

Co3DCAE: Feature Compression for Intermediate Fusion Collaborative Perception for Autonomous Driving

Mingrui Wang, Judit Salavedra Pujol, Dongjie Li, Josep R. Casas, Javier Ruiz-Hidalgo

Abstract—In recent years, multi-agent sensing technology has attracted considerable attention, particularly in the field of vehicle-to-vehicle (V2V) collaborative perception for autonomous driving. By sharing complementary information, connected autonomous vehicles (CAVs) can significantly improve scene understanding. However, this collaboration also presents challenges, such as balancing data transmission efficiency with the accuracy of environmental perception. To address these issues, we propose an autoencoder based on intermediate-level feature fusion. Specifically, we design a 3D convolutional autoencoder with a residual structure to maximize the compression of inter-vehicle bandwidth usage while retaining essential feature information. By encoding the spatial-channel dimensions of intermediate feature maps extracted from point clouds, network bandwidth is reduced by 3–10 $\times$ under different configurations. Additionally, we propose a channel-wise autoencoder training strategy to reduce model parameters and enhance feature compression efficiency, while preserving collaborative perception performance. Finally, we conduct a comprehensive evaluation of perception accuracy and bandwidth consumption using a collaborative detection dataset collected via LiDAR. Extensive experimental results demonstrate that our method offers significant advantages in balancing object detection accuracy and bandwidth compression for CAVs.

Index Terms—Autonomous Driving, Collaborative Perception, Data Fusion, Object Detection, Bandwidth Compression.

I. INTRODUCTION

AUTONOMOUS driving systems rely on technologies such as object detection to perceive the surrounding environment in real time and enable intelligent decision-making. However, the perception capability of a single vehicle is limited, especially in complex traffic scenarios, where restricted detection range, blind spots, and sparse sensor observations hinder comprehensive and accurate environmental perception [1], [2]. Data fusion from sensors with different viewpoints, either from vehicles (Vehicle-to-Vehicle communication, V2V) or roadside units (Vehicle-to-Everything commu-

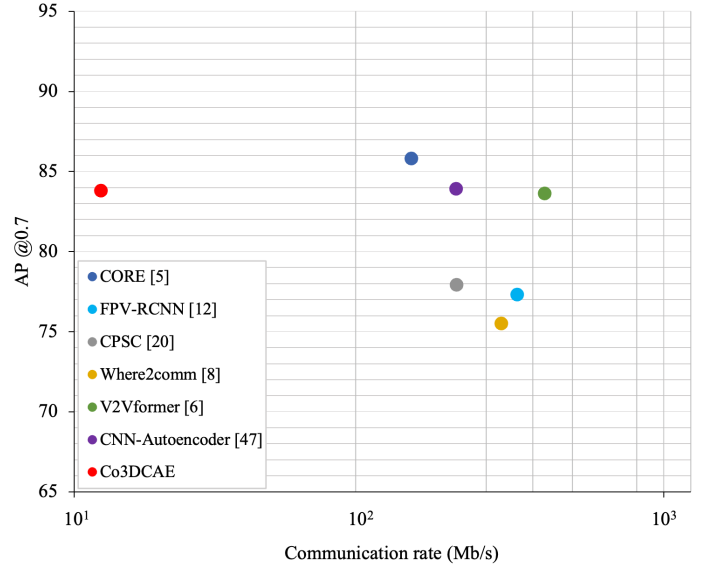


Fig. 1. Comparison of representative state-of-the-art methods benchmarked in terms of perception accuracy versus communication rate. The proposed method, Co3DCAE (represented by the red point), substantially reduces the communication rate while maintaining competitive perception accuracy compared with the direct exchange of high-dimensional sensing data.

nication, V2X) has become increasingly necessary to enhance perception accuracy and decision-making reliability [3]. In addition, given the high complexity of vehicular networks and limited bandwidth, communication overhead during the perception process is also a problem that needs to be addressed [4]. Although new technologies [5]–[8] for multi-agent collaborative perception have been proposed to address the trade-off between perception accuracy and bandwidth, real-world challenges remain unresolved, including object detection accuracy, redundant data transmission, and communication overload. Fig. 1 provides a motivational comparison of several representative methods benchmarked in terms of average perception accuracy (AP@0.7, y-axis) and communication rate (Mb/s, x-axis). Each point corresponds to one method, and the red point denotes the proposed Co3DCAE method. The figure shows that Co3DCAE provides a favorable trade-off between communication efficiency and detection accuracy; detailed experimental settings and benchmark descriptions are presented in later sections.

