LR-V2X: Loss-Resilient Collaborative Perception under Low-Bandwidth Communication

Kang Yang[1] **Tianci Bu**[2] **Peng Wang**[1]
Deying Li[1] **Yongcai Wang**[1,3*]

[1]Renmin University of China
[2]The Hong Kong University of Science and Technology (HKUST)
[3]Hebei Key Laboratory of Real-virtual Integrated Autonomous Systems (RIAS)
y1127238112@gmail.com, btc010001@gmail.com
{peng.wang,deyingli,ycw}@ruc.edu.cn

Abstract

Given the inherent unpredictability of packet loss in vehicular wireless communications, V2X collaborative perception can yield practical benefits only if agents can achieve reliable collaboration under lossy and low-bandwidth communication conditions. Existing dense BEV feature fusion methods exchange large feature maps, while compact-communication methods compress messages aggressively and have limited ability to recover missing content after packet loss. In this paper, we present LR-V2X, a loss-resilient latent-space reconstruction framework. It converts corrupted received latents, even under 90% packet loss, into a spatial prior and reconstructs missing BEV information using this prior and ego context. The model is trained under complete communication and applied directly to lossy conditions at test time, avoiding packet-loss-specific training. Experiments on DAIR-V2X and V2XREAL show that LR-V2X preserves reliable collaboration as communication quality degrades and achieves the highest overall mAP among the compared methods at a matched 128 KiB message budget under 90% packet loss. It reduces communication overhead by $64\times$ compared to dense BEV feature fusion baselines. Code is available at `https://github.com/sidiangongyuan/LR-V2X`.

1 Introduction

V2X collaborative perception extends the sensing coverage of the ego vehicle by sharing intermediate representations among multiple agents, thereby enhancing driving safety in complex traffic scenarios [43, 48, 44, 40]. At present, mainstream intermediate fusion paradigms predominantly transmit BEV feature maps, enabling the ego agent to fuse multi-agent contextual information for accurate 3D object detection [1, 3, 37, 43, 42]. However, a critical practical bottleneck lies in maintaining reliable collaborative perception under real-world wireless communication constraints. The *bandwidth limits* force each agent to communicate compact features, while *packet loss* may randomly drop spatial portions of those transmitted features during delivery [30, 9, 15]. Therefore, under low-bandwidth conditions, the ego agent often receives compact yet incomplete information from collaborative agents. Reliable cooperation thus hinges on effectively recovering meaningful BEV contextual features from such corrupted transmissions. Fig. 1 illustrates this failure scenario and our reconstruction-first paradigm.

Existing methods have only addressed this deployment bottleneck partially. Dense BEV feature fusion baselines [43, 42, 25, 41] can preserve accuracy by transmitting highly redundant BEV features,

*Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

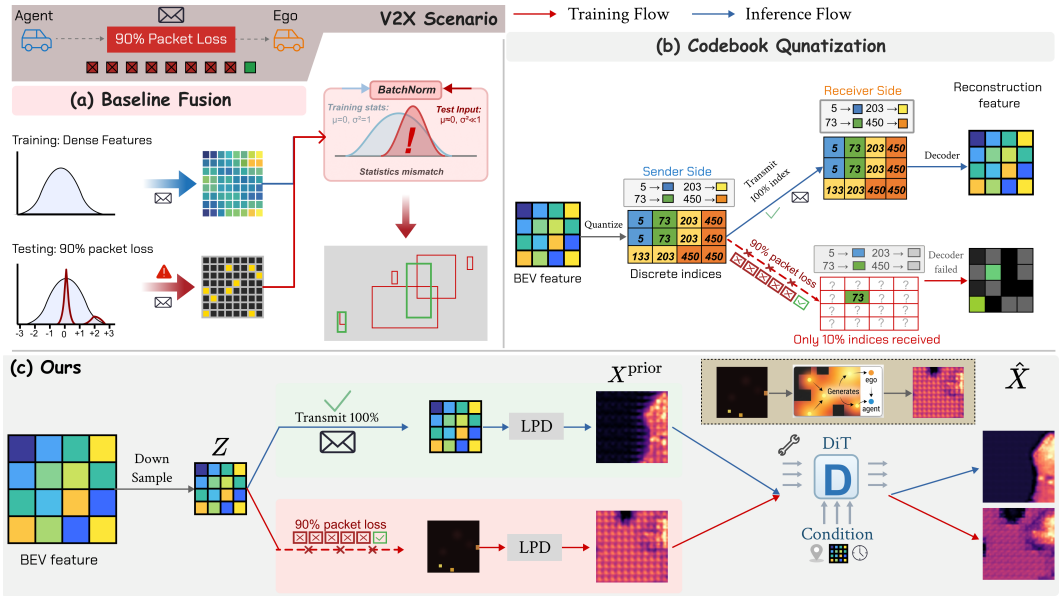


Figure 1: Motivation and overview. (a) Baseline fusion receives spatially incomplete feature maps when spatial packets are lost. (b) Codebook-based methods transmit compact codes, but missing indices leave spatial holes that degrade detection. (c) LR-V2X reconstructs complete BEV features from corrupted received latents via prior-initialized feature reconstruction, enabling graceful degradation under packet loss.

but this robustness is tied to heavy communication overhead. Communication-efficient methods [12, 14, 52, 4, 7] reduce traffic through region selection, compact codes, or learned compression, yet once packets are lost, they generally lack an explicit mechanism to recover the missing feature content. The central challenge is to maintain reliable collaboration when a compact transmitted message is further corrupted in transit.

Packet loss removes local evidence and induces an input distribution shift relative to ideal communication. When a received feature map becomes spatially sparse, its statistics deviate substantially from those observed during training with complete messages. As shown in Fig. 2, severe packet loss moves the packet-loss regime away from the no-packet-loss reference regime, so the fusion module must operate on inputs that are systematically mismatched to its training distribution. A dense BEV feature fusion baseline such as F-Cooper can partially buffer this shift through communication redundancy. At low bandwidth, explicit recovery of missing regions provides a way to use the limited surviving evidence. This motivates a reconstruction-centric design for reliable low-bandwidth collaboration under packet loss [45, 6].

To reconstruct a neighbor's BEV feature from a received, packet-corrupted compact latent, we propose two key components for loss-resilient V2X: (1) inspired by the capability of overlapping transposed convolutions [18, 27], which can propagate evidence from received locations into missing spatial regions, we design a **Latent Prior Decoder (LPD)**. LPD extrapolates features from the successfully received locations into a spatially complete prior BEV map. This map provides a structured starting point for the noise-conditioned reconstructor. (2) a **noise-conditioned DiT [29] reconstructor** that reconstructs from the prior BEV map and is guided by ego-side context. The key insight is that the ego-side observation provides supplementary information for reconstructing corrupted latent features received from neighbors, since the ego agent generally shares a large common view with the neighbors. Therefore, the noise-conditioned DiT reconstructor refines the prior constructed by LPD, conditioned jointly on the received compact latent and the ego agent's own BEV to fill the missing regions. Following a diffusion-style training strategy, we perturb the prior with noise at randomly sampled levels, so the denoiser learns to recover from priors of varying quality and generalizes to unseen packet-loss rates at test time.

Our contributions are:

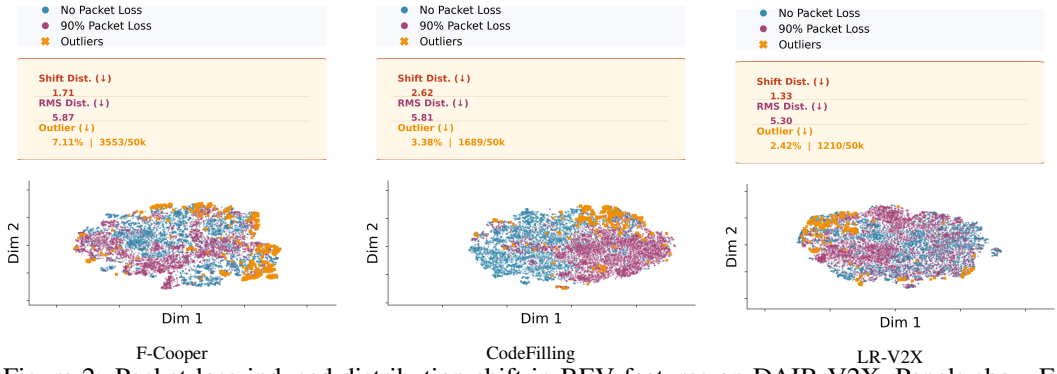


Figure 2: Packet-loss-induced distribution shift in BEV features on DAIR-V2X. Panels show F-Cooper, CodeFilling, and LR-V2X from left to right. Shift Dist. measures the Mahalanobis distance between packet-loss and reference centroids, RMS Dist. measures the root-mean-square Mahalanobis distance of packet-loss features, and Outlier Rate counts packet-loss features outside the reference high-tail threshold. Lower values indicate a smaller departure from the no-packet-loss reference regime.

- We identify feature-level packet loss as a critical yet underexplored failure mode in collaborative perception and show that sparse receptions induce a distribution shift that existing methods do not explicitly address.
- We propose LR-V2X, a reconstruction framework that recovers missing BEV context from compact, packet-corrupted collaborator latents through prior-guided, ego-conditioned feature reconstruction.
- We show that training only under complete communication, combined with noise-conditioned prior refinement, can handle packet-loss-corrupted inputs at test time without packet-loss-specific training.
- We show that LR-V2X achieves the highest overall mAP among the compared methods at a matched 128 KiB message budget under 90% packet loss, with $64\times$ less communication than dense BEV feature fusion baselines.

2 Related Work

2.1 Collaborative Perception

Collaborative perception enables agents to share sensory information and overcome individual perceptual limitations. The prevailing paradigm is intermediate fusion, which exchanges BEV feature maps to balance communication efficiency and fusion effectiveness, avoiding the prohibitive bandwidth of raw-data exchange [1, 3] while surpassing late fusion. Early works used graph neural network (GNN)-based message passing [37]; AttnFuse [43] introduced attention-based inter-agent modeling, later scaled to global context via V2X-ViT [42]. DiscoNet [22] explored knowledge distillation as a complementary enrichment strategy.

Robustness to real-world deployment uncertainties has been studied along several axes: pose errors [35, 24, 16, 49, 34], communication latency [20, 21, 33, 38], and heterogeneous agent configurations [41, 13, 47, 36, 51, 25, 39, 8]. These efforts address localization imprecision, asynchrony, and cross-agent incompatibilities, which are orthogonal to the *transmission reliability* problem we study.

Dense BEV feature fusion baselines also pose bandwidth challenges. Where2comm [12] learns to select salient regions, while instance-level methods further reduce bandwidth via object-centric representations [4, 7, 11, 46]. CodeFilling [14] frames collaboration as compression-reconstruction, extended by QuantV2X [52] with quantized representations. Generative methods use diffusion to reconstruct compressed features [28], correct pose-induced degradation [17], or bridge heterogeneous feature spaces [53]. Our work addresses *stochastic packet loss* on already-compact latents, where the missingness pattern varies per transmission.

