BRIDGE TRANSFORMER: BEYOND EGO-CENTERED FUSION IN COLLABORATIVE SEMANTIC OCCUPANCY PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

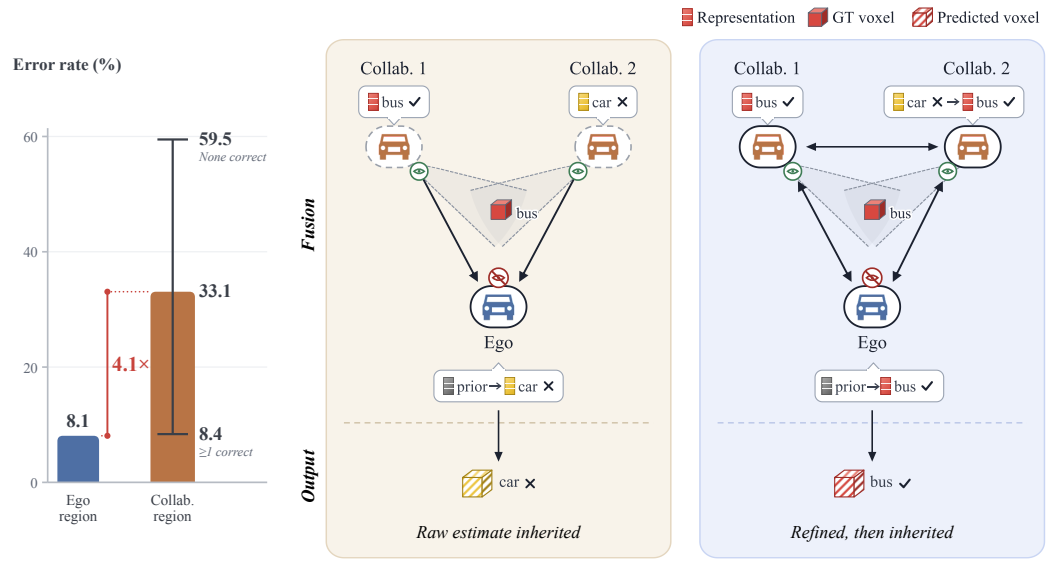
Collaborative 3D semantic occupancy prediction enables multiple agents to exchange complementary information, yet, regardless of how many agents participate, the prediction remains a single semantic occupancy grid in the ego frame. The task is thus Many-to-One (M2O) in its output alone. Existing methods impose M2O on fusion as well. The resulting M2O fusion is centered on the ego representation and treats collaborator representations as inputs to be aggregated into it. Each collaborator representation, however, carries the errors of its single viewpoint. We empirically observe that M2O fusion passes these errors into the ego grid uncorrected, so the error rate rises in regions observed only by collaborators. This motivates Many-to-Many (M2M) fusion, which jointly refines the representations of all agents to correct these errors and defers the reduction to the ego grid until the output. To this end, we propose the Bridge Transformer, a feed-forward architecture for M2M fusion. Each agent represents its scene as 3D semantic Gaussians. These Gaussians are grouped, and each group is embedded into hierarchical tokens comprising an anchor token and its member tokens. Inter-agent attention over the anchor tokens of all agents alternates with intra-agent attention over each anchor and its members. Because every agent's anchor tokens serve as queries, this alternation jointly refines all agents' representations. A dual-branch Gaussian decoder then refines and extends each agent's Gaussians for splatting onto the ego grid. Experiments show that the Bridge Transformer outperforms the strongest prior method by 3.88 mIoU and 2.01 IoU. An ablation further shows that M2M fusion corrects more collaborator errors than M2O fusion.

1 INTRODUCTION

3D semantic occupancy prediction, a fundamental perception task in autonomous driving, assigns an occupancy state and a semantic class to every voxel around a vehicle (Cao & de Charette, 2022; Huang et al., 2023; Wei et al., 2023; Zhang et al., 2023). A single vehicle, however, observes the scene from only one vantage point, leaving unobserved the voxels that are occluded or outside its cameras' fields of view. Collaborative perception overcomes this limitation by enabling nearby agents to exchange complementary information (Xu et al., 2022; Hu et al., 2022; 2024). Collaborative semantic occupancy prediction (CSOP) extends this paradigm so that the ego agent predicts the semantic occupancy of its surroundings from its own observations and those shared by collaborators.

Regardless of how many agents participate, the final prediction remains a single semantic occupancy grid in the ego frame: the task is **M**any-**to**-**O**ne (**M2O**). Although M2O describes the output alone, existing methods impose it on fusion as well. This ego-centered fusion refines the ego representation and aggregates collaborator representations into it without letting them refine one another. This design persists across representations. VOGS-CP (Chen et al., 2026) updates each ego Gaussian from neighboring shared Gaussians while leaving the shared Gaussians unchanged. VQSOP (Li et al., 2026) restricts cross-attention such that the ego volume attends to aligned collaborator volumes, which enter solely as keys and values. CoHFF (Song et al., 2024), as implemented, has the orthogonal planes of all agents attend only to the ego planes, then collapses all planes into a single ego-frame volume and predicts occupancy from this volume alone.

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) Error rate by region (b) Many-to-one fusion (c) Many-to-many fusion

Figure 1: The collaborator region under M2O and M2M fusion. (a) Error rate of VOGS-CP (Chen et al., 2026) on Semantic-OPV2V, defined as the percentage of occupied ground-truth voxels that it misclassifies. Within the collaborator region, *≥1 correct* and *None correct* denote the voxels where at least one collaborator's single-agent estimate is correct and where none is, respectively. (b, c) A bus voxel is observed by both collaborators but not by the ego agent, whose representation of it therefore starts from a prior. (b) M2O fusion refines the ego representation (solid border) and leaves the collaborator representations uncorrected (dashed border), so the ego output inherits the error of Collab. 2 (car). (c) M2M fusion refines the representations of all agents with context from one another and corrects the representation of Collab. 2 to bus, so the ego output inherits this correction.

A characteristic weakness of M2O fusion appears in regions that the ego agent cannot see but one or more collaborators can. These regions are common and hold information that the ego agent can obtain only through collaboration. We refer to these regions as the collaborator region, and to the regions the ego agent observes as the ego region. VOGS-CP, a representative M2O method, misclassifies 33.1% of the collaborator region near the ego agent on Semantic-OPV2V (Song et al., 2024), four times its 8.1% error on the ego region at the same distances (Figure 1a). This weakness arises because the output there follows the collaborators (8.4% error where at least one collaborator's single-agent estimate is correct vs. 59.5% where none is). Collaborator representations, however, inevitably carry the errors of their single viewpoints. M2O fusion never draws on the context of other collaborators to correct these errors, which therefore pass directly into the ego grid. This weakness is thus structural: it stems from M2O fusion itself rather than from any particular method, and it also appears in CoHFF and VQSOP (Sec. 5.3).

Nothing in the task requires fusion to be M2O. Fusion can instead refine and correct the collaborator representations, using the context of all other agents. Such fusion is **M**any-**to**-**M**any (**M2M**), jointly refining the representations of all agents and deferring the reduction to a single ego grid until the output. However, M2M fusion does not apply directly to CSOP. This task predicts every voxel, so its methods operate on scene-level 3D representations rather than on one global vector (Liu et al., 2020b;a), a few object queries (Chen et al., 2023; Wang et al., 2025b;c), or a bird's-eye-view (BEV) map (Wang et al., 2020; Xu et al., 2022; 2023) per agent. M2M fusion over such representations is costly, and this cost may explain why existing methods for CSOP settle for M2O fusion. We thus seek M2M fusion over scene-level representations at an acceptable cost.

