2025 International Conference on Intelligent Computing
July 26-29, Ningbo, China
https://www.ic-icc.cn/2025/index.php

V2VSR: Keypoints Feature Fusion-Based Cooperative Perception Method Under Communication Delays

Junxi Chen, Xiangfeng Luo, Liyan Ma, and Xue Chen(✉)

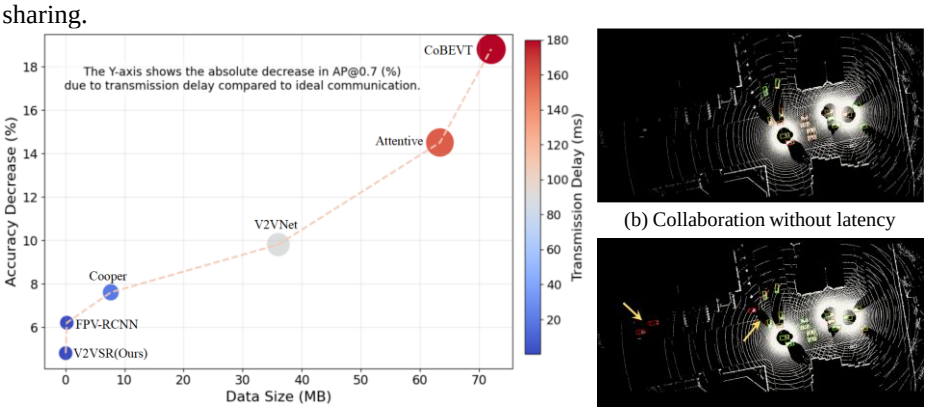
School of Computer Engineering and Science, Shanghai University, Shanghai 200444, China
xuechen@shu.edu.cn

Abstract. Single-agent LiDAR-based perception has significantly progressed but remains constrained by factors such as sensor range and occlusion. Multi-agent point clouds cooperative perception leverages inter-agent communication to share sensory information, thereby enhancing the perception capabilities of individual agents. Existing methods often assume ideal communication conditions. However, In the real world, data transmission delays are inevitable, which can cause the central agent to receive inaccurate features, leading to significant misguidance in perception results. This paper proposes a novel framework for multi-agent point clouds cooperative perception to efficiently extract key features and reduce latency. Specifically, we introduce a Sparse-Residual PointPillar (SRPP) backbone, improving inference speed and receptive field, and a Pillar Set Abstract Module (PSM), which abstracts scenes into compact keypoint features, significantly reducing shared feature map size. Additionally, we employ an inter-agent attention module, leveraging the characteristic of the main agent's own feature map, which requires no transmission and thus has no latency, to correct potential feature distortions and mitigate the impact of partially unavoidable delays, thereby improving system robustness. Our method can significantly reduce the shared feature map size to less than 0.1 MB, approximately 40 times smaller than highly advanced approaches. Even with significantly reduced shared feature maps, our model still outperforms other methods under ideal communication conditions and demonstrates a substantial advantage under delayed communication scenarios, indicating that our method significantly enhances the perception system's performance and delay robustness.

Keywords: Cooperative perception, 3D object detection, Sparse-Residual convolution, Key Feature Selection, Transformer.

1 Introduction

Given the inherent limitations of single-agent perception capabilities, multi-agent point clouds cooperative perception in connected autonomous driving has attracted considerable interest from researchers in recent years [28-29]. Recent studies [4,9] have shown that cooperative perception among agents, as opposed to the perception of individual agents, can utilize inter-agent communication technologies to facilitate information sharing.



(a) Data Size vs Accuracy Difference

(b) Collaboration without latency

(c) Collaboration with 1s latency

Fig. 1. (a) shows how data size (MB) affects the accuracy decrease (%) from ideal to delayed communication, using FPV-RCNN [31], Cooper [3], V2VNet [22], Attentive [29], and CoBEVT [27] as benchmarks, all models are trained under delayed communication using the public OPV2V dataset [29]. The driving speed of the agent is approximately 20 km/h. (b) and (c) are collaborative three-dimensional detection, where red indicates detection and green represents ground truth. Some false positives or missed detections are highlighted with yellow arrows.

However, recent research predominantly assumes an ideal communication environment, overlooking the fact that in delayed communication environments, delayed feature maps received by the central agent can significantly mislead the perception system, potentially leading to catastrophic consequences [8].

To address this, we first investigate the impact of communication delays on model performance. As shown in Fig. 1(a), in a delayed environment, the increase in the size of the shared feature map exacerbates communication delays, leading to a significant degradation in model performance compared to an ideal communication environment. The main reason, as indicated in [8], is that differing delays among communication channels lead to significant temporal asynchrony. Delay issues can significantly hinder cooperative perception systems, often resulting in severe false detections and misdetections. Fig. 1(b,c) illustrates the presence of multiple false-positive bounding boxes in scenario (b), characterized by a 1-second delay, where the agent's driving speed is approximately 20 km/h. The delayed cooperative data reflects conditions from one second earlier, resulting in the detector generating output boxes that are significantly deviated. This observation leads us to propose a robust cooperative perception system designed

to efficiently extract key target features, thereby reducing the effects of communication delays.

Therefore, this paper examines the detrimental effects of communication delays on multi-agent point clouds cooperative perception among agents and subsequently introduces a new framework aimed at extracting key shared features and alleviating delays. The method comprises a backbone network utilizing improved 2D sparse residual convolutions based on PointPillars [7] (SRPP), a pillar set abstraction module (PSM) designed for the efficient integration of scene information to obtain key target features, and a V2V self-attention module (V2VSA) aimed at rectifying delay-affected cooperative feature maps. V2VSA integrates self-attention mechanisms for the central agent alongside inter-agent attention processes, incorporating multi-head attention to capture diverse feature information from multiple subspaces, thereby improving feature fusion.