2.2 Generative Reconstruction for Collaborative Perception

When transmitted BEV features are aggressively compressed or spatially corrupted by packet loss, recovering complete, detection-ready features on the receiver side becomes the central challenge. Diffusion models offer a principled mechanism for this: from DDPM [10] through efficient samplers such as DDIM [32] and latent-space variants [31], they learn to reverse noise and synthesize structured signals from corrupted inputs. Conditioning mechanisms including ILVR [5], ControlNet [50], and RePaint [26] extend this to reconstruction from partial or spatially incomplete observations, making them well-suited for feature recovery in V2X settings.

These properties have motivated applying generative reconstruction in collaborative perception. CodeFilling [14] and QuantV2X [52] treat collaboration as compression-reconstruction using discrete codebooks. DiffCP [28] reconstructs individual-agent features from ultra-compact representations before fusion. CoDiff [17] denoises fused features degraded by pose errors. GenComm [53] generates features in the ego's semantic space to bridge gaps between heterogeneous agents. LR-V2X places reconstruction at the receiver after packet erasure and before fusion. It combines a prior decoded from the surviving latent, aligned ego context, and one noise-conditioned refinement step to recover the collaborator BEV.

3 Method

3.1 Problem Formulation

Scenario. We consider a V2X collaborative perception system with N agents $\mathcal{A} = \{A_0, A_1, \ldots, A_{N-1}\}$ communicating over a wireless network. Agent A_0 denotes the ego vehicle. Each agent A_i operates in a local coordinate frame and observes the scene through onboard sensors, producing bird's-eye-view (BEV) features $X_i \in \mathbb{R}^{H\times W\times C}$. The spatial transformation from agent A_i to agent A_j is represented by an affine matrix $T_{i\to j} \in \mathbb{R}^{2\times 3}$, which stacks a 2×2 linear part (rotation/scale) and a 2×1 translation vector.

Communication Graph. Agents communicate over a directed graph $\mathcal{G} = (\mathcal{A}, \mathcal{P})$, where an edge $(i, j) \in \mathcal{P}$ exists if agent A_i can transmit to agent A_j. Each agent A_i transmits a message S_i derived from its local BEV feature X_i, subject to a communication budget:

$$S_i = \mathcal{C}(X_i), \qquad \text{size}(S_i) \le B, \tag{1}$$

where $\mathcal{C}(\cdot)$ denotes a generic communication function and B is the maximum message-size budget per transmitted collaborator frame.

Packet Loss Model. During transmission over edge (i, j), we consider feature-level packet loss at the spatial-packet granularity. Let $M_{i\to j}$ denote the reception mask defined on the spatial layout of the transmitted message, where $M_{i\to j}(u, v) = 1$ indicates successful reception of the packet at spatial location (u, v). We assume the following random packet corruption process:

$$M_{i\to j}(u, v) \sim \text{Bernoulli}(p), \quad \tilde{S}_{i\to j} = S_i \odot M_{i\to j}, \tag{2}$$

where $p \in [0, 1]$ is the retention rate and $\odot$ denotes element-wise multiplication across all channels. We refer to $\tilde{S}_{i\to j}$ as the received message. In the ego-centric setting, we use the shorthand $\tilde{S}_i \equiv \tilde{S}_{i\to 0}$ for $i \neq 0$, where the subscript 0 denotes the ego agent.

Optimization Objective. For each ego vehicle A_0, the goal is to perform collaborative 3D object detection from its own BEV feature and the messages received from other agents. The ego uses its own feature directly, while each collaborator feature is reconstructed in the collaborator coordinate frame from the received message, the ego feature, and the relative transform:

$$\hat{X}_0 = X_0, \qquad \hat{X}_i = \mathcal{R}(\tilde{S}_i, X_0, T_{0\to i}; \boldsymbol{\theta}_r), \quad \forall i \neq 0, \tag{3}$$

where $\boldsymbol{\theta}_r$ is the learnable parameter of the reconstruction model $\mathcal{R}$. Then, before fusion, each reconstructed collaborator feature is warped back to the ego coordinate frame, denoted by $\bar{X}_i = \mathcal{W}(\hat{X}_i, T_{i\to 0})$ for $i \neq 0$ and $\bar{X}_0 = X_0$. These ego-aligned features are fused for ego-centric detection:

$$\hat{Y} = \mathcal{D}\left(\mathcal{F}\left(\{\bar{X}_i\}_{i=0}^{N-1}; \boldsymbol{\theta}_f\right); \boldsymbol{\theta}_d\right). \tag{4}$$

Here $\boldsymbol{\theta}_f$ and $\boldsymbol{\theta}_d$ denote the learnable parameters of the multi-agent fusion module $\mathcal{F}$ and the detection head $\mathcal{D}$, respectively. The training objective minimizes the detection loss

$$\mathcal{L}_{\text{det}}(\hat{Y}, Y), \tag{5}$$

where Y denotes ground-truth object annotations. We evaluate the robustness of the above collaborative perception process under varying retention rates p and random masks M to reflect stochastic packet loss during transmission.

3.2 Framework Overview

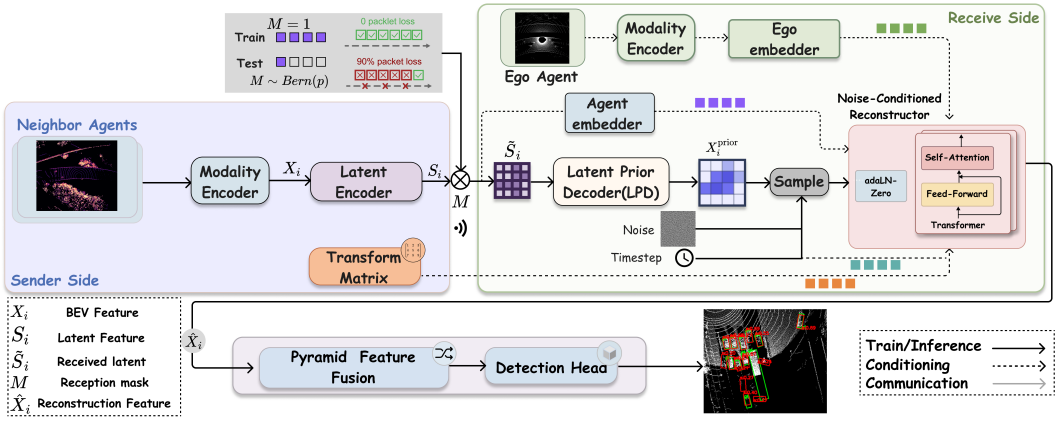


Figure 3: Framework of LR-V2X. Each agent encodes BEV features into a compact latent for transmission. The receiver reconstructs the agent's BEV via the Latent Prior Decoder and a noise-conditioned reconstructor, then fuses with its own features for 3D detection.

The received message is simultaneously (i) compact after compression and (ii) spatially incomplete after packet loss, making direct BEV regression from corrupted compact observations ill-conditioned. LR-V2X resolves this tension through a coarse-to-fine recovery pipeline. First, latent encoding reduces the transmitted BEV feature to a compact spatial carrier, establishing the low-bandwidth operating point. Second, the Latent Prior Decoder (LPD) converts the corrupted received latent into a dense but coarse BEV prior by propagating the remaining spatial evidence into neighboring missing regions. Third, a noise-conditioned reconstructor refines this prior using both the received latent and ego-side context.

We instantiate the communication function as a learned latent encoder:

$$S_i = \mathcal{E}(X_i; \boldsymbol{\theta}_e) \in \mathbb{R}^{h \times w \times C'}, \tag{6}$$

where $\mathcal{E}(\cdot; \boldsymbol{\theta}_e)$ maps the collaborator BEV feature to a compact spatial latent with $h \ll H$, $w \ll W$, and $C' \leq C$. Under packet loss, the receiver observes $\tilde{S}_i = S_i \odot M_{i \to 0}$, a spatially incomplete version of this latent. The following subsections detail how the LPD converts $\tilde{S}_i$ into a dense prior and how the noise-conditioned reconstructor refines it into $\hat{X}_i$.

3.3 Latent Prior Decoder

The LPD provides a structured initialization before the transformer denoiser is applied. Packet loss turns the compact latent into a spatially incomplete observation whose missing locations provide no direct evidence. The LPD converts the received latent $\tilde{S}_i$ into a spatially complete but coarse BEV map X_i^{prior} that anchors the following reconstruction stage.

Concretely, the LPD maps $\tilde{S}_i$ to X_i^{prior} via four transposed convolutional layers [23] with overlapping receptive fields, upsampling from latent resolution (h, w) to BEV resolution (H, W) (Fig. 4):

$$X_i^{\text{prior}} = \text{LPD}(\tilde{S}_i). \tag{7}$$

The overlapping kernels spread activations from received locations into neighboring missing regions, converting a spatially incomplete latent into a spatially complete prior BEV map [18, 27]. The LPD

is trained on complete latents ($\tilde{S}_i = S_i$). At test time, the same overlapping transposed convolutions aggregate the available neighboring evidence from sparse receptions.

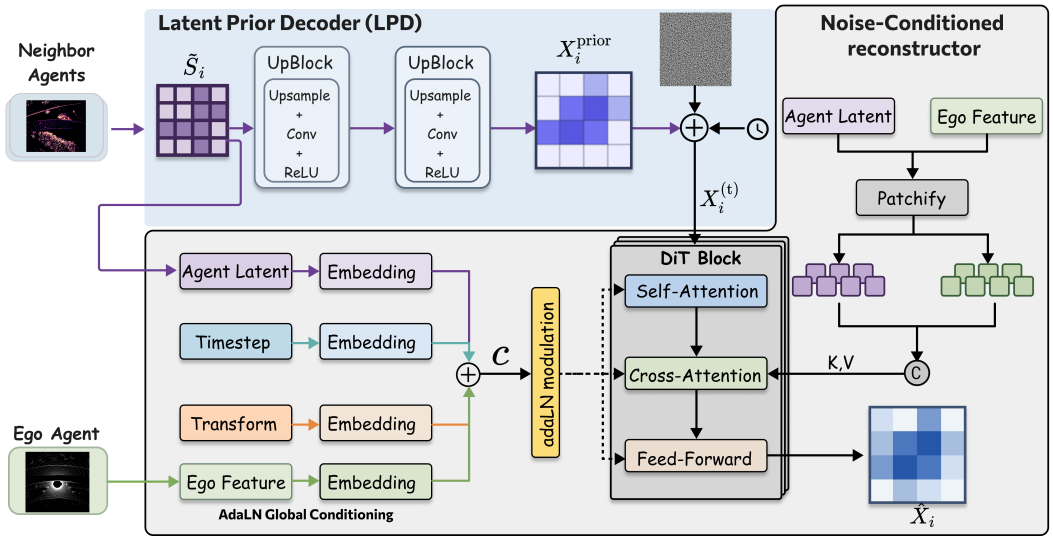


Figure 4: Architecture of the Latent Prior Decoder (LPD) and the DiT denoiser used for conditional feature reconstruction.

3.4 Noise-Conditioned Feature Reconstruction

The noise-conditioned reconstructor refines an imperfect LPD prior. During training, we perturb this prior with Gaussian noise and supervise recovery of the clean collaborator BEV. This exposes the model to priors of varying quality. At test time, the input prior is degraded by missing latent packets. Both settings use the same conditional recovery interface and a single reconstruction step.

