HDDPNet: Hierarchical Cascade and Dual-Stream Diffusion for Multi-Task Perception in Autonomous Driving

Xiangxiang Zhang, Jun Hu, Wei Liu, Kewen Li, *Member, IEEE*, Shuai Cheng, *Member, IEEE*, Zuotao Ning, Jinghan Xu, and Yuefeng Wang

***Abstract*—Environmental perception is fundamental to the safety and reliability of autonomous driving systems. Despite recent progress, existing frameworks often struggle to achieve multi-granularity fusion of BEV spatial features in complex traffic scenarios. Furthermore, some approaches treat perception tasks separately and do not adequately model inter-task associations. To address these challenges, we propose HDDPNet, a multi-task perception model for autonomous driving, which adopts an encoder-decoder architecture to enhance semantic modeling and cross-task collaboration. In the encoding stage, we introduce a Hierarchical Cascade Encoder (HCE), which incorporates a TransMamba module to enhance global context modeling and feature representation. In addition, a Progressive Feature Pyramid (PFP) module is designed to facilitate multi-scale and fine-grained information fusion. In the decoding stage, we develop an Isomorphic Dual-stream Diffusion Decoder Network (ID^{3}Net), which adopts a diffusion-inspired single-step denoising process to improve feature robustness and promote cross-task collaboration by sharing semantic and geometric information within a dual-stream architecture. Experimental results show that the proposed method achieves significant improvements in multiple perception tasks: NDS is increased by 10.09% in 3D object detection; AMOTA is improved by 7.29% in multi-object tracking; and mAP is increased by 2.17% in online mapping. These results demonstrate that HDDPNet consistently improves multi-task perception performance, providing a reliable solution for autonomous driving.**

***Index Terms*—Autonomous driving, multi-task perception, Mamba structure, diffusion decoder, multi-granularity feature fusion.**

Received 28 January 2026; revised 6 June 2026; accepted 15 August 2026. This work was supported by the Intelligent Control Theories and Key Technologies of Heterogeneous Unmanned Electric Commercial Vehicles Formation under Grant U22A2043. The Associate Editor for this article was Y. Dong. *(Xiangxiang Zhang and Jun Hu are co-first authors.) (Corresponding author: Wei Liu.)*

Xiangxiang Zhang and Jinghan Xu are with the School of Computer Science and Engineering, Northeastern University, Shenyang 110169, China (e-mail: zhangxx4@mails.neu.edu.cn; xujinghan@mails.neu.edu.cn).

Jun Hu is with the School of Software, Shenyang University of Technology, Shenyang 110870, China (e-mail: hujun@sut.edu.cn).

Wei Liu is with the School of Computer Science and Engineering, Northeastern University, Shenyang 110169, China, and also with Neusoft Reach Automotive Technology (Shenyang), Shenyang 110179, China (e-mail: weiliu@mail.neu.edu.cn).

Kewen Li is with the College of Science, Liaoning University of Technology, Jinzhou 121001, China (e-mail: likewen2018@163.com).

Shuai Cheng, Zuotao Ning, and Yuefeng Wang are with Neusoft Reach Automotive Technology (Shenyang), Shenyang 110179, China (e-mail: cheng.shuai@reachauto.com; ningzt@reachauto.com; wangyuefeng@reachauto.com).

Digital Object Identifier 10.1109/TITS.2026.3726633

I. Introduction

In RECENT years, autonomous driving technology has become a major research focus in both academia and industry. The perception system, as a key component of environmental understanding, plays a pivotal role in autonomous driving [1]. Existing studies have made significant progress in single-task perception tasks such as 3D object detection [2], [3], [4], [5], [6], multi-object tracking (MOT) [7], [8], and online mapping [9], [10]. For instance, DeepInteraction++ [11], MDNet [12], and MSSF [13] use interaction enhancement, lightweight decoding, and semantic guidance strategies, respectively, to effectively improve the accuracy and stability of 3D object detection under multimodal conditions. OptiPMB [14], ReferGPT [15], and Contour Errors [16] improve multi-object tracking performance and perception robustness in complex scenarios from the perspectives of model optimization, language reasoning, and metric design, respectively. HisTrackMap [17], MapGS [18], and MapTRv2 [19] provide efficient, scalable, and generalizable solutions for online mapping from the perspective of temporal consistency, cross-sensor generalization, and single-frame accuracy, respectively. Although these methods have made significant progress, treating tasks as independent single-task models impedes cross-task collaboration, causing cascading errors, task fragmentation, computational redundancy, and limited global scene understanding.

To overcome the limitations of single-task perception in task decoupling, redundant computation, and insufficient global understanding, multi-task perception has gradually become a research hotspot [20]. Through the feature sharing and joint optimization mechanism, multi-task perception simultaneously models subtasks, such as 3D detection, multi-object tracking and mapping under a unified framework, which enhances inter-task interactions and effectively mitigates barriers to task collaboration. For example, Sparse4D [21] achieves multi-task perception through sparse 4D keypoint sampling and hierarchical feature fusion, effectively reducing computational cost while improving detection accuracy. CDRP3 [22] achieves multi-task joint perception, prediction, and planning through a cascaded deep reinforcement learning framework, enhancing safety and decision-making robustness in complex urban traffic. Uni-EPM [23] realizes multi-task detection and segmentation through a scalable perception architecture,

which significantly reduces the number of parameters while effectively avoiding repeated labeling and model stacking. The above research indicates that multi-task perception not only reduces redundant computations and cascading errors but also fully exploits complementary information among tasks, thereby significantly enhancing the accuracy and robustness of the overall perception system. However, the current multi-task perception methods are confronted with several challenges: they struggle to capture long-range dependencies while ensuring real-time, efficient learning for dynamic traffic participants. In addition, in the multi-scale feature fusion, coarse-grained features such as road structure and lane topology and fine-grained information such as small traffic participants and obstacle boundaries are not sufficiently modeled, which restricts the overall understanding of traffic scenarios. Moreover, most existing studies mainly focus on upstream perception, while the influence of perception quality on downstream decision-making remains insufficiently explored.

To address the above issues, this paper proposes **HDDPNet**: **H**ierarchical Cascade and **D**ual-stream **D**iffusion for Multi-task **P**erception in Autonomous Driving. This model realizes cross-task collaboration through unified multi-task perception modeling, while enhancing the reliability of feature interaction. Specifically, in the encoding stage, a Hierarchical Cascade Encoder (HCE) is constructed, where TransMamba is introduced to enhance global context modeling. In addition, a Progressive Feature Pyramid (PFP) is incorporated to achieve coarse-to-fine multi-scale feature extraction and cross-level semantic fusion. In the decoding stage, the Isomorphic Dual-stream Diffusion Decoder Network (ID^{3}Net) enables collaborative modeling between the detection-tracking stream and the mapping stream by sharing intermediate features and maintaining cross-task information consistency.

The main contributions of this work are summarized as follows:

(1) This paper proposes HDDPNet, a multi-task perception model for autonomous driving, which uses a Hierarchical Cascade Encoder (HCE) to capture long-range dependencies and fine-grained features for dynamic target perception and global scene understanding, together with an Isomorphic Dual-stream Diffusion Decoder Network (ID^{3}Net) to enhance feature robustness under noisy conditions.

(2) In the encoding stage, HCE is constructed with two key components, namely TransMamba and the Progressive Feature Pyramid (PFP). This design captures long-range dependencies along the driving direction while preserving fine-grained semantic features in local target regions, thereby improving dynamic object perception and global scene understanding.

(3) In the decoding stage, ID^{3}Net is introduced to perform a diffusion-inspired single-step denoising process on disturbed BEV features, while promoting feature interaction between the detection-tracking stream and the mapping stream, thereby improving the stability of multi-task perception.

(4) Experiments on the nuScenes dataset show that HDDPNet improves object detection accuracy while exhibiting stable and reliable performance in multi-task perception and downstream planning tasks.

The remainder of this paper is organized as follows: In Section II, we review related work on single-task and multi-task perception in autonomous driving. In Section III, the proposed HDDPNet model is introduced, and we provide a description of its components. We report and analyze the results of comparative and ablation experiments in Section IV. Finally, the work of this paper is summarized in Section V.

II. Related Work

A. Single-Task Perception in Autonomous Driving

Autonomous driving systems rely on high-precision perception to understand complex and dynamic traffic environments [24]. Single-task perception methods mainly include 3D object detection, multi-object tracking, and online mapping, which provide key semantic and geometric information for subsequent prediction and planning. In recent years, with the rapid development of multi-modal fusion, perception methods that combine heterogeneous sources such as images, point clouds, and radar data have become an important research direction. For 3D object detection, existing methods can be broadly categorized into point cloud-based [25], image-based [26], and multi-modal fusion approaches [27], all of which have significantly improved perception accuracy and spatial representation capability. For tracking, detection-based multi-object tracking [28] remains the dominant paradigm, where object trajectories are continuously associated and updated based on detection results and correlation algorithms. For online mapping, existing methods combine semantic segmentation with spatial reconstruction to extract key elements such as roads, lane lines, and pedestrian areas, thereby providing structured environmental priors for higher-level decision-making and planning [29]. Although single-task perception has achieved substantial progress in 3D object detection, object tracking, and semantic map construction, the independence of each task and the lack of cross-task information transfer may lead to cascading errors and limit the performance of collaborative perception. Motivated by this limitation, we propose HDDPNet, a multi-task perception framework that jointly models detection, tracking, and mapping through hierarchical cascade encoding and an isomorphic dual-stream diffusion decoder. This design is intended to better exploit the correlations among subtasks and provide more reliable perception features for downstream autonomous driving applications.

B. Multi-Task Perception in Autonomous Driving

With the increasing complexity of autonomous driving scenarios, multi-task perception has become essential for comprehensive scene understanding. In the literature, multi-task perception can be broadly categorized into homogeneous multi-task perception and heterogeneous multi-task perception. Homogeneous multi-task perception typically relies on a unified perception paradigm, where multiple fine-grained subtasks (e.g., object detection, lane detection, and drivable area segmentation) are jointly learned on top of shared feature representations to improve semantic and geometric consistency while reducing redundant computation. For example, YOLOP [31] constructs an end-to-end architecture with a

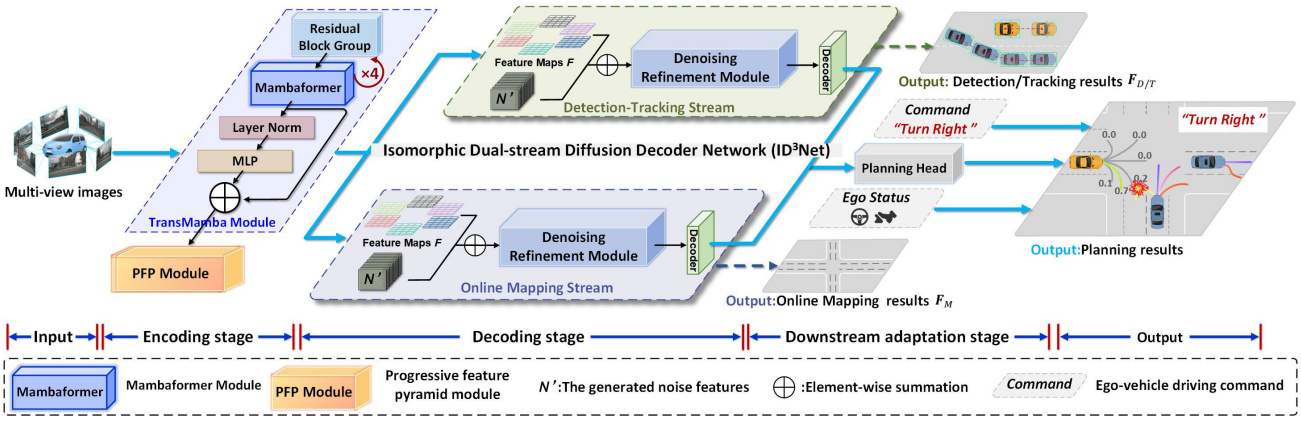


Fig. 1. Overall architecture of the proposed HDDPNet. (a) Encoding stage: a hierarchical cascade encoder is employed, where the TransMamba structure is introduced to model dynamic object features. In addition, a progressive feature pyramid module is integrated to achieve multi-scale semantic fusion, thereby producing high-quality BEV representations. (b) Decoding stage: an isomorphic dual-stream diffusion decoder architecture is adopted, which incorporates a diffusion-inspired single-step denoising process and consists of a detection-tracking stream and an online mapping stream. These streams output 3D detection, multi-object tracking, and online mapping results, respectively. (c) Downstream adaptation stage: based on the perception outputs produced by HDDPNet, an existing motion and planning head is used to initialize candidate trajectories, which are then refined through collision checking and ego-vehicle control command filtering to generate the final planning trajectory.

shared encoder and task-specific decoders, balancing accuracy and real-time performance across tasks. UF-Net [32] achieves unified optimization of lane detection, drivable area segmentation, and object detection through a two-stage feature refinement and gradient homogenization mechanism. In the point cloud domain, PAttFormer [33] performs unified modeling of detection and segmentation based on pure point cloud representations, significantly improving efficiency. Similarly, DRMNet [34] employs a dual-path multi-scale structure to effectively fuse shallow details and deep semantics, showing strong performance in vehicle detection and lane parsing.

