

Which2comm: Efficient collaborative perception framework with connected and automated vehicles

Duanrui Yu^{1,†}, Anqi Qu^{1,†}, Jing You^{1,†}, Dingyu Wang¹, Rongsong Li¹, Shaocheng Jia^{2,✉}, Xin Pei^{1,✉}

Cite this article: Yu D, Qu A, You J, et al. *Commun Transp Res* 2026, 6(3): 9640048. <https://doi.org/10.26599/COMMTR.2026.9640048>

ABSTRACT: Collaborative perception allowed real-time interagent information exchange and thus offered invaluable opportunities to enhance the perception capabilities of individual agents. However, limited communication bandwidth in practical scenarios restricted the interagent data transmission volume. This implied a trade-off between perception performance and communication cost. To address this issue, we proposed Which2comm, a novel multiagent three-dimensional (3D) object detection framework leveraging object-level sparse features. By integrating semantic information of objects into detection boxes, we introduced semantic detection boxes (SemDBs). Innovatively transmitting these object-level sparse features among agents not only significantly reduces the demanding communication volume but also improves object detection performance. Moreover, an adaptive strategy was further proposed to select only safety critical connected and automated vehicles (CAVs) for collaborative perception when there were multiple CAVs available, thereby maintaining stable communication costs. To validate the proposed method, a large-scale, multimodal dataset, Multi-V2X, is established for vehicle-to-everything perception tasks with various CAV penetration rates. Multi-V2X comprises 1.46×10^5 frames with over 4.2×10^6 3D annotations, featuring high agent density (up to 31 agents) to evaluate perception robustness in complex traffic environments. Extensive experiments demonstrate that Which2comm consistently outperforms other state-of-the-art methods on both detection performance and communication cost, exhibiting superior robustness to real-world latency.

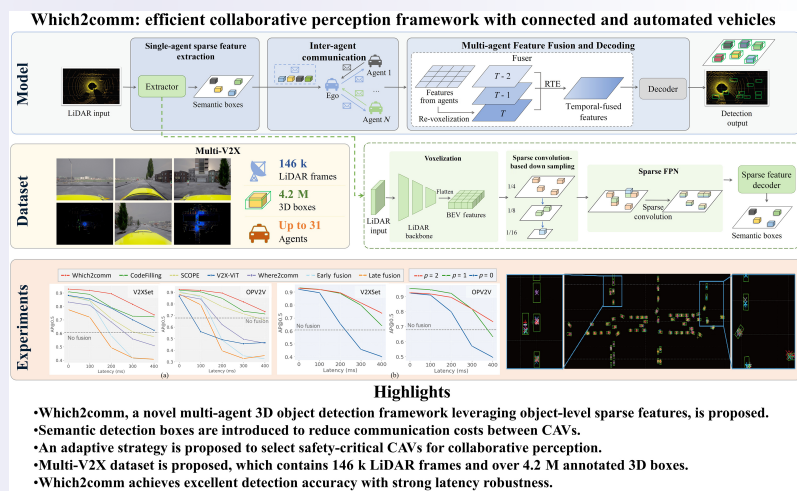
KEYWORDS: multiagent collaborative perception; three-dimensional (3D) object detection; traffic safety; vehicle-to-everything (V2X); communication efficiency

1 Introduction

Multiagent collaborative perception allows information exchange among multiple agents through interterminal communication, effectively mitigating the occlusion limitations inherent to individual agents (Hu et al., 2022, 2024; Ren et al., 2023; Wang et al., 2020). Thus, it has been widely adopted in various tasks to enhance perception performance (Lu et al., 2024; Wei et al., 2023; Yin et al., 2021), such as three-dimensional (3D) object detection (Hu et al., 2023; Liu et al., 2023; Yuan et al., 2022). In particular, vehicle-to-everything (V2X) technologies enable information

sharing between vehicles and roadside infrastructures, facilitating connected and automated vehicles (CAVs) to perceive information beyond the visual range and significantly improving traffic safety (Hu et al., 2022; Ren et al., 2023).

However, in real-world scenarios, the performance of collaborative perception systems is constrained by a range of realistic factors (Huang et al., 2025; Liu et al., 2023). For example, because collaborative perception is built upon interagent information exchange, communication cost must be considered in developing collaborative perception algorithms. Given that the communication bandwidth between agents is often limited in



† Duanrui Yu, Anqi Qu, and Jing You contributed equally to this work.

¹ Department of Automation, BNRist, Tsinghua University, Beijing 100084, China. ² Institute of Intelligent Transportation Systems, College of Civil Engineering and Architecture, Zhejiang University, Hangzhou 310058, China.

✉ Corresponding authors. E-mail: S. Jia, shaochengjia@zju.edu.cn; X. Pei, peixin@mail.tsinghua.edu.cn

Received: March 6, 2026; Revised: June 20, 2026; Accepted: August 17, 2026

© The Author(s) 2026. This is an open access article under the terms of the Creative Commons Attribution 4.0 International License (CC BY 4.0, <http://creativecommons.org/licenses/by/4.0/>).

practical scenarios (Filippi et al., 2017), it is challenging to transmit complete information among agents. Thus, collaborative perception performance is heavily degraded with low communication volumes (Liu et al., 2023). This represents an inevitable trade-off between perception performance and communication costs for collaborative perception methods (Hu et al., 2022, 2024; Ren et al., 2023; Xu et al., 2022a).

Current collaborative perception methods primarily use an intermediate fusion approach to achieve a balance between bandwidth and performance (Hu et al., 2022, 2024; Lu et al., 2023; Xu et al., 2022a; Yang et al., 2023). Such an approach is characterized by preserving intermediate network features with semantic information for transmission and reduces transmission cost via feature compression (Xu et al., 2022a), feature sparsification (Hu et al., 2022), or task-oriented feature selection (Gan et al., 2026; Shao et al., 2024). However, excessively compressed features frequently lead to a drastic drop in detection performance. Moreover, existing feature sparsification and task-oriented methods mainly operate at the regional level (Yuan et al., 2022), thereby retaining structural redundancies. This implies the potential of further exploration for more effective feature compression.

Furthermore, the CAV penetration rate is also an important factor for efficient collaborative perception. More specifically, a higher CAV penetration rate means more potential collaboration partners and thus better perception performance. However, the accompanying rise in communication demand increases communication costs. This may compromise both perception accuracy and robustness. To overcome scenarios with varying CAV penetration levels, a mechanism that can adaptively determine the number of collaboration agents for efficient perception is needed. In addition, a comprehensive collaborative perception dataset with various CAV penetration rates is also needed for training and validating models. However, the aforementioned research gaps, to the best of our knowledge, remain unexplored.

In this study, we propose Which2comm, a novel multiagent 3D object detection framework leveraging object-level sparse features. The sparsity of observation data is used to construct object-level sparse features for transmission, thereby reducing communication cost. Considering the advantages of late fusion in communication efficiency and intermediate fusion in precision and robustness, we propose combining their strengths by incorporating semantic information of objects into 3D detection boxes and establishing semantic detection boxes (SemDBs)—object-level sparse features for transmission and fusion between agents. As illustrated in Fig. 1, SemDBs can significantly reduce the feature transmission cost. Moreover, we incorporate temporal information to compensate for accuracy degradation caused by data transmission latency among agents. To accommodate sparse features, a relative temporal encoding (RTE) mechanism is introduced to explicitly distinguish features from different timestamps, thereby effectively integrating temporal information. Finally, a safety-aware adaptive collaborative perception mechanism is proposed to select collaboration agents according to the estimated safety risks. The mechanism prioritizes high-risk traffic participants and thus maximizes safety gains under resource-constrained conditions.

