sub-network to simultaneously predict decision-making actions and describe the ego-vehicle's surrounding environment, a large-scale dataset is constructed, containing manually labeled driving actions and natural-language descriptions. This dataset, termed *BDD Actions and Descriptions* (BDD-AD), is de-

Table 3.5: The categories of the actions and descriptions in the proposed BDD-AD dataset. The ratio refers to the percentage of each category in the dataset.

Annotation	Category	Ratio
Action	Move forward	73.68%
	Stop/slow down	24.78%
	Turn left/change to left lane	39.42%
	Turn right/change to right lane	44.34%
Description	Traffic light allows	73.02%
	Front area is free of obstruction	82.37%
	Left/left-turn area is clear	65.00%
	Right/right-turn area is clear	59.10%
	Left side has solid line	28.83%
	Right side has solid line	18.15%

rived from 10,000 images selected from the BDD-OIA dataset [29], representing a range of weather conditions and times of day. Each image in BDD-AD is annotated with four possible driving actions ("move forward", "stop/slow down", "turn left/change to left lane", "turn right/change to right lane") and six descriptive labels of the ego-vehicle's environment. Additionally, every image includes at least five pedestrians or cyclists, along with more than five vehicles, reflecting the complexity of real-world driving scenarios. Multiple actions and descriptions are assigned to each image to capture these complexities. Tab. 3.5 provides a breakdown of the number

of instances for each category of actions and descriptions. Both the driving actions and environment descriptions are represented in the one-hot encoding. For example, if the vehicle's actions are "move forward" and "change to right lane" and the environment descriptions include "traffic light allows", "front area is free of obstruction", "left/left-turn area is clear", "right/right-turn area is clear", "left side has solid line" and "right side has no solid line", the corresponding annotations for actions and descriptions would be $[1, 0, 0, 1]^T$ and $[1, 1, 1, 1, 1, 0]^T$, respectively.

Tab. 3.6 presents a comparative analysis between the Act-Rea ($\lambda = 1.0$) and Act-Desc ($\lambda = 1.0$) sub-networks. The Act-Rea sub-network jointly predicts decision-making actions alongside natural-language reasons, whereas the Act-Desc sub-network outputs decision-making actions and natural-language descriptions of the surrounding environment. Since both datasets (BDD-OIA for Act-Rea and BDD-AD for Act-Desc) share similar levels of scene complexity and traffic conditions, the two sub-networks are directly comparable. As demonstrated in Tab. 3.6, the Act-Desc sub-network outperforms the Act-Rea sub-network in terms of decision-making action predictions, with $\mathrm{F1}_{\mathrm{m}}^{\mathrm{act}}$ and $\mathrm{F1}_{\mathrm{oval}}^{\mathrm{act}}$ scores approximately 20% higher. Regarding the natural-language explanations, the $\mathrm{F1}_{\mathrm{m}}^{\mathrm{desc}}$ score of Act-Desc sub-network exceeds the $\mathrm{F1}_{\mathrm{m}}^{\mathrm{rea}}$ of the Act-Rea sub-network by around 200%, while its $\mathrm{F1}_{\mathrm{oval}}^{\mathrm{desc}}$ score is about 70% higher. Thus, the Act-Desc sub-network demonstrates superior performance in both decision-making and explanation tasks.

There are several possible explanations for why the Act-Desc sub-network achieves

Table 3.6: Comparative results of the prediction performance for the Act-Desc sub-network and Act-Rea sub-network.

Networks	$F1_{m}^{act}$	$F1_{oval}^{act}$	$F1_{m}^{desc}$	$F1_{oval}^{desc}$	$F1_{m}^{rea}$	$F1_{oval}^{rea}$
Act-Desc	0.876	0.877	0.907	0.880	–	–
Act-Rea	0.723	0.733	–	–	0.312	0.517

better results than the Act-Rea sub-network:

- The six natural-language descriptions of the ego-vehicle's surroundings are more specific and direct than some of the reasons used in Act-Rea (e.g., "follow traffic", "front car turning left/right", "on the left/right turn lane"). This specificity likely contributes to the superior performance of the Act-Desc sub-network in generating natural-language explanations.

- The availability of clear, well-defined natural-language explanations improves the accuracy of decision-making action predictions. This correlation between better natural-language explanations and improved decision-making performance is further supported by the ablation study discussed below.

Fig. 3.4 displays qualitative examples showing the Act-Desc sub-network's ability to predict both decision-making actions and environment descriptions. The examples span various weather conditions and times of day, highlighting the robustness of the proposed model. As illustrated, the predictions for decision-making actions are highly accurate, and most of the predicted descriptions match the ground truth, underscoring

Table 3.7: Comparative results on BDD-AD and nu-AD of the Act-Desc sub-network.

Test Set	$F1_{m}^{act}$	$F1_{oval}^{act}$	$F1_{m}^{desc}$	$F1_{oval}^{desc}$
BDD-AD	0.876	0.877	0.907	0.880
nu-AD	0.760	0.836	0.882	0.879

the network's strong performance.

To further evaluate the generalization capability of the Act-Desc sub-network, a subset of 1.5k images from the nuScenes dataset is manually labeled with the same four driving actions and six natural-language descriptions as in BDD-AD. This new dataset, referred to as *nuScenes Actions and Descriptions* (nu-AD), was used to assess how well the Act-Desc sub-network trained on BDD-AD generalizes to new data. The results, summarized in Tab. 3.7, show that the Act-Desc sub-network achieves around 10% higher action prediction accuracy on BDD-AD compared to nu-AD, with description prediction performance being slightly better on BDD-AD. The $F1_{m}^{desc}$ score is approximately 3% higher on BDD-AD, while the $F1_{oval}^{desc}$ scores for both datasets are almost identical. These findings confirm the effectiveness and generalization capability of the proposed Act-Desc sub-network.

3.3.4 Ablation Study

In the ablation study, we initially examine the relationship between decision-making actions and the associated natural-language explanations. Tab. 3.8 presents

Figure 3.4: The sample prediction results of the decision-making actions and the surrounding environment descriptions of the ego-vehicle.

the prediction performance of the Act-Rea sub-network under different values of λ in the loss function (3.5). As previously mentioned, λ serves as the weighting parameter, adjusting the relative importance between decision-making actions and their corresponding reasons. Setting $\lambda = 0.0$ for the Act-Rea sub-network implies that reason prediction is excluded. Conversely, $\lambda = \infty$ represents a model focused solely on reason prediction, without considering action prediction. As shown in Tab. 3.8, the Act-Rea sub-network with $\lambda = 0.0$ (i.e., action prediction only) demonstrates poorer action prediction performance compared to the Act-Rea with $\lambda = 1.0$ (joint action and reason prediction). This suggests that incorporating reason predictions enhances the accuracy of decision-making action predictions. Similarly, for the Act-Rea sub-network with $\lambda = 0.5$, the action prediction performance exceeds that of the $\lambda = 0.0$ variant but falls short of the $\lambda = 1.0$ configuration. This further confirms the positive influence of reason predictions on action accuracy. However, it is essential to note that this improvement has limits. The Act-Rea with $\lambda = 2.0$ exhibits weaker action prediction performance compared to the model with $\lambda = 1.0$, implying diminishing returns from increased emphasis on reasons.

For the Act-Rea sub-network with $\lambda = \infty$ (reason prediction only), the prediction performance of reasons surpasses that of the Act-Rea with $\lambda = 1.0$, indicating that action predictions do not contribute positively to reason prediction. Additionally, the Act-Rea with $\lambda = 2.0$ outperforms the $\lambda = 1.0$ variant in reason prediction but is less effective than the $\lambda = \infty$ model, which further supports the inference that action

Table 3.8: The ablation study results of prediction performance for Act-Rea sub-networks with the different relative importance of action and reason. The best and second-best results are highlighted in bold font and italic font.

λ	F	S	L	R	$F1_{m}^{act}$	$F1_{oval}^{act}$	$F1_{m}^{rea}$	$F1_{oval}^{rea}$
0.0	0.808	0.710	0.609	0.631	0.690	0.697	–	–
0.5	0.815	0.769	0.646	0.644	*0.718*	0.725	0.302	0.506
1.0	0.827	0.760	0.651	0.653	**0.723**	**0.733**	0.312	0.517
2.0	0.813	0.768	0.649	0.643	*0.718*	*0.728*	*0.350*	*0.546*
∞	–	–	–	–	–	–	**0.372**	**0.568**

predictions do not aid reason predictions.

A similar pattern emerges in the Act-Desc sub-network when examining the relationship between decision-making actions and surrounding environment descriptions (see the top rows of Tab. 3.9). The Act-Desc sub-network with $\lambda = 0.0$ (solely predicting actions) delivers lower accuracy in decision-making compared to the model with $\lambda = 1.0$ (predicting both actions and descriptions), indicating that environment descriptions improve action prediction. Similarly, the Act-Desc with $\lambda = 0.5$ outperforms the model with $\lambda = 0.0$ but underperforms relative to the $\lambda = 1.0$ variant, further confirming the beneficial effect of descriptions on action prediction. For the Act-Desc sub-network with $\lambda = \infty$ (description prediction only), its performance in predicting descriptions exceeds that of the $\lambda = 1.0$ version, suggesting that action predictions do not enhance description accuracy. The prediction accuracy of descriptions

Table 3.9: The ablation study results of the Act-Desc sub-network. The best and second-best results are highlighted in bold font and italic font.

λ /Encoder	F	S	L	R	$F1^{act}_{m}$	$F1^{act}_{oval}$	$F1^{desc}_{m}$	$F1^{desc}_{oval}$
0.0	0.913	0.753	0.725	0.751	0.785	0.794	–	–
0.5	0.941	0.845	0.830	0.836	*0.863*	*0.865*	0.900	0.876
1.0	0.949	0.858	0.832	0.866	**0.876**	**0.877**	0.907	0.880
2.0	0.938	0.848	0.829	0.836	0.862	0.863	*0.909*	*0.889*
∞	–	–	–	–	–	–	**0.921**	**0.894**
ResNet50	0.949	0.858	0.832	0.866	0.876	0.877	*0.907*	0.880
ResNet101	0.955	0.859	0.836	0.870	**0.880**	**0.881**	**0.916**	**0.892**
MobileNetV3-Small	0.921	0.782	0.780	0.788	0.818	0.821	0.824	0.827
MobileNetV3-Large	0.950	0.854	0.840	0.870	*0.878*	*0.880*	0.903	*0.885*

in the Act-Desc sub-network with $\lambda = 2.0$ is superior to that of $\lambda = 1.0$ but lower than $\lambda = \infty$, again validating that action predictions do not contribute to description prediction.

We hypothesize that these outcomes are driven by the intrinsic relationship between decision-making actions and their corresponding explanations. Although the architecture processes action and explanation predictions in parallel, there are interactions or dependencies between them. Decision-making actions can be seen as the outcome of the associated explanations, implying that accurate explanation pre-

dictions help improve action predictions. However, the reverse—action predictions influencing explanation accuracy—does not hold.

In this section, we also evaluate various feature extraction backbones, including ResNet50 (baseline), ResNet101 [123], MobileNetV3-Small, and MobileNetV3-Large [124], within the Act-Desc sub-network ($\lambda = 1.0$) to compare their prediction performance. As shown in the lower portion of Tab. 3.9, the Act-Desc sub-network with the ResNet101 backbone achieves the best prediction results in both decision-making actions and environment descriptions. The performance of the Act-Desc sub-network with the MobileNetV3-Large backbone is comparable to that of the ResNet50-based variant. Despite the Act-Desc sub-network with the MobileNetV3-Small backbone delivering the lowest prediction performance among the tested models, its results remain acceptable, indicating that the Act-Desc sub-network could potentially be deployed on resource-limited mobile devices.

3.3.5 Limitations

Although the proposed proposed network demonstrates significant advantages, it still has certain limitations. Firstly, the current model relies on a single image frame for decision-making, whereas human drivers typically use a sequence of visual information to guide their decisions. Incorporating a sequence of frames, rather than just one, could potentially enhance the accuracy of predicted actions. Furthermore, both the Act-Rea and Act-Desc sub-networks are limited to predicting decision-making ac-

tions from only four predefined categories. Expanding the range of action categories would allow the network to handle more complex situations and function effectively in real-world environments.

3.4 Summary

In summary, we have developed an explainable end-to-end network capable of explaining decision-making actions in autonomous driving by simultaneously predicting both the actions and corresponding natural-language explanations. Two distinct forms of natural-language explanations are provided: the reasons behind the actions and detailed descriptions of the ego-vehicle's surrounding environment. Additionally, we present a dataset that contains manually annotated ground truth, featuring four types of driving actions and six categories of natural-language descriptions of the environment. Through extensive experiments, we demonstrated the effectiveness of the proposed network, outperforming other approaches on both our datasets and a publicly available dataset. Finally, ablation studies shed light on the relationship between decision-making actions and their associated natural-language explanations.

Chapter 4

Multimodal-XAD: Multimodal Explanations for Driving Decisions of Autonomous Driving

4.1 Introduction

In recent years, autonomous driving research has significantly advanced, largely due to breakthroughs in deep learning technologies [125–129]. Despite this progress, most deep learning-based autonomous driving models still struggle with a lack of explainability, as deep neural networks function like black boxes. Without clear explanations for the generated control commands, deploying these systems in real-world scenarios poses safety risks. To address this issue, various methods have been pro-

posed to make autonomous driving networks more explainable. Broadly speaking, these methods offer two types of explanations: visual and natural-language. Visual explanations, such as saliency maps and attention heatmaps, reveal the internal workings of a network [130–132], while natural-language explanations describe network outputs through phrases, including reasons for driving actions or intended goals. Natural-language explanations are often easier for end users to understand compared to visual explanations, providing clearer insights into why a particular action was taken [30]. However, they lack the ability to shed light on the internal processes driving network outputs. As a result, integrating both visual and natural-language explanations may offer a more comprehensive way to interpret the decisions of autonomous driving systems. In this work, an explainable deep neural network is proposed to jointly predict driving actions and explanations in a multimodal format, combining bird-eye-view (BEV) maps with natural-language descriptions of the traffic environment.

BEV perception in traffic scenes has recently garnered significant attention, as BEV maps offer a clear, useful representation for various downstream tasks, such as motion planning and prediction [92]. Current BEV perception methods can be classified based on the sensors used: point cloud-based, vision-based, and multimodal. Point cloud-based methods, which utilize radar or LiDAR, typically do not require view transformation. Vision-based methods, using RGB images from cameras, transform information from the Perspective View (PV) to BEV. Meanwhile, multimodal

approaches combine inputs from multiple sensors (radar, LiDAR, cameras, etc.) to achieve BEV perception. While vision-based methods are more cost-effective than their point cloud and multimodal counterparts, their semantic perception performance tends to be inferior [92]. Moreover, errors in BEV segmentation from vision-based systems may propagate to downstream tasks, exacerbating error accumulation and diminishing overall performance.

To mitigate these challenges, the proposed network incorporates both context information from BEV perception and local information from semantic segmentation (derived from surrounding images) before predicting driving actions and natural-language environment descriptions. This approach aims to reduce error accumulation and improve prediction accuracy. To facilitate the training and evaluation of the proposed model, we introduce a dataset of 12,000 image sequences, each sequence contains images from multiple cameras, along with hand-annotated ground truth for driving actions and multimodal traffic scene descriptions. Experimental results demonstrate that combining context and local information enhances the prediction performance of both driving actions and environment descriptions, leading to improved safety and explainability in autonomous driving systems.

