UniMM-V2X: MoE-Enhanced Multi-Level Fusion for End-to-End Cooperative Autonomous Driving

Anonymous submission

Abstract

Autonomous driving holds transformative potential but remains fundamentally constrained by the limited perception and isolated decision-making with standalone intelligence. While recent multi-agent approaches introduce cooperation, they often focus merely on perception-level tasks, overlooking the alignment with downstream planning and control, or fall short in leveraging the full capacity of the recent emerging end-to-end autonomous driving. In this paper, we present UniMM-V2X, a novel end-to-end multi-agent framework that enables hierarchical cooperation across perception, prediction, and planning. At the core of our framework is a multi-level fusion strategy that unifies perception and prediction cooperation, allowing agents to share structured queries and cooperatively reason about the environment to achieve globally consistent and safe decision-making. To address the scalability and complexity issues, we incorporate a Mixture-of-Experts (MoE) architecture to dynamically enhance the BEV representation to handle multiple downstream tasks. We further extend MoE into the decoder to better capture diverse motion patterns. Extensive experiments on the DAIR-V2X dataset demonstrate the effectiveness of our approach, achieving state-of-the-art (SOTA) performance with a 39.7% improvement in perception accuracy, a 7.2% reduction in prediction error, and a 33.2% improvement in planning performance compared with UniV2X, showcasing the strength of our MoE-enhanced multi-level cooperative paradigm. The code is included in the supplementary materials.

Introduction

Traditional autonomous driving pipelines, with their modular structure, suffer from error propagation and limited generalization. As (Li et al. 2024; Philion and Fidler 2020) improved environmental perception through bird's-eye-view (BEV) representations, end-to-end autonomous driving has been widely studied in (Hu et al. 2023; Jiang et al. 2023; Jia et al. 2025; Sun et al. 2024). Although end-to-end autonomous driving offers a solution by directly mapping raw sensor data to final control, this standalone-intelligence system is constrained by sensor range and the difficulty in inferring other agents' intentions, struggling with rare critical events. Vehicle-to-Everything (V2X) communication emerges as a key enabler to overcome these limitations by facilitating real-time information exchange.

As shown in Figure 1(a), V2X communication is widely applied in cooperative perception, improving environmental awareness and system robustness through multi-agent cooperation (Xu et al. 2022b,a; Chen et al. 2019a; Hu et al. 2022b). However, since the objectives of cooperative perception are typically optimized for intermediate metrics such as detection accuracy or segmentation quality, they do not necessarily align with ultimate planning goals, leading to representations that may be accurate but not necessarily relevant for decision-making (Zeng et al. 2019). For instance, a cooperative perception system might precisely detect distant static objects, while failing to prioritize dynamic agents that directly impact motion decisions. To address this issue, several end-to-end cooperative autonomous driving approaches are designed to optimize the final planning per-

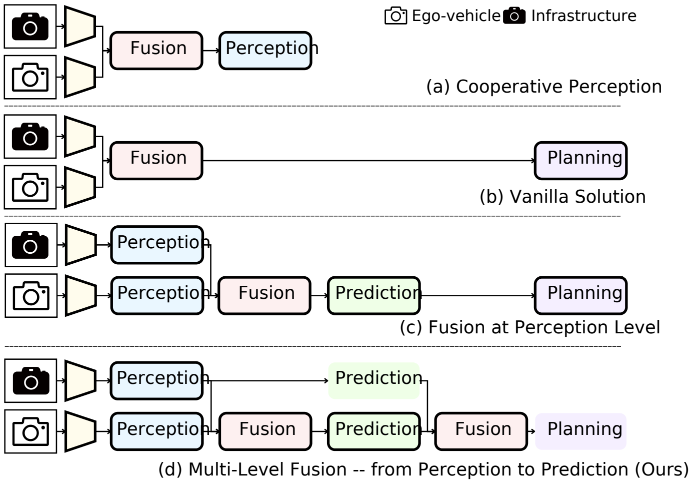


Figure 1: V2X communication modes in the VICAD (Vehicle-to-Infrastructure Cooperation Autonomous Driving) problem (Yu et al. 2022). (a) Cooperative perception methods focus on multi-agent detection and tracking, but may not align with planning objectives. (b) Vanilla solutions fuse features directly to generate planning outputs, with limited interpretability and compromised safety. (c) Module results can be supervised, but only enable perception-level cooperation. (d) Our design employs multi-level, multi-agent cooperation that integrates perception and prediction to enable cooperative decision-making.

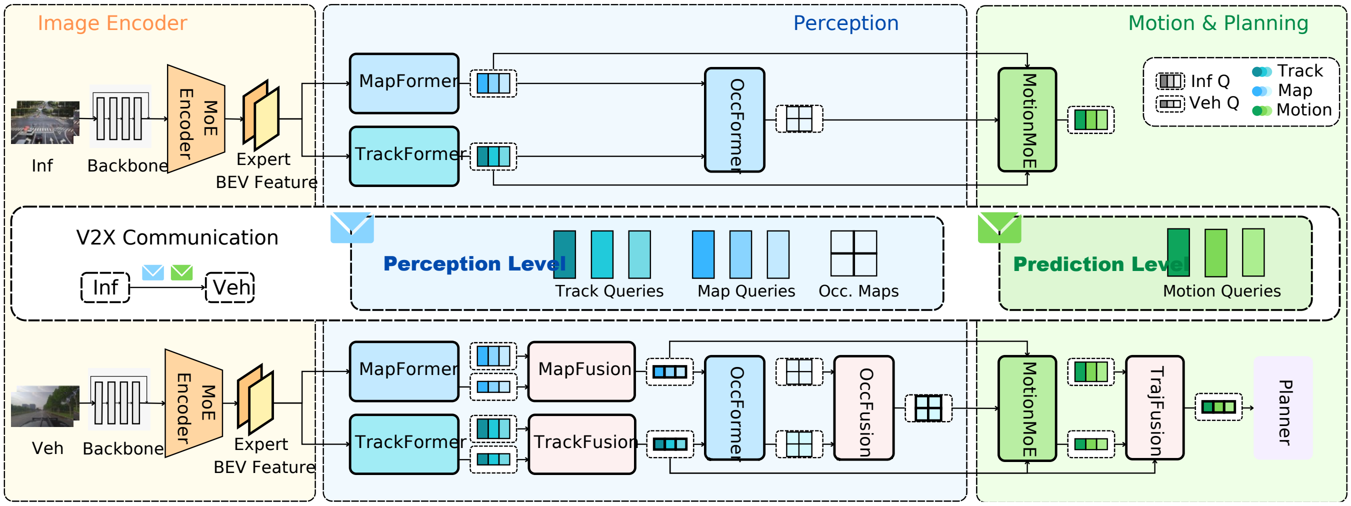


Figure 2: The overview of the UniMM-V2X framework. The system performs explicit multi-level fusion by integrating perception-level and prediction-level information from multiple agents to enhance downstream planning. Both the BEV encoder and motion decoder are equipped with MoE architectures, where the encoder generates task-adaptive BEV features tailored for various downstream tasks, and the decoder employs specialized experts to model diverse motion patterns, enabling more robust and safety-aware planning performance. This unified MoE-enhanced multi-level fusion framework facilitates effective cooperation among agents throughout the entire autonomous driving pipeline.

formance. CooperNaut (Cui et al. 2022) encodes LiDAR information into compact point-based representations, allowing it to be transmitted as messages between vehicles. However, as shown in Figure 1(b), this method is a vanilla solution, which significantly hinders the interpretability of the decision-making process. UniV2X (Yu et al. 2025), on the other hand, employs a query-based network architecture which is similar to UniAD (Hu et al. 2023), integrating sparse-dense hybrid data transmission for more efficient V2X communication. However, the fusion mechanism remains confined to the perception level, failing to fully explore the potential of multi-agent end-to-end cooperative autonomous driving, as illustrated in Figure 1(c).

Due to the complexity of end-to-end autonomous driving, relying solely on perception-level fusion is often insufficient, as accurate multi-agent motion prediction plays a more critical role in ensuring safety and efficiency. To address this, we propose UniMM-V2X, an MoE-enhanced *multi-level* fusion framework that performs cooperative information fusion at both the perception and prediction levels, addressing the VICAD problem identified in (Yu et al. 2025). At the perception level, we transmit track and map queries from different agents to facilitate cooperation and exchange the occupied probability map to support dense scene understanding. Based on this, we also transmit motion queries at the prediction level for cooperation. In each cooperation level, we employ attention-based dynamic fusion methods to ensure efficient multi-agent cooperation. Moreover, the interpretability of queries at both the instance and scene levels enhances the reliability of the system.