To validate the proposed method, we introduce Multi-V2X, a large-scale, multimodal, multipenetration-rate dataset for benchmarking V2X perception tasks. Simulation of urban mobility (SUMO) (Lopez et al., 2018) and cars learning to act (CARLA) (Dosovitskiy et al., 2017) are used to generate data, including cars, roadside units (RSUs), and various sensors. The

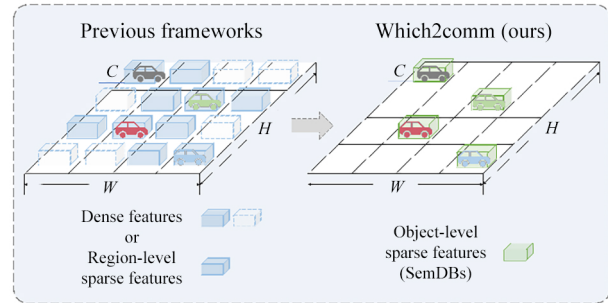


Fig. 1 Which2comm innovatively transmits object-level sparse features leveraging SemDBs, which significantly reduces the demanding communication volume. In Fig. 1, H , W , and C represent the height, width of detection scenario, and feature dimensionality, respectively.

CAV penetration rate can be adjusted by treating some equipped cars as conventional vehicles. The highest CAV penetration rate reaches 86.21%, under which 31 agents are within the communication range. Effectively choosing collaborators, therefore, becomes a new challenge in this context. Extensive experiments for collaborative 3D object detection are also conducted on Multi-V2X.

The main contributions of this study can be summarized as follows:

A novel multiagent 3D object detection framework, Which2comm, is proposed, which significantly reduces the interagent communication cost while maintaining excellent detection accuracy and robustness. This demonstrates that transmitting only object-level sparse features can achieve accurate and robust performance in multiagent collaborative 3D object detection.

An adaptive collaborative perception mechanism is proposed to dynamically select collaboration agents by evaluating safety risks, thereby establishing a unified framework for improving efficiency, accuracy, and safety.

A large-scale, multimodal, multipenetration-rate dataset, Multi-V2X, is introduced for benchmarking V2X perception tasks, paving the way for training, validating, and testing various collaborative perception models.

The rest of this paper is organized as follows: Section 2 reviews the related works. Sections 3 and 4 present Which2comm and the adaptive collaboration mechanism. Section 5 introduces the Multi-V2X dataset. Section 6 conducts comprehensive experiments to demonstrate the effectiveness of the proposed method. Section 7 concludes the study.

2 Related works

2.1 Collaborative perception

Multiagent collaborative perception addresses the occlusion issue of individual agents through information exchange between multiple agents using intelligent connected technologies. The emergence of various large-scale open datasets has promoted the development of perception frameworks (Xu et al., 2022a, 2022b; Yu et al., 2022), particularly for 3D object detection. These frameworks can be grouped into three categories according to the collaboration stage: early fusion that transmits raw sensor data between agents (Chen et al., 2019), intermediate fusion that transmits intermediate features of neural networks (Hu et al., 2022, 2024; Liu et al., 2020; Lu et al., 2023; Qiao and Zulkernine, 2023; Xu et al., 2022a; Yang et al., 2023), and late fusion that transmits detection outputs. Early fusion exhibits a significant advantage in detection precision but incurs high data transmission

costs. Late fusion performs better in terms of data transmission but presents poorer robustness against latency and localization errors. Intermediate fusion, serving as a good balance between accuracy and robustness, has attracted wide research attention. Current studies primarily focus on the bandwidth-performance trade-off (Hu et al., 2022, 2024; Liu et al., 2020; Yuan et al., 2022), interterminal latency mitigation (Wei et al., 2023), and localization error reduction. Using agent selection (Liu et al., 2020), spatial sparsification (Hu et al., 2022), feature compression encoding (Hu et al., 2024; Xu et al., 2022a; Yang et al., 2023), and task-oriented semantic communication (Gan et al., 2026; Shao et al., 2024), communication overhead can be reduced while improving overall performance under constrained communication conditions.

However, transmitting region-level sparse features, including recent task-oriented spatial masks, still includes redundant information associated with dense spatial grids. To address this, we propose Which2comm, leveraging the sparsity of LiDAR point cloud observations with a fully sparse network to obtain object-level sparse features for transmission and feature fusion. Unlike prior adaptive methods, such as When2com (Liu et al., 2020) and Where2comm (Hu et al., 2022), which use perception-centric collaboration mechanisms and retain redundant background voxels, Which2comm only transmits critical semantic information decoupled from dense grids. Moreover, we introduce a novel selection criterion based on the safety-centric driving safety index (DSI) to adaptively prioritize communication with high-risk agents. By integrating object-level sparse features and temporal information, our framework achieves a superior trade-off between perception accuracy and bandwidth efficiency.

2.2 Collaborative perception datasets

Recent studies have facilitated the development of synthetic (Li et al., 2022; Wang et al., 2024; Xu et al., 2022a, 2022b) and real-world (Hao et al., 2024; Xu et al., 2023; Yu et al., 2022) cooperative perception datasets. Given that generating synthetic datasets is more flexible, many works investigate synthetic datasets for collaborative perception research. V2XSet (Xu et al., 2022a) and open dataset for perception with V2X communication (OPV2V) (Xu et al., 2022b) are two commonly used collaborative perception datasets. The former is a large-scale simulation dataset specifically designed for V2X perception with CAVs and RSUs. The latter is a collaborative V2V perception dataset comprising over 70 scenes, providing a variety of collaborative perception tasks in different traffic scenarios. The following drawbacks of the existing datasets impede further development.

However, there are only limited categories in existing datasets; they predominantly include only cars, omitting vulnerable road

users such as cyclists and pedestrians. This implies that some important traffic participants cannot be accurately detected through training. Moreover, no existing collaborative perception dataset accounts for different CAV penetration rates, a realistic and important parameter for collaborative perception. The CAV penetration rate is defined as the ratio of the number of CAVs to the total number of vehicles on the road. In reality, both the number of CAVs and the total number of vehicles are highly dynamic over time, resulting in varying CAV penetration rates. Thus, various CAV penetration rates should be considered to simulate realistic traffic conditions. To this end, we propose a large-scale dataset, Multi-V2X, with variable CAV penetration rates for comprehensively benchmarking cooperative perception tasks.

3 Which2comm

This section presents Which2comm, a multiagent 3D object detection framework leveraging SemDB, which can take full advantage of the sparsity of point clouds to reduce transmission cost while enhancing detection performance. Section 3.1 first introduces the overall architecture, followed by detailed descriptions of the key modules in Sections 3.2 and 3.3. Finally, Section 3.4 presents the proposed shuffle-ego (SE) training strategy.

3.1 Overall architecture

Fig. 2 illustrates the structure of Which2comm, which comprises two main stages: single-agent sparse feature extraction and multiagent temporal feature fusion and decoding. During the feature extraction stage, a fully sparse feature extraction module, the extractor, is constructed. Each individual agent extracts SemDBs from its own observations. These sparse features are subsequently transmitted among terminals. During the multiagent perception stage, SemDBs are enriched with temporal encoding and are further fed to the Fuser and Decoder to obtain 3D object detection boxes.

Sparse feature extraction. Single-agent sparse feature extraction is applied to the input unstructured point cloud data, $\mathbf{X}_i \in \mathbb{R}^{n_p \times 4}$, to obtain sparse features, where n_p is the number of points and 4 represents the number of channels (i.e., 3D coordinates (x, y, z) and intensity).