Given $\tilde{S}_i$, the LPD first produces the prior BEV map X_i^{prior}. Gaussian noise is then injected at a randomly sampled timestep t, yielding the noisy prior $X_i^{(t)}$:

$$X_i^{(t)} = \sqrt{\bar{\alpha}_t}\, X_i^{\text{prior}} + \sqrt{1-\bar{\alpha}_t}\, \epsilon, \tag{8}$$

where $\epsilon \sim \mathcal{N}(0, I)$ and $\bar{\alpha}_t$ is the cumulative noise schedule [10]. A conditional denoiser $f_{\boldsymbol{\theta}_r}$ then recovers the clean BEV feature $\hat{X}_i$ from the noisy prior, conditioned on the received latent $\tilde{S}_i$, the per-collaborator aligned ego BEV $X_0^{\text{aligned}} = \mathcal{W}(X_0, T_{0\to i})$ (warped from X_0 to agent i's coordinate frame via bilinear interpolation), and the spatial transform $T_{0\to i}$:

$$\hat{X}_i = f_{\boldsymbol{\theta}_r}(X_i^{(t)}, t, \tilde{S}_i, X_0^{\text{aligned}}, T_{0\to i}). \tag{9}$$

We train with an x_0-prediction objective:

$$\mathcal{L}_{\text{diff}} = \mathbb{E}_{t,\epsilon}\left[\| f_{\boldsymbol{\theta}_r}(X_i^{(t)}, t, \tilde{S}_i, X_0^{\text{aligned}}, T_{0\to i}) - X_i \|^2\right]. \tag{10}$$

3.4.1 Noise-Conditioned Reconstructor Architecture

We adopt DiT [29] as the denoiser backbone. The received latent $\tilde{S}_i$ constrains reconstruction in two complementary ways: it provides a global conditioning signal that steers the denoising trajectory toward the transmitted collaborator content, and it also serves as spatial evidence during cross-attention. The aligned ego BEV X_0^{aligned} is introduced as additional cross-attention context, allowing the model to combine the transmitted message with ego-side scene structure when recovering missing regions. For layer ℓ, the DiT denoiser block performs (Fig. 4)

$$\begin{aligned} h_\ell &= \text{SelfAttn}(\text{AdaLN}(x_\ell, c)) + x_\ell \\ h'_\ell &= \text{CrossAttn}(\text{AdaLN}(h_\ell, c),\, [\tilde{S}_i;\, X_0^{\text{aligned}}]) + h_\ell \\ x_{\ell+1} &= \text{FFN}(\text{AdaLN}(h'_\ell, c)) + h'_\ell. \end{aligned} \tag{11}$$

The global conditioning vector is

$$c = \phi_t(t) + \phi_S(\text{pool}(\tilde{S}_i)) + \phi_X(\text{pool}(X_0^{\text{aligned}})) + \phi_T(T_{0\to i}), \tag{12}$$

where $\phi_t, \phi_S, \phi_X, \phi_T$ are learned MLP projections, and $\text{AdaLN}(x, c)$ applies layer normalization with per-channel scale and shift linearly projected from c. Cross-attention allows the model to query both transmitted evidence and ego context when reconstructing missing regions. This combination lets the denoiser preserve global consistency with the received message while restoring local object structure in corrupted regions.

3.4.2 Training and Inference

Training follows a stage-wise protocol. We first train the PointPillar encoder, pyramid fusion backbone, and detection heads under complete communication. We then freeze the sensor encoder/backbone and train the latent encoder $\mathcal{E}$, LPD, and DiT reconstructor by minimizing $\mathcal{L}_{\text{diff}}$ (Eq. 10) under complete communication ($\tilde{S}_i = S_i$), with t sampled uniformly from $\{1, \ldots, T\}$ at each step. Finally, we fine-tune the reconstruction module together with the fusion and detection heads while keeping the sensor encoder/backbone frozen. This exposes the denoiser to priors of varying quality without requiring packet-loss-corrupted messages during training. Gaussian perturbation and packet erasure produce different corruptions, but both require recovering the collaborator BEV from an imperfect prior. Aligned ego context remains available in both settings and reduces ambiguity in overlapping regions. At inference, given $\tilde{S}_i$ from agent i ($i \neq 0$), we compute $X_i^{\text{prior}} = \text{LPD}(\tilde{S}_i)$, draw Gaussian noise, perturb the prior to a fixed noise level t_{inf} (a hyperparameter), and recover $\hat{X}_i$ in a single forward pass of $f_{\boldsymbol{\theta}_r}$. The reconstructed collaborator feature is then aligned back to the ego frame inside the fusion module. For end-to-end optimization with the detection task, the total loss combines reconstruction and detection objectives

$$\mathcal{L}_{\text{total}} = \lambda_{\text{det}}\mathcal{L}_{\text{det}}(\mathcal{D}(\mathcal{F}(\{\bar{X}_i\})), Y) + \lambda_{\text{diff}}\mathcal{L}_{\text{diff}}, \tag{13}$$

where $\lambda_{\text{det}}, \lambda_{\text{diff}}$ are the balancing weights.

4 Experiments

4.1 Setup

We evaluate on two real-world V2X datasets: DAIR-V2X [48] and V2XREAL [40]. All methods use a PointPillars [19] backbone and are trained under complete communication (no packet loss). The protocols pair one vehicle with one roadside unit on DAIR-V2X and two vehicles on V2XREAL. At test time, each spatial packet is independently dropped with probability $1 - p$, where p is the retention rate. Section 4.3 and Appendix B.1 additionally evaluate temporally persistent and spatially correlated loss. Masks affect transmitted collaborator messages; the ego feature remains available locally. We use fixed random seeds for mask generation and inference noise. We report dataset-specific AP breakdowns and overall mAP at IoU 0.3, 0.5, and 0.7, per-collaborator pre-loss payload, and inference speed (FPS). Full implementation and training details appear in Appendix A.

4.2 Performance

Table 1 reports dataset-specific AP breakdowns and overall mAP at IoU 0.3, 0.5, and 0.7, along with communication cost and inference speed. On V2XREAL (Table 1a), LR-V2X achieves the highest overall mAP across all three IoU thresholds under 90% packet loss (44.87/38.60/24.18), while transmitting only a 128 KiB collaborator message ($64\times$ less than dense BEV feature fusion baselines). Compared to CodeFilling at the same 128 KiB budget, LR-V2X improves mAP@0.5 by 1.91 and mAP@0.7 by 1.02 points, and it also outperforms the protocol-matched GenComm† re-implementation. On DAIR-V2X (Table 1b), LR-V2X likewise achieves the highest overall mAP across all three IoU thresholds (69.35/66.90/53.37), surpassing all dense BEV feature fusion baselines that require $64\times$ more communication. Under complete communication, LR-V2X also attains the highest overall mAP among the no-packet-loss references reported here on both datasets.

The range-wise results show complementary strengths: Attn leads the DAIR-V2X Middle range at IoU 0.3 and 0.5, while LR-V2X leads at IoU 0.7. The overall metric pools detections across the full

Table 1: Performance comparison under **Complete Communication (0% Packet Loss)** reference conditions and the main **90% Packet Loss** robustness setting.

(a) V2XREAL dataset (per-class AP and mAP).

Models	AP_vehicle			AP_pedestrian			AP_truck			mAP (%)			Comm	FPS
	0.3	0.5	0.7	0.3	0.5	0.7	0.3	0.5	0.7	0.3	0.5	0.7		
No Packet Loss														
No Fusion	32.27	17.20	1.31	36.39	30.91	10.79	55.11	52.76	37.14	41.26	33.63	16.41	0	2.47
Early Fusion	68.17	64.73	40.03	33.53	17.14	0.87	27.22	25.57	14.18	42.97	35.81	18.36	∼2.4MB	1.03
Late Fusion	32.54	16.23	1.17	34.12	30.13	13.70	61.55	58.96	38.99	42.73	35.10	17.96	≤5.4KB	1.21
LR-V2X	76.70	73.85	49.58	35.98	17.40	1.12	48.38	43.60	32.64	53.69	44.95	27.78	128KiB	1.54
90% Packet Loss														
Attn	59.35	55.93	39.25	29.30	15.20	1.08	39.00	35.50	20.66	42.55	35.54	20.33	8MiB	1.24
F-Cooper	57.94	54.22	38.00	31.92	16.31	1.08	37.60	32.40	12.60	42.49	34.30	17.21	8MiB	0.82
CoBEVT	51.54	46.45	30.60	25.25	11.35	0.50	39.99	34.98	15.82	38.92	30.93	15.64	8MiB	0.82
V2X-VIT	49.37	46.18	31.40	**34.23**	**17.77**	**1.38**	38.60	32.90	24.00	40.73	32.29	18.93	8MiB	0.70
CodeFilling	62.14	57.95	39.36	16.41	12.92	0.60	**43.07**	39.19	**29.52**	40.54	36.69	23.16	128KiB	**1.68**
HEAL	62.31	60.85	39.29	26.53	12.96	0.99	40.36	39.09	27.10	43.07	37.64	21.79	8MiB	1.61
GenComm†	63.11	56.60	30.38	12.99	3.83	0.15	33.33	29.97	17.48	36.48	30.14	16.00	128KiB	0.37
LR-V2X	**64.60**	**63.59**	**46.31**	28.55	12.23	1.09	41.48	**39.98**	25.12	**44.87**	**38.60**	**24.18**	128KiB	1.54

(b) DAIR-V2X dataset (per-range AP and mAP).

Models	Short			Middle			Long			mAP (%)			Comm	FPS
	0.3	0.5	0.7	0.3	0.5	0.7	0.3	0.5	0.7	0.3	0.5	0.7		
No Packet Loss														
No Fusion	79.25	77.24	70.12	75.70	72.49	62.20	54.78	51.06	37.94	62.51	59.33	49.67	0	8.70
Early Fusion	84.47	81.40	68.88	78.60	72.88	56.54	59.79	54.55	36.13	73.52	68.61	52.37	∼1.0MB	14.95
Late Fusion	79.25	77.24	70.12	75.70	72.49	62.20	54.78	51.06	37.94	69.51	66.33	55.67	≤5.4KB	7.75
LR-V2X	88.48	85.84	72.00	83.56	78.38	58.51	71.56	66.04	46.16	81.03	76.16	57.74	128KiB	6.04
90% Packet Loss														
Attn	80.53	79.02	66.72	**77.47**	**71.70**	56.83	**57.23**	51.41	**38.45**	66.87	66.20	52.56	8MiB	6.67
F-Cooper	80.43	77.14	67.13	76.96	71.62	57.04	57.15	**51.65**	38.44	67.89	65.90	52.87	8MiB	5.39
CoBEVT	79.98	76.17	61.25	75.15	69.79	51.91	53.46	47.68	33.25	68.83	63.55	44.47	8MiB	4.92
V2X-VIT	80.01	77.06	66.55	75.34	71.04	56.20	53.23	48.04	34.61	68.78	64.33	50.97	8MiB	4.70
CodeFilling	76.92	75.24	66.22	69.68	67.03	55.20	47.43	44.80	35.29	63.38	60.93	50.62	128KiB	**9.17**
HEAL	76.88	75.26	65.77	70.48	67.64	54.60	49.39	46.20	35.69	64.43	61.68	50.50	8MiB	7.24
GenComm†	67.89	62.56	37.75	49.68	44.51	22.82	32.00	27.81	13.20	48.03	43.16	23.27	128KiB	0.49
LR-V2X	**82.59**	**80.93**	**69.98**	75.88	70.03	**57.93**	55.25	49.86	36.96	**69.35**	**66.90**	**53.37**	128KiB	6.04

† Re-implementation of GenComm's DDIM reconstruction module under our backbone and training protocol.