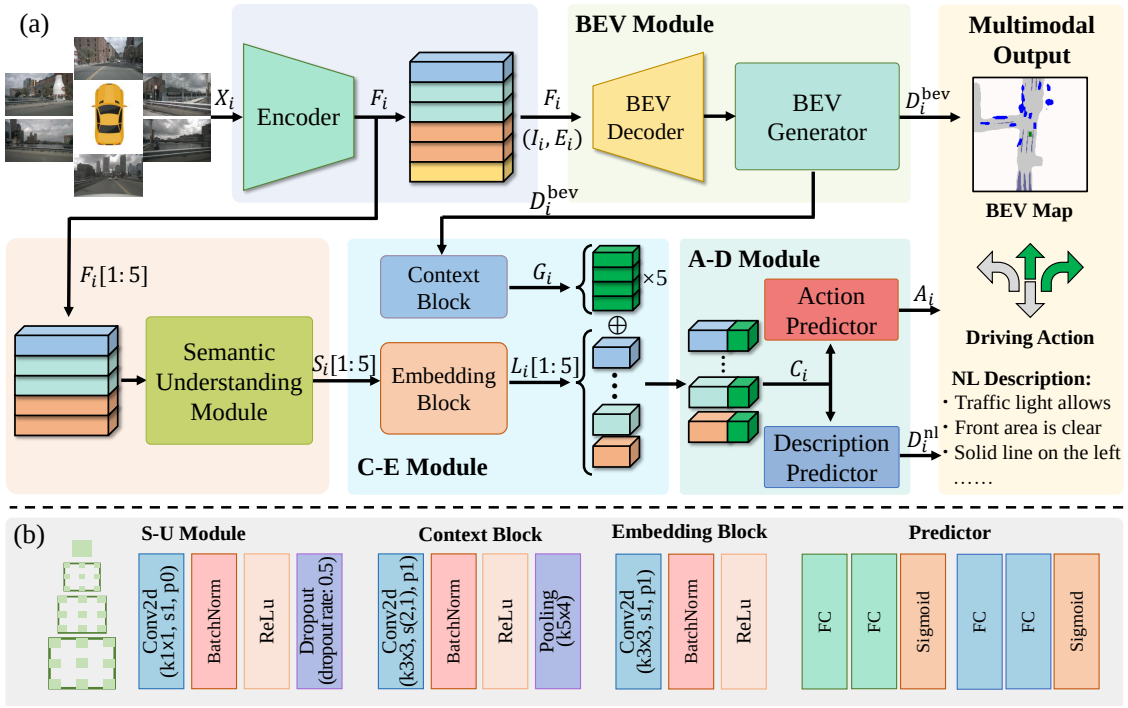


Figure 4.1: The architecture of proposed network. (a) The workflow of the network. (b) Details of the S-U module, context/embedding block and predictors.

4.2 Methodology

4.2.1 The Network Architecture

As illustrated in Fig. 4.1, the proposed Multimodal-XAD network is composed of five key components: the encoder, BEV module, semantic understanding (S-U) module, context embedding (C-E) module, and action-description (A-D) module. The network takes as input images along with the intrinsic and extrinsic parameters from multiple monocular cameras positioned around the vehicle (front, front left, front right, back, back left, and back right). Its objective is to predict driving actions and provide multimodal environmental descriptions, specifically BEV maps and natural-

language descriptions of traffic scenes.

Let $X_i[k]$ represent the k-th image within the sequence $X_i[1:n]$, where i refers to the sequence index and n is the number of surrounding cameras. Each image is associated with an intrinsic matrix $I_i[k] \in \mathbb{R}^{3\times3}$ and an extrinsic matrix $E_i[k] \in \mathbb{R}^{3\times4}$. Here, $X_i[k]$ has dimensions $h \times w \times c$, corresponding to image height, width, and channels. Given the complexity of traffic environments, multiple driving actions might be suitable. Consequently, Multimodal-XAD is designed to predict multiple driving actions, denoted by A_i. Four categories of driving actions are considered: "move forward", "turn left/change to left lane", "turn right/change to right lane", and "stop/slow down", with each action represented as $A_i \in \{0,1\}^4$.

For environment descriptions, the network predicts both BEV maps D_i^{bev} and natural-language descriptions D_i^{nl}. The BEV map is a multi-class semantic grid representing traffic scenes, with a spatial resolution of 0.5 meters per grid in a 100 meter $\times$ 100 meter area. Four semantic classes are used: road, vehicle, road/lane divider, and background, which results in $D_i^{\text{bev}} \in \{0,1,2,3\}^{200\times200}$. On the other hand, the natural-language environment description D_i^{nl} consists of eight categories, such as "traffic light allows", "front area is clear" and "solid line on the left" with each description represented as $D_i^{\text{nl}} \in \{0,1\}^8$. The overall process of Multimodal-XAD (f_{xad}) is summarized as:

$$f_{\text{xad}}(X_i, I_i, E_i) \rightarrow (A_i, D_i^{\text{bev}}, D_i^{\text{nl}}), 1 \leq i \leq T, \tag{4.1}$$

where i is the index and T denotes the total number of image sequences.

EfficientNet [133] is utilized as the encoder due to its favorable balance between efficiency and accuracy [134]. This encoder processes the image sequence X_i and extracts image features F_i from the six cameras. These features are then passed to both the BEV module and the S-U module.

The BEV module, inspired by Lift-Splat [70], consists of a BEV decoder (f_{dec}) and a BEV generator (f_{gen}). Its role is to generate BEV maps D_i^{bev} of traffic scenes. The BEV module operates as:

$$f_{\text{gen}}(f_{\text{dec}}(F_i, I_i, E_i)) \rightarrow D_i^{\text{bev}}, 1 \leq i \leq T, \tag{4.2}$$

where i and T represent the index and total number of sequences. The BEV maps provide a holistic view of the traffic environment in a 100 meter $\times$ 100 meter grid.

The S-U module, responsible for understanding traffic scenes at a finer granularity, leverages Atrous Spatial Pyramid Pooling (ASPP, f_{aspp}) [121], along with convolutional (f_{conv}), batch normalization (f_{bn}), ReLu activation (f_{relu}), and dropout layers (f_{drop}). Further details on ASPP can be found in [121]. The inputs to the S-U module are features $F_i[1:5]$ from the five cameras (front, front left, front right, back left, and back right), while the back camera feature $F_i[6]$ is excluded. The output is a set of semantic features $S_i[1:5]$, which focuses on local road details captured by the cameras. The S-U module's process can be expressed as:

$$f_{\text{drop}}(f_{\text{relu}}(f_{\text{bn}}(f_{\text{conv}}(f_{\text{aspp}}(F_i[1:5]))))) \rightarrow S_i[1:5], 1 \leq i \leq T, \tag{4.3}$$

where i and T represent the sequence index and total number of sequences, respec-

tively.

Next, the BEV maps D_i^{bev} and semantic features $S_i[1:5]$ are processed by the C-E module. The C-E module comprises a context block (f_{cb}) and an embedding block (f_{eb}). The context block includes convolutional, batch normalization, ReLu, and pooling layers, while the embedding block consists of convolutional, batch normalization, and ReLu layers. The context and local information from the BEV maps and semantic features are encoded into G_i and $L_i[1:5]$, respectively, and then concatenated to produce the final features C_i:

$$(f_{\text{cb}}(D_i^{\text{bev}}) \oplus f_{\text{eb}}(S_i[1:5])) \rightarrow C_i, 1 \leq i \leq T, \tag{4.4}$$

where i and T indicate the index and total number of sequences.

Finally, the concatenated features C_i are passed to the A-D module, which predicts both the driving actions and natural-language environment descriptions. This module includes the action predictor (f_{ap}) and the description predictor (f_{dp}), both containing two fully connected layers (f_{fc}) with Sigmoid activation (f_{sgm}). The final process of the A-D module is:

$$(f_{\text{ap}}(C_i), f_{\text{dp}}(C_i)) \rightarrow (A_i, D_i^{\text{nl}}), 1 \leq i \leq T, \tag{4.5}$$

where i and T represent the index and total number of sequences in the dataset.

4.2.2 Training Details

The training of the networks is conducted on an NVIDIA GeForce RTX 3090 GPU. Initially, the encoder and BEV module are pre-trained using the nuScenes dataset for 30 epochs with a batch size of 12. After this, the Multimodal-XAD model is trained on our custom dataset for an additional 60 epochs, using a batch size of 8. It is important to note that not all nuScenes image sequences are utilized during the pre-training phase. The sequences included in the proposed dataset are excluded to avoid overlap. For optimization, the Adam optimizer is employed with an initial learning rate of 1×10^{-4} and a weight decay of 1×10^{-8}. The network training process is governed by a multi-task loss function, which is defined as:

$$\mathcal{L}_{\text{total}} = \lambda_1 \mathcal{L}_{\text{act}} + \lambda_2 \mathcal{L}_{\text{desc}}^{\text{nl}} + \lambda_3 \mathcal{L}_{\text{desc}}^{\text{bev}}, \tag{4.6}$$

where $\mathcal{L}_{\text{total}}$ represents the total loss. The terms $\mathcal{L}_{\text{act}}$ and $\mathcal{L}_{\text{desc}}^{\text{nl}}$ correspond to the binary cross entropy losses for driving action and natural-language environment description predictions, respectively, while $\mathcal{L}_{\text{desc}}^{\text{bev}}$ denotes the cross entropy loss for BEV map predictions. The parameters λ_1, λ_2, and λ_3 control the relative weighting of the losses for driving actions, natural-language descriptions, and BEV maps, respectively.

Table 4.1: Explainable datasets for autonomous driving. The size refers to the number of explanations in the dataset. The action refers to the driving action.

Dataset	Size	Action	Explanation
BDD-X [30]	26,228	✓	Textual justification
BDD-OIA [29]	23,000	✓	Natural-language reasons
BDD-AD [39]	10,000	✓	Natural-language descriptions
HDD [105]	47,533	✓	Textual causal reasoning
PSI [106]	11,902	✓	Text-based reasons
nu-A2D	12,000	✓	Multimodal environment descriptions

4.3 Experimental Results and Discussions

4.3.1 The Dataset

Tab. 4.1 presents a comparison of various explainable autonomous driving datasets. Existing datasets primarily focus on providing natural-language explanations. However, to enhance explainability, combining both visual and natural-language explanations offers a more comprehensive approach for interpreting the outputs of autonomous driving systems. With this goal in mind, we introduce the *nuScenes Action and Multimodal Environment Descriptions* (nu-A2D) dataset, which includes driving actions as well as multimodal environmental descriptions.

The nu-A2D dataset consists of 12,000 image sequences selected from the nuScenes [96] dataset. Each sequence contains six images captured by surrounding cameras, along with ground truth data for driving actions, natural-language descriptions, and BEV maps depicting traffic scenes. The driving actions and natural-language descriptions were manually annotated by our team. To label each sequence, we carefully analyzed the six surrounding camera images to assign the correct driving actions and corresponding environment descriptions. Specifically, the nu-A2D dataset includes four driving action categories and eight types of natural-language environment descriptions. Tab. 4.2 outlines the categories and distribution ratios for both driving actions and natural-language descriptions. The BEV map ground truth is generated by projecting the 3D bounding boxes of objects onto the BEV plane and converting the nuScenes map layers into the ego-vehicle frame.

4.3.2 Evaluation Metrics

To assess the accuracy of the predicted BEV maps, the Intersection over Union (IoU) values for the semantic classes of road, vehicle, and road/lane divider is calculated. For instance, the IoU value for the road class is determined as follows:

$$\text{IoU} = \frac{\text{Area of Intersection}}{\text{Area of Union}} \times 100\%, \tag{4.7}$$

where the intersection refers to the region where both the predicted BEV map and the ground truth label the semantic class as road, while the union represents the total area where either the prediction or the ground truth identifies the road class.

Table 4.2: The categories of the proposed nu-A2D dataset.

Annotation	Category	Ratio (%)
Driving Action	Move forward	80.76
	Stop/slow down	19.23
	Turn left/change to left lane	29.43
	Turn right/change to right lane	38.07
NL Description	Traffic light allows	84.14
	Front area is clear	89.64
	Solid line on the left	27.61
	Solid line on the right	23.74
	Front left area is clear	41.57
	Back left area is clear	42.91
	Front right area is clear	46.93
	Back right area is clear	50.54

Additionally, the mean IoU (mIoU) is calculated to provide the average IoU value across all semantic categories.

To measure the prediction performance for both driving actions and natural-language environment descriptions, the standard F1 score metric is used. Specifically, the overall and mean F1 scores are employed. The overall F1 score for driving actions

is computed as:

$$F1^{\text{act}}_{\text{oval}} = \frac{1}{N}\sum_{i=1}^{N} \text{F1}(A_i, \hat{A}_i), \tag{4.8}$$

while for natural-language environment descriptions, it is given by:

$$F1^{\text{desc}}_{\text{oval}} = \frac{1}{M}\sum_{j=1}^{M} \text{F1}(D_j, \hat{D}_j), \tag{4.9}$$

where $F1^{\text{act}}_{\text{oval}}$ and $F1^{\text{desc}}_{\text{oval}}$ represent the overall F1 scores for driving actions and natural-language descriptions, respectively. Here, A_i and $\hat{A}_i$ are the predicted and actual driving actions, and D_j and $\hat{D}_j$ are the predicted and actual natural-language environment descriptions. The total numbers of predictions for driving actions and natural-language descriptions are denoted by N and M, respectively.

Given the class imbalance in the nu-A2D dataset, where the ratios of different driving actions and natural-language descriptions vary (as detailed in Tab. 4.2), the mean F1 score for both driving actions and natural-language descriptions is also calculate. The mean F1 score for driving actions is computed as:

$$\begin{aligned} F1^{\text{act}}_{\text{m}} = \frac{1}{4}\Big(&\frac{1}{N^{\text{f}}}\sum_{i=1}^{N^{\text{f}}} \text{F1}(A^{\text{f}}_i, \hat{A}^{\text{f}}_i) + \frac{1}{N^{\text{s}}}\sum_{j=1}^{N^{\text{s}}} \text{F1}(A^{\text{s}}_j, \hat{A}^{\text{s}}_j) \\ &+ \frac{1}{N^{\text{l}}}\sum_{k=1}^{N^{\text{l}}} \text{F1}(A^{\text{l}}_k, \hat{A}^{\text{l}}_k) + \frac{1}{N^{\text{r}}}\sum_{p=1}^{N^{\text{r}}} \text{F1}(A^{\text{r}}_p, \hat{A}^{\text{r}}_p)\Big), \end{aligned} \tag{4.10}$$

where $F1^{\text{act}}_{\text{m}}$ is the mean F1 score for driving actions. The number of predictions for each action category, including "move forward", "stop/slow down", "turn left/change to left lane" and "turn right/change to right lane", are denoted by N^{f}, N^{s}, N^{l}, and N^{r}, respectively. The symbols A^{f}_i, $\hat{A}^{\text{f}}_i$, and similarly for the other actions, represent the predicted and ground truth actions for each respective category.