Sparse feature transmission. Agents interact with each other through data transmission. During this process, the transmission cost from collaborator j to ego agent i for a single frame can be computed as

$$C_{j \rightarrow i}^{\text{tran}} = m_j \times (\dim(\mathbf{F}_j) + \dim(\mathbf{I}_j^{\text{spac}})) \quad (1)$$

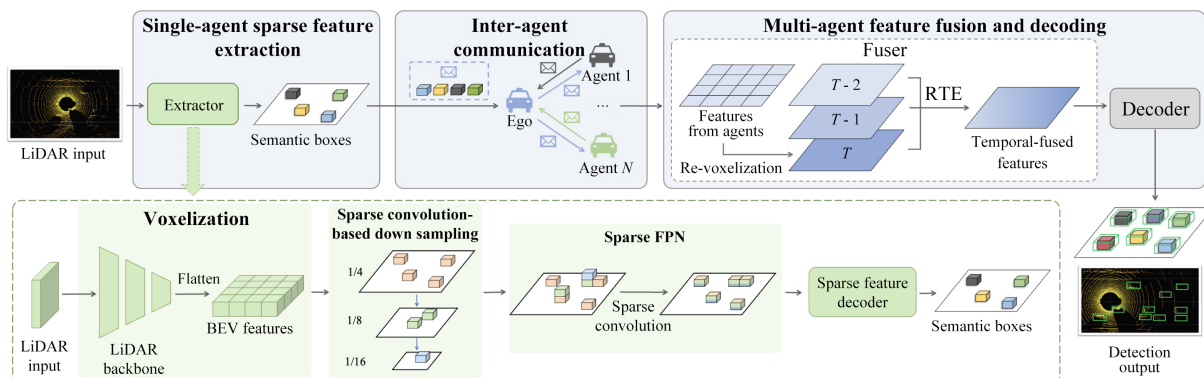


Fig. 2 Overview of Which2comm, a multiagent 3D object detection framework based on object-level sparse features. Light detection and ranging (LiDAR) point cloud is input into Extractor, obtaining the sparse SemDBs. Multiagent sparse features are encoded with RTE, and further processed through Fuser and Decoder.

where m_j , $\dim(\mathbf{F}_j)$, and $\dim(\mathbf{I}_j^{\text{spac}})$ represent the number of SemDBs that agent j detects, the dimension of transmitted features, and the dimension of corresponding spatial indices, respectively. The set of features detected and received by ego agent i is denoted as $\tilde{\mathbf{F}}_i$. Let $\mathbf{F}_{j \rightarrow i}$ represent the features transmitted from agent j to ego agent i , and N be the total number of connected agents. In this case, the transmission process can be represented as

$$\tilde{\mathbf{F}}_i = \bigcup_{j, j \neq i} (\mathbf{F}_{j \rightarrow i}) \quad (2)$$

Sparse feature fusion and decoding. The Fuser module is used to incorporate temporal encoding into sparse features from multiple agents and further combine features from the past p time steps. For ego agent i , denote $f_{\text{tmp_enc}}(\cdot)$ and $\Phi_{\text{fus}}(\cdot)$ as the temporal encoding mechanism and the fusion operator, respectively. The Fuser module can be formulated as

$$\mathbf{T}_{i,j}^k = f_{\text{tmp_enc}}(\tilde{\mathbf{F}}_{i,j}^k) \quad (3)$$

$$\mathbf{H}_i = \Phi_{\text{fus}}\left(\{\mathbf{T}_{i,j}^k\}_{k=c-1, \dots, c-p; j=0, 1, \dots, N}\right) \quad (4)$$

where $t_k \in \{t_c, t_{c-1}, \dots, t_{c-p}\}$ is the temporal index representing the k -th timestamp before the current time t_c (including it); $\tilde{\mathbf{F}}_{i,j}^k \in \mathbb{R}^{m_j \times C}$ with C being the number of channels is the feature transmitted from agents j to i at timestamp t_k ; $\mathbf{T}_{i,j}^k \in \mathbb{R}^{m_j \times C}$ is the corresponding temporally encoded feature; and $\mathbf{H}_i \in \mathbb{R}^{\tilde{m}_i \times C}$ represents the fused sparse feature of agent i with the final object count of $\tilde{m}_i$. Subsequently, the fused feature, $\mathbf{H}_i$, is fed into the decoder to transform the implicit sparse semantic representations into explicit 3D object bounding boxes.

While Which2comm optimizes single-link transmission, practical V2X environments often involve numerous agents. This necessitates a higher-level strategy to select the most critical collaborators, which will be discussed in Section 4.

3.2 Single-agent sparse feature extraction

The extractor preserves sparsity throughout the feature propagation process by employing a fully sparse structure. In this module, the input point clouds, $\mathbf{X}_i$, are initially converted into voxel-level features through voxelization and then downsampled through consecutive sparse convolution modules to capture multiscale features. Subsequently, a sparse feature pyramid network (FPN) is employed for multilevel feature fusion. SemDBs are predicted through a carefully designed sparse feature decoding module.

Voxel feature encoding. Two-dimensional (2D) voxelization is applied to transform the input point clouds $\mathbf{X}_i$ into voxel-level features, denoted as $\mathbf{V}_i \in \mathbb{R}^{n_v \times c}$, where n_v is the number of voxels. Additionally, voxel indices $\mathbf{I}_i \in \mathbb{R}^{n_v \times 4}$ are provided for subsequent convolutions.

Multiscale sparse feature extraction. Multiscale sparse feature extraction is used to extract object information from voxel features and capture multiscale features while preserving sparsity. This extraction process involves stacked sparse convolutional building blocks containing a sparse convolutional layer and multiple submanifold sparse convolutional structures. To alleviate the degradation phenomenon, residual connection structures are integrated into sparse convolutional layers. As features propagate in the network, the receptive field of the features continuously increases, extracting more abstract features. Ultimately, downsampled features by factors of 1, 1/2, 1/4, 1/8, and 1/16 are

obtained, which are denoted as $\mathbf{F}_i^{\text{ds}_1 \times}$, $\mathbf{F}_i^{\text{ds}_2 \times}$, $\mathbf{F}_i^{\text{ds}_4 \times}$, $\mathbf{F}_i^{\text{ds}_8 \times}$ and $\mathbf{F}_i^{\text{ds}_{16} \times}$, respectively.

Sparse FPN. The multiscale features $\mathbf{F}_i^{\text{ds}_4 \times}$, $\mathbf{F}_i^{\text{ds}_8 \times}$, and $\mathbf{F}_i^{\text{ds}_{16} \times}$ are fused through a sparse FPN to obtain downsampled features with a factor of 1/4. The sparse FPN is built upon the conventional FPN by replacing dense operators with their sparse counterparts. To further process the fused sparse features, spatial index upsampling is employed, which solely upsamples the spatial indices of higher-level features without altering the values. The upsampled sparse features with identical positional indices are directly summed to merge features from different levels, referred to as sparse addition. The sparse FPN layer outputs fused features that match the dimension of the downsampled features with a factor of 1/4, denoted as $\mathbf{F}_i^{\text{ds}_4 \times}$.

Sparse feature decoding. The decoding structure comprises a confidence head and a box center head, enabling three critical processes: box center adjustment, box selection, and feature reweighting (RW). The decoding heads are constructed with sparse convolutional layers, predicting the classification confidences $\{\mathbf{Y}_{i,k}^{\text{conf}}\}_{i=1,2,\dots,N; k=1,2,\dots,m_i}$ and the relative positional deviations between the sparse features and the actual detection box center $\{\mathbf{Y}_{i,k}^{\text{dev}}\}_{i=1,2,\dots,N; k=1,2,\dots,m_i}$.

As LiDAR point clouds mainly capture the outer surfaces of objects, sparse feature positions often deviate from the actual center of the 3D object detection box. To mitigate the negative impact on feature fusion across agents, a box center head is introduced to predict the positional deviation between sparse features and box centers. The adjusted spatial index of the sparse feature is calculated as $\hat{\mathbf{I}}_i^k = \mathbf{I}_i^k + \mathbf{Y}_{i,k}^{\text{dev}}$, which corresponds to the original index plus the relative deviation.

Apart from the required object features, sparse features obtained through sparse FPN also contain background and redundant features. Features with confidence scores below the threshold are considered background and are excluded from the candidate set to maintain feature sparsity. Meanwhile, a submanifold sparse max-pooling (Graham and Maaten, 2017) layer is employed to remove redundant features, which serves as a substitute for nonmaximum suppression (NMS) but is suitable for sparse features.

Moreover, features from diverse agents can be influenced by noise and uncertainties. Feature RW leverages this characteristic by embedding uncertainties into the sparse features, enabling a more precise representation of their reliability. Specifically, the sparse features are multiplied by their predicted confidence to suppress features with higher uncertainties. The ablation studies in Section 6.4 demonstrate that feature RW is an effective strategy to enhance model performance.