By contrast, heterogeneous multi-task perception emphasizes cross-paradigm joint modeling and feature sharing, typically based on shared BEV representations or hierarchical decoders, for tasks such as 3D object detection, multi-object tracking, and online mapping. For instance, the Task-Adaptive Attention Network [35] employs task-specific attention to improve the balance between collaboration and real-time performance, while Zhou and Zeng [30] propose a two-stage framework to balance efficiency and task diversity.

C. Relation to Prior Work

According to the taxonomy in Section II-B, HDDPNet belongs to the category of heterogeneous multi-task perception methods, as it leverages shared representations for joint learning of detection, tracking, and online mapping. Different from prior work, our method places greater emphasis on hierarchical feature modeling and cross-task interaction. Specifically, HDDPNet introduces a hierarchical cascade encoder to strengthen long-range dependency modeling and multi-scale feature fusion, and further adopts an Isomorphic Dual-stream Diffusion Decoder Network with a diffusion-inspired single-step denoising process to enhance feature robustness and promote interaction between task branches. In this way, the proposed framework provides a more unified and reliable perception representation for autonomous driving.

III. Methodology

In this section, we first provide an overview of the architecture and processing pipeline of HDDPNet (Section III-A), as illustrated in Fig. 1. We then introduce the Hierarchical Cascade Encoder in the encoding stage (Section III-B). In the decoding stage, we present the Isomorphic Dual-stream Diffusion Decoder Network (ID^{3}Net) (Section III-C), followed by the downstream adaptation stage (Section III-D).

A. Overview

The detailed workflow is presented in Algorithm 1, and the operation of HDDPNet can be summarized in the following three stages. **(1) Encoding stage:** The Hierarchical Cascade Encoder (HCE) extracts global contextual features from multi-view images $I = \{I_1, I_2, \ldots, I_M\}$ through the TransMamba module, and generates multi-scale BEV representations $F^i_{\text{bev}} \in \mathbb{R}^{C\times H\times W}$ ($i \in [1, 2, 3, 4]$). Then, the Progressive Feature Pyramid (PFP) module is employed to fuse F^i_{bev} across scales to obtain the unified BEV representation F_{bev}, thereby improving multi-granularity feature representation and semantic structure modeling. **(2) Decoding stage:** The Isomorphic Dual-stream Diffusion Decoder Network (ID^{3}Net) introduces Gaussian perturbation N' into the fused BEV representation F_{bev}, and employs a self-attention module together with a diffusion inspired single-step denoising process to obtain stable feature representations. Subsequently, a dual-stream decoder generates 3D detection and multi-object tracking results $F_{D/T}$, as well as online mapping results F_M, thereby enabling collaborative multi-task perception modeling. **(3) Downstream adaptation stage:** To further evaluate the applicability of HDDPNet outputs to downstream trajectory prediction and planning, the ego feature F_{ego} is initialized from front-view camera inputs. Then, an existing motion and planning head is used to generate multiple candidate planning trajectories $\mathcal{T}_i$ (where i denotes the trajectory index), which are further refined with collision constraints and ego-vehicle control commands to produce the final executable path $\mathcal{T}_l$.

Algorithm 1 The HDDPNet Algorithm

Input: Multi-view images $I = \{I_1, I_2, \ldots, I_M\}$
Output: 3D detection & tracking results $F_{D/T}$, online map F_M, executable planning trajectory $\mathcal{T}_l$

Encoding Stage (HCE):
  for *each view* I_i *in* I **do**
    $F^i_{\text{bev}} \leftarrow \text{TransMamba}(I_i)$; // multi-scale BEV features
  $F_{\text{bev}} \leftarrow PFP\{F^1_{\text{bev}}, F^2_{\text{bev}}, F^3_{\text{bev}}, F^4_{\text{bev}}\}$; // feature fusion
Decoding Stage (ID^{3}Net):
  $F_{\text{noisy}} \leftarrow F_{\text{bev}} + N'$; // Gaussian perturbation
  $F_{\text{att}} \leftarrow \text{SelfAtt}(F_{\text{noisy}})$; // feature enhancement
  $F_{\text{out}} \leftarrow \text{Dec}(\text{Gna}(\text{Enc}(F_{\text{att}})))$; // single-step recovery
  $(F_{D/T}, F_M) \leftarrow \text{DualStreamDecoder}(F_{\text{out}})$; // dual-stream decoding
Downstream Adaptation Stage:
  Initialize ego vehicle feature F_{ego}
  for $i = 1, \ldots, N$ **do**
    $\mathcal{T}_i \leftarrow \text{PlanningHead}(F_{\text{ego}}, F_{D/T}, F_M)$
  $\mathcal{T}_l \leftarrow \text{Select}(\mathcal{T}_i, \text{Collision constraint}, \text{Command})$; // trajectory selection
return $\{F_{D/T}, F_M, \mathcal{T}_l\}$

B. Hierarchical Cascade Encoder Module

In autonomous driving scenarios, traffic participants exhibit complex long-range dependencies, nonlinear motions, and road structure constraints. To enhance scene understanding and improve multi-scale BEV feature representation, we propose the Hierarchical Cascade Encoder (HCE) in the encoding stage, which strengthens the model's representational capacity in complex scenarios. Specifically, the TransMamba module (Section III-B.1) is constructed to enhance global contextual modeling, while a progressive feature pyramid module (Section III-B.2) is designed to achieve coarse-to-fine multi-scale feature fusion.

1) TransMamba Module: BEV perception in autonomous driving usually faces several challenges, including sparse foreground targets, a large proportion of background regions, and significant long-range dependencies among road structures and traffic participants. Although linear attention can model long-range interactions, BEV features with sparse foreground targets and dominant background regions still require further enhancement of useful information and suppression of redundant responses. SSM can adaptively control information propagation through input-dependent sequence modeling, thereby efficiently aggregating long-range contextual information and reducing redundant background responses to some extent. However, the information propagation of SSM mainly relies on recurrent state transitions, which limits its ability to explicitly model relations between different spatial positions. Therefore, we construct the Mambaformer block by integrating SSM into the feature transformation process before the linear attention operation, where it is used for context aggregation and input-dependent feature selection. Subsequently, linear attention further models explicit interactions among different spatial positions. In this way, Mambaformer can enhance key-region feature representations and improve BEV feature representation.

As shown in Fig. 2, the TransMamba module consists of 4 layers of stacked units, each unit contains a residual block group and a TransMamba module. Given an input BEV image $I_{\text{BEV}} \in \mathbb{R}^{3\times H\times W}$, the network progressively extracts multi-scale features $F^i_{\text{bev}} \in \mathbb{R}^{C\times H\times W} (i \in [1,2,3,4])$, where H and W denote the input height and width, and 3 is the channel number. The computation is formulated as:

$$F^i_{\text{bev}} = M\big(R(F^{i-1}_{\text{bev}})\big) \tag{1}$$

where $R(\cdot)$ denotes the residual block group and $M(\cdot)$ represents the TransMamba module.

As shown in Eq. (2), the final feature representation includes multi-scale BEV features.

$$F_{\text{bev}} = \{F^1_{\text{bev}}, F^2_{\text{bev}}, F^3_{\text{bev}}, F^4_{\text{bev}}\} \tag{2}$$

As shown in Fig. 2, the TransMamba module integrates SSM, linear attention, and the gating mechanism within the Transformer framework. Its process is given as follows:

$$\hat{F} = F_R + \text{TranM}\big(\text{LN}(F_R)\big) \tag{3}$$

$$F_{\text{bev}} = \hat{F} + \text{MLP}\big(\text{LN}(\hat{F})\big) \tag{4}$$

where F_R denotes the feature produced by the residual block $R(\cdot)$, $\hat{F}$ is the output of the MambaFormer layer, and TranM$(\cdot)$, LN$(\cdot)$, and MLP$(\cdot)$ represent the MambaFormer block, layer normalization, and multi-layer perceptron.

The MambaFormer module is illustrated in Fig. 2 and operates as follows:

$$\hat{x} = \sigma\big(\text{linear}(x)\big), \tag{5}$$

$$\ddot{x} = \text{LinAttention}\Big(\text{SSM}\big(\sigma[\text{Conv}(\text{linear}(x))]\big)\Big) \tag{6}$$

$$Y = \text{linear}\big(\hat{x} \odot \ddot{x}\big) \tag{7}$$

where x denotes the output of LN(F_R), linear$(\cdot)$ is a linear layer, $\hat{x}$ is the gated-MLP output, Conv$(\cdot)$ is a 1×1 convolution, σ is the activation function, and $\odot$ indicates the element-wise multiplication.

2) Progressive Feature Pyramid Module (PFP): Feature pyramids effectively support multi-scale feature aggregation and are important for perceiving targets of different sizes. However, conventional FPN-based methods usually fuse adjacent-scale features by fixed element-wise addition or direct concatenation, which implicitly assumes similar contributions from different feature levels and thus limits the modeling of cross-scale semantic complementarity. This limitation is particularly problematic in autonomous driving scenarios, where small-scale targets, distant objects, and structural regions such as lane boundaries often require both low-level detailed cues and high-level semantic information. To address this issue, we propose a Progressive Feature Pyramid module (PFP), as

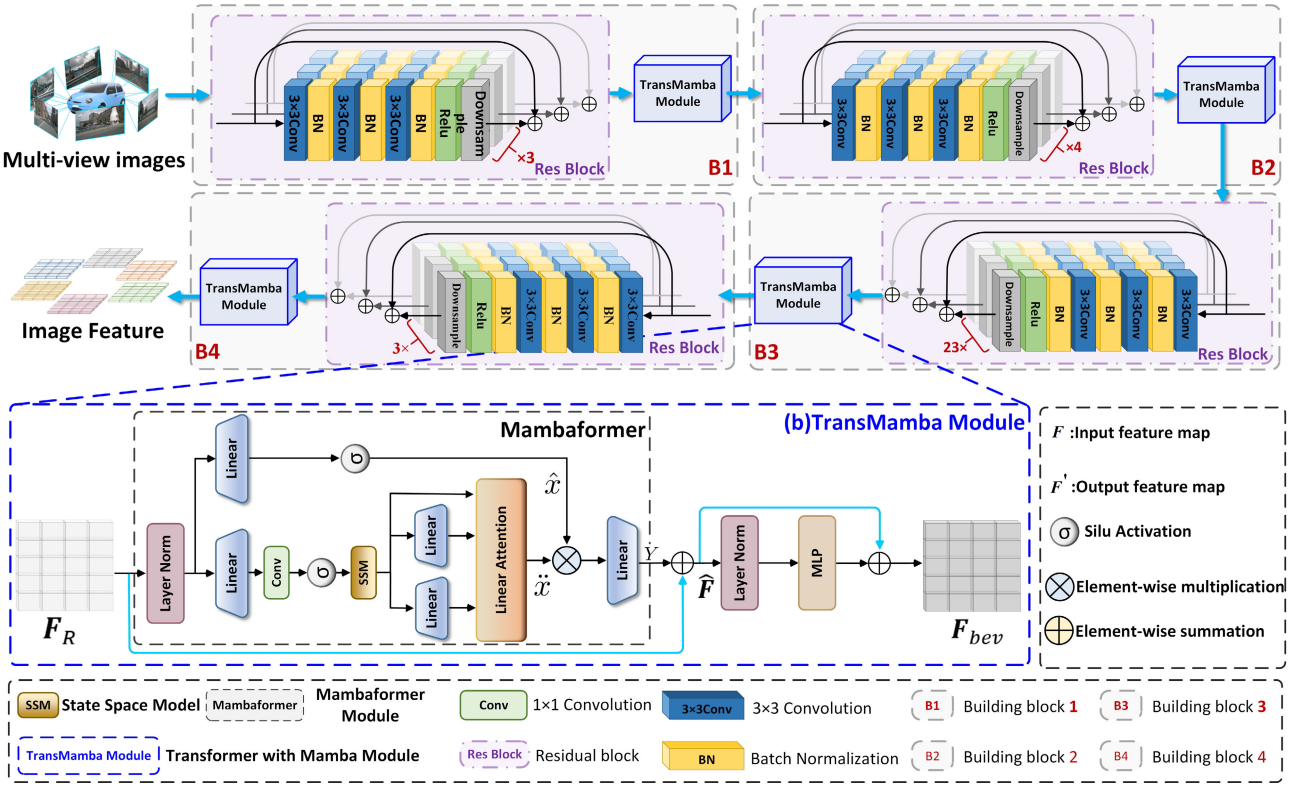


Fig. 2. Structure of the TransMamba module. Each basic unit consists of a residual block group and a TransMamba module. The former enhances the expression of local features, while the latter captures long-distance dependencies and global context, achieving hierarchical feature fusion.