In additional, conventional end-to-end autonomous driving systems often fail to effectively adapt upstream modules to downstream tasks, resulting in limited flexibility and suboptimal task performance. In order to overcome this limitation, we innovatively integrate the MoE architecture into both the *BEV encoder* and *motion decoder* to address the complex joint demands of perception, prediction and planning. The MoE-enhanced encoder dynamically generates task-specialized BEV representations, supporting diverse autonomous driving tasks. Meanwhile, the MoE-equipped decoder, placed within motion prediction module for optimal performance, further dynamically generates motion queries via expert branches, each modeling distinct motion patterns such as keeping forward, turning left, or turning right, enabling more stable and safety-aware trajectory guidance for downstream planning. This design is motivated by the fact that the motion module is more tightly coupled with the final planning stage than the others.

Our contributions are summarized as follows:

- To the best of our knowledge, we are the first to explore multi-level cooperation in multi-agent end-to-end autonomous driving, enabling cooperation across both perception and prediction, thereby improving decision-making accuracy and reliability under complicated driving scenarios to a considerable extent.
- We innovatively integrate the MoE architecture into both the encoder and decoder of the end-to-end autonomous driving framework, with the encoder enhancing BEV representations to support a wide variety of autonomous driving tasks, and the decoder enabling dynamic adaptation to different motion patterns.
- We compare UniMM-V2X with both single-agent end-to-end autonomous driving systems and several existing cooperative methods, achieving SOTA performance in perception, prediction, and planning.

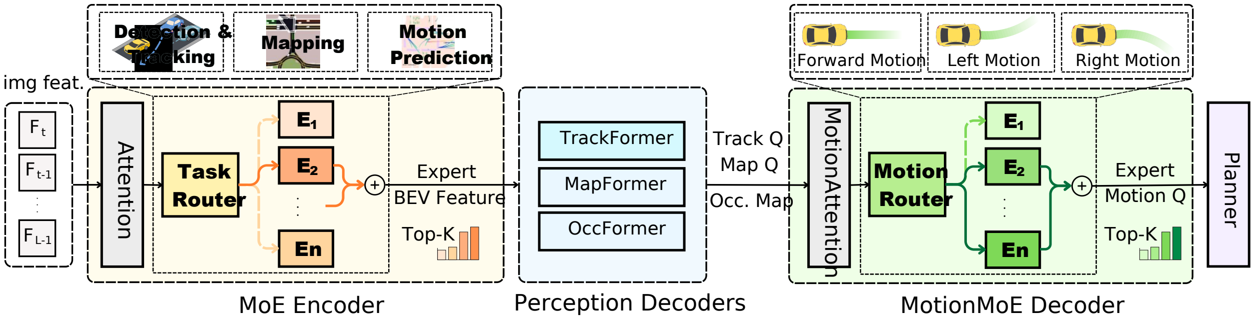


Figure 3: MoE-enhanced encoder and decoder in UniMM-V2X. The encoder enriches BEV feature extraction for diverse downstream tasks (e.g., detection, tracking, mapping, motion prediction), while the decoder generates motion queries through motion-specific experts (e.g., going forward, turning left, turning right) to improve planning quality.

Related Works

End-to-End Autonomous Driving

End-to-end autonomous driving has attracted growing attentions. Early methods (Codevilla et al. 2018, 2019; Zhang et al. 2021; Prakash, Chitta, and Geiger 2021) skip intermediate tasks, leading to poor interpretability and optimization. Recent works address this by introducing intermediate representations and unified architectures. ST-P3 (Hu et al. 2022a) builds interpretable maps from perception; UniAD (Hu et al. 2023) unifies perception, prediction, and planning via a query-based framework; VAD (Jiang et al. 2023) and SparseDrive (Sun et al. 2024) reduce computational cost through vectorization and sparse design; DiffusionDrive (Liao et al. 2025) employs diffusion models for planning. However, these methods are limited to single-agent inputs. In this work, we extend the paradigm to a multi-agent setting by incorporating cross-agent communication, joint perception and prediction fusion to achieve cooperative planning within a unified end-to-end framework.

Cooperative Autonomous Driving

V2X communication in cooperative autonomous driving has its roots in early frameworks such as Cooper (Chen et al. 2019b) and F-Cooper (Chen et al. 2019a). With the emergence of Transformer-based architectures, methods like (Liu et al. 2020; Hu et al. 2022b; Xu et al. 2022b) have improved communication strategies by learning when, where and with whom to communicate. CooperNaut (Cui et al. 2022) goes further by linking perception and control into a unified end-to-end framework, enabling direct cooperation-based decision making. UniV2X (Yu et al. 2025) introduces a sparse-dense hybrid communication protocol with cross-view interaction, effectively coordinating vehicle-to-infrastructure information and improving overall planning outcomes. However, these methods either adopt vanilla fusion strategies or perform fusion only at the perception stage, limiting their effectiveness in downstream planning. In contrast, we propose a multi-level fusion framework that operates across both perception and prediction stages, enabling agents to cooperatively reason from spatial observations to motion intents for safer and more cooperative planning.

Mixture of Experts

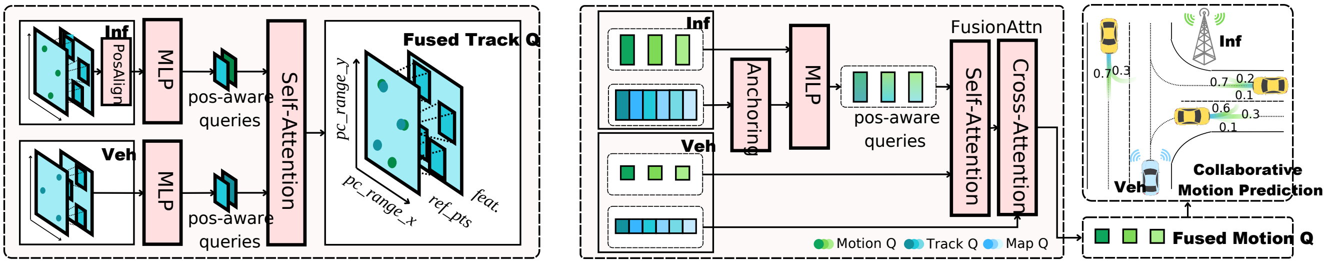
MoE operates under the principle of conditional computation with a learnable gating function selecting specialized parameters. (Shazeer et al. 2018) enables MoE-based scaling in Transformers by replacing standard feed-forward networks (FFNs) with sparsely activated expert modules. Subsequently, GShard (Lepikhin et al. 2020), Switch Transformers (Fedus, Zoph, and Shazeer 2022), and ST-MoE (Zoph et al. 2022) extend this idea to large-scale encoder-decoder architectures, addressing training instability and fine-tuning challenges. GLaM (Du et al. 2022) later explores a decoder-only formulation to improve inference efficiency. For E2E-AD system, DriveMoE (Yang et al. 2025) applies MoE to schedule sensors and guide the actions. Motivated by the potential of MoE for multi-task handling, we integrate it into both the BEV encoder and motion decoder of our model, where the encoder generates task-specialized BEV representations for multiple task transformers, while the decoder enhances trajectory prediction by routing queries to experts specialized in distinct motion patterns.

Method

Overview

The overall framework of UniMM-V2X is illustrated in Figure 2. It performs explicit *multi-level* fusion across agents by integrating information at both the perception level and the prediction level, thereby enhancing the safety and robustness of downstream planning decisions. The MoE architecture is also integrated into both the BEV encoder and the motion decoder. The encoder generates feature representations that are better adapted to the distinct needs of various tasks, while the decoder exploits expert specialization to more precisely capture diverse motion patterns, ultimately delivering more robust and planning-aware trajectory outputs.