This study makes several noteworthy contributions, including:

(1) We conduct the first investigation into the impact of data size and communication latency on multi-agent point clouds cooperative perception, proposing a novel key feature fusion network to enhance system robustness to latency significantly.

(2) We enhance the widely used feature extraction network PointPillar for the first time by introducing the SRPP backbone and the PSM module. This innovation effectively extracts key target features, drastically curtails feature map dimensionality, leading to improved communication efficiency.

(3) We propose a novel inter-agent attention-based feature fusion method that leverages the feature map from the central agent, which does not require communication, to modify the delayed collaborative feature maps using this lossless feature map, significantly improving robustness.

2 Related Works

2.1 Vehicle-to-Vehicle Cooperative Perception

Based on how information is shared, cooperative approaches are typically grouped into early, intermediate, and late fusion strategies.

Early Fusion. Cooper [3] and AUTOCAST [16] achieved superior performance by sharing raw point data between different vehicles to form a more complete perception point cloud. However, due to the fact that early fusion typically requires a large transmission bandwidth, this strategy faces challenges in meeting the real-time requirements of collaborative perception.

Late Fusion. [1,17,30] improved fine-grained detection results by integrating the individual 3D perception outputs of each vehicle. However, In some cases, late fusion could produce unexpected results due to predictions with large individual deviations. Therefore, to achieve a better performance-bandwidth balance, recent methods have focused on fusing shared vehicle feature maps in the intermediate stage.

Intermediate Fusion. V2VNet [22] utilizes a graph neural network to consolidate the LiDAR features obtained from each vehicle. Various methods, including V2X-ViT [28], OPV2V [29], and CoBEVT [27], present distinct feature fusion strategies grounded in the Vision Transformer architecture, aimed at enhancing feature interaction among vehicles and advancing cooperative perception performance. Despite their promising performance, these intermediate fusion methods fundamentally hinge on the premise of flawless communication. In practical scenarios, communication delays will result in position deviations in the feature maps received by the central vehicle. To mitigate this issue, the designed system selectively shares the most representative key features, thereby minimizing communication delay and its associated impact.

As mentioned above, early fusion demands excessive bandwidth, and late fusion is susceptible to individual misrecognition results. In contrast, intermediate fusion better aligns with the cooperative perception design principles of balancing transmission bandwidth and model performance. Therefore, we develop our model based on the fundamental approach of intermediate fusion.

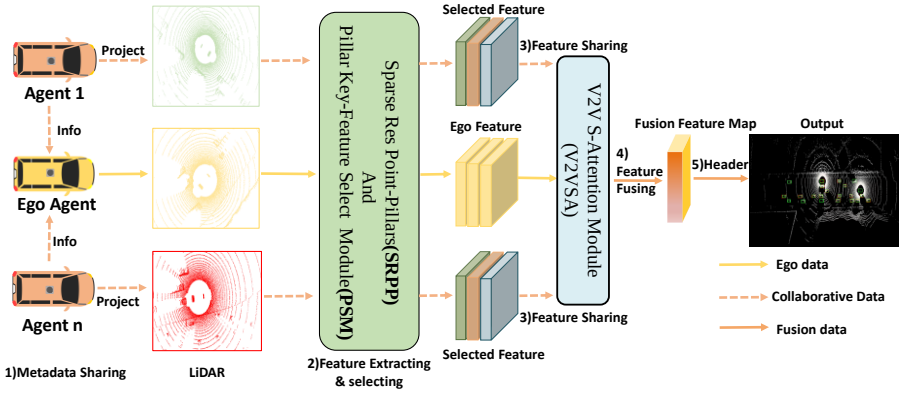


Fig. 2. The Architecture of the V2VSR. Our framework is composed of five key functional blocks: 1) Metadata Sharing, 2) Feature Extracting & Selecting, 3) Feature Sharing, 4) V2V Attention Feature Fusion, 5) Header. The specifics of each component are detailed in Section 3.1.

2.2 Delayed Communication in Vehicle-to-Vehicle Cooperative Perception

Most existing studies [14,24], use prediction-based time series methods to anticipate the deviation of feature maps caused by communication delays, and then reverse adjust the shared feature maps to correct the damaged ones. However, unlike predicting changes in feature maps [5,25], we noticed that since the vehicle's own feature map does not experience communication delays, we use this perfect feature map to adjust the collaborative feature maps with delays, effectively handling communication delays. Specifically, we designed a special V2V attention module called V2VSA, which includes both self-attention and inter-vehicle attention mechanisms, along with multi-head attention to enhance the optimization of collaborative feature maps.

3 Methods and Materials

3.1 Overview of architecture

This study acknowledges that all vehicles encounter communication delays during data transmission to simulate real-world scenarios. Our objective is to develop a robust fusion system that reduces the effects of communication delays in the collaboration process, thereby improving the cooperative perception capabilities of vehicles. To achieve this, we design a novel cooperative perception system focused on key target features and reduce transmission data by fusing only critical collaborative features, thereby effectively mitigating the impact of delays. Our proposed framework, as depicted in Fig. 2, consists of five primary components: 1) V2V metadata sharing, 2) Individual vehicle feature extracting and selecting, 3) Collaborative feature sharing, 4) V2V key feature fusion module, and 5) Detection head.

Metadata Sharing. We select one vehicle as the central vehicle (ego) to conduct communication-based collaborative perception around it. In this phase, all collaborative vehicles project their point clouds into the coordinate system of the central vehicle before feature extraction, performing an initial data alignment.

LiDAR Feature Extraction & LiDAR Feature Selection. Recent studies have shown that PointPillars [10,18] suffers from performance bottlenecks due to the lack of robust pillar feature encoding, which leads to poor performance in communication delay environments. Therefore, we have designed a real-time, high-performance target feature extraction method based on 2D sparse convolutions, called SRPP. Additionally, to further compress the data size, we have devised a feature selection module (PSM) to prioritize and transform critical features into shareable embeddings. Our proposed SRPP and PSM will be detailed in Sections 3.2 and 3.3, respectively.