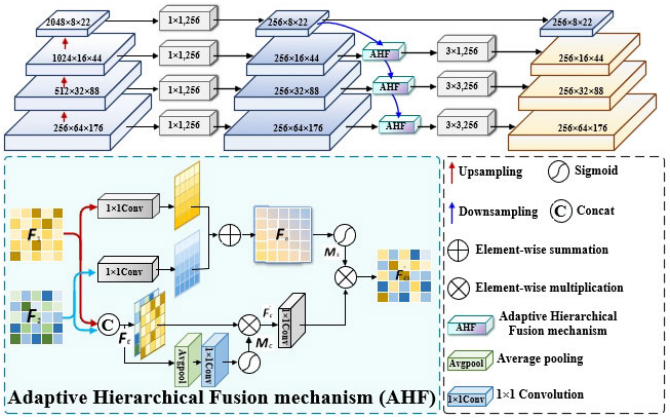


Fig. 3. Architecture of the Progressive Feature Pyramid module (PFP), which uses the Adaptive Hierarchical Fusion (AHF) mechanism to fuse multi-scale features from coarse to fine, and adaptively models collaborative relationships in spatial and channel dimensions, thereby enhancing semantic expression and multi-scale perception ability.

illustrated in Fig. 3. Rather than redesigning the overall top-down pyramid topology, PFP improves the fusion operation at each cross-scale fusion node through the Adaptive Hierarchical Fusion (AHF) mechanism. Specifically, AHF adaptively reweights adjacent-scale features in both channel and spatial dimensions, enabling input-dependent fusion instead of fixed fusion rules. This allows PFP to progressively aggregate multi-scale features from coarse to fine and improve feature representation.

The AHF mechanism is illustrated in Fig. 3. the input features $\mathbf{F}_1 \in \mathbb{R}^{C\times H\times W}$ and $\mathbf{F}_2 \in \mathbb{R}^{C\times H\times W}$ are concatenated along the channel dimension to obtain $\mathbf{F}_c \in \mathbb{R}^{2C\times H\times W}$. Then, channel-attention weights $\mathbf{M}_c$ are generated through average pooling, a 1×1 Convolution, normalization, and then multiplied with $\mathbf{F}_c$ to yield the channel-enhanced feature $\mathbf{F}'_c$:

$$\mathbf{F}_c = \mathrm{Concat}(\mathbf{F}_1, \mathbf{F}_2) \tag{8}$$

$$\mathbf{M}_c = \sigma\big(\mathrm{Conv}\big(\mathrm{AvgPool}(\mathbf{F}_c)\big)\big) \tag{9}$$

$$\mathbf{F}'_c = \mathbf{F}_c \odot \mathbf{M}_c \tag{10}$$

where $\sigma(\cdot)$ denotes the Sigmoid function and $\mathrm{Conv}(\cdot)$ is the 1×1 convolution.

In order to further extract the spatial dependencies between features, the input features F_1 and F_2 are added element by element to obtain the fused feature F_a, and the spatial attention weight M_S is generated through the normalization operation, which is multiplied with the channel-enhanced feature F'_c to produce the final fused representation F'_{CS}. This mechanism achieves efficient interaction and enhancement of multi-scale features by jointly modeling channels and spatial attention, thereby significantly improving feature representation and contextual modeling capability. The formulation is:

$$\mathbf{F}_a = \mathbf{F}_1 \oplus \mathbf{F}_2 \tag{11}$$

$$\mathbf{M}_S = \sigma(\mathbf{F}_a) \tag{12}$$

$$\mathbf{F}'_{CS} = \mathbf{F}'_c \odot \mathbf{M}_S \tag{13}$$

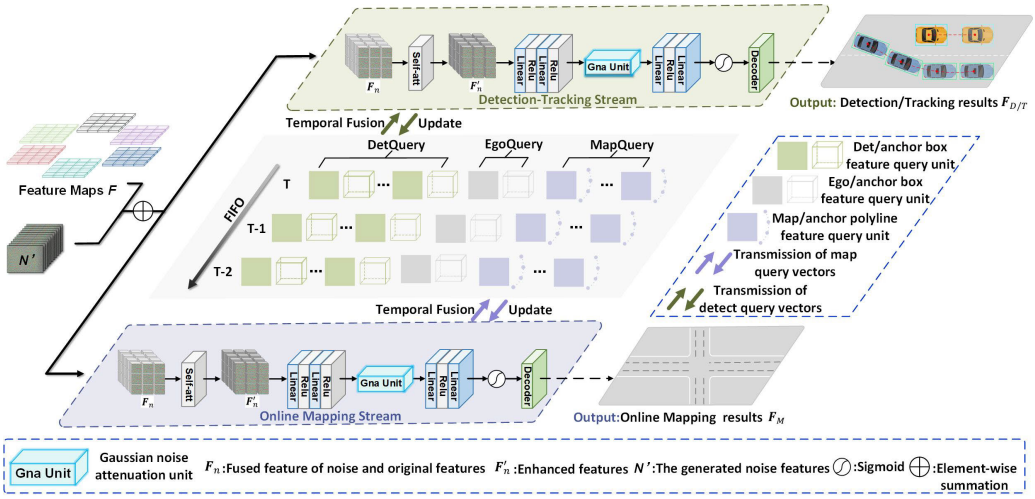


Fig. 4. Overall architecture of the proposed Isomorphic Dual-stream Diffusion Decoder Network (ID^{3}Net).

C. Isomorphic Dual-Stream Diffusion Decoder Network (ID^{3}Net)

In autonomous driving, BEV features are susceptible to occlusion, sensor uncertainty, and environmental noise, which may cause local disturbances and reduce the stability of feature representations. To improve the robustness of intermediate features in complex scenes, we propose the Isomorphic Dual-stream Diffusion Decoder Network (ID^{3}Net), inspired by the diffusion idea of noise perturbation and feature recovery. Specifically, ID^{3}Net introduces a single-step denoising process, where Gaussian perturbation is first applied to the input BEV features to simulate uncertainty and local noise in real scenes. Then, a recovery process together with a Gaussian noise attenuation unit is used to smooth disturbed features and recover more stable feature representations. As shown in Fig. 4, ID^{3}Net consists of two parallel task streams: a detection-tracking stream for dynamic object extraction and 3D detection/tracking, and an online mapping stream for geometric modeling. These streams interact at the feature level for complementary information sharing, thereby enhancing accuracy and robustness.

The decoding stage of ID^{3}Net mainly consists of two parts: the non-spatiotemporal decoder and the spatiotemporal decoder. In the non-spatiotemporal decoder, the encoded feature $\mathbf{F}$ is first processed by the diffusion-inspired single-step denoising module, and then sequentially refined by a deformable aggregation module and a feed-forward network, producing the initial non-spatiotemporal feature through the output layer. Subsequently, in the spatiotemporal decoder, temporal cross-attention and self-attention are employed to further fuse spatiotemporal features, yielding the final outputs for detection, tracking, and online mapping. The formulation of ID^{3}Net is given as follows:

$$\mathbf{F}_{\text{n-st}} = \mathcal{O}\Big(\text{FFN}\big[\text{Defo}(\text{Denoise}(\mathbf{F}))\big]\Big) \tag{14}$$

$$\mathbf{F}_{\text{D/T}} = \mathcal{O}\Big\{\text{FFN}\big[\text{SelfAtt}(\text{AttCross}(\mathbf{F}_{\text{n-st}}))\big]\Big\} \tag{15}$$

$$\mathbf{F}_{\text{M}} = \mathcal{O}\Big\{\text{FFN}\big[\text{SelfAtt}(\text{AttCross}(\mathbf{F}_{\text{n-st}}))\big]\Big\} \tag{16}$$

where Denoise($\cdot$) denotes the diffusion-inspired single-step denoising module, Defo($\cdot$) denotes the deformable aggregation module, FFN($\cdot$) represents the feed-forward network, and $\mathcal{O}(\cdot)$ is the output layer. $\mathbf{F}_{\text{n-st}}$ denotes the feature produced by the non-spatiotemporal decoder, AttCross($\cdot$) represents the temporal cross-attention module, SelfAtt($\cdot$) denotes the self-attention module, $\mathbf{F}_{\text{D/T}}$ denotes the detection and tracking outputs, and $\mathbf{F}_{\text{M}}$ denotes the online mapping output.

The diffusion-inspired single-step denoising process is formulated as follows. First, random Gaussian noise $\mathbf{N}'$ is added to the input BEV feature to obtain a noisy representation:

$$\mathbf{F}_{\text{noisy}} = \mathbf{F}_{\text{bev}} + \mathbf{N}' \tag{17}$$

The noisy feature is then enhanced by a self-attention module:

$$\mathbf{F}_{\text{att}} = \text{SelfAtt}(\mathbf{F}_{\text{noisy}}) \tag{18}$$

The enhanced feature $\mathbf{F}_{\text{att}}$ is mapped into the latent space by an encoder and then processed by a Gaussian noise attenuation unit for denoising. Finally, the decoder transforms the latent representation into the output feature:

$$\mathbf{F}_{\text{out}} = \text{Dec}\left(\text{Gna}\left(\text{Enc}(\mathbf{F}_{\text{att}})\right)\right) \tag{19}$$

Since the proposed module performs only a single-step recovery process without recursive denoising iterations, it reduces the risk of repeatedly propagating intermediate recovery errors in multi-step sampling.

The Gaussian noise attenuation (Gna) unit performs local Gaussian-weighted aggregation on the latent representation $\mathbf{H} \in \mathbb{R}^{L\times C}$ along the sequence dimension, where L and C denote the sequence length and channel dimension, respectively. For each position $l \in \{1,\ldots,L\}$ and channel $c \in \{1,\ldots,C\}$, Gna aggregates features within a local neighborhood to suppress high-frequency random noise while preserving the main structural information. Specifically, the output at position l is defined as:

$$\text{Gna}(\mathbf{H})_{l,c} = \sum_{u\in\Omega_l} \tilde{g}_l(u;\sigma)\,\mathbf{H}_{l+u,c} \tag{20}$$

where $\Omega_l = \{u \mid -r \le u \le r,\ 1 \le l + u \le L\}$ denotes the valid local neighborhood around position l, and the normalized Gaussian weights are defined as:

$$\tilde{g}_l(u;\sigma) = \frac{\exp\left(-\frac{u^2}{2\sigma^2}\right)}{\sum_{v\in\Omega_l}\exp\left(-\frac{v^2}{2\sigma^2}\right)}, \quad u \in \Omega_l \tag{21}$$

Here, $r \in \mathbb{N}$ denotes the neighborhood radius and $\sigma > 0$ controls the smoothing strength. In this way, neighboring positions closer to the center contribute more to the aggregation, which helps reduce local noise fluctuations while maintaining the continuity of the latent representation.

D. Downstream Adaptation Stage

In the downstream adaptation stage, the outputs of the perception module are integrated into the path planning process. Specifically, the ego feature F_{ego} is initialized using the front-view camera features and combined with the detection-tracking results $F_{D/T}$ and online mapping results F_M. The fused representation is then processed by a planning head to produce a set of candidate trajectories $\{\mathcal{T}_i\}_{i=1}^{N}$, where i denotes the trajectory index and N denotes the number of candidate trajectories. The candidate trajectories are filtered according to the high-level driving command (e.g., left turn, go straight, right turn) to obtain the subset $\mathcal{T}_{\text{cmd}}$ that matches the current driving intention. Finally, the optimal trajectory $\mathcal{T}_l$ is selected from this subset based on the collision constraint $C_{\text{collision}}(\cdot)$ to obtain the final planning path. The calculation process is as follows:

$$\mathbf{F}_{\text{ego}} = \mathbf{\Phi}_{\text{ego}}\left(\mathbf{F}_{\text{cam}}\right) \tag{22}$$

$$\{\mathcal{T}_i\}_{i=1}^{N} = \text{PlanHead}\left(\mathbf{F}_{\text{ego}}, \mathbf{F}_{D/T}, \mathbf{F}_M\right) \tag{23}$$

$$\mathcal{T}_{\text{cmd}} = \left\{\mathcal{T}_i \;\middle|\; \text{Direction}(\mathcal{T}_i) = \text{cmd},\ i = 1, \ldots, N\right\} \tag{24}$$

$$\mathcal{T}_l = \arg\min_{\mathcal{T}_i \in \mathcal{T}_{\text{cmd}}} C_{\text{collision}}(\mathcal{T}_i) \tag{25}$$

where PlanHead$(\cdot)$ represents the motion and planning head, cmd denotes the high-level driving command, $\mathcal{T}_{\text{cmd}}$ is the subset of candidate trajectories selected according to the driving command, and Direction$(\mathcal{T}_i)$ represents the direction category of trajectory $\mathcal{T}_i$.

E. Loss Function

In order to achieve joint perception and collaborative optimization of multiple tasks, HDDPNet introduces a multi-task Loss Function during the training process. The Joint Loss Function designed in this paper is defined as:

$$\mathcal{L} = \mathcal{L}_{\text{det}} + \mathcal{L}_{\text{map}} + \mathcal{L}_{\text{traj}} + \mathcal{L}_{\text{plan}} \tag{26}$$

where $\mathcal{L}_{\text{det}}$ denotes the 3D object detection loss, $\mathcal{L}_{\text{map}}$ denotes the online mapping loss, $\mathcal{L}_{\text{traj}}$ denotes the trajectory prediction loss, and $\mathcal{L}_{\text{plan}}$ denotes the planning loss.