We therefore propose the Bridge Transformer, a feed-forward architecture that implements M2M fusion via alternating attention (Wang et al., 2025a) over all agents' representations (Figure 2). Each agent represents its scene as 3D semantic Gaussians (Kerbl et al., 2023; Huang et al., 2024; 2025) and shares them with the ego agent. Since full attention over all Gaussians of all agents would be prohibitively costly, each agent groups its Gaussians into one anchor token and K member tokens

per group. Inter-agent attention over the anchor tokens of all agents then alternates with intra-agent attention within each group, keeping M2M fusion at an acceptable cost. Because inter-agent attention is unmasked, every agent's anchor tokens serve as queries rather than passive keys and values, so refinement accumulates across collaborators as well as in the ego agent (Figure 1c). A dual-branch Gaussian decoder then decodes both token types into Gaussians, and only then does Gaussian-to-voxel splatting (Huang et al., 2024) reduce all agents' Gaussians into the ego grid.

Our contributions are as follows:

- We identify M2O fusion as a structural limitation of existing methods for CSOP. The task does not require it, yet these methods impose it. To address this, we introduce M2M fusion to this task. It jointly refines the representations of all agents and defers the reduction to the ego grid until the output.
- We propose the Bridge Transformer to realize M2M fusion. It tokenizes each agent's Gaussians into anchor and member tokens. Inter-agent attention over all agents' anchor tokens then alternates with intra-agent attention over each anchor and its members. A dual-branch Gaussian decoder then decodes member tokens into refined Gaussians and anchor tokens into new ones, which are splatted onto the ego grid.
- Extensive experiments show that the Bridge Transformer improves over the strongest prior method by +3.88 mIoU and +2.01 IoU. An ablation further shows that M2M fusion reduces the error rate in the collaborator region by 1.15 points relative to M2O fusion (Sec. 5.2).

2 Related Work

3D Semantic Occupancy Prediction. Vision-based 3D semantic occupancy prediction typically relies on grid-based scene representations. MonoScene (Cao & de Charette, 2022) lifts 2D features from a single RGB image into a dense 3D voxel grid, and SurroundOcc (Wei et al., 2023) builds multi-scale 3D volumes from surround-view images. TPVFormer (Huang et al., 2023) projects the scene onto three orthogonal planes, and OccFormer (Zhang et al., 2023) processes the 3D volume with a dual-path transformer. To avoid dense-voxel overhead, GaussianFormer (Huang et al., 2024) and GaussianFormer-2 (Huang et al., 2025) represent the scene with sparse 3D semantic Gaussians and splat them onto an occupancy grid. However, these single-agent models do not address how Gaussian sets predicted independently by collaborating agents should be fused.

Collaborative Perception. Collaborative perception supplements the ego agent's observations with spatially aligned representations from neighboring vehicles or infrastructure. In collaborative 3D object detection, V2X-ViT (Xu et al., 2022) models heterogeneous multi-agent interactions and relative spatial relationships. Its attention updates the BEV features of all agents, but only the ego feature is decoded. Where2comm (Hu et al., 2022) selects sparse BEV features with spatial confidence and request maps from the receiver, and CodeFilling (Hu et al., 2024) encodes complementary messages as indices into a shared codebook. In collaborative 3D semantic occupancy prediction, CoHFF (Song et al., 2024) shares sparsified orthogonal-plane features, and VQSOP (Li et al., 2026) compresses informative 3D regions with sparse-aware vector quantization. Since these approaches decouple message compression from the fusion operator, we instead focus on the fusion operator itself across multiple Gaussian sets.

3D Gaussians for Collaboration. Recent work adopts 3D semantic Gaussians as the shared representation. VOGS-CP (Chen et al., 2026) shares the Gaussians within a region of interest for vision-only collaborative 3D semantic occupancy prediction and aggregates them locally into the ego Gaussians. It establishes primitive selection, sharing, and local aggregation, but its fusion remains M2O, whereas we study M2M fusion over the Gaussian sets of all agents.

3 Methodology

3.1 Problem Setup

Collaborative semantic occupancy prediction. Let $\mathcal{A} = \{1, \ldots, A\}$ denote the agents within communication range at a given time step. Agent 1 is the ego agent, while the remaining agents

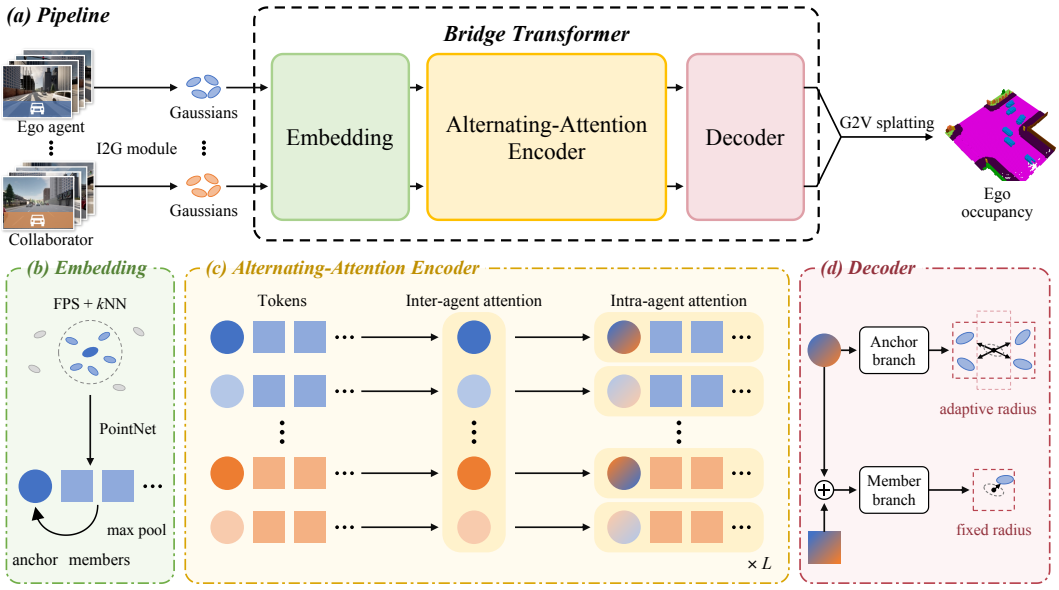


Figure 2: Overview of the Bridge Transformer. The image-to-Gaussian (I2G) module (Chen et al., 2026) converts each agent's images into 3D semantic Gaussians. The Bridge Transformer jointly refines the Gaussians of all agents, and Gaussian-to-voxel (G2V) splatting (Huang et al., 2024) then maps the Gaussians onto the ego grid. Blue and orange denote two agents, and blended colors indicate anchor tokens that incorporate information from both.