Feature Sharing. At this stage, the central vehicle will receive feature maps from neighboring vehicles after feature extraction. However, in real-world scenarios, data transmission between vehicles takes time, and the feature maps received by the central vehicle from the collaborative vehicles inevitably exhibit positional discrepancies upon reception. Due to limited transmission bandwidth, the larger the feature maps to be transmitted, the greater the positional deviation. When the feature maps become too large, this can even result in significant performance degradation, as shown in 0.

V2V Attention Feature Fusion Module. Integrating delayed features directly into the central vehicle's attributes presents specific risks. We developed a V2V attention feature fusion module that leverages the optimal feature map of the center vehicle to address potential deficiencies in the collaborative feature map. Section 3.4 will provide a detailed description of the proposed V2V feature fusion module.

Detection Head. Two 1×1 convolutional layers are subsequently employed on the fused feature map to carry out bounding box regression and object classification.

3.2 Design of SRPP for 3D Collaborative Object Feature Extraction

To address the issue of PointPillar [7] lacking a larger receptive field and insufficient fine-grained information extraction, we redesigned the encoder and neck network by referencing the design ideas of [10,18].

Encoder Design. During the feature encoding phase, we use sparse 2D CNNs to progressively downsample the sparse pillar features. Compared to PointPillars [7], our encoder network, based on sparse 2D convolutions, allows us to leverage efficient 2D image object detection network structures, such as VGGNet [20] and ResNet [6], significantly enhancing 3D detection performance. In this study, we adopt a backbone structure similar to ResNet-18.

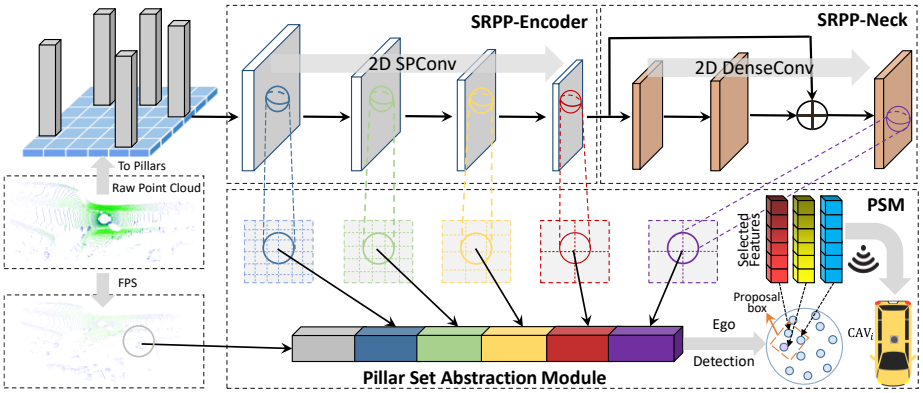


Fig. 3. Overview of the Proposed SRPP Architecture. The point cloud is voxelized into pillars and encoded via 2D sparse convolutions to extract multi-scale spatial features. These are fused with semantic cues in the neck, and distilled into compact key features for efficient sharing.

Neck Design. As in FPN [11], the neck serves to unify deep semantic representations with shallow spatial information, thereby producing more reliable base proposals B_i for the detection head. A set of dense 2D CNNs is employed to abstract high-level semantic features, thereby enhancing the receptive field for large objects. Additional convolutional layers enhance abstract semantic representations extracted from compact feature grids, integrating spatial and semantic features through further convolutional layers to facilitate deeper and more robust feature extraction.

3.3 Pillar Feature Abstraction Module for Key Feature Selection

Experimental results indicate that the key feature selection modules FPV-RCNN [31] and PV-RCNN [19] demonstrate superior performance in collaborative perception for autonomous driving. These modules aggregate multi-scale feature point clouds that represent the entire scene into a limited number of key points, thereby effectively reducing the volume of shared information. We developed a distinctive feature selection module, with the specifics of the PSM module illustrated in Fig. 3.

Initially, Furthest Points Sampling (FPS) is employed to select a specified number of uniformly distributed sparse key points. Then, features are further filtered based on these sampled key points. Specifically, we denote $F^{(p_k)} = \left\{f_1^{(p_k)}, \ldots, f_{N_k}^{(p_k)}\right\}$ as the set of semantic feature vectors of individual pillars in the \$k\$-th Pillar CNN, and $V^{(p_k)} = \left\{v_1^{(p_k)}, \ldots, v_{N_k}^{(p_k)}\right\}$ as the set of 3D coordinates calculated through pillar indices and actual pillar sizes, where $v_j^{(p_k)}$ represents the center position of the j-th pillar. For each key point p_i, we search for the neighboring non-empty pillars within radius r_k at the k-th level, forming the set $S_i^{(p_k)}$.

$$S_i^{(p_k)} = \left(v_j^{(p_k)}, f_j^{(p_k)}\right) \mid\mid v_j^{(p_k)} - p_i \mid\mid \leq r_k, j = 1, \ldots, N_k \tag{1}$$

For the neighboring pillars $S_i^{(p_k)}$ of key point p_i we concatenate the relative coordinates of each pillar's center point with respect to and its semantic feature vector to obtain the new feature vector:

$$z_j^{(p_k)} = concat\left(v_j^{(p_k)} - p_i, f_j^{(p_k)}\right), \forall \left(v_j^{(p_k)}, f_j^{(p_k)}\right) \in S_i^{(p_k)} \tag{2}$$

where $concat(\cdot)$ denotes the vector concatenation operation, $v_j^{(p_k)} - p_i$ is the relative position coordinate of the j-th pillar, and $f_j^{(p_k)}$ is the semantic feature of the j-th pillar.

