WM-V2X: COOPERATIVE WORLD MODELS VIA V2X COMMUNICATION FOR CONSISTENT OCCUPANCY PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

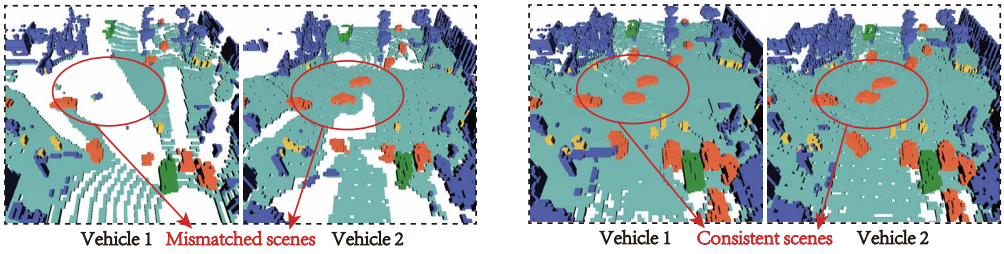
World models have demonstrated promising capability for 4D occupancy prediction and motion planning in autonomous driving. However, existing methods are designed for individual vehicles and are inherently limited by occlusions and restricted sensing ranges, which leads to inconsistent scene predictions across agents. In this paper, we propose WM-V2X, a cooperative world model framework that addresses this challenge by leveraging Vehicle-to-Everything (V2X) communication to share sensing information across connected agents for 4D forecasting on real multi-agent data. First, we design a cooperative scene tokenizer that fuses multi-agent sensory data into compact latent representations via early or intermediate fusion pipelines to facilitate 4D occupancy prediction. To further improve prediction performance, we introduce satellite imagery as an auxiliary conditioning signal that provides static scene context and visual trajectory information to a spatio-temporal transformer, which auto-regressively predicts future occupancy scenes and ego-trajectories. To evaluate cooperative world models, we reprocess the V2X-Real dataset to generate 3D voxel occupancy ground truth and introduce new cross-agent consistency metrics based on connected component analysis. Extensive experiments demonstrate that WM-V2X achieves the best 4D occupancy prediction with average mIoU of 46.04% and 39.53% under early and intermediate cooperation, and the lowest position error of 0.304 m with a 0.174% collision rate in motion planning. Cooperation further enhances cross-agent prediction consistency, raising the mean detection rate for dynamic objects by up to 41.56% over the single-agent baseline.

1 INTRODUCTION

World models have emerged as a promising paradigm that enables autonomous systems to understand the physical world by simulating its inherent dynamics, laying a foundation for the next revolution in embodied intelligence (Feng et al., 2025; Li et al., 2025; Agarwal et al., 2025). With advances in generative models such as VAEs, GANs, diffusion models, and Vision Transformers, world models have demonstrated remarkable performance on 2D-based autonomous driving tasks such as video generation and motion planning (Hu et al., 2023a; Gao et al., 2024; Wang et al., 2024b;a). Despite their success, these 2D video-based approaches lack explicit 3D geometric reasoning, making it difficult to directly support perception tasks such as 3D object detection and motion planning (Shang et al., 2024; Zuo et al., 2025). To better align world models with the physical world, recent works have explored 3D representations, such as point clouds and neural radiance fields for LiDAR generation and scene reconstruction (Yang et al., 2024; Zyrianov et al., 2025; Nunes et al., 2024). Among these 3D representations, 3D occupancy provides a unified and regular grid structure that discretizes the surrounding environment into semantic voxels, which is well-suited for downstream planning and navigation tasks (Huang et al., 2023b; Wei et al., 2023; Tian et al., 2023).

In parallel, several studies have formulated sequential 3D occupancy prediction (also known as 4D occupancy prediction) and motion planning via world modeling in an auto-regressive manner, achieving competitive results with high segmentation accuracy and low position errors (Zheng et al., 2024; Gu et al., 2024; Liao et al., 2025; Shi et al., 2025). However, these methods are designed for individual vehicle perception and planning, therefore inherently limited by occlusions, which can

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.



(a) Individual-agent-based world models.

(b) Cooperative world models.

Figure 1: The limitation of individual-agent-based world models for 3D occupancy prediction. The discrepancy between agents is significantly reduced by cooperation.

lead to scene discrepancies across agents. As illustrated in Fig. 1a, the 3D occupancy predictions from two individual-agent world models are mismatched at a future frame: vehicle 1's occupancy area is almost blank due to occlusion, while vehicle 2 predicts irregular vehicle shapes.

To bridge this gap, we develop cooperative world models based on the principle of cooperative perception, where the ego agent fuses sensing information from other connected agents via Vehicle-to-Everything (V2X) communication (Xu et al., 2022a; 2023a; Cui et al., 2022). As illustrated in Fig. 1b, compared to individual world models, the occupancy areas predicted by the two vehicles in cooperative world models are better aligned and exhibit more precise geometric shapes. The proposed framework comprises a cooperative tokenizer and a world model for V2X (WM-V2X), where the tokenizer processes sensing data into latent features for accelerating the subsequent training of WM-V2X through both early data fusion and intermediate feature fusion paradigms. Meanwhile, satellite imagery is incorporated to provide global environmental context and visual trajectory information for 4D occupancy prediction. Since existing occupancy datasets such as Occ3D (Tian et al., 2023) and OpenScene (Peng et al., 2023) are built on single-agent settings, we reprocess the real-world cooperative perception dataset V2X-Real (Xiang et al., 2024) to generate 3D voxel ground truth by segmenting raw LiDAR point clouds into 5 categories and then fusing and voxelizing across consecutive frames. To measure cross-agent scene consistency, we design detection rate and bounding box size error metrics based on connected component analysis.

Our contributions are threefold: **(1)** We formulate cooperative auto-regressive 4D occupancy and trajectory forecasting on real multi-agent data and propose WM-V2X, which mitigates cross-agent inconsistency by fusing V2X-shared sensing data with established early or intermediate fusion operators and conditions a spatio-temporal transformer on satellite imagery as a persistent static prior. **(2)** We build, to our knowledge, the first benchmark for this setting by enriching V2X-Real with 3D occupancy labels and aligned satellite images, together with cross-agent consistency metrics based on connected component analysis. **(3)** Through a controlled shared-tokenizer comparison and stress tests on imperfect V2X communication, we find that cooperation raises mIoU by 11.44 points over ego-only prediction under early fusion with a gain that decays with collaborator distance, that cooperation remains beneficial under substantial delay and packet loss while early fusion is sensitive to pose errors, and that the satellite benefit grows with the prediction horizon, while WM-V2X achieves the best mIoU in 4D occupancy prediction and the lowest position error in motion planning.

2 Related Work

2.1 World Models for 4D Occupancy Prediction

World models aim to enable autonomous systems to understand dynamic environments through inherent simulations (Peper et al., 2025; Ding et al., 2025), demonstrating competitive performance on video generation and understanding for autonomous driving (Hu et al., 2023a; Gao et al., 2024; Wang et al., 2024b). Different from 2D-based representations, 3D occupancy networks provide more precise and comprehensive scene representations applicable to downstream tasks (Wei et al., 2023; Tian et al., 2023). Building on this, several works apply the world model paradigm to 4D occupancy prediction in an auto-regressive manner, forecasting future scenes and ego trajectories simultaneously (Zheng et al., 2024; Liao et al., 2025; Yang et al., 2025; Zhang et al., 2024). To

further improve prediction quality, diffusion transformers have been explored for controllable and diverse scene generation (Gu et al., 2024; Shi et al., 2025), while flow-based methods explicitly model dynamic object motion by incorporating an auxiliary flow forecasting module (Xu et al., 2025; Chen et al., 2025b). However, these methods are all designed for individual vehicles and are inherently limited by occlusions and restricted sensing ranges, leading to inconsistent scene predictions across agents that degrade downstream planning reliability. To this end, we propose cooperative world models that leverage multi-agent sensing data via V2X communication to produce high-quality and consistent 4D occupancy predictions.

2.2 Cooperative Perception in Autonomous Driving

Cooperative perception was proposed to overcome the limitation of single-agent line-of-sight sensing in autonomous driving, enabling the ego vehicle to broaden its perceptual areas by fusing sensing information from other agents via Vehicle-to-Everything (V2X) communication (Xu et al., 2022a; 2023a), demonstrating competitive results on object detection and map segmentation tasks (Chen et al., 2019b; Arnold et al., 2022; Chen et al., 2019a; Huang et al., 2023a). However, these 2D-based representation methods are limited in their applicability to the 3D real world. To bridge this gap, recent works implement cooperative perception in 3D occupancy prediction to improve the 3D reconstruction quality by incorporating perceptual information from other traffic participants (Song et al., 2024; Chen et al., 2026; Wu et al., 2025; Lyu et al., 2024), e.g., CoHFF fuses hybrid features to estimate the current semantic occupancy on simulated data (Song et al., 2024; Xu et al., 2022b). Meanwhile, some works focus on cooperation via satellites and unmanned aerial vehicles, which provide wide-range and high-precision 3D representations from a bird's eye view (BEV) (Lin et al., 2025; Guo et al., 2025), and satellite imagery has been shown to assist current-scene occupancy prediction and semantic scene completion (Chen et al., 2025a; Guo et al., 2025). However, these methods estimate the current scene, and extending cooperative 3D occupancy prediction to sequential data for long-term reasoning remains largely unexplored. In this paper, we leverage the world model training paradigm for cooperative 4D occupancy to improve long-term prediction on real multi-agent data, complemented by satellite visual auxiliary information that provides global environmental context as a persistent prior across future rollouts.

3 Methodology

3.1 Overall Framework

In cooperative perception, the ego agent $n^* \in \mathcal{N}$ receives perception scenes $\{O^{(n)}\}_{n\in\mathcal{N}}$ from a connected agent set $\mathcal{N}$ (including itself) and fuses them into its own scene $O^{(n^*)} = \Phi(\{O^{(n)}\}_{n\in\mathcal{N}})$ (Xu et al., 2023b). Based on this, we devise a cooperative world model framework for 4D occupancy prediction and motion planning, where the ego agent predicts the next-frame scene $O_{t+1}^{(n^*)} \in \mathbb{R}^{H\times W\times Z}$ and trajectory $j_{t+1}^{(n^*)} \in \mathbb{R}^D$ based on previous T frames of scenes $\{O_{t-T+1:t}^{(n)}\}_{n\in\mathcal{N}}$, ego trajectories $j_{t-T+1:t}^{(n^*)}$, and satellite images $S_{t-T+1:t}^{(n^*)}$. The predicted outputs are appended to the input sequence to forecast long-term future scenes and trajectories in an auto-regressive manner. To reduce computational complexity, we decompose the framework into two components: the cooperative tokenizer and WM-V2X. The tokenizer fuses and compresses all scenes $\{O_{t-T+1:t}^{(n)}\}_{n\in\mathcal{N}}$ into latent tokens $z_{t-T+1:t}^{(n^*)} \in \mathbb{R}^{\frac{H}{d}\times\frac{W}{d}\times C}$ in the ego coordinate with down-sampling factor d. The WM-V2X model then predicts $z_{t+1}^{(n^*)}$ and $j_{t+1}^{(n^*)}$ based on $z_{t-T+1:t}^{(n^*)}$, $j_{t-T+1:t}^{(n^*)}$, and $S_{t-T+1:t}^{(n^*)}$, followed by decoding $O_{t+1}^{(n^*)}$ from $z_{t+1}^{(n^*)}$. The training involves two stages: we first train the cooperative tokenizer with early and intermediate fusion via a VAE pipeline, then freeze it for encoding and decoding scenes during WM-V2X training. Note that cross-agent interaction is completed entirely within the tokenizer: WM-V2X contains no cross-agent module and models the spatio-temporal evolution of the fused ego-centric latent sequence, which also allows any single-agent forecasting model to consume the same latent interface.

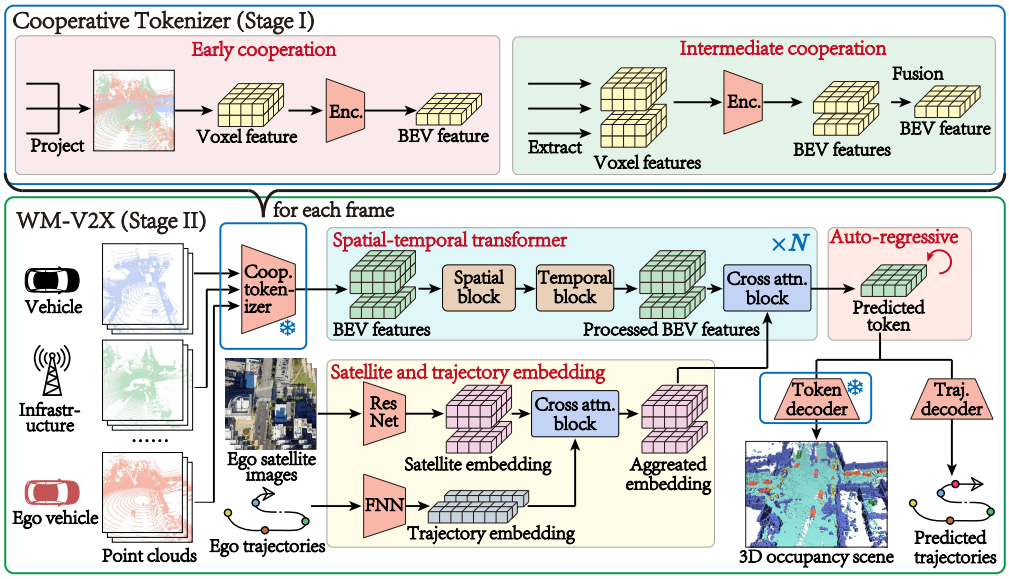


Figure 2: Overview of WM-V2X. **Top**: The cooperative scene tokenizer encodes multi-agent LiDAR point clouds into BEV latents via early fusion (merged point cloud) or intermediate fusion (per-agent encoding followed by attention-based fusion). **Bottom**: WM-V2X predicts future scenes and trajectories auto-regressively from historical latents using a spatio-temporal transformer with attention-based context conditioning from ego trajectories and satellite images. The tokenizer encoder and decoder are frozen during WM-V2X training.