$i \in \mathcal{A} \setminus \{1\}$ are collaborators. Each agent i captures surround-view images $\mathcal{I}_i$ with known camera intrinsics and extrinsics. The rigid transformation $T_i = (\boldsymbol{R}_i, \boldsymbol{t}_i)$ from agent i's local frame to the ego frame is also known, with T_1 the identity. The objective is to predict semantic occupancy within a predefined region of interest (RoI) Ω in the ego frame. This region is discretized into a voxel grid $\mathcal{V}$, where each voxel takes a label from $\mathcal{C}$, comprising C_s semantic classes and free space. Following the collaborative perception paradigm (Hu et al., 2022; Song et al., 2024), each agent first maps its images to an intermediate representation via a shared network ψ_θ. Collaborators then transmit their representations to the ego agent, which fuses all representations using a collaborative predictor Φ_θ:

$$\hat{\mathbf{Y}} = \Phi_\theta\big(\{(\psi_\theta(\mathcal{I}_i), T_i)\}_{i\in\mathcal{A}}\big) \in \mathcal{C}^{\mathcal{V}}. \tag{1}$$

Equation 1 is M2O in its *output*: multi-agent observations are unified into a single ego-frame grid. The *computation* inside Φ_θ, however, need not be ego-centered, and Sec. 3.3 makes it M2M.

Semantic Gaussians as the shared representation. To limit communication bandwidth, agents share a compact intermediate representation rather than the raw images. 3D semantic Gaussians provide a favorable trade-off because their communication payload scales with the number of primitives rather than the spatial scene volume (Chen et al., 2026). Accordingly, for each agent i, the image-to-Gaussian (I2G) module of VOGS-CP (Chen et al., 2026) maps raw images $\mathcal{I}_i$ to a set of P 3D semantic Gaussians $\mathcal{G}_i = \{\boldsymbol{g}_{i,p}\}_{p=1}^{P}$. Each Gaussian $\boldsymbol{g}_{i,p}$ is parameterized by its mean $\boldsymbol{\mu}_{i,p} \in \mathbb{R}^3$, scale $\boldsymbol{s} \in \mathbb{R}^3_{>0}$, unit rotation quaternion $\boldsymbol{q} \in \mathbb{S}^3$, opacity $\alpha \in (0,1)$, and semantics $\boldsymbol{c} \in \mathbb{R}^{|\mathcal{C}|}$. Before fusion, we transform collaborator Gaussians into the ego frame. Among the five attributes, only the mean and the quaternion vary under this rigid transformation. We therefore apply T_i to these two attributes: $\boldsymbol{\mu} \mapsto \boldsymbol{R}_i\boldsymbol{\mu} + \boldsymbol{t}_i$ and $\boldsymbol{q} \mapsto \boldsymbol{q}_{\boldsymbol{R}_i} \otimes \boldsymbol{q}$. Here, $\boldsymbol{q}_{\boldsymbol{R}_i}$ is the unit quaternion corresponding to $\boldsymbol{R}_i$, and $\otimes$ denotes quaternion multiplication.

3.2 Hierarchical Gaussian Tokenization

Grouping and embedding. For each agent i, we apply farthest point sampling (FPS) over the Gaussian centers $\{\boldsymbol{\mu}_{i,p}\}_{p=1}^{P}$ in $\mathcal{G}_i$ to select N anchor Gaussians. For each anchor, a k-nearest-neighbor (kNN) search then selects the K nearest Gaussians as its member Gaussians (Yu et al.,

2022). Unlike the standard PointNet, which produces either point-wise or globally pooled representations (Qi et al., 2017; Yu et al., 2022), our adapted PointNet preserves both. It projects each member Gaussian into a member token $\boldsymbol{m}_{i,n,k} \in \mathbb{R}^D$ and applies channel-wise max pooling over the K member tokens to produce the corresponding anchor token $\boldsymbol{a}_{i,n} \in \mathbb{R}^D$.

Positional encoding. Each anchor or member token takes the ego-frame mean of its Gaussian, $\boldsymbol{\mu}_{i,n}$ or $\boldsymbol{\mu}_{i,n,k}$, as its position. The Fourier bands expect inputs in $[-1, 1]$; raw metric coordinates span tens of meters and would make every band oscillate too rapidly to preserve locality. We therefore first normalize this position so that the RoI Ω maps onto $[-1, 1]^3$. We then encode each normalized coordinate with F octave-spaced Fourier bands (Mildenhall et al., 2020), concatenate the three encodings, and project the result to $\mathbb{R}^D$ with an MLP. Because anchors are compared scene-wide while members are compared only locally, they favor different frequency bands. Anchor and member tokens accordingly use separate encoders of this form.

3.3 Alternating-Attention Encoder

The encoder refines the tokens of all agents across L rounds of alternating attention (Wang et al., 2025a). Each round $\ell \in \{1, \dots, L\}$ comprises an inter-agent attention block $\mathrm{Block}^{(\ell)}_{\mathrm{inter}}(\cdot)$ followed by an intra-agent attention block $\mathrm{Block}^{(\ell)}_{\mathrm{intra}}(\cdot)$. Both are standard pre-norm Transformer blocks (Xiong et al., 2020) with multi-head self-attention.

Inter-agent attention. Inter-agent attention concatenates the anchor tokens of all A agents into a unified sequence in $\mathbb{R}^{AN \times D}$ and updates them jointly into intermediate anchor tokens $\boldsymbol{a}'^{(\ell)}_{i,n}$:

$$[\boldsymbol{a}'^{(\ell)}_{1,1}; \dots; \boldsymbol{a}'^{(\ell)}_{A,N}] = \mathrm{Block}^{(\ell)}_{\mathrm{inter}}\left([\boldsymbol{a}^{(\ell-1)}_{1,1}; \dots; \boldsymbol{a}^{(\ell-1)}_{A,N}]\right). \tag{2}$$

Because no attention mask is imposed across agents, the anchor tokens of each agent are refined at the scene level with context from all agents.

Intra-agent attention. Intra-agent attention concatenates each intermediate anchor token $\boldsymbol{a}'^{(\ell)}_{i,n}$ with its K member tokens into a separate sequence in $\mathbb{R}^{(K+1) \times D}$ and updates them jointly:

$$[\boldsymbol{a}^{(\ell)}_{i,n}; \boldsymbol{m}^{(\ell)}_{i,n,1}; \dots; \boldsymbol{m}^{(\ell)}_{i,n,K}] = \mathrm{Block}^{(\ell)}_{\mathrm{intra}}\left([\boldsymbol{a}'^{(\ell)}_{i,n}; \boldsymbol{m}^{(\ell-1)}_{i,n,1}; \dots; \boldsymbol{m}^{(\ell-1)}_{i,n,K}]\right). \tag{3}$$

Intra-agent attention thereby refines fine-grained geometric and semantic details within each local group. Since the anchor token has just been updated by inter-agent attention, it also propagates information from the other agents to its member tokens.

Complexity. In each round, inter-agent attention operates on AN anchor tokens, and intra-agent attention on AN groups of $K + 1$ tokens. Including token-wise projections and MLPs, a round costs $\mathcal{O}(A^2N^2D + ANK^2D + ANKD^2)$, whereas full attention over all AP Gaussians would cost $\mathcal{O}(A^2P^2D)$ for attention alone.

3.4 Dual-Branch Gaussian Decoder

The Gaussian decoder applies an anchor branch f_{anc} and a member branch f_{mem} to the tokens of every agent. The member branch refines each member Gaussian, following the refinement module of GaussianFormer (Huang et al., 2024). Prior Gaussian-based fusion (Chen et al., 2026) only refines the ego Gaussians. The anchor branch additionally generates new Gaussians from the anchor tokens, which carry context from all agents. The decoded Gaussians can therefore also represent scene content that no agent's predicted Gaussians capture.