Through the aforementioned extraction pipeline, multiscale features of objects are extracted, fused, filtered, and refined, resulting in highly compact and informative object-level sparse representations, namely, SemDBs. Formally, a SemDB transmitted from agent j is defined as a tuple $(\mathbf{F}_j, \mathbf{I}_j^{\text{spac}})$, where $\mathbf{F}_j \in \mathbb{R}^{1 \times 128}$ is a compressed semantic feature vector representing the object's properties, and $\mathbf{I}_j^{\text{spac}} \in \mathbb{R}^{1 \times 3}$ is its adjusted 3D spatial index in the voxel coordinate system. Each SemDB provides sufficient semantic information for a detection box and leads to superior detection performance while maintaining communication efficiency. A detailed qualitative analysis of the SemDBs is provided in Section 6.3.

3.3 Multiagent feature fusion and decoding

Agents exchange SemDBs during the interagent communication

phase. Then, the sparse feature fusion module, Fuser, is utilized to obtain the fused sparse features, $\mathbf{H}_i$. Subsequently, the explicit 3D object detection boxes, $\mathbf{B}_i$, are generated through the sparse feature decoding module, i.e., Decoder.

RTE mechanism. To distinguish features from different timestamps, the RTE mechanism is utilized. Inspired by the positional encoding mechanism in Transformer (Vaswani et al., 2017), the RTE is represented as

$$F_{\text{RTE}}(\Delta t, 2k) = \sin\left(\frac{\Delta t}{10,000^{2k/D}}\right) \quad (5)$$

$$F_{\text{RTE}}(\Delta t, 2k+1) = \cos\left(\frac{\Delta t}{10,000^{2k/D}}\right) \quad (6)$$

where Δt represents the difference between historical and current features' timestamp; F_{RTE} denotes the RTE function applied to features; D and k denote the dimension of features and the index of channel within the feature, respectively. The timestamp-aware vector $F_{\text{RTE}}(\Delta t, k)$ is added to features directly, obtaining a temporally encoded vector, $\mathbf{T}_{i,j}^k$.

Revoxelization (RV). In the Fuser module, features collected from different agents and timestamps are fused through sparse addition. However, due to the lower voxel resolution after downsampling, sparse features from different time steps frequently map to the same voxel. To address this issue, we employ a RV method that upsamples the spatial indices of these features to high-resolution coordinates before fusion. This effectively enhances voxel resolution and reduces feature overlapping.

Feature fusion. The revoxelized features encoded with RTE are fed to a sparse addition module to fuse features with the same spatial index across different time steps. Subsequently, further fusion is conducted through downsampling and sparse FPN, reusing the structures within the extractor. The fused feature, $\mathbf{H}_i$, is obtained.

Feature decoding. The decoder consists of four feature decoding heads: a confidence head, a box center head, a box dimension head, and a box rotation head, which share nearly identical structures. The confidence head outputs the detection confidence; the box center head outputs the deviations in the x , y , and z directions; the box dimension head outputs the length, width, and height of the 3D object detection box; and the box rotation head outputs the rotation angle yaw around the z -axis. Combining and transforming this information yields the output 3D object detection boxes, $\mathbf{B}_i$.

3.4 SE data augment strategy

Multiagent datasets contain data from multiple agents within the same scene. During the training and inference processes of multiagent object detection models, it is necessary to specify one agent as the reference center, known as the ego agent. When evaluating the effectiveness of a model, a fixed ego agent is chosen to establish a unified benchmark for comparisons. Although many methods (Hu et al., 2022; Wang et al., 2020; Xu et al., 2022a) commonly adopt this setting during training, they do not take full advantage of multiterminal datasets, resulting in insufficient data utilization.

To address this limitation, we propose a SE strategy. For each training epoch, we randomly select one of the available connected agents in a scene as the ego agent. This does not increase the data size but trains the model under diverse ego perspectives across epochs, significantly enhancing data utilization and model generalization.

4 Adaptive collaboration mechanism

While Which2comm effectively minimizes the communication payload per agent by transmitting object-level sparse features, real-world V2X environments frequently feature dense traffic participants and varying CAV penetration rates. Under such conditions, data broadcasting among all available agents often saturates the communication bandwidth. To address this macrolevel communication bottleneck, we propose an adaptive collaboration mechanism to further optimize communication efficiency, especially in high-density scenarios. This mechanism acts as a pluggable decision-making layer that prioritizes high-risk agents. Although this mechanism is integrated into Which2comm in this study, it is independent of the collaborative perception tasks, enabling broad applicability across different V2X systems. Two steps in this mechanism, i.e., traffic risk assessment and adaptive agent selection, are sequentially presented below.

4.1 Traffic risk assessment

In dense traffic scenarios with high CAV penetration rates, selecting collaborative agents based on safety assessment can facilitate efficient bandwidth utilization. Existing selection methods typically rely on random sampling or simple physical distance. However, physical distance alone does not fully reflect actual traffic risks. For example, a distant vehicle approaching at a high relative speed poses a greater collision risk than a stationary vehicle close to the ego agent. To quantify safety risks from the ego-vehicle's perspective, we adopt the Traffic Safety Field model proposed by Wang et al. (2015). Although the original model superimposes potential, kinetic, and behavior fields to comprehensively represent traffic risks, accurate measurement of all coefficients (e.g., driving risk factors) is challenging in collaborative perception tasks. Therefore, we adapt the model by setting the unobservable environmental factor R and driver factor D_R to constants ($R = D_R = 1$) and by applying a typical vehicle mass. The simplified safety field, E_s , is derived as

$$E_s = (1 + \exp(0.05v\cos\theta)) \cdot \frac{0.001M}{|\mathbf{r}|} \cdot \frac{\mathbf{r}}{|\mathbf{r}|} \quad (7)$$

where v , M , $\mathbf{r}$, and θ represent the velocity, virtual mass, distance vector, and velocity direction angle of the ego vehicle, respectively.

On this basis, a conflict warning evaluation indicator called the DSI (Wang et al., 2016) based on the safety field is introduced to stably and comprehensively assess traffic risks. First, consider a virtual force, $\mathbf{F}_{ij}$, experienced within the safety field:

$$\mathbf{F}_{ij} = E_i \cdot M_j \cdot \frac{\mathbf{r}}{|\mathbf{r}|} \quad (8)$$

where E_i is the safety field from the perspective of ego vehicle i ; and M_j is the virtual mass of target j . It can be proven that $\mathbf{F}_{ij}$ is a conservative force, so that the safety potential energy can be defined as

$$\text{SPE}_{ij} = - \int_{\infty}^{|\mathbf{r}_{ij}|} \mathbf{F}_{ij} \cdot d\mathbf{r} \quad (9)$$

Furthermore, the motion relationship between two vehicles can be characterized by the change rate of the safety potential energy:

$$\dot{\text{SPE}}_{ij} = \frac{d\text{SPE}_{ij}}{dt} = \frac{d\text{SPE}_{ij}}{d\mathbf{r}} \cdot \frac{d\mathbf{r}}{dt} = \mathbf{F}_{ij} \cdot (\mathbf{v}_i - \mathbf{v}_j) \quad (10)$$

where r_j denotes the relative distance between ego vehicle i and target vehicle j ; and $\mathbf{v}_i$ and $\mathbf{v}_j$ represent the velocity vectors of ego

Table 1 Comparisons among the representative public cooperative perception datasets

Dataset	Year	Source	V2X	RGB image	LiDAR	3D box	Category	Connection
DAIR-V2X (Yu et al., 2022)	2022	Real	V2I	7.1×10^4	7.1×10^4	1.2×10^6	10	1
V2V4Real (Xu et al., 2023)	2023	Real	V2V	4.0×10^4	2.0×10^4	2.4×10^5	5	1
RCooper (Hao et al., 2024)	2024	Real	I2I	5.0×10^4	3.0×10^4	—	10	—
OPV2V (Xu et al., 2022b)	2022	Sim	V2V	1.32×10^5	3.3×10^4	2.3×10^5	1	1–6
V2XSet (Xu et al., 2022a)	2022	Sim	V2X	1.32×10^5	3.3×10^4	2.3×10^5	1	1–4
V2X-Sim (Li et al., 2022)	2022	Sim	V2X	2.83×10^5	4.7×10^4	2.66×10^4	1	1–4
Multi-V2X	2025	Sim	V2X	5.49×10^5	1.46×10^5	4.219×10^6	6	0–31

vehicle i and target vehicle j , respectively.