3.2 Cooperative Scene Tokenizer

To improve the computational efficiency of cooperative world models, we utilize a cooperative scene tokenizer to process scenes across multiple frames from all agents in advance. As illustrated in Fig. 2, we consider early and intermediate fusion pipelines to support various cooperation paradigms. Since 3D occupancy is annotation data that cannot be directly obtained during real-world deployment, we utilize raw LiDAR as the input of the tokenizer for reconstructing 3D occupancy scenes.

Early fusion pipeline. In early fusion, we first project all point clouds $\{P^{(n)}\}_{n\in\mathcal{N}}$ from the agent set $\mathcal{N}$ into the ego coordinate system and merge them into a unified point cloud P^{fused} (Chen et al., 2019b). The fused point cloud is then voxelized into voxel features $V^{\text{fused}} \in \mathbb{R}^{H\times W\times Z\times C^{\text{v}}}$ within a predefined spatial range, where C^{v} indicates the number of voxel feature channels. Residual blocks are employed to encode the voxel features into BEV representations $F^{\text{BEV}} \in \mathbb{R}^{\frac{H}{d}\times\frac{W}{d}\times C}$ by collapsing the height dimension into the channel dimension. Subsequently, the BEV features are mapped to a continuous latent variable $z \sim \mathcal{N}(\mu, \sigma^2)$, where $\mu, \sigma \in \mathbb{R}^{\frac{H}{d}\times\frac{W}{d}\times C}$ are obtained from F^{BEV} through separate convolutional layers. Finally, the decoder reconstructs the 3D occupancy scene $\hat{O}^{(n^*)}$ of the ego vehicle from the latent variable z in subsequent stages.

Intermediate fusion pipeline. Compared to early fusion, the intermediate fusion strategy allows each agent $n \in \mathcal{N}$ to independently process its own point cloud $P^{(n)}$ to generate voxel features $V^{(n)} \in \mathbb{R}^{H\times W\times Z\times C^{\text{v}}}$, followed by encoding these voxel features into BEV representations $F^{(n)} \in \mathbb{R}^{\frac{H}{d}\times\frac{W}{d}\times C}$ through a shared encoder equipped with residual blocks. Subsequently, we project all BEV features $\{F^{(n)}\}_{n\in\mathcal{N}}$ into the ego coordinate system and aggregate them into a unified BEV feature $F^{\text{BEV}} = \Phi(\{F^{(n)}\}_{n\in\mathcal{N}})$ via an attention-based fusion module $\Phi(\cdot)$ (Xu et al., 2023a; 2022b; Hu et al., 2022), for which we adopt the established local–global (FAX) attention design of CoBEVT (Xu et al., 2023a). Similar to early fusion, the fused feature is then reparameterized to obtain the latent variable $z \sim \mathcal{N}(\mu, \sigma^2)$ for the WM-V2X training.

Loss function. To optimize the parameters of the cooperative scene tokenizer using a VAE objective, a weighted combination of CE loss, Dice loss, and KL divergence regularization is applied as the

total loss for tokenizer training, formulated as,

$$\mathcal{L}_{\text{token}} = \lambda_{\text{CE}}\mathcal{L}_{\text{CE}}(O^{(n^*)}, \hat{O}^{(n^*)}) + \lambda_{\text{Dice}}\mathcal{L}_{\text{Dice}}(O^{(n^*)}, \hat{O}^{(n^*)}) + \beta\mathcal{L}_{\text{KL}}, \tag{1}$$

where λ_{CE} and λ_{Dice} are the weights for cross-entropy and Dice loss, respectively, and β is the weight for the KL divergence term.

3.3 WM-V2X

The WM-V2X module is the core predictive engine of our framework. Given historical fused scene latents $z^{(n^*)}_{t-T+1:t}$ and ego-trajectories $j^{(n^*)}_{t-T+1:t}$, it auto-regressively predicts the next scene latent $\hat{z}^{(n^*)}_{t+1}$ and trajectory $\hat{j}^{(n^*)}_{t+1}$. As illustrated in Fig. 2, the backbone is a spatio-temporal transformer conditioned on satellite imagery. Since LiDAR observations are inherently sparse and range-limited, static scene elements such as road layouts and building boundaries are difficult to maintain consistently over long-horizon rollouts. Satellite images serve as a persistent, dense BEV prior that anchors static predictions and mitigates error accumulation across auto-regressive steps.

Spatio-temporal transformer. Each historical latent $z^{(n^*)}_t \in \mathbb{R}^{\frac{H}{d}\times\frac{W}{d}\times C}$ is first enriched with a learnable spatial positional embedding E^{pos} to encode token locations within the BEV grid, and a temporal embedding E^{temp}_t to encode frame order, i.e., $x^{\text{latent}}_{t,0} = z^{(n^*)}_t + E^{\text{pos}} + E^{\text{temp}}_t$. The resulting sequence is then processed by K interleaved spatial, temporal, and context conditioning blocks. For the spatial block, a lightweight residual convolution block $\text{Conv}(\cdot)$ is employed rather than a more expensive spatial self-attention mechanism to efficiently capture local spatial patterns within each BEV frame. Subsequently, multi-head self-attention $\text{MHA}(\cdot)$ (Vaswani et al., 2017) is applied in the temporal block to model long-range dependencies across frames, followed by the context conditioning block that attends scene features to the fused satellite-trajectory embedding F^{fused}_t via cross-attention $\text{CA}(\cdot)$. These steps are formulated as,

$$x^{\text{spatial}}_{t,k-1} = x^{\text{latent}}_{t,k-1} + \text{Conv}(\text{SiLU}(\text{Conv}(\text{GN}(x^{\text{latent}}_{t,k-1})))), \tag{2}$$

$$x^{\text{temp}}_{t,k-1} = x^{\text{spatial}}_{t,k-1} + \text{MHA}(\text{LN}(x^{\text{spatial}}_{t,k-1})), \tag{3}$$

$$x^{\text{latent}}_{t,k} = x^{\text{temp}}_{t,k-1} + \text{CA}(x^{\text{temp}}_{t,k-1}, F^{\text{fused}}_t, F^{\text{fused}}_t), \tag{4}$$

where $\text{GN}(\cdot)$, $\text{SiLU}(\cdot)$, $\text{Conv}(\cdot)$, and $\text{LN}(\cdot)$ denote group normalization, SiLU activation, convolutional layers, and layer normalization, respectively. By interleaving these three block types for K iterations, the model progressively builds a rich spatio-temporal representation that is both dynamically aware and spatially grounded.

Multi-modal context embedding and fusion. The ego satellite image $S^{(n^*)}_t \in \mathbb{R}^{H^{\text{s}}\times W^{\text{s}}\times 3}$ is passed through a ResNet (He et al., 2016) encoder to produce low-resolution BEV satellite features $F^{\text{sat}}_t \in \mathbb{R}^{\frac{H^{\text{s}}}{d^{\text{s}}}\times\frac{W^{\text{s}}}{d^{\text{s}}}\times C^{\text{sat}}}$, capturing the global static layout of the scene. In parallel, the historical ego-trajectory $j^{(n^*)}_t \in \mathbb{R}^D$ is projected into a compact latent vector $F^{\text{traj}}_t = \text{MLP}(j^{(n^*)}_t)$ for encoding the vehicle's motion context. Then these two embeddings are fused via cross-attention, where the satellite BEV features attend to the trajectory embedding to highlight map regions most relevant to the vehicle's planned motion:

$$F^{\text{fused}}_t = F^{\text{sat}}_t + \text{CA}(F^{\text{sat}}_t, F^{\text{traj}}_t, F^{\text{traj}}_t). \tag{5}$$

The resulting F^{fused}_t integrates high-level map semantics with motion-specific information, and is injected into each context conditioning block of the spatio-temporal transformer. Satellite tiles are collected once and cached offline, so the ego pose only indexes a cached crop at runtime, and the satellite branch can be dropped as a fallback (Sec. 4.3).

Prediction heads. The output $x^{\text{latent}}_{t,K}$ from the final block is decoded by two parallel heads: a lightweight 1×1 convolution with GN and activation produces the predicted scene latent $\hat{z}^{(n^*)}_{t+1} \in \mathbb{R}^{\frac{H}{d}\times\frac{W}{d}\times C}$, while an AvgPool-MLP head yields the predicted trajectory $\hat{j}^{(n^*)}_{t+1} \in \mathbb{R}^D$. The predicted outputs replace the oldest frame in the input window and are fed back for the next auto-regressive step, enabling long-horizon forecasting.

Table 1: Performance comparison of 4D occupancy prediction. The mIoU and fwIoU for VAE **reconstruction** of occupancy in Early fusion are **52.46** and **64.18**, respectively. For intermediate fusion, these two values are **44.99** and **55.71**. Avg. denotes the average value over 4 frames. The **best** results are highlighted in orange. All methods share the same frozen tokenizer.

Coop.	Method	mIoU (%) ↑					fwIoU (%) ↑				
		0.5s	1s	1.5s	2s	Avg.	0.5s	1s	1.5s	2s	Avg.
Early	OccWorld	47.00	41.28	36.49	32.57	39.34	58.64	53.18	48.55	44.74	51.28
	DOME	45.10	40.17	36.95	34.76	39.25	57.56	53.36	50.72	48.99	52.66
	OccProphet	50.15	47.03	43.78	41.19	45.54	**61.74**	**58.90**	56.05	54.05	57.69
	EfficientOCF	46.00	34.62	16.77	9.45	26.71	57.35	44.62	25.53	18.46	36.49
	I2-World	46.52	40.25	35.20	31.45	38.36	58.21	52.39	47.78	44.43	50.70
	COME	45.61	40.59	37.22	34.91	39.58	58.13	53.92	51.19	49.38	53.16
	WM-V2X	**50.18**	**47.19**	**44.50**	**42.27**	**46.04**	61.61	58.80	**56.46**	**54.70**	**57.89**
Intermediate	OccWorld	40.68	37.03	34.04	31.71	35.87	52.75	49.82	47.45	45.62	48.91
	DOME	38.75	34.90	32.25	30.28	34.05	51.25	48.04	45.86	44.30	47.36
	OccProphet	42.27	39.73	37.40	34.70	38.52	54.21	51.94	49.99	47.58	50.93
	EfficientOCF	38.34	25.40	8.64	5.06	19.36	49.44	36.51	18.09	11.82	28.96
	I2-World	40.46	36.41	33.26	30.83	35.24	52.56	49.17	46.55	44.53	48.20
	COME	38.66	34.72	32.18	30.36	33.98	51.21	47.94	45.83	44.36	47.34
	WM-V2X	**42.85**	**40.65**	**38.38**	**36.25**	**39.53**	**54.70**	**52.63**	**50.63**	**48.88**	**51.71**

Loss function. The WM-V2X model is trained end-to-end by minimizing a combined loss function that supervises both future scene and trajectory predictions. The loss is a weighted sum of a latent prediction loss and a trajectory prediction loss, where both losses use the mean squared error (MSE):

$$\mathcal{L}_{\text{pred}} = \lambda_{\text{latent}}\mathcal{L}_{\text{mse}}(z_{t+1}^{(n^*)}, \hat{z}_{t+1}^{(n^*)}) + \lambda_{\text{traj}}\mathcal{L}_{\text{mse}}(j_{t+1}^{(n^*)}, \hat{j}_{t+1}^{(n^*)}), \tag{6}$$

where λ_{latent} and λ_{traj} are scalar weights to balance the two tasks.

4 Experiments

4.1 Dataset and Experimental Setup

Dataset and metrics. The V2X-Real dataset is adopted for validating the performance of the proposed WM-V2X, which comprises 5772 training frames and 1253 test frames collected from 63 diverse scenarios involving a total of two infrastructures and two vehicles (Xiang et al., 2024). Since the dataset does not provide ground truth for occupancy prediction, we utilize the Patchwork++ toolkit (Ren et al., 2023) and bounding boxes to segment the raw LiDAR point clouds. We then split and combine static scenes and dynamic objects across multiple consecutive frames, respectively, to construct the final occupancy ground truth. Details are described in Appendix B.1. In terms of satellite images, we use the Google Earth API to collect satellite images for each frame based on the relative GPS positions of vehicles. Details are provided in Appendix B.2. The mean Intersection over Union (mIoU) and frequency-weighted IoU (fwIoU) are used to evaluate the model performance on 3D occupancy prediction (Caesar et al., 2020), and the position error and collision rate are applied for validating motion planning (Hu et al., 2023b). For evaluating scene consistency across vehicles, we devise new detection rate and bounding box size error metrics by extracting dynamic objects from predicted scene via connected component analysis. These metrics are reported per future timestep and averaged over all timesteps.