Anchor branch. The anchor branch f_{anc} takes the anchor token $\boldsymbol{a}_{i,n}$ and its mean $\boldsymbol{\mu}_{i,n}$ as input and predicts J new Gaussians along with a shared radius logit $\boldsymbol{z}_{i,n} \in \mathbb{R}^3$:

$$\left(\{\hat{\boldsymbol{g}}_{i,n,j}\}_{j=1}^{J}, \boldsymbol{z}_{i,n}\right) = f_{\mathrm{anc}}(\boldsymbol{a}_{i,n}, \boldsymbol{\mu}_{i,n}). \tag{4}$$

The mean of each new Gaussian is offset from the anchor mean by at most an adaptive radius $\boldsymbol{r}_{i,n}$:

$$\hat{\boldsymbol{\mu}}_{i,n,j} = \boldsymbol{\mu}_{i,n} + \boldsymbol{r}_{i,n} \odot \boldsymbol{d}_{i,n,j}, \qquad \boldsymbol{r}_{i,n} = \boldsymbol{r}_{\mathrm{anc}} \odot \sigma(\boldsymbol{z}_{i,n}), \tag{5}$$

where the normalized offset $\boldsymbol{d}_{i,n,j} \in (-1,1)^3$ is a scaled sigmoid of the MLP output, $\boldsymbol{r}_{\text{anc}} = (4,4,0.8)\,\text{m}$ bounds the radius, $\sigma(\cdot)$ is the element-wise sigmoid, and $\odot$ is the Hadamard product.

Member branch. The member branch f_{mem} takes the summed token $\boldsymbol{m}_{i,n,k} + \boldsymbol{a}_{i,n}$ and the member mean $\boldsymbol{\mu}_{i,n,k}$ as input and predicts a single refined Gaussian:

$$\hat{\boldsymbol{g}}_{i,n,k} = f_{\text{mem}}(\boldsymbol{m}_{i,n,k} + \boldsymbol{a}_{i,n}, \boldsymbol{\mu}_{i,n,k}). \tag{6}$$

Its mean is offset from the member mean by at most a fixed radius:

$$\hat{\boldsymbol{\mu}}_{i,n,k} = \boldsymbol{\mu}_{i,n,k} + \boldsymbol{r}_{\text{mem}} \odot \boldsymbol{d}_{i,n,k}, \tag{7}$$

where $\boldsymbol{d}_{i,n,k}$ is defined as in equation 5, and $\boldsymbol{r}_{\text{mem}} = (2,2,0.5)\,\text{m}$ is tighter than $\boldsymbol{r}_{\text{anc}}$. Gaussian-to-voxel splatting (Huang et al., 2024) then maps all decoded Gaussians onto the ego grid.

We train the I2G module ψ_θ and the Bridge Transformer Φ_θ end to end with the class-weighted cross-entropy loss, the Lovász-softmax loss (Berman et al., 2018), and the geometric and semantic scene-class affinity losses of MonoScene (Cao & de Charette, 2022), as detailed in Appendix A.

4 Experiments

Dataset and baselines. We evaluate on Semantic-OPV2V (Song et al., 2024). Each frame contains two to seven agents, and we let the ego agent collaborate with every agent within 70 m. Every agent carries four surround-view cameras, the only input to all methods. The region of interest spans $[-20, 20]$ m longitudinally and laterally and $[-2.3, 0.9]$ m vertically around the ego agent, voxelized at 0.4 m into a $100 \times 100 \times 8$ grid. Following CoHFF (Song et al., 2024) and VOGS-CP (Chen et al., 2026), we aggregate the semantic-LiDAR scans of all agents in the ego frame to obtain the collaborative ground truth, using the 4LidarSurround variant with $C_s = 12$ semantic classes plus free space. We train on the OPV2V training split (6,374 frames) and report on the validation split (1,980 frames). We compare with the three collaborative semantic occupancy methods evaluated on Semantic-OPV2V, comprising CoHFF (Song et al., 2024), VQSOP (Li et al., 2026), and VOGS-CP (Chen et al., 2026). We also compare with Ours w/ mask, a variant with the same architecture and parameter count whose inter-agent attention is restricted so that only the ego agent attends to collaborators. This restriction reduces fusion to M2O.

Implementation details. The I2G module adopts the architecture and hyperparameters of VOGS-CP (Chen et al., 2026) and produces $P = 25{,}600$ Gaussians per agent. The tokenizer selects $N = 512$ anchors with $K = 384$ members each, and the Fourier positional encoding uses $F = 6$ frequencies. The encoder has $L = 6$ rounds with width $D = 192$, 6 attention heads, and an MLP ratio of 4. The anchor branch of the Gaussian decoder produces $J = 3$ Gaussians per anchor. We train the model end to end for 50 epochs on 8 NVIDIA A800 GPUs. Appendix A provides the remaining implementation details.

Evaluation metrics. Following prior work (Song et al., 2024; Chen et al., 2026), we report intersection over union (IoU) and mean IoU (mIoU) for semantic occupancy prediction. For the downstream BEV segmentation task, we report 2D IoU, collapsing the predicted and ground-truth voxels along the height axis into a BEV map and scoring vehicles, roads, and other general objects. We further report communication volume, the average size of the message a collaborator transmits per frame, and inference latency, the average time of one forward pass over a frame. Model size is the number of parameters loaded in that pass, with those shared across agents counted once.

5 Results

5.1 Main Results

3D semantic occupancy prediction. We study cross-agent fusion, so we evaluate only under the collaborative perception setting. Table 1 presents the quantitative comparison on Semantic-OPV2V. Unlike the M2O methods CoHFF, VQSOP, and VOGS-CP, our Bridge Transformer performs M2M fusion and achieves the best IoU and mIoU. It surpasses the strongest baseline, VQSOP, by 2.01 in

Table 1: 3D semantic occupancy prediction results on Semantic-OPV2V under the collaborative perception setting. * denotes results reported in the original papers. The best and second-best results are highlighted in **bold** and underlined, respectively.

Method	CoHFF*	VOGS-CP*	VQSOP*	Ours w/ mask	**Ours**
Fusion	M2O	M2O	M2O	M2O	M2M
IoU	50.46	72.87	73.79	74.70	**75.80**
mIoU	34.16	37.44	41.54	44.24	**45.42**
Building	**25.72**	9.61	11.22	8.37	9.05
Fence	27.83	29.20	33.19	33.51	**34.62**
Terrain	48.30	74.51	63.19	79.04	**80.79**
Pole	**42.74**	12.19	20.16	19.63	23.59
Road	61.77	83.05	**87.06**	82.76	84.72
Sidewalk	39.62	**78.22**	67.40	77.82	78.11
Vegetation	20.59	20.43	26.78	**28.87**	28.65
Vehicle	63.28	60.49	63.36	67.76	**68.74**
Wall	**58.27**	36.45	36.51	39.45	40.23
Guard rail	1.94	32.50	**54.20**	49.44	53.34
Traffic sign	**16.33**	8.26	14.02	14.78	15.90
Bridge	3.53	4.35	21.37	**29.49**	27.31

Table 2: BEV semantic segmentation results on Semantic-OPV2V in 2D IoU with 2 agents and with up to 7 agents. * denotes results reported in the original papers, and – denotes an unavailable result.