Table 4.3: Comparative results of the prediction performance of driving actions for different networks. Label F denotes "move forward", label S denotes "stop/slow down", label L denotes "turn left/change to left lane" and label R denotes "turn right/change to right lane". The best results are highlighted in bold font.

Networks	F	S	L	R	$F1_{\mathrm{m}}^{\mathrm{act}}$	$F1_{\mathrm{oval}}^{\mathrm{act}}$
Decision Model [37]	0.927	0.621	0.805	0.805	0.790	0.863
VPN [71]	0.916	0.280	0.806	0.866	0.717	0.859
CVT [78]	0.936	0.743	0.835	0.880	0.849	0.898
Multimodal-XAD	0.959	0.798	0.847	0.875	**0.870**	**0.913**

Similarly, the mean F1 score for natural-language environment descriptions is computed as:

$$F1_{\mathrm{m}}^{\mathrm{desc}} = \frac{1}{8}\sum_{e=1}^{8}(\frac{1}{M^e}\sum_{i=1}^{M^e}\mathrm{F1}(D_i^e, \hat{D}_i^e)), \tag{4.11}$$

where $F1_{\mathrm{m}}^{\mathrm{desc}}$ is the mean F1 score for natural-language environment descriptions across the eight categories. The number of predictions for each description category is denoted by M^e, and D_i^e and $\hat{D}_i^e$ represent the prediction and ground truth for each category.

4.3.3 Comparative Results

Tab. 4.3 and Tab. 4.4 present comparative results on the prediction performance of driving actions and multimodal environment descriptions, respectively. The Decision Model [37], which was trained on the A2D dataset (without BEV maps), is

Table 4.4: Comparative results of the prediction performance of multimodal environment descriptions for different networks. The natural-language environment description is labelled as NLD. The best results are highlighted in bold font.

Descriptions	Categories	F1 Score / IoU (%)			
		Decision Model [37]	VPN [71]	CVT [78]	Multimodal-XAD
NLD	Traffic light allows	0.934	0.900	0.928	0.952
	Front area is clear	0.955	0.958	0.970	0.973
	Solid line on the left	0.892	0.907	0.893	0.886
	Solid line on the right	0.876	0.920	0.784	0.811
	Front left area is clear	0.794	0.869	0.862	0.893
	Back left area is clear	–	0.730	0.891	0.899
	Front right area is clear	0.623	0.820	0.877	0.885
	Back right area is clear	–	0.844	0.810	0.750
	$F1_{\mathrm{m}}^{\mathrm{desc}}$	0.846	0.869	0.877	**0.881**
	$F1_{\mathrm{oval}}^{\mathrm{desc}}$	0.857	0.859	0.893	**0.897**
BEV Map	Road	–	60.6	57.5	59.5
	Vehicle	–	25.1	22.8	25.7
	Road/lane divider	–	21.8	26.0	31.0
	mIoU	–	35.8	35.4	**38.7**

designed to jointly predict both driving actions and natural-language environment descriptions. VPN [71] and CVT [78] were modified and trained on the nu-A2D dataset to perform the same task of predicting driving actions and multimodal descriptions. In both VPN and CVT, predictions rely solely on the context information

of traffic scenes. To ensure a fair comparison, all models, including VPN, CVT, and Multimodal-XAD, were pre-trained for the same number of epochs on the nuScenes dataset before being trained on the nu-A2D dataset.

As shown in Tab. 4.3, Multimodal-XAD outperforms the other models in terms of both $F1_{\mathrm{m}}^{\mathrm{act}}$ and $F1_{\mathrm{oval}}^{\mathrm{act}}$. Specifically, $F1_{\mathrm{m}}^{\mathrm{act}}$ for Multimodal-XAD is approximately 10%, 21%, and 2% higher than those for Decision Model, VPN, and CVT, respectively. The overall F1 score, $F1_{\mathrm{oval}}^{\mathrm{act}}$, is similarly higher for Multimodal-XAD, with increases of 6%, 6%, and 2% over Decision Model, VPN, and CVT, respectively. These results indicate that Multimodal-XAD delivers superior driving action prediction, which is crucial for enhancing safety in autonomous driving.

Tab. 4.4 compares the prediction performance for multimodal environment descriptions across models. The $F1_{\mathrm{m}}^{\mathrm{desc}}$ and $F1_{\mathrm{oval}}^{\mathrm{desc}}$ values for Multimodal-XAD and CVT are very similar, with both outperforming the Decision Model and VPN. In particular, $F1_{\mathrm{m}}^{\mathrm{desc}}$ for Multimodal-XAD is about 4% and 1% higher than for Decision Model and VPN, respectively, while $F1_{\mathrm{oval}}^{\mathrm{desc}}$ is 5% and 4% higher. Additionally, the mIoU of BEV maps for Multimodal-XAD exceeds those of VPN and CVT by approximately 8% and 9%, respectively. The enhanced performance in predicting multimodal descriptions provides more accurate and effective explanations for driving actions.

As discussed earlier, the performance of vision-based BEV perception is inherently limited, and this can negatively impact downstream tasks. Unlike VPN and CVT, which rely solely on context information from traffic scenes, Multimodal-XAD

Table 4.5: Computational complexity for different networks on the nu-A2D dataset. The inference speed is tested using an NVIDIA GeForce RTX 3060 GPU.

Configuration	Param	FLOPs	FPS
Decision Model [37]	6.96M	82.91G	18.06
VPN [71]	82.92M	263.79G	13.91
CVT [78]	6.82M	69.40G	19.96
Multimodal-XAD	14.78M	41.57G	21.14

leverages both context from BEV perception and local information from semantic segmentation. This combination helps mitigate error accumulation, which may explain the superior performance of Multimodal-XAD compared to VPN and CVT. On the other hand, the Decision Model depends exclusively on local information, without access to broader context, which limits its ability to comprehensively understand traffic scenes. This may account for its lower prediction performance compared to Multimodal-XAD.

To assess the computational complexity of each model, three key metrics are compared: the number of parameters (Param), Floating Point Operations (FLOPs), and Frames Per Second (FPS) during inference. As shown in Tab. 4.5, while the number of parameters for the Decision Model and CVT are similar and lower than Multimodal-XAD, Multimodal-XAD has the fewest FLOPs, resulting in the highest FPS, indicating superior efficiency during inference.

Table 4.6: Comparative results of the prediction performance for different networks on the BDD-OIA dataset. The best results are highlighted in bold font.

Methods	F	S	L	R	$F1^{act}_{m}$	$F1^{act}_{oval}$	$F1^{rea}_{m}$	$F1^{rea}_{oval}$
OIA [29]	0.829	0.781	0.630	0.634	0.718	0.734	0.208	0.422
CBM-AUC [32]	0.803	0.751	0.551	0.525	0.658	0.704	0.342	0.522
C-SENN [33]	0.772	0.744	0.469	0.486	0.618	–	–	–
CBM [34]	0.795	0.732	0.483	0.431	0.610	0.661	0.292	0.412
Interrelation Model [35]	0.802	0.753	0.619	0.625	0.701	0.722	0.335	**0.537**
Multimodal-XAD	0.822	0.789	0.638	0.641	**0.723**	**0.743**	**0.360**	0.535

Fig. 4.2 illustrates sample qualitative results from the different networks. Challenging traffic scenes were selected to test the models' prediction performance and generalization capabilities. Unlike other models, Multimodal-XAD correctly predicts all driving actions and most natural-language descriptions, closely matching the ground truth. This demonstrates its robust ability to perceive traffic scenes and accurately predict corresponding driving actions and environment descriptions. Regarding BEV map visualization, Multimodal-XAD generates more detailed and accurate BEV maps than the other networks, particularly in identifying vehicles and road/lane dividers—critical elements for safe navigation.

To further validate the effectiveness of Multimodal-XAD, we tested it on the BDD-OIA dataset [29], which contains images labeled with four driving actions and 21 natural-language reasons. For this experiment, Multimodal-XAD was adapted to

Table 4.7: Ablation study results of the prediction performance of driving actions for different networks. The best results are highlighted in bold font.

Networks	F	S	L	R	$F1_{\mathrm{m}}^{\mathrm{act}}$	$F1_{\mathrm{oval}}^{\mathrm{act}}$
Multimodal-XAD	0.959	0.798	0.847	0.875	**0.870**	**0.913**
No Context	0.932	0.584	0.827	0.847	0.798	0.879
No Local	0.940	0.618	0.835	0.848	0.810	0.887
No NLD	0.938	0.655	0.820	0.868	0.820	0.887

predict both driving actions and corresponding reasons. Tab. 4.6 provides comparative results on the BDD-OIA dataset, showing that Multimodal-XAD achieved the highest $F1_{\mathrm{m}}^{\mathrm{act}}$, $F1_{\mathrm{oval}}^{\mathrm{act}}$, and $F1_{\mathrm{m}}^{\mathrm{rea}}$ scores among the models. For $F1_{\mathrm{oval}}^{\mathrm{rea}}$, Multimodal-XAD and the Interrelation Model performed similarly, both outperforming the other networks. These results demonstrate that Multimodal-XAD consistently outperforms the alternatives in predicting both driving actions and their corresponding natural-language explanations.

4.3.4 Ablation Study

In the ablation study, we first examine the impact of combining context and local information on the prediction performance of driving actions and multimodal environment descriptions. Tab. 4.7 displays the results of this study for various networks. In the `No Context` network, context information is excluded from the C-E

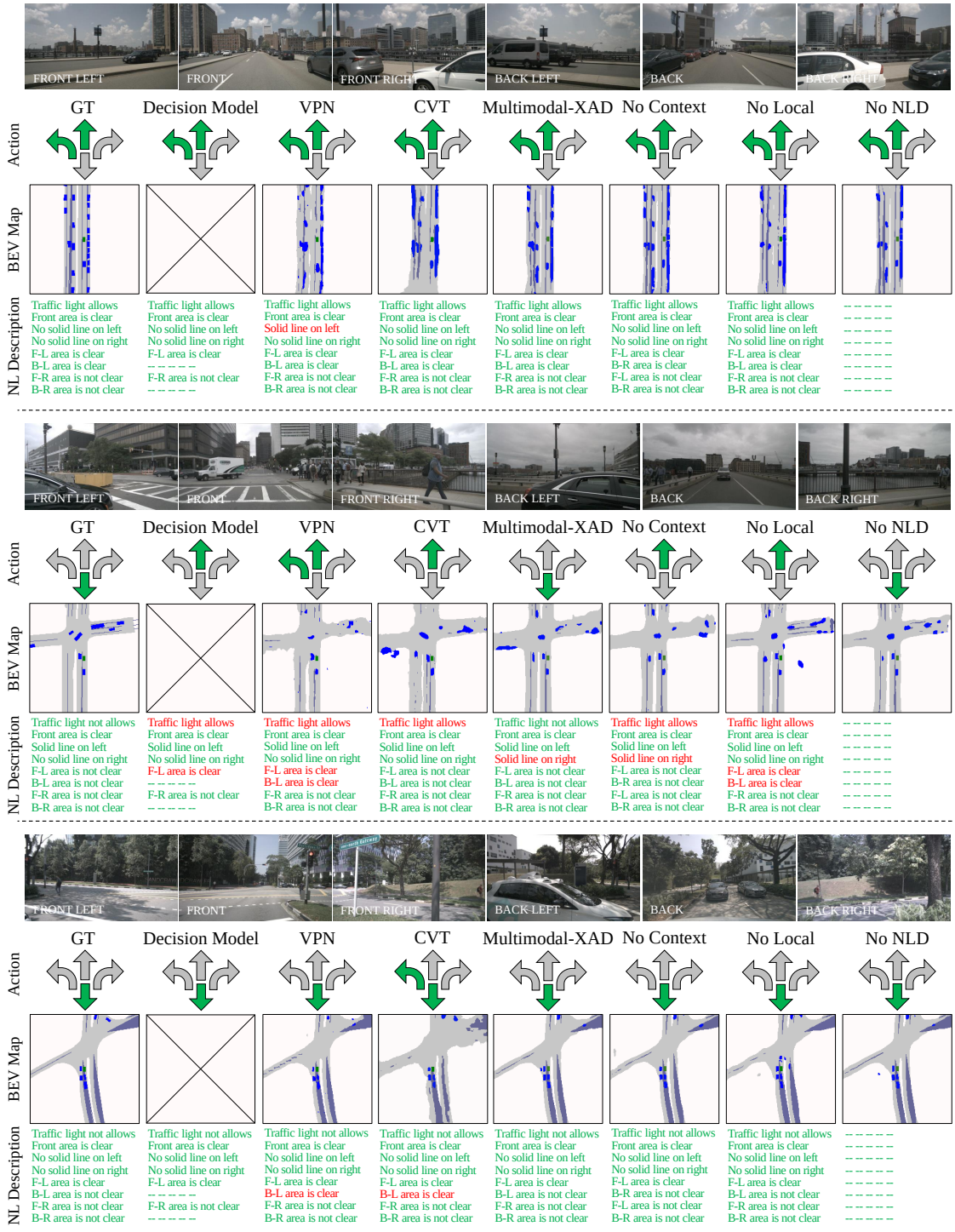


Figure 4.2: Sample qualitative results of predictions of driving actions and multimodal environment descriptions for different networks.

Table 4.8: Ablation study results of the prediction performance of multimodal environment descriptions for different networks. The natural-language environment description is labelled as NL Description. The best results are highlighted in bold font.

Descriptions	Categories	F1 Score / IoU (%)			
		Multimodal-XAD	No Context	No Local	No NLD
NL Description	Traffic light allows	0.952	0.933	0.928	–
	Front area is clear	0.973	0.969	0.966	–
	Solid line on the left	0.886	0.860	0.826	–
	Solid line on the right	0.811	0.802	0.835	–
	Front left area is clear	0.893	0.901	0.882	–
	Back left area is clear	0.899	0.878	0.835	–
	Front right area is clear	0.885	0.856	0.820	–
	Back right area is clear	0.750	0.752	0.782	–
	$F1_{\mathrm{m}}^{\mathrm{desc}}$	**0.881**	0.869	0.859	–
	$F1_{\mathrm{oval}}^{\mathrm{desc}}$	**0.897**	0.884	0.869	–
BEV Map	Road	59.5	59.2	60.7	61.3
	Vehicle	25.7	25.8	26.2	27.6
	Road/lane divider	31.0	31.1	29.0	31.6
	mIoU	38.7	38.7	38.6	**40.2**

Module, meaning only local information from semantic perception is used to predict driving actions and natural-language environment descriptions. Conversely, in the `No Local` network, local information is omitted, and predictions are based solely on context information from BEV perception. As shown in Tab. 4.7, both the $F1_{\text{m}}^{\text{act}}$ and $F1_{\text{oval}}^{\text{act}}$ values of Multimodal-XAD outperform those of the `No Context` and `No Local` networks. These findings confirm that integrating both context and local information significantly enhances the prediction performance for driving actions.

Similarly, Tab. 4.8 presents the results for the prediction performance of multimodal environment descriptions. As with driving actions, the $F1_{\text{m}}^{\text{desc}}$ and $F1_{\text{oval}}^{\text{desc}}$ scores for Multimodal-XAD are higher than those of the `No Context` and `No Local` models. This further validates the advantage of using both context and local information in improving natural-language environment description predictions. For the BEV maps, the mIoU values for Multimodal-XAD, `No Context`, and `No Local` networks are comparable, indicating that the use of context and local information has minimal effect on BEV map prediction performance.