Implementation details. We use 8 historical frames spanning 4 s (2 Hz) to forecast up to 10 future frames spanning 5 s (2 Hz). The communication range is 50 m, and the LiDAR range covers $[-64, 64]$ m in x and y, and $[-4, 4]$ m in z. Point clouds are voxelized into $256 \times 256 \times 16$ grids at 0.5 m resolution and encoded into 512-channel BEV latents at 64×64 spatial resolution via a ResNet-18 backbone with downsampling factor d=4. The WM-V2X transformer uses K=6 blocks with 8 attention heads. Both the cooperative tokenizer and WM-V2X are trained with AdamW and a cosine annealing scheduler with 10 warmup epochs. The latent and trajectory loss weights are λ_{latent}=0.7 and λ_{traj}=0.3. All experiments are conducted on an RTX 5090 GPU. Full hyperparameter settings are provided in Appendix C.1.

Table 2: Performance comparison of motion planning, where Coop. represents cooperation paradigms. The position error is the L2 distance between the predicted trajectory and the ground truth in x and y coordinates. The collision rate is measured as the ratio of overlap cases between the ego bounding box and other object bounding boxes over all test samples.

Coop.	Method	Position error (m) ↓					Collision rate (%) ↓				
		0.5s	1s	1.5s	2s	Avg.	0.5s	1s	1.5s	2s	Avg.
Early	OccWorld	0.054	0.065	0.076	0.085	0.070	0.047	0.047	0.047	0.094	0.059
	DOME	0.034	0.036	0.039	0.043	0.038	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
	OccProphet	0.059	0.083	0.107	0.120	0.092	**0.000**	**0.000**	0.047	0.094	0.035
	EfficientOCF	0.084	0.108	0.176	0.259	0.157	0.187	0.468	0.795	1.165	0.654
	I2-World	0.057	0.060	0.069	0.078	0.066	0.047	0.047	0.047	0.047	0.047
	COME	0.041	0.041	0.043	0.046	0.043	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
	WM-V2X	**0.031**	**0.031**	**0.035**	**0.039**	**0.034**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
Intermediate	OccWorld	0.523	0.661	0.963	1.265	0.853	1.123	1.543	2.713	3.181	2.140
	DOME	0.334	0.387	0.515	0.673	0.477	0.094	0.374	0.421	0.514	0.351
	OccProphet	0.573	1.158	1.817	2.425	1.493	0.047	0.655	2.479	4.069	1.813
	EfficientOCF	0.994	1.631	2.182	2.208	1.754	1.684	2.151	2.760	3.976	2.643
	I2-World	0.228	0.367	0.522	0.694	0.453	**0.000**	0.187	0.420	0.468	0.269
	COME	0.338	0.412	0.532	0.661	0.486	**0.000**	0.190	0.287	0.427	0.226
	WM-V2X	**0.187**	**0.258**	**0.339**	**0.431**	**0.304**	**0.000**	**0.094**	**0.230**	**0.370**	**0.174**

4.2 Quantitative Evaluation

4D occupancy prediction. We evaluate the 4D occupancy prediction performance of WM-V2X under both early and intermediate cooperation paradigms. Different from previous works that utilize 3D occupancy as input, we use point cloud data captured from LiDAR sensors since vehicles cannot obtain 3D occupancy directly from onboard sensors. Since cross-agent fusion is confined to the tokenizer, all baselines, i.e., OccWorld (Zheng et al., 2024), DOME (Gu et al., 2024), OccProphet (Chen et al., 2025b), EfficientOCF (Xu et al., 2025), I2-World (Liao et al., 2025), and COME (Shi et al., 2025), are retrained on the same frozen tokenizer and ego trajectory while retaining their native conditions, i.e., a controlled comparison of forecasting models rather than native end-to-end reproductions (Appendix D). As demonstrated in Table 1, WM-V2X achieves competitive prediction performance with the best average mIoU of 46.04% and 39.53% and average fwIoU of 57.89% and 51.71% over 4 frames for early and intermediate cooperation, respectively. WM-V2X also achieves the best results across most individual frames. Although OccProphet achieves marginally higher fwIoU at the 0.5 and 1 s horizons under early cooperation, WM-V2X surpasses it at the 1.5 and 2 s horizons and on all intermediate cooperation metrics, demonstrating more consistent long-horizon accuracy. With native tokenizers, I2-World is unchanged (38.22%) and OccWorld rises to 43.39% under early cooperation, still below WM-V2X. The multi-view image-based results are summarized in Appendix C.2, where the input sensor data and encoder for cooperative tokenizer were replaced by multi-view images and corresponding backbones.

Motion planning. We also evaluate the motion planning performance of WM-V2X under both cooperation paradigms. Since some methods do not consider trajectory prediction, we apply a unified trajectory decoder to enable a controlled comparison. It is clear from Table 2 that WM-V2X achieves the lowest position error and collision rate across all future frames and cooperation paradigms. Notably, the position errors are lower compared to those on other datasets such as KITTI (Geiger et al., 2013) and nuScenes (Caesar et al., 2020), which can be attributed to the lower velocity of agents in intersection scenarios. Consequently, the collision rate is reduced to 0 when the position error is sufficiently low.

Cross-agent scene consistency. Table 3 reports the consistency of dynamic object predictions across agents under each cooperation paradigm, and the formal metric definitions are included in the Appendix A.3. Without cooperation, vehicle and infrastructure agents achieve mean detection rates of only 19.93% and 7.78%, respectively, along with high bounding box size errors, confirming the inherent limitation of individual-agent-based world models. Under early cooperation, the detection rate improves by 23.50% for vehicle agents and 41.56% for infrastructure agents, while the bounding box size error decreases by 17.13% and 22.93%, respectively. Intermediate cooperation also yields substantial improvements over the no-cooperation baseline, though slightly below early cooperation.

Table 3: Cross-agent scene consistency under various cooperation paradigms. The detection rate is computed as the number of matched bounding boxes divided by the maximum bounding box count across agents, with an IoU threshold of 0.3 for matching. Scene consistency is further assessed by the mean bounding box size error across the x, y, and z dimensions of matched objects.

Ego agent	Cooperation	Mean detection rate (%) ↑				Mean bbox size error (m) ↓			
		0.5s	1s	1.5s	2s	0.5s	1s	1.5s	2s
Vehicle	✗	22.52	21.19	19.00	17.00	0.4636	0.4737	0.4799	0.4881
	Intermediate	31.01	28.80	26.33	23.82	0.4302	0.4320	0.4459	0.4582
	Early	**48.42**	**45.37**	**42.00**	**37.94**	**0.3731**	**0.3773**	**0.4033**	**0.4252**
Infrastructure	✗	8.90	7.68	7.51	7.02	0.5552	0.5905	0.6633	0.6855
	Intermediate	42.47	41.26	40.47	39.39	0.5229	0.5325	0.5344	0.5466
	Early	**50.90**	**49.95**	**48.88**	**47.64**	**0.4621**	**0.4762**	**0.4885**	**0.4958**

These results demonstrate that cooperative world models substantially improve cross-agent prediction consistency across all future frames.

4.3 Ablation and Robustness Studies

Table 4: Ablation study on components of WM-V2X. T-block, S-block, Traj., and Sat. represent temporal block, spatial block, trajectory embedding, and satellite embedding, respectively.

T-block	S-block	Traj.	Sat.	mIoU (%) ↑				Position error (m) ↓			
				0.5s	1s	1.5s	2s	0.5s	1s	1.5s	2s
✗	✗	✗	✗	46.38	41.03	37.00	33.99	0.162	0.162	0.166	0.170
✓	✗	✗	✗	49.07	45.02	41.64	39.05	0.089	0.109	0.121	0.134
✓	✓	✗	✗	49.52	45.89	42.81	40.40	0.079	0.089	0.099	0.113
✓	✓	✓	✗	49.38	45.85	42.87	40.49	**0.031**	0.039	0.039	0.041
✓	✓	✓	✓	**50.18**	**47.19**	**44.50**	**42.27**	**0.031**	**0.031**	**0.035**	**0.039**

Component analysis. The ablation study on both 4D occupancy prediction and motion planning across future 4 frames is summarized in Table 4, where ✓and ✗denote enabling and disabling the corresponding components. A component is excluded from the network if it is marked as disabled. It is clear that the mIoU increases and position error decreases by enabling temporal and spatial blocks, with improvements of 5.05% and 0.070 m, respectively. This result highlights the importance of modeling spatial and temporal information in 4D occupancy prediction. Furthermore, the position error is reduced substantially (by around 60.5%) when trajectory embedding is enabled, which highlights that trajectory information plays a key role in motion planning. Finally, by enabling satellite embedding, the performance further improves, reaching an average mIoU of 46.04% and an average position error of 0.034 m across 4 frames. The satellite gain of $+1.39$ points grows from $+0.80$ at 0.5 s to $+1.78$ at 2 s. Without it, WM-V2X (44.65%), i.e., the input-matched control for Table 1, exceeds all baselines except OccProphet (45.54%).

Robustness to imperfect V2X. Table 5 perturbs only collaborator messages, and the detailed evaluation protocol is presented in Appendix E. Early fusion yields the larger gain of $+11.44$ mIoU points over ego-only prediction and stays above it under 4 s delay and 75% packet loss, but falls below it at 1.0 m translation or 5° yaw noise, since misaligned raw points are merged directly into the 0.5 m voxel grid. Intermediate fusion gains only $+4.53$ points but is far less sensitive: its mIoU changes by at most 0.14 points up to 1.0 m or 5° noise and by 0.25/0.43 points at 5 m/15° (see Appendix E.3 for a discussion of this insensitivity). Its insensitivity to delay is consistent with the current-pose alignment of the delayed features, the stationary infrastructure in V2X-Real, and the 58% of vehicle frames that are nearly static. Packet loss degrades both paradigms monotonically.

Communication cost. Lossless bit packing reduces the early-fusion payload to 128 KiB per collaborator and frame, and INT4 quantization compresses intermediate BEV features by $8\times$ to 1 MiB with a 0.10-point mIoU drop. On an ideal shared 100 Mbps link, the complete per-frame path from collaborator preprocessing to four-step rollout and decoding takes 406.47/438.41 ms for early/intermediate fusion, within the 500 ms sampling interval (Appendix F). Long-term rollouts and inference efficiency are analyzed in Appendices C.3 and F.1.

Table 5: Robustness to imperfect collaborator messages (mIoU in %, averaged over 0.5–2 s). Ego-only corresponds to 100% packet loss, underlined values fall below it, and "–" marks settings not evaluated.

Fusion	Clean	Translation σ_t (m)			Yaw σ_θ			Delay (s)		Packet loss		Ego-only
		0.5	1.0	5.0	2°	5°	15°	2.5	4.0	50%	75%	
Early	46.04	39.34	32.11	–	37.14	29.95	–	40.29	38.89	40.98	38.10	34.60
Intermediate	39.53	39.53	39.53	39.28	39.50	39.39	39.10	39.53	39.52	37.63	36.34	35.00

4.4 Qualitative Results

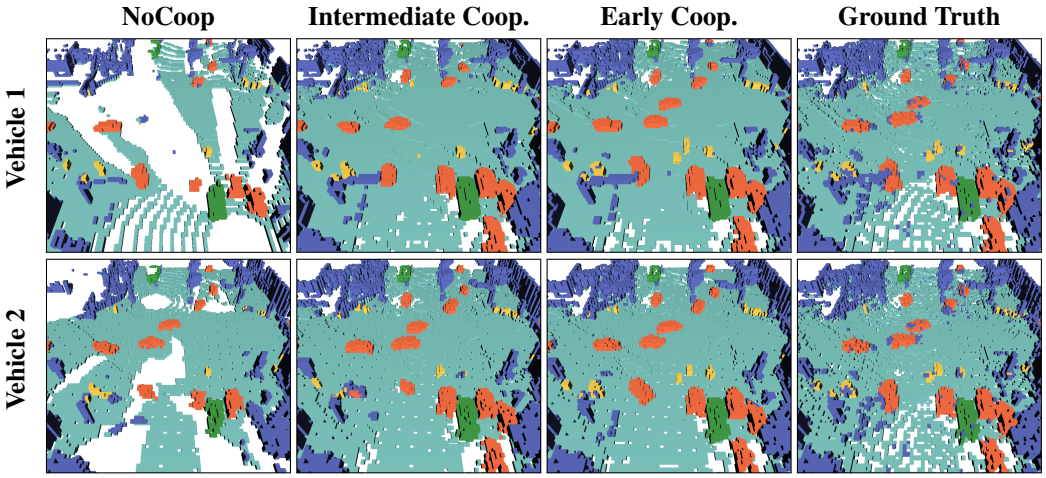


Figure 3: Visual comparison of occupancy predictions from two vehicle agents under different cooperation paradigms at t=2 s. Ground, building, vehicle, pedestrian, and truck voxels are colored by teal, indigo, orange, yellow, and green, respectively. Cooperation significantly improves the consistency of dynamic object predictions across agents.

Fig. 3 shows occupancy predictions from two vehicle agents at future $t = 2$s under no, intermediate, and early cooperation paradigms alongside ground truth. Without cooperation, almost all grids in Vehicle 1's scene are unlabeled due to occlusion by trucks, and Vehicle 2's prediction is fragmented with scattered and incomplete dynamic objects. Intermediate cooperation aligns both predictions with structured vehicle clusters but still misses several objects for Vehicle 1, while early cooperation yields the closest match to the ground truth from both viewpoints.