The framework consists of three main components: image encoders, a multi-agent perception module, and a multi-agent motion and planning module. The image encoder incorporates the MoE architecture to extract task-adaptive visual features from input images of different agents, producing more expressive and flexible representations. The perception module performs cooperative detection, tracking,



(a) Perception-Level Fusion

(b) Prediction-Level Fusion

Figure 4: Multi-level fusion in UniMM-V2X. (a) Perception-level fusion introduces positional priors via reference point embeddings and uses attention-based dynamic fusion across agents. (b) Prediction-level fusion employs anchor-based embedding and dynamic fusion to support motion reasoning in complex multi-agent settings.

and mapping. The motion and planning module first generates predicted trajectories using the MoE-based motion decoder (MotionMoE), then fuses these predictions from multiple agents to produce safer and more reliable planning decisions for the ego vehicle. Together, the perception-level and prediction-level fusion form a unified multi-level fusion framework that enables effective cooperation across agents throughout the decision-making process.

Mixture of Experts Design

To effectively address the complex joint demands of perception, prediction and planning in end-to-end autonomous driving, we place the MoE architecture in both the BEV encoder and motion decoder (MotionMoE), as shown in Figure 3. We adopt a standard sparse MoE design (Shazeer et al. 2017; Fedus, Zoph, and Shazeer 2022), in which traditional FFNs are replaced by a set of expert networks. Each input token is dynamically routed to a small, specialized subset of experts via a learned gating mechanism.

Given token embeddings $x \in \mathbb{R}^{N \times d}$, the gating network produces logits $G(x) \in \mathbb{R}^{N \times E}$ for E experts. To enable sparse routing, Gumbel noise is added to the logits, and the top-k experts are stochastically selected for each token:

$$\mathrm{MoE}(x) = \sum_{i \in I_k(x)} \tilde{G}_i(x) \cdot f_i(x), \tag{1}$$

where f_i is the i-th expert, $\tilde{G}_i(x)$ the normalized routing weight, and $I_k(x)$ the selected experts. To avoid expert collapse and ensure balanced usage, we add a load balancing loss (Fedus, Zoph, and Shazeer 2022):

$$L_{\mathrm{moe}} = \lambda\,(\mathrm{Var}(p) + \mathrm{Var}(l)), \tag{2}$$

where p are the routing probabilities, l the expert loads, and λ a weighting factor, encouraging uniform expert activation.

Within the encoder, we replace the FFNs with the MoE block to enable adaptive and specialized processing:

$$\mathbf{z}^{(l+1)} = \mathrm{MoE}(\mathrm{CrossAttn}(\mathrm{SelfAttn}(\mathbf{z}^{(l)}))), \tag{3}$$

where $\mathbf{z}^{(l)}$ is the BEV feature representations at layer l, and the MoE module selectively activates top-k experts (e.g., $k = 2$) to process the attended BEV features. Similarly, in the decoder, we replace the FFNs with the MoE architecture in the MotionMoE module to generate motion queries $\mathbf{Q}_{\mathbf{M}}^{\mathrm{veh}}$ and $\mathbf{Q}_{\mathbf{M}}^{\mathrm{other}}$ that adapt to diverse motion patterns.

Multi-Agent Perception-Level Fusion

In perception-level fusion, we incorporate track fusion, map fusion, and occupancy fusion. Among them, track fusion is particularly critical, because the resulting track queries function as crucial contributing inputs for downstream tasks. To enhance their quality, we introduce *TrackFusion* module, which dynamically builds associations in perception between agents through attention, as illustrated in Figure 4(a). The designs of the map fusion and occupancy fusion modules are provided in the Appendix, where the former uses an MLP and the latter adopts a max operation.

In the TrackFusion block, an attention mechanism is employed to model complex inter-agent query relationships and perform weighted feature fusion based on learned relevance scores, overcoming the limitations of hard matching methods that rely on fixed distance thresholds in previous works. Initially, queries from other agents Q_A^{other} are transformed into the ego-vehicle's coordinate system using an MLP:

$$Q_A^{\mathrm{other}} = \mathrm{MLP}([Q_A^{\mathrm{other}}, R]), \tag{4}$$

where R is the rotation matrix. Subsequently, the reference point information P_A^{other} and P_A^{veh} are integrated as spatial contextual priors into the dynamic feature correlation learning process, as formulated below:

$$Q_A = \mathrm{MHSA}(X_A + \mathrm{MLP}(P_A)), \tag{5}$$

$$X_A = \mathrm{Concat}(Q_A^{\mathrm{veh}}, Q_A^{\mathrm{other}}), \tag{6}$$

$$P_A = \mathrm{Concat}(P_A^{\mathrm{veh}}, P_A^{\mathrm{other}}). \tag{7}$$

We employ an MLP to embed the spatial coordinates of each agent into a learnable representation. These spatial embeddings are concatenated with agent-specific queries and jointly fed into a multi-head self-attention (MHSA) mechanism. This design allows the model to capture semantic dependencies across agents while incorporating their relative spatial positions, enabling context-aware and spatially sensitive feature fusion that enhances cooperative understanding.

Cross-View Prediction-Level Fusion

In prediction-level fusion, as shown in Figure 4(b), we fuse motion queries from multiple agents through *TrajFusion* module to enable cooperative motion prediction of surrounding objects across agents, which in turn improves the performance of final planning decisions.

The fusion process begins with other agents transmitting their trajectory queries Q_M^{other} to the ego agent via inter-agent communication. To spatially align the heterogeneous trajectory data, we first transform the agent-level anchors P_{anchor}, derived from Q_A^{other}, into the coordinate frame of the ego-vehicle using the rotation matrix R :

$$P_M^{\text{other}} = \text{MLP}([P_{\text{anchor}}, R]). \quad (8)$$

The transformed positional information is then projected through an MLP for position embedding:

$$\tilde{Q}_M^{\text{other}} = \text{MLP}([Q_M^{\text{other}}, P_M^{\text{other}}]). \quad (9)$$

The ego-agent motion queries and the positionally enhanced queries from other agents are then concatenated and processed via an attention-based mechanism:

$$F_M = \text{Concat}(Q_M^{\text{veh}}, \tilde{Q}_M^{\text{other}}), \quad (10)$$

$$Q_M = \text{MHCA}(\text{MHSA}(F_M), Q_A). \quad (11)$$

Here, the MHSA component captures intra-agent contextual dependencies within the combined motion queries F_M , allowing the model to identify salient behaviors and motion patterns. Then the multi-head cross-attention (MHCA) incorporates perception-aware context by attending to the fused perception queries Q_A . These perception queries are historically enriched and semantically aligned, thereby providing a strong contextual prior that guides motion reasoning under complex multi-agent interactions.

Learning

The overall training objective is to jointly optimize multiple sub-tasks involved in end-to-end autonomous driving. Specifically, the loss function consists of six components: detection and tracking, online mapping, occupancy prediction, motion prediction, trajectory planning, and the auxiliary load balancing term introduced by the MoE module.

$$L = L_{\text{track}} + L_{\text{map}} + L_{\text{occ}} + L_{\text{mot}} + L_{\text{plan}} + L_{\text{moe}}. \quad (12)$$

All components are jointly optimized in an end-to-end manner to achieve unified perception, prediction, and planning.

Experiments

Experimental Settings

The overall framework is trained with the challenging DAIR-V2X dataset (Yu et al. 2022), which comprises approximately 100 scenes captured at 28 complex traffic intersections in the real world. The perception range of the ego vehicle is $[-51.2m, -51.2m, 51.2m, 51.2m]$ while that of the infrastructure is $[0, -51.2m, 102.4m, 51.2m]$ We use the AdamW optimizer with a learning rate of 1×10^{-4} and a weight decay of 0.01. We train the tracking stage for 40 epochs on 8 NVIDIA A800 GPUs, and subsequently perform end-to-end motion and planning training for 20 epochs using the same GPU setup. During training, the MoE layers select the top-2 experts for each token to balance specialization and computational efficiency. Evaluation metrics of each task are described in the Appendix. We also implement UniMM-V2X on the V2X-Sim dataset (Li et al. 2022), a large-scale simulation benchmark with diverse traffic scenarios for cooperative autonomous driving, and implement details and results are provided in the Appendix.

Main Results

We compare UniMM-V2X against a range of baselines, including several single-agent end-to-end autonomous driving models (Jiang et al. 2023; Hu et al. 2023; Sun et al. 2024), as well as multi-agent cooperative driving frameworks. For the cooperative baselines, we evaluate both cooperative perception methods (Lu et al. 2023; Hu et al. 2022b; Xu et al. 2022a,b) and fully end-to-end cooperative driving approaches (Cui et al. 2022; Yu et al. 2025). To facilitate comparison, the primary evaluation metrics are highlighted with a gray background in the result table.