Enhancing object detection accuracy can be achieved through cooperative perception by sharing multimodal perceptual information among CAVs. Current CAVs are equipped with multiple sensors such as LiDAR, millimeter-wave radar, and cameras, which effectively expand the vehicle’s sensing

Copyright (c) 2026 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

This work was partially supported by the Spanish Research Agency (AEI) under project PID2020-117142GB-I00 of the call MCIN/AEI/10.13039/501100011033 and by the 6G-EWOC project from the Smart Networks and Services Joint Undertaking (SNS JU) under the European Union’s Horizon Europe research and innovation programme under Grant Agreement No. 101139182. (Corresponding author: Javier Ruiz-Hidalgo)

Mingrui Wang, Judit Salavedra Pujol, Josep R. Casas, Javier Ruiz-Hidalgo are with the Department of Signal Theory and Communications, UPC - Univ. Politècnica de Catalunya, 08034 Barcelona, Spain (e-mail: mingrui.wang@upc.edu; judit.salavedra@estudiantat.upc.edu; josep.ramon.casas@upc.edu; j.ruiz@upc.edu).

Mingrui Wang, Dongjie Li are with the Key Laboratory of Advanced Manufacturing and Intelligent Technology Ministry of Education, School of Mechanical and Power Engineering, Harbin University of Science and Technology, Harbin 150080, China (e-mail: mingrui.wang@upc.edu; dongjieli@hrbust.edu.cn).

modalities and field of view [9], but also generate large amounts of perception data. Blindly sharing all raw data or unoptimized intermediate features can lead to redundant transmission, wasted bandwidth and, for high-speed unmanned vehicles, perception delays of hundreds of milliseconds which may cause misjudgments or degrade the real-time performance of collaborative perception [10]–[12]. Therefore, an effective cooperative perception framework should not only improve detection accuracy by exploiting complementary information from multiple CAVs, but also determine what type of information should be exchanged and at which stage it should be fused. This motivates the study of fusion strategies, including raw-data-level fusion, feature-level fusion, and decision-level fusion, each of which provides a different trade-off between perception performance and communication efficiency.

Redundant data transmission and communication overload can be alleviated through feature compression [13], [14], targeted regional partitioning or weighted assignment of transmission features [15], [16], communication strategies [17], [18], and spatial filtering [19], [20]. However, due to the dynamic complexity of real traffic environments, these methods still face challenges related to semantic integrity, global collaboration, and positioning accuracy.

Building on the above approaches, we propose a generalizable autoencoder, Co3DCAE, which can be trained independently or integrated into an end-to-end intermediate feature fusion framework for collaborative perception. The proposed autoencoder is designed to better balance overall communication bandwidth and object detection precision. In this study, we contribute to three critical stages of collaborative perception. Specifically:

- 1) We propose a compression module, Co3DCAE, based on a residual-structured 3D convolutional autoencoder. This module is designed to jointly capture spatial and channel-wise information in intermediate feature maps, maximizing bandwidth compression while retaining essential feature information. The inclusion of 3D Res-blocks preserves critical information at high compression rates, achieving superior feature reconstruction quality and maintaining higher perception accuracy compared to existing direct compression methods.
- 2) We introduce a channel-wise training strategy for the autoencoder. This method optimizes feature compactness by leveraging cross-channel dependencies and sequentially minimizing reconstruction residuals along the channel dimension. Using this channel-wise strategy, we significantly reduce computational overhead and training costs, while enabling dynamic adjustment of the target latent channel dimensionality based on real-time bandwidth constraints without the need for retraining.
- 3) We present a comprehensive evaluation of three state-of-the-art (SOTA) collaborative perception algorithms AttFuse [21], AdaFusion [22] and TS-IFF [23] on three public 3D object detection benchmarks. These datasets include both synthetically simulated environments (OPV2V [21], CODD [24]) and real-world scenarios (V2V4Real [25]). Extensive experiments, including ablation studies, show that our proposed compression

framework consistently achieves a better trade-off between perception accuracy and bandwidth efficiency. Specifically, it reduces communication bandwidth requirements by a factor of 3-10 compared to existing SOTA approaches, while preserving comparable detection performance.

The remainder of this paper is organized as follows. Section II reviews the related work; Section III presents the formal problem formulation and describes the system model in detail; Section IV discusses extensive simulation results; Section V concludes the paper and outlines directions for future work.