Finally, through the PillarNetblock [18], the individual pillar features within the neighboring pillar set $S_i^{(p_k)}$ of p_i are subject to nonlinear transformation and aggregation to generate the global feature of the key point p_i:

$$h(p_i) = PillarNetBlock\left(z_j^{(p_k)} \mid \left(v_j^{(p_k)}, f_j^{(p_k)}\right) \in S_i^{(p_k)}\right) \tag{3}$$

Where $z_j^{(p_k)}$ represents the input features, and the $PillarNetBlock$ extracts high-level features for the key point.

After obtaining the basic key features of a single vehicle, we input these key features into the vehicle's perception detection head to generate the base pre-selection box B_i and further downsample these points by selecting only the key points within the pre-selection box B_i for generating shared key features.

The ablation results presented in 0 demonstrate that, leveraging the pillar set abstraction operation described earlier, the proposed PSM module significantly reduces data volume. This reduction alleviates communication latency and enhances recognition accuracy under delayed communication conditions, as further detailed in Section 4.4.

3.4 V2V Attention Module for Fusing Key Features in Collaboration

After receiving the key features from each collaborative vehicle, our objective is to optimize the feature map of the central vehicle using these collaborative key features. To achieve this, as shown in Fig. 4, we combine the intra-vehicle self-attention mechanism with the inter-vehicle attention mechanism. Furthermore, we leverage the multi-head attention mechanism to optimize and fuse the collaborative feature maps. By attending to diverse representation subspaces, the multi-head attention mechanism enriches the fusion process with complementary feature cues.

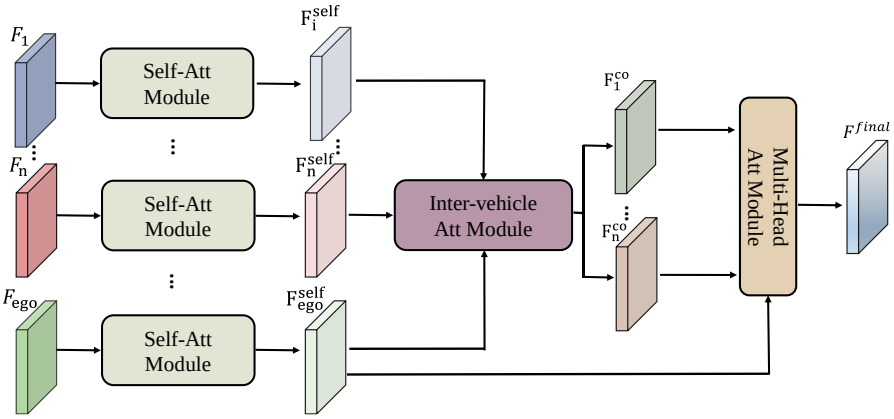


Fig. 4. The collaborative V2V attention module first optimizes the vehicle's feature map through the self-attention mechanism. After feature map sharing, the ego vehicle complements the collaborative features using its own lossless feature map. Finally, the optimized features are fused using a multi-head attention mechanism to generate a higher-quality optimized feature.

Vehicle Self-Attention Mechanism. For the feature map F_i each vehicle i, the query, key, and value are defined as:

$$Q_i = W_Q F_i,\ K_i = W_K F_i,\ V_i = W_V F_i \tag{4}$$

The weights of the self-attention mechanism for each vehicle i are calculated as:

$$\beta_i = softmax\left(\frac{Q_i K_i^{\mathrm{T}}}{\sqrt{d_k}}\right) \tag{5}$$

where Q_i, K_i, and d_k represent the query, key, and the dimension of the key, respectively.

The feature map of vehicle i optimized through the self-attention mechanism is:

$$F_i^{\mathrm{self}} = \beta_i V_i \tag{6}$$

where F_i^{self} represents the self-feature map of each vehicle after extracting local dependencies, resulting in higher quality.

Inter-vehicle Attention Mechanism. After being optimized through self-attention, collaborative vehicles send their respective feature maps F_i^{self} to ego vehicle, then ego

vehicle uses the inter-vehicle attention mechanism to complement potentially missing feature maps during communication. Its optimized feature map serves as the query, while the feature maps of other collaborating vehicles are used as keys and values. At this stage, the query originates from the optimized ego self-feature map F_{ego}^{self}, and the keys and values come from the feature maps of the collaborating vehicles, i.e.:

$$Q_{ego} = W_Q F_{ego}^{self\sim},\ K_i = W_K \tilde{F}_{ego}^{i},\ V_i = W_V \tilde{F}_{ego}^{i},\ \forall i \in C_{ego} \tag{7}$$

The attention weight of the ego vehicle for the feature map of the i-th vehicle is calculated as:

$$\alpha_{ego,i} = softmax_i\left(\frac{Q_{ego}K_i^{\top}}{\sqrt{d_k}}\right) \tag{8}$$

The completed collaborative feature map is obtained by weighted aggregation of the feature maps from other agents:

$$F_i^{co} = \sum_{i \in C_{ego}} \alpha_{ego,i} V_i \tag{9}$$

Multi-Head Attention Mechanism for Feature Fusion. To effectively fuse the optimized self-feature map $F_{ego}^{\text{self}} \in \mathbb{R}^{B\times N\times C\times H\times W}$ and multiple collaborative feature maps $F_i^{co} = \{F_1^{co}, F_2^{co}, \ldots, F_n^{co}\}$, where $F_i^{co} \in R^{B\times C\times H\times W}$, we employ a multi-head attention mechanism. To simplify processing, all collaborative feature maps are concatenated along the agent dimension to form a combined feature tensor $F_{ego}^{\text{co\_all}} \in \mathbb{R}^{B\times N\times C\times H\times W}$. The attention mechanism operates as follows:

First, F_{ego}^{self} and $F^{\text{co\_all}}$ are mapped into query (Q), key (K), and value (V) spaces:

$$Q = F_{ego}^{\text{self}} W_Q, \quad K = F^{\text{co\_all}} W_K, \quad V = F^{\text{co\_all}} W_V \tag{10}$$

where $W_Q, W_K, W_V \in \mathbb{R}^{C\times d_k}$ are learnable projection matrices, and d_k is the dimension of each attention head.