5 Conclusion

We presented WM-V2X, a cooperative world model framework that leverages Vehicle-to-Everything communication to address scene prediction inconsistency across connected vehicles. The framework introduces a cooperative scene tokenizer that fuses multi-agent sensory observations into unified latent representations via early and intermediate fusion pipelines, and a spatio-temporal transformer conditioned on satellite imagery to auto-regressively predict future 3D occupancy scenes and ego-trajectories. We also extended the V2X-Real dataset with 3D occupancy labels and satellite images, and proposed new cross-agent consistency metrics based on connected component analysis. Experiments demonstrate that WM-V2X achieves the best mIoU in 4D occupancy prediction and the lowest position error in motion planning, with cooperation improving the cross-agent detection rate by up to 41.56% over single-agent baselines.

Limitations and future work. Early fusion becomes worse than ego-only prediction under large collaborator pose errors, our transmission latencies are ideal lower bounds without protocol overheads, and cached satellite imagery may be stale. In the future, we will explore pose-robust fusion and adaptive communication strategies to handle these degradations.

AI USE STATEMENT

In this work, we used generative AI tools to assist in writing and editing software code and in implementing parts of the proposed methods, which includes a task with required disclosure under the ICLR 2027 AI Policy for Authors (implementing methods). We have not used generative AI tools for the other tasks with required disclosure, such as generating data sets, formulating hypotheses, designing the research methodology or experiments, or interpreting results. Additionally, we used generative AI tools for tasks with recommended disclosure, namely editing the manuscript for grammar, phrasing, and readability, drafting revisions of some passages from author-written notes, LaTeX and reference formatting, and creating scripts that visualize experimental results. The authors reviewed, tested, and validated all AI-assisted code before it was used to produce any reported result, reviewed all AI-assisted text, formatting, and visualizations, and checked every reported number against the experiment logs. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work involves no human subjects or crowdsourcing. The V2X-Real dataset is publicly released for research use and is adopted in accordance with its license terms. Satellite images are collected from Google Earth in compliance with its Terms of Service and are provided only as low-resolution crops aligned with the dataset coordinates. Sharing perception data across agents may raise privacy concerns, and imperfect V2X communication may indirectly affect driving safety; we discuss these risks and possible mitigations in Appendix I.

REPRODUCIBILITY STATEMENT

The experimental setup is described in Sec. 4.1, and full hyperparameters are listed in Appendix C.1. Appendix A details the intermediate fusion module, the auto-regressive inference procedure, and the cross-agent consistency metrics, while Appendices B.1 and B.2 document the occupancy ground-truth generation and the satellite-image alignment. The comparison protocol, the perturbation protocols, and the latency measurement settings are given in Appendices D, E, and F, respectively, and the computing resources are reported in Appendix H. Anonymized code, preprocessing scripts, and a batch of preprocessed data are provided in the supplementary material.

REFERENCES

Niket Agarwal, Arslan Ali, Maciej Bala, Yogesh Balaji, Erik Barker, Tiffany Cai, Prithvijit Chattopadhyay, Yongxin Chen, Yin Cui, Yifan Ding, et al. Cosmos world foundation model platform for physical ai. *arXiv preprint arXiv:2501.03575*, 2025.

Eduardo Arnold, Mehrdad Dianati, Robert de Temple, and Saber Fallah. Cooperative perception for 3d object detection in driving scenarios using infrastructure sensors. *IEEE Transactions on Intelligent Transportation Systems*, 23(3):1852–1864, 2022. doi: 10.1109/TITS.2020.3028424.

Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

Chen Chen, Zhirui Wang, Taowei Sheng, Yi Jiang, Yundu Li, Peirui Cheng, Luning Zhang, Kaiqiang Chen, Yanfeng Hu, Xue Yang, and Xian Sun. SA-Occ: Satellite-assisted 3d occupancy prediction in real world. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 27021–27030, October 2025a.

Cheng Chen, Hao Huang, and Saurabh Bagchi. Vision-only gaussian splatting for collaborative semantic occupancy prediction. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40 (4):2796–2804, Mar. 2026. doi: 10.1609/aaai.v40i4.37269.

Junliang Chen, Huaiyuan Xu, Yi Wang, and Lap-Pui Chau. Occprophet: Pushing the efficiency frontier of camera-only 4d occupancy forecasting with an observer-forecaster-refiner framework. In *The Thirteenth International Conference on Learning Representations*, 2025b.

Qi Chen, Xu Ma, Sihai Tang, Jingda Guo, Qing Yang, and Song Fu. 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*, SEC '19, pp. 88–100, New York, NY, USA, 2019a. Association for Computing Machinery. ISBN 9781450367332. doi: 10.1145/3318216.3363300.

Qi Chen, Sihai Tang, Qing Yang, and Song Fu. Cooper: Cooperative perception for connected autonomous vehicles based on 3d point clouds. In *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, pp. 514–524, 2019b. doi: 10.1109/ICDCS.2019.00058.

Jiaxun Cui, Hang Qiu, Dian Chen, Peter Stone, and Yuke Zhu. Coopernaut: End-to-end driving with cooperative perception for networked vehicles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17252–17262, June 2022.

Jingtao Ding, Yunke Zhang, Yu Shang, Yuheng Zhang, Zefang Zong, Jie Feng, Yuan Yuan, Hongyuan Su, Nian Li, Nicholas Sukiennik, et al. Understanding world or predicting future? a comprehensive survey of world models. *ACM Computing Surveys*, 58(3):1–38, 2025.

Tuo Feng, Wenguan Wang, and Yi Yang. A survey of world models for autonomous driving. *arXiv preprint arXiv:2501.11260*, 2025.

Shenyuan Gao, Jiazhi Yang, Li Chen, Kashyap Chitta, Yihang Qiu, Andreas Geiger, Jun Zhang, and Hongyang Li. Vista: A generalizable driving world model with high fidelity and versatile controllability. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 91560–91596. Curran Associates, Inc., 2024. doi: 10.52202/079017-2906.

A Geiger, P Lenz, C Stiller, and R Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013. doi: 10.1177/0278364913491297.

Songen Gu, Wei Yin, Bu Jin, Xiaoyang Guo, Junming Wang, Haodong Li, Qian Zhang, and Xiaoxiao Long. Dome: Taming diffusion model into high-fidelity controllable occupancy world model. *arXiv preprint arXiv:2410.10429*, 2024.

Xiyue Guo, Jiarui Hu, Junjie Hu, Hujun Bao, and Guofeng Zhang. Sgformer: Satellite-ground fusion for 3d semantic scene completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11929–11938, June 2025.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

Anthony Hu, Lloyd Russell, Hudson Yeo, Zak Murez, George Fedoseev, Alex Kendall, Jamie Shotton, and Gianluca Corrado. Gaia-1: A generative world model for autonomous driving. *arXiv preprint arXiv:2309.17080*, 2023a.

Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, Lewei Lu, Xiaosong Jia, Qiang Liu, Jifeng Dai, Yu Qiao, and Hongyang Li. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17853–17862, June 2023b.

Yue Hu, Shaoheng Fang, Zixing Lei, Yiqi Zhong, and Siheng Chen. Where2comm: Communication-efficient collaborative perception via spatial confidence maps. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 4874–4886. Curran Associates, Inc., 2022.

Tao Huang, Jianan Liu, Xi Zhou, Dinh C Nguyen, Mostafa Rahimi Azghadi, Yuxuan Xia, Qing-Long Han, and Sumei Sun. V2x cooperative perception for autonomous driving: Recent advances and challenges. *arXiv preprint arXiv:2310.03525*, 2023a.

Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. Tri-perspective view for vision-based 3d semantic occupancy prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9223–9232, June 2023b.

Xinqing Li, Xin He, Le Zhang, Min Wu, Xiaoli Li, and Yun Liu. A comprehensive survey on world models for embodied ai. *arXiv preprint arXiv:2510.16732*, 2025.

Zhimin Liao, Ping Wei, Ruijie Zhang, Shuaijia Chen, Haoxuan Wang, and Ziyang Ren. I2-world: Intra-inter tokenization for efficient dynamic 4d scene forecasting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 25810–25819, October 2025.

Haotong Lin, Sili Chen, Jun Hao Liew, Donny Y. Chen, Zhenyu Li, Yang Zhao, Sida Peng, Hengkai Guo, Xiaowei Zhou, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views. In *The Fourteenth International Conference on Learning Representations*, 2026.

Zefu Lin, Wenbo Chen, Xiaojuan Jin, Yuran Yang, Lue Fan, Yixin Zhang, Yufeng Zhang, and Zhaoxiang Zhang. Mcop: Multi-uav collaborative occupancy prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 27242–27251, October 2025.

Chuandong Lyu, Shengbang Guo, Bin Zhou, Hailiang Xiong, and Hongchao Zhou. 3dopformer: 3d occupancy perception from multi-camera images with directional and distance enhancement. *IEEE Transactions on Intelligent Vehicles*, 9(9):5478–5489, 2024. doi: 10.1109/TIV.2023.3343749.

Lucas Nunes, Rodrigo Marcuzzi, Benedikt Mersch, Jens Behley, and Cyrill Stachniss. Scaling diffusion models to real-world 3d lidar scene completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14770–14780, June 2024.

Songyou Peng, Kyle Genova, Chiyu "Max" Jiang, Andrea Tagliasacchi, Marc Pollefeys, and Thomas Funkhouser. Openscene: 3d scene understanding with open vocabularies. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 815–824, June 2023.

Jordan Peper, Zhenjiang Mao, Yuang Geng, Siyuan Pan, and Ivan Ruchkin. Four principles for physically interpretable world models. *arXiv preprint arXiv:2503.02143*, 2025.

Hao Ren, Mingwei Wang, Wenpeng Li, Chen Liu, and Mengli Zhang. Adaptive patchwork: Real-time ground segmentation for 3d point cloud with adaptive partitioning and spatial-temporal context. *IEEE Robotics and Automation Letters*, 8(11):7162–7169, 2023.

Yu Shang, Yuming Lin, Yu Zheng, Hangyu Fan, Jingtao Ding, Jie Feng, Jiansheng Chen, Li Tian, and Yong Li. Urbanworld: An urban world model for 3d city generation. *arXiv preprint arXiv:2407.11965*, 2024.

Yining Shi, Kun Jiang, Qiang Meng, Ke Wang, Jiabao Wang, Wenchao Sun, Tuopu Wen, Mengmeng Yang, and Diange Yang. Come: Adding scene-centric forecasting control to occupancy world model. *arXiv preprint arXiv:2506.13260*, 2025.

Rui Song, Chenwei Liang, Hu Cao, Zhiran Yan, Walter Zimmer, Markus Gross, Andreas Festag, and Alois Knoll. Collaborative semantic occupancy prediction with hybrid feature fusion in connected automated vehicles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17996–18006, June 2024.

Xiaoyu Tian, Tao Jiang, Longfei Yun, Yucheng Mao, Huitong Yang, Yue Wang, Yilun Wang, and Hang Zhao. Occ3d: A large-scale 3d occupancy prediction benchmark for autonomous driving. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 64318–64330. Curran Associates, Inc., 2023.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Xiaofeng Wang, Zheng Zhu, Guan Huang, Xinze Chen, Jiagang Zhu, and Jiwen Lu. Drivedreamer: Towards real-world-drive world models for autonomous driving. In *European conference on computer vision*, pp. 55–72. Springer, 2024a.

Yuqi Wang, Jiawei He, Lue Fan, Hongxin Li, Yuntao Chen, and Zhaoxiang Zhang. Driving into the future: Multiview visual forecasting and planning with world model for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14749–14759, June 2024b.

Yi Wei, Linqing Zhao, Wenzhao Zheng, Zheng Zhu, Jie Zhou, and Jiwen Lu. Surroundocc: Multi-camera 3d occupancy prediction for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 21729–21740, October 2023.

Hanlin Wu, Pengfei Lin, Ehsan Javanmardi, Naren Bao, Bo Qian, Hao Si, and Manabu Tsukada. A synthetic benchmark for collaborative 3d semantic occupancy prediction in v2x autonomous driving. *arXiv preprint arXiv:2506.17004*, 2025.

Hao Xiang, Zhaoliang Zheng, Xin Xia, Runsheng Xu, Letian Gao, Zewei Zhou, Xu Han, Xinkai Ji, Mingxi Li, Zonglin Meng, et al. V2x-real: A largs-scale dataset for vehicle-to-everything cooperative perception. In *European Conference on Computer Vision*, pp. 455–470. Springer, 2024.

Jingyi Xu, Xieyuanli Chen, Junyi Ma, Jiawei Huang, Jintao Xu, Yue Wang, and Ling Pei. Spatiotemporal decoupling for efficient vision-based occupancy forecasting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22338–22347, June 2025.

Runsheng Xu, Hao Xiang, Zhengzhong Tu, Xin Xia, Ming-Hsuan Yang, and Jiaqi Ma. V2x-vit: Vehicle-to-everything cooperative perception with vision transformer. In *European conference on computer vision*, pp. 107–124. Springer, 2022a.

Runsheng Xu, Hao Xiang, Xin Xia, Xu Han, Jinlong Li, and Jiaqi Ma. Opv2v: An open benchmark dataset and fusion pipeline for perception with vehicle-to-vehicle communication. In *2022 International Conference on Robotics and Automation (ICRA)*, pp. 2583–2589, 2022b. doi: 10.1109/ICRA46639.2022.9812038.