	2 agents			Up to 7 agents		
Method	Vehicle	Road	Others	Vehicle	Road	Others
CoBEVT*	46.13	52.41	–	60.40	63.00	–
CoHFF*	47.40	63.36	40.27	64.44	57.28	45.89
VOGS-CP*	70.25	82.69	79.37	75.30	84.96	80.19
VQSOP*	73.23	85.05	79.42	77.48	87.04	**81.10**
Ours	**80.42**	**85.31**	**80.49**	**81.15**	**87.20**	81.04

IoU and 3.88 in mIoU. We retain the I2G module of VOGS-CP and replace its fusion operator, yet the margins over VOGS-CP widen further to 2.93 in IoU and 7.98 in mIoU. This indicates that the gain comes from how collaborator representations are fused rather than from how they are produced. To separate the fusion paradigm from the architecture and the parameter count, we compare Ours with Ours w/ mask. The mask lowers IoU by 1.10 and mIoU by 1.18, a drop that quantifies the contribution of the M2M fusion paradigm itself. Our method also achieves the best or second-best result on nearly all semantic classes. The class-wise results show a strength on small, thin, and geometrically complex objects, and this strength extends to large ones. Our method improves over VQSOP by 3.43 on Pole and 1.88 on Traffic sign, and by 5.38 on Vehicle. These gains on both small and large objects may stem from the alternating-attention encoder, which combines fine-grained refinement within local groups with scene-level context.

BEV semantic segmentation. Table 2 compares all methods under two collaborative settings, and our method achieves the best result in five of the six columns. The gain is concentrated on the Vehicle class, where our method outperforms the strongest baseline, VQSOP, by 7.19 IoU with two agents and by 3.67 with up to seven agents. On the Road and Others classes, our method is on par with VQSOP, differing by at most 1.07 IoU. This contrast likely arises because a slight positional shift sharply lowers the IoU of a compact vehicle but barely affects that of a large road. M2M fusion may correct such shifts by jointly refining the representations of all agents.

Efficiency. Table 3 compares the efficiency of all methods, with latency measured on a single NVIDIA RTX 4090 GPU. VQSOP, designed for communication efficiency, leads on all three metrics

Table 3: Efficiency comparison on Semantic-OPV2V. CV is the communication volume, and latency is reported as mean ± standard deviation. * denotes values reported in the original papers.

Method	CV (MB) ↓	Params (M) ↓	Latency (ms) ↓
CoHFF	0.78*	573.1	965.8 ± 169.4
VOGS-CP	1.07*	75.7	378.7 ± 207.6
VQSOP	0.013*	63.7	196.9 ± 90.9
Ours	2.46	82.3	559.2 ± 252.0

Table 4: Error rates of Ours w/ mask (M2O fusion) and Ours (M2M fusion) on Semantic-OPV2V in the regions of Figure 1a.

	Error rate (%) ↓	
Region	Ours w/ mask	**Ours**
Ego region	7.51	**7.50**
Collaborator region	16.08	**14.93**
≥1 correct	**4.45**	4.58
None correct	28.63	**26.20**

Table 5: Architecture ablation on Semantic-OPV2V. The first row is our full model, and each row below it changes one design choice. Blank cells repeat the value in the first row. In each round, *Inter-first* and *Intra-first* place the inter-agent block and the intra-agent block first, respectively, and *Intra-only* replaces the inter-agent block with an intra-agent block. *Masked* is Ours w/ mask.

Attention	N	K	Pooling	mIoU	IoU
Inter-first	512	384	Max	45.42	75.80
Intra-only				44.88	74.66
Masked				44.24	74.70
Intra-first				44.51	74.46
	256			44.11	74.78
	768			44.46	75.42
		256		45.07	75.81
		512		45.11	76.00
			Attn	44.64	74.90

but trails our method by 3.88 mIoU (Table 1). Built on VOGS-CP, our method improves mIoU by 7.98 while adding 180.5 ms of latency and only 6.6 M parameters. Compared with CoHFF, which runs two task-specific networks, our single network has about one-seventh the parameters and 42% lower latency. Our method has the largest communication volume because each collaborator transmits all its Gaussians, whereas VOGS-CP transmits only those inside the ego agent's RoI. This volume nevertheless remains on the same order of magnitude as those of CoHFF and VOGS-CP, while our accuracy gains over both are substantial.

5.2 Ablation Study

Does M2M fusion address the weakness of M2O fusion? To answer this question, we test whether replacing M2O fusion with M2M fusion (i) reduces the error rate in the collaborator region, (ii) reduces it particularly where no collaborator's single-agent estimate is correct, and (iii) leaves the error rate in the ego region largely unchanged. Table 4 compares Ours w/ mask and Ours, which differ only in the fusion paradigm. All three hold. In the collaborator region, M2M fusion reduces the error rate by 1.15 points. The reduction grows to 2.43 points on the *None correct* voxels, whereas the two variants perform comparably on the *≥1 correct* voxels. In the ego region, they differ by only 0.01 points. These results indicate that M2M fusion mitigates the weakness of M2O fusion by correcting collaborator errors rather than by improving predictions uniformly.

Attention in the encoder. *Intra-only* lowers mIoU by 0.54 and IoU by 1.14 because it lacks cross-agent interaction. *Masked* lowers mIoU by 1.18 and IoU by 1.10 because its M2O fusion refines only the ego agent's tokens with cross-agent context. *Intra-first* also lowers mIoU by 0.91 and IoU by 1.34, showing the advantage of the global-to-local order. Overall, the full model outperforms all three variants in both metrics, which supports our encoder design.

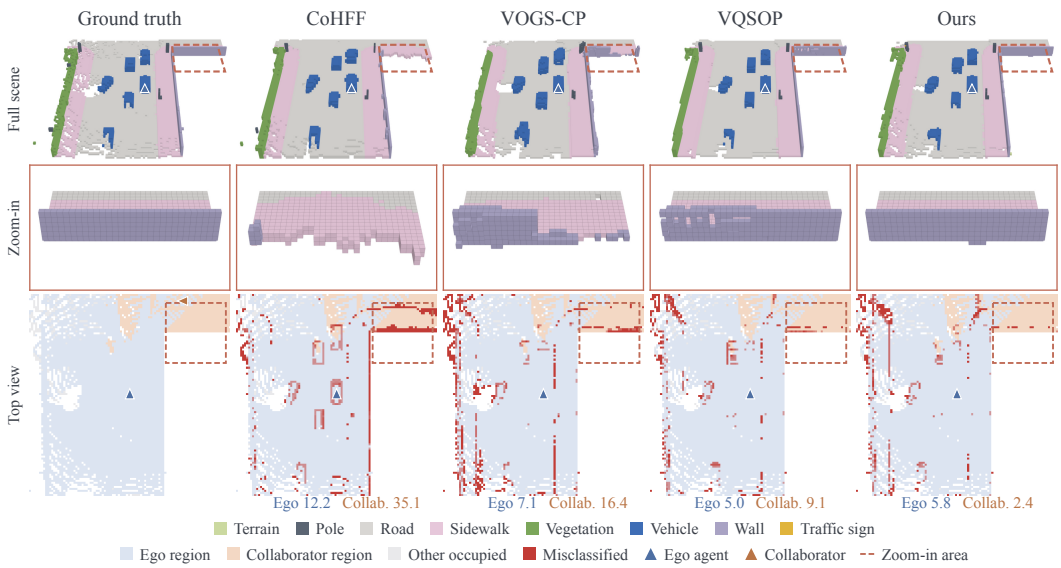


Figure 3: Qualitative comparison on the Semantic-OPV2V validation set. The rows show the full scene, a zoom-in on the dashed box in the collaborator region, and a top view. In the top view, red denotes the occupied ground-truth voxels that each method misclassifies. Below each method column are its error rates (%) in the ego and collaborator regions, respectively.