The weighted sum of the above two components yields the driving safety indicator:

$$DSI = \alpha \cdot SPE + (1 - \alpha) \cdot SPE \quad (11)$$

where α is a weighting factor between 0–1.

4.2 Adaptive selection of collaborative agents

The DSI can quantify traffic risks in complex scenarios and indicate the safety requirements for perception. To obtain the necessary parameters for calculating DSI, we conduct a coarse-grained collaboration before the perception stage. Specifically, each CAV performs single-agent object detection by leveraging the same PointPillars (Lang et al., 2019) backbone used in Which2comm with a simplified decoding head and then transmits the preliminary results to other agents. This process occupies little communication bandwidth.

To align the collaboration with actual traffic risks, the agent selection follows a risk-driven, target-centric logic rather than simple spatial proximity to the ego vehicle. When selecting collaborative agents, the method involves repeatedly choosing the target with the highest DSI, which comprehensively quantifies the dynamic traffic risk posed by each traffic participant. Assuming that CAVs spatially close to a target can provide more detailed and high-quality information for collaborative perception, the agent closest to the identified high-risk target is sequentially added to the collaboration list. This process continues until the required number of agents is fulfilled.

It is worth highlighting that the proposed adaptive selection mechanism appears as a standalone and architecture-agnostic module. This allows it to be seamlessly integrated into various collaborative perception models as a general strategy to optimize communication efficiency in dense traffic.

5 Multi-V2X dataset

To evaluate the effectiveness of the proposed Which2comm framework and adaptive collaboration mechanism, a testing dataset capable of simulating dense traffic is needed. However, existing public datasets primarily focus on low-density scenarios with fixed CAV penetration rates. To address this limitation, this section introduces a new large-scale dataset named Multi-V2X. By supporting variable CAV penetration rates and dense agent connections, Multi-V2X provides a practical foundation for evaluating the methods under more realistic V2X conditions.

5.1 Dataset construction and statistics

Multi-V2X was generated through CARLA-SUMO cosimulation. Specifically, SUMO was used to simulate the real sensing data and traffic movements (including route planning, action, and signal control), and CARLA was used for sensor simulation and data recording. The cosimulation was achieved by synchronizing the

states of actors in SUMO to CARLA and then updating the sensor simulation at each timestep. During simulations, vehicles, pedestrians, and their routes were randomly generated in the scene. Hundreds of vehicles and pedestrians were generated in six scenes (i.e., Town01, Town03, Town06, Town07, Town10HD). Road types included crossroads, T-junctions, segments, mid-blocks, rural roads, etc. We manually selected periods with a length of 30–40 s to record data, including sensing results, annotations, and movements of all agents during these periods.

As shown in Table 1, Multi-V2X includes 5.49×10^5 images, 1.46×10^5 point clouds, 4.219×10^6 3D bounding boxes (within the range of $x \in [-140 \text{ m}, 140 \text{ m}]$ and $y \in [-40 \text{ m}, 40 \text{ m}]$ around an agent, same as V2XSet and OPV2V), for 6 categories (car, van, truck, cyclist, motor, pedestrian), supporting V2V and V2I cooperation. The maximum number of connections within the communication range of 70 m reaches 31, which is much larger than that in the existing dataset. This poses great challenges to balancing performance and bandwidth.

5.2 CAV penetration rate

The maximum CAV penetration rates in six scenes range from 55.17% to 86.21%, providing foundations for further exploration. To reduce data redundancy, we propose obtaining various CAV penetration rates in a single dataset instead of running separate simulations for different CAV penetration rates. Specifically, during the simulation process, a subset of simulated vehicles is equipped with autonomous sensor suites for data collection. During dataset deployment, some sensor-equipped vehicles are considered nonequipped vehicles to obtain a dataset of the specified CAV penetration rate.

Furthermore, the systematic sampling approach presented here also serves as the foundation for cooperative risky testing-scenario generation (CoRTSG) (Li et al., 2025), which is a specialized dataset designed for evaluating the robustness of cooperative driving systems under challenging edge cases. By adopting the same data pipeline, CoRTSG extends our research to safety-critical testing. Both datasets are publicly available at <https://github.com/RadetzkyLi/Multi-V2X> and <https://github.com/RadetzkyLi/CoRTSG>, respectively.

6 Experiments

6.1 Metrics and experimental settings

Datasets. We evaluate our method on three datasets, including the proposed Multi-V2X and widely used V2XSet and OPV2V, to ensure a comprehensive comparison with existing state-of-the-art methods. V2XSet (Xu et al., 2022a) is a large-scale simulation dataset well suited for exploring typical challenges, such as communication latency. It comprises 11,447 frames of perception scenarios with both CAVs and RSUs and is partitioned into training, validation, and testing splits of 6694, 1920, and 2833 frames, respectively. OPV2V (Xu et al., 2022b) is a cooperative

Table 2 Quantitative results of detection performance and communication cost

Dataset	V2XSet			OPV2V			V2V4Real			MultiV2X ($r = 0.3$)		
Method/metric	AP@0.5	AP@0.7	Cost (MB)	AP@0.5	AP@0.7	Cost (MB)	AP@0.5	AP@0.7	Cost (MB)	AP@0.5	AP@0.7	Cost (MB)
No fusion	0.606	0.402	0	0.679	0.602	0	0.393	0.194	0	0.223	0.102	0
Early fusion	0.881	0.793	178.527	0.908	0.854	256.178	0.452	0.238	121.280	0.422	0.249	275.001
Late fusion	0.776	0.649	0.0001	0.885	0.824	0.0001	0.428	0.199	0.0001	0.238	0.133	0.0001
V2X-ViT	0.882	0.713	0.416	0.866	0.769	0.429	0.443	0.228	0.174	0.332	0.218	1.962
Where2comm	0.833	0.624	4.011	0.877	0.775	7.459	0.490	0.204	3.281	0.341	0.205	4.361
CoAlign	0.881	0.774	14.754	0.903	0.833	13.728	0.457	0.233	11.245	0.360	0.214	16.244
Which2comm	0.929	0.833	0.022	0.925	0.873	0.014	0.502	0.258	0.011	0.450	0.277	0.053
Which2comm – with adaptive collaborative	0.930	0.833	0.014	0.924	0.872	0.014	0.502	0.257	0.011	0.431	0.259	0.031

Vehicle-to-Vehicle perception dataset and provides diverse collaborative perception tasks across various traffic scenarios. The training, validation, and testing splits of OPV2V include 6764, 1981, and 2709 frames, respectively. To test robustness in real-world situations, we also evaluated methods on V2V4Real (Xu et al., 2023), a large-scale, real-world collaborative perception dataset collected across diverse urban environments. V2V4Real comprises 20,000 LiDAR frames captured by sensors mounted on vehicles. Given real-world conditions, these data involve actual global positioning system (GPS) localization drifts and sensor noises. To ensure fair comparisons, we only utilized LiDAR point clouds as input for all methods.

Metrics. The evaluation focuses on both perception performance and communication efficiency. The perception performance was evaluated using average precision (AP) at IoUs of 0.5 and 0.7 (denoted as AP@0.5 and AP@0.7). The communication efficiency was evaluated using average communication cost, which counted the data volume transmitted from collaborators to the ego vehicle and was averaged across all frames.

Experimental settings. Following Xu et al. (2022b), all models employed PointPillars (Lang et al., 2019) as the backbone network. The point cloud range was set as follows: $x \in [-140.8 \text{ m}, 140.8 \text{ m}]$, $y \in [-38.4 \text{ m}, 38.4 \text{ m}]$, and $z \in [-3 \text{ m}, 1 \text{ m}]$. The voxel size was set to $[0.4 \text{ m}, 0.4 \text{ m}, 4 \text{ m}]$. For our proposed Which2comm framework, temporal fusion (TF) was used to integrate historical frames (i.e., $p = 2$). The models were trained with the Adam optimizer in an end-to-end manner for 50 epochs. The batch size and weight decay were set to 8 and 1×10^{-2} , respectively. A cosine annealing learning rate scheduler was adopted, with a peak learning rate of 1×10^{-3} and a 20-epoch warm-up phase.