Runsheng Xu, Zhengzhong Tu, Hao Xiang, Wei Shao, Bolei Zhou, and Jiaqi Ma. Cobevt: Cooperative bird's eye view semantic segmentation with sparse transformers. In *Conference on Robot Learning*, pp. 989–1000. PMLR, 2023a.

Runsheng Xu, Xin Xia, Jinlong Li, Hanzhao Li, Shuo Zhang, Zhengzhong Tu, Zonglin Meng, Hao Xiang, Xiaoyu Dong, Rui Song, Hongkai Yu, Bolei Zhou, and Jiaqi Ma. V2v4real: A real-world large-scale dataset for vehicle-to-vehicle cooperative perception. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13712–13722, June 2023b.

Yu Yang, Jianbiao Mei, Yukai Ma, Siliang Du, Wenqing Chen, Yijie Qian, Yuxiang Feng, and Yong Liu. Driving in the occupancy world: Vision-centric 4d occupancy forecasting and planning via world models for autonomous driving. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(9):9327–9335, Apr. 2025. doi: 10.1609/aaai.v39i9.33010.

Zetong Yang, Li Chen, Yanan Sun, and Hongyang Li. Visual point cloud forecasting enables scalable autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14673–14684, June 2024.

Haiming Zhang, Ying Xue, Xu Yan, Jiacheng Zhang, Weichao Qiu, Dongfeng Bai, Bingbing Liu, Shuguang Cui, and Zhen Li. An efficient occupancy world model via decoupled dynamic flow and image-assisted training. *arXiv preprint arXiv:2412.13772*, 2024.

Wenzhao Zheng, Weiliang Chen, Yuanhui Huang, Borui Zhang, Yueqi Duan, and Jiwen Lu. Occworld: Learning a 3d occupancy world model for autonomous driving. In *European conference on computer vision*, pp. 55–72. Springer, 2024.

Sicheng Zuo, Wenzhao Zheng, Yuanhui Huang, Jie Zhou, and Jiwen Lu. Gaussianworld: Gaussian world model for streaming 3d occupancy prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6772–6781, June 2025.

Vlas Zyrianov, Henry Che, Zhijian Liu, and Shenlong Wang. Lidardm: Generative lidar simulation in a generated world. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6055–6062, 2025. doi: 10.1109/ICRA55743.2025.11128001.

A Methodology Details

A.1 Intermediate Fusion Attention Mechanism

In the intermediate fusion pipeline, each agent independently encodes its point cloud into BEV features. These per-agent features are then projected into the ego coordinate and aggregated via the fused axial attention (FAX) module of CoBEVT (Xu et al., 2023a), which alternates local window attention and global grid attention over the tokens of all connected agents; we refer to Xu et al. (2023a) for its formulation. In our tokenizer, the input of FAX is the stack of warped per-agent BEV features of size $|\mathcal{N}|\times64\times64\times512$, and its output is aggregated along the agent dimension into the fused ego BEV feature F^{BEV}. The module is stacked for 3 layers with window size w=4.

A.2 Auto-Regressive Inference

During inference, WM-V2X generates long-horizon predictions via a sliding-window auto-regressive procedure. Given an initial context of T frames $\{z_{t-T+1}^{(n^*)},\dots,z_t^{(n^*)}\}$ and corresponding trajectories $\{j_{t-T+1}^{(n^*)},\dots,j_t^{(n^*)}\}$, the model iteratively predicts the next frame and updates the context window as follows: **(1)** Feed both initial context into the WM-V2X to predict $\hat{z}_{t+1}^{(n^*)}$ and $\hat{j}_{t+1}^{(n^*)}$. **(2)** Update the context by discarding the oldest frame and appending the prediction, i.e., $\{z_{t-T+2}^{(n^*)},\dots,z_t^{(n^*)},\hat{z}_{t+1}^{(n^*)}\}$ and $\{j_{t-T+2}^{(n^*)},\dots,j_t^{(n^*)},\hat{j}_{t+1}^{(n^*)}\}$. **(3)** Repeat for S steps to produce a sequence of future predictions $\{\hat{z}_{t+1}^{(n^*)},\dots,\hat{z}_{t+S}^{(n^*)}\}$ and $\{\hat{j}_{t+1}^{(n^*)},\dots,\hat{j}_{t+S}^{(n^*)}\}$. Each predicted latent is decoded back to a 3D occupancy scene via the frozen cooperative tokenizer decoder.

A.3 Cross-Agent Consistency Metrics

To quantify scene prediction consistency across connected agents, we propose two metrics based on connected component analysis. Given the predicted occupancy grids $\hat{O}^{(n_1)}$ and $\hat{O}^{(n_2)}$ from two vehicles at the same future timestep, we first extract dynamic object instances from each grid. For each dynamic class k (vehicle, truck, or pedestrian), we create a binary mask and apply 3D connected component labeling to identify individual object instances, filtering components with fewer than τ=10 voxels. Each instance is represented by its axis-aligned 3D bounding box b=$(x_{\min},x_{\max},y_{\min},y_{\max},z_{\min},z_{\max})$ computed from the voxel coordinates.

Detection rate. We match bounding boxes between the two agents using the Hungarian algorithm on the 3D IoU matrix with a minimum IoU threshold of 0.3, thus the detection rate is formulated as,

$$R_{\text{det}}=\frac{|\mathcal{M}|}{\max(|\mathcal{B}^{(n_1)}|,|\mathcal{B}^{(n_2)}|)}, \tag{7}$$

where $\mathcal{M}$ is the set of matched pairs and $\mathcal{B}^{(n)}$ is the set of detected bounding boxes from agent n. A higher detection rate indicates that more objects are consistently detected by both agents.

Bounding box size error. For each matched pair $(b_i^{(n_1)},b_i^{(n_2)})\in\mathcal{M}$, we compute the mean absolute difference in bounding box dimensions:

$$E_{\text{size}}=\frac{1}{|\mathcal{M}|}\sum_{(b_i^{(n_1)},b_i^{(n_2)})\in\mathcal{M}}\frac{1}{3}\sum_{a\in\{x,y,z\}}|s_{i,a}^{(n_1)}-s_{i,a}^{(n_2)}|, \tag{8}$$

where $s_{i,a}^{(n)}=a_{\max}-a_{\min}$ is the size of bounding box $b_i^{(n)}$ along axis a. A lower size error indicates more geometrically consistent object predictions across agents.

B Dataset Preprocessing

B.1 3D Occupancy Ground Truth

The V2X-Real dataset provides 5772 training frames and 1253 test frames collected from 63 driving scenarios involving 2 vehicles and 2 infrastructures that are designed for the 3D object detection

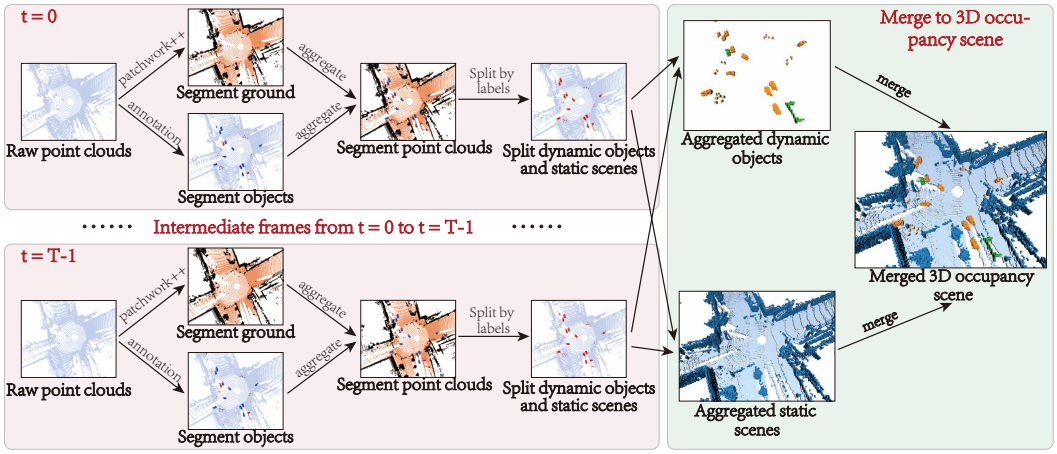


Figure 4: Occupancy label generation process.

task, but it does not include 3D occupancy annotations. V2X-Real is publicly released for research use, and we adopt it in accordance with its original license terms. We therefore generate occupancy ground truth from the raw LiDAR point clouds and corresponding annotations. The generated labels contain 6 semantic classes: free space (0), ground (1), building (2), vehicle (3), pedestrian (4), and truck (5).

Generating segmented point clouds. Fig. 4 illustrates the occupancy label generation pipeline. Each agent's LiDAR sensor records point clouds at 0.5 s intervals. We first apply Patchwork++ (Ren et al., 2023) to separate ground points and use the provided 3D bounding box annotations to extract dynamic object points, i.e., vehicle, pedestrian, and truck, from the remaining non-ground points. The residual points are treated as building. This produces segmented point clouds divided into dynamic objects and static scenes.

Generating dense voxels. Since single-frame LiDAR is inherently sparse, we densify the occupancy labels via a sliding window of 5 consecutive frames. Dynamic objects and static scenes are aggregated separately. For dynamic objects, only the current-frame bounding box position is used while multi-frame points within the same object are accumulated to densify the object shape. For static scenes, points from all frames in the window are projected to the reference LiDAR coordinate via ego poses and concatenated directly. The aggregated point clouds are then voxelized using a KD-tree at 0.5 m resolution over a spatial range of $[-64, 64] \times [-64, 64] \times [-4, 4]$ m, yielding occupancy grids of size $256 \times 256 \times 16$, where each voxel is assigned the semantic label of its nearest segmented point.

B.2 Satellite Images

We enrich the V2X-Real dataset with satellite images obtained from the Google Earth API to provide scene environment and visual trajectory information, where all imagery is collected and used in compliance with the Google Earth Terms of Service, and is included in our supplementary materials only as low-resolution crops aligned with the dataset coordinates. To align dataset coordinates with GPS locations, we use the two roadside infrastructures as reference anchors to establish the coordinate transformation, from which the GPS positions of all vehicles are computed by linear projection to the infrastructure coordinate system. Each satellite image is captured at zoom level 20 with a resolution of 640×640 pixels, centered on the vehicle's GPS location. Since the vehicle moves across frames, a separate satellite image is cropped per frame to reflect the current position, providing a dynamic BEV context that tracks the ego agent's trajectory. Fig. 5 illustrates this alignment, where the two infrastructures, identifiable in the satellite image and in the LiDAR point clouds, serve as cross-modal anchors that tie the satellite imagery to the V2X-Real sensor coordinate system. WM-V2X does not query the Google Earth API during online inference: the satellite tiles are historical, geo-referenced imagery collected once during preprocessing, which can be cached on the vehicle or an edge server, and the ego pose only indexes the corresponding cached crop at runtime. Network availability, API latency, and instantaneous cloud cover therefore do not affect inference. The satellite input is also optional rather than a hard dependency: removing it yields 44.65% average mIoU and 0.0375 m

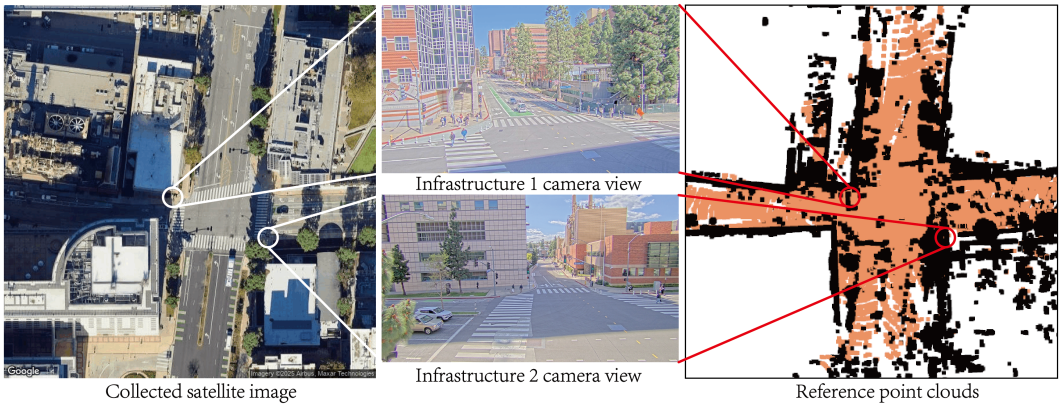


Figure 5: Cross-modal alignment between Google Earth satellite imagery and V2X-Real on-site sensor data. **Left:** collected satellite image, with the two roadside infrastructures highlighted by white circles and zoomed-in callouts. **Middle:** corresponding RGB views from Infrastructure 1 (top) and Infrastructure 2 (bottom). **Right:** reference LiDAR point clouds, with red lines marking the same two infrastructure locations. The two infrastructures act as anchor points that establish the coordinate transformation between the GPS locations and sensor frames.