Numbers of anchors and members. Setting N to 256 or 768 lowers mIoU by 1.31 and 0.96, respectively, so performance peaks at $N = 512$. Fewer anchors cover the scene too sparsely for inter-agent attention, whereas more anchors likely introduce redundant tokens from overlapping fixed-size groups. By contrast, setting K to 256 or 512 changes mIoU by at most 0.35 and IoU by at most 0.20, since K affects only the group size, not the number of tokens in inter-agent attention.

Deriving anchor tokens from member tokens. Replacing max pooling with attention pooling (Lee et al., 2019) lowers mIoU by 0.78 and IoU by 0.90. Max pooling likely yields better anchor tokens because it keeps the strongest response of each channel among the K member tokens, whereas attention pooling averages them, diluting features carried by only a few members.

5.3 Visualization

Figure 3 presents qualitative results of all methods on a frame from the Semantic-OPV2V validation set. Despite their different representations, all three M2O baselines have higher error rates in the collaborator region than in the ego region, as VOGS-CP does in Figure 1a. The zoom-in shows how such errors distort objects. CoHFF misclassifies most of a wall as sidewalk, VOGS-CP recovers only its left half, and VQSOP leaves gaps in it. Our method recovers the whole wall. In this frame, it also has the lowest error rate in the collaborator region, whereas its error rate in the ego region remains close to those of VOGS-CP and VQSOP. This pattern is consistent with Table 4, which varies only the fusion paradigm. Additional qualitative results in Appendix B exhibit the same pattern.

6 Conclusion

CSOP requires M2O only of its output, not of its fusion. Existing methods nonetheless impose M2O on fusion, so collaborator errors reach the ego grid uncorrected. The Bridge Transformer instead performs M2M fusion, jointly refining the representations of all agents and reducing them to the ego grid only at the output. An ablation shows that M2M fusion lowers the error rate in the collaborator region while leaving that in the ego region largely unchanged, indicating that it reduces the collaborator errors that M2O fusion leaves uncorrected. Our method, however, has the largest communication volume and higher latency than VQSOP and VOGS-CP, and is evaluated on a single simulated benchmark. Message compression and validation on real data are left to future work.

AI USE STATEMENT

In this work, we used generative AI tools for implementing methods. We have not used generative AI tools for generating synthetic data sets, helping develop theoretical models or conceptual frameworks, proposing or refining hypotheses, designing or providing feedback on research methodology or experiments, cleaning and reformatting data sets, supporting qualitative and thematic data analysis, interpreting results, or assisting with translation, and formulating mathematical claims, providing critical ingredients for proving them, and assisting in the writing of proofs are not applicable to this work, which contains no theoretical results. Additionally, we used generative AI tools for creating or editing software code, editing this paper to improve readability, and formatting references. We have reviewed all AI-assisted work. AI coding assistants wrote parts of the model, training, and evaluation code from the authors' specifications, and the authors reviewed this code line by line before merging it. The many-to-many fusion paradigm, the Bridge Transformer architecture, and the design of every experiment were conceived by the authors. Language assistance was limited to polishing the authors' English and suggesting revisions to the wording of the paper. The authors ran all training and evaluation themselves, and every number in this paper, except the baseline results taken from the original papers, comes from these runs. The authors also checked the technical descriptions and hyperparameters in the paper against the code and configuration files. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

All experiments use Semantic-OPV2V (Song et al., 2024), a public benchmark simulated in CARLA (Dosovitskiy et al., 2017). It contains no human subjects, real-world recordings, or personally identifiable information. We use the publicly released benchmark for academic research only. Collaborative perception targets autonomous driving, a safety-critical application. Our results are obtained in simulation under ideal communication and accurate agent poses, so deployment on real vehicles would require validation under pose error, communication latency, faulty and adversarial collaborators, and the presence of vulnerable road users. Sharing semantic Gaussians rather than raw images limits the visual information that leaves each vehicle.

REPRODUCIBILITY STATEMENT

Section 3 specifies the Bridge Transformer, including the tokenizer, the alternating attention encoder, and the Gaussian decoder. Section 4 describes the public dataset and its splits, the construction of the ground truth, the architecture hyperparameters, the hardware, and the definition of every metric, and Appendix A gives the training objective and the remaining implementation details, including the optimization hyperparameters. Section 4 also identifies the components that follow the public implementation of VOGS-CP (Chen et al., 2026), and Table 1 marks the results from the original papers. We will release the code, configuration files, and trained weights upon acceptance.

REFERENCES

Maxim Berman, Amal Rannen Triki, and Matthew B. Blaschko. The Lovász-Softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4413–4421, 2018.

Anh-Quan Cao and Raoul de Charette. MonoScene: Monocular 3D semantic scene completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3991–4001, 2022.

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, 2026.

Ziming Chen, Yifeng Shi, and Jinrang Jia. TransIFF: An instance-level feature fusion framework for vehicle-infrastructure cooperative 3D detection with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 18205–18214, 2023.

Mostafa Dehghani, Josip Djolonga, Basil Mustafa, Piotr Padlewski, Jonathan Heek, Justin Gilmer, Andreas Steiner, Mathilde Caron, Robert Geirhos, Ibrahim Alabdulmohsin, Rodolphe Jenatton, Lucas Beyer, Michael Tschannen, Anurag Arnab, Xiao Wang, Carlos Riquelme, Matthias Minderer, Joan Puigcerver, Utku Evci, Manoj Kumar, Sjoerd van Steenkiste, Gamaleldin F. Elsayed, Aravindh Mahendran, Fisher Yu, Avital Oliver, Fantine Huot, Jasmijn Bastings, Mark Patrick Collier, Alexey Gritsenko, Vighnesh Birodkar, Cristina Vasconcelos, Yi Tay, Thomas Mensink, Alexander Kolesnikov, Filip Pavetić, Dustin Tran, Thomas Kipf, Mario Lučić, Xiaohua Zhai, Daniel Keysers, Jeremiah Harmsen, and Neil Houlsby. Scaling vision transformers to 22 billion parameters. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 7480–7512. PMLR, 2023.

Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. CARLA: An open urban driving simulator. In *Proceedings of the 1st Annual Conference on Robot Learning*, volume 78 of *Proceedings of Machine Learning Research*, pp. 1–16. PMLR, 2017.

Yue Hu, Shaoheng Fang, Zixing Lei, Yiqi Zhong, and Siheng Chen. Where2comm: Communication-efficient collaborative perception via spatial confidence maps. In *Advances in Neural Information Processing Systems*, volume 35, pp. 4874–4886, 2022.

Yue Hu, Juntong Peng, Sifei Liu, Junhao Ge, Si Liu, and Siheng Chen. Communication-efficient collaborative perception via information filling with codebook. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15481–15490, 2024.

Gao Huang, Yu Sun, Zhuang Liu, Daniel Sedra, and Kilian Q. Weinberger. Deep networks with stochastic depth. In *European Conference on Computer Vision*, volume 9908 of *Lecture Notes in Computer Science*, pp. 646–661. Springer, 2016.

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*, pp. 9223–9232, 2023.

Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. GaussianFormer: Scene as Gaussians for vision-based 3D semantic occupancy prediction. In *European Conference on Computer Vision*, volume 15085 of *Lecture Notes in Computer Science*, pp. 376–393. Springer, 2024.

Yuanhui Huang, Amonnut Thammatadatrakoon, Wenzhao Zheng, Yunpeng Zhang, Dalong Du, and Jiwen Lu. GaussianFormer-2: Probabilistic Gaussian superposition for efficient 3D occupancy prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 27477–27486, 2025.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D Gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4):139:1–139:14, 2023.

Juho Lee, Yoonho Lee, Jungtaek Kim, Adam R. Kosiorek, Seungjin Choi, and Yee Whye Teh. Set Transformer: A framework for attention-based permutation-invariant neural networks. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 3744–3753. PMLR, 2019.

Feng Li, Chaokun Zhang, and Gong Chen. Sparse-aware vector quantization for bandwidth-efficient collaborative 3D semantic occupancy prediction. In *European Conference on Computer Vision*, volume 17029 of *Lecture Notes in Computer Science*, pp. 172–189. Springer, 2026.

Yen-Cheng Liu, Junjiao Tian, Nathaniel Glaser, and Zsolt Kira. When2com: Multi-agent perception via communication graph grouping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4106–4115, 2020a.

Yen-Cheng Liu, Junjiao Tian, Chih-Yao Ma, Nathan Glaser, Chia-Wen Kuo, and Zsolt Kira. Who2com: Collaborative perception via learnable handshake communication. In *IEEE International Conference on Robotics and Automation*, pp. 6876–6883, 2020b.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. In *European Conference on Computer Vision*, volume 12346 of *Lecture Notes in Computer Science*, pp. 405–421. Springer, 2020.

Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. PointNet: Deep learning on point sets for 3D classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 652–660, 2017.

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*, pp. 17996–18006, 2024.

Hugo Touvron, Matthieu Cord, Alexandre Sablayrolles, Gabriel Synnaeve, and Hervé Jégou. Going deeper with image transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 32–42, 2021.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. VGGT: Visual geometry grounded transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5294–5306, 2025a.

Rujia Wang, Xiangbo Gao, Hao Xiang, Runsheng Xu, and Zhengzhong Tu. CoCMT: Communication-efficient cross-modal transformer for collaborative perception. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2471–2478, 2025b.

Shaohong Wang, Bin Lu, Xinyu Xiao, Hanzhi Zhong, Bowen Pang, Tong Wang, Zhiyu Xiang, Hangguan Shan, and Eryun Liu. RayFusion: Ray fusion enhanced collaborative visual perception. In *Advances in Neural Information Processing Systems*, volume 38, pp. 92357–92378, 2025c.

Tsun-Hsuan Wang, Sivabalan Manivasagam, Ming Liang, Bin Yang, Wenyuan Zeng, and Raquel Urtasun. V2VNet: Vehicle-to-vehicle communication for joint perception and prediction. In *European Conference on Computer Vision*, volume 12347 of *Lecture Notes in Computer Science*, pp. 605–621. Springer, 2020.

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*, pp. 21729–21740, 2023.

Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tie-Yan Liu. On layer normalization in the transformer architecture. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 10524–10533. PMLR, 2020.

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*, volume 13699 of *Lecture Notes in Computer Science*, pp. 107–124. Springer, 2022.

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 *Proceedings of the 6th Conference on Robot Learning*, volume 205 of *Proceedings of Machine Learning Research*, pp. 989–1000. PMLR, 2023.

Xumin Yu, Lulu Tang, Yongming Rao, Tiejun Huang, Jie Zhou, and Jiwen Lu. Point-BERT: Pre-training 3D point cloud transformers with masked point modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19313–19322, 2022.

Yunpeng Zhang, Zheng Zhu, and Dalong Du. OccFormer: Dual-path transformer for vision-based 3D semantic occupancy prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9433–9443, 2023.

A Implementation Details

Encoder. Each of the $2L = 12$ blocks in the alternating-attention encoder (Sec. 3.3) uses QK normalization (Dehghani et al., 2023), LayerScale (Touvron et al., 2021) initialized at 0.01, and stochastic depth (Huang et al., 2016) with a drop rate of 0.1.

Loss function. We train the I2G module ψ_θ and the Bridge Transformer Φ_θ end to end by minimizing

$$\mathcal{L} = \lambda_{\text{ce}}\mathcal{L}_{\text{CE}} + \lambda_{\text{lov}}\mathcal{L}_{\text{Lov}} + \lambda_{\text{geo}}\mathcal{L}_{\text{geo}} + \lambda_{\text{sem}}\mathcal{L}_{\text{sem}}, \tag{8}$$

where $\mathcal{L}_{\text{CE}}$ is the class-weighted cross-entropy loss and $\mathcal{L}_{\text{Lov}}$ is the Lovász-softmax loss (Berman et al., 2018), a smooth surrogate for the mean Intersection-over-Union (mIoU) across the C_s semantic classes. The remaining two terms, $\mathcal{L}_{\text{geo}}$ and $\mathcal{L}_{\text{sem}}$, are the geometric and semantic scene-class affinity losses of MonoScene (Cao & de Charette, 2022). They optimize the soft precision, recall, and specificity of occupancy and of each class. Following standard practice, we set the loss weights to $\lambda_{\text{ce}} = 8$, $\lambda_{\text{lov}} = 1.5$, and $\lambda_{\text{geo}} = \lambda_{\text{sem}} = 1$.

Optimization. We use AdamW (Loshchilov & Hutter, 2019) with an effective batch size of 16. The learning rate is 3.75×10^{-4} with a weight decay of 0.05, except for the I2G module, which uses 2.5×10^{-4} with a weight decay of 0.01. Both follow the same schedule: linear warm-up over the first 2,500 iterations, then cosine decay to $0.1\times$ the initial value.

B Additional Visualization

Figures 4 and 5 present additional qualitative results. In Figure 4a, CoHFF fragments three vehicles, and VOGS-CP merges them into a single object. Our method separates the three vehicles. In Figure 4b, VOGS-CP merges parked vehicles into a single object, and VQSOP leaves gaps in one vehicle. Our method separates the vehicles, although it introduces a small spurious one. In Figure 4c, CoHFF misses most of both vehicles, and VOGS-CP distorts them. VQSOP and our method recover both. In Figure 5a, VOGS-CP merges two guard rails and misses a vehicle, and VQSOP leaves gaps in the guard rails. Our method recovers both guard rails and the vehicle. In Figure 5b, VOGS-CP fragments a fence, and VQSOP spreads it over the terrain. Our method recovers it as a thin, continuous line. In Figure 5c, VOGS-CP misses most of a building and the vegetation along it. VQSOP and our method recover both. As in Figure 3, every baseline has a higher error rate in the collaborator region than in the ego region, and our method has the lowest in the collaborator region.