For each attention head, the scaled dot-product attention is computed to obtain the attention scores:

$$S_h = \text{Softmax}\left(\frac{QK^{\top}}{\sqrt{d_k}}\right) \tag{11}$$

where $S_h \in \mathbb{R}^{B\times H\times W\times N}$ represents the normalized attention weights for the h-th head. These weights are then used to aggregate the value features:

$$\text{Head}_h = S_h \cdot V \tag{12}$$

where $\text{Head}_h \in \mathbb{R}^{B\times d_k\times H\times W}$ is the output of the h-th head.

All head-specific outputs are aggregated through concatenation and subsequently mapped back to the initial feature space using the trainable projection W_O:

$$\text{MHA}(Q,K,V) = \text{Concat}(\text{Head}_1, \text{Head}_2, \dots, \text{Head}_H)W_O \tag{13}$$

where $W_O \in R^{H\cdot d_k\times C}$, and H is the total number of attention heads.

To integrate the fused collaborative features with the optimized self-feature map, a residual connection is applied:

$$F^{\text{final}} = F_{ego}^{\text{self}} + \text{MHA}(Q,K,V) \tag{14}$$

The final feature map $F^{\text{final}} \in R^{B\times C\times H\times W}$ retains the self-perception information from F_{ego}^{self} while incorporating contextual collaboration information from F_i^{co}.

4 Experiments

4.1 Datasets

Given the significant challenges and expenses associated with gathering collaborative perception data from multiple vehicles in real-world contexts, we assessed the method utilizing a simulation dataset derived from digital twins. The experiments utilized the publicly available vehicle-to-vehicle (V2V) collaborative perception dataset, OPV2V [29]. OPV2V is an extensive open-source simulation dataset designed for vehicle-to-vehicle perception. It comprises 8 digital towns from Carla Town [4] and the digital town of Culver City in Los Angeles, totaling 11464 frames and 232913 annotated 3D vehicle bounding boxes. Using OPV2V's normal settings, we use 3382 frames as the training set and 1920 frames as the validation set. For a fair comparison, we use 2170 frames from Carla Town and 594 frames from Culver City as the test sets for all methods.

4.2 Experiments Setup

Evaluation Metrics. We assess the performance of the framework by measuring the final 3D vehicle detection precision. In accordance with OPV2V [29], the evaluation range is established as $x \in [-140, 140]$ meters and $y \in [-40, 40]$ meters, encompassing all Cooperative Autonomous Vehicles (CAVs) within this spatial domain, with the number of CAVs varying from 1 to 5 in the experiments. Model performance is quantified via average precision (AP), detection performance is assessed using average precision at dual IoU benchmarks: 0.5 and 0.7.

Evaluation Details. This section evaluates the vehicle cooperative target detection model utilizing LiDAR technology. The model is primarily evaluated in two distinct scenarios: 1) Ideal communication, where data is transmitted without any delays; 2) Delayed communication, where intermediate features from cooperative vehicles experience communication delays. To simulate delayed communication, we assume that the

waiting time between vehicle A and vehicle B is $T_{A\to B}$, the size of the transmitted feature map is $S_{A\to B}$, the communication bandwidth is B_w, the minimum transmission time is T_{min}, and we assume the random interference time follows a truncated normal distribution ϵ. The total waiting time can then be represented as:

$$\epsilon \sim TruncNorm(\mu, \sigma, a, b) \tag{15}$$

$$T_{A\to B} = \frac{S_{A\to B}}{B_w} + \epsilon \tag{16}$$

Where $T_{A\to B}$ represents the total transmission time from vehicle A to vehicle B, $S_{A\to B}$ represents the size of the feature map transmitted from vehicle A to vehicle B, B_w represents the fixed transmission bandwidth, and ϵ represents the random interference time, which follows a truncated normal distribution. To simulate a highly disturbed or severely fluctuating network environment, we set the communication bandwidth to 100 Mbps, the expected value μ of the interference time to 10 ms, the standard deviation σ to 20 ms, and the upper and lower bounds of the interference time to $[0,200]$ ms.

Table 1. Comparison of 3D object detection performance across two testing sets based on the training of Scheme I in OPV2V. The transmission delay is the average delay calculated using Formula 15. A 'x' in the table represents the bandwidth requirement is too large, which can not be considered to employ in practice. Since there is no delay in ideal conditions, the data size in such scenarios is not shown.

Method	Type	Data Size (Mb)	Transmission Delay (ms)	Deafult Towns		Culver Citys	
				AP@0.5	AP@0.7	AP@0.5	AP@0.7
No Fusion	Ideal	-	0	0.683	0.613	0.553	0.469
	Latecey	0	10	0.683	0.613	0.553	0.469
OPV2V [29]	Ideal	-	0	0.865	0.787	0.868	0.745
	Latecey	126.8	1278	0.135	0.083	0.126	0.075
FPV-RCNN [31]	Ideal	-	0	0.876	0.763	0.865	0.745
	Latecey	0.24	12.4	0.293	0.158	0.283	0.165
CoBEVT [27]	Ideal	-	0	0.908	0.840	0.875	0.746
	Latecey	72.08	730.8	0.155	0.091	0.154	0.083
V2X-ViT [28]	Ideal	-	0	0.839	0.729	0.873	0.718
	Latecey	3.84	48.4	0.164	0.109	0.145	0.113
V2VAM [9]	Ideal	-	0	0.918	0.857	0.881	0.779
	Latecey	x	x	0.083	0.019	0.079	0.023
V2VSR(Ours)	Ideal	-	0	**0.930**	**0.872**	**0.893**	**0.788**
	Latecey	**0.048**	**10.4**	0.325	0.188	0.305	0.179

In the training phase, two evaluation schemes are employed to investigate the effect of varying training datasets on the V2V 3D object detection performance. Scheme I uses only perfect data for training, whereas Scheme II incorporates the previously mentioned simulated communication delays. Both methodologies utilize identical training parameter configurations, we evaluate the trained models across the V2V Default CARLA Towns and Culver City test sets, analyzing performance in both ideal and delayed communication contexts. In the experiment, to facilitate a fair comparison, the voxel height and width resolution for all methods is standardized at 0.4m. We utilize the Adam optimizer [12], initializing the learning rate at 10^{-3}, which undergoes a 0.1 factor reduction every 10 epochs.We utilize the same hyperparameters as those in OPV2V.