We also explore the role of natural-language environment descriptions in predicting driving actions and BEV maps. As shown in Tab. 4.7, both the $F1_{\text{m}}^{\text{act}}$ and $F1_{\text{oval}}^{\text{act}}$ values for Multimodal-XAD surpass those of the `No NLD` network (which lacks natural-language descriptions). This suggests that incorporating natural-language environment descriptions improves the prediction of driving actions. However, for BEV map prediction (Tab. 4.8), the mIoU for Multimodal-XAD is lower than for

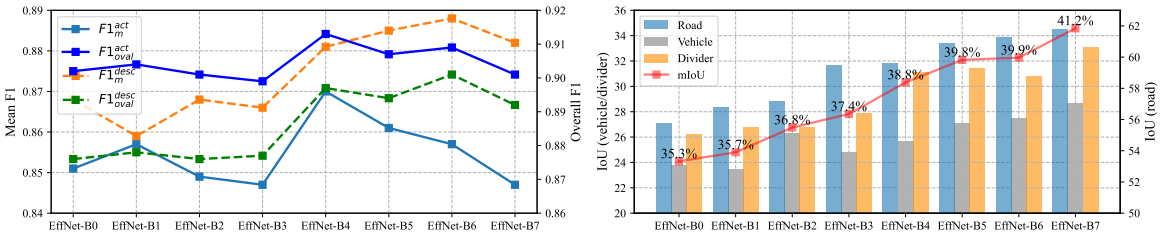


Figure 4.3: Ablation study results of the prediction performance of Multimodal-XAD with different encoders of the EfficientNet family. The left figure shows the F1 scores of the driving action and natural-language environment description predictions. The right figure shows the IoU of predictions of BEV maps. EffNet is the short for EfficientNet.

the `No NLD` network, indicating that natural-language descriptions do not contribute positively to BEV map prediction performance.

Fig. 4.2 also provides qualitative results for the `No Context`, `No Local`, and `No NLD` networks. The prediction performance of driving actions and natural-language environment descriptions for both the `No Context` and `No Local` models is notably lower than that of Multimodal-XAD. This highlights the importance of integrating both context and local information in improving the accuracy of driving actions and natural-language environment descriptions. In the case of the `No NLD` network, the overall explainability is reduced compared to Multimodal-XAD due to the absence of natural-language environment descriptions.

In this section, we also examine the effect of using different encoders on the performance of Multimodal-XAD. Fig. 4.3 shows the results of an ablation study where Multimodal-XAD is tested with various EfficientNet variants, ranging from Efficient-B0 to Efficient-B7. The left plot in Fig. 4.3 presents the F1 scores for

driving actions and natural-language descriptions with different encoders. As shown, Multimodal-XAD with Efficient-B4 achieves the highest $F1^{\text{act}}_{\text{m}}$ and $F1^{\text{act}}_{\text{oval}}$, indicating that this variant provides the best driving action prediction performance among all tested encoders. For natural-language environment descriptions, Multimodal-XAD with Efficient-B6 performs the best, with both $F1^{\text{desc}}_{\text{m}}$ and $F1^{\text{desc}}_{\text{oval}}$ reaching the highest values. Efficient-B4, on the other hand, ranks fourth and second in $F1^{\text{desc}}_{\text{m}}$ and $F1^{\text{desc}}_{\text{oval}}$, respectively.

The right-hand plot in Fig. 4.3 illustrates the IoU values for different BEV map classes when using various EfficientNet encoders. As shown, there is a clear upward trend in BEV map prediction performance as the complexity of the EfficientNet variant increases from B0 to B7. To balance prediction performance with computational efficiency, EfficientNet-B4 is selected as the default encoder for Multimodal-XAD model.

The ablation study also explores how the relative importance assigned to driving actions, natural-language environment descriptions, and BEV maps affects the prediction performance of Multimodal-XAD. This relative importance is controlled by the values of λ_1, λ_2, and λ_3 in the loss function (4.6). We evaluated four different configurations to examine the impact of these weight parameters. As shown in Tab. 4.9, when driving actions are given greater importance ($\lambda_1 = 2$), Multimodal-XAD achieves its best performance in predicting driving actions. Similarly, increasing the importance of natural-language environment descriptions ($\lambda_2 = 2$) results in the high-

Table 4.9: The ablation study results of prediction performance for Multimodal-XAD networks with different relative importance between driving actions, natural-language environment descriptions and BEV maps. The best results are highlighted in bold font.

$\lambda_1 : \lambda_2 : \lambda_3$	$F1^{\text{act}}_{\text{m}}$	$F1^{\text{act}}_{\text{oval}}$	mIoU (%)	$F1^{\text{desc}}_{\text{m}}$	$F1^{\text{desc}}_{\text{oval}}$
1:1:1	0.870	0.913	38.7	0.881	0.897
2:1:1	**0.873**	**0.914**	38.0	0.863	0.875
1:2:1	0.855	0.903	37.8	**0.895**	**0.906**
1:1:2	0.828	0.892	**40.3**	0.878	0.892

est performance for this task, and emphasizing BEV maps ($\lambda_3 = 2$) leads to the best results for BEV map predictions. To ensure balanced performance across driving actions, natural-language environment descriptions, and BEV maps, we selected a configuration where $\lambda_1 : \lambda_2 : \lambda_3 = 1 : 1 : 1$ as the default for Multimodal-XAD.

4.3.5 Limitations

Despite the advantages of the proposed Multimodal-XAD, there are still some limitations. Firstly, the prediction performance of the multimodal environment description is currently assessed using the IoU for the BEV map and the F1 score for natural-language descriptions. However, to more thoroughly evaluate the explainability of Multimodal-XAD, a new metric that can holistically measure the effectiveness of multimodal environment descriptions should be explored. Secondly, while the driv-

ing action is connected to BEV perception through the use of context information for action prediction, we believe that a stronger coupling between driving actions and BEV perception could further enhance performance. One potential approach to achieve this is by employing a generative adversarial network (GAN), where the driving action predictor acts as the generator, and the BEV module functions as the discriminator.

4.4 Summary

To enhance both the safety and explainability of deep learning-based end-to-end autonomous driving systems, we introduced an explainable network designed to jointly predict driving actions and multimodal environment descriptions, which include BEV maps and natural-language descriptions of traffic scenes. In the proposed model, both context information from BEV perception and local information from semantic segmentation are incorporated before making predictions about driving actions and environment descriptions. Additionally, we have released a new dataset consisting of 12,000 image sequences, where each sequence includes six frames captured by surrounding visual cameras, along with manually annotated ground truth for driving actions and multimodal environment descriptions. Experimental results demonstrate that combining context and local information significantly improves the accuracy of predictions for both driving actions and environment descriptions.

Chapter 5

PolarPoint-BEV: BEV Perception of Driving Environment in Polar Points

5.1 Introduction

In recent years, end-to-end autonomous driving has gained significant attention. This approach processes raw sensory inputs and outputs either waypoints or direct control actions. The waypoints can be utilized by low-level controllers such as Proportional-Integral-Derivative (PID) or Model Predictive Control (MPC) to generate control signals. Compared to modular pipelines, which involve multiple interconnected modules like localization [135, 136], perception [137], planning [113], and

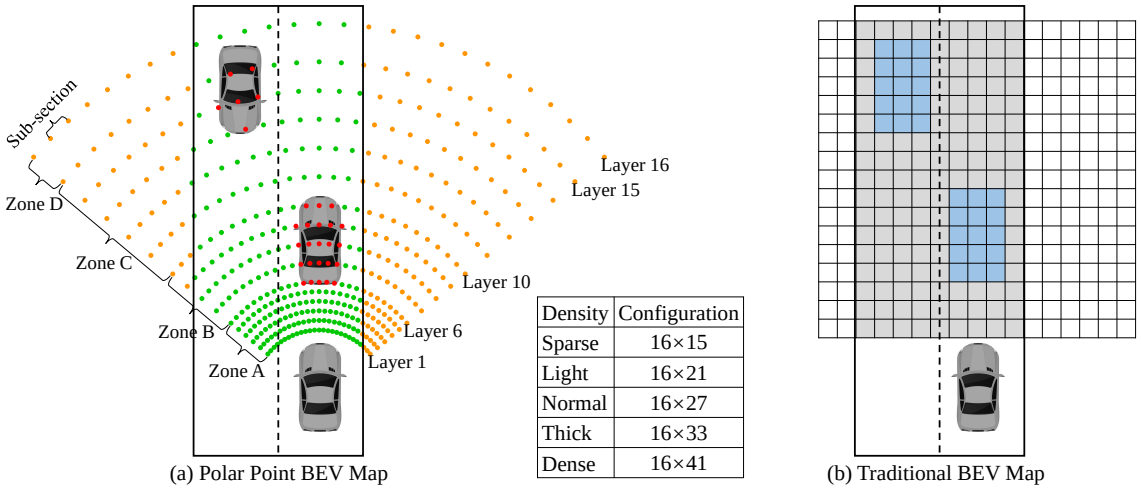
control [138], end-to-end methods bypass the issue of cumulative errors across modules, and exhibit better scalability for handling complex scenarios. Numerous research efforts [1–12] have advanced the field, leading to substantial progress. However, these methods often lack explainability due to the opaque nature of deep neural networks, making them susceptible to unpredictable mistakes and safety risks. This absence of transparency creates a barrier to their widespread adoption in real-world driving scenarios. To address this issue, various eXplainable Artificial Intelligence (XAI) techniques have been introduced, such as producing semantic bird's-eye-view (BEV) maps to provide interpretable insights into the decision-making process of end-to-end driving systems.

The generation of semantic BEV maps has recently become an active research topic in autonomous driving, as this representation offers a clear and intuitive format for downstream tasks like trajectory planning [24] and control [27]. Furthermore, these maps enable visualization of how autonomous driving systems perceive and interpret surrounding traffic conditions, offering a valuable tool for explaining decisions in end-to-end driving models. There have been numerous contributions in this area [66–86].

Despite the successes, traditional semantic BEV maps exhibit certain drawbacks. First, they typically treat all regions within the traffic scene equally, despite the fact that objects near the ego vehicle are generally more critical for ensuring safety [139, 140]. In traditional BEV maps, distant regions, which may pose lower im-

mediate risks, receive the same focus as areas in close proximity to the vehicle. This uniform attention allocation can lead to a failure in prioritizing crucial safety-related information. Additionally, traditional BEV maps rely on dense, pixel-level representations of the environment, which impose substantial computational, communication, and memory overheads. Given the limited processing power on autonomous vehicles, these high resource demands may introduce delays that jeopardize safety [141, 142].

To address these limitations, an innovative BEV perception method called PolarPoint-BEV is proposed. Unlike traditional approaches, the proposed method emphasizes the regions near the ego vehicle, which are more pertinent to safe driving. Moreover, PolarPoint-BEV adopts a sparse representation, significantly reducing the computational load compared to dense BEV maps. This lightweight structure makes it a more practical solution for deployment on vehicles with restricted computational capacity. To assess the effectiveness of PolarPoint-BEV in enhancing both explainability and driving performance, an end-to-end autonomous driving network, called eXplainable Planning (XPlan) is introduced. XPlan utilizes a multi-task architecture to jointly predict control commands and generate polar point BEV maps as interpretable explanations. We validate the proposed network's performance in the CARLA simulator [42]. Experimental results demonstrate that PolarPoint-BEV improves both driving performance and the explainability of the end-to-end driving model.



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

Figure 5.1: Schematic diagram of the proposed polar point BEV map (Sub-fig. (a)) and the traditional BEV map (Sub-fig. (b)). In the proposed polar point BEV map, the orange, red, green colors represent background, vehicle and road.

5.2 The proposed network

5.2.1 The PolarPoint-BEV

As discussed earlier, traditional BEV methods have certain limitations. To address these, the proposed PolarPoint-BEV introduces a polar point-based BEV map to demonstrate how the network interprets and perceives the surrounding environment, providing an explainable output for the end-to-end autonomous driving model. Fig. 5.1 illustrates the comparison between the proposed PolarPoint-BEV map and the traditional BEV approach. In the traditional method, the traffic scene is represented by a uniformly distributed rectangular grid map aligned with Cartesian axes, while the PolarPoint-BEV map employs a series of points dispersed around the ego vehicle, offering a more adaptive representation. Each point on the PolarPoint-BEV

Table 5.1: Details of each zone for 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}

map is assigned a semantic class based on the object located at that point. More specifically, the PolarPoint-BEV map categorizes points into three distinct semantic classes $\{0, 1, 2\}$, representing different types of objects in the scene. These are visualized in Fig. 5.1 using different colors: class $\{0\}$ corresponds to the background (orange), class $\{1\}$ represents vehicles (red), and class $\{2\}$ denotes roads (green).

The location of each point is expressed in polar coordinates. In the angular dimension, the field of view (FOV) spans 100°, consistent with the horizontal FOV of the front-view camera. The angular range is divided into subsections, and we experiment with configurations ranging from 15 to 41 subsections (see Fig. 5.1). In the radial dimension, the map is divided into 16 layers, forming the complete PolarPoint-BEV map. Therefore, the map can be defined as $P_i \in \{0, 1, 2\}^{16\times n}$, where 16 refers to the number of radial layers and n is the number of angular subsections determined by the chosen configuration.

The 16 radial layers are grouped into four distinct zones: Zone A, Zone B, Zone

C, and Zone D. As outlined in Table 5.1, each zone has different intervals between the layers. Recognizing that regions nearer to the ego vehicle are more safety-critical, Zone A and Zone B feature finer resolution (smaller intervals), while Zones C and D have coarser resolution (Zone A $<$ Zone B $<$ Zone C $<$ Zone D). The density of the PolarPoint-BEV map is defined as the ratio of semantic points to the area they occupy. Table 5.1 summarizes the densities for the different zones under the normal configuration. The density in Zone A is approximately nine times higher than in Zone D, reflecting the increased focus on closer, more critical regions. By contrast, traditional BEV maps distribute semantic points uniformly across the scene, neglecting the varying importance of different regions. This allows the PolarPoint-BEV map to prioritize areas near the ego vehicle, where immediate safety considerations are most crucial.