Communication cost is reported per transmitted collaborator frame before packet loss. Raw-point early fusion uses float32 XYZI payloads estimated from validation files (V2XREAL: ∼2.4 MB; DAIR-V2X: ∼1.0 MB). Dense BEV and compact latent messages use float32 tensor sizes. Late fusion reports a compact 7-DoF box, score, and class payload with the post-NMS cap of 150 boxes, giving ≤5.4 KB.

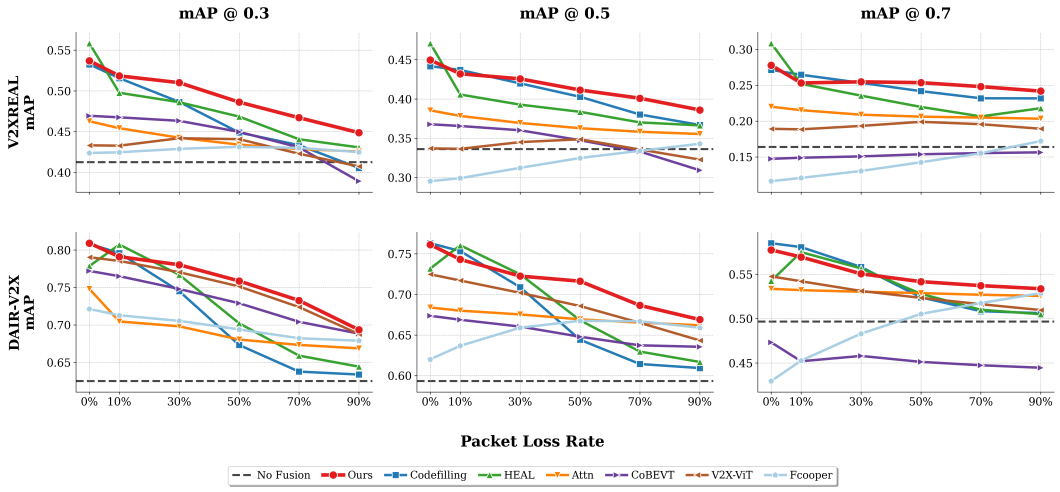


Figure 5: Performance under varying packet loss rates on DAIR-V2X and V2XREAL. Across the sweep, LR-V2X remains consistently competitive as communication quality decreases and reaches the highest overall mAP at the 90% packet-loss endpoint.

Table 2: Temporal packet loss on DAIR-V2X at 90% marginal loss and approximately 128 KiB per collaborator frame. Results are mean ± standard deviation over three mask seeds; bold marks the best mean at each K.

Method	K	mAP@0.3	mAP@0.5	mAP@0.7
LR-V2X	1	$\mathbf{69.36 \pm 0.12}$	$\mathbf{66.33 \pm 0.09}$	$\mathbf{53.47 \pm 0.04}$
LR-V2X	10	$\mathbf{69.01 \pm 0.12}$	$\mathbf{66.27 \pm 0.08}$	$\mathbf{53.14 \pm 0.02}$
CoBEVT	1	68.72 ± 0.05	63.80 ± 0.04	44.35 ± 0.03
CoBEVT	10	68.31 ± 0.10	64.86 ± 0.10	44.00 ± 0.09
Where2comm	1	61.89 ± 0.01	57.28 ± 0.02	39.62 ± 0.01
Where2comm	10	61.89 ± 0.03	57.30 ± 0.01	39.61 ± 0.01

evaluation range. Fig. 5 shows that LR-V2X remains competitive as loss increases and achieves the highest overall mAP among the plotted methods at the 90% endpoint.

Bandwidth–robustness asymmetry. Dense BEV feature fusion baselines transmit 8 MiB per collaborator message ($64\times$ our budget). At 90% loss, their expected retained payload is 819.2 KiB, compared with 12.8 KiB for a 128 KiB latent. This communication redundancy helps explain their robustness (Appendix B.3). Appendix Table 5 controls for message size: matching all five dense baselines to 128 KiB decreases their mAP@0.5. At this budget, LR-V2X achieves the highest overall mAP (69.35/66.90/53.37), supporting the value of explicit reconstruction for low-bandwidth robustness.

Appendix Table 3 replaces i.i.d. erasures with correlated spatial burst loss. At the same 128 KiB budget, LR-V2X outperforms CodeFilling by 2.47/1.59/0.45 mAP points. Appendix Table 6 provides additional comparisons with Where2comm and DiscoNet.

4.3 Robustness and Efficiency

Temporal packet loss. We evaluate DAIR-V2X at 90% marginal loss with a 128 KiB payload. With $K = 1$, the packet mask is resampled every frame; $K = 10$ holds the same mask for ten consecutive frames within a sequence. Table 2 reports mean and standard deviation over three packet-mask seeds with fixed trained models.

LR-V2X decreases by 0.35/0.06/0.33 points from $K = 1$ to $K = 10$ and remains highest at all three thresholds. Each frame uses fresh ego context and surviving collaborator values, with no recurrent reconstruction state. This helps explain its stability when missing locations persist. Appendix B.2 details the protocol.

Inference efficiency. We profile one RTX 4090 at batch size 1 with two agents. Sender compression takes 2.64 ms; receiver reconstruction takes 47.44 ms, including 45.80 ms for the single-step DiT. Total forward time is 84.79 ± 3.24 ms and peak allocated memory is 1398.94 ± 0.76 MiB. We use 20 warmup frames and time 100 frames in each of three repeats. Appendix D.3 gives the full profile and communication-time estimates.

Qualitative comparison. Fig. 6 compares detection outputs on both datasets under 90% packet loss. LR-V2X preserves more detections aligned with the ground truth in these examples. Appendix Fig. 10 complements these results with a feature-level reconstruction example.

4.4 Ablation Study

Conditioning inputs. Appendix Table 7 ablates the reconstructor conditions under complete communication with the latent-derived prior fixed. Ego BEV has the largest isolated effect: adding it raises mAP@0.3 by 11.62 points. The transformation matrix adds 0.79 points on top of ego BEV, while the direct latent branch alone changes mAP@0.3 by 0.02 points. The overlapping fields of view explain this role: aligned ego features provide spatial anchors for reconstructing collaborator content.

Downsampling ratio. We use $\times 8$ (128 KiB), which cuts communication $16\times$ from $\times 2$ with a 1.52-point mAP@0.3 drop (Appendix Table 8).

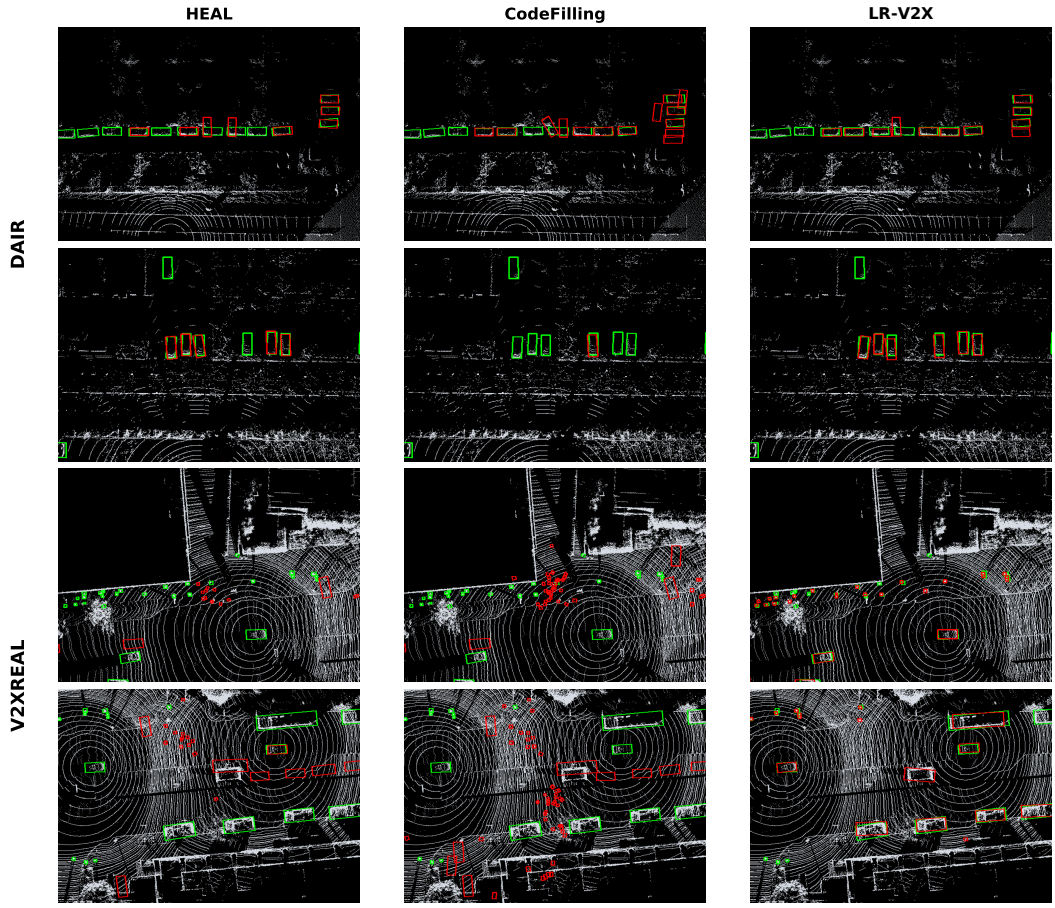


Figure 6: Detection results under 90% packet loss. Columns: HEAL, CodeFilling, and LR-V2X. The top two rows show DAIR-V2X; the bottom two show V2XREAL. Red boxes are predictions; green boxes are ground truth.

Latent Prior Decoder. Appendix Fig. 8 compares the full model with prior-only, which fuses the LPD output directly, and zero-prior, which starts the same noise-conditioned reconstructor from an all-zero BEV prior. Zero-prior retains aligned ego context and direct latent conditioning. The full model consistently outperforms both, supporting complementary roles for LPD initialization and conditioned refinement. The LPD takes 1.64 ms in the profile, a small cost relative to the 45.80 ms DiT refinement.

Inference timestep and noise-conditioned training. Appendix Table 9 shows that noise conditioning improves over a protocol-matched no-noise reconstructor by +7.16/+7.22/+4.23 mAP. This no-noise variant still uses the LPD prior and a learned reconstructor, but feeds the prior directly at $t = 0$ without the forward-noise perturbation. Appendix Fig. 9 further shows broadly stable performance with $t_{\text{inf}} = 700$ selected from the sweep.