1) 3D Object Detection Loss $\mathcal{L}_{det}$:

$$\mathcal{L}_{\text{det}} = \lambda_{\text{det-cls}}\mathcal{L}_{\text{det-cls}} + \lambda_{\text{det-reg}}\mathcal{L}_{\text{det-reg}} + \lambda_{\text{center}}\mathcal{L}_{\text{center}} + \lambda_{\text{yaw}}\mathcal{L}_{\text{yaw}} \tag{27}$$

The classification loss $\mathcal{L}_{\text{det-cls}}$ employs Focal Loss with weight $\lambda_{\text{det-cls}} = 2$; the regression loss $\mathcal{L}_{\text{det-reg}}$ uses L1 loss with weight $\lambda_{\text{det-reg}} = 0.25$; the centerness loss $\mathcal{L}_{\text{center}}$ adopts Sigmoid Cross-Entropy to measure the confidence of the predicted box center; the orientation regression loss $\mathcal{L}_{\text{yaw}}$ applies Gaussian Focal Loss to regularize obstacle orientation.

2) Online Mapping Loss $\mathcal{L}_{map}$: The online mapping task predicts a vectorized map that includes drivable-area boundaries, lane dividers, and crosswalks. Its loss function is defined as:

$$\mathcal{L}_{\text{map}} = \lambda_{\text{map-cls}}\mathcal{L}_{\text{map-cls}} + \lambda_{\text{map-reg}}\mathcal{L}_{\text{map-reg}} \tag{28}$$

The classification loss $\mathcal{L}_{\text{map-cls}}$ employs Focal Loss with weight $\lambda_{\text{map-cls}} = 1$; the regression loss $\mathcal{L}_{\text{map-reg}}$ adopts Lines L1 Loss with weight $\lambda_{\text{map-reg}} = 10$.

3) Trajectory Prediction Loss $\mathcal{L}_{traj}$: The trajectory prediction task estimates the future trajectories of all traffic participants in the scene. The loss function is composed of a classification loss $\mathcal{L}_{\text{traj-cls}}$ and a regression loss $\mathcal{L}_{\text{traj-reg}}$:

$$\mathcal{L}_{\text{traj}} = \lambda_{\text{traj-cls}}\mathcal{L}_{\text{traj-cls}} + \lambda_{\text{traj-reg}}\mathcal{L}_{\text{traj-reg}} \tag{29}$$

The classification loss $\mathcal{L}_{\text{traj-cls}}$ employs Focal Loss with weight $\lambda_{\text{traj-cls}} = 0.2$; the regression loss $\mathcal{L}_{\text{traj-reg}}$ uses L1 Loss with weight $\lambda_{\text{traj-reg}} = 0.2$.

4) Planning Loss $\mathcal{L}_{plan}$: This loss is designed to optimize the generation of the ego vehicle's future trajectory and consists of classification loss, regression loss, and status loss, defined as follows:

$$\mathcal{L}_{\text{plan}} = \lambda_{\text{plan-cls}}\mathcal{L}_{\text{plan-cls}} + \lambda_{\text{plan-reg}}\mathcal{L}_{\text{plan-reg}} + \lambda_{\text{plan-status}}\mathcal{L}_{\text{plan-status}} \tag{30}$$

Specifically, the classification Loss $\mathcal{L}_{\text{plan-cls}}$ employs Focal Loss with a weight of $\lambda_{\text{plan-cls}} = 0.5$; the regression loss $\mathcal{L}_{\text{plan-reg}}$ uses L1 Loss with a weight of $\lambda_{\text{plan-reg}} = 1.0$; and the status loss $\mathcal{L}_{\text{plan-status}}$ also adopts L1 Loss with a weight of $\lambda_{\text{plan-status}} = 1.0$, which is used to ensure the consistency of the vehicle's speed, acceleration and heading state.

IV. Experiments and Analysis

This section presents the experimental design and result analysis. The dataset and evaluation metrics are introduced in Section IV-A, and the experimental settings are described in Section IV-B. Comparison experiments and ablation studies are reported in Section IV-C and Section IV-D, respectively. Finally, experimental discussion and limitations are provided in Section IV-E.

A. Dataset and Evaluation Metrics

1) Dataset: To evaluate multi-task perception and downstream task adaptation, we conduct experiments on the nuScenes autonomous driving dataset [37]. It contains 1,000 driving scenarios of 20 seconds each, with 40k keyframes in total, and follows the standard 700/150/150 train/val/test split.

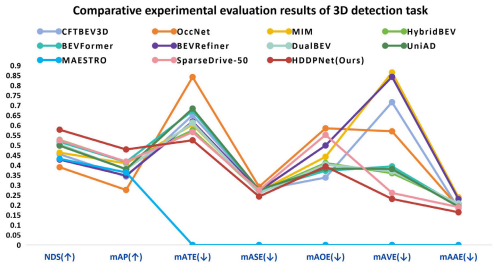


Fig. 5. Line graph of 3D detection evaluation metrics. The x-axis denotes the evaluation metrics and the y-axis shows their corresponding values.

2) Evaluation Metrics: The proposed method is evaluated on four core tasks: 3D object detection and tracking, online mapping, trajectory prediction, and planning. For 3D detection and tracking, standard nuScenes metrics are reported, including NDS, mAP, mATE, mASE, mAOE, mAVE, mAAE, AMOTA, AMOTP, Recall, and IDS. Online mapping performance is measured by the AP of pedestrian areas, lane dividers, and road boundaries. For trajectory prediction, minADE, minFDE, MR, and EPA are used to assess prediction accuracy and trajectory consistency, while L2 error and collision rate are adopted for planning to evaluate trajectory accuracy and safety.

B. Experimental Settings

All experiments are implemented on multiple NVIDIA RTX 4090 GPUs within the `mmdetection3d` framework based on PyTorch and CUDA. The network is optimized using AdamW [38] with an initial learning rate of 1×10^{-4} and trained for 100 epochs under a fixed schedule. Our method takes 3 consecutive historical frames as input to capture short-term temporal dependencies across tasks.

C. Comparison Experiments

To validate the effectiveness of HDDPNet, we conduct quantitative and qualitative experiments on the nuScenes validation set under a perception–prediction–planning framework. We compare HDDPNet with representative single-task perception models and multi-task methods, and further apply its perception outputs to downstream trajectory prediction and planning tasks. Results show that HDDPNet performs well on perception sub-tasks and provides effective inputs for downstream prediction and planning. Detailed qualitative comparisons are provided in Appendix A.

1) The Comparative Experimental Results and Analysis of Perception Tasks: This subsection evaluates the perceptual task performance from three aspects: 3D detection, multi-object tracking, and online mapping. We combined both quantitative metrics and qualitative visualizations to comprehensively evaluate the perception module's performance from different dimensions.

a) Comparative results and analysis of the 3D detection task: We systematically compare the proposed method with mainstream single-task models for 3D detection and multi-task methods on the nuScenes validation set. The quantitative results are summarized in Fig. 5 and Table I. To provide a more comprehensive and fair comparison, we further report the performance of HDDPNet under different historical frame settings and backbone configurations. In particular, with ResNet-101 and 3 historical frames, HDDPNet achieves the best results, reaching 0.619 NDS and 0.526 mAP. Under the ResNet-50 backbone, HDDPNet also achieves competitive results with 1, 2, and 3 historical frames, further supporting its effectiveness under different experimental settings. From the qualitative results in Fig. 6 and Appendix A, it can be observed that HDDPNet maintains reliable detection performance in representative challenging scenarios, such as going-straight, complex-intersection, and low-visibility scenarios. Even in the presence of dense traffic participants and partial occlusions, the proposed method still produces more accurate detection results, particularly for small-scale and occluded targets, demonstrating its robustness in complex environments.

b) Comparative results and analysis of multi-object tracking task: We compare HDDPNet with six representative single-task multi-object tracking methods and two multi-task methods on the nuScenes validation set. As shown in Table II, HDDPNet improves AMOTA by 7.29% and reaches a Recall of 0.556. Meanwhile, AMOTP is reduced by 3.54%, and IDS is reduced to 491, indicating better tracking accuracy and stronger identity consistency. The qualitative results in Fig. 6 and Appendix A further show that HDDPNet achieves stable tracking of multiple classes of dynamic objects while maintaining strong identity consistency, especially in complex scenarios with dense targets and occlusions.

c) Comparative results and analysis of online mapping task: We compare HDDPNet with representative single-task and multi-task methods. The quantitative results are shown in Table III, and qualitative results are provided in Fig. 6 and Appendix A. HDDPNet improves AP_{ped}, AP_{div}, and AP_{bdr} by 4.46%, 1.66%, and 0.85%, respectively. The largest gain is observed on pedestrian crossings, indicating better modeling of fine-grained geometric elements. The qualitative results further show that HDDPNet produces more complete map structures and clearer boundaries in complex traffic scenarios, which is beneficial for more reliable scene understanding in downstream tasks.

2) The Comparative Experimental Results and Analysis of Trajectory Prediction Tasks: To evaluate the downstream trajectory prediction performance, we compare HDDPNet with representative baselines.

Comparative Results and Analysis of Trajectory Prediction task: As shown in Table IV, HDDPNet reduces minADE to 0.61 and minFDE to 0.96, while improving EPA by 6.43% compared with the representative baselines. These results indicate that the proposed perception framework provides more informative scene representations for downstream trajectory prediction. Fig. 6 further shows that HDDPNet generates more plausible and continuous trajectories in complex turning scenarios, as illustrated in Fig. 6(a) and Fig. 6(c).

3) The Comparative Experimental Results and Analysis of Planning Tasks: We compare our method with single-task and multi-task methods using L2 error and collision rate.

TABLE I

COMPARATIVE RESULTS OF 3D DETECTION. BEST IN BOLD; SECOND-BEST UNDERLINED

Type	Methods	Frames	Modality	Backbone	NDS↑	mAP↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓
Single-Task Model (3D detection)	CFT-BEV3D [39]	1	Camera	ResNet-101	0.455	0.343	0.651	0.274	**0.338**	0.716	0.184
	OccNet [40]	1	Camera	-	0.390	0.276	0.842	0.292	0.585	0.570	0.190
	MIM [6]	1	Camera	ResNet-101	0.463	0.409	0.603	0.268	0.442	0.865	0.239
	HybridBEV [41]	1	Camera	ResNet-50	0.527	0.417	0.575	0.266	0.411	0.360	0.207
	BEVFormer [42]	2	Camera	ResNet-101	0.517	0.416	0.673	0.274	0.372	0.394	0.198
	BEVRefiner [5]	2	Camera	ResNet-50	0.428	0.348	0.619	0.271	0.499	0.843	0.230
	DualBEV [43]	2	Camera	ResNet-50	0.504	0.380	0.612	0.259	0.403	0.370	0.207
Multi-Task Model (3D detection)	UniAD [44]	1	Camera	ResNet-101	0.498	0.380	0.684	0.277	0.383	0.381	0.192
	HDDPNet (our)	1	Camera	ResNet-50	0.578	0.466	0.529	0.252	0.371	0.236	*0.161*
	MAESTRO [45]	-	Camera	ResNet-50	0.432	0.364	-	-	-	-	-
	HDDPNet (our)	2	Camera	ResNet-50	*0.591*	*0.482*	*0.524*	0.250	0.368	*0.207*	**0.153**
	SparseDrive-50 [46]	3	Camera	ResNet-50	0.525	0.418	0.566	0.275	0.552	0.261	0.190
	HDDPNet (our)	3	Camera	ResNet-50	0.578	0.467	0.525	*0.243*	0.394	0.231	0.164
	HDDPNet (our)	3	Camera	ResNet-101	**0.619**	**0.526**	**0.490**	**0.240**	*0.364*	**0.195**	**0.153**

Note: The proposed method is evaluated on the nuScenes validation set and compared with mainstream 3D detection models and end-to-end autonomous driving approaches.