Loss Functions. For Which2comm, the decoding heads in both Extractor and Decoder consisted of classification and regression branches. We used the focal loss (Lin et al., 2017) and L1 loss for the classification and regression tasks, represented as L_{cla} and L_{reg} . The overall loss is the weighted sum of the losses from each decoding head, i.e., $L = L_{\text{cla}} + \alpha L_{\text{reg}}$, where the weighting factor, α , was set to 1 in this study.

6.2 Quantitative results

Comparison with benchmarks. Table 2 presents the overall performance of collaborative 3D object detection methods on simulation (V2XSet, OPV2V, Multi-V2X with $r = 0.3$) and real-world (V2V4Real) datasets. Comparisons were conducted on various baseline models, including no fusion, early fusion, and late fusion models, as well as intermediate fusion models V2X-ViT (Xu et al., 2022a), Where2comm (Hu et al., 2022), and SCOPE (Yang et al., 2023). Compared with baseline models and intermediate fusion models, Which2comm consistently achieved

superior detection precision with the lowest communication costs. Specifically, Which2comm attained state-of-the-art AP@0.5 scores on V2XSet and OPV2V (0.929 and 0.925, respectively) while requiring minimal bandwidth. Moreover, the excellent performance on V2V4Real validated its robustness against real-world perturbations. The AP@0.5 with the single agent was boosted from 0.393–0.502 under Which2comm, outperforming other intermediate methods. This was achieved with a payload of only 0.011 MB, substantially reducing bandwidth compared with Where2comm (3.281 MB) and CoAlign (11.245 MB). These results collectively demonstrated that transmitting object-level sparse features could provide a resilient and bandwidth-efficient solution across diverse traffic conditions.

When testing Which2comm with the adaptive collaboration, we constrained the maximum number of collaborative terminals to 3. Almost all scenarios in OPV2V had no more than 3 collaborative terminals, so the adaptive collaboration mechanism had a negligible effect on OPV2V. For V2XSet, the adaptive collaboration mechanism effectively reduced communication cost while maintaining satisfactory detection performance. For Multi-V2X, accuracy slightly decreased due to reduced collaborative terminals, but it exhibited better performance with latency (Fig. 3). These results demonstrated that the adaptive collaboration mechanism was robust to communication latency, enabling practical applications. Moreover, it was found that the metrics from Multi-V2X were much lower than those from other datasets. This was attributed to the high agent density, severe occlusion, and challenging multiple-class evaluation in Multi-V2X.

Latency robustness. Various latencies were applied to evaluate the models' robustness. The experimental results are shown in Fig. 3. Which2comm demonstrated enhanced robustness to latency, particularly to latencies within 0–200 ms. The enhanced performance was attributed to (1) the temporal encoding and fusion designs that fused features from the past two timestamps and (2) SemDBs that effectively integrated global semantic information with local details of objects.

Complexity analysis. We benchmarked Which2comm on V2XSet using a single RTX 3090 GPU (Table 3). Which2comm exhibited significant computational advantages with fewer parameters and less memory occupancy than most competitors, owing to its sparse convolution architecture. In terms of inference speed, Which2comm achieved approximately 20 Hz, outperforming V2VNet (Wang et al., 2020) and V2X-ViT (Xu et al., 2022a). As shown in the last row in Table 3, incorporating historical information introduced negligible computational overhead, and each additional historical frame required ~150 MB of memory. The ablation studies in Table 4 revealed that the sparse convolutional backbone accounted for most of the computational load, which can be further optimized through quantum computing in the future.

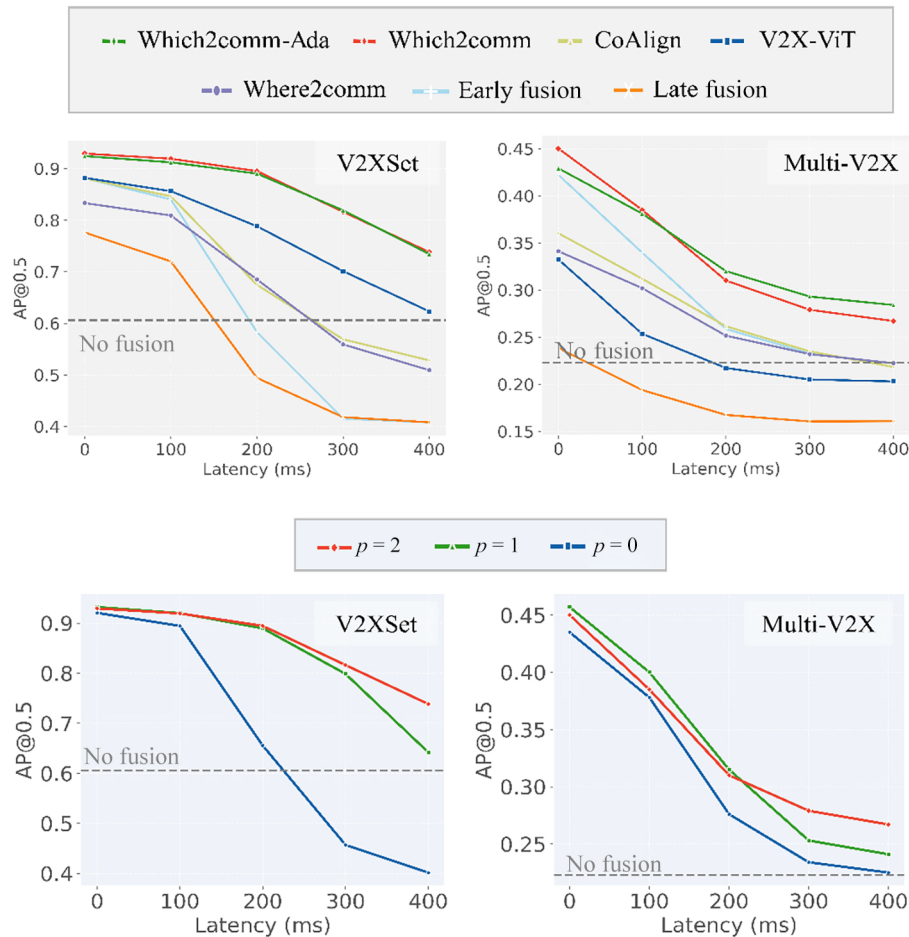


Fig. 3 Evaluation of latency robustness on V2XSet and Multi-V2X, as well as the ablation of the TF mechanism.

Table 3 Comparison of different methods in time and space complexities

Method	#param (M)	Inference GPU memory (MB)	Inference speed (FPS)
V2VNet	14.6	1996	15.8
V2X-ViT	14.2	1432	10.4
Where2comm	8.1	1701	48.7
CoAlign	11.1	2703	52.6
Which2comm ($p = 0/p = 1/p = 2$)	10.5/10.5/ 10.5	1296/1392/ 1556	20.3/18.4/ 19.0

Table 4 Ablation studies on single-agent and multiagent modules using V2XSet

Inference stage	Inference time when $p = 0/1/2$ (ms)
Single-agent voxel feature encoding	0.79/0.86/0.81
Single-agent multi-scale sparse feature extraction + sparse FPN	22.12/23.65/22.65
Single-agent sparse feature decoding	5.07/6.69/7.36
Multi-agent RTE + RV + feature fusion	16.56/17.95/17.03
Multi-agent sparse feature decoding	4.72/5.10/4.84
All	49.26/54.25/52.69

6.3 Qualitative results

Fig. 4 visualizes the SemDBs and detection boxes projected to 2D space on V2XSet. As shown in Fig. 4a, in which different agents

are marked with different colors, SemDBs are precisely localized through our RV process. Moreover, Which2comm leveraged sequential information to refine detection box positions and capture object motion patterns, effectively mitigating the impact of location discrepancies on detection results. Observing the detection results shown in Figs. 4b–4e, we found that perception models typically exhibited an increased false negative rate in areas with occlusions, while Which2comm showed higher recall rates compared with other models. Additionally, Which2comm provided more precise bounding-box pose estimates, particularly in right-side regions that were primarily covered by a single agent's LiDAR. The excellent performance of Which2comm partially stems from its TF mechanism, which allows it to track the motion status of objects, thereby effectively addressing occlusion issues and enhancing estimation consistency.