Table 2. Comparison of 3D object detection performance across two testing sets based on the training of Scheme II in OPV2V.

Method	Type	Data Size (Mb)	Transmission Delay (ms)	Deafult Towns		Culver Citys	
				AP@0.5	AP@0.7	AP@0.5	AP@0.7
No Fusion	Ideal	-	0	0.683	0.613	0.553	0.469
	Latecey	0	10	0.683	0.613	0.553	0.469
OPV2V [29]	Ideal	-	0	0.785	0.622	0.764	0.634
	Latecey	126.8	1278	0.714	0.578	0.702	0.564
FPV-RCNN [31]	Ideal	-	0	0.846	0.721	0.831	0.725
	Latecey	0.24	12.4	0.815	0.702	0.812	0.693
CoBEVT [27]	Ideal	-	0	0.875	0.789	0.843	0.727
	Latecey	72.08	730.8	0.733	0.601	0.701	0.587
V2X-ViT [28]	Ideal	-	0	0.825	0.698	0.821	0.702
	Latecey	3.84	48.4	0.712	0.585	0.681	0.564
V2VAM [9]	Ideal	-	0	0.891	0.802	0.865	0.758
	Latecey	X	x	0.745	0.610	0.733	0.585
V2VSR(Ours)	Ideal	-	0	**0.902**	**0.815**	**0.875**	**0.769**
	Latecey	**0.048**	**10.4**	**0.865**	**0.758**	**0.867**	**0.721**

Comparison methods. The baseline approach is evaluated using only the LIDAR point cloud from the central vehicle, without incorporating any fusion strategies. This study assesses five advanced techniques utilizing intermediate fusion as the main fusion approach. CoBEVT [27], V2VAM [9], OPV2V [29], FPV-RCNN [31], and V2X-ViT [28] represent various models and frameworks in the field of vehicle communication and perception. To demonstrate the significant influence of high-latency communication, we first train these techniques in two scenarios: ideal communication and delayed communication. We then test these approaches in the same two scenarios to determine their effectiveness.

4.3 Experimental Results

0 presents a comparative analysis of the performance of all models trained using Scheme I, evaluated on two distinct data types: ideal scenarios and delayed communication. In optimal communication conditions, all cooperative perception methods demonstrate a marked improvement over the NO Fusion baseline. In the V2V Default CARLA Town test set, our proposed V2VSR surpasses the other five advanced fusion methods, attaining 93.0%/87.2% for AP@0.5/0.7, as indicated in bold in 0. In the V2V Culver City test set, V2VAM [9] attains 88.1%/77.9% for AP@0.5/0.7, whereas V2VSR demonstrates superior performance at 89.3%/78.8% for AP@0.5/0.7, representing an enhancement of 1.2%/0.9% over the second-best fusion method V2VAM.The results demonstrate that under ideal communication conditions, cooperative perception methods outperform single-vehicle perception systems in terms of perception performance, and the proposed V2VSR method effectively expands the model's receptive field to achieve the best perception performance. However, in high-latency communication test scenarios, the performance of all intermediate fusion methods sharply decreases on both test sets, with the accuracy of these methods even falling below NO Fusion.

Table 3. Component Ablation Study under delayed communication conditions based on the training of Scheme II in OPV2V. SRPP, V2VSA, and PSM respectively indicate the addition of 1) Sparse-Residual PointPillars Backbone Network, 2) Vehicle-to-Vehicle Self Attention Mechanism, and 3) Pillar Key Feature Selection Module.

SRPP	V2VSA	PSM	Data Size (Mb)	Transmission Delay (ms)	Deafult Towns		Culver Citys	
					AP@0.5	AP@0.7	AP@0.5	AP@0.7
			3.84	48.4	0.658	0.459	0.687	0.455
√			1.92	29.2	0.682	0.491	0.713	0.493
	√		3.84	48.4	0.726	0.597	0.785	0.576
		√	0.096	10.9	0.801	0.679	0.801	0.597
√	√		1.92	29.2	0.737	0.628	0.802	0.612
√		√	0.048	10.4	0.821	0.702	0.833	0.656
	√	√	0.096	10.9	0.843	0.732	0.848	0.689
√	√	√	**0.048**	**10.4**	**0.865**	**0.758**	**0.867**	**0.721**

In the V2V Default CARLA Town test set, the cooperative perception performance of OPV2V [29], FPV-RCNN [31], CoBEVT [27], V2X-ViT [28], and V2VAM [9] decreased by 73.0%, 58.3%, 75.3%, 67.5%, and 83.5% at AP@0.5, respectively. Notably,

the performance of V2VAM, which requires the largest data transmission and highest communication latency, deteriorates most significantly. Obviously, any intermediate fusion techniques that do not take delayed communication into consideration are not feasible for real-world implementation. The outcomes of 3D object detection on the two OPV2V test sets, utilizing Scheme II for training, are detailed in 0. Despite all intermediate fusion methods demonstrating superior performance compared to those in 0 due to their training with delayed communication, they remain inadequate in managing high-latency communication data, resulting in diminished perception performance.