5.2.2 The Network Structure

Fig. 5.2 illustrates the architecture of proposed XPlan network for autonomous driving. XPlan primarily consists of three main components: an encoder, the Control Prediction (C-P) module, and the Polar-Point (P-P) module. The network processes input from both the front-view RGB image I_i and the navigation data S_i to produce control commands C_i, as well as a polar point BEV map P_i for explainability. Here, $I_i \in \mathbb{R}^{\mathrm{h}\times\mathrm{w}\times\mathrm{c}}$ represents the RGB image, with h, w, and c denoting its height, width, and number of channels. The navigation input S_i includes the vehicle's current speed

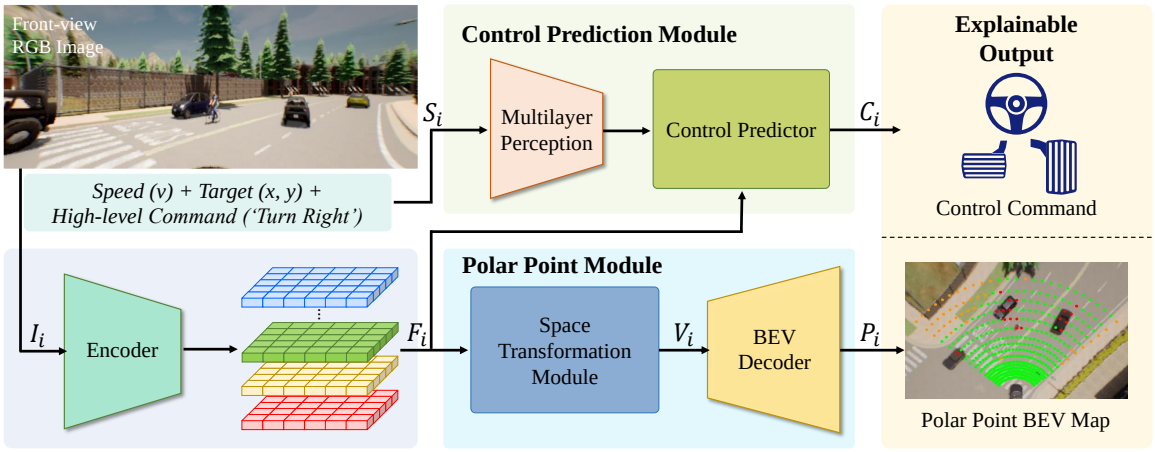


Figure 5.2: The structure of proposed XPlan network. This explainable end-to-end network takes as input the front-view RGB image as well as navigation information, and outputs control commands along with polar point BEV map as the explanations.

v, the target position (x, y), and a high-level command. The control output C_i consists of steering, throttle, and brake values. Thus, the overall process of the XPlan network can be represented as:

$$\text{XPlan: } (I_i, S_i) \rightarrow (C_i, P_i \in \{0, 1, 2\}^{16 \times n}). \tag{5.1}$$

The network employs ResNet-34 as the encoder, extracting feature maps F_i from the input images. These feature maps are then fed into both the C-P and P-P modules. The C-P module, based on the Trajectory-guided Control Prediction (TCP) network [7], takes the feature maps F_i and navigation data S_i as inputs to generate control outputs C_i. For detailed information on the TCP network, we refer readers to [7]. The C-P module's function can thus be summarized as $(F_i, S_i) \rightarrow C_i$.

The P-P module is responsible for generating the polar point BEV map, P_i, to provide explanation for the control decisions made by the network. It consists

of two submodules: the space transformation module and the BEV decoder. In the space transformation module, feature maps from the front-view are flattened and processed by a Multilayer Perceptron (MLP), which learns the transformation from front-view to BEV space. The MLP translates the perspective of the front-view feature maps into the BEV domain. These transformed feature maps are then reshaped and passed through the BEV decoder to generate the polar point BEV map, consisting of semantic points. The process for the P-P module can be expressed as: $F_i \rightarrow P_i$.

Five configurations of the polar point BEV map are evaluated, ranging from sparse to dense representations. The version of XPlan that predicts the normal polar point BEV map (16×27) is referred to as XPlan-N, while the variations predicting sparse, light, thick, and dense BEV maps are denoted as XPlan-S, XPlan-L, XPlan-T, and XPlan-D, respectively.

5.2.3 Dataset and Training Details

The dataset is collected within the CARLA simulator by running randomly generated routes under a variety of weather and lighting conditions. These include 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

created using Roach [143] as the expert, which has access to privileged information about traffic elements such as roads, vehicles, pedestrians, and traffic lights. This dataset comprises over 92k data batches from 6 of CARLA public towns (Town01, Town03, Town04, Town06, Town07, and Town10), encompassing small towns, quiet rural areas, and inner-city environments. Each data batch includes a front-view RGB image, current speed, future target coordinates, high-level commands, ground-truth waypoints, control values, and the BEV map. The XPlan network is trained using this dataset. For evaluation, the routes from [6] are utilized, which include traffic scenarios from two other towns in CARLA (Town02 and Town05), featuring both small town and urban settings.

All networks are trained on an NVIDIA GeForce RTX 3090 GPU, while inference speeds are measured on an NVIDIA GeForce RTX 3060 GPU. Training proceeds in two stages: first, the C-P module is pre-trained using the dataset from [7]. In the second stage, the entire XPlan network is trained and evaluated using our collected dataset. The Adam optimizer is applied with an initial learning rate of 1×10^{-4} and a weight decay of 1×10^{-7}. A multi-task loss function is designed for XPlan, calculated as $\mathcal{L}_{total} = \mathcal{L}_{ctrl} + \lambda_1 \mathcal{L}_{bev}$. Here, $\mathcal{L}_{total}$ represents the total loss, $\mathcal{L}_{ctrl}$ corresponds to the loss associated with control command prediction, and $\mathcal{L}_{bev}$ relates to the loss for generating the polar point BEV map. The coefficient λ_1 adjusts the balance between control command prediction and BEV map generation. For the polar point BEV map, each semantic point falls into one of three classes. Therefore, $\mathcal{L}_{bev}$ is

calculated as the sum of cross-entropy losses for each class: $\mathcal{L}_{bev} = \mathcal{L}_{vehicle} + \lambda_2 \mathcal{L}_{road} + \lambda_3 \mathcal{L}_{backg}$. Here, $\mathcal{L}_{vehicle}$, $\mathcal{L}_{road}$, and $\mathcal{L}_{backg}$ represent the loss terms for vehicle, road, and background classes, respectively. The weighting coefficients λ_2 and λ_3 control the relative importance of these semantic classes in the BEV map generation.

5.3 Experimental Results and Discussions

5.3.1 Evaluation Metrics

The driving performance and polar point BEV map prediction capabilities of the proposed XPlan network are assessed within the CARLA simulator. In this environment, the autonomous agent has two key objectives: 1) to safely and efficiently navigate a predefined path to reach its designated destination, and 2) to accurately predict the polar point BEV map of the traffic scene, providing explanations for its control commands. To evaluate driving performance, several metrics are employed in the CARLA simulator, including three main indicators: route completion, infraction score, and driving score. Route completion measures the percentage of the path the agent successfully covers. The infraction score reflects the number of rule violations, such as collisions with pedestrians or vehicles, running red lights, and improper use of road layouts. The driving score is calculated as the product of route completion and infraction score. Additional metrics are also considered to target specific driving behaviors, such as collisions with vehicles or road layouts, red light violations, off-road

infractions, and instances where the agent becomes blocked.

To assess the overall accuracy of the polar point BEV map predictions, the Intersection-over-Union (IoU) and F1 scores are computed for road, vehicle, and background classes. However, traditional IoU and F1 metrics assign equal weight to all points on the BEV map. Recognizing that areas closer to the ego vehicle are more critical for safety, we introduce a new metric called the weighted Intersection-over-Union (wIoU). The wIoU is calculated as:

$$\text{wIoU} = \sum_{z=A}^{D}(\text{mIoU}_z \times N_z), \quad N_z = L_z^{-1} / \sum_{z=A}^{D}(L_z^{-1}), \tag{5.2}$$

where mIoU_z represents the mean IoU for each zone (A to D) in the BEV map, and N_z is the normalized weight for each zone. The term L_z^{-1} denotes the reciprocal of the distance from each zone to the ego vehicle. This weighting ensures that regions closer to the ego vehicle have a greater influence on the wIoU score.

5.3.2 Comparative Results

We first compare the prediction performance of polar point and traditional BEV maps. We propose an end-to-end network called the Planning-BEV (Plan-B) network, which jointly predicts control commands for the agent along with the traditional BEV map. In the Plan-B network, the encoder and C-P module remain identical to those in the XPlan network, but the P-P module is replaced with a traditional BEV generation module based on the View Parsing Network (VPN) [75]. Both XPlan and

Table 5.2: Comparative results of the overall prediction performance for the polar point and traditional BEV maps. The mean and standard deviations are calculated over 3 runs. The best results are highlighted in bold font.

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

Table 5.3: Comparative results of the mIoU of different zones and the wIoU for the polar point and traditional BEV maps. The mean and standard deviations are calculated over 3 runs. The best results are highlighted in bold font.

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**

Plan-B networks are trained for 60 epochs with the same pre-trained weights for the C-P module, and the weighting coefficients (λ_2 and λ_3 in the loss function) are kept the same for both networks.

Tab. 5.2 presents the comparative results for the overall prediction performance of the polar point and traditional BEV maps. Each network was tested over three runs, and the mean and standard deviations for IoU and F1 scores are reported. As shown in the table, the mIoU and overall F1 scores of the polar point BEV map are similar to those of the traditional BEV map, suggesting comparable overall prediction performance. However, for vehicles and roads, the polar point BEV map achieves

Table 5.4: Computational complexity for different networks. The inference speed is tested using an NVIDIA GeForce RTX 3060 GPU.

Network	Param	MACs	FPS
TCP [7]	25.77M	17.09G	137.88
Plan-B	38.58M	27.21G	90.08
XPlan-N	27.57M	17.58G	132.66

higher IoU and F1 scores than the traditional BEV map.

Tab. 5.3 provides a comparison of mIoU for different zones and wIoU for both polar point and traditional BEV maps. The polar point BEV map outperforms the traditional BEV map in Zones A and B, while the traditional BEV map performs better in Zone D. In Zone C, the performance of both maps is comparable. The wIoU score for the polar point BEV map is approximately 4% higher than for the traditional BEV map, indicating that while their overall performance is similar, the polar point BEV map excels in predicting areas closer to the ego vehicle.

Fig. 5.3 shows examples of polar point and traditional BEV maps under different weather and lighting conditions, such as SoftRainDawn, ClearNoon, CloudySunset, and HardRainNight. Both polar point and traditional BEV maps effectively represent the traffic scenes and demonstrate how the network perceives and understands the surrounding environment. Thus, both approaches offer valid explanations for the control commands generated by the end-to-end networks.

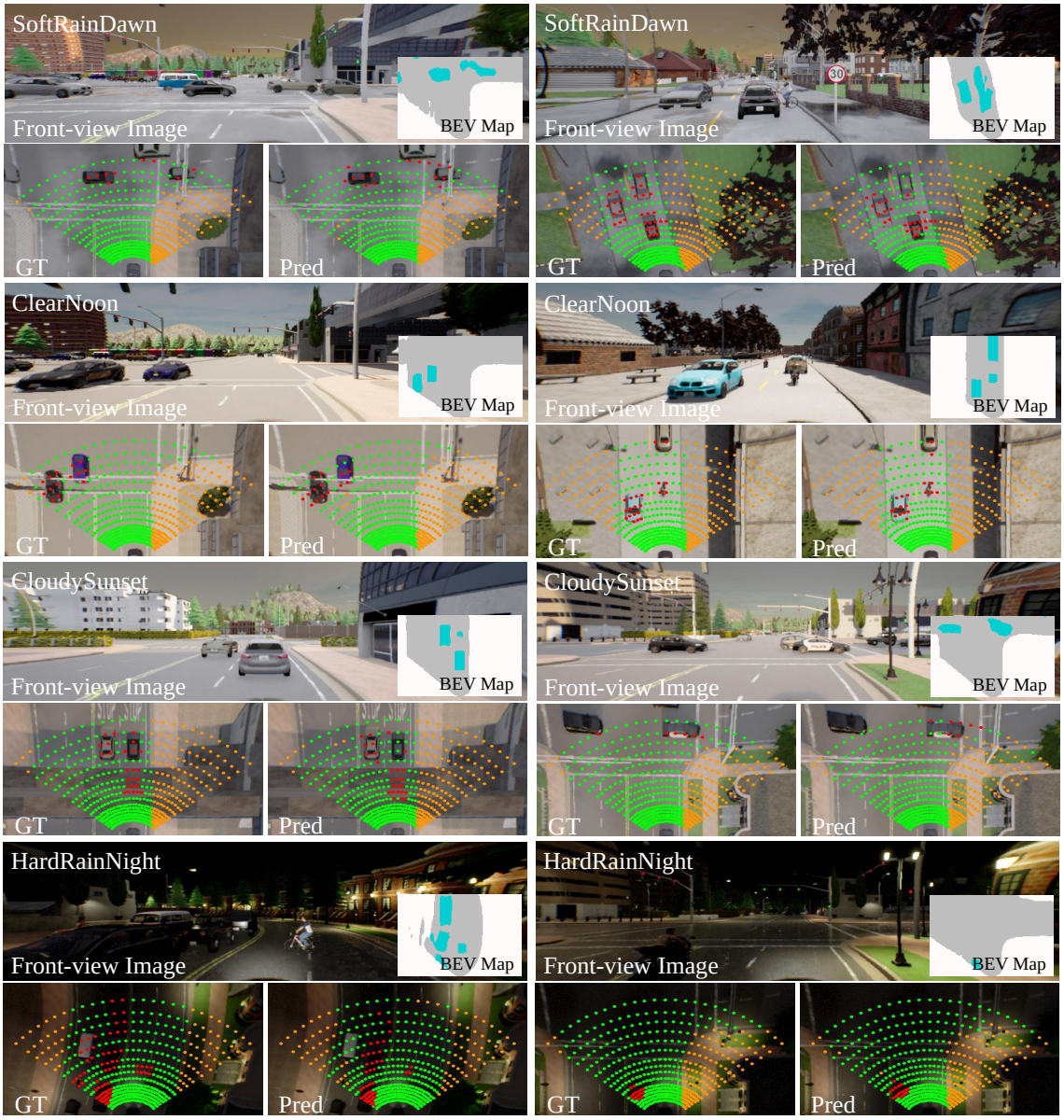


Figure 5.3: Sample qualitative results of the polar point and traditional BEV maps. GT and Pred refer to ground truth and prediction. In the polar point BEV maps, the points with orange, red and green colors respectively represent the background, vehicle and road.

Tab. 5.4 compares the computational complexity of different networks. The TCP [7] network can be considered as the XPlan/Plan-B network without the P-

Table 5.5: Comparative results of the driving performance for different networks. The mean and standard deviations are calculated over 3 runs. The best and second-best results are highlighted in bold font and italic font.

Networks	Driving Score	Route Completion	Infraction Score	Vehicle Collisions	Layout Collisions	Red Light Infractions	Off-road Infractions	Agent Blocked
LAV [6]	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 [7]	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

P/BEV module. Compared to TCP, the XPlan-N network has a slight increase in complexity, with the number of parameters (Param) and multiply–accumulate operations (MACs) being about 7% and 3% higher, respectively. The XPlan-N network's inference speed is approximately 4% lower than TCP. In contrast, Plan-B significantly increases computational complexity, with its Param and MACs being about 50% and 60% higher than TCP, respectively, and its inference FPS around 35% lower. This indicates that the PolarPoint-BEV approach offers a lightweight and efficient method for describing traffic scenes and explaining control commands. By using the polar point BEV map, XPlan reduces the computational costs associated with BEV generation, striking a balance between efficiency and accuracy, making it a suitable choice for autonomous driving systems where real-time performance and computational resources are critical.