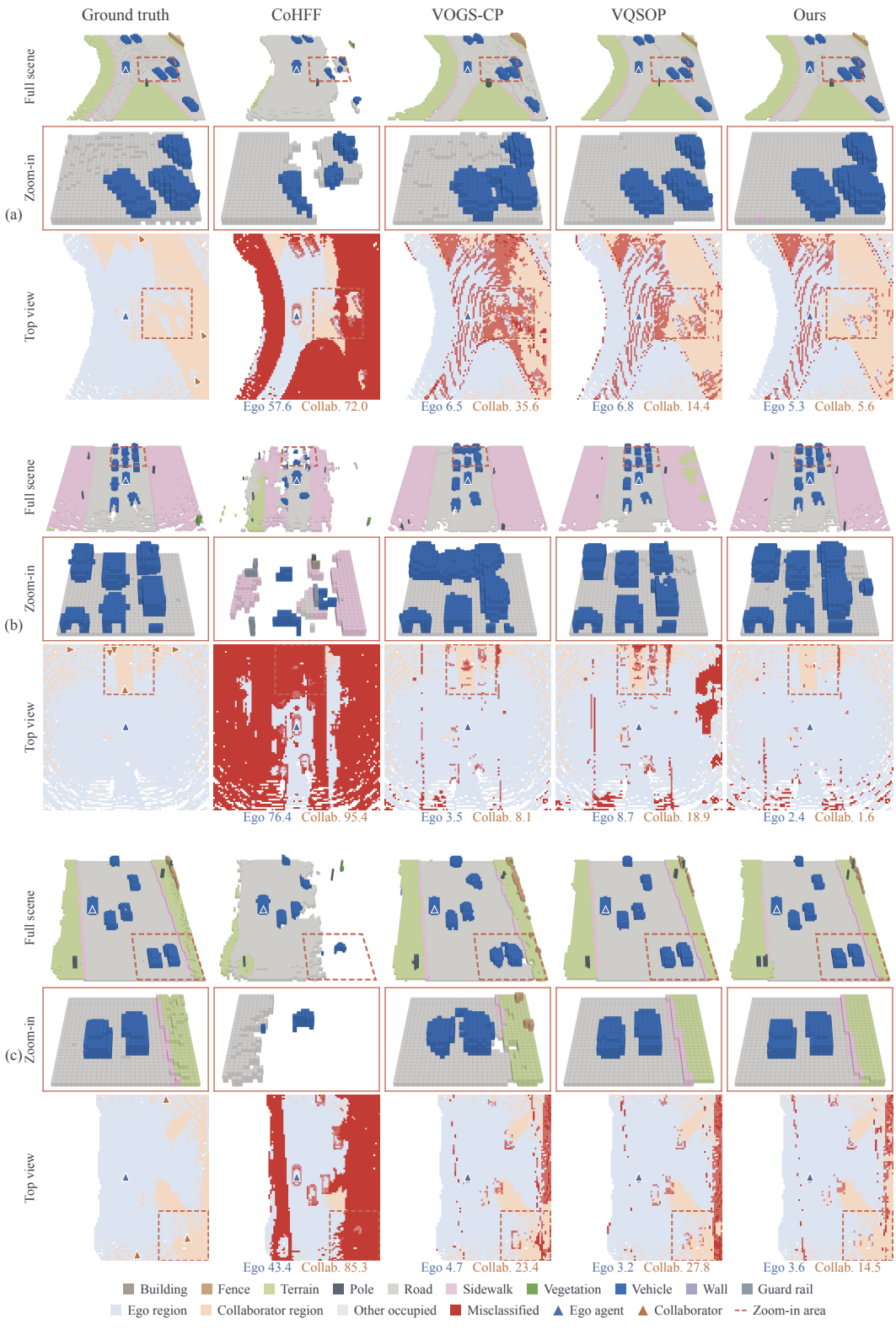


Figure 4: Additional qualitative comparison on the Semantic-OPV2V validation set, with the same layout as Figure 3. The zoom-ins focus on vehicles.

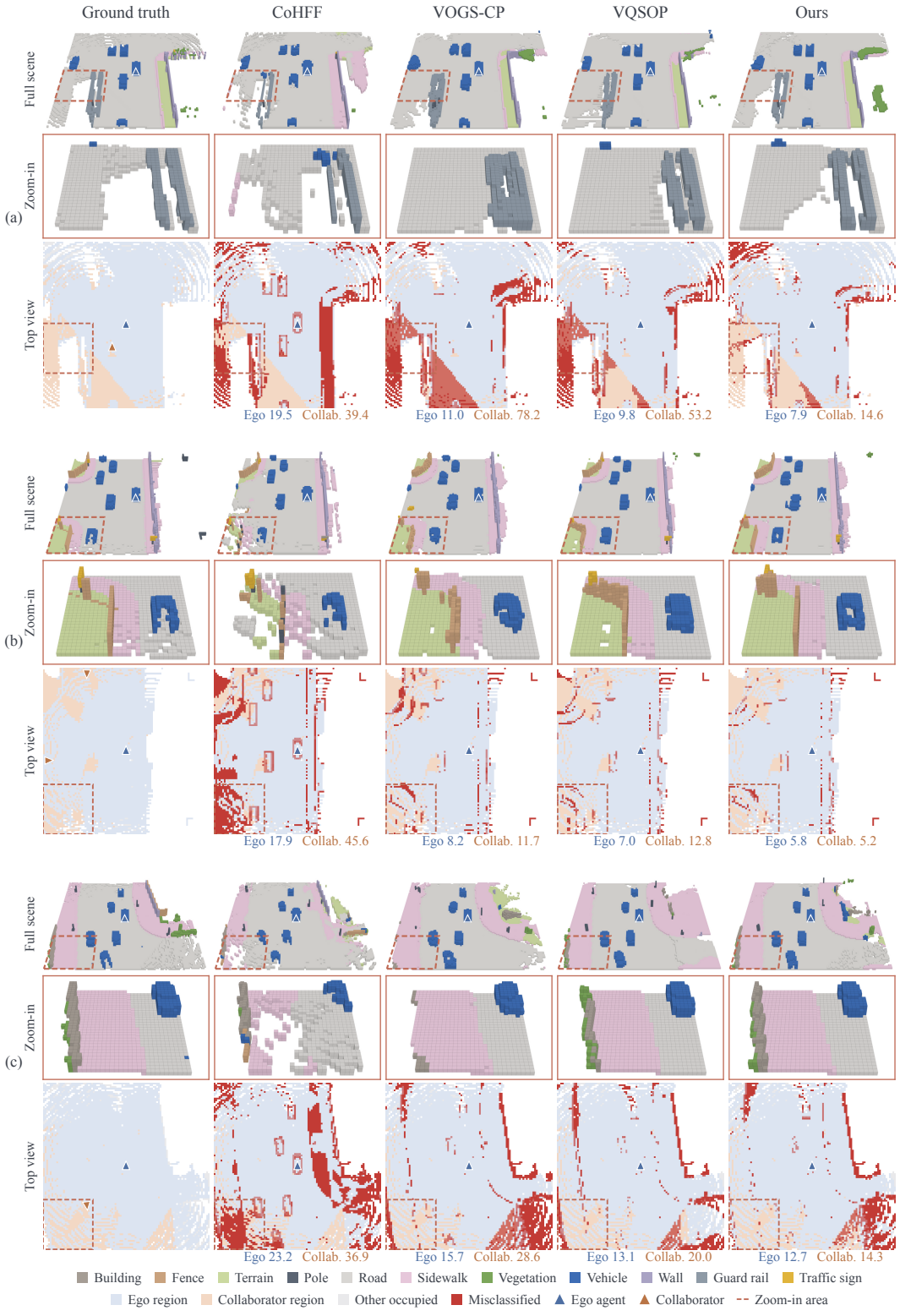


Figure 5: Additional qualitative comparison on the Semantic-OPV2V validation set, with the same layout as Figure 3. The zoom-ins focus on guard rails, a fence, and a building.