6.4 Ablation studies

Ablations on model components. Table 5 shows the results of ablation studies on model components using V2XSet. Both communication with (100 ms) and without latency were considered. The baseline model only transmitted current-frame sparse features without temporal encoding and excluded the RW module, RV operation, and SE augmentation. The results indicated that (1) the RTE mechanism enhanced the model's performance when latency existed; (2) the SE data augmentation strategy significantly improved the detection performance; (3) sparse feature RW effectively suppressed unstable features in model predictions, enhancing overall prediction accuracy; and (4) simultaneously applying TF and feature RV led to significant

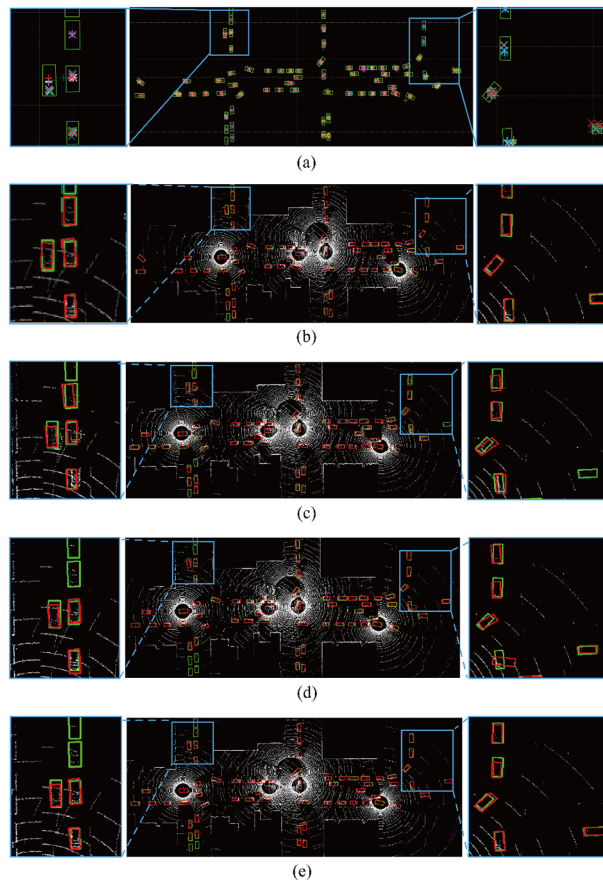


Fig. 4 Visualization of (a) SemDBs and detection results of (b) Which2comm, (c) V2X-ViT, (d) Where2comm, and (e) CoAlign in an intersection scenario in V2XSet. Colored '+' and 'x' symbols represent SemDBs from the current frame and historical frames. Red and green boxes represent the predictions and ground truth, respectively.

Table 5 Ablation studies on different model components using V2XSet

RTE	SE	RW	RV	TF	w/o latency	w/latency (100 ms)
					AP@0.5	AP@0.5
—	—	—	—	—	0.877	0.805
√	—	—	—	—	0.868	0.843
√	√	—	—	—	0.905	0.891
√	√	√	—	—	0.917	0.900
√	√	√	—	√	0.910	0.893
√	√	√	√	—	0.920	0.908
√	√	√	√	√	0.929	0.919

performance improvement. All these results clearly demonstrated the effectiveness of different components in the model.

Ablation on the TF mechanism. Fig. 3 depicts the performance with varying numbers of previous frames used in the TF process. The TF mechanism led to a slight decrease in precision under zero-latency conditions but significantly enhanced robustness to communication delays. The number of previous frames considered was proportional to the model's robustness. Moreover, we conducted a complexity analysis of the RTE mechanism.

7 Conclusions

This study proposes Which2comm, a multiagent collaborative 3D object detection framework leveraging sparse SemDBs. By incorporating semantic information into object detection boxes,

the proposed SemDB takes full advantage of the late fusion and intermediate fusion approaches, enabling object-level sparse feature transmission among agents. Moreover, a TF method is further adopted to improve model performance and latency robustness. A SE data augmentation strategy is designed to enhance the utilization of multiterminal perceptual data. Furthermore, an adaptive collaboration mechanism is proposed to achieve intelligent selection of collaboration agents, significantly improving communication efficiency in high-density and complex traffic scenarios without sacrificing perception accuracy. A high-fidelity, large-scale dataset, Multi-V2X, that supports various CAV penetration rates and features complex multiagent interactions is established. Comprehensive experiments using existing public datasets and the proposed Multi-V2X dataset demonstrate Which2comm's SOTA performance on both perception accuracy and communication efficiency.

Limitations and future work. Despite the advancement of Which2comm, its robustness to real-world communication conditions, such as pose errors, localization noise, and packet loss, was not investigated in this study. In the future, possible research directions include addressing the above limitations and exploring approaches to incorporating multimodal sensor data to further enhance detection performance.

Author contributions

Duanrui Yu: Methodology, Validation, Writing – original draft. **Anqi Qu:** Validation, Visualization. **Jing You:** Conceptualization, Methodology, Validation. **Dingyu Wang:** Methodology, Validation, Writing – original draft. **Rongsong Li:** Software, Data curation. **Shaocheng Jia:** Supervision and revision. **Xin Pei:** Supervision, Funding acquisition.

Replication and data sharing

The data and codes used in this study are available at <https://doi.org/10.26599/ETSD.2026.9190032>.

Acknowledgements

The work described in this paper was partially supported by the National Key R&D Program of China (No. 2022ZD0162200) and the National Natural Science Foundation of China (Nos. 72601363, 72525009, and 72431009).

Declaration of competing interest

The authors have no competing interests to declare that are relevant to the content of this article.

References

- Chen, Q., Tang, S., Yang, Q., Fu, S., 2019. Cooper: Cooperative perception for connected autonomous vehicles based on 3D point clouds. In: 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS), 514–524.
- Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., Koltun, V., 2017. CARLA: An open urban driving simulator. In: Proceedings of the 1st Annual Conference on Robot Learning, 1–16.
- Filippi, A., Moerman, K., Martinez, V., Turley, A., Haran, O., Toledano, R., 2017. IEEE 802.11p Ahead of LTE-V2V for Safety Applications. https://www.hit.bme.hu/~jakab/edu/litr/V2x_Cellular/LTE-V2V-WP.pdf
- Gan, J., Sheng, Y., Zhang, H., Liang, L., Ye, H., Guo, C., et al., 2026. SComCP: Task-oriented semantic communication for collaborative perception. *IEEE Trans Veh Technol*, 75, 13080–13094.
- Graham, B., Maaten, L. V. D., 2017. Submanifold Sparse Convolutional Networks. <https://doi.org/10.48550/arXiv.1706.01307>
- Hao, R., Fan, S., Dai, Y., Zhang, Z., Li, C., Wang, Y., et al., 2024. RCooper: A real-world large-scale dataset for roadside cooperative perception.