The effect of polar point and traditional BEV maps on driving performance in end-to-end networks are also examined. Tab. 5.5 summarizes the comparative driv-

ing performance of different networks. The results of the LAV network are taken from [6], trained on 186K data points from four towns and tested on the same routes as the other networks. As shown in Tab. 5.5, both XPlan-N and Plan-B outperform LAV and TCP in terms of driving performance. Since TCP can be considered as XPlan/Plan-B without the P-P/BEV module, we can conclude that both PolarPoint-BEV and traditional BEV generation positively impact driving performance. Additionally, XPlan-N outperforms Plan-B in driving metrics: the driving score for XPlan-N is approximately 9% higher than for Plan-B, with route completion and infraction score being 2% and 4% higher, respectively. We attribute this superior performance to the following reasons:

1. The polar point BEV map emphasizes areas close to the ego vehicle, where critical safety decisions are often made. As the results show, the polar point BEV map performs better than the traditional BEV map in regions near the ego vehicle, which are crucial for ensuring safe navigation. This likely enhances the system's ability to perceive and respond to potential hazards, leading to safer autonomous driving.

2. While the overall prediction performance of the polar point and traditional BEV maps is similar, the polar point BEV map achieves better results for vehicle and road classes. These are far more relevant to the safety of the autonomous agent than background elements, so the improved prediction of these classes translates into better driving performance.

Table 5.6: Comparative results of the overall prediction performance for the polar point and traditional BEV maps on the nuScenes dataset. The best results are highlighted in bold font.

Network	IoU (%)					F1 Score				
	Vehicle	Road	Divider	Background	Mean	Vehicle	Road	Divider	Background	Overall
VED [74]	35.5	76.6	29.8	**92.1**	58.5	0.52	0.87	0.46	**0.96**	**0.91**
VPN [75]	37.7	77.8	29.7	91.8	59.2	0.55	0.87	0.46	**0.96**	**0.91**
PON [76]	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.

Given the domain gap between synthetic environments and real-world traffic scenes, the PolarPoint-BEV method on the nuScenes [96] dataset is also assessed. Tab. 5.6 compares the overall prediction performance of polar point and traditional BEV maps using different networks on the nuScenes dataset. In this case, the polar point BEV map is predicted using the XPlan-N network without the C-P module. As seen in Tab. 5.6, both mIoU and overall F1 scores for the polar point BEV map are close to those of the traditional BEV map, indicating comparable performance on the nuScenes dataset.

The mIoU scores across different zones and wIoU values for polar point and traditional BEV maps on nuScenes are shown in Tab. 5.7. Consistent with results from the CARLA simulator, the polar point BEV map outperforms the traditional map in Zones A and B but falls behind in Zone D. The wIoU of the polar point BEV map is higher than that of the traditional BEV map on the nuScenes dataset,

Table 5.7: Comparative results of the prediction performance (mIoU of different zones and the wIoU) and the computational complexity for different networks on the nuScenes dataset. The best results are highlighted in bold font.

Network	Prediction Performance					Complexity		
	Zone A (%)	Zone B (%)	Zone C (%)	Zone D (%)	wIoU (%)	Param	MACs	FPS
VED [74]	53.9	54.6	60.2	67.1	56.0	45.59	159.09	59.61
VPN [75]	54.2	54.8	61.3	69.5	56.6	37.15	43.59	79.11
PON [76]	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.

further validating its superior performance in regions near the ego vehicle. Tab. 5.7 also compares the computational complexity of different networks, revealing that the XPlan-N network (without the C-P module) has significantly lower complexity than other networks, demonstrating that PolarPoint-BEV is an efficient and lightweight solution for describing traffic scenes in real-world environments.

5.3.3 Ablation Study

In the ablation study, we first examine the prediction performance of polar point BEV maps across different configurations, as well as the impact of these configurations on driving performance. We evaluate five different configurations of the polar point BEV map, with the number of semantic points ranging from 16×15 to 16×41. Tab. 5.8 presents the computational complexity of the XPlan networks with these various configurations. The results indicate that all configurations have very similar

Table 5.8: Computational complexity for the XPlan networks with different configurations of the polar point BEV map. The inference speed is tested using an NVIDIA GeForce RTX 3060 GPU.

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

computational costs, and more importantly, they all offer lightweight and efficient methods for describing traffic scenes and explaining control commands.

The left panels of Fig. 5.4 summarize the prediction performance for polar point BEV maps with different configurations. As shown in Fig. 5.4(a), the normal configuration of the polar point BEV map achieves the highest mIoU, approximately 3% higher than the lowest mIoU from the sparse configuration. Similarly, in Fig. 5.4(b), the overall F1 score for the normal configuration is about 3% higher than the lowest score, which is obtained from the light configuration. These findings demonstrate that the prediction performance across different configurations remains consistent, highlighting the robustness and reliability of the proposed PolarPoint-BEV approach.

Tab. 5.9 presents the mIoU values for different zones, along with the wIoU for polar point BEV maps under various configurations. The normal configuration yields

Table 5.9: Ablation study results of the mIoU of different zones and the wIoU for polar point BEV maps with different configurations. The mean and standard deviations are calculated over 3 runs. The best results are highlighted in bold font.

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 highest mIoU across all zones. Specifically, from Zone A to Zone D, the mIoU for the normal configuration is 4%, 5%, 4%, and 7% higher, respectively, than the lowest values. The wIoU for the normal configuration is approximately 4% higher than the lowest wIoU. These results further validate the robustness and reliability of the proposed PolarPoint-BEV model.

The impact of different polar point BEV map configurations on driving performance is also explored. The right panels of Fig. 5.4 display the driving performance of XPlan networks under various configurations. Each network was tested three times. As seen in Fig. 5.4(c), the XPlan-L network achieves the highest driving score and infraction score, with a driving score that is about 13.5% higher than the lowest score from the XPlan-T network. The infraction score of XPlan-L is 6.7% higher than the lowest value from the XPlan-D network. In terms of route completion, the XPlan-N network performs best, with a score approximately 5.7% higher than that of the

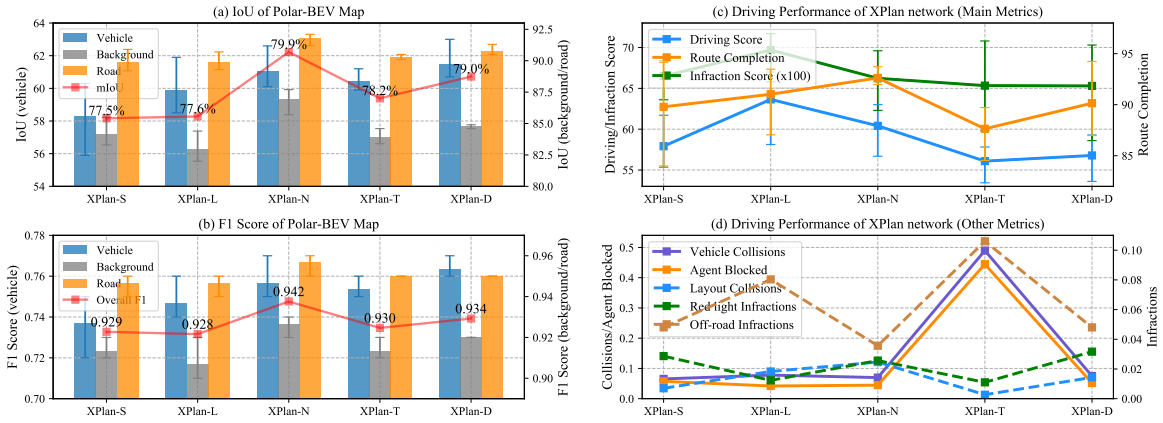


Figure 5.4: Ablation study results of the prediction performance of XPlan networks with different configurations of polar point BEV map. (a) and (b) show the prediction performance of the polar point BEV maps with different configurations. (c) and (d) show the driving performance of the XPlan networks with different configurations of the polar point BEV map.

XPlan-T network.

As previously noted, the parameters of the C-P module are not fixed during the second stage of training. To further analyze the effect of fixing the C-P module parameters on driving performance, a comparative experiment between the XPlan-N network with the C-P module fixed and the XPlan-N network with the C-P module unfixed is conducted. Both networks were trained for 60 epochs using the same pre-trained weights for the C-P module. Tab. 5.10 shows that the driving performance of the XPlan-N network with unfixed C-P module parameters is superior to that of the fixed version. These results suggest that leaving the C-P module parameters unfixed during the second stage of training can improve the driving performance of the proposed network.

Table 5.10: Ablation study results of the driving performance for the XPlan-N network with different configurations of the C-P module. The mean and standard deviations are calculated over 3 runs.

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

5.3.4 Limitations

Despite the strengths of the PolarPoint-BEV method and the XPlan network, there are still a few limitations. First, the current polar point BEV map is limited to only three classes. Expanding it to include additional categories, such as road dividers, road signs, pedestrians, and other relevant objects, would provide a more comprehensive representation of the traffic scene. This enhanced level of detail could further improve the explainability of end-to-end autonomous driving systems. Additionally, the fixed interval between layers in the polar point BEV map may not be ideal, especially when handling complex corner cases involving objects of varying sizes and shapes. To address this, experimenting with different layer intervals could help optimize the polar point BEV map configuration for better performance across diverse scenarios.

5.4 Summary

In this work, we introduced PolarPoint-BEV, a novel BEV perception method designed to overcome the limitations of traditional BEV approaches in explainable end-to-end autonomous driving. Observing that regions close to the ego vehicle are typically more critical for safety, and that the fine-grained, pixel-level detail of traditional BEV maps may be excessive, we developed the polar point BEV map. This method uses a sequence of semantic points distributed around the ego vehicle to represent the traffic scene. To assess the impact of the polar point BEV map on driving performance in an end-to-end system, we designed an end-to-end multi-task explainable network that jointly predicts control commands and polar point BEV maps. Experimental results demonstrate that incorporating PolarPoint-BEV not only enhances driving performance but also improves the explainability of the network.

Chapter 6

Conclusion and Suggestions for Future Work

This thesis aims to address the issue of the lack of explainability in end-to-end autonomous driving. Through a series of innovative approaches, it seeks to enhance the explainability of end-to-end autonomous driving, making it more transparent and understandable to humans. The core contributions of this research include the proposal of a novel network for generating natural-language explanations, the introduction of multimodal explanations that integrate both natural-language and visual explanations, and the development of an advanced BEV perception technique that enhances safety while maintaining computational efficiency. The main work, research results, and innovations of this thesis are summarized below:

To enhance the explainability of end-to-end autonomous driving systems by us-

ing natural-language explanations, we proposed a new deep neural network architecture that not only predicts driving actions but also generates natural-language explanations based on semantic scene understanding. This approach aims to bridge the gap between autonomous decision-making and human understanding by providing explanations behind driving action. Specifically, two types of natural-language explanations are generated: reasons for each driving action and descriptions of the surrounding environment relevant to the ego-vehicle. To train and test the proposed network, a large-scale dataset consisting of $10,000$ images from the BDD-OIA dataset have been labeled with 4 distinct driving actions and 6 environment descriptions. Additionally, to further validate the network's prediction accuracy and assess its ability to generalize to diverse driving environments, we have selected a subset of $1,500$ frames from the nuScenes dataset, labeled with 4 driving actions and corresponding natural-language explanations (covering 21 reasons and 6 descriptions). Experimental results on both publicly available dataset and our annotated datasets demonstrate the model's effectiveness, showing that the proposed architecture enhances the accuracy of decision-making predictions and significantly improves the explainability of the network.

While natural-language explanations provide a useful means for explaining driving decisions, they often lack insights into the internal processes of end-to-end networks. To address this gap and further enhance explainability, we proposed combining natural-language and visual explanations to better interpret the outputs of

autonomous driving models. Specifically, we introduce an explainable end-to-end network that simultaneously predicts driving actions and generates multimodal descriptions of traffic scenes, incorporating both semantic BEV maps and natural-language environment descriptions. In this network, context information from BEV perception and local information from semantic segmentation are integrated before predicting driving decisions and describing the environment. This design aims to minimize error propagation and boost prediction accuracy. To support the training and validation of the proposed approach, we present a dataset of $12,000$ image sequences, each containing images from multiple cameras and manually annotated with ground truth labels for driving actions and multimodal environment descriptions. Experimental findings reveal that combining context and local information enhances the accuracy of both driving action predictions and environment descriptions, promoting safer and more explainable autonomous driving systems.

At last, we introduced an innovative BEV perception approach, PolarPoint-BEV, designed to improve the explainability of end-to-end autonomous driving. In contrast to traditional BEV methods, the proposed approach focuses on areas close to the ego vehicle, which are crucial for safe driving. Additionally, PolarPoint-BEV employs a sparse representation, significantly reducing computational demands compared to traditional dense BEV methods. This lightweight structure makes it a practical choice for vehicles with limited processing resources. To evaluate the effectiveness of PolarPoint-BEV in enhancing explainability and driving performance, we present an end-to-end

autonomous driving network, named XPlan. XPlan uses a multi-task framework that simultaneously predicts control commands and generates polar point BEV maps as visual explanations. We validate the network's performance within the CARLA simulator, with experimental results indicating that PolarPoint-BEV enhances both driving performance and the explainability of the end-to-end driving network.

Regarding the suggestions for future work, several challenging and meaningful directions deserve investigation to enhance the explainability of end-to-end autonomous driving systems. One promising area involves the integration of LLMs into autonomous driving systems. As LLMs continue to advance, their potential to improve both the generalization capacity and explainability of these systems is becoming increasingly evident. By leveraging common-sense reasoning, LLMs may enable autonomous driving systems to better manage unexpected or anomalous situations, potentially enhancing the safety and reliability of autonomous vehicles in real-world environments. For example, LLMs can use reasoning to anticipate complex, unplanned scenarios, such as a pedestrian unexpectedly crossing the street or an unusual obstacle in the roadway. This capacity to reason through unconventional situations offers a promising approach to augmenting the decision-making process of autonomous vehicles, making them safer and more adaptable in dynamic environments. Moreover, LLMs can generate comprehensive, text-rich explanations for driving decisions, addressing the "black box" nature of end-to-end networks. This capability could be crucial in promoting transparency and building user trust. For

instance, an LLM might provide detailed insights into why an autonomous system chose a particular path, slowed down, or avoided certain objects. By translating model outputs into natural language, LLMs can deliver more user-friendly explanations that clarify complex model behavior in a manner understandable to passengers, engineers, and regulators alike.

However, several challenges need to be addressed to effectively integrate LLMs into autonomous driving systems. A primary challenge is the "hallucination" problem, where LLMs may generate plausible but factually incorrect information. This issue could be critical in autonomous driving, as inaccurate information could lead to misunderstandings about the vehicle's environment, potentially resulting in errors in perception or decision-making. For instance, if an LLM incorrectly describes an object or event, the system's response could jeopardize safety. Another significant challenge is that most LLMs are designed to work primarily with text data and cannot process raw sensor data, such as images, LiDAR, or radar, directly. To fully realize the potential of LLMs in autonomous driving, it would be necessary to create cross-modal methods capable of accurately translating visual and spatial information from sensor data into natural language representations. Developing such cross-modal systems would involve significant research and could require advanced techniques for bridging the gap between diverse data types, potentially combining computer vision, sensor fusion, and language processing to allow for seamless integration. In addition, the computational demands of LLMs present a notable hurdle. LLMs are inherently

resource-intensive, posing a challenge given the limited computational resources typically available in autonomous vehicles. Achieving efficient deployment of LLMs on autonomous vehicle platforms is a technical obstacle that must be overcome. Techniques such as model pruning, quantization, or the development of more efficient LLM architectures will be essential for balancing real-time performance requirements with the computational constraints of edge hardware. These approaches could help reduce the memory cost and processing power needed for LLMs, making them more viable for autonomous driving applications without sacrificing the speed and reliability necessary for safety-critical decisions. In summary, while LLMs hold promising potential to advance the explainability and adaptability of autonomous driving systems, substantial research is required to address challenges like hallucination, cross-modal integration, and computational efficiency. Addressing these challenges could lead to more explainable, trustworthy, and robust autonomous driving systems capable of managing the complex realities of real-world driving.