Planning. The planning results are summarized in Table 1. Our method achieves the lowest average L2 error of **1.49m**, reducing by **33.2%** compared with UniV2X (Yu et al. 2025), outperforming all the baselines, including advanced single-agent methods and existing V2X approaches. More importantly, UniMM-V2X demonstrates superior safety, attaining the lowest average collision rate of **0.12%**, which represents a **52.0%** reduction compared to UniV2X (Yu et al. 2025), significantly mitigating potential driving risks. Although our approach introduces slightly higher communication overhead due to the transmission of motion queries, the improvements in safety performance clearly justify the additional cost.

Perception. Table 2 presents the performance of UniMM-V2X on perception tasks. Compared to the SOTA no-fusion baseline (Sun et al. 2024), our method achieves a **+0.098** improvement in mAP and a **+0.297** improvement in AMOTA, demonstrating the effectiveness of cooperation. We also compare our approach with several cooperative perception methods (Lu et al. 2023; Hu et al. 2022b; Xu et al. 2022a,b), and observe significantly better performance. Compared to the SOTA end-to-end cooperative driving framework (Yu et al. 2025), our method achieves an improvement of **39.7%** in mAP and **77.2%** in AMOTA, without introducing additional communication cost at the perception level. For occupancy tasks, as shown in Table 3, UniMM-V2X improves IoU-n by **+0.8%** compared to UniV2X (Yu et al. 2025).

Prediction. The motion prediction results are shown in Table 4. UniMM-V2X achieves the best performance with **0.64m** minADE, **0.69m** minFDE and **13.2%** MissRate, reducing errors by **7.2%** and **6.8%** on minADE and minFDE respectively compared with UniV2X (Yu et al. 2025). These improvements contribute significantly to the improvement of the final planning performance mentioned above.

Method	L2 Error (m)↓				Collision Rate (%)↓				Trans. Cost
	1*s*	2*s*	3*s*	Avg.	1*s*	2*s*	3*s*	Avg.	(BPS)↓
VAD* (Jiang et al. 2023)	1.65	2.72	3.80	2.72	0.86	1.21	1.28	1.12	-
UniAD* (Hu et al. 2023)	1.26	2.22	3.06	2.18	0.88	1.18	1.32	1.13	-
SparseDrive* (Sun et al. 2024)	1.02	1.69	2.37	1.69	0.46	1.23	1.28	0.99	-
Vanilla	1.36	2.29	3.32	2.32	1.03	0.88	1.32	1.08	8.19×10^7
V2VNet (Wang et al. 2020)	1.96	2.37	3.41	2.58	0.74	0.88	1.03	0.88	8.19×10^7
CooperNaut (Cui et al. 2022)	2.69	4.07	5.50	4.09	1.18	1.32	1.76	1.42	8.19×10^7
UniV2X (Yu et al. 2025)	1.45	2.19	3.04	2.23	0.15	**0.15**	0.44	0.25	$8.09\boldsymbol{\times10^5}$
UniMM-V2X	**0.78**	**1.63**	**2.05**	**1.49**	**0.05**	**0.15**	**0.15**	**0.12**	$9.32\boldsymbol{\times10^5}$

Table 1: **Planning** performance. *: Single-agent no fusion method. We achieve improvements in reducing L2 error and collision rate, enhancing overall system safety.

Method	Detection	Tracking	Mapping		Trans. Cost
	mAP↑	AMOTA↑	Lane (%)↑	Crossing (%)↑	(BPS)↓
UniAD* (Hu et al. 2023)	0.181	0.197	13.3	8.7	-
SparseDrive* (Sun et al. 2024)	0.324	0.130	-	5.2	-
Early Fusion	0.243	0.209	16.7	17.8	8.19×10^7
Late Fusion	0.236	0.263	13.4	9.1	$\mathbf{6.60}\times10^2$
CoAlign† (Lu et al. 2023)	0.261	0.234	-	-	8.19×10^7
Where2comm† (Hu et al. 2022b)	0.221	0.106	-	-	5.40×10^5
CoBEVT† (Xu et al. 2022a)	0.264	0.243	15.6	16.4	2.56×10^6
V2X-ViT† (Xu et al. 2022b)	0.261	0.287	-	-	2.56×10^6
UniV2X (Yu et al. 2025)	0.302	0.241	17.7	19.7	2.17×10^5
UniMM-V2X	**0.422**	**0.427**	**17.9**	**20.3**	2.17×10^5

Table 2: **Perception** performance. *: Single-agent no fusion method. †: Cooperative perception methods. We significantly improve all performance metrics without increasing transmission cost.

Ablation Study

We conduct ablation studies on the effectiveness of the multi-level fusion and the MoE design. The results are shown in Table 5 and Table 6.

Effect of Multi-Level Fusion. As shown in Table 5, perception-level fusion significantly improves the performance of both detection (+0.187) and tracking (+0.160). This is because the perception information from the infrastructure effectively complements that of the ego vehicle. However, the performance of motion prediction and planning does not exhibit a similar improvement, and in some cases even degrades. This may be attributed to the fact that the goals of accurate perception and safe decision-making are not always fully aligned. On the other hand, incorporating only prediction-level fusion enhances planning safety with -0.33m for L2 Error and -0.09% for Collision Rate, as infrastructure can supplement motion information for objects that are not visible to the ego vehicle, and refine low-confidence predictions. Nevertheless, perception performance remains similar to the single-agent baseline, since no fusion is applied at the perception stage.

To ensure consistent performance gains throughout the entire pipeline, multi-level fusion is essential. Our results confirm that this strategy delivers substantial improvements across all tasks including perception, prediction and planning, highlighting the strength of multi-level fusion in delivering end-to-end performance gains.

Effect and Configuration of MoE. As shown in Table 5, integrating MoE into the BEV encoder significantly enhances environmental understanding, improving both perception and planning performance. Using MoE only in the motion decoder yields limited gains, likely due to insufficient task-specific BEV features. The best results occur when MoE is applied to both encoder and decoder, with the encoder providing task-adaptive BEV representations that better support downstream modules.

Since all decoder modules are Transformer-based, any FFNs in these modules can be replaced with MoE. We conduct ablation studies under our end-to-end cooperative autonomous driving framework with multi-level fusion to determine the optimal expert number and placement. All variants employ the MoE-based encoder, where spatially specialized experts consistently enhance performance. For the perception decoder, MoE is applied only in TrackFormer,

Method	IoU-n (%)↑	IoU-f (%)↑
UniAD* (Hu et al. 2023)	16.3	13.1
UniV2X (Yu et al. 2025)	22.2	**26.0**
UniMM-V2X	**23.0**	23.7

Table 3: **Occupancy prediction** performance. "n" and "f" denote near (30×30m) and far (50×50m) ranges. *: Single-agent no fusion method.

Method	minADE (m)↓	minFDE (m)↓	MR↓
UniAD* (Hu et al. 2023)	0.78	0.82	0.21
SparseDrive* (Sun et al. 2024)	1.02	1.87	0.34
UniV2X (Yu et al. 2025)	0.69	0.74	0.17
UniMM-V2X	**0.64**	**0.69**	**0.13**

Table 4: **Moion prediction** performance. *: Single-agent no fusion method.

Multi-Level Fusion		MoE		Perception		Motion Prediction	Planning L2 Error (m)				Coll. (%)
P-Level	M-Level	Enc.	Dec.	mAP↑	AMOTA↑	minADE (m)↓	1*s*	2*s*	3*s*	Avg.↓	Avg.↓
-	-	-	-	0.181	0.197	0.78	1.26	2.22	3.06	2.18	1.13
-	-	✓	-	0.238	0.269	0.81	1.28	1.97	2.92	2.06	0.39
-	-	-	✓	0.179	0.198	0.78	1.24	1.84	2.98	2.02	0.54
-	-	✓	✓	0.240	0.267	0.75	1.02	1.73	2.82	1.85	0.24
✓	-	✓	✓	**0.427**	**0.427**	0.74	0.91	1.78	2.47	1.72	0.40
-	✓	✓	✓	0.238	0.271	<u>0.65</u>	0.96	**1.53**	2.08	<u>1.52</u>	<u>0.15</u>
✓	✓	✓	✓	<u>0.422</u>	**0.427**	**0.64**	**0.78**	1.63	**2.05**	**1.49**	**0.12**

Table 5: **Ablation study results.** We conduct experiments to evaluate the effectiveness of multi-level fusion and the MoE mechanism. P-Level and M-Level refer to the perception level fusion and motion prediction level fusion, while Enc. and Dec. indicate applying MoE to the BEV encoder and motion decoder, respectively.