- In: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 22347–22357.
- Hu, Y., Fang, S., Lei, Z., Zhong, Y., Chen, S., 2022. Where2comm: Communication-Efficient Collaborative Perception via Spatial Confidence Maps. <https://doi.org/10.48550/arXiv.2209.12836>
- Hu, Y., Lu, Y., Xu, R., Xie, W., Chen, S., Wang, Y., 2023. Collaboration helps camera overtake LiDAR in 3D detection. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 9243–9252.
- Hu, Y., Peng, J., Liu, S., Ge, J., Liu, S., Chen, S., 2024. Communication-efficient collaborative perception via information filling with codebook. In: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 15481–15490.
- Huang, T., Liu, J., Zhou, X., Nguyen, D. C., Rahimi Azghadi, M., Xia, Y., et al., 2025. Vehicle-to-everything cooperative perception for autonomous driving. *Proc IEEE*, **113**, 443–477.
- Lang, A. H., Vora, S., Caesar, H., Zhou, L., Yang, J., Beijbom, O., 2019. PointPillars: Fast encoders for object detection from point clouds. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 12689–12697.
- Li, R., Pei, X., Xing, L., 2025. CoRTSG: A general and effective framework of risky testing scenario generation for cooperative perception in mixed traffic. *Accid Anal Prev*, **220**, 108163.
- Li, Y., Ma, D., An, Z., Wang, Z., Zhong, Y., Chen, S., et al., 2022. V2X-sim: Multi-agent collaborative perception dataset and benchmark for autonomous driving. *IEEE Robot Autom Lett*, **7**, 10914–10921.
- Lin, T. Y., Goyal, P., Girshick, R., He, K., Dollar, P., 2017. Focal loss for dense object detection. In: 2017 IEEE International Conference on Computer Vision (ICCV), 2999–3007.
- Liu, S., Gao, C., Chen, Y., Peng, X., Kong, X., Wang, K., et al., 2023. Towards Vehicle-to-Everything Autonomous Driving: A Survey on Collaborative Perception. <https://doi.org/10.48550/arXiv.2308.16714>
- Liu, Y. C., Tian, J., Glaser, N., Kira, Z., 2020. When2com: Multi-agent perception via communication graph grouping. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 4105–4114.
- Lopez, P. A., Wiessner, E., Behrisch, M., Bieker-Walz, L., Erdmann, J., Flotterod, Y. P., et al., 2018. Microscopic traffic simulation using SUMO. In: 2018 21st International Conference on Intelligent Transportation Systems (ITSC), 2575–2582.
- Lu, Y., Hu, Y., Zhong, Y., Wang, D., Wang, Y., Chen, S., 2024. An Extensible Framework for Open Heterogeneous Collaborative Perception. <https://doi.org/10.48550/arXiv.2401.13964>
- Lu, Y., Li, Q., Liu, B., Dianati, M., Feng, C., Chen, S., et al., 2023. Robust collaborative 3D object detection in presence of pose errors. In: 2023 IEEE International Conference on Robotics and Automation (ICRA), 4812–4818.
- Qiao, D., Zulkernine, F., 2023. Adaptive feature fusion for cooperative perception using LiDAR point clouds. In: 2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), 1186–1195.
- Ren, S., Chen, S., Zhang, W., 2023. Collaborative perception for autonomous driving: Current status and future trend. In: Proceedings of 2021 5th Chinese Conference on Swarm Intelligence and Cooperative Control, 682–692.
- Shao, J., Li, T., Zhang, J., 2024. Task-oriented communication for vehicle-to-infrastructure cooperative perception. In: 2024 IEEE 34th International Workshop on Machine Learning for Signal Processing (MLSP), 1–6.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al., 2017. Attention is all you need. In: Proceedings of the 31st International Conference on Neural Information Processing Systems, 6000–6010.
- Wang, J., Wu, J., Li, Y., 2015. The driving safety field based on driver-vehicle-road interactions. *IEEE Trans Intel Transport Syst*, **16**, 2203–2214.
- Wang, J., Wu, J., Zheng, X., Ni, D., Li, K., 2016. Driving safety field theory modeling and its application in pre-collision warning system. *Transp Res Part C Emerg Technol*, **72**, 306–324.
- Wang, T., Kim, S., Wenxuan, J., Xie, E., Ge, C., Chen, J., et al., 2024. DeepAccident: A motion and accident prediction benchmark for V2X autonomous driving. *AAAI*, **38**, 5599–5606.
- Wang, T. H., Manivasagam, S., Liang, M., Yang, B., Zeng, W., Urtasun, R., 2020. V2VNet: Vehicle-to-vehicle communication for joint perception and prediction. In: *Computer Vision – ECCV 2020*. Vedaldi, A., Bischof, H., Brox, T., et al., Eds. Cham: Springer International Publishing, 605–621.
- Wei, S., Wei, Y., Hu, Y., Lu, Y., Zhong, Y., Chen, S., et al., 2023. Asynchrony-robust collaborative perception via bird's eye view flow. In: *Advances in Neural Information Processing Systems 36*. Neural Information Processing Systems Foundation, 28462–28477.
- Xu, R., Xia, X., Li, J., Li, H., Zhang, S., Tu, Z., et al., 2023. V₂V₄Real: A real-world large-scale dataset for vehicle-to-vehicle cooperative perception. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 13712–13722.
- Xu, R., Xiang, H., Tu, Z., Xia, X., Yang, M. H., Ma, J., 2022a. V2X-ViT: Vehicle-to-Everything Cooperative Perception with Vision Transformer. <https://doi.org/10.48550/arXiv.2203.10638>
- Xu, R., Xiang, H., Xia, X., Han, X., Li, J., Ma, J., 2022b. OPV₂V: An open benchmark dataset and fusion pipeline for perception with vehicle-to-vehicle communication. In: 2022 International Conference on Robotics and Automation (ICRA), 2583–2589.
- Yang, K., Yang, D., Zhang, J., Li, M., Liu, Y., Liu, J., et al., 2023. Spatio-temporal domain awareness for multi-agent collaborative perception. In: 2023 IEEE/CVF International Conference on Computer Vision (ICCV), 23326–23335.
- Yin, T., Zhou, X., Krahenbuhl, P., 2021. Center-based 3D object detection and tracking. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 11779–11788.
- Yu, H., Luo, Y., Shu, M., Huo, Y., Yang, Z., Shi, Y., et al., 2022. DAIR-V2X: A large-scale dataset for vehicle-infrastructure cooperative 3D object detection. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 21329–21338.
- Yuan, Y., Cheng, H., Sester, M., 2022. Keypoints-based deep feature fusion for cooperative vehicle detection of autonomous driving. *IEEE Robot Autom Lett*, **7**, 3054–3061.



Duanrui Yu received the B.S. degree in control science and engineering in 2023, and the M.S. degree in artificial intelligence (AI) from Tsinghua University, Beijing, China, in 2026. Her research interests include traffic safety and intelligent transportation systems.



Anqi Qu received the B.S. degree from the Department of Automation, Tsinghua University, Beijing, China, in 2025. She is currently pursuing the M.S. degree with the Department of Automation, Tsinghua University. Her research interests include computer vision and intelligent transportation systems.



Jing You received the B.S. and M.S. degrees in engineering from the Department of Automation from Tsinghua University, Beijing, China, in 2021 and 2024, respectively. He is currently a Software Engineer. His research interests include computer vision and intelligent transportation systems.



Dingyu Wang received the M.S. degree in engineering from the Department of Automation, Tsinghua University, Beijing, China, in 2025. His research interests include computer vision, traffic safety, and collaborative perception.



Rongsong Li received the B.S. degree in engineering, and the M.S. degree in electronic information from Tsinghua University, Beijing, China, in 2021 and 2024, respectively. He is currently an Engineer. His research interests include intelligent transportation systems and transportation safety.



Shaocheng Jia received the B.S. degree in measurement and control technology and instruments from China University of Petroleum (Beijing), Beijing, China, in 2018, the M.S. degree in control engineering from Tsinghua University, Beijing, China, in 2021, and the Ph.D. degree in traffic and transportation engineering from The University of Hong Kong, Hong Kong, China, in 2025. He is currently a Professor with the Institute of Intelligent Transportation Systems, Zhejiang University. His research interests include connected and automated transportation systems, ground-air collaborative transportation systems, and AI for transportation.



Xin Pei received the B.S. degree in automation in 2005 and the M.S. degree in transportation in 2007 from Tsinghua University, Beijing, China, and the Ph.D. degree in traffic engineering from The University of Hong Kong, Hong Kong, China, in 2011. She is currently a Research Associate Professor, with the Department of Automation, Tsinghua University. Her research interests include road safety evaluation and driving behavior analysis.