References

[1] Mariusz Bojarski. End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316*, 2016.

[2] Daniel Coelho and Miguel Oliveira. A review of end-to-end autonomous driving in urban environments. *Ieee Access*, 10:75296–75311, 2022.

[3] Yi Xiao, Felipe Codevilla, Akhil Gurram, Onay Urfalioglu, and Antonio M López. Multimodal end-to-end autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, 23(1):537–547, 2020.

[4] Aditya Prakash, Kashyap Chitta, and Andreas Geiger. Multi-modal fusion transformer for end-to-end autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7077–7087, June 2021.

[5] Pranav Singh Chib and Pravendra Singh. Recent advancements in end-to-end autonomous driving using deep learning: A survey. *IEEE Transactions on Intelligent Vehicles*, 2023.

[6] Dian Chen and Philipp Krähenbühl. Learning from all vehicles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17222–17231, 2022.

[7] Penghao Wu, Xiaosong Jia, Li Chen, Junchi Yan, Hongyang Li, and Yu Qiao. Trajectory-guided control prediction for end-to-end autonomous driving: A simple yet strong baseline. *Advances in Neural Information Processing Systems*, 35:6119–6132, 2022.

[8] Long Chen, Yuchen Li, Chao Huang, Bai Li, Yang Xing, Daxin Tian, Li Li, Zhongxu Hu, Xiaoxiang Na, Zixuan Li, et al. Milestones in autonomous driving and intelligent vehicles: Survey of surveys. *IEEE Transactions on Intelligent Vehicles*, 8(2):1046–1056, 2022.

[9] Oskar Natan and Jun Miura. End-to-end autonomous driving with semantic depth cloud mapping and multi-agent. *IEEE Transactions on Intelligent Vehicles*, 8(1):557–571, 2022.

[10] Kashyap Chitta, Aditya Prakash, Bernhard Jaeger, Zehao Yu, Katrin Renz, and Andreas Geiger. Transfuser: Imitation with transformer-based sensor fusion for autonomous driving. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(11):12878–12895, 2023.

[11] Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, et al. Planning-oriented au-

tonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17853–17862, 2023.

[12] Li Chen, Penghao Wu, Kashyap Chitta, Bernhard Jaeger, Andreas Geiger, and Hongyang Li. End-to-end autonomous driving: Challenges and frontiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[13] Shahin Atakishiyev, Mohammad Salameh, Hengshuai Yao, and Randy Goebel. Explainable artificial intelligence for autonomous driving: A comprehensive overview and field guide for future research directions. *IEEE Access*, 12:101603–101625, 2024.

[14] Éloi Zablocki, Hédi Ben-Younes, Patrick Pérez, and Matthieu Cord. Explainability of deep vision-based autonomous driving systems: Review and challenges. *International Journal of Computer Vision*, 130(10):2425–2452, 2022.

[15] Jiqian Dong, Sikai Chen, Mohammad Miralinaghi, Tiantian Chen, Pei Li, and Samuel Labi. Why did the ai make that decision? towards an explainable artificial intelligence (xai) for autonomous driving systems. *Transportation research part C: emerging technologies*, 156:104358, 2023.

[16] Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador García, Sergio Gil-López, Daniel Molina, Richard Benjamins, et al. Explainable artificial intelligence

(xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information fusion*, 58:82–115, 2020.

[17] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. A survey of methods for explaining black box models. *ACM Comput. Surv.*, 51(5), August 2018.

[18] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[19] Jeffrey L Elman. Finding structure in time. *Cognitive science*, 14(2):179–211, 1990.

[20] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.

[21] Mariusz Bojarski, Philip Yeres, Anna Choromanska, Krzysztof Choromanski, Bernhard Firner, Lawrence Jackel, and Urs Muller. Explaining how a deep neural network trained with end-to-end learning steers a car. *arXiv preprint arXiv:1704.07911*, 2017.

[22] Jinkyu Kim and John Canny. Interpretable learning for self-driving cars by visualizing causal attention. In *Proceedings of the IEEE international conference on computer vision*, pages 2942–2950, 2017.

[23] Jianyu Chen, Shengbo Eben Li, and Masayoshi Tomizuka. Interpretable end-to-end urban autonomous driving with latent deep reinforcement learning. *IEEE Transactions on Intelligent Transportation Systems*, 23(6):5068–5078, Jun. 2022.

[24] Hengli Wang, Peide Cai, Yuxiang Sun, Lujia Wang, and Ming Liu. 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)*, pages 13731–13737, 2021.

[25] Siyu Teng, Long Chen, Yunfeng Ai, Yuanye Zhou, Zhe Xuanyuan, and Xuemin Hu. Hierarchical interpretable imitation learning for end-to-end autonomous driving. *IEEE Transactions on Intelligent Vehicles*, 8(1):673–683, Jan. 2023.

[26] Katrin Renz, Kashyap Chitta, Otniel-Bogdan Mercea, A. Sophia Koepke, Zeynep Akata, and Andreas Geiger. Plant: Explainable planning transformers via object-level representations. In *Proceedings of Conference on Robotic Learning (CoRL)*, pages 459–470, 2022.

[27] Shengchao Hu, Li Chen, Penghao Wu, Hongyang Li, Junchi Yan, and Dacheng Tao. St-p3: End-to-end vision-based autonomous driving via spatial-temporal feature learning. In *European Conference on Computer Vision*, pages 533–549. Springer, 2022.

[28] Jianyu Chen, Shengbo Eben Li, and Masayoshi Tomizuka. Interpretable end-to-end urban autonomous driving with latent deep reinforcement learning. *IEEE Transactions on Intelligent Transportation Systems*, 23(6):5068–5078, 2022.

[29] Yiran Xu, Xiaoyin Yang, Lihang Gong, Hsuan-Chu Lin, Tz-Ying Wu, Yunsheng Li, and Nuno Vasconcelos. Explainable object-induced action decision for autonomous vehicles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9523–9532, 2020.

[30] Jinkyu Kim, Anna Rohrbach, Trevor Darrell, John Canny, and Zeynep Akata. Textual explanations for self-driving vehicles. In *European Conference on Computer Vision*, pages 563–578, 2018.

[31] Jiqian Dong, Sikai Chen, Shuya Zong, Tiantian Chen, and Samuel Labi. Image transformer for explainable autonomous driving system. In *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, pages 2732–2737. IEEE, 2021.

[32] Yoshihide Sawada and Keigo Nakamura. Concept bottleneck model with additional unsupervised concepts. *IEEE Access*, 10:41758–41765, 2022.

[33] Yoshihide Sawada and Keigo Nakamura. C-senn: Contrastive self-explaining neural network. *arXiv preprint arXiv:2206.09575*, 2022.

[34] Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. In *International Conference on Machine Learning*, pages 5338–5348. PMLR, 2020.

[35] Zhengming Zhang, Renran Tian, Rini Sherony, Joshua Domeyer, and Zhengming Ding. Attention-based interrelation modeling for explainable automated driving. *IEEE Transactions on Intelligent Vehicles*, 8(2):1564–1573, 2023.

[36] Hédi Ben-Younes, Éloi Zablocki, Patrick Pérez, and Matthieu Cord. Driving behavior explanation with multi-level fusion. *Pattern Recognition*, 123:108421, 2022.

[37] Paul Jacob, Éloi Zablocki, Hedi Ben-Younes, Mickaël Chen, Patrick Pérez, and Matthieu Cord. Steex: steering counterfactual explanations with semantics. In *European Conference on Computer Vision*, pages 387–403. Springer, 2022.

[38] Hao Sha, Yao Mu, Yuxuan Jiang, Li Chen, Chenfeng Xu, Ping Luo, Shengbo Eben Li, Masayoshi Tomizuka, Wei Zhan, and Mingyu Ding. Languagempc: Large language models as decision makers for autonomous driving. *arXiv preprint arXiv:2310.03026*, 2023.

[39] Yuchao Feng, Wei Hua, and Yuxiang Sun. Nle-dm: Natural-language explanations for decision making of autonomous driving based on semantic scene understanding. *IEEE Transactions on Intelligent Transportation Systems*, 24(9):9780–9791, 2023.

[40] Yuchao Feng, Zhen Feng, Wei Hua, and Yuxiang Sun. Multimodal-xad: Explainable autonomous driving based on multimodal environment descriptions. *IEEE Transactions on Intelligent Transportation Systems*, 25(12):19469–19481, 2024.

[41] Yuchao Feng and Yuxiang Sun. Polarpoint-bev: Bird-eye-view perception in polar points for explainable end-to-end autonomous driving. *IEEE Transactions on Intelligent Vehicles*, early acess, 2024.

[42] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. Carla: An open urban driving simulator. In *Conference on robot learning*, pages 1–16. PMLR, 2017.

[43] David Gunning, Mark Stefik, Jaesik Choi, Timothy Miller, Simone Stumpf, and Guang-Zhong Yang. Xai—explainable artificial intelligence. *Science Robotics*, 4(37):eaay7120, 2019.

[44] Been Kim, Cynthia Rudin, and Julie A Shah. The bayesian case model: A generative approach for case-based reasoning and prototype classification. *Advances in neural information processing systems*, 27, 2014.

[45] Been Kim, Rajiv Khanna, and Oluwasanmi O Koyejo. Examples are not enough, learn to criticize! criticism for interpretability. *Advances in neural information processing systems*, 29, 2016.

[46] David Baehrens, Timon Schroeter, Stefan Harmeling, Motoaki Kawanabe, Katja Hansen, and Klaus-Robert Müller. How to explain individual classification decisions. *The Journal of Machine Learning Research*, 11:1803–1831, 2010.

[47] Marko Robnik-Šikonja and Igor Kononenko. Explaining classifications for individual instances. *IEEE Transactions on Knowledge and Data Engineering*, 20(5):589–600, 2008.

[48] Quanshi Zhang, Yu Yang, Haotian Ma, and Ying Nian Wu. Interpreting cnns via decision trees. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6261–6270, 2019.

[49] Anh Nguyen, Alexey Dosovitskiy, Jason Yosinski, Thomas Brox, and Jeff Clune. Synthesizing the preferred inputs for neurons in neural networks via deep generator networks. *Advances in neural information processing systems*, 29, 2016.

[50] Sebastian Bach, Alexander Binder, Klaus-Robert Müller, and Wojciech Samek. Controlling explanatory heatmap resolution and semantics via decomposition depth. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 2271–2275. IEEE, 2016.

[51] Aravindh Mahendran and Andrea Vedaldi. Understanding deep image representations by inverting them. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5188–5196, 2015.

[52] Matthew D Zeiler, Dilip Krishnan, Graham W Taylor, and Rob Fergus. Deconvolutional networks. In *2010 IEEE Computer Society Conference on computer vision and pattern recognition*, pages 2528–2535. IEEE, 2010.

[53] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2625–2634, 2015.

[54] Lisa Anne Hendricks, Zeynep Akata, Marcus Rohrbach, Jeff Donahue, Bernt Schiele, and Trevor Darrell. Generating visual explanations. In *European conference on computer vision*, pages 3–19. Springer, 2016.

[55] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016.

[56] Fei Wang, Mengqing Jiang, Chen Qian, Shuo Yang, Cheng Li, Honggang Zhang, Xiaogang Wang, and Xiaoou Tang. Residual attention network for image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3156–3164, 2017.

[57] Quanshi Zhang, Ying Nian Wu, and Song-Chun Zhu. Interpretable convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8827–8836, 2018.

[58] Zachary C Lipton. The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, 16(3):31–57, 2018.

[59] Saumitra Mishra, Bob L Sturm, and Simon Dixon. Local interpretable model-agnostic explanations for music content analysis. In *ISMIR*, volume 53, pages 537–543, 2017.

[60] Liat Sless, Bat El Shlomo, Gilad Cohen, and Shaul Oron. Road scene understanding by occupancy grid learning from sparse radar clusters using semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[61] Bin Yang, Runsheng Guo, Ming Liang, Sergio Casas, and Raquel Urtasun. Radarnet: Exploiting radar for robust perception of dynamic objects. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII 16*, pages 496–512. Springer, 2020.

[62] Yue Wang, Alireza Fathi, Abhijit Kundu, David A Ross, Caroline Pantofaru, Tom Funkhouser, and Justin Solomon. Pillar-based object detection for autonomous driving. In *Computer Vision–ECCV 2020: 16th European Confer-*

ence, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXII 16, pages 18–34. Springer, 2020.

[63] Simon T Isele, Fabian Klein, Mathis Brosowsky, and J Marius Zöllner. Learning semantics on radar point-clouds. In *2021 IEEE Intelligent Vehicles Symposium (IV)*, pages 810–817. IEEE, 2021.

[64] Raphael Van Kempen, Bastian Lampe, Timo Woopen, and Lutz Eckstein. A simulation-based end-to-end learning framework for evidential occupancy grid mapping. In *2021 IEEE Intelligent Vehicles Symposium (IV)*, pages 934–939. IEEE, 2021.

[65] Jinyu Li, Chenxu Luo, and Xiaodong Yang. Pillarnext: Rethinking network designs for 3d object detection in lidar point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17567–17576, 2023.

[66] Youngseok Kim and Dongsuk Kum. Deep learning based vehicle position and orientation estimation via inverse perspective mapping image. In *2019 IEEE Intelligent Vehicles Symposium (IV)*, pages 317–323. IEEE, 2019.

[67] 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), pages 1–7. IEEE, 2020.

[68] 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*, pages 11138–11147, 2020.

[69] Shuang Gao, Qiang Wang, and Yuxiang Sun. S2g2: Semi-supervised semantic bird-eye-view grid-map generation using a monocular camera for autonomous driving. *IEEE Robotics and Automation Letters*, 7(4):11974–11981, 2022.

[70] Jonah Philion and Sanja Fidler. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *European Conference on Computer Vision*, pages 194–210. Springer, 2020.

[71] Bowen Pan, Jiankai Sun, Ho Yin Tiga Leung, Alex Andonian, and Bolei Zhou. Cross-view semantic segmentation for sensing surroundings. *IEEE Robotics and Automation Letters*, 5(3):4867–4873, 2020.

[72] Noureldin Hendy, Cooper Sloan, Feng Tian, Pengfei Duan, Nick Charchut, Yuesong Xie, Chuang Wang, and James Philbin. Fishing net: Future inference of semantic heatmaps in grids. *arXiv preprint arXiv:2006.09917*, 2020.

[73] Anthony Hu, Zak Murez, Nikhil Mohan, Sofía Dudas, Jeffrey Hawke, Vijay Badrinarayanan, Roberto Cipolla, and Alex Kendall. Fiery: Future instance

prediction in bird's-eye view from surround monocular cameras. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15273–15282, 2021.