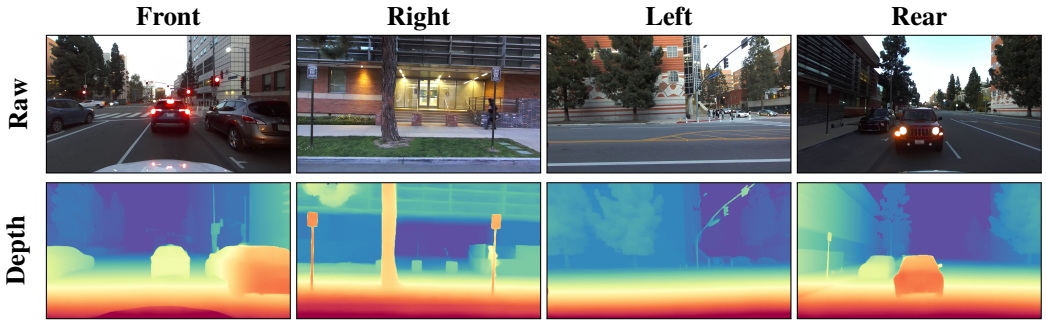


Figure 6: Visualization of depth information. The first row presents the raw images from the front, right, left, and rear views; the corresponding depth maps are shown below. The DepthAnything 3 (Lin et al., 2026) toolkit was applied to extract the depth information from the raw images.

position error, compared with 46.04% and 0.034 m for the full model (Table 4). Stale or misaligned map imagery nevertheless remains a deployment limitation.

B.3 Depth Information

To support future camera-based occupancy prediction, we preprocess dense depth maps for all four surround-view cameras of each connected vehicle using the DepthAnything 3 (Lin et al., 2026) metric depth estimator. Fig. 6 visualizes the raw images and their corresponding depth maps; the results are acceptable, as pixel colors transition smoothly from red (near) to blue (far) with increasing depth.

C More Experimental Results

C.1 Implementation details

For the cooperative tokenizer, a combined CE and Dice loss is used with weights $\lambda_{\text{CE}}{=}0.6$ and $\lambda_{\text{Dice}}{=}0.4$. The satellite encoder uses a pretrained ResNet-18 with 8 attention heads and window size 8, producing 256-channel satellite features. The intermediate fusion module comprises a 3-layer FAX attention network with window size 4. Both tokenizers are trained with AdamW (lr $=2{\times}10^{-4}$, weight decay $=0.01$) and a cosine annealing scheduler with 10 warmup epochs. The early fusion tokenizer uses batch size 16 for 80 epochs, and the intermediate fusion tokenizer uses batch size 4 for 100 epochs. For WM-V2X, the spatio-temporal transformer uses 512 latent channels, 256-dimensional

Table 6: Camera-only evaluation on 4D occupancy prediction and motion planning. The reconstructed mIoU and fwIoU are 25.95% and 35.77%, respectively. "Avg." is the mean across the 0.5–2 s horizon.

Task	Method	Metric I					Metric II				
		0.5s	1s	1.5s	2s	Avg.	0.5s	1s	1.5s	2s	Avg.
		mIoU (%) ↑					fwIoU (%) ↑				
4D Occupancy	OccWorld	24.08	22.17	20.29	18.60	21.29	34.41	32.89	31.19	29.53	32.00
	DOME	23.01	21.92	20.86	19.97	21.44	33.63	32.71	31.85	31.10	32.32
	OccProphet	**25.06**	**23.78**	22.21	20.73	22.95	34.95	33.75	32.04	30.25	32.75
	EfficientOCF	21.29	18.73	16.13	13.99	17.54	30.76	27.54	24.20	21.54	26.01
	I2-World	23.49	21.58	19.70	18.00	20.69	33.48	31.87	30.09	28.34	30.95
	COME	24.08	23.08	22.19	21.22	22.69	34.38	33.49	32.71	32.04	33.16
	WM-V2X	24.94	23.51	**22.35**	**21.30**	**23.03**	**35.48**	**34.30**	**33.31**	**32.40**	**33.87**
		Position error (m) ↓					Collision rate (%) ↓				
Motion Planning	OccWorld	0.148	0.168	0.172	0.176	0.166	**0.052**	**0.052**	**0.052**	**0.052**	**0.052**
	DOME	**0.025**	0.034	0.039	0.045	**0.036**	**0.052**	**0.052**	**0.052**	**0.052**	**0.052**
	OccProphet	0.235	0.273	0.300	0.328	0.284	0.104	0.123	0.147	0.156	0.133
	EfficientOCF	0.150	0.172	0.197	0.232	0.188	0.104	0.160	0.260	0.312	0.209
	I2-World	0.045	0.046	0.050	0.056	0.049	**0.052**	**0.052**	**0.052**	**0.052**	**0.052**
	COME	0.032	0.037	0.041	0.044	0.039	**0.052**	**0.052**	**0.052**	**0.052**	**0.052**
	WM-V2X	0.032	**0.033**	**0.038**	**0.042**	**0.036**	**0.052**	**0.052**	**0.052**	**0.052**	**0.052**

trajectory and satellite embeddings. The trajectory comprises relative position (x, y) and heading θ waypoints. The latent and trajectory loss weights are λ_{latent}=0.7 and λ_{traj}=0.3. WM-V2X is trained with AdamW (lr $=1\times10^{-4}$, weight decay $=0.01$) and cosine annealing with 10 warmup epochs, using batch size 2 for 60 epochs.

C.2 Multi-view Image-based Results

To extend WM-V2X to camera-only inputs, we replace the LiDAR-based encoder with a depth-based BEV lifting encoder that uses the surround-view RGB images and the metric depth maps preprocessed in Sec. B.3, while keeping the downstream world model and planner unchanged.

Multi-view image encoder. Given N connected agents each carrying M cameras, we flatten the inputs to shape $(B{\cdot}N{\cdot}M, 3, H, W)$ and feed them to a shared ResNet-34 backbone. The intermediate layers produce feature maps at $1/4$, $1/8$, and $1/16$ resolution, respectively; these are then fused top-down with 1×1 lateral convolutions and bilinear upsampling, yielding a 128-channel feature map at $1/4$ resolution per view.

Depth-based BEV lifting. For every pixel (u, v) with metric depth ρ on camera m of agent n, we recover its 3D location in the ego frame as,

$$\mathbf{p}_{\text{ego}} \;=\; \mathbf{T}^{(n)}_{a\to e}\,\mathbf{T}^{(n,m)}_{c\to a}\left(\rho\cdot\mathbf{K}^{-1}_{n,m}\,[u,v,1]^{\top}\right), \tag{9}$$

where $\mathbf{K}_{n,m}$, $\mathbf{T}^{(n,m)}_{c\to a}$, and $\mathbf{T}^{(n)}_{a\to e}$ are the camera intrinsics, camera-to-agent extrinsics, and agent-to-ego pose, respectively. We discretize $\mathbf{p}_{\text{ego}}$ on a 0.5 m BEV grid of size 256×256 and aggregate all points falling into the same cell, where each cell includes the mean image feature with 128 channels, the mean point height $\bar{z}$, and a Z-bin normalized height histogram with 16 channels. Because all $N\times M$ views from every agent project into the same ego-coordinate BEV grid, this step also performs cross-agent feature fusion. The resulting tensor has 146 channels at 256×256 resolution.

BEV encoder–decoder and 3D head. A 1×1 projection compresses the 146-channel BEV input to 128 channels, followed by a U-Net-style encoder with channel widths $[128, 256, 512]$, where three residual blocks are applied per stage for a total $4\times$ downsampling. The resulting bottleneck of size $(512, 64, 64)$ is used as the latent token. On the decoder side, a skip-free decoder $[512\to256\to128]$ upsamples the latent features back to 256×256 resolution, followed by three 3×3 Conv–BN–ReLU blocks with 256 channels to predict the 3D occupancy scene. The network is trained with the same CE and Dice losses as the LiDAR tokenizer.

The multi-view image-based 4D occupancy prediction and motion planning results are presented in Table 6. On 4D occupancy, OccProphet's auxiliary flow-forecasting module explicitly models

per-voxel dynamic motion and yields the best mIoU at the 0.5–1 s horizons, while WM-V2X attains the best average mIoU of 23.03% and the best average fwIoU of 33.87%, surpassing the strongest camera-only baselines OccProphet with mIoU 22.95% and COME with fwIoU 33.16%. For motion planning, WM-V2X achieves the lowest position error at the 1–2 s horizons, ties DOME for the best average position error of 0.036 m, and matches the 0.052% collision rate shared by all baselines except OccProphet and EfficientOCF, both of which degrade rapidly over the prediction horizon. These observations suggest that WM-V2X generalizes across both LiDAR- and camera-based tokenizers, indicating that its world model design is robust to the choice of input modality.

C.3 Long-term Prediction and IoU Drop

Fig. 7 shows the IoU curves across 10 prediction steps for vehicle, pedestrian, and truck classes, where OccWorld and I2-World exhibit the largest degradation from timestep 1 to timestep 10, and DOME, OccProphet, and COME maintain moderate performance degradation. Notably, WM-V2X consistently outperforms all baselines across the three object classes. The consistent advantage is attributable to satellite image conditioning, which provides a stable static-scene prior that mitigates error accumulation over extended rollouts.

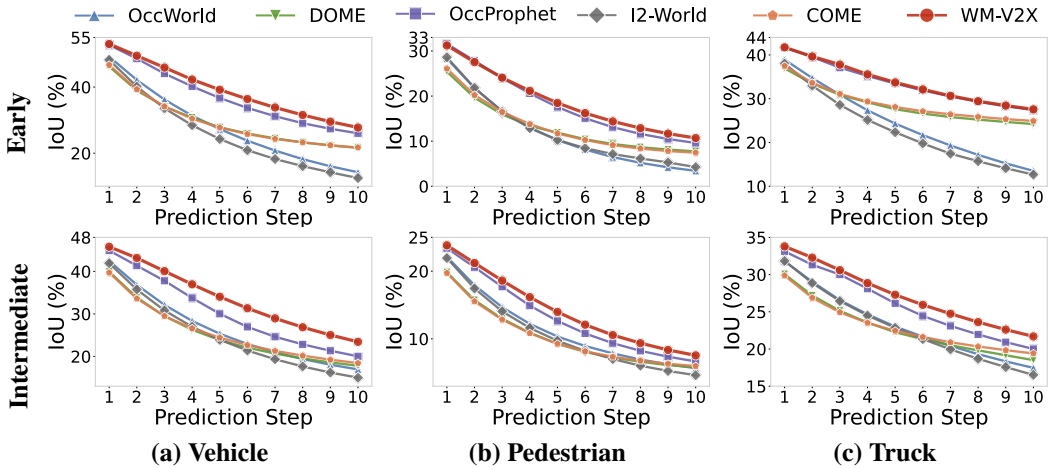


Figure 7: Long-term prediction performance comparison over 10 prediction steps for vehicle, pedestrian, and truck classes under early and intermediate cooperation paradigms.

Table 7 reports the per-class IoU drop from step 1 to step 10 for all methods under both early and intermediate cooperation paradigms. OccWorld and I2-World exhibit the largest degradation across all classes, with mean drops exceeding 74% and 63% under early and intermediate cooperation, respectively. DOME, OccProphet, and COME achieve intermediate stability, while WM-V2X consistently achieves the lowest mean drop of 49.1% and 51.0% under early and intermediate cooperation, respectively. The pedestrian class exhibits the highest degradation across all methods due to the small occupancy volume and unpredictable motion patterns of pedestrians. On the truck class, COME yields a slightly smaller drop than WM-V2X (gaps of 0.4% and 0.8% under early and intermediate cooperation, respectively), which can be attributed to the iterative denoising in COME's diffusion-based decoding that better preserves the geometry of large and slowly evolving rigid objects than the auto-regressive prediction used in WM-V2X.

D Comparison Protocol

D.1 Shared-Tokenizer Protocol and Native-Tokenizer Check

In the controlled comparison of Tables 1 and 2, the cooperative tokenizer is trained only to reconstruct occupancy from multi-agent LiDAR and is then frozen, so that every forecasting model receives the same 64×64 early- or intermediate-fusion latent history. Neither trajectory nor satellite information enters tokenizer training, all forecasting models are retrained on the same V2X-Real split with the same four-horizon protocol, and no baseline-specific multi-agent fusion module is introduced.

Table 7: IoU performance drop (%) from step 1 to step 10 across classes and cooperation paradigms. Lower values indicate better long-term temporal stability.

Method	Early coop. IoU drop (%) ↓				Intermediate coop. IoU drop (%) ↓			
	Vehicle	Pedestrian	Truck	Mean	Vehicle	Pedestrian	Truck	Mean
OccWorld	71.1	87.9	65.3	74.8	60.1	74.4	45.1	59.9
DOME	53.2	69.5	34.5	52.4	55.4	70.9	38.4	54.9
OccProphet	50.7	69.6	34.2	51.5	55.4	71.5	39.6	55.5
I2-World	74.1	85.2	66.7	75.3	64.2	78.9	48.1	63.7
COME	53.5	71.5	**33.5**	52.8	53.5	69.7	**35.0**	52.7
WM-V2X	**47.6**	**65.7**	33.9	**49.1**	**48.8**	**68.3**	35.8	**51.0**

To measure the effect of this shared representation, we retrain OccWorld and I2-World, whose quantized tokenizers differ most from ours, with their native tokenizers under early fusion (Table 8); the native tokenizers take voxelized occupancy derived from the same multi-agent LiDAR observations. I2-World is insensitive to the tokenizer choice (38.22% vs. 38.36% mIoU), whereas OccWorld benefits from its native VQ-VAE by 4.05 points despite a lower reconstruction accuracy, with the gain widening from $+0.62$ at 0.5 s to $+6.91$ at 2 s. With native tokenizers, both methods remain below WM-V2X with and without satellite conditioning (46.04%/44.65%). DOME and COME use continuous Occ-VAEs, which are closer to our tokenizer in representation type.