Num.	P	M	mAP↑	AMOTA↑	L2 (m)↓	CR (%)↓
4	-	-	0.240	0.229	1.77	0.39
4	✓	-	0.235	0.230	1.85	0.34
4	-	✓	0.237	0.231	1.60	0.20
4	✓	✓	0.230	0.229	1.78	0.37
8	-	-	<u>0.421</u>	<u>0.425</u>	<u>1.66</u>	<u>0.20</u>
8	✓	-	0.366	0.347	1.75	0.43
8	-	✓	**0.422**	**0.427**	**1.49**	**0.12**
8	✓	✓	0.368	0.351	1.68	0.25
16	-	-	0.403	0.374	1.71	0.36
16	✓	-	0.359	0.341	1.76	0.41
16	-	✓	0.401	0.374	1.64	0.15
16	✓	✓	0.361	0.339	1.72	0.20

Table 6: **Ablation on MoE expert number and decoder placement** conducted within the multi-level fusion framework. All the experiments utilize the MoE-based encoder. "P" indicates MoE applied to the perception decoder and "M" denotes its placement in the motion decoder.

since its track queries impact motion prediction more than other decoders like MapFormer or OccFormer. As shown in Table 6, using 8 experts and placing MoE in the motion decoder achieves the best results. Regarding the number of experts, too few experts limit the specialization, while too many experts cause data sparsity and under-utilization. Additionally, in terms of decoder placement, limited token diversity in the perception decoder reduces MoE benefits and may introduce gating noise. In contrast, placing MoE in the motion decoder, which is more tightly coupled with the final planner, enables better adaptation to diverse future behaviors, leading to more flexible and accurate planning.

Conclusion

In this work, we proposed UniMM-V2X, an end-to-end framework designed for robust multi-agent cooperative driving. UniMM-V2X innovates by explicitly fusing information at both perception and prediction levels across agents. To effectively handle the inherent heterogeneity and dynamic nature of such cooperation, we integrated MoE modules into both the BEV encoder and the motion decoder, enabling adaptive processing optimized for diverse driving tasks and motion patterns. Extensive evaluations on the DAIR-V2X benchmark demonstrate the effectiveness of our approach: UniMM-V2X achieves SOTA performance with significant improvements across the autonomous driving pipeline. Critically, this includes substantial gains in detection (39.7% improvement), tracking accuracy (77.2% increase), motion prediction (7.2% reduction in error), and notably enhanced planning safety (33.2% reduction in L2 error and 52.0% lower collision rate) compared to UniV2X. These results strongly validate the effectiveness of MoE-enhanced multi-level fusion in providing a flexible and scalable solution for complex cooperative driving scenarios. The current evaluation is limited to an open-loop setting, which cannot fully capture the impact of downstream feedback and control interactions. Future work will focus on extending UniMM-V2X to closed-loop evaluation, developing more communication-efficient fusion strategies, and further enhancing the robustness and scalability of multi-agent coordination under real-world constraints.

References

Chen, Q.; Ma, X.; Tang, S.; Guo, J.; Yang, Q.; and Fu, S. 2019a. F-cooper: Feature based cooperative perception for autonomous vehicle edge computing system using 3D point clouds. In *Proceedings of the 4th ACM/IEEE Symposium on Edge Computing*, 88–100.

Chen, Q.; Tang, S.; Yang, Q.; and Fu, S. 2019b. Cooper: Cooperative perception for connected autonomous vehicles based on 3d point clouds. In *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, 514–524. IEEE.

Codevilla, F.; Müller, M.; López, A.; Koltun, V.; and Dosovitskiy, A. 2018. End-to-end driving via conditional imitation learning. In *2018 IEEE international conference on robotics and automation (ICRA)*, 4693–4700. IEEE.

Codevilla, F.; Santana, E.; López, A. M.; and Gaidon, A. 2019. Exploring the limitations of behavior cloning for autonomous driving. In *Proceedings of the IEEE/CVF international conference on computer vision*, 9329–9338.

Cui, J.; Qiu, H.; Chen, D.; Stone, P.; and Zhu, Y. 2022. Coopernaut: End-to-end driving with cooperative perception for networked vehicles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 17252–17262.

Du, N.; Huang, Y.; Dai, A. M.; Tong, S.; Lepikhin, D.; Xu, Y.; Krikun, M.; Zhou, Y.; Yu, A. W.; Firat, O.; et al. 2022. Glam: Efficient scaling of language models with mixture-of-experts. In *International conference on machine learning*, 5547–5569. PMLR.

Fedus, W.; Zoph, B.; and Shazeer, N. 2022. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120): 1–39.

Hu, S.; Chen, L.; Wu, P.; Li, H.; Yan, J.; and Tao, D. 2022a. St-p3: End-to-end vision-based autonomous driving via spatial-temporal feature learning. In *European Conference on Computer Vision*, 533–549. Springer.

Hu, Y.; Fang, S.; Lei, Z.; Zhong, Y.; and Chen, S. 2022b. Where2comm: Communication-efficient collaborative perception via spatial confidence maps. *Advances in neural information processing systems*, 35: 4874–4886.

Hu, Y.; Yang, J.; Chen, L.; Li, K.; Sima, C.; Zhu, X.; Chai, S.; Du, S.; Lin, T.; Wang, W.; et al. 2023. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 17853–17862.

Jia, X.; You, J.; Zhang, Z.; and Yan, J. 2025. Drivetransformer: Unified transformer for scalable end-to-end autonomous driving. *arXiv preprint arXiv:2503.07656*.

Jiang, B.; Chen, S.; Xu, Q.; Liao, B.; Chen, J.; Zhou, H.; Zhang, Q.; Liu, W.; Huang, C.; and Wang, X. 2023. Vad: Vectorized scene representation for efficient autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 8340–8350.

Lepikhin, D.; Lee, H.; Xu, Y.; Chen, D.; Firat, O.; Huang, Y.; Krikun, M.; Shazeer, N.; and Chen, Z. 2020. Gshard: Scaling giant models with conditional computation and automatic sharding. *arXiv preprint arXiv:2006.16668*.

Li, Y.; Ma, D.; An, Z.; Wang, Z.; Zhong, Y.; Chen, S.; and Feng, C. 2022. V2X-Sim: Multi-agent collaborative perception dataset and benchmark for autonomous driving. *IEEE Robotics and Automation Letters*, 7(4): 10914–10921.

Li, Z.; Wang, W.; Li, H.; Xie, E.; Sima, C.; Lu, T.; Yu, Q.; and Dai, J. 2024. Bevformer: learning bird's-eye-view representation from lidar-camera via spatiotemporal transformers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Liao, B.; Chen, S.; Yin, H.; Jiang, B.; Wang, C.; Yan, S.; Zhang, X.; Li, X.; Zhang, Y.; Zhang, Q.; et al. 2025. Diffusiondrive: Truncated diffusion model for end-to-end autonomous driving. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 12037–12047.

Liu, Y.-C.; Tian, J.; Ma, C.-Y.; Glaser, N.; Kuo, C.-W.; and Kira, Z. 2020. Who2com: Collaborative perception via learnable handshake communication. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 6876–6883. IEEE.

Lu, Y.; Li, Q.; Liu, B.; Dianati, M.; Feng, C.; Chen, S.; and Wang, Y. 2023. Robust collaborative 3d object detection in presence of pose errors. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 4812–4818. IEEE.

Philion, J.; and Fidler, S. 2020. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*, 194–210. Springer.

Prakash, A.; Chitta, K.; and Geiger, A. 2021. Multi-modal fusion transformer for end-to-end autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 7077–7087.

Shazeer, N.; Cheng, Y.; Parmar, N.; Tran, D.; Vaswani, A.; Koanantakool, P.; Hawkins, P.; Lee, H.; Hong, M.; Young, C.; et al. 2018. Mesh-tensorflow: Deep learning for supercomputers. *Advances in neural information processing systems*, 31.