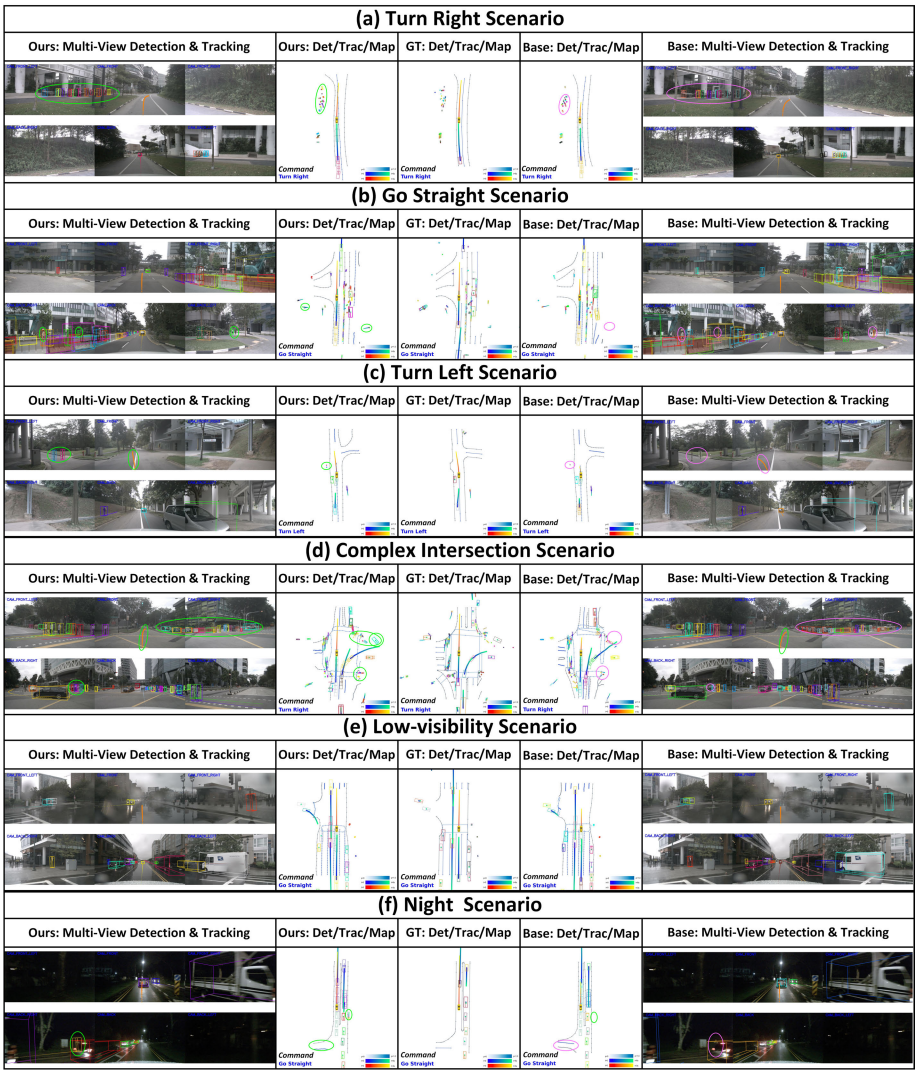


Fig. 6. Visualization results of HDDPNet in various complex scenarios. We present six scenarios: **(a)** Turn Right, **(b)** Go Straight, **(c)** Turn Left, **(d)** Complex Intersection, **(e)** Low-visibility, and **(f)** Night. Columns from left to right show: Ours multi-view detection and tracking, Ours joint perception, GT, baseline joint perception, and baseline multi-view detection and tracking. Det/Trac/Map denote 3D detection, multi-object tracking, and online mapping.

Comparative Results and Analysis of Planning task: As shown in Table V and Fig. 6, HDDPNet is compared with nine representative methods on the nuScenes validation set. Compared with the baseline, HDDPNet reduces the average

TABLE II

COMPARATIVE RESULTS OF MULTI-OBJECT TRACKING. BEST IN BOLD; SECOND-BEST UNDERLINED

Type	Methods	AMOTA↑	AMOTP↓	Recall↑	IDS↓
Single-Task Model (multi-object tracking)	ViP3D [47]	0.217	1.625	0.363	-
	QD3DT [48]	0.242	1.518	0.399	-
	Mutr3d [49]	0.294	1.498	0.427	3822
	SRCN3D [50]	0.398	1.317	*0.538*	4090
	CC-3DT [51]	0.410	1.274	*0.538*	3334
	MoMA-M3T [52]	*0.425*	*1.240*	-	-
Multi-Task Model	UniAD [44]	0.359	1.320	0.467	906
	SparseDrive-50 [46]	0.386	1.254	0.499	*886*
Ours	HDDPNet	**0.456**	**1.196**	**0.556**	**491**

Note: All methods are evaluated on the nuScenes validation set.

TABLE III

COMPARATIVE RESULTS OF ONLINE MAPPING. BEST IN BOLD; SECOND-BEST UNDERLINED

Type	Methods	AP_{ped}↑	AP_{div}↑	AP_{bdr}↑	mAP↑
Single-Task Model(mapping)	HDMapNet [53]	14.4	21.7	33.0	23.0
	VectorMapNet [54]	36.1	47.3	39.3	40.9
Multi-Task Model	VAD [55]	40.6	51.5	50.6	47.6
	SparseDrive-50 [46]	*49.9*	*57.0*	*58.4*	*55.1*
Ours	HDDPNet	**52.13**	**57.95**	**58.9**	**56.3**

Note: All methods are evaluated on the nuScenes validation set.

TABLE IV

COMPARATIVE RESULTS OF TRAJECTORY PREDICTION. BEST IN BOLD; SECOND-BEST UNDERLINED

Type	Methods	minADE(m)↓	minFDE(m)↓	MR↓	EPA↑
Single-Task Model (trajectory prediction)	Cons Pos. [44]	5.80	10.27	0.347	-
	Cons Vel. [44]	2.13	4.01	0.318	-
	Traditional [47]	2.06	3.02	0.277	0.209
	PnPNet [56]	1.15	1.95	0.226	0.222
	ViP3D [47]	2.05	2.84	0.246	0.226
	MoST [57]	**0.53**	1.10	**0.117**	-
Multi-Task Model	UniAD [44]	0.71	1.02	0.151	0.456
	SparseDrive-50 [46]	0.62	*0.99*	0.136	*0.482*
Ours	HDDPNet	*0.61*	**0.96**	*0.134*	**0.513**

Note: All methods are evaluated on the nuScenes dataset. In this table, "UniAD" refers to the motion forecasting result of the UniAD model reported in [44]. Likewise, Cons Pos. [44] and Cons Vel. [44] denote two forecasting baselines from the same work, corresponding to constant-position and constant-velocity assumptions, respectively.

L2 error by 1.66% and lowers the collision rates at the 2nd and 3rd seconds from 0.04% and 0.18% to 0.03% and 0.14%, respectively. Qualitative results in the Turn Left scenario (Fig. 6(c)) and the Complex Intersection scenario (Fig. 6(d)) further show that HDDPNet generates safer and more stable planned trajectories while maintaining smooth and feasible motion.

4) Efficiency Comparison Results and Analysis: This section reports both the inference efficiency comparison and the performance–complexity trade-off analysis.

TABLE V

COMPARATIVE RESULTS OF PLANNING. BEST IN BOLD; SECOND-BEST UNDERLINED

Type	Methods	L2 (m) ↓				Col. Rate (%) ↓			
		1 s	2 s	3 s	Avg.	1 s	2 s	3 s	Avg.
Single-Task Model	FF [58]	0.55	1.20	2.54	1.43	0.06	0.17	1.07	0.43
	EO [59]	0.67	1.36	2.78	1.60	0.04	0.09	0.88	0.33
	OccNet [40]	1.29	2.13	2.99	2.13	0.21	0.59	1.37	0.72
Multi-Task Model	ST-P3 [60]	1.33	2.11	2.90	2.11	0.23	0.62	1.27	0.71
	UniAD [44]	0.45	0.70	1.04	0.73	0.62	0.58	0.63	0.61
	VAD [55]	0.41	0.70	1.05	0.72	*0.03*	0.19	0.43	0.21
	SparseDrive-50 [46]	**0.29**	*0.58*	0.96	0.61	**0.01**	*0.05*	**0.18**	**0.08**
	GenAD [61]	0.36	0.83	1.55	0.91	0.06	0.23	1.00	0.43
	MomAD [62]	0.31	**0.57**	**0.91**	*0.60*	**0.01**	*0.05*	0.22	*0.09*
Ours	HDDPNet	*0.30*	**0.57**	*0.92*	**0.596**	**0.01**	**0.03**	*0.20*	**0.08**

Note: All methods are evaluated on the nuScenes validation set.

TABLE VI

EFFICIENCY COMPARISON RESULTS

Method	GPU Memory (M)	FLOPs (G)	Params (M)	FPS	Latency (ms)
SparseDrive-ResNet-50 [46]	1420	192.7	85.9	5.6	178.6
HDDPNet-ResNet-50	1558	284.6	108	5.4	185.2
SparseDrive-VMamba [63]	1376	106.4	102.3	7.8	128.2

Note: The inference efficiency of all methods is evaluated on a single RTX 4090 GPU.

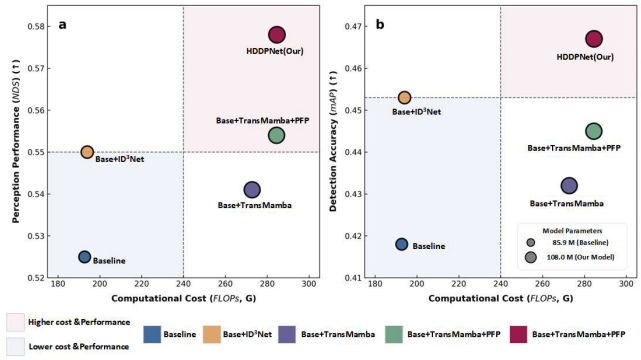


Fig. 7. Performance–complexity trade-off analysis of different HDDPNet variants. The x-axis denotes FLOPs, the y-axis denotes NDS and mAP, and the bubble size is proportional to the number of parameters.

a) Efficiency comparison results: We evaluate inference efficiency on a single NVIDIA GeForce RTX 4090 GPU, as shown in Table VI. SparseDrive-VMamba achieves higher inference efficiency, with 106.4G FLOPs and 7.8 FPS. However, lower detection and tracking performance is observed in Table VII, where NDS drops from 0.5250 to 0.4818 and AMOTA from 0.3860 to 0.3425. Compared with SparseDrive-ResNet-50, HDDPNet-ResNet-50 increases FLOPs from 192.7G to 284.6G, while FPS slightly decreases from 5.6 to 5.4. With this additional computation, HDDPNet-ResNet50 improves the 3D detection mAP from 0.418 to 0.4674 and the tracking AMOTA from 0.386 to 0.456. These results indicate that HDDPNet achieves higher perception performance at the cost of moderate additional computation.

b) Performance–Complexity trade-off analysis: To further analyze the relationship between computational

TABLE VII
COMPARISON WITH A MAMBA-BASED BACKBONE ON PERCEPTION TASKS

No.	Method	3D Detection							Multi-object Tracking			
		NDS↑	mAP↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓	AMOTA↑	AMOTP↓	Recall↑	IDS↓
Exp1	Baseline (ResNet-50)	0.5250	0.4180	0.5660	0.2750	0.5520	0.2610	0.1900	0.3860	1.2540	0.4990	886
Exp2	Baseline (VMamba)	0.4818	0.3909	0.6217	0.2831	0.6175	0.4161	0.1983	0.3425	1.3186	0.4728	1047
Exp3	HDDPNet (our)	**0.5780**	**0.4674**	**0.5250**	**0.2430**	**0.3940**	**0.2310**	**0.1640**	**0.4560**	**1.1964**	**0.5560**	**491**

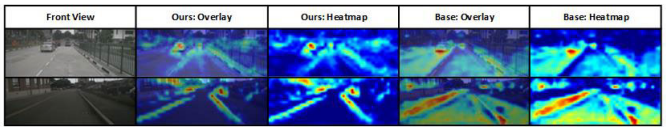


Fig. 8. Visualization of feature response heatmaps for the TransMamba module. From left to right: front-view image, ours: overlay, ours: heatmap, base: overlay, and base: heatmap.

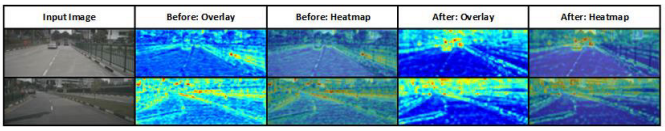


Fig. 9. Visualization of the P_3-level feature response after Progressive Feature Pyramid module (PFP).

cost and perception performance, we provide a performance–complexity trade-off visualization in Fig. 7. FLOPs are used as the computation axis, NDS and mAP are used as performance indicators, and the bubble size represents the number of parameters. As shown in Fig. 7, base+Denoise achieves clear gains with limited additional computation, increasing NDS from 0.525 to 0.550 and mAP from 0.418 to 0.453, while FLOPs increase only from 192.7G to 194.018G. Although TransMamba and PFP introduce higher computational costs, they further improve feature representation. The full HDDPNet achieves the best performance, with NDS and mAP reaching 0.578 and 0.467, respectively. The full HDDPNet achieves the best performance, with NDS and mAP reaching 0.578 and 0.467, respectively. Compared with the baseline, HDDPNet improves NDS by 0.053 and mAP by 0.049, while increasing FLOPs from 192.7G to 284.6G.

5) Comparison With Mamba-Based Vision Backbones: Table VII compares the original baseline, the baseline with its visual backbone replaced by VMamba [63], and our method. Under the same detection and tracking framework, Exp2 shows lower performance on both 3D detection and multi-object tracking, with NDS dropping from 0.5250 to 0.4818 and AMOTA dropping from 0.3860 to 0.3425. In contrast, our method achieves the best detection and tracking results among the compared methods, with NDS and AMOTA reaching 0.5780 and 0.4560. This indicates that, under the current framework, directly replacing the visual backbone with VMamba is not sufficient to obtain better detection and tracking performance.