Table 8: Native-tokenizer sensitivity check under early fusion. Recon. denotes tokenizer reconstruction; prediction metrics are averaged over 0.5–2 s unless a horizon is given; PE and Col. denote the average position error and collision rate. The two WM-V2X rows use the shared tokenizer and are listed for reference; "–" marks values not evaluated.

Method	Tokenizer	Recon. (%)		Prediction mIoU (%) ↑					Pred. fwIoU	PE	Col.
		mIoU	fwIoU	0.5s	1s	1.5s	2s	Avg.	(%) ↑	(m) ↓	(%) ↓
I2-World	Shared	52.46	64.18	46.52	40.25	35.20	31.45	38.36	50.70	0.066	0.047
I2-World	Native	49.89	56.67	43.55	39.27	36.21	33.84	38.22	47.17	0.140	0.072
OccWorld	Shared	52.46	64.18	47.00	41.28	36.49	32.57	39.34	51.28	0.070	0.059
OccWorld	Native	49.37	57.88	47.62	44.61	41.86	39.48	43.39	52.09	0.062	0.024
WM-V2X w/o Sat.	Shared	52.46	64.18	49.38	45.85	42.87	40.49	44.65	–	0.0375	–
WM-V2X	Shared	52.46	64.18	50.18	47.19	44.50	42.27	46.04	57.89	0.034	0.000

D.2 Forecasting-Side Conditioning Protocol

Besides the shared latent history, all forecasting models receive the same historical ego trajectory and retain their native forecasting-side conditions (Table 9): OccProphet and EfficientOCF keep the ego pose used for temporal alignment in their original pipelines, and COME keeps its scene-centric forecasting control branch. None of the baselines uses satellite imagery natively. Instead of attaching new satellite adapters to published methods, we report WM-V2X without satellite conditioning (44.65% mIoU) as the input-matched control alongside the complete satellite-conditioned model (46.04%).

E Robustness under Realistic V2X Conditions

E.1 Perturbation Protocol

All perturbations are applied at inference time to collaborator messages only, while the ego observation, satellite crop, ego trajectory, and clean cooperative ground truth remain fixed. Each perturbation is evaluated independently on the full test set with the 50 m communication radius, and all metrics are averaged over the four 0.5–2 s horizons. **(1) Pose noise:** zero-mean Gaussian noise is added to the collaborator translation (independently in x and y) or yaw, and is resampled for every collaborator and frame. **(2) Message delay:** at the 2 Hz sampling rate, every collaborator message is replaced by an older frame. **(3) Packet loss:** every collaborator message is dropped independently at each frame with probability p, using one fixed Bernoulli realization per loss rate, so that 100% loss

Table 9: Native representation and forecasting-side conditions retained by each method in the controlled shared-tokenizer comparison.

Method	Native tokenizer / representation	Trajectory / pose protocol	Flow	Scene control	Satellite
OccWorld	Discrete VQ-VAE	Ego trajectory retained	–	–	–
DOME	Continuous Occ-VAE	Pose/trajectory condition retained	–	–	–
OccProphet	No standalone tokenizer; 3D features	Current pose retained; common trajectory added	✓	–	–
EfficientOCF	BEV occupancy–height representation	Current pose retained; common trajectory added	✓	–	–
I2-World	Intra/inter quantization	Trajectory/transformation condition retained	–	–	–
COME	Continuous Occ-VAE	Ego trajectory retained	–	✓	–
WM-V2X	Continuous cooperative VAE	Ego trajectory retained	–	–	✓

corresponds to ego-only prediction. The detection rate and bounding box size error are computed for vehicle-to-vehicle predictions per frame with the protocol of Appendix A.3. Since the size error is conditional on successful matches, it should be read together with the detection rate and the number of matches.

E.2 Early Fusion

As shown in Table 10, early fusion degrades gracefully under moderate communication and localization errors, and remains above ego-only prediction at 4 s delay and 75% packet loss ($+4.29$ and $+3.50$ mIoU points). Spatial misalignment is more damaging: at 1.0 m translation or 5° yaw noise, early fusion becomes worse than ego-only prediction, since misaligned raw points are merged directly into the 0.5 m voxel grid. Dynamic classes are also more sensitive to stale messages than static ones. At the 0.5 s forecasting horizon, a 2.5 s delay lowers the vehicle and pedestrian IoU by 11.59 and 9.81 points, but the ground and truck IoU by only 3.60 points. From 2.5 s to 4.0 s delay, the mIoU decreases only slightly (40.29% to 38.89%) because static classes barely change, while dynamic classes continue to deteriorate (vehicle IoU from 41.45% to 38.46% and pedestrian IoU from 21.41% to 18.68%).

Table 10: Robustness of early fusion to collaborator pose noise, message delay, and packet loss. Δ is the mIoU change from the clean setting; 100% packet loss is the ego-only endpoint.

Setting	mIoU (%)	Δ	fwIoU (%)	Det. rate (%)	Size err. (m)	Matches
Clean	46.04	–	57.89	43.65	0.383	61,833
Pose noise						
Translation σ=0.2 m	44.61	−1.43	56.17	42.32	0.408	58,288
Translation σ=0.5 m	39.34	−6.70	51.14	36.27	0.476	45,832
Translation σ=1.0 m	32.11	−13.93	44.25	24.21	0.634	27,168
Yaw σ=1°	41.64	−4.40	52.54	39.49	0.443	51,955
Yaw σ=2°	37.14	−8.90	47.91	32.84	0.499	41,421
Yaw σ=5°	29.95	−16.09	41.09	19.46	0.631	23,583
Message delay						
Delay 0.5 s	44.40	−1.64	56.64	43.22	0.416	60,733
Delay 2.5 s	40.29	−5.75	54.04	34.13	0.514	49,557
Delay 4.0 s	38.89	−7.15	53.23	30.15	0.553	45,099
Packet loss						
25%	43.61	−2.43	55.12	39.31	0.413	52,912
50%	40.98	−5.06	52.26	34.80	0.451	43,568
75%	38.10	−7.94	49.09	28.93	0.442	33,532
100% (ego-only)	34.60	−11.44	45.18	21.47	0.484	23,211

E.3 INTERMEDIATE FUSION

As shown in Table 11, intermediate fusion trades clean gain for robustness. Its $+4.53$-point cooperation gain is largely preserved under all tested pose noise, including 5 m translation and 15° yaw noise. This observation should be interpreted with care. Our intermediate-fusion tokenizer inherits the STTF configuration of CoBEVT (Xu et al., 2023a), which converts metric offsets into feature cells at $0.390625\,\text{m} \times 8 = 3.125$ m per cell, whereas our latent grid has $128\,\text{m}/64 = 2$ m per cell. Collaborator features are therefore warped by only 64% of the required offset during both training and evaluation, and the injected pose noise is scaled by the same factor. The fusion module may consequently have learned to rely less on precise collaborator geometry, which, together with the coarse 2 m cells and the attention-based fusion, can contribute to the observed insensitivity. For the same reason, the intermediate-fusion results in Tables 1–3 may underestimate the benefit of correctly aligned intermediate fusion. Early fusion merges collaborator point clouds before voxelization and is not affected. Correcting the scale only at inference time would introduce a train/test mismatch, so we report the model as trained and leave retraining with the correct scale to future work. Message delay has little aggregate effect, consistent with the current-pose alignment of the delayed features, the stationary infrastructure in V2X-Real, and the observation that 58% of vehicle frames are nearly static, whereas packet loss causes monotonic degradation. The bounding box size error stays within 0.441–0.448 m across these settings and is therefore omitted.

Table 11: Robustness of intermediate fusion to collaborator pose noise, message delay, and packet loss. Δ is the mIoU change from the clean setting; 100% packet loss is the ego-only endpoint.

Setting	mIoU (%)	Δ	fwIoU (%)	Det. rate (%)
Clean	39.53	–	51.71	27.85
Pose noise				
Translation σ=0.5 m	39.53	0.00	51.71	27.83
Translation σ=1.0 m	39.53	0.00	51.70	27.80
Translation σ=5.0 m	39.28	-0.25	51.24	27.17
Yaw σ=2°	39.50	-0.04	51.67	27.70
Yaw σ=5°	39.39	-0.14	51.46	27.62
Yaw σ=15°	39.10	-0.43	50.86	26.65
Message delay				
Delay 0.5 s	39.53	0.00	51.71	27.85
Delay 2.5 s	39.53	0.00	51.71	27.85
Delay 4.0 s	39.52	0.00	51.69	27.83
Packet loss				
25%	38.70	-0.83	50.70	26.42
50%	37.63	-1.90	49.31	24.49
75%	36.34	-3.20	47.62	23.22
100% (ego-only)	35.00	-4.53	45.88	21.27

F COMMUNICATION AND SYSTEM EFFICIENCY

F.1 FORECASTING-ONLY INFERENCE EFFICIENCY

Table 12 compares the forecasting-only inference efficiency of WM-V2X with OccWorld and I2-World. Since DOME and COME are diffusion-based methods that require more inference resources, and OccProphet and EfficientOCF involve extra modules for occupancy flow prediction, we only compare WM-V2X against these two methods. All predictors are benchmarked with their evaluated configurations, i.e., six spatio-temporal blocks for WM-V2X, under the same latent resolution, precision, batch size of 1, CUDA synchronization, warm-up, and timing protocol. The measurement covers one auto-regressive latent-prediction call, and the complete pipeline is analyzed in Appendix F.3. WM-V2X requires 55.90 ms per forecasting step, which is slower than I2-World (41.49 ms) but 52.7% faster than OccWorld, and it has the smallest peak GPU memory of 961.8 MB, i.e., 33.5% and 20.2% less than OccWorld and I2-World, respectively. Meanwhile, WM-V2X outperforms both methods on 4D occupancy prediction and motion planning, offering a favorable latency–memory–accuracy trade-off.

Table 12: Forecasting-only latency and memory for one auto-regressive step on an RTX 5090 GPU. The mIoU and position error (PE) are averaged over 0.5–2 s under intermediate cooperation (Tables 1 and 2).

Model	Params (M)	Mean (ms)	Std. (ms)	P95 (ms)	Peak GPU (MB)	mIoU (%)	PE (m)
OccWorld	159.11	118.06	0.50	118.86	1,446.0	35.87	0.853
I2-World	8.13	**41.49**	0.46	**41.96**	1,204.8	35.24	0.453
WM-V2X	63.05	55.90	**0.28**	56.32	**961.8**	**39.53**	**0.304**

F.2 Payload and Rate–Accuracy Trade-off

Transmitting raw collaborator point clouds requires 2.88 MiB (FP32) or 1.44 MiB (FP16) per collaborator and frame, whereas bit packing the 0.5 m voxel mask used by early fusion reduces the payload to 128 KiB without accuracy loss. The native FP32 intermediate BEV feature is larger (8.00 MiB, i.e., 671 ms at 100 Mbps), and INT4 quantization reduces it to 1.00 MiB (84 ms). Quantization and serialization take less than 0.5 ms and are negligible compared with transmission. Table 13 reports the accuracy of each transmitted representation on the complete test set, together with the minimum ideal per-collaborator link rate that keeps the measured four-step end-to-end path within the 500 ms sampling interval. Lossless bit packing and INT4 quantization are the preferred operating points for early and intermediate fusion, respectively: INT4 reduces the FP32 payload by $8\times$ with only a 0.10-point mIoU decrease, while further compression causes substantial degradation. Early fusion thus trades a lower payload for collaborator-side voxelization, whereas intermediate INT4 requires more bandwidth but less preprocessing.

Table 13: Rate–accuracy trade-off of the transmitted representations. The last column is the minimum ideal per-collaborator link rate for a 500 ms end-to-end path, excluding protocol and contention overheads.

Fusion	Transmitted representation	mIoU (%)	Δ	Payload	Min. rate
Early	0.5 m bit-packed voxel mask	46.04	–	128 KiB	9.0 Mbps
	1.0 m spatial downsampling	38.47	−7.57	16 KiB	1.1 Mbps
	2.0 m spatial downsampling	22.70	−23.34	2 KiB	0.14 Mbps
Intermediate	BEV FP32	39.53	–	8.00 MiB	276 Mbps
	BEV FP16	39.53	0.00	4.00 MiB	138 Mbps
	BEV INT8	39.53	0.00	2.00 MiB	69 Mbps
	BEV INT4	39.43	−0.10	1.00 MiB	35 Mbps
	BEV INT2	31.99	−7.54	0.50 MiB	17 Mbps

F.3 End-to-End Latency

We measure the complete per-frame pipeline in online operation after the history buffer has been initialized, using lossless bit packing for early fusion and INT4 BEV features for intermediate fusion. Preprocessing, codec, fusion, rollout, and decoding times are measured, while the transfer time is calculated from the serialized payload at a nominal link rate of 100 Mbps. $N{=}1$ denotes one collaborator, which also represents independent parallel links, while "Shared" places the average $N_{\text{avg}}{=}2.160$ collaborators per ego frame within 50 m (0.80 vehicles and 1.36 infrastructures) on one aggregate link. In early fusion, each collaborator voxelizes and packs its point cloud, and the ego unpacks, aligns, and fuses the occupancy masks before tokenizer encoding; in intermediate fusion, ego and collaborator encoding run in parallel. As shown in Table 14, both pipelines remain below the 500 ms sampling interval in all cases. These are ideal lower-bound estimates that exclude protocol, queuing, interference, and retransmission overheads.