Shazeer, N.; Mirhoseini, A.; Maziarz, K.; Davis, A.; Le, Q.; Hinton, G.; and Dean, J. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*.

Sun, W.; Lin, X.; Shi, Y.; Zhang, C.; Wu, H.; and Zheng, S. 2024. Sparsedrive: End-to-end autonomous driving via sparse scene representation. *arXiv preprint arXiv:2405.19620*.

Wang, T.-H.; Manivasagam, S.; Liang, M.; Yang, B.; Zeng, W.; and Urtasun, R. 2020. V2vnet: Vehicle-to-vehicle communication for joint perception and prediction. In *Computer vision–ECCV 2020: 16th European conference, Glasgow, UK, August 23–28, 2020, proceedings, part II 16*, 605–621. Springer.

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

Xu, R.; Xiang, H.; Tu, Z.; Xia, X.; Yang, M.-H.; and Ma, J. 2022b. V2x-vit: Vehicle-to-everything cooperative perception with vision transformer. In *European conference on computer vision*, 107–124. Springer.

Yang, Z.; Chai, Y.; Jia, X.; Li, Q.; Shao, Y.; Zhu, X.; Su, H.; and Yan, J. 2025. DriveMoE: Mixture-of-Experts for Vision-Language-Action Model in End-to-End Autonomous Driving. arXiv:2505.16278.

Yu, H.; Luo, Y.; Shu, M.; Huo, Y.; Yang, Z.; Shi, Y.; Guo, Z.; Li, H.; Hu, X.; Yuan, J.; et al. 2022. Dair-v2x: A large-scale dataset for vehicle-infrastructure cooperative 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 21361–21370.

Yu, H.; Yang, W.; Zhong, J.; Yang, Z.; Fan, S.; Luo, P.; and Nie, Z. 2025. End-to-end autonomous driving through v2x cooperation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 9598–9606.

Zeng, W.; Luo, W.; Suo, S.; Sadat, A.; Yang, B.; Casas, S.; and Urtasun, R. 2019. End-to-end interpretable neural motion planner. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 8660–8669.

Zhang, Z.; Liniger, A.; Dai, D.; Yu, F.; and Van Gool, L. 2021. End-to-end urban driving by imitating a reinforcement learning coach. In *Proceedings of the IEEE/CVF international conference on computer vision*, 15222–15232.

Zoph, B.; Bello, I.; Kumar, S.; Du, N.; Huang, Y.; Dean, J.; Shazeer, N.; and Fedus, W. 2022. St-moe: Designing stable and transferable sparse expert models. *arXiv preprint arXiv:2202.08906*.

Technical Appendix

Implementation Details

MoE Settings. The architecture of both the encoder and decoder in our model consists of 6 layers of MoE, providing sufficient depth to capture spatial and temporal dependencies across multi-agent observations. Each MoE layer is equipped with 8 experts, employing top-2 expert routing to dynamically select relevant experts. Furthermore, we introduce a load balancing loss weight of $\lambda = 0.03$ to ensure a balanced utilization of all experts. This setup enables adaptive specialization in processing tasks, improving the overall efficiency and performance of the model.

Perception. The perception decoder is configured with 6 layers, and the perception range is set to a radius of 50 meters around the ego vehicle. We use a tracking threshold of 0.25, which is the minimum matching confidence required to associate objects across consecutive frames. A relatively low threshold ensures robustness when dealing with occlusion or partial views in challenging environments.

For the perception-level fusion, we focus on three key components: TrackFusion, MapFusion and OccFusion. TrackFusion, which plays a critical role in downstream tasks, is detailed in the main text. For MapFusion, we fuse map queries using an MLP network:

$$Q_L = \mathrm{MLP}([Q_L^{\mathrm{veh}}, \tilde{Q}_L^{\mathrm{other}}]), \tag{1}$$

where $\tilde{Q}_L^{\mathrm{other}}$ is transformed into the coordinate frame of the ego vehicle. This transformation ensures that the map queries are aligned with the perspective of the ego vehicle. For OccFusion, aligned occupancy maps from different agents are combined using the following a max operation:

$$\mathbf{P} = \max(\mathbf{P}^{\mathrm{veh}}, \tilde{\mathbf{P}}^{\mathrm{other}}), \tag{2}$$

$$O(x, y) = \begin{cases} 1, & \mathbf{P}(x, y) > \tau \\ 0, & \text{otherwise} \end{cases}, \tag{3}$$

where $\tilde{\mathbf{P}}^{\mathrm{other}}$ is transformed into the view of the ego vehicle via a coordinate transformation, and $\tau = 0.1$ is the threshold used for occupancy determination in our experiments.

Prediction. In the prediction module, we define 6 motion modes per target, which allows the model to account for the uncertainty and multi-modality of future behavior. This enhances the ability of the model to predict a range of complex future trajectories. For trajectory-related tasks, the model uses 4 past steps and predicts the next 12 steps in the future.

Comparative Experiments. To evaluate the effectiveness of our method, we conduct a series of comparative experiments across several fusion strategies and SOTA cooperative perception methods. We perform evaluations on the DAIR-V2X (Yu et al. 2022) and V2X-Sim (Li et al. 2022) datasets, using the following fusion strategies:

- **No Fusion:** No fusion of infrastructure data is performed. For this baseline, we compare against the UniAD (Hu et al. 2023) framework as well as the VAD (Jiang et al. 2023) and SparseDrive (Sun et al. 2024) single-vehicle end-to-end SOTA algorithms to demonstrate the importance of multi-agent cooperation.
- **Early Fusion:** Raw data from the infrastructure is directly fused at the initial stage, following the setup from UniV2X (Yu et al. 2025).
- **Late Fusion:** Infrastructure perception results are fused using the Hungarian method (Kuhn 1955).
- **Intermediate Fusion:** We reproduce current SOTA cooperative perception methods, including V2X-ViT (Xu et al. 2022b), Where2comm (Hu et al. 2022b), CoBEVT (Xu et al. 2022a), and CoAlign (Lu et al. 2023). For a fair comparison, we use only the image data from the infrastructure and standardize evaluation settings.

For planning comparisons, we evaluate the performance of several representative baselines and fusion strategies. The vanilla method generates the final trajectory by simply fusing BEV features and passing them through an MLP to generate final planning results. For the V2VNet (Wang et al. 2020) and CooperNaut (Cui et al. 2022) methods, we use approaches similar to UniV2X (Yu et al. 2025), where the BEV features are fused and passed into the UniAD (Hu et al. 2023) framework for further processing. The results for UniV2X (Yu et al. 2025) are directly reproduced from the official checkpoints for comparison.

Training. Our training process is divided into four stages to ensure efficient learning of cooperative driving:

- Stage 1: Pre-train the perception module of the infrastructure, which includes tasks such as detection, tracking and mapping.
- Stage 2: Pre-train the perception part of the ego vehicle.
- Stage 3: Introduce cooperation into the system and train the cooperative perception tasks between the ego vehicle and other agents.
- Stage 4: Freeze the perception modules and focus on training the cooperative prediction and planning tasks, followed by fine-tuning to optimize overall performance.

Others. The temporal query length is set to 5, meaning the model processes information from the past five frames. This enables the model to leverage historical context for more stable object association and trajectory prediction. To balance task-specific performance with computational efficiency, we set the number of queries for tracking, mapping, and motion prediction to 1500, 300, and 500, respectively. The tracking queries ensure high recall in dense multi-object tracking scenarios, while the mapping and motion branches benefit from more compact and focused query sets, optimizing their respective tasks.

Experiments

Evaluation Metrics

We comprehensively evaluate our framework across six key tasks relevant to autonomous driving: object detection,

Method	L2 Error (m)↓				Collision Rate (%)↓			
	1*s*	2*s*	3*s*	Avg.	1*s*	2*s*	3*s*	Avg.
No Fusion	2.87	3.66	4.78	3.77	1.22	1.33	1.33	1.29
UniV2X	2.44	2.92	3.90	3.09	1.00	0.88	**0.77**	0.88
UniMM-V2X	**2.00**	**2.22**	**3.16**	**2.46**	**0.66**	**0.77**	1.00	**0.81**

Table 1: Planning performance on V2X-Sim dataset (Li et al. 2022).

multi-object tracking, online mapping, occupancy prediction, motion prediction, and planning, along with transmission cost to reflect communication overhead in the cooperative autonomous driving settings.