D. Ablation Experiment

To comprehensively verify the effectiveness and contribution of each module, we conduct systematic ablation studies on two levels: perception sub-tasks and trajectory prediction–decision planning tasks.

1) Results and Analysis of Ablation Experiments on Perception Tasks: Table VIII reports the perception ablation results on 3D detection, multi-object tracking, and online mapping. Compared with Exp1, Exp2 improves mAP and NDS by 3.34% and 3.04%, respectively. In multi-object tracking, it reaches an AMOTA of 0.4217, which is 9.2% higher than that of Exp1, while reducing IDS by 277. To further illustrate the effectiveness of the TransMamba module, Fig. 8 shows the corresponding feature response heatmaps. Compared with the baseline model, Exp2 produces more concentrated and clearer responses in critical regions, such as foreground objects, lane structures, and road boundaries, while suppressing activation in irrelevant background regions. These observations provide qualitative support for the improvements in detection and tracking performance. Exp3 improves the detection performance by introducing ID^{3}Net in the decoding stage. Based on this, Exp4 further enhances online mapping performance, with pedestrian AP reaching 51.13. Furthermore, Fig. 9 visualizes the P_3-level feature produced by the PFP module. Compared with conventional feature fusion, the proposed PFP generates more concentrated responses on key objects and clearer activations around road boundaries and structural regions. This indicates that the PFP module can better enhance semantic representation and multi-scale perception through progressive feature fusion. Finally, Exp5 achieves the best overall performance, with NDS and mAP reaching 0.578 and 0.467 in 3D detection. It also improves online mapping mAP by 2.17% over Exp1, indicating improved robustness in online mapping.

2) Results and Analysis of Ablation Experiments for Trajectory Prediction and Planning Tasks: Table IX shows the ablation results for trajectory prediction and motion planning. Exp1 is the baseline model. Exp2 introduces the Hierarchical Cascade Encoder (HCE) and achieves an EPA of 0.496, indicating that global context modeling and fine-grained target perception are beneficial for downstream trajectory prediction. Exp3 further introduces the Isomorphic Dual-stream Diffusion Decoder Network (ID^{3}Net), reducing the L2 error and collision rate to 0.59 m and 0.08%, respectively. This indicates that ID^{3}Net provides more reliable perception outputs for planning.

E. Experimental Discussion and Limitations

Although the proposed method achieves strong overall performance, it still shows limitations in challenging

TABLE VIII
THE ABLATION EXPERIMENTS EVALUATION RESULTS OF THE PERCEPTION TASK

No.	HCE		ID^{3}Net	3D Detection							Multi-object Tracking				Online Mapping			
	TransMamba	PFP		NDS↑	mAP↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓	AMOTA↑	AMOTP↓	Recall↑	IDS↓	AP_{ped}↑	AP_{div}↑	AP_{bdr}↑	mAP↑
Exp1	×	×	×	0.525	0.418	0.566	0.275	0.552	0.261	0.1900	0.3860	1.254	0.4990	886	49.90	57.00	58.40	55.10
Exp2	✓	×	×	0.541	0.432	0.543	0.268	0.504	0.251	0.1787	0.4217	1.2287	0.5298	609	51.06	57.62	58.50	55.72
Exp3	×	×	✓	0.550	0.453	0.551	0.261	0.515	0.256	0.1820	0.4040	1.2360	0.5200	652	50.60	57.32	58.45	55.45
Exp4	✓	✓	×	0.554	0.445	0.539	0.263	0.462	0.248	0.1749	0.4236	1.2164	0.5323	581	51.13	57.60	58.54	55.75
Exp5	✓	✓	✓	**0.578**	**0.467**	**0.525**	**0.243**	**0.394**	**0.231**	**0.1640**	**0.4560**	**1.1964**	**0.5560**	**491**	**52.13**	**57.95**	**58.90**	**56.30**

Note: The table shows the ablation experimental results of perception tasks, mainly covering three aspects: 3D detection, multi-object tracking and online mapping. " × " denotes the corresponding module is removed, " ✓ " denotes the module is included.

TABLE IX
THE ABLATION EXPERIMENTS EVALUATION RESULTS OF THE TRAJECTORY PREDICTION AND PLANNING ABLATION EXPERIMENTAL TASKS

No.	HCE	ID^{3}Net	Trajectory prediction				Planning	
			minADE(m)↓	minFDE(m)↓	MR↓	EPA↑	L2(m)↓	C. R(%)↓
Exp1	×	×	0.62	0.99	0.136	0.482	0.61	0.08
Exp2	✓	×	0.62	0.97	0.136	0.496	0.62	0.10
Exp3	✓	✓	**0.61**	**0.96**	**0.134**	**0.513**	**0.59**	**0.08**

Note: " × " denotes the corresponding module is removed, " ✓ " denotes the module is included.

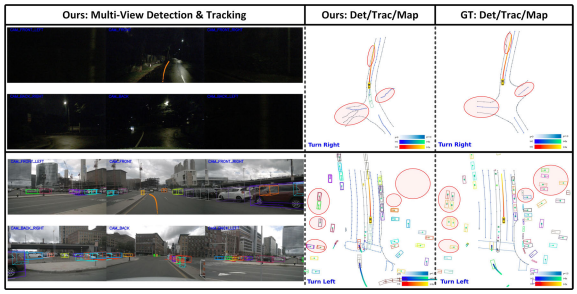


Fig. 10. Visualization results of HDDPNet in failure cases.

multi-task scenarios, as illustrated in Fig. 10. First, in the online mapping task, low-light nighttime conditions may lead to local omissions of map boundaries and incomplete or inaccurate topological relationships. Such errors may result in an incomplete understanding of the drivable area and scene topology, thereby providing insufficient structural constraints for downstream planning candidates. Second, in the 3D detection task, heavily occluded traffic participants remain difficult to detect reliably. Missed detections in these cases may weaken the understanding of dynamic interactions in the scene, which in turn affects the risk assessment of the motion forecasting and planning modules. As a result, these perception errors may propagate to downstream motion forecasting and planning, leading to suboptimal path selection in complex scenes. These qualitative observations are consistent with the quantitative performance gaps observed in challenging scenarios. In future work, we plan to improve the robustness of the system from two aspects: (1) incorporating multi-modal information fusion, such as LiDAR and millimeter-wave radar, to enhance mapping quality under poor illumination and improve the detection of occluded agents; and (2) designing more robust planning mechanisms under imperfect perception to reduce error propagation across tasks.

V. CONCLUSION AND DISCUSSION

A. Conclusion

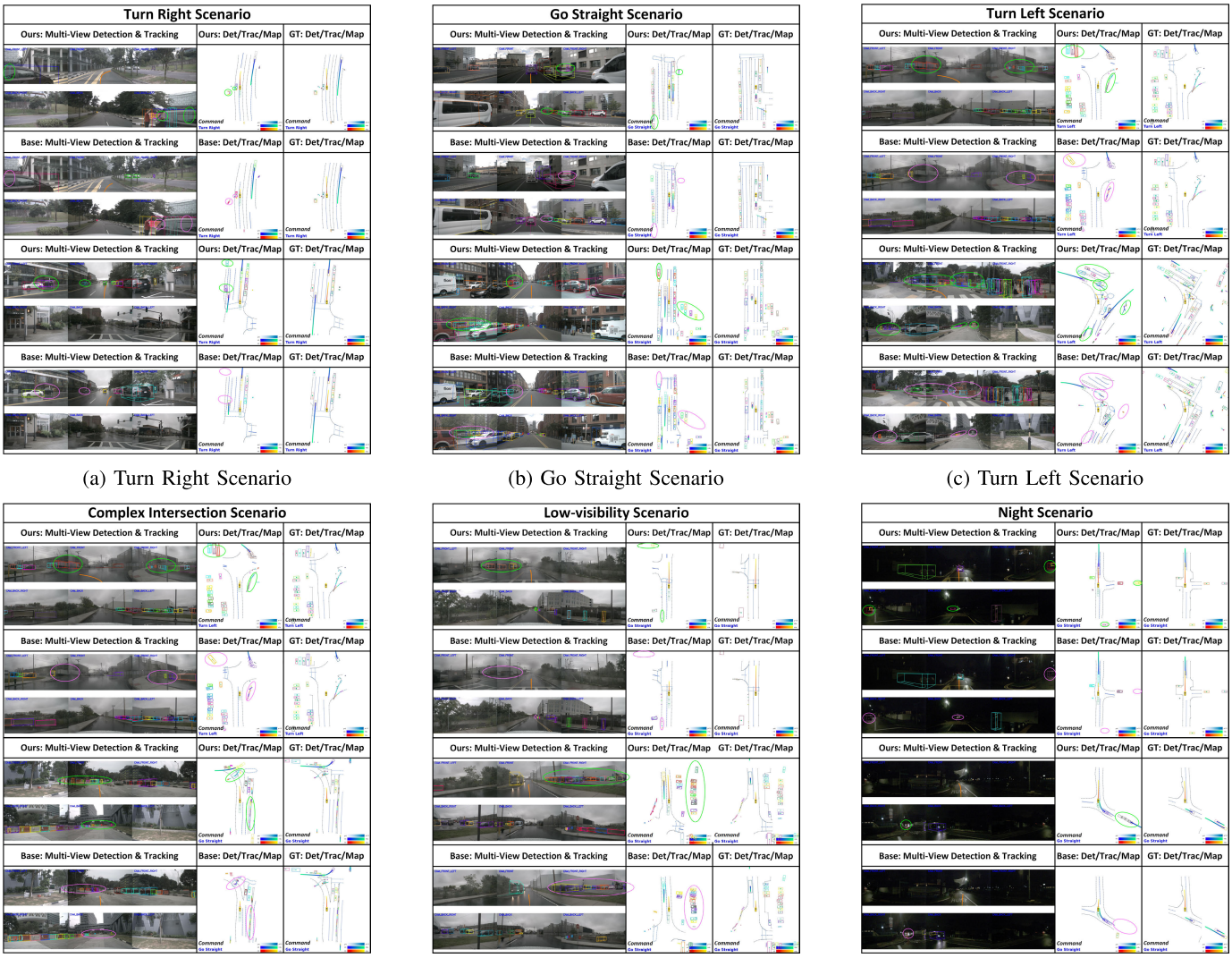
To address the limitations of existing methods in cross-task collaboration and scene representation, we propose HDDP-Net, which combines a Hierarchical Cascade Encoder (HCE) with an Isomorphic Dual-stream Diffusion Decoder Network (ID^{3}Net), thereby enhancing semantic representation and collaborative perception performance across tasks. Experimental results demonstrate that HDDPNet achieves consistent improvements across multiple subtasks, improving NDS by 10.09% in 3D object detection and AMOTA by 7.29% in multi-object tracking. By enhancing inter-task collaboration, HDDPNet provides an effective multi-task perception framework for autonomous driving.

B. Discussion

Although HDDPNet achieves improvements in perception consistency, trajectory prediction accuracy, and overall task collaboration, several aspects still deserve further exploration. As discussed in Section IV-E, the model still has room for improvement in online mapping under nighttime and low-visibility conditions, where incomplete map structures may affect downstream planning. In addition, heavily occluded traffic participants remain difficult to perceive reliably in complex interaction scenarios, which may influence trajectory prediction and risk assessment. In future work, we will explore multi-modal information fusion and uncertainty modeling to further improve the robustness and generalization of the proposed framework.

APPENDIX A
DETAILED VISUALIZATION RESULTS

To further demonstrate the effectiveness and robustness of the proposed HDDPNet, we present additional visualization results under different driving scenarios, including Turn Right, Go Straight, Turn Left, Complex Intersection, Low-visibility, and Night scenes. These examples illustrate the model's ability to detect and track small and occluded objects, maintain



(a) Turn Right Scenario (b) Go Straight Scenario (c) Turn Left Scenario

(d) Complex Intersection Scenario (e) Low-visibility Scenario (f) Night Scenario

Fig. 11. Visualization results of HDDPNet across six different driving scenarios. For each scenario, from left to right, they are: (1) multi-view detection and Tracking results of our method (Ours: Multi-View Detection & Tracking), (2) joint perception results (Ours: Det/Trac/Map), (3) ground-truth results (GT: Det/Trac/Map).

lane topology continuity, and generate reliable online maps under various challenging conditions. The visualizations are provided in Fig. 11. Specifically, the results in Fig. 11(b) and Fig. 11(c) show that the model can effectively detect and track small and partially occluded objects. In turning and complex intersection scenarios, as illustrated in Fig. 11(d), the model is able to preserve the continuity of lane geometry and topological relationships. In low-visibility and nighttime scenes, as shown in Fig. 11(e) and Fig. 11(f), the model still maintains relatively stable object perception and online map generation results.