5 Conclusion

LR-V2X reconstructs collaborator BEV features from packet-corrupted compact latents through an LPD prior and ego-conditioned, noise-conditioned refinement. Experiments on DAIR-V2X and V2XREAL show strong robustness under severe packet loss with a 128 KiB message. Training uses complete communication, while noise-conditioned refinement supports recovery at test time. Future work will evaluate larger cooperative groups and richer channel models.

Acknowledgments and Disclosure of Funding

This work received no third-party funding. The authors declare no competing interests.

References

[1] 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.

[2] Karthikeyan Chandra Sekaran, Markus Geisler, Dominik Rößle, Adithya Mohan, Daniel Cremers, Wolfgang Utschick, Michael Botsch, Werner Huber, and Torsten Schön. UrbanIng-V2X: A large-scale multi-vehicle, multi-infrastructure dataset across multiple intersections for cooperative perception. In *Advances in Neural Information Processing Systems*, volume 38. Curran Associates, Inc., 2025. doi: 10.52202/085713-4130.

[3] 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)*, pages 514–524. IEEE, 2019.

[4] 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*, pages 18205–18214, 2023.

[5] Jooyoung Choi, Sungwon Kim, Yonghyun Jeong, Youngjune Gwon, and Sungroh Yoon. Ilvr: Conditioning method for denoising diffusion probabilistic models. *arXiv preprint arXiv:2108.02938*, 2021.

[6] Gabriela Csurka. Domain adaptation for visual applications: A comprehensive survey. *arXiv preprint arXiv:1702.05374*, 2017.

[7] Siqi Fan, Haibao Yu, Wenxian Yang, Jirui Yuan, and Zaiqing Nie. Quest: Query stream for practical cooperative perception. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 18436–18442. IEEE, 2024.

[8] Xiangbo Gao, Runsheng Xu, Jiachen Li, Ziran Wang, Zhiwen Fan, and Zhengzhong Tu. Stamp: Scalable task and model-agnostic collaborative perception. *arXiv preprint arXiv:2501.18616*, 2025.

[9] Yushan Han, Hui Zhang, Huifang Li, Yi Jin, Congyan Lang, and Yidong Li. Collaborative perception in autonomous driving: Methods, datasets, and challenges. *IEEE Intelligent Transportation Systems Magazine*, 15(6):131–151, 2023.

[10] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[11] Tianyu Hong, Xiaobo Zhou, Wenkai Hu, Qi Xie, Zhihui Ke, and Tie Qiu. Communication-efficient multi-vehicle collaborative semantic segmentation via sparse 3d gaussian sharing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 28622–28631, October 2025.

[12] Yue Hu, Shaoheng Fang, Zixing Lei, Yiqi Zhong, and Siheng Chen. Where2comm: Communication-efficient collaborative perception via spatial confidence maps. *Advances in neural information processing systems*, 35:4874–4886, 2022.

[13] Yue Hu, Yifan Lu, Runsheng Xu, Weidi Xie, Siheng Chen, and Yanfeng Wang. Collaboration helps camera overtake lidar in 3d detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9243–9252, June 2023.

[14] 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*, pages 15481–15490, 2024.

[15] Tao Huang, Jianan Liu, Xi Zhou, Dinh C Nguyen, Mostafa Rahimi Azghadi, Yuxuan Xia, Qing-Long Han, and Sumei Sun. Vehicle-to-everything cooperative perception for autonomous driving. *Proceedings of the IEEE*, 2025.

[16] Zhe Huang, Shuo Wang, Yongcai Wang, Wanting Li, Deying Li, and Lei Wang. Roco: Robust cooperative perception by iterative object matching and pose adjustment. In *ACM Multimedia 2024*, 2024.

[17] Zhe Huang, Shuo Wang, Yongcai Wang, and Lei Wang. Codiff: Conditional diffusion model for collaborative 3d object detection. *arXiv preprint arXiv:2502.14891*, 2025.

[18] Jason Ku, Ali Harakeh, and Steven L Waslander. In defense of classical image processing: Fast depth completion on the cpu. In *2018 15th Conference on Computer and Robot Vision (CRV)*, pages 16–22. IEEE, 2018.

[19] Alex H. Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[20] Zixing Lei, Shunli Ren, Yue Hu, Wenjun Zhang, and Siheng Chen. Latency-aware collaborative perception. In *European Conference on Computer Vision*, pages 316–332. Springer, 2022.

[21] Jinlong Li, Runsheng Xu, Xinyu Liu, Jin Ma, Zicheng Chi, Jiaqi Ma, and Hongkai Yu. Learning for vehicle-to-vehicle cooperative perception under lossy communication. *IEEE Transactions on Intelligent Vehicles*, 2023.

[22] Yiming Li, Shunli Ren, Pengxiang Wu, Siheng Chen, Chen Feng, and Wenjun Zhang. Learning distilled collaboration graph for multi-agent perception. *Advances in Neural Information Processing Systems*, 34: 29541–29552, 2021.

[23] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[24] Yifan Lu, Quanhao Li, Baoan Liu, Mehrdad Dianati, Chen Feng, Siheng Chen, and Yanfeng Wang. Robust collaborative 3d object detection in presence of pose errors. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4812–4818. IEEE, 2023.

[25] Yifan Lu, Yue Hu, Yiqi Zhong, Dequan Wang, Yanfeng Wang, and Siheng Chen. An extensible framework for open heterogeneous collaborative perception. *arXiv preprint arXiv:2401.13964*, 2024.

[26] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11461–11471, 2022.

[27] Fangchang Ma, Guilherme Venturelli Cavalheiro, and Sertac Karaman. Sparse-to-dense: Depth prediction from sparse depth samples and a single image. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1–8. IEEE, 2018.

[28] Ruiqing Mao, Haotian Wu, Yukuan Jia, Zhaojun Nan, Yuxuan Sun, Sheng Zhou, Deniz Gündüz, and Zhisheng Niu. Diffcp: Ultra-low bit collaborative perception via diffusion model. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6587–6593. IEEE, 2025.

[29] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023.

[30] Shunli Ren, Siheng Chen, and Wenjun Zhang. Collaborative perception for autonomous driving: Current status and future trend. In *Proceedings of 2021 5th Chinese Conference on Swarm Intelligence and Cooperative Control*, pages 682–692. Springer, 2022.

[31] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[32] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

[33] Zhiying Song, Lei Yang, Fuxi Wen, and Jun Li. Traf-align: Trajectory-aware feature alignment for asynchronous multi-agent perception. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 12048–12057, 2025.

[34] Sanbao Su, Songyang Han, Yiming Li, Zhili Zhang, Chen Feng, Caiwen Ding, and Fei Miao. Collaborative multi-object tracking with conformal uncertainty propagation. *IEEE Robotics and Automation Letters*, 9 (4):3323–3330, 2024.

[35] Nicholas Vadivelu, Mengye Ren, James Tu, Jingkang Wang, and Raquel Urtasun. Learning to communicate and correct pose errors. In *Conference on Robot Learning*, pages 1195–1210. PMLR, 2021.

[36] Rujia Wang, Xiangbo Gao, Hao Xiang, Runsheng Xu, and Zhengzhong Tu. Cocmt: Communication-efficient cross-modal transformer for collaborative perception. *arXiv preprint arXiv:2503.13504*, 2025.

[37] 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 *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 605–621. Springer, 2020.

[38] Sizhe Wei, Yuxi Wei, Yue Hu, Yifan Lu, Yiqi Zhong, Siheng Chen, and Ya Zhang. Asynchrony-robust collaborative perception via bird's eye view flow. *Advances in Neural Information Processing Systems*, 36: 28462–28477, 2023.

[39] Hao Xiang, Runsheng Xu, and Jiaqi Ma. Hm-vit: Hetero-modal vehicle-to-vehicle cooperative perception with vision transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 284–295, 2023.

[40] Hao Xiang, Zhaoliang Zheng, Xin Xia, Runsheng Xu, Letian Gao, Zewei Zhou, Xu Han, Xinkai Ji, Mingxi Li, Zonglin Meng, Li Jin, Mingyue Lei, Zhaoyang Ma, Zihang He, Haoxuan Ma, Yunshuang Yuan, Yingqian Zhao, and Jiaqi Ma. V2X-Real: a largs-scale dataset for vehicle-to-everything cooperative perception. In *European Conference on Computer Vision*, pages 455–470. Springer, 2024. doi: 10.1007/978-3-031-72943-0_26.

[41] Runsheng Xu, Zhengzhong Tu, Hao Xiang, Wei Shao, Bolei Zhou, and Jiaqi Ma. Cobevt: Cooperative bird's eye view semantic segmentation with sparse transformers, 2022.

[42] Runsheng Xu, Hao Xiang, Zhengzhong Tu, Xin Xia, Ming-Hsuan Yang, and Jiaqi Ma. V2x-vit: Vehicle-to-everything cooperative perception with vision transformer, 2022.

[43] 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)*, pages 2583–2589. IEEE, 2022.

[44] Runsheng Xu, Xin Xia, Jinlong Li, Hanzhao Li, Shuo Zhang, Zhengzhong Tu, Zonglin Meng, Hao Xiang, Xiaoyu Dong, Rui Song, et al. 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*, pages 13712–13722, 2023.

[45] Jingkang Yang, Kaiyang Zhou, Yixuan Li, and Ziwei Liu. Generalized out-of-distribution detection: A survey. *International Journal of Computer Vision*, 132(12):5635–5662, 2024.

[46] Kang Yang, Tianci Bu, Lantao Li, Chunxu Li, Yongcai Wang, and Deying Li. Is discretization fusion all you need for collaborative perception? In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9590–9596, 2025. doi: 10.1109/ICRA55743.2025.11128776.

[47] Hongbo Yin, Daxin Tian, Chunmian Lin, Xuting Duan, Jianshan Zhou, Dezong Zhao, and Dongpu Cao. V2vformer++: Multi-modal vehicle-to-vehicle cooperative perception via global-local transformer. *IEEE Transactions on Intelligent Transportation Systems*, 25(2):2153–2166, 2023.

[48] Haibao Yu, Yizhen Luo, Mao Shu, Yiyi Huo, Zebang Yang, Yifeng Shi, Zhenglong Guo, Hanyu Li, Xing Hu, Jirui Yuan, et al. Dair-v2x: A large-scale dataset for vehicle-infrastructure cooperative 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21361–21370, 2022.

[49] Yunshuang Yuan, Yan Xia, Daniel Cremers, and Monika Sester. Sparsealign: A fully sparse framework for cooperative object detection. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 22296–22305, 2025.

[50] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3836–3847, 2023.

[51] Binyu Zhao, Wei Zhang, and Zhaonian Zou. Bm2cp: Efficient collaborative perception with lidar-camera modalities. *arXiv preprint arXiv:2310.14702*, 2023.

[52] Seth Z Zhao, Huizhi Zhang, Zhaowei Li, Juntong Peng, Anthony Chui, Zewei Zhou, Zonglin Meng, Hao Xiang, Zhiyu Huang, Fujia Wang, et al. Quantv2x: A fully quantized multi-agent system for cooperative perception. *arXiv preprint arXiv:2509.03704*, 2025.