Object Detection and Multi-Object Tracking. Following the standard NuScenes protocol (Caesar et al. 2020), we evaluate detection performance using mean Average Precision (mAP) based on the n-points interpolated precision-recall curve, which quantifies the average overlap between the predicted and ground-truth bounding boxes across different thresholds:

$$\text{mAP} = \frac{1}{n-1}\sum_{r\in\frac{1}{n-1},\frac{2}{n-1}...,1.0}\text{AP}_r, \tag{4}$$

$$\text{AP}_r = \frac{\text{TP}_r}{\text{TP}_r + \text{FP}_r + \text{FN}_r}, \tag{5}$$

where TP_r, FP_r, and FN_r represent true positives, false positives, and false negatives at IoU threshold r.

For tracking, we employ Average Multi-Object Tracking Accuracy (AMOTA), which captures both association accuracy and detection quality over recall levels:

$$\text{AMOTA} = \frac{1}{n-1}\sum_{r\in\frac{1}{n-1},\frac{2}{n-1},...,1.0}\text{MOTA}_r, \tag{6}$$

$$\text{MOTA} = \max(0, 1 - \frac{\text{FN}_r + \text{FP}_r + \text{IDSW}_r - (1-r)\text{GT}}{r\text{GT}}), \tag{7}$$

where IDSW_r denotes identity switches, and GT is the total number of ground-truth objects. Our detection and tracking experiments focus on the ”Car” class, which is the most safety-critical in autonomous driving scenarios.

Online Mapping. We assess the quality of high-definition map prediction using the IoU between predicted and ground-truth semantic map elements (e.g., lane markings, pedestrian crossings) from a BEV perspective:

$$\text{IoU} = \frac{|\text{Prediction} \cap \text{Ground Truth}|}{|\text{Prediction} \cup \text{Ground Truth}|}. \tag{8}$$

This metric reflects the accuracy and completeness of online semantic map generation of the model.

Occupancy Prediction. We follow UniAD (Hu et al. 2023) and measure scene-level semantic segmentation performance using IoU across two spatial ranges: near (30×30m) and far (50×50m). The metric evaluates the model's ability to differentiate between occupied and free space covering both static infrastructure (e.g., buildings, sidewalks) and dynamic agents (e.g., vehicles, pedestrians).

Motion Prediction. We evaluate future trajectory forecasting using the following metrics:

- Minimum Average Displacement Error (minADE): average ℓ_2 distance between predicted and ground-truth points over the best-matching trajectory.
- Minimum Final Displacement Error (minFDE): ℓ_2 distance at the final prediction timestamp.
- Miss Rate (MR): percentage of ground-truth trajectories that deviate beyond a threshold δ (typically 2m) from all predicted modes.

Formally, for K predicted trajectories $\hat{Y}^k$ and a ground-truth Y, we compute:

$$\text{minADE} = \min_k \frac{1}{T}\sum_{t=1}^{T} \|\hat{Y}_t^k - Y_t\|_2, \tag{9}$$

$$\text{minFDE} = \min_k \|\hat{Y}_T^k - Y_T\|_2. \tag{10}$$

Planning. We assess planning accuracy and safety using two primary metrics:

- **L2 Error:** average ℓ_2 distance between predicted and expert trajectories over time.
- **Collision Rate:** the percentage of predicted trajectories that result in collisions with obstacles or other agents.

We report these metrics at $1s$, $2s$, and $3s$ prediction horizons to reflect both short-term and long-term planning quality. The settings are the same as in ST-P3 (Hu et al. 2022a).

Transmission Cost. To evaluate communication efficiency, we measure the bandwidth consumption using Bytes Per Second (BPS). Following the protocol of UniV2X (Yu et al. 2025), we assume that the communication frequency is 2 Hz and that all transmitted data including features and queries are serialized using 32-bit floats. Calibration matrices and ego-pose data are excluded from transmission cost computation. BPS reflects the total bandwidth required to support cooperation, and enables a trade-off analysis between performance and communication overhead. The slight increase in communication overhead of UniMM-V2X compared to UniV2X (Yu et al. 2025) is due to the introduction of motion prediction level fusion.

Results on V2X-Sim Dataset

We also implement our proposed UniMM-V2X framework on the V2X-Sim dataset (Li et al. 2022), which is crucial to evaluate the robustness of our model across various simulated cooperative driving scenarios. The V2X-Sim dataset offers a valuable benchmark to test the performance of our approach in a simulated environment with multiple cooperative agents.

V2X-Sim Dataset. The V2X-Sim dataset is a large-scale simulated dataset built on the CARLA platform, specifically designed for V2X-based cooperative autonomous driving research. It consists of 100 diverse urban driving scenes, which include a range of driving environments with varying levels of complexity and dynamic scenarios. Each scene features multiple connected vehicles equipped with multi-view cameras, LiDAR sensors, and high-definition (HD) maps. The dataset provides rich annotations for tasks such as object detection, multi-object tracking, motion prediction, and planning. These characteristics make V2X-Sim an ideal dataset for evaluating end-to-end multi-agent cooperation and assessing the effectiveness of different cooperative strategies in realistic driving contexts.

Method	minADE (m) ↓	minFDE (m) ↓
No Fusion	0.79	0.97
UniV2X (Yu et al. 2025)	0.76	0.92
UniMM-V2X	**0.64**	**0.75**

Table 2: Motion prediction performance on the V2X-Sim dataset (Li et al. 2022).

Experiment Settings and Results. In our experiments, we select two vehicles from the V2X-Sim dataset: one is designated as the ego vehicle, and the other as the cooperating agent. To ensure a fair comparison with existing methods, we use only the front-view camera data.

For training, we use a total of 80 scenes from the V2X-Sim dataset, ensuring a diverse range of driving environments. We use 10 scenes for validation and another 10 scenes for testing to evaluate the generalization ability of the model across different scenarios. In the original UniV2X (Yu et al. 2025) framework, the planning evaluation follows a non-standard timing protocol. To maintain consistency and fairness in our evaluation, we adopt the standard evaluation protocol at $1s$, $2s$, and $3s$ intervals, which is widely used in the field to assess the accuracy and safety of autonomous driving systems.

The performance results for planning and motion prediction tasks are summarized in Table 1 and Table 2. These results show that, compared to the SOTA end-to-end multi-agent cooperative approach UniV2X (Yu et al. 2025), our UniMM-V2X framework achieves significant improvements in key performance metrics. Specifically, our model reduces the L2 error by **20.39%**, indicating a marked improvement in the accuracy of the planned trajectories. Additionally, the collision rate is lowered by **7.95%**, which demonstrates a considerable enhancement in driving safety by minimizing risky behaviors in complex driving scenarios.

Furthermore, in terms of motion prediction, we observe a **15.8%** reduction in the motion prediction error, which is crucial for improving planning performance. More accurate motion predictions provide the planner with better estimates of future trajectories, thereby enabling more informed and precise decision-making in dynamic and complex driving environments. This substantial improvement in prediction accuracy directly contributes to the overall safety and efficiency of the cooperative driving system, reinforcing the importance of accurate multi-agent cooperation.

Method	Mem. (MB)↓	FPS ↑	Trans. (MB)↓
UniV2X	6301	5.86	8.09×10^5
UniMM-V2X	6483	5.39	9.32×10^5

Table 3: Inference complexity and transmission cost.