In the V2V Default CARLA Town test set, OPV2V [29] achieves 57.8% for AP@0.7, FPV-RCNN [31] achieves 70.2%, CoBEVT [27] achieves 60.1%, V2X-ViT [28] achieves 58.5%, and V2VAM achieves 61.0%. Among them, OPV2V, CoBEVT, V2X-ViT, and V2VAM perform even worse than the single-vehicle baseline, NO Fusion. FPV-RCNN, due to the use of a key feature selection algorithm, achieves better results in comparison, highlighting the severe negative impact of high-latency communication. Our proposed method demonstrates performance metrics of 86.5%/75.8% for AP@0.5/0.7 in the V2V Default CARLA Town test set, and 86.7%/72.1% for AP@0.5/0.7 in the Culver City test set. The proposed method demonstrates superior performance in both ideal and high-latency communication scenarios, as indicated in 0. Our proposed V2VSR method effectively reduces communication latency, resulting in enhanced collaborative perception performance.

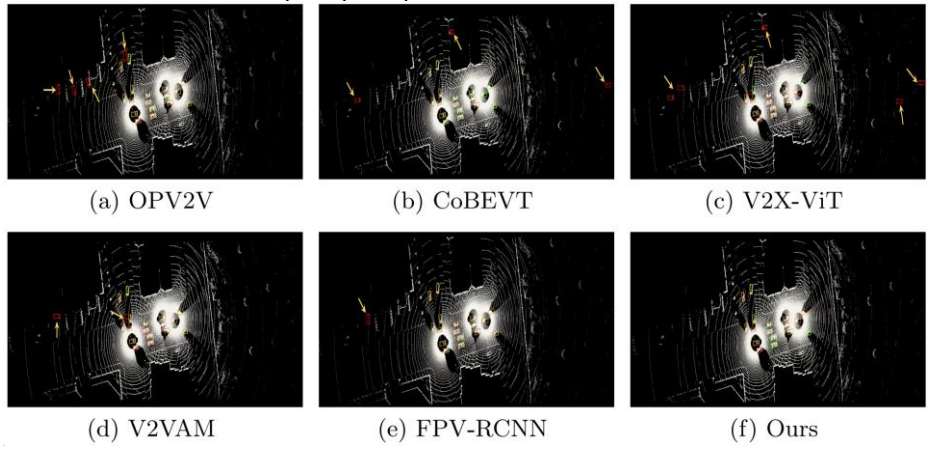


Fig. 5. A visual analysis of a crowded intersection, where ground truth is marked with green 3D boxes and model predictions are outlined in red. Our method produces more precise detection outcomes. Examples of false detections are indicated with yellow arrows.

4.4 Ablation Study

This study investigates the effectiveness of three components: SRPP, PSM, and V2VSA. All methods were assessed using the V2V CARLA Town and Culver City test sets. In the ablation experiments, the conventional PointPillar served as the baseline backbone network, while a 1×1 convolution was employed as the baseline fusion method to average all intermediate features. The impact of each component on the

model was evaluated by incrementally adding 1) SRPP, 2) V2VSA, and 3) PSM modules.As the experimental results in 0 show, all modules contribute to performance improvement. Under the condition of high communication latency, in the V2V Default CARLA Town test set, the V2VSA method outperforms the feature-averaging baseline method by 6.8% and 13.8% for AP@0.5 and AP@0.7, respectively, demonstrating its ability to efficiently fuse multi-agent features.The SRPP module compresses the original feature size from 3.84 Mb to 1.92 Mb, while the PSM module further reduces it to 0.096 Mb, indicating the effectiveness of both modules in significantly reducing data volume and alleviating communication latency. If the PSM module is removed from V2VSR or if SRPP is replaced with the conventional PointPillar, the performance decreases to varying degrees. Therefore, it is clear that due to the SRPP, PSM, and V2VSA components, the ultimate performance of 3D object detection is enhanced in high-latency communication circumstances.

4.5 Qualitative Analysis

Fig. 5 shows the detection visualization in the intersection street under communication delay scenarios for OPV2V [29], CoBEVT [27], V2X-ViT [28], V2VAM [9], FPV-RCNN [31], and V2VSR. Our model accurately predicts the bounding boxes, while other methods suffer from varying degrees of deviation in the detected target boxes due to their inability to effectively handle the misleading effects caused by delayed information. More importantly, V2VSR maintains good performance in a delayed environment (with no false detections or missed detections), demonstrating its ability to effectively cope with potential network fluctuations, reduce the amount of data to be transmitted, and decrease the time required for data transmission. This ensures a good balance between bandwidth and performance as well as high robustness in communication delay environments.

5 Conclusions

In this paper, we have conducted a study on the issue of communication delays in multi-agent point clouds cooperative perception systems. To mitigate the impact of communication delays, we have designed a novel model focused on key target features. By leveraging 2D sparse convolution and pillar set abstraction, we have developed a key feature extraction network that expands the model's receptive field while effectively extracting key target features and minimizing communication delays. Additionally, we have designed a specialized V2V attention module to address the remaining small yet unavoidable communication delays, incorporating self-attention, inter-agent attention, and multi-head attention mechanisms. Experimental results have demonstrated that our approach has effectively handled real-world delay issues and has enhanced model robustness.

Acknowledgments. This study was funded by National Natural Science Foundation of China (grant number 62421004).