APPENDIX B
FAILURE CASES

Although HDDPNet performs well in most scenarios, it still exhibits failure cases in challenging environments. As shown in Fig. 12, the constructed online maps may be incomplete or inaccurate due to low visibility. In narrow-road scenarios, such

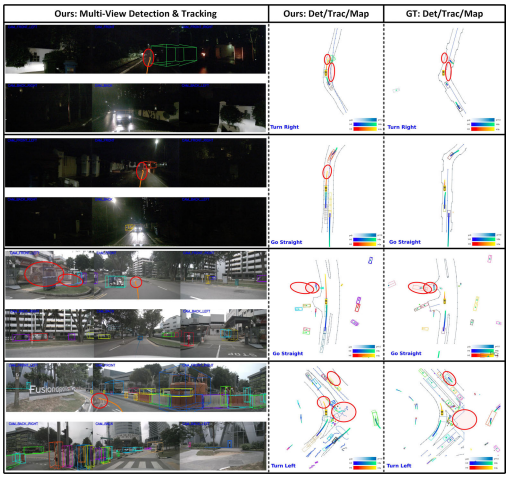


Fig. 12. Visualization of HDDPNet failure cases.

TABLE X
ABLATION ON THE 3D DETECTION CLASSIFICATION LOSS WEIGHT $\lambda_{\text{DET-CLS}}$

No.	$\lambda_{\text{det-cls}}$	3D Detection							Multi-object Tracking			
		NDS↑	mAP↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓	AMOTA↑	AMOTP↓	Recall↑	IDS↓
Exp1	1	0.5348	0.3923	0.5374	0.2484	0.3811	0.2658	0.1810	0.2106	1.2827	0.4784	599
Exp2	2	**0.5780**	**0.4674**	**0.5250**	**0.2430**	0.3940	0.2310	**0.1640**	**0.4560**	**1.1960**	**0.5560**	**491**
Exp3	3	0.5756	0.4609	0.5288	0.2495	**0.3746**	**0.2210**	0.1747	0.4385	1.2058	0.5487	526

TABLE XI
ABLATION ON THE 3D DETECTION REGRESSION LOSS WEIGHT $\lambda_{\text{DET-REG}}$

No.	$\lambda_{\text{det-reg}}$	3D Detection							Multi-object Tracking			
		NDS↑	mAP↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓	AMOTA↑	AMOTP↓	Recall↑	IDS↓
Exp1	0.125	0.5497	0.4264	0.5397	0.2547	0.3995	0.2676	0.1734	0.4494	**1.1882**	0.5334	**473**
Exp2	0.2	0.5780	**0.4674**	0.5250	**0.2430**	0.3940	0.2310	0.1640	**0.4560**	1.1960	0.5560	491
Exp3	0.5	**0.5789**	0.4662	**0.5210**	0.2481	**0.3892**	**0.2283**	**0.1554**	0.3641	1.2185	**0.5775**	542

TABLE XII
ABLATION ON THE ONLINE MAPPING CLASSIFICATION LOSS WEIGHT $\lambda_{\text{MAP-CLS}}$

No.	$\lambda_{\text{map-cls}}$	3D Detection							Multi-object Tracking				Online Mapping			
		NDS↑	mAP↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓	AMOTA↑	AMOTP↓	Recall↑	IDS↓	AP_{ped}↑	AP_{div}↑	AP_{bdr}↑	mAP↑
Exp1	1	**0.5780**	**0.4674**	**0.5250**	**0.2430**	0.3940	**0.2310**	0.1640	**0.4560**	**1.1964**	0.5560	**491**	52.13	57.95	58.90	56.30
Exp2	2	0.5662	0.4441	0.5311	0.2529	**0.3766**	0.2411	**0.1567**	0.3992	1.2066	0.5260	506	**54.76**	59.48	**60.06**	**58.10**
Exp3	3	0.5403	0.4071	0.5597	0.2508	0.4004	0.2635	0.1580	0.3298	1.2388	**0.5596**	573	53.21	**59.88**	58.87	57.32

TABLE XIII
ABLATION ON THE ONLINE MAPPING REGRESSION LOSS WEIGHT $\lambda_{\text{MAP-REG}}$

No.	$\lambda_{\text{map-reg}}$	3D Detection							Multi-object Tracking				Online Mapping			
		NDS↑	mAP↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓	AMOTA↑	AMOTP↓	Recall↑	IDS↓	AP_{ped}↑	AP_{div}↑	AP_{bdr}↑	mAP↑
Exp1	5	**0.5856**	**0.4819**	0.5334	0.2479	**0.3919**	0.2236	**0.1567**	**0.4727**	**1.1873**	**0.5634**	570	51.43	58.96	60.21	56.87
Exp2	10	0.5780	0.4674	**0.5250**	**0.2430**	0.3940	0.2310	0.1640	0.4560	1.1964	0.5560	491	52.13	57.95	58.90	56.30
Exp3	15	0.5737	0.4591	0.5291	0.2496	0.3943	**0.2185**	0.1671	0.4569	1.1874	0.5419	**476**	**54.62**	**59.98**	**60.24**	**58.28**

map errors can further affect downstream planning, causing the predicted trajectories to stay too close to the road boundaries and increasing the risk of collision. In addition, at complex intersections, severe occlusion may lead to missed detections of some objects, while the map structure in occluded regions is also difficult to recover accurately. In turning scenarios, the predicted trajectories tend to be conservative and may stay too close to the road boundaries. These observations indicate that HDDPNet still has room for improvement in nighttime perception, occlusion modeling, complex-scene map reconstruction, and planning coordination.

APPENDIX C
ABLATION ON LOSS WEIGHTING AND MULTI-TASK OPTIMIZATION

In this section, we further analyze the effects of loss weighting and multi-task optimization on the proposed method. We first conduct an ablation study on the loss weights of different task branches, including the classification and regression losses for 3D detection and map prediction. We then compare the adopted static weighting strategy with three representative multi-task loss balancing methods, namely PCGrad [64], GradNorm [66], and Uncertainty Weighting [65], to evaluate their impact on model performance.

A. Ablation Study on Loss Weights

1) 3D Detection Classification Loss $\lambda_{det\text{-}cls}$: Table X shows the effect of the 3D detection classification loss weight $\lambda_{\text{det-cls}}$. When $\lambda_{\text{det-cls}} = 1$, the model performs weakly on both 3D detection and multi-object tracking. Setting $\lambda_{\text{det-cls}} = 2$ achieves the best overall performance across the two tasks. Although $\lambda_{\text{det-cls}} = 3$ improves some detection-related error metrics, such as mAOE and mAVE, its overall detection and tracking performance are slightly worse than those of Exp2. Therefore, we adopt $\lambda_{\text{det-cls}} = 2$ as the default setting.

2) 3D Detection Regression Loss $\lambda_{det\text{-}reg}$: Table XI shows the effect of the 3D detection regression loss weight $\lambda_{\text{det-reg}}$. Compared with Exp1, setting $\lambda_{\text{det-reg}} = 0.2$ improves the overall 3D detection performance and achieves the best AMOTA in multi-object tracking. Although Exp3 obtains a slightly higher

TABLE XIV
COMPARISON OF DIFFERENT MULTI-TASK TRAINING STRATEGIES ON PERCEPTION TASKS

No.	Training Strategy	3D Detection							Multi-object Tracking				Online Mapping			
		NDS↑	mAP↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓	AMOTA↑	AMOTP↓	Recall↑	IDS↓	AP_{ped}↑	AP_{div}↑	AP_{bdr}↑	mAP↑
Exp1	PCGrad [64]	**0.5830**	0.4718	**0.5135**	**0.2409**	0.3831	0.2301	**0.1615**	0.2941	1.2274	**0.5680**	696	**54.26**	**59.68**	**60.47**	**58.14**
Exp2	Uncertainty [65]	0.5539	0.4275	0.5398	0.2503	0.3958	0.2466	0.1659	0.2686	1.2427	0.5202	587	53.37	59.06	59.90	57.44
Exp3	GradNorm [66]	0.5828	**0.4736**	0.5340	0.2473	**0.3827**	**0.2107**	0.1653	**0.4720**	**1.1868**	0.5508	**470**	52.91	58.58	59.46	56.98
Exp4	HDDPNet (our)	0.5780	0.4674	0.5250	0.2430	0.3940	0.2310	0.1640	0.4560	1.1964	0.5560	491	52.13	57.95	58.90	56.30

NDS, its mAP and most tracking metrics are worse than those of Exp2. Therefore, we choose $\lambda_{\text{det-reg}} = 0.2$ as the final setting.

3) Map Classification Loss $\lambda_{map\text{-}cls}$: Table XII shows the effect of the map classification loss weight $\lambda_{\text{map-cls}}$. Setting $\lambda_{\text{map-cls}} = 1$ achieves the best 3D detection and multi-object tracking results, whereas $\lambda_{\text{map-cls}} = 2$ provides the best online mapping performance, with a mAP of 58.10. When $\lambda_{\text{map-cls}} = 3$, the overall performance declines. Therefore, $\lambda_{\text{map-cls}} = 2$ is adopted as the default setting to balance the three tasks.

4) Map Regression Loss $\lambda_{map\text{-}reg}$: Table XIII shows the effect of the map regression loss weight $\lambda_{\text{map-reg}}$. When $\lambda_{\text{map-reg}} = 5$, the model achieves the best 3D detection and multi-object tracking results, whereas $\lambda_{\text{map-reg}} = 15$ provides the best online mapping performance, with a mAP of 58.28. Considering the trade-off among the three tasks, we adopt $\lambda_{\text{map-reg}} = 10$ as the final setting.

B. Comparison With Multi-Task Loss Balancing Methods

To further investigate the role of multi-task optimization, we compare the adopted static weighting strategy with three representative multi-task loss balancing methods, namely PCGrad [64], GradNorm [66], and Uncertainty Weighting [65]. As shown in Table XIV, these methods show different task preferences. PCGrad achieves the best 3D detection performance in terms of NDS (0.5830) and the best online mapping performance in terms of mapping mAP (58.14), but shows relatively weak tracking performance, with AMOTA dropping to 0.2941. GradNorm achieves the best tracking results, with AMOTA of 0.4720 and IDS of 470, while also maintaining competitive detection performance. By comparison, Uncertainty Weighting brings only limited overall gains. These results suggest that multi-task loss balancing methods can provide benefits for specific tasks or metrics, but no single method achieves the best performance across all tasks.

REFERENCES

[1] Y. Wang, B. Gong, Y. Wei, R. Ma, and L. Wang, "Video-based vehicle re-identification via channel decomposition saliency region network," *Appl. Intell.*, vol. 52, no. 11, pp. 12609–12629, Sep. 2022.

[2] F. Ma et al., "Monocular 3D lane detection for autonomous driving: Recent achievements, challenges, and outlooks," *IEEE Trans. Intell. Transp. Syst.*, vol. 26, no. 6, pp. 7380–7400, Jun. 2025.

[3] Y. Wang, H. Li, Y. Wei, C. Wang, and L. Wang, "Vehicle re-identification based on unsupervised local area detection and view discrimination," *Image Vis. Comput.*, vol. 104, Dec. 2020, Art. no. 104008.

[4] Y. Wang, Y. Wei, R. Ma, L. Wang, and C. Wang, "Unsupervised vehicle re-identification based on mixed sample contrastive learning," *Signal, Image Video Process.*, vol. 16, no. 8, pp. 2083–2091, Nov. 2022.

[5] B. Wang et al., "BEVRefiner: Improving 3D object detection in bird's-eye-view via dual refinement," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 10, pp. 15094–15105, Oct. 2024, doi: 10.1109/TITS.2024.3394550.

[6] J. Xiao, S. Wang, J. Zhou, Z. Tian, H. Zhang, and Y.-F. Wang, "MIM: High-definition maps incorporated multi-view 3D object detection," *IEEE Trans. Intell. Transp. Syst.*, vol. 26, no. 3, pp. 3989–4001, Mar. 2025, doi: 10.1109/TITS.2024.3520814.

[7] C. Jiang, M. Zhang, Y. Wang, and A. Zhang, "AHMOT: Adaptive Kalman filtering and hierarchical data association for 3-D multiobject tracking in IoT-enabled autonomous vehicles," *IEEE Internet Things J.*, vol. 12, no. 12, pp. 21290–21303, Jun. 2025.

[8] C. Xie, C. Lin, X. Zheng, B. Gong, and A. M. López, "Multi-modal 3D multi-object tracking with robust association and track drift compensation," *Neurocomputing*, vol. 657, Dec. 2025, Art. no. 131687.

[9] J. Chen, L. Pan, S. Ji, J. Zhao, and Z. Zhang, "Online temporal fusion for vectorized map construction in mapless autonomous driving," *IEEE Robot. Autom. Lett.*, vol. 10, no. 4, pp. 3948–3955, Apr. 2025.

[10] S. Wang et al., "Stream query denoising for vectorized HD-map construction," in *Proc. Eur. Conf. Comput. Vis.*, 2024, pp. 203–220.

[11] Z. Yang, N. Song, W. Li, X. Zhu, L. Zhang, and P. H. S. Torr, "DeepInteraction++: Multi-modality interaction for autonomous driving," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 47, no. 8, pp. 6749–6763, Aug. 2025.