[53] Junfei Zhou, Penglin Dai, Quanmin Wei, Bingyi Liu, Xiao Wu, and Jianping Wang. Pragmatic heterogeneous collaborative perception via generative communication mechanism. *arXiv preprint arXiv:2510.19618*, 2025.

A Experimental Protocol and Implementation Details

This appendix follows the reading path of the main paper. Appendix A documents the datasets, packet-loss protocol, and implementation details used in Sec. 4.1. Appendix B extends the robustness discussion in Sec. 4.2. Appendix C expands the ablations summarized in Sec. 4.4. Appendix D clarifies two method choices from Sec. 3.1, and Appendix E states the experimental scope and future extensions.

A.1 Datasets and Metrics

We evaluate LR-V2X on two real-world collaborative perception benchmarks for LiDAR-based cooperative 3D object detection: DAIR-V2X [48] and V2XREAL [40]. DAIR-V2X represents the vehicle-to-infrastructure setting, while V2XREAL provides real-world vehicle-side collaborative driving scenes and is evaluated here in the LiDAR-only V2V setting used by OpenCOOD-style baselines. We follow the standard benchmark training and evaluation protocol for each dataset and preserve the native metric breakdown of each benchmark in the main tables: DAIR-V2X is reported with range-based AP, and V2XREAL is reported with class-based AP. Unless otherwise stated, all compared methods use the same PointPillar backbone so that the experiments isolate the effect of communication and reconstruction under packet loss.

For DAIR-V2X, Short, Middle, and Long denote diagnostic AP values computed on objects within the corresponding distance ranges, while the overall mAP columns are computed from all evaluated detections across the full range and are not the arithmetic mean of the three range bins. For V2XREAL, the overall mAP is the mean of the unrounded AP values over the vehicle, pedestrian, and truck super-classes.

A.2 Packet-Loss Evaluation Protocol

Unless otherwise stated, all methods are trained under complete communication and evaluated under the same packet-loss simulation protocol. During testing, each spatial packet on the transmitted communication tensor is independently dropped with probability $1 - p$, where p is the retention rate. The main robustness setting uses $p = 0.1$ (90% packet loss), and Fig. 5 sweeps the corresponding packet-loss rate from 0% to 90%. The ego feature is not transmitted and is always available to the ego-side fusion module. Evaluation uses fixed random seeds for mask generation and inference noise so that methods are compared under the same stochastic protocol.

A.3 Implementation Details

Our experiments are implemented in the OpenCOOD [43] framework. We use PointPillar [19] as the encoder with a voxel size of $(0.4\text{m}, 0.4\text{m})$ and a maximum of 32 points per pillar over the LiDAR range $[-102.4, -51.2, -3.5, 102.4, 51.2, 1.5]$m. After PointPillars and backbone downsampling, the BEV feature tensor used by the latent encoder has spatial resolution 128×256 with 64 channels. All methods use the same evaluation protocol with fixed packet-loss masks and inference seeds. Code, configurations, and reproduction instructions are available at `https://github.com/sidiangongyuan/LR-V2X`.

A.3.1 Training Pipeline

We adopt a three-stage training pipeline:

Stage 1: Baseline Detection. We train the PointPillar encoder, pyramid fusion backbone (ResNeXt with 3 scales: $[64, 128, 256]$ channels, upsample strides $[1, 2, 4]$), and detection head for 40 epochs with batch size 8. The optimizer is Adam with learning rate 2×10^{-3}, L2 regularization implemented via weight decay 10^{-4}, and a multistep LR schedule (decay $\times 0.1$ at epochs 15 and 30). The detection loss combines Sigmoid Focal Loss ($\alpha = 0.25$, $\gamma = 2.0$) for classification, Weighted Smooth L1 Loss ($\sigma = 3.0$) for regression, and pyramid losses at relative downsamples $[1, 2, 4]$ with weights $[0.4, 0.2, 0.1]$.

Stage 2: Reconstructor Training. We freeze the PointPillar sensor encoder, pyramid fusion backbone, and detection heads, and train the compact latent encoder together with the LPD and DiT denoiser for 100 epochs with batch size 12. The optimizer is Adam with learning rate 2×10^{-4},

L2 regularization implemented via weight decay 10^{-4}, and a multistep LR schedule (decay $\times 0.1$ at epochs 50 and 80). Only the x_0-prediction MSE loss is used in this stage.

Stage 3: End-to-End Fine-tuning. We jointly fine-tune the latent encoder, LPD, DiT denoiser, pyramid fusion module, and detection heads while keeping the PointPillar sensor encoder/backbone frozen. This stage runs for 20 epochs with batch size 8. The total loss is $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{det}} + 0.1 \cdot \mathcal{L}_{\text{diff}}$, using Adam with learning rate 2×10^{-4} and the same L2 regularization setting.

A.3.2 Architecture Details

Latent Encoder. The encoder compresses BEV features from $128 \times 256 \times 64$ to a target spatial resolution of 16×32 with 64 latent channels (default $\times 8$ downsampling), yielding a 128 KiB message per transmitted collaborator frame.

Communication Accounting. For compact latent communication, the default payload contains $16 \times 32 \times 64$ float32 values. Dense BEV feature fusion transmits the corresponding $128 \times 256 \times 64$ float32 tensor, giving 8 MiB. Early fusion sends raw point clouds, so we estimate its payload from validation files under float32 XYZI storage: about 1.0 MB on DAIR-V2X and about 2.4 MB on V2XREAL. Late fusion sends post-NMS detections; using the implementation cap of 150 boxes and a compact 7-DoF box, score, and class payload gives an upper bound of 5.4 KB.

Latent Prior Decoder (LPD). The LPD consists of four transposed convolutional layers that upsample from latent resolution $(16, 32)$ back to BEV resolution $(128, 256)$. Overlapping kernels enable spatial propagation from received locations into missing regions.

DiT Denoiser. We use the DiT-B configuration: 12 transformer layers, hidden dimension 768, 12 attention heads, MLP ratio 4.0, and patch size 4. Cross-attention layers attend to concatenated tokens from the received latent and aligned ego BEV. Gradient checkpointing is enabled to reduce memory usage.

A.3.3 Noise Schedule and Inference Configuration

We use a cosine noise schedule with 1000 timesteps ($\beta_{\text{start}} = 10^{-4}$, $\beta_{\text{end}} = 0.02$). At inference, we sample Gaussian noise for the LPD prior, perturb it to a fixed inference timestep $t_{\text{inf}} = 700$, and perform a single DDIM-style [32] denoising forward pass from the noise-augmented prior to the reconstructed BEV feature. We keep this timestep fixed for all datasets and packet-loss rates, so the reported results do not rely on condition-specific tuning.

B Robustness Analyses Beyond the Main Sweep

This section complements Secs. 4.2 and 4.3 with spatial burst-loss results, temporal protocol details, and bandwidth-matched comparisons.

B.1 Correlated Burst Loss

To assess robustness beyond i.i.d. packet erasures, we evaluate correlated spatial burst loss on DAIR-V2X under 90% packet loss. Figure 7 illustrates scattered erasures and contiguous missing patches. For each transmitted communication feature map, we sample a coarse 8×16 Bernoulli keep mask with retention rate $p = 0.1$, then upsample it with nearest-neighbor interpolation to the native communication resolution. Each coarse erasure therefore removes a contiguous spatial patch. The pattern is normalized across methods with different communication resolutions. The ego feature is always kept, and the same per-sample coarse mask is reused across methods.

Table 3 shows that the relative trend between low-bandwidth methods is preserved under burst loss. At the same 128 KiB budget, LR-V2X outperforms CodeFilling across all overall mAP thresholds (+2.47/+1.59/+0.45 at IoU 0.3/0.5/0.7) and achieves the best short-range AP at all three thresholds. F-Cooper attains the highest overall mAP in this setting, consistent with its much larger 8 MiB communication budget. These results support the benefit of compact-message reconstruction under spatially correlated erasures.

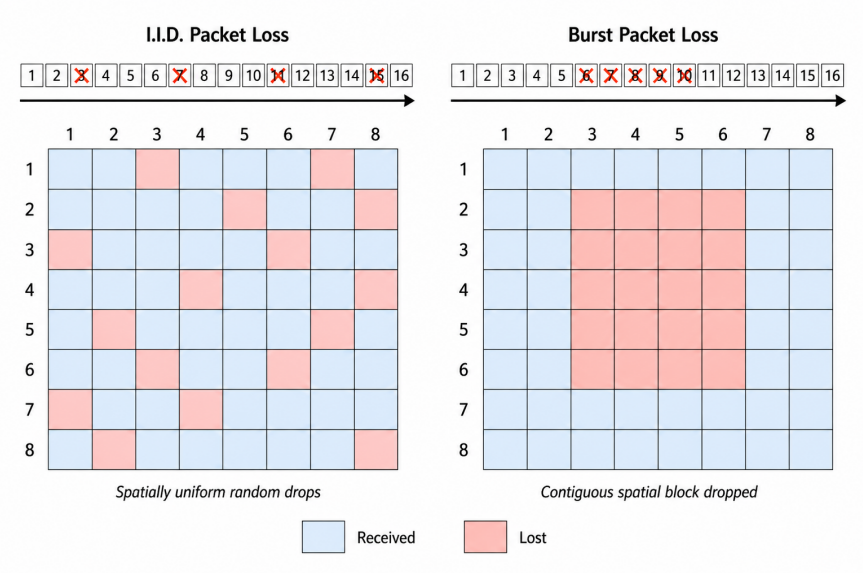


Figure 7: Illustration of i.i.d. packet loss (left) vs. burst packet loss (right) on an 8×8 BEV spatial grid. Blue cells are received; red cells are lost. Under i.i.d. loss, dropped packets are spatially scattered at random. Under burst loss, consecutive packets are dropped together, producing a contiguous missing region.

Table 3: Burst-loss results on DAIR-V2X under 90% packet loss. F-Cooper is a dense BEV feature fusion baseline with 8 MiB communication, while CodeFilling and LR-V2X both operate at 128 KiB per transmitted collaborator frame.

Models	Short			Middle			Long			mAP (%)			Comm
	0.3	0.5	0.7	0.3	0.5	0.7	0.3	0.5	0.7	0.3	0.5	0.7	
F-Cooper	79.15	76.03	65.84	**76.34**	**71.05**	**56.56**	**55.91**	**50.68**	**37.78**	**69.88**	**65.08**	**52.12**	8MiB
CodeFilling	78.02	76.29	66.81	71.36	68.42	55.72	50.26	47.19	36.63	65.36	62.62	51.48	128KiB
LR-V2X	**79.96**	**77.53**	**66.97**	74.15	70.22	56.38	52.70	48.75	36.99	67.83	64.21	51.93	128KiB

B.2 Temporally Correlated Packet Loss

Section 4.3 and Table 2 evaluate temporal persistence on DAIR-V2X at 90% marginal loss. The same non-ego packet mask is reused for K consecutive frames within a sequence and reset at sequence boundaries. The surviving collaborator values and ego observation are updated every frame. The marginal loss rate and packetization stay fixed within each method, so K controls only the duration of the missing locations. Results report mean and standard deviation over three packet-mask seeds with fixed trained models.