References

1. Arnold, E., Dianati, M., de Temple, R., Fallah, S.: Cooperative perception for 3d object detection in driving scenarios using infrastructure sensors. IEEE Transactions on Intelligent Transportation Systems 23(3), 1852–1864 (2020)
2. Chen, Q., Ma, X., Tang, S., Guo, J., Yang, Q., Fu, S.: F-cooper: Feature based cooperative perception for autonomous vehicle edge computing system using 3d point clouds. In: Proceedings of the 4th ACM/IEEE Symposium on Edge Computing. pp. 88–100 (2019)
3. Chen, Q., Tang, S., Yang, Q., Fu, S.: Cooper: Cooperative perception for connected autonomous vehicles based on 3d point clouds. In: 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS). pp. 514–524. IEEE (2019)
4. Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., Koltun, V.: Carla: An open urban driving simulator. In: Conference on robot learning. pp. 1–16. PMLR (2017)
5. Escontrela, A., Adeniji, A., Yan, W., Jain, A., Peng, X.B., Goldberg, K., Lee, Y., Hafner, D., Abbeel, P.: Video prediction models as rewards for reinforcement learning. Advances in Neural Information Processing Systems 36 (2024)
6. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
7. Lang, A.H., Vora, S., Caesar, H., Zhou, L., Yang, J., Beijbom, O.: Pointpillars: Fast encoders for object detection from point clouds. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 12697–12705 (2019)
8. Lei, Z., Ren, S., Hu, Y., Zhang, W., Chen, S.: Latency-aware collaborative perception. In: European Conference on Computer Vision. pp. 316–332. Springer (2022)
9. Li, J., Xu, R., Liu, X., Ma, J., Chi, Z., Ma, J., Yu, H.: Learning for vehicle-to-vehicle cooperative perception under lossy communication. IEEE Transactions on Intelligent Vehicles 8(4), 2650–2660 (2023)
10. Li, J., Luo, C., Yang, X.: Pillarnext: Rethinking network designs for 3d object detection in lidar point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 17567–17576 (2023)
11. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2117–2125 (2017)
12. Loshchilov, I.: Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101 (2017)
13. Lu, Y., Li, Q., Liu, B., Dianati, M., Feng, C., Chen, S., Wang, Y.: Robust collaborative 3d object detection in presence of pose errors. In: 2023 IEEE International Conference on Robotics and Automation (ICRA). pp. 4812–4818. IEEE (2023)
14. Oliu, M., Selva, J., Escalera, S.: Folded recurrent neural networks for future video prediction. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 716–731 (2018)
15. Qiao, D., Zulkernine, F.: Adaptive feature fusion for cooperative perception using lidar point clouds. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision. pp. 1186–1195 (2023)

16. Qiu, H., Huang, P., Asavisanu, N., Liu, X., Psounis, K., Govindan, R.: Autocast: Scalable infrastructure-less cooperative perception for distributed collaborative driving. arXiv preprint arXiv:2112.14947 (2021)
17. Rawashdeh, Z.Y., Wang, Z.: Collaborative automated driving: A machine learning based method to enhance the accuracy of shared information. In: 2018 21st International Conference on Intelligent Transportation Systems (ITSC). pp. 3961–3966. IEEE (2018)
18. Shi, G., Li, R., Ma, C.: Pillarnet: Real-time and high-performance pillar-based 3d object detection. In: European Conference on Computer Vision. pp. 35–52. Springer (2022)
19. Shi, S., Guo, C., Jiang, L., Wang, Z., Shi, J., Wang, X., Li, H.: Pv-rcnn: Point-voxel feature set abstraction for 3d object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10529–10538 (2020)
20. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
21. Su, J., Byeon, W., Kossaifi, J., Huang, F., Kautz, J., Anandkumar, A.: Convolutional tensor-train lstm for spatio-temporal learning. Advances in Neural Information Processing Systems 33, 13714–13726 (2020)
22. Wang, T.H., Manivasagam, S., Liang, M., Yang, B., Zeng, W., Urtasun, R.: 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. pp. 605–621. Springer (2020)
23. Wang, Y., Gao, Z., Long, M., Wang, J., Philip, S.Y.: Predrnn++: Towards a resolution of the deep-in-time dilemma in spatiotemporal predictive learning. In: International conference on machine learning. pp. 5123–5132. PMLR (2018)
24. Wang, Y., Jiang, L., Yang, M.H., Li, L.J., Long, M., Fei-Fei, L.: Eidetic 3d lstm: A model for video prediction and beyond. In: International conference on learning representations (2018)
25. Wang, Y., Long, M., Wang, J., Gao, Z., Yu, P.S.: Predrnn: Recurrent neural networks for predictive learning using spatiotemporal lstms. Advances in neural information processing systems 30 (2017)
26. Wang, Y., Zhang, J., Zhu, H., Long, M., Wang, J., Yu, P.S.: Memory in memory: A predictive neural network for learning higher-order non-stationarity from spatiotemporal dynamics. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9154–9162 (2019)
27. Xu, R., Tu, Z., Xiang, H., Shao, W., Zhou, B., Ma, J.: Cobevt: Cooperative bird's eye view semantic segmentation with sparse transformers. arXiv preprint arXiv:2207.02202 (2022)
28. Xu, R., Xiang, H., Tu, Z., Xia, X., Yang, M.H., Ma, J.: V2x-vit: Vehicle-to everything cooperative perception with vision transformer. In: European conference on computer vision. pp. 107–124. Springer (2022)
29. Xu, R., Xiang, H., Xia, X., Han, X., Li, J., Ma, J.: Opv2v: An open benchmark dataset and fusion pipeline for perception with vehicle-to-vehicle communication. In: 2022 International Conference on Robotics and Automation (ICRA). pp. 2583–2589. IEEE (2022)
30. Yu, H., Luo, Y., Shu, M., Huo, Y., Yang, Z., Shi, Y., Guo, Z., Li, H., Hu, X., Yuan, J., 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. pp. 21361–21370 (2022)

31. Yuan, Y., Cheng, H., Sester, M.: Keypoints-based deep feature fusion for cooperative vehicle detection of autonomous driving. IEEE Robotics and Automation Letters 7(2), 3054–3061 (2022)
32. Zhang, Z., Hu, J., Cheng, W., Paudel, D., Yang, J.: Extdm: Distribution extrapolation diffusion model for video prediction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 19310–19320 (2024)