[12] X. Guo, F. Jiang, Q. Chen, Y. Wang, K. Sha, and J. Chen, "Deep learning-enhanced environment perception for autonomous driving: MDNet with CSP-DarkNet53," *Pattern Recognit.*, vol. 160, Apr. 2025, Art. no. 111174.

[13] H. Liu, J. Liu, G. Jiang, and X. Jin, "MSSF: A 4D radar and camera fusion framework with multi-stage sampling for 3D object detection in autonomous driving," *IEEE Trans. Intell. Transp. Syst.*, vol. 26, no. 6, pp. 8641–8656, Jun. 2025.

[14] G. Ding et al., "OptiPMB: Enhancing 3D multi-object tracking with optimized Poisson multi-Bernoulli filtering," 2025, *arXiv:2503.12968*.

[15] T. Chamiti, L. D. Bella, A. Munteanu, and N. Deligiannis, "ReferGPT: Towards zero-shot referring multi-object tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2025, pp. 3849–3858.

[16] S. Kaul, S. Bultmann, M. Berk, and A. Valada, "Contour errors: Ego-centric matching for 3D multi-object tracking performance evaluation," 2025, *arXiv:2506.04122*.

[17] J. Yang, S. Yang, X. Tan, and H. Wang, "HisTrackMap: Global vectorized high-definition map construction via history map tracking," 2025, *arXiv:2503.07168*.

[18] H. Zhang, D. Paz, Y. Guo, X. Huang, H. I. Christensen, and L. Ren, "MapGS: Generalizable pretraining and data augmentation for online mapping via novel view synthesis," 2025, *arXiv:2501.06660*.

[19] B. Liao et al., "MapTRv2: An end-to-end framework for online vectorized HD map construction," *Int. J. Comput. Vis.*, vol. 133, no. 3, pp. 1352–1374, Mar. 2025.

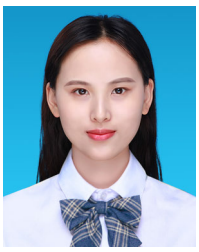
[20] Z. Li et al., "MtpNet: Multi-task panoptic driving perception network," *IEEE Trans. Intell. Transp. Syst.*, vol. 26, no. 6, pp. 7600–7609, Jun. 2025.

[21] X. Lin, T. Lin, Z. Pei, L. Huang, and Z. Su, "Sparse4D: Multi-view 3D object detection with sparse spatial–temporal fusion," 2022, *arXiv:2211.10581*.

[22] Y. Yang, F. Ge, J. Fan, J. Zhao, and Z. Dong, "CDRP3: Cascade deep reinforcement learning for urban driving safety with joint perception, prediction, and planning," *IEEE Trans. Intell. Transp. Syst.*, vol. 26, no. 3, pp. 3976–3988, Mar. 2025.

[23] Y. Gao, S. Mu, and S. Xu, "Uni-EPM: A unified extensible perception model without labeling everything," *IEEE Trans. Intell. Transp. Syst.*, vol. 26, no. 1, pp. 1002–1014, Jan. 2025.

[24] L. Dal'Col, M. Oliveira, and V. Santos, "Joint perception and prediction for autonomous driving: A survey," 2024, *arXiv:2412.14088*.
[25] R. Guan et al., "Talk2PC: Enhancing 3D visual grounding through LiDAR and radar point clouds fusion for autonomous driving," 2025, *arXiv:2503.08336*.
[26] S. Xie et al., "Benchmarking and improving bird's eye view perception robustness in autonomous driving," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 47, no. 5, pp. 3878–3894, May 2025.
[27] M. Chen et al., "Multi-modal BEV enhancement fusion for 3D object detection in autonomous driving," *IEEE Trans. Intell. Transp. Syst.*, vol. 26, no. 11, pp. 18447–18459, Nov. 2025.
[28] Z. Tian, Z. Zhou, X. Hua, K. Li, and F. R. Yu, "CATER: Causal adaptive tracking with environmental reasoning," *Signal Process.*, vol. 238, Jan. 2026, Art. no. 110169.
[29] N. Peng, X. Zhou, M. Wang, G. Chen, and W. Xu, "Uni-PrevPredMap: Extending PrevPredMap to a unified framework of prior-informed modeling for online vectorized HD map construction," 2025, *arXiv:2504.06647*.
[30] Y. Zhou and X. Zeng, "Towards comprehensive understanding of pedestrians for autonomous driving: Efficient multi-task-learning-based pedestrian detection, tracking and attribute recognition," *Robot. Auto. Syst.*, vol. 171, Jan. 2024, Art. no. 104580.
[31] D. Wu et al., "YOLOP: You only look once for panoptic driving perception," *Mach. Intell. Res.*, vol. 19, no. 6, pp. 550–562, Dec. 2022.
[32] Z. Zhou, P. Liu, and H. Huang, "UF-Net: A unified network for panoptic driving perception with two-stage feature refinement," *Expert Syst. Appl.*, vol. 260, Jan. 2025, Art. no. 125434.
[33] C. Lang, A. Braun, L. Schillingmann, and A. Valada, "A point-based approach to efficient LiDAR multi-task perception," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2024, pp. 13261–13268.
[34] J. Zhao, D. Wu, Z. Yu, and Z. Gao, "DRMNet: A multi-task detection model based on image processing for autonomous driving scenarios," *IEEE Trans. Veh. Technol.*, vol. 72, no. 12, pp. 15341–15355, Dec. 2023.
[35] W. Choi, M. Shin, H. Lee, J. Cho, J. Park, and S. Im, "Multi-task learning for real-time autonomous driving leveraging task-adaptive attention generator," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2024, pp. 14732–14739.
[36] W. Luo, Z. Huang, Z. Geng, J. Kolter, and G.-J. Qi, "One-step diffusion distillation through score implicit matching," in *Proc. Adv. Neural Inf. Process. Syst.*, 2024, pp. 115377–115408.
[37] H. Caesar et al., "nuScenes: A multimodal dataset for autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11621–11631.
[38] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," 2017, *arXiv:1711.05101*.
[39] H. Jiang, W. Meng, H. Zhu, Q. Zhang, and J. Yin, "Multi-camera calibration free BEV representation for 3D object detection," 2022, *arXiv:2210.17252*.
[40] C. Sima et al., "Scene as occupancy," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Jun. 2023, pp. 8406–8415.
[41] J. Wang, C. Du, H. Liu, Z. Xia, B. Liu, and S. Xiong, "HybridBEV: Hybrid encode and distillation for improved BEV 3D object detection," *IEEE Trans. Intell. Transp. Syst.*, vol. 26, no. 11, pp. 21257–21270, Nov. 2025.
[42] Z. Li et al., "BEVFormer: Learning bird's-eye-view representation from LiDAR-camera via spatiotemporal transformers," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 47, no. 3, pp. 2020–2036, Mar. 2025.
[43] P. Li, W. Shen, Q. Huang, and D. Cui, "DualBEV: Unifying dual view transformation with probabilences," in *Proc. Eur. Conf. Comput. Vis. Cham*, Switzerland: Springer, 2024, pp. 286–302.
[44] Y. Hu et al., "Planning-oriented autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2023, pp. 17853–17862.
[45] C. Kang, J. Kim, H. Shin, J. Park, and J. W. Choi, "MAESTRO: Task-relevant optimization via adaptive feature enhancement and suppression for multi-task 3D perception," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2025, pp. 28313–28323.
[46] W. Sun, X. Lin, Y. Shi, C. Zhang, H. Wu, and S. Zheng, "SparseDrive: End-to-end autonomous driving via sparse scene representation," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2025, pp. 8795–8801.
[47] J. Gu et al., "ViP3D: End-to-end visual trajectory prediction via 3D agent queries," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 5496–5506.
[48] H.-N. Hu, Y.-H. Yang, T. Fischer, T. Darrell, F. Yu, and M. Sun, "Monocular quasi-dense 3D object tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 2, pp. 1992–2008, Feb. 2023.
[49] T. Zhang, X. Chen, Y. Wang, Y. Wang, and H. Zhao, "MUTR3D: A multi-camera tracking framework via 3D-to-2D queries," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2022, pp. 4536–4545.
[50] Y. Shi et al., "SRCN3D: Sparse R-CNN 3D for compact convolutional multi-view 3D object detection and tracking," 2022, *arXiv:2206.14451*.
[51] T. Fischer, Y.-H. Yang, S. Kumar, M. Sun, and F. Yu, "CC-3DT: Panoramic 3D object tracking via cross-camera fusion," 2022, *arXiv:2212.01247*.
[52] K.-C. Huang, M.-H. Yang, and Y.-H. Tsai, "Delving into motion-aware matching for monocular 3D object tracking," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 6886–6895.
[53] Q. Li, Y. Wang, Y. Wang, and H. Zhao, "HDMapNet: An online HD map construction and evaluation framework," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2022, pp. 4628–4634.
[54] Y. Liu, T. Yuan, Y. Wang, Y. Wang, and H. Zhao, "VectorMapNet: End-to-end vectorized HD map learning," in *Proc. Int. Conf. Mach. Learn.*, 2023, pp. 22352–22369.
[55] B. Jiang et al., "VAD: Vectorized scene representation for efficient autonomous driving," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 8306–8316.
[56] M. Liang et al., "PnPNet: End-to-end perception and prediction with tracking in the loop," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11550–11559.
[57] N. Mu et al., "MoST: Multi-modality scene tokenization for motion prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 14988–14999.
[58] P. Hu, A. Huang, J. Dolan, D. Held, and D. Ramanan, "Safe local motion planning with self-supervised freespace forecasting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 12727–12736.
[59] T. Khurana, P. Hu, A. Dave, J. Ziglar, D. Held, and D. Ramanan, "Differentiable raycasting for self-supervised occupancy forecasting," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 353–369.
[60] S. Hu, L. Chen, P. Wu, H. Li, J. Yan, and D. Tao, "ST-P3: End-to-end vision-based autonomous driving via spatial–temporal feature learning," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 533–549.
[61] W. Zheng, R. Song, X. Guo, C. Zhang, and L. Chen, "GenAD: Generative end-to-end autonomous driving," in *Proc. Eur. Conf. Comput. Vis.*, 2024, pp. 87–104.
[62] Z. Song et al., "Don't shake the wheel: Momentum-aware planning in end-to-end autonomous driving," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2025, pp. 22432–22441.
[63] Y. Liu et al., "VMamba: Visual state space model," in *Proc. Adv. Neural Inf. Process. Syst.*, 2024, pp. 103031–103063.
[64] T. Yu, S. Kumar, A. Gupta, S. Levine, K. Hausman, and C. Finn, "Gradient surgery for multi-task learning," in *Proc. Int. Conf. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 5824–5836.
[65] R. Cipolla, Y. Gal, and A. Kendall, "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 7482–7491.
[66] Z. Chen, V. Badrinarayanan, C.-Y. Lee, and A. Rabinovich, "GradNorm: Gradient normalization for adaptive loss balancing in deep multitask networks," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2018, pp. 794–803.



Xiangxiang Zhang received the M.S. degree from North Minzu University, China, in 2024. She is currently pursuing the Ph.D. degree with Northeastern University, China. Her research interests include computer vision and autonomous driving.

Jun Hu received the Ph.D. degree from Northeastern University, Shenyang, China. He is currently a Professor with Shenyang University of Technology. His research interests include computer vision, artificial intelligence, and autonomous driving.

Wei Liu received the M.S. and Ph.D. degrees in control theory and control engineering from Northeastern University, Shenyang, China, in 2001 and 2005, respectively. He is currently a Professor-Level Senior Engineer with Research Academy, Northeastern University, China. He is the Director of Intelligent Vision Laboratory, Neusoft Corporation, China. His research interests include computer vision, image processing, and pattern recognition with applications to intelligent video surveillance and advanced driver assistance systems.

Kewen Li (Member, IEEE) received the Ph.D. degree from Qufu Normal University in 2023. He is currently a Ph.D. Supervisor with Liaoning University of Technology. His research interests include nonlinear intelligent control, optimal/optimization control, and containment/formation control.

Shuai Cheng (Member, IEEE) received the Ph.D. degree from Changchun University of Science and Technology in 2016. He is currently a Research Engineer with the Advanced Automotive Electronics Technology Research Center, Neusoft Corporation, China. His research interests include object detection and deep learning.

Zuotao Ning received the M.S. degree in artificial intelligence from The University of Edinburgh, Edinburgh, U.K., in 2016. He is currently with Neusoft Reach Automotive Technology (Shenyang) Company Ltd., where he is on the autonomous driving system as an Associate Researcher. His research interests include the environmental perception of vehicles and mobile robots.

Jinghan Xu received the M.S. degree from Liaoning University of Technology, China, in 2025. He is currently pursuing the Ph.D. degree with Northeastern University, China. His research interests include computer vision and decision planning.

Yuefeng Wang received the B.S. and M.S. degrees from Jilin University, China, in 2013 and 2017, respectively, and the Ph.D. degree from Northeastern University, China, in 2022. He is currently a full-time Researcher with ReachAuto, China. His research interests include image processing and pattern recognition, deep learning, and intelligent decision and planning.