Holding a partial-erasure mask fixed tests persistent missing locations. A complete outage removes all current collaborator values, and communication delay introduces stale observations. These settings require separate outage-duration and temporal-alignment protocols and remain future extensions.

B.3 Bandwidth–Robustness Analysis

A recurring observation in our experiments is that dense BEV feature fusion baselines (8 MiB per transmitted collaborator frame) remain comparatively stable under packet loss. This behavior is closely tied to absolute information retention: under the same retention rate, a much larger communication budget leaves substantially more surviving information after transmission.

Information retention under packet loss. Let B denote the pre-loss payload per transmitted collaborator frame and p the packet retention rate. The expected retained payload is $B \cdot p$. Table 4 compares this quantity across representative message sizes at 90% packet loss ($p = 0.1$):

Table 4: Expected retained payload under 90% packet loss at different message sizes.

Method class	**Payload B**	**Expected retained**	**Ratio**
Dense BEV feature fusion (e.g., F-Cooper)	8 MiB	819.2 KiB	$64\times$
Medium compression ($\times 2$)	2 MiB	204.8 KiB	$16\times$
Our default ($\times 8$)	128 KiB	12.8 KiB	$1\times$
Aggressive ($\times 16$)	32 KiB	3.2 KiB	$0.25\times$

Table 5: Budget-matched comparison on DAIR-V2X under 90% packet loss. Dense BEV feature fusion baselines are contrasted between their native dense BEV setting (left) and the protocol-matched 128 KiB setting (right). LR-V2X is shown once as the native 128 KiB reference.

Method	**Dense Native (8 MiB)**				**Budget-Matched (128 KiB)**				
	mAP@0.3	mAP@0.5	mAP@0.7	FPS	mAP@0.3	mAP@0.5	mAP@0.7	FPS	Δ mAP@0.5
Attn	66.87	66.20	52.56	6.67	62.99	60.25	48.65	6.91	-5.95
F-Cooper	67.89	65.90	52.87	5.39	63.24	61.67	48.67	9.30	-4.23
CoBEVT	68.83	63.55	44.47	4.92	67.73	62.85	52.99	5.30	-0.70
V2X-VIT	68.78	64.33	50.97	4.70	64.69	60.53	49.21	5.32	-3.80
HEAL	64.43	61.68	50.50	7.24	60.81	58.08	47.25	10.00	-3.60
LR-V2X	—				69.35	66.90	53.37	6.04	—

Dense-native values are copied from Table 1b under 90% packet loss. The 128 KiB block reports equalized-budget re-evaluation, and Δ mAP@0.5 is defined as mAP@0.5(128 KiB) $-$ mAP@0.5(native dense setting). LR-V2X is shown only in the 128 KiB block because its native communication budget is already 128 KiB.

At 90% loss, a dense 8 MiB message retains 819.2 KiB on average. This is $64\times$ the expected retained payload of a 128 KiB message and $6.4\times$ a complete 128 KiB transmission. Table 5 controls this difference by re-evaluating the dense baselines at the same 128 KiB budget as LR-V2X.

Connecting to experimental results. This analysis helps interpret the trends in Table 1. The larger expected retained payload helps explain the stability of dense baselines such as Attn, F-Cooper, and HEAL. At a common 128 KiB budget, all five baselines decrease in mAP@0.5, with Attn decreasing by 5.95 points and CoBEVT by 0.70 points. LR-V2X achieves the highest mAP among these matched-budget entries and exceeds CoBEVT by 1.62/4.05/0.38 points at mAP@0.3/0.5/0.7.

B.4 Additional Communication Baselines

Table 6 compares Where2comm [12] and DiscoNet [22] on DAIR-V2X under the same 90% Bernoulli packet-loss evaluation. Payload is measured before packet loss for one transmitted collaborator frame. Where2comm uses spatial-confidence selection and a communication mask. Its smaller operating point is calibrated to approximately 128 KiB using transmitted bytes on the evaluation split, without labels or AP, and the threshold is then fixed. DiscoNet's 128 KiB operating point is a separately trained configuration with $16\times$ downsampling along each spatial axis.

Table 6: Additional baselines on DAIR-V2X at 90% packet loss. Bold marks the best result among operating points near 128 KiB.

Method	Payload (KiB)	mAP@0.3	mAP@0.5	mAP@0.7
Where2comm	28,860.99	61.89	57.31	39.68
Where2comm	128.01	61.88	57.30	39.67
DiscoNet	32,768	72.64	65.32	52.63
DiscoNet	128	68.16	64.19	53.05
LR-V2X	128	**69.35**	**66.90**	**53.37**

At approximately 128 KiB, LR-V2X exceeds Where2comm by 7.47/9.60/13.70 points and DiscoNet by 1.19/2.71/0.32 points. The dense DiscoNet setting has higher mAP@0.3, while LR-V2X has higher mAP@0.5 and mAP@0.7 with $1/256$ of its payload. Where2comm's AP changes little between its two operating points. This is consistent with redundancy in the additional lower-confidence

cells retained by the larger message under severe loss. DiscoNet's mAP@0.7 also increases after compression. These observations show that accuracy depends on which evidence survives as well as on payload size.

C Additional Ablations and Qualitative Results

This section expands the ablations summarized in Sec. 4.4. Unless noted otherwise, these results use DAIR-V2X and the same backbone, training schedule, and packet-loss protocol as the main experiments.

C.1 Reconstructor Conditioning Inputs

Table 7: Ablation of reconstructor conditioning inputs on DAIR-V2X under complete communication (no packet loss). All variants reconstruct from the same latent-derived prior; the table only removes ego-BEV, transformation, and direct latent-conditioning pathways inside the reconstructor.

Condition			mAP (%)		
Ego feat.	Trans. mat.	Latent feat.	mAP@0.3	mAP@0.5	mAP@0.7
			67.28	63.71	52.01
✓			78.90	74.15	55.77
✓	✓		79.69	75.34	56.56
		✓	67.30	63.75	52.04
	✓	✓	68.30	64.74	53.04
✓		✓	79.52	75.29	56.72
✓	✓	✓	**81.03**	**76.16**	**57.74**

Table 7 shows that ego-side BEV context is the most important additional reconstructor condition under complete communication. Every row reconstructs from the same collaborator-derived prior; the first row tests refinement with the additional conditioning pathways removed. Adding aligned ego BEV gives the largest gain, consistent with the ego feature supplying geometric anchors in overlapping regions and reducing ambiguity during reconstruction.

The transformation matrix and direct latent branch provide smaller but complementary refinements. The transformation contributes most when ego BEV is present because it tells the reconstructor how the ego context should be interpreted in the collaborator frame. The direct latent branch alone gives little gain because most collaborator evidence has already been decoded into the LPD prior, but combining all three conditions gives the highest mAP, suggesting that the branch still preserves residual compact cues that are useful after ego-guided refinement.

C.2 Latent Prior Decoder Ablation

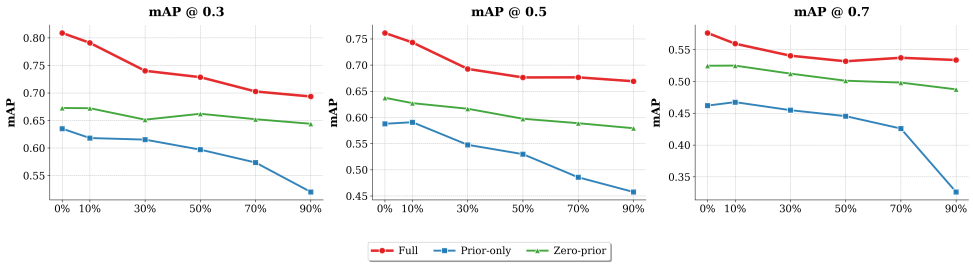


Figure 8: Ablation of the Latent Prior Decoder (LPD) on DAIR-V2X. We compare the full model, prior-only (using the LPD output directly and skipping the denoising reconstructor), and zero-prior (replacing the LPD initialization with an all-zero BEV prior before denoising) under varying packet-loss rates.

Fig. 8 separates three roles of the reconstruction module. The full model uses the LPD output as a coarse initialization and then applies the noise-conditioned reconstructor. The prior-only variant

removes this refinement stage and sends the LPD output directly to fusion, whereas the zero-prior variant keeps the denoising reconstructor but removes the latent-derived initialization by starting from an all-zero BEV prior. The full model remains consistently above both alternatives across packet-loss rates and IoU thresholds, showing that the two stages address different parts of the recovery problem.

The prior-only variant degrades most rapidly as packet loss increases. The LPD is a deterministic decoder, so its output depends on the limited spatial evidence remaining in the received latent. The zero-prior variant still uses ego context, direct latent conditioning, and learned scene regularities. Its lower performance than the full model shows the additional value of the collaborator-derived initialization. The LPD provides a spatially aligned starting point, and noise-conditioned refinement combines this prior with the available context.

C.3 Spatial Downsampling Ratio

Table 8: Effect of spatial downsampling ratio on DAIR-V2X.

Downsample	mAP@0.3	mAP@0.5	mAP@0.7	Comm
$\times 2$	82.55	77.64	59.42	2MiB
$\times 4$	81.52	76.79	58.52	512KiB
$\times 8$	81.03	76.16	57.74	128KiB
$\times 16$	79.58	73.51	54.52	32KiB

Table 8 shows that the spatial ratio controls both bandwidth and the amount of local structure retained in the latent prior. Moving from $\times 2$ to $\times 8$ reduces communication from 2 MiB to 128 KiB, a $16\times$ reduction, while the loss is only 1.52/1.48/1.68 mAP points at IoU 0.3/0.5/0.7. The default $\times 8$ setting therefore sits near the knee of the accuracy–bandwidth curve: it removes most of the communication load while retaining enough spatial detail for the LPD and denoiser to reconstruct object-level BEV evidence.

At $\times 16$, the mAP@0.7 drop is larger than the mAP@0.3 drop. This suggests that aggressive compression affects precise localization more strongly than coarse object presence. Each latent cell summarizes a larger BEV region, leaving fewer transmitted spatial cues for the LPD and denoiser to resolve object structure.

C.4 Noise-Conditioned Training and Inference Timestep

Table 9: Effect of noise-conditioned training on DAIR-V2X under 90% packet loss. The no-noise reconstructor keeps the same latent-derived prior and conditioning inputs but feeds the prior directly to the reconstructor at $t = 0$, without forward-noise perturbation.

Method	mAP@0.3	mAP@0.5	mAP@0.7
No-noise reconstructor	62.19	59.68	49.14
LR-V2X ($t_{\text{inf}} = 700$)	**69.35**	**66.90**	**53.37**

Table 9 isolates the value of noise-conditioned training under the main 90% packet-loss setting. The no-noise reconstructor is different from both prior-only and zero-prior in Fig. 8: it still uses the LPD prior and a learned reconstructor, but it removes the diffusion-style perturbation by feeding the prior directly at $t = 0$. Without this perturbation, the module is closer to an ordinary deterministic reconstructor. It learns to map a received latent prior to a target BEV feature, but it is not explicitly trained to calibrate how much of the prior should be trusted when large parts of the message are missing. Noise-conditioned training exposes the model to a continuum of corrupted priors, making the denoiser learn when to preserve the prior, when to use ego context, and when to fill missing collaborator evidence from learned spatial regularities.