G More Visual Results

This section presents additional visual results of occupancy scene predictions across agents, cooperation paradigms, and time horizons for various baselines. Fig. 8 shows the predicted scenes of vehicle 1 and vehicle 2 under no-cooperation, intermediate, and early cooperation. With cooperation, the

Table 14: Stage-wise end-to-end latency of the complete four-step pipelines at 100 Mbps.

	Early, 0.5 m bit-packed		Intermediate, INT4 BEV	
Stage	N=1	Shared	N=1	Shared
Collaborator preprocessing	101.68 ms	101.68 ms	8.61 ms	8.61 ms
V2X transfer	10.5 ms	22.7 ms	83.9 ms	181.2 ms
Ego unpack/dequantization + alignment/fusion	24.97 ms	24.97 ms	0.128 ms	0.276 ms
Ego tokenizer/BEV encoding	8.80 ms	8.80 ms	parallel	parallel
WM-V2X rollout, four steps (incl. satellite)	227.28 ms	227.28 ms	227.28 ms	227.28 ms
Occupancy decoding, four steps	21.04 ms	21.04 ms	21.04 ms	21.04 ms
Total latency	**394.27 ms**	**406.47 ms**	**340.96 ms**	**438.41 ms**

consistency of the predicted 4D occupancy between agents is significantly improved and approaches the ground truth. In addition, Figs. 9 and 10 depict the predicted occupancy across different baselines. Since the scene changes between consecutive frames are subtle, we select the 0.5 s, 2 s, 3.5 s, and 5 s future frames as the visualization horizons. Across these long-horizon comparisons, WM-V2X retains coherent scene structure and dynamic-object identity at horizons where competing methods exhibit shape drift, class confusion, and fragmented foreground objects.

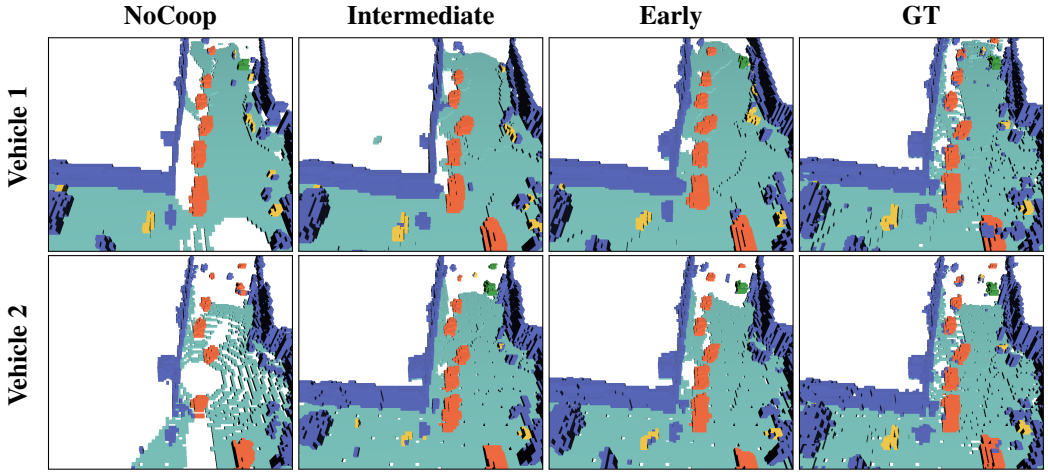


Figure 8: Visual comparison of occupancy prediction results from different agents with various cooperative paradigms at $t = 4$ s.

H Computer Resources

All experiments in this paper were conducted on a local server running Ubuntu 24.04. The hardware configuration includes an Intel(R) Core(TM) i9-14900F CPU, 128 GB of RAM, and one NVIDIA RTX 5090 GPU with 32 GB of memory.

Our models were implemented using Python 3.9 and PyTorch 2.9.1 with CUDA 12.8. For the cooperative tokenizer, the training time was approximately 14 and 32 GPU-hours for the early and intermediate cooperation paradigms, respectively. For WM-V2X, the training time was around 27 and 52 GPU-hours for these two cooperation paradigms. The evaluation phase of WM-V2X took roughly 1 GPU-hour.

I Broader Impacts

Positive impacts. Cooperative world models for autonomous driving offer clear safety benefits: by sharing perception information across connected agents via V2X communication, the proposed framework mitigates blind spots caused by occlusion and limited sensing range, which improves the reliability of downstream prediction and planning in complex traffic scenarios such as urban

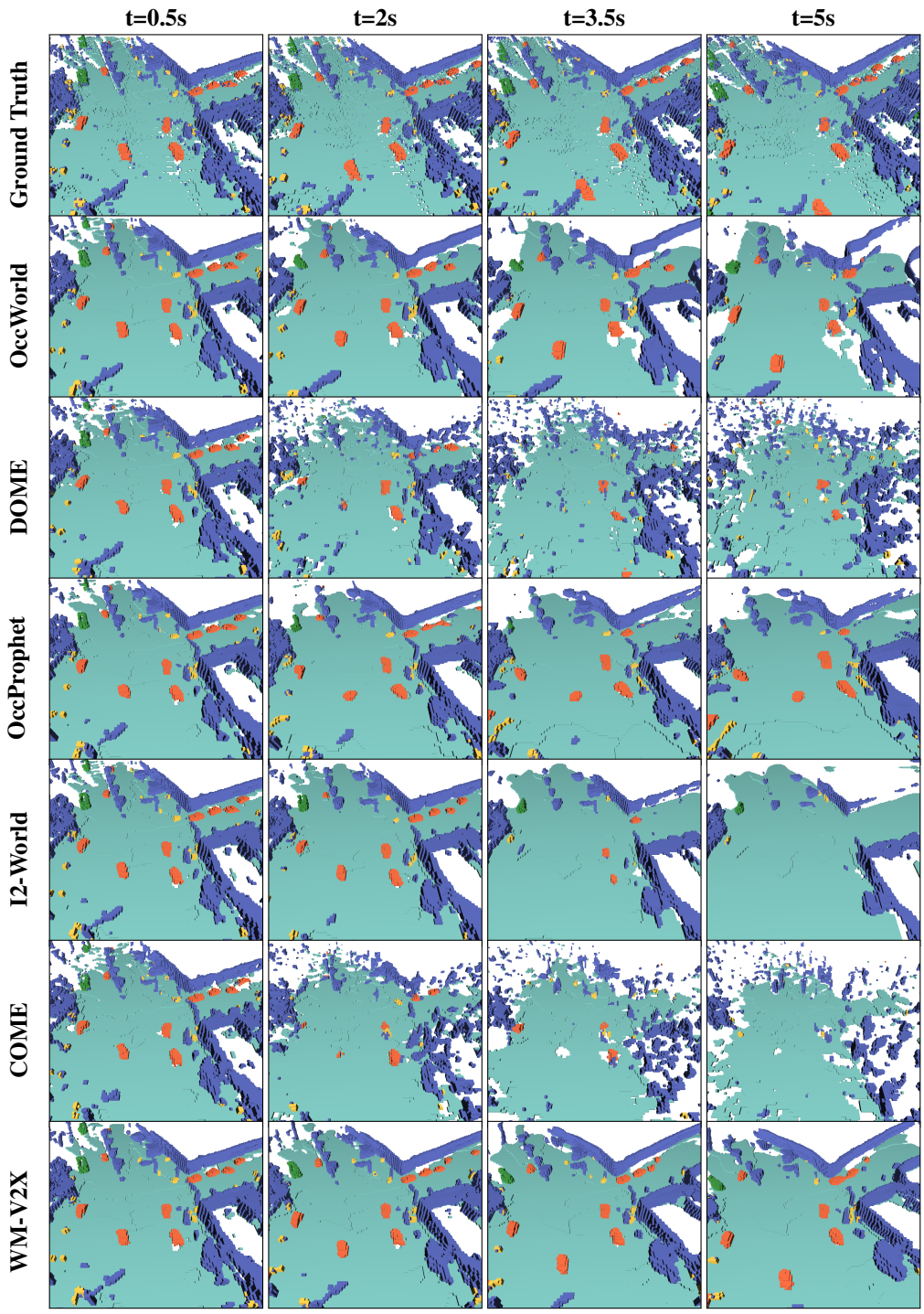


Figure 9: Additional qualitative long-term prediction comparison across multiple methods at $t = 0.5, 2, 3.5, 5$ s. Each row shows predictions from a different method; each column a different time step. Ground, building, vehicle, pedestrian, and truck voxels are colored teal, indigo, orange, yellow, and green, respectively. WM-V2X maintains coherent scene structure and dynamic objects at longer horizons where other methods degrade.

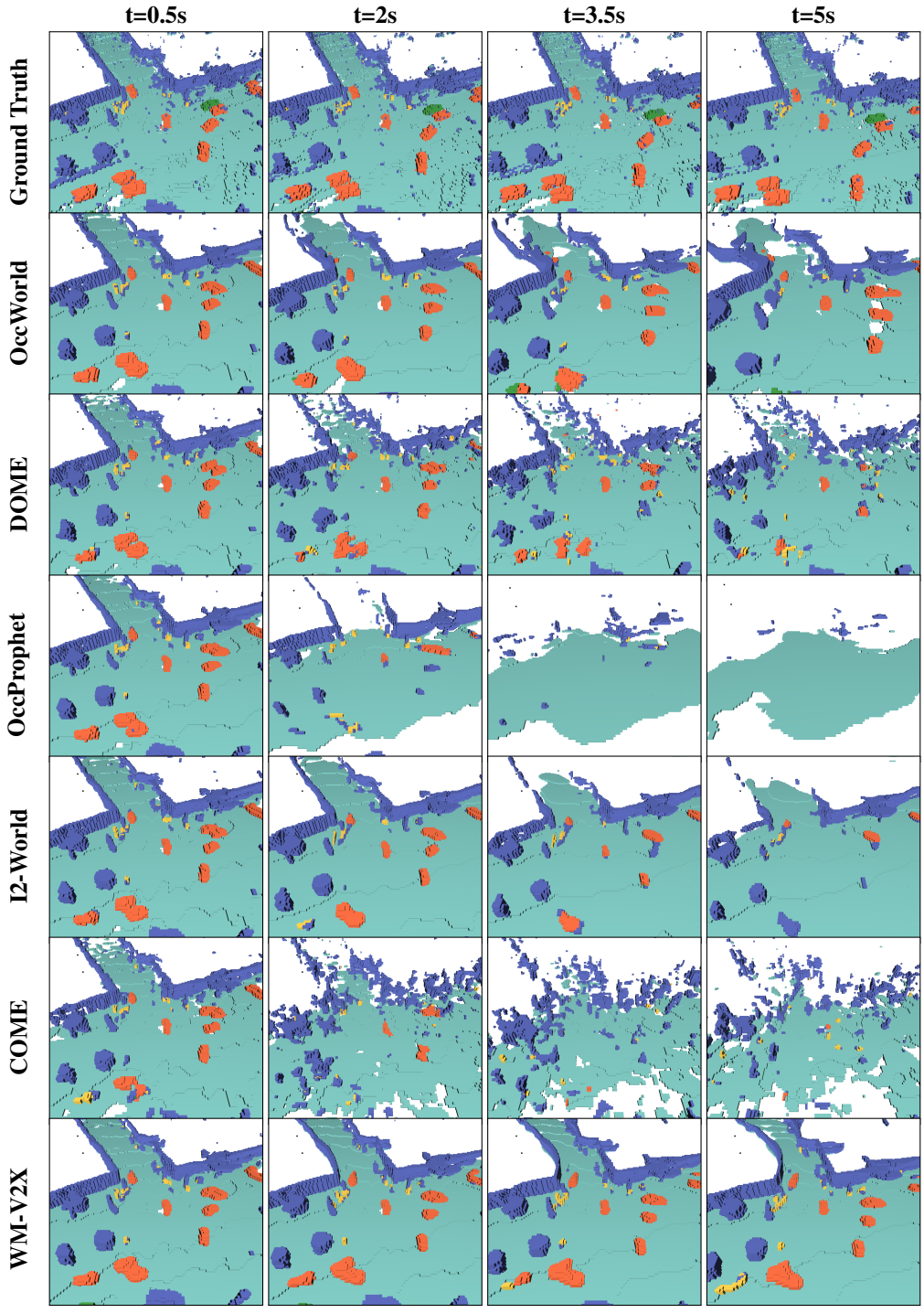


Figure 10: Additional qualitative comparison across multiple methods. The row and column present predictions from different methods and timesteps, respectively.

intersections. The cross-agent prediction consistency has been improved as well quantified in Table 3, which reduces collision risk for connected vehicles and lower the perception burden on each individual vehicle.

Negative impacts and mitigations. Two concerns deserve explicit acknowledgement. First, exchanging dense perception data, i.e, raw point clouds and intermediate features across agents introduces potential privacy risks, since vehicle locations, identities, and traffic participants such as pedestrians may be inferred from shared data. Standard mitigations include on-device feature extraction prior to transmission, data anonymization, and access control over V2X channels. Second, although our stress tests in Appendix E show graceful degradation under moderate delay and packet loss, early fusion becomes worse than ego-only prediction under large collaborator pose errors, and latency, packet loss, or adversarial interference in deployment could degrade prediction quality and indirectly affect safety. Pose-robust fusion, misbehavior detection for collaborator messages, and human oversight in the planning loop should be addressed before real-world deployment.