[74] Chenyang Lu, Marinus Jacobus Gerardus van de Molengraft, and Gijs Dubbelman. Monocular semantic occupancy grid mapping with convolutional variational encoder–decoder networks. *IEEE Robotics and Automation Letters*, 4(2):445–452, Apr. 2019.

[75] Bowen Pan, Jiankai Sun, Ho Yin Tiga Leung, Alex Andonian, and Bolei Zhou. Cross-view semantic segmentation for sensing surroundings. *IEEE Robotics and Automation Letters*, 5(3):4867–4873, Jul. 2020.

[76] 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*, Seattle, WA, USA, 2020, pp.11138-11147.

[77] Junjie Huang and Guan Huang. Bevdet4d: Exploit temporal cues in multi-camera 3d object detection. *arXiv preprint arXiv:2203.17054*, 2022.

[78] 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*, pages 13760–13769, 2022.

[79] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Yu Qiao, and Jifeng Dai. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. In *European conference on computer vision*, pages 1–18. Springer, 2022.

[80] Yue Wang, Vitor Campagnolo Guizilini, Tianyuan Zhang, Yilun Wang, Hang Zhao, and Justin Solomon. Detr3d: 3d object detection from multi-view images via 3d-to-2d queries. In *Conference on Robot Learning*, pages 180–191. PMLR, 2022.

[81] Yingfei Liu, Tiancai Wang, Xiangyu Zhang, and Jian Sun. Petr: Position embedding transformation for multi-view 3d object detection. In *European Conference on Computer Vision*, pages 531–548. Springer, 2022.

[82] Yingfei Liu, Junjie Yan, Fan Jia, Shuailin Li, Aqi Gao, Tiancai Wang, and Xiangyu Zhang. Petrv2: A unified framework for 3d perception from multi-camera images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3262–3272, October 2023.

[83] Siran Chen, Yue Ma, Yu Qiao, and Yali Wang. M-bev: Masked bev perception for robust autonomous driving. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 1183–1191, 2024.

[84] Zhi Liu, Shaoyu Chen, Xiaojie Guo, Xinggang Wang, Tianheng Cheng, Hongmei Zhu, Qian Zhang, Wenyu Liu, and Yi Zhang. Vision-based uneven bev

representation learning with polar rasterization and surface estimation. In *Conference on Robot Learning*, pages 437–446. PMLR, 2023.

[85] Yanqin Jiang, Li Zhang, Zhenwei Miao, Xiatian Zhu, Jin Gao, Weiming Hu, and Yu-Gang Jiang. Polarformer: Multi-camera 3d object detection with polar transformer. In *Proceedings of the AAAI conference on Artificial Intelligence*, volume 37, pages 1042–1050, 2023.

[86] Huitong Yang, Xuyang Bai, Xinge Zhu, and Yuexin Ma. One training for multiple deployments: Polar-based adaptive bev perception for autonomous driving. 2023, *arXiv:2304.00525*.

[87] Apoorv Singh. Vision-radar fusion for robotics bev detections: A survey. In *2023 IEEE Intelligent Vehicles Symposium (IV)*, pages 1–7. IEEE, 2023.

[88] Tingting Liang, Hongwei Xie, Kaicheng Yu, Zhongyu Xia, Zhiwei Lin, Yongtao Wang, Tao Tang, Bing Wang, and Zhi Tang. Bevfusion: A simple and robust lidar-camera fusion framework. *Advances in Neural Information Processing Systems*, 35:10421–10434, 2022.

[89] Zhijian Liu, Haotian Tang, Alexander Amini, Xinyu Yang, Huizi Mao, Daniela L Rus, and Song Han. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In *2023 IEEE international conference on robotics and automation (ICRA)*, pages 2774–2781. IEEE, 2023.

[90] Yunze Man, Liang-Yan Gui, and Yu-Xiong Wang. Bev-guided multi-modality fusion for driving perception. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21960–21969, 2023.

[91] Adam W Harley, Zhaoyuan Fang, Jie Li, Rares Ambrus, and Katerina Fragkiadaki. Simple-bev: What really matters for multi-sensor bev perception? In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2759–2765. IEEE, 2023.

[92] Yuexin Ma, Tai Wang, Xuyang Bai, Huitong Yang, Yuenan Hou, Yaming Wang, Yu Qiao, Ruigang Yang, Dinesh Manocha, and Xinge Zhu. Vision-centric bev perception: A survey. *arXiv preprint arXiv:2208.02797*, 2022.

[93] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3354–3361. IEEE, 2012.

[94] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016.

[95] Xinyu Huang, Peng Wang, Xinjing Cheng, Dingfu Zhou, Qichuan Geng, and Ruigang Yang. The apolloscape open dataset for autonomous driving and its

application. *IEEE transactions on pattern analysis and machine intelligence*, 42(10):2702–2719, 2019.

[96] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020.

[97] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2636–2645, 2020.

[98] Will Maddern, Geoffrey Pascoe, Chris Linegar, and Paul Newman. 1 year, 1000 km: The oxford robotcar dataset. *The International Journal of Robotics Research*, 36(1):3–15, 2017.

[99] Guodong Rong, Byung Hyun Shin, Hadi Tabatabaee, Qiang Lu, Steve Lemke, Mārtiņš Možeiko, Eric Boise, Geehoon Uhm, Mark Gerow, Shalin Mehta, et al. Lgsvl simulator: A high fidelity simulator for autonomous driving. In *2020 IEEE 23rd International conference on intelligent transportation systems (ITSC)*, pages 1–6. IEEE, 2020.

[100] Zhiyao Xie, Xiaoqing Xu, Matt Walker, Joshua Knebel, Kumaraguru Palaniswamy, Nicolas Hebert, Jiang Hu, Huanrui Yang, Yiran Chen, and Shidhartha Das. Apollo: An automated power modeling framework for runtime power introspection in high-volume commercial microprocessors. In *MICRO-54: 54th Annual IEEE/ACM International Symposium on Microarchitecture*, pages 1–14, 2021.

[101] Alexander Amini, Tsun-Hsuan Wang, Igor Gilitschenski, Wilko Schwarting, Zhijian Liu, Song Han, Sertac Karaman, and Daniela Rus. Vista 2.0: An open, data-driven simulator for multimodal sensing and policy learning for autonomous vehicles. In *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022.

[102] Tsun-Hsuan Wang, Alexander Amini, Wilko Schwarting, Igor Gilitschenski, Sertac Karaman, and Daniela Rus. Learning interactive driving policies via data-driven simulation. In *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022.

[103] Xiaofeng Wang, Zheng Zhu, Guan Huang, Xinze Chen, Jiagang Zhu, and Jiwen Lu. Drivedreamer: Towards real-world-driven world models for autonomous driving. *arXiv preprint arXiv:2309.09777*, 2023.

[104] Yanchen Guan, Haicheng Liao, Zhenning Li, Guohui Zhang, and Chengzhong Xu. World models for autonomous driving: An initial survey. *arXiv preprint*

arXiv:2403.02622, 2024.

[105] Vasili Ramanishka, Yi-Ting Chen, Teruhisa Misu, and Kate Saenko. Toward driving scene understanding: A dataset for learning driver behavior and causal reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7699–7707, 2018.

[106] Tina Chen, Taotao Jing, Renran Tian, Yaobin Chen, Joshua Domeyer, Heishiro Toyoda, Rini Sherony, and Zhengming Ding. Psi: A pedestrian behavior dataset for socially intelligent autonomous car. *arXiv preprint arXiv:2112.02604*, 2021.

[107] Santokh Singh. Critical reasons for crashes investigated in the national motor vehicle crash causation survey. Technical report, 2015.

[108] Long Chen, Shaobo Lin, Xiankai Lu, Dongpu Cao, Hangbin Wu, Chi Guo, Chun Liu, and Fei-Yue Wang. Deep neural network based vehicle and pedestrian detection for autonomous driving: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 22(6):3234–3246, 2021.

[109] Zhen Feng, Yanning Guo, Qing Liang, M. Usman Maqbool Bhutta, Hengli Wang, Ming Liu, and Yuxiang Sun. Mafnet: Segmentation of road potholes with multi-modal attention fusion network for autonomous vehicles. *IEEE Transactions on Instrumentation and Measurement*, pages 1–1, 2022.

[110] Hai Wang, Yanyan Chen, Yingfeng Cai, Long Chen, Yicheng Li, Miguel Angel Sotelo, and Zhixiong Li. Sfnet-n: An improved sfnet algorithm for semantic

segmentation of low-light autonomous driving road scenes. *IEEE Transactions on Intelligent Transportation Systems*, 2022.

[111] M. Usman Maqbool Bhutta, Yuxiang Sun, Darwin Lau, and Ming Liu. Why-so-deep: Towards boosting previously trained models for visual place recognition. *IEEE Robotics and Automation Letters*, 7(2):1824–1831, 2022.

[112] Peide Cai, Yuxiang Sun, Hengli Wang, and Ming Liu. Vtgnet: A vision-based trajectory generation network for autonomous vehicles in urban environments. *IEEE Transactions on Intelligent Vehicles*, 6(3):419–429, 2020.

[113] Bai Li, Yakun Ouyang, Li Li, and Youmin Zhang. Autonomous driving on curvy roads without reliance on frenet frame: A cartesian-based trajectory planning method. *IEEE Transactions on Intelligent Transportation Systems*, 2022.

[114] Yuxiang Sun, Weixun Zuo, and Ming Liu. See the future: A semantic segmentation network predicting ego-vehicle trajectory with a single monocular camera. *IEEE Robotics and Automation Letters*, 5(2):3066–3073, 2020.

[115] Zihao Sheng, Yunwen Xu, Shibei Xue, and Dewei Li. Graph-based spatial-temporal convolutional network for vehicle trajectory prediction in autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, 2022.

[116] Peide Cai, Xiaodong Mei, Lei Tai, Yuxiang Sun, and Ming Liu. High-speed autonomous drifting with deep reinforcement learning. *IEEE Robotics and Automation Letters*, 5(2):1247–1254, 2020.

[117] Mohamed A Daoud, Mohamed W Mehrez, Derek Rayside, and William W Melek. Simultaneous feasible local planning and path-following control for autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, 2022.

[118] Peng Hang, Chen Lv, Yang Xing, Chao Huang, and Zhongxu Hu. Human-like decision making for autonomous driving: A noncooperative game theoretic approach. *IEEE Transactions on Intelligent Transportation Systems*, 22(4):2076–2087, 2020.

[119] Qi Liu, Xueyuan Li, Shihua Yuan, and Zirui Li. Decision-making technology for autonomous vehicles: Learning-based methods, applications and future outlook. In *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, pages 30–37. IEEE, 2021.

[120] Florin Leon and Marius Gavrilescu. A review of tracking and trajectory prediction methods for autonomous driving. *Mathematics*, 9(6):660, 2021.

[121] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *CoRR*, abs/1706.05587, 2017.

[122] Dequan Wang, Coline Devin, Qi-Zhi Cai, Fisher Yu, and Trevor Darrell. Deep object-centric policies for autonomous driving. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8853–8859. IEEE, 2019.

[123] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[124] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for mobilenetv3. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1314–1324, 2019.

[125] Siyu Teng, Xuemin Hu, Peng Deng, Bai Li, Yuchen Li, Yunfeng Ai, Dongsheng Yang, Lingxi Li, Zhe Xuanyuan, Fenghua Zhu, et al. Motion planning for autonomous driving: The state of the art and future perspectives. *IEEE Transactions on Intelligent Vehicles*, 2023.

[126] Peide Cai, Sukai Wang, Yuxiang Sun, and Ming Liu. Probabilistic end-to-end vehicle navigation in complex dynamic environments with multimodal sensor fusion. *IEEE Robotics and Automation Letters*, 5(3):4218–4224, 2020.

[127] Hengli Wang, Peide Cai, Rui Fan, Yuxiang Sun, and Ming Liu. 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*, pages 2229–2238, June 2021.

[128] Yuxiang Sun, Weixun Zuo, Huaiyang Huang, Peide Cai, and Ming Liu. Pointmoseg: Sparse tensor-based end-to-end moving-obstacle segmentation in 3-d

lidar point clouds for autonomous driving. *IEEE Robotics and Automation Letters*, 6(2):510–517, 2021.

[129] Peide Cai, Hengli Wang, Yuxiang Sun, and Ming Liu. Dq-gat: Towards safe and efficient autonomous driving with deep q-learning and graph attention networks. *IEEE Transactions on Intelligent Transportation Systems*, 23(11):21102–21112, 2022.

[130] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

[131] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proc. IEEE Int. Conf. Comput. Vis.*, pages 618–626, 2017.

[132] Aditya Chattopadhay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *IEEE Winter Conf. Appl. Comput. Vis.*, pages 839–847. IEEE, 2018.

[133] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019.

[134] Omar Elharrouss, Younes Akbari, Noor Almaadeed, and Somaya Al-Maadeed. Backbones-review: Feature extraction networks for deep learning and deep reinforcement learning approaches. *arXiv preprint arXiv:2206.08016*, 2022.

[135] Weixin Ma, Shoudong Huang, and Yuxiang Sun. Triplet-graph: Global metric localization based on semantic triplet graph for autonomous vehicles. *IEEE Robotics and Automation Letters*, 9(4):3155–3162, 2024.

[136] Stephanie Lowry, Niko Sünderhauf, Paul Newman, John J Leonard, David Cox, Peter Corke, and Michael J Milford. Visual place recognition: A survey. *ieee transactions on robotics*, 32(1):1–19, 2015.

[137] Zhen Feng, Yanning Guo, and Yuxiang Sun. Cekd: Cross-modal edge-privileged knowledge distillation for semantic scene understanding using only thermal images. *IEEE Robotics and Automation Letters*, 8(4):2205–2212, 2023.

[138] Fangyin Tian, Zhiheng Li, Fei-Yue Wang, and Li Li. Parallel learning-based steering control for autonomous driving. *IEEE Transactions on Intelligent Vehicles*, 8(1):379–389, 2022.

[139] Richa Nahata, Daniel Omeiza, Rhys Howard, and Lars Kunze. Assessing and explaining collision risk in dynamic environments for autonomous driving safety. In *2021 IEEE international intelligent transportation systems conference (ITSC)*, pages 223–230. IEEE, 2021.

[140] Ayoosh Bansal, Jayati Singh, Micaela Verucchi, Marco Caccamo, and Lui Sha. Risk ranked recall: Collision safety metric for object detection systems in autonomous vehicles. In *2021 10th Mediterranean Conference on Embedded Computing (MECO)*, pages 1–4. IEEE, 2021.

[141] Mingyue Cui, Shipeng Zhong, Boyang Li, Xu Chen, and Kai Huang. Offloading autonomous driving services via edge computing. *IEEE Internet of Things Journal*, 7(10):10535–10547, 2020.

[142] Ruimin Ke, Zhiyong Cui, Yanlong Chen, Meixin Zhu, Hao Yang, Yifan Zhuang, and Yinhai Wang. Lightweight edge intelligence empowered near-crash detection towards real-time vehicle event logging. *IEEE Transactions on Intelligent Vehicles*, 8(4):2737–2747, 2023.

[143] Zhejun Zhang, Alexander Liniger, Dengxin Dai, Fisher Yu, and Luc Van Gool. End-to-end urban driving by imitating a reinforcement learning coach. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 15222–15232, 2021.