Fig. 9 further explains why the diffusion-style formulation is useful even with a single inference step. Low inference timesteps keep the input close to the LPD prior and may leave packet-loss artifacts under-corrected; very high timesteps inject more uncertainty and can weaken transmitted evidence.

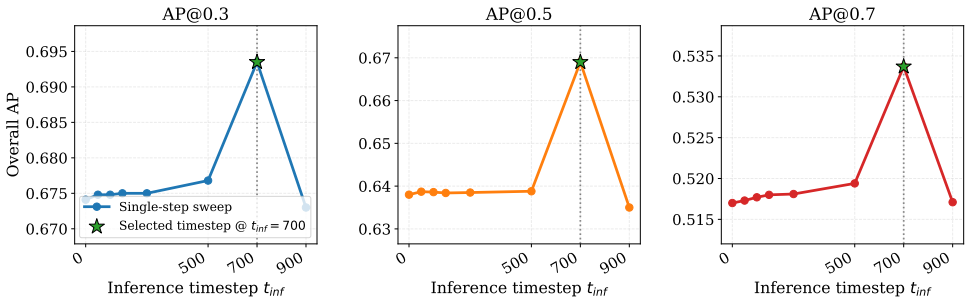


Figure 9: Sensitivity to the single-step inference timestep t_{inf} on DAIR-V2X under 90% packet loss. Solid curves show sweep results from a fixed checkpoint across different t_{inf} values, and the star marks the selected timestep $t_{\text{inf}} = 700$. The trend is broadly stable across timesteps, indicating that the single-step denoiser is not highly sensitive to the exact inference timestep within a wide operating range.

The selected $t_{\text{inf}} = 700$ performs best in this sweep, while neighboring settings remain broadly stable, indicating that the method is not relying on a narrow condition-specific timestep.

C.5 Qualitative Reconstruction

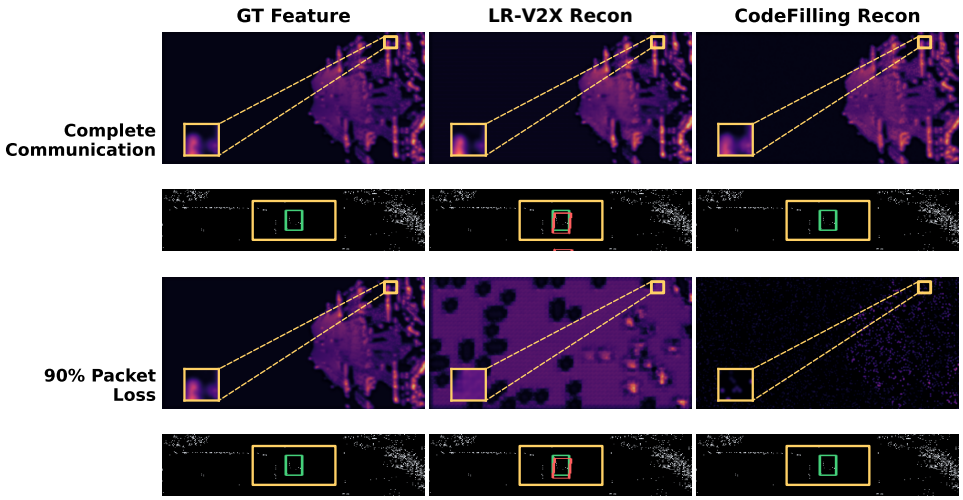


Figure 10: Detection-guided qualitative comparison between LR-V2X and CodeFilling on a fixed DAIR-V2X sample. The same target region is highlighted across complete communication and 90% packet loss. Under severe loss, LR-V2X reconstructs a clearer object-centric BEV response and preserves the correct downstream detection, while CodeFilling misses the same target.

Fig. 10 provides a qualitative example consistent with the quantitative packet-loss results. Under complete communication, both reconstructions retain a visible response around the highlighted target. Under 90% packet loss, CodeFilling produces a weaker and noisier response in this region, and the downstream detector misses the object. LR-V2X retains a more coherent object-centric activation that aligns with the ground-truth feature and preserves the detection.

This example illustrates the high-IoU trend in Table 1. Packet loss removes spatially localized evidence, and feature filling can leave holes or diffuse responses that weaken box regression. The LPD supplies a coarse prior from surviving packets, while the denoising reconstructor refines it using ego context. The resulting feature remains useful to the detection head in this example.

D Clarifications on Design Choices

This section provides brief technical clarifications for two design choices used in LR-V2X.

D.1 Single-Step Reconstruction vs. Iterative Sampling

LR-V2X uses a single DDIM-style step at inference, initialized from the LPD prior BEV map X_i^{prior}. The denoiser is trained with an x_0-prediction objective over noisy versions of this prior. At inference, Gaussian noise perturbs the prior to $t_{\text{inf}} = 700$, and one forward pass predicts the reconstructed collaborator BEV. Fig. 9 shows broadly stable performance across a wide range of timesteps, supporting this single-step operating choice.

D.2 Role of the Ego BEV Condition

The ego vehicle's own BEV feature serves as complementary spatial context for reconstruction. For collaborator i, the ego feature is first warped into the collaborator coordinate frame as $X_0^{\text{aligned}} = \mathcal{W}(X_0, T_{0\to i})$, matching the reconstruction formulation in Sec. 3.1. This aligned ego feature provides geometric anchors in overlapping regions and broader scene context around missing collaborator features. After reconstruction, the collaborator feature is warped back to the ego frame before fusion. Table 7 supports this role: removing the ego BEV condition produces the largest performance drop among the conditioning ablations.

D.3 Runtime, Memory, and Communication Time

We profile LR-V2X on one RTX 4090 with batch size 1 and two agents. After 20 warmup frames, we time 100 frames in each of three repeats. Forward and module times use synchronized CUDA events; post-processing uses synchronized wall-clock timing. Table 10 reports the resulting means and standard deviations where available. The model contains 179.15 million parameters and uses 1398.94 ± 0.76 MiB of peak allocated GPU memory.

Table 10: Compute profile on DAIR-V2X at 90% packet loss. The LPD and DiT run at the receiver; latent encoding runs at the sender. Summed rows use unrounded means.

Component	Time (ms)
Local encoder	13.75
Sender latent encoder	2.64
Latent prior decoder (LPD)	1.64
Single-step DiT	45.80
Fusion and detection head	16.39
Receiver reconstruction (LPD + DiT)	47.44
Latent encoding + receiver reconstruction	50.08 ± 0.41
Total forward	84.79 ± 3.24
Post-processing	46.42 ± 3.10
Total compute (sum of means)	131.20

Receiver reconstruction accounts for about 56% of forward time, with most of the cost coming from the single-step DiT. The LPD takes 1.64 ms. The 6.04 FPS in Table 1 is averaged over the full evaluation loop, while this profile uses a warmed fixed subset. Their timing protocols explain the difference between the two summaries.

For a payload of B bytes and a link rate of R bits/s, serialization takes

$$t_{\text{ser}} = \frac{8B}{R}. \tag{14}$$

Sending 128 KiB takes 104.9 ms at 10 Mbps and 10.5 ms at 100 Mbps. Adding these estimates to the measured 131.20 ms compute total under a sequential compute-plus-serialization model gives 236.1 ms and 141.7 ms, respectively. These calculations isolate compute and payload serialization; network-stack and queueing costs depend on the network implementation.

E Experimental Scope and Extensions

The real-world evaluation uses two-party protocols: one vehicle and one roadside unit on DAIR-V2X, and two vehicles in the selected V2XREAL V2V setting. The reconstruction interface reuses the same latent encoder, LPD, and DiT for each collaborator and applies the corresponding pairwise transform. With M non-ego collaborators, learned parameter count stays fixed, inbound payload is $128M$ KiB/frame, and the receiver makes M reconstruction calls. For five total agents, this gives 512 KiB/frame and four calls. This is an architectural resource accounting; multi-agent accuracy and latency also depend on overlap, visibility, and execution scheduling.

We will extend accuracy and latency evaluation to UrbanIng-V2X [2], which includes two vehicles and up to three infrastructure sensor poles. Load-aware network simulation will complement this study by modeling how channel access and competing traffic affect loss, delay, and throughput. Complete outages and stale messages require dedicated sequence-level protocols. The loss-rate sweep, spatial-burst test, and temporal-persistence test provide complementary evidence for the reconstruction problem studied here. Lightweight denoisers, distillation, and adaptive invocation offer directions for reducing receiver computation. Extending reconstruction to forecasting, mapping, occupancy prediction, and planning will require task-specific supervision and evaluation.

Societal impact. More reliable collaboration can improve the availability of surrounding-scene information when communication is degraded. Reconstructed features can still contain errors, so any use in safety-critical driving should include uncertainty monitoring and independent system-level validation.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state the packet-loss/low-bandwidth setting, the reconstruction-based contribution, the complete-communication training protocol, and the experimentally supported robustness claims. The strongest claims are tied to the 90% packet-loss tables and the 0–90% packet-loss sweep.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Appendix E discusses the evaluated two-party protocols, multi-agent resource scaling, richer channel models, and task extensions. Appendix D.3 reports reconstruction cost and memory use.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: The paper is empirical and does not present theoretical theorems or formal proofs.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Section 4.1 and Appendix A provide datasets, backbone, packet-loss protocol, architecture, training stages, hyperparameters, and inference settings. The public repository provides code, configurations, and reproduction instructions: `https://github.com/sidiangongyuan/LR-V2X`.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The experiments use public DAIR-V2X and V2XREAL datasets. The public code repository provides installation, data preparation, training, and evaluation instructions: `https://github.com/sidiangongyuan/LR-V2X`.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 4.1 describes the datasets, metrics, packet-loss protocol, and evaluation setting; Appendix A gives training stages, optimizer settings, schedules, architecture dimensions, and inference timestep.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

Justification: The main comparison uses one trained model and fixed evaluation seeds per method. Table 2 reports mean and standard deviation over three packet-mask seeds, which characterize mask variation; it does not assess variability across independently trained models.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [No]

   Justification: Appendix A reports training schedules, and Appendix D.3 reports the profiling GPU, inference memory, and timing protocol. Per-run training wall-clock time and total project compute are not fully enumerated.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The work uses public autonomous-driving datasets with credit to their creators. It involves no new human-subject experiments or private data collection, and Appendix E discusses safety considerations.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.

- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Appendix E discusses improved scene-information availability and risks from reconstruction errors in safety-critical driving, together with system-level validation and uncertainty monitoring.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not release high-risk generative models, scraped datasets, or models such as pre-trained language/image generators that would require special release safeguards.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The paper cites DAIR-V2X, V2XREAL, OpenCOOD, PointPillars, and other reused methods. The public repository includes upstream license files and attribution notices; the datasets remain subject to their original access terms.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The released research code includes a README, dependency requirements, dataset preparation instructions, and training and evaluation configurations. The repository also records licensing and upstream attribution.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing experiments or research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or human-subject research, so IRB approval is not applicable.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core method does not use LLMs as a technical component.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.