Efficiency of the System

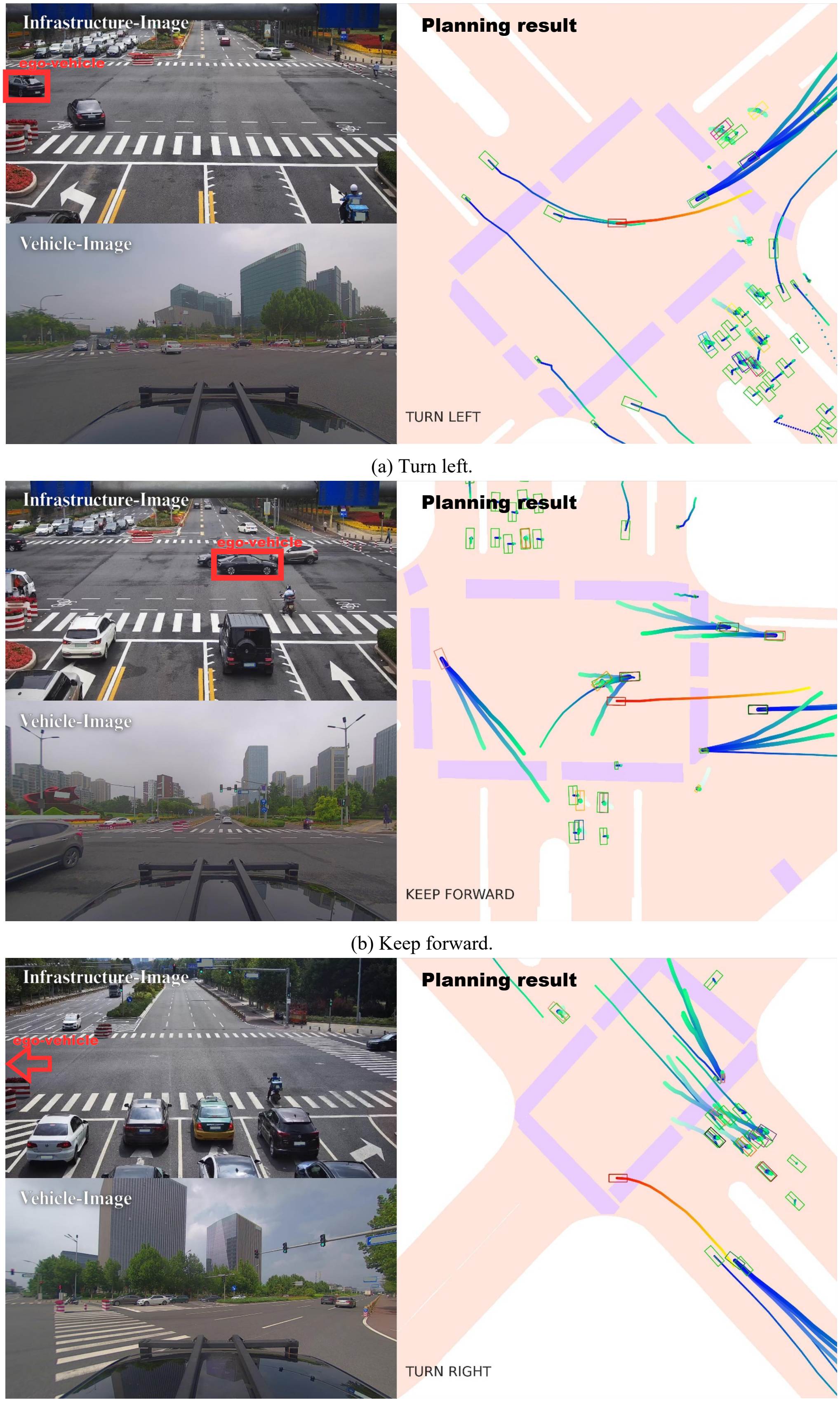
To assess the practicality and efficiency of our proposed method, we compare the inference complexity and communication cost with UniV2X (Yu et al. 2025), as summarized in Table 3. Specifically, we report GPU memory usage (Mem.), processing speed in frames per second (FPS), and the total amount of data transmitted across agents (Trans.). These metrics are critical in understanding the trade-offs between computational efficiency and model performance in real-time multi-agent cooperative driving scenarios.

Overall, UniMM-V2X achieves competitive runtime performance while introducing moderate increases in memory and communication overhead. Compared to the baseline UniV2X, our full model increases memory usage by approximately 2.9% and reduces FPS by 0.47, primarily due to the incorporation of the MoE architecture and multi-level fusion across agents. This additional computational overhead, however, is offset by substantial improvements in downstream performance, as demonstrated in the main text. The introduction of the MoE layer enables adaptive processing with specialized experts, which enhances model expressiveness and task-specific performance with only a marginal increase in computational cost.

While there is a slight increase in communication overhead due to the need for transmitting more information between agents, the benefits in terms of motion prediction accuracy and planning safety more than justify the additional cost. In particular, the improved cooperation between agents leads to more robust predictions and safer planning in complex driving environments. Therefore, UniMM-V2X achieves a favorable trade-off, achieving an effective balance between performance, efficiency, and scalability for real-time multi-agent autonomous driving systems.

Qualitative Visualization

In this section, we present qualitative results to illustrate the effectiveness of our MoE-enhanced multi-level end-to-end cooperative autonomous driving framework across various driving tasks. As shown in Figure 1, the visualizations cover a range of scenarios, including left turns, straight driving, and right turns, highlighting the excellent performance of UniMM-V2X in enhancing perception, prediction, and final planning.



(a) Turn left.

(b) Keep forward.

(c) Turn right.

Figure 1: UniMM-V2X's planning performance on the DAIR-V2X dataset (Yu et al. 2025), including left turn, going straight, and right turn.

References

Caesar, H.; Bankiti, V.; Lang, A. H.; Vora, S.; Liong, V. E.; Xu, Q.; Krishnan, A.; Pan, Y.; Baldan, G.; and Beijbom, O. 2020. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11621–11631.

Cui, J.; Qiu, H.; Chen, D.; Stone, P.; and Zhu, Y. 2022. Coopernaut: End-to-end driving with cooperative perception for networked vehicles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 17252–17262.

Hu, S.; Chen, L.; Wu, P.; Li, H.; Yan, J.; and Tao, D. 2022a. St-p3: End-to-end vision-based autonomous driving via spatial-temporal feature learning. In *European Conference on Computer Vision*, 533–549. Springer.

Hu, Y.; Fang, S.; Lei, Z.; Zhong, Y.; and Chen, S. 2022b. Where2comm: Communication-efficient collaborative perception via spatial confidence maps. *Advances in neural information processing systems*, 35: 4874–4886.

Hu, Y.; Yang, J.; Chen, L.; Li, K.; Sima, C.; Zhu, X.; Chai, S.; Du, S.; Lin, T.; Wang, W.; et al. 2023. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 17853–17862.

Jiang, B.; Chen, S.; Xu, Q.; Liao, B.; Chen, J.; Zhou, H.; Zhang, Q.; Liu, W.; Huang, C.; and Wang, X. 2023. Vad: Vectorized scene representation for efficient autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 8340–8350.

Kuhn, H. W. 1955. The Hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2): 83–97.

Li, Y.; Ma, D.; An, Z.; Wang, Z.; Zhong, Y.; Chen, S.; and Feng, C. 2022. V2X-Sim: Multi-agent collaborative perception dataset and benchmark for autonomous driving. *IEEE Robotics and Automation Letters*, 7(4): 10914–10921.

Lu, Y.; Li, Q.; Liu, B.; Dianati, M.; Feng, C.; Chen, S.; and Wang, Y. 2023. Robust collaborative 3d object detection in presence of pose errors. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 4812–4818. IEEE.

Sun, W.; Lin, X.; Shi, Y.; Zhang, C.; Wu, H.; and Zheng, S. 2024. Sparsedrive: End-to-end autonomous driving via sparse scene representation. *arXiv preprint arXiv:2405.19620*.

Wang, T.-H.; Manivasagam, S.; Liang, M.; Yang, B.; Zeng, W.; and Urtasun, R. 2020. V2vnet: Vehicle-to-vehicle communication for joint perception and prediction. In *Computer vision–ECCV 2020: 16th European conference, Glasgow, UK, August 23–28, 2020, proceedings, part II 16*, 605–621. Springer.

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

Xu, R.; Xiang, H.; Tu, Z.; Xia, X.; Yang, M.-H.; and Ma, J. 2022b. V2x-vit: Vehicle-to-everything cooperative perception with vision transformer. In *European conference on computer vision*, 107–124. Springer.

Yu, H.; Luo, Y.; Shu, M.; Huo, Y.; Yang, Z.; Shi, Y.; Guo, Z.; Li, H.; Hu, X.; Yuan, J.; et al. 2022. Dair-v2x: A large-scale dataset for vehicle-infrastructure cooperative 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 21361–21370.

Yu, H.; Yang, W.; Zhong, J.; Yang, Z.; Fan, S.; Luo, P.; and Nie, Z. 2025. End-to-end autonomous driving through v2x cooperation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 9598–9606.