II. RELATED WORK

A. V2V Collaborative Perception

To address the limitations of individual CAVs in complex environments, cooperative perception leveraging multi-vehicle collaboration has gained attention [26]. By sharing LiDAR and camera data, collaborative perception extends the perception range and improves detection precision for ego vehicles. Existing cooperative perception methods are generally categorized as early fusion [27], intermediate feature fusion [28], [29], and late fusion [30] according to the stage at which information is shared. Early fusion aggregates raw sensor data from multiple CAVs to provide a global view but requires transmitting large volumes of data and may cause heavy communication overhead. Late fusion exchanges local detection results, which reduces bandwidth consumption but strongly depends on the perception quality and computational capacity of each individual CAV, limiting the potential performance gain. Between these, intermediate feature fusion provides a favorable trade-off between perception accuracy and communication efficiency by transmitting compact semantic features rather than raw sensor data or final detection results [31], [32]. Based on intermediate feature fusion, F-Cooper [33] introduces max-value fusion for overlapping regions but overlooks low-confidence features, while CoFF [34] improves fusion by addressing this limitation. Attention-based models such as V2X-ViT [35] and CoBEVT [36] use transformer architectures for feature correlation, with the multi-scale Swin Transformer (MSwin) [37]–[39] capturing long-range spatial dependencies. AttFuse [21] applies self-attention for spatially precise fusion, and AdaFusion [22] provides trainable adaptive fusion methods. CORE [5] employs a learn-to-reconstruct framework that combines compression, collaboration, and reconstruction modules to mitigate scene incompleteness in single-vehicle perception, enhancing overall collaborative perception performance. Building on these advances, TS-IFF [23] introduces a two-stage intermediate-level feature fusion framework that combines pseudo-image and point cloud intermediate features channel-wise. This method effectively fuses multimodal sensor data and improves collaborative performance, particularly enhancing perception accuracy in complex scenarios where pedestrians and vehicles coexist.

B. Bandwidth Optimization and Compression

In collaborative perception systems, limited communication bandwidth is a major bottleneck that restricts practical deployment and hinders perception performance [40]. Because

large amounts of communication data (such as images, point clouds, and intermediate feature maps) must be shared among multiple vehicles, the high data throughput places significant pressure on communication resources. To address this issue, researchers have proposed various bandwidth optimization strategies from multiple perspectives. One category of methods focuses on designing adaptive communication strategies that improve bandwidth efficiency by flexibly controlling when data is transmitted and which targets receive it [41], [42]. Another category aims to reduce the volume of raw data transmitted by using compression [43], downsampling [44], or sparse encoding techniques [45].

Early multi-agent communication research primarily relied on manually defined communication protocols or heuristic strategies to guide information exchange among agents. Representative works such as When2com [7] and Where2comm [8] use handshake mechanisms or spatial confidence maps to enable communication only at necessary times or in critical areas thereby conserving resources. However, these methods generally lack flexibility and struggle to adapt to real-world applications as task complexity increases. In recent years, researchers have gradually shifted their focus to data compression methods based on neural network encoders [46]–[48].

Encoder architectures based on convolutional neural networks (CNNs) have demonstrated strong representational capacity and compression efficiency in practical applications, thereby improving generalization in dynamic environments. For example, Zhang et al. [13] proposed an efficient vehicular collaborative perception framework based on spatiotemporal feature compression, while Bai et al. [14] introduced a pillar attention encoder that adaptively enhances collaborative perception performance. In contrast, transformer-based models, such as V2V collaborative perception with spatial-channel transformer [6], focus on capturing long-range spatial and channel-wise dependencies, delivering improved collaborative perception performance in multi-agent scenarios. Although existing work has achieved initial success in bandwidth optimization, it still faces several limitations, including the loss of critical features during compression and insufficient mitigation of communication burden under bandwidth-constrained conditions. In collaborative perception, when wireless throughput is limited, large intermediate feature messages can increase transmission delay, leading to outdated collaborative information, temporal inconsistency, and degraded perception accuracy. Note that this work does not focus on physical-layer communication modeling, antenna design, wireless coding, or modulation. Instead, we assume the existence of a V2V/V2X wireless communication link whose delay remains within an application-acceptable range, and we abstract its communication capability using a per-second bandwidth budget. Under this setting, our proposed compression method aims to reduce the size of transmitted intermediate features while preserving key information, thereby alleviating the communication load and maintaining perception performance under bandwidth-constrained conditions.

Based on the above review, existing work mainly focuses on improving fusion performance or reducing communication overhead. However, fusion-oriented methods often lack ex-

plicit bandwidth-aware feature compression, while bandwidth-oriented methods may sacrifice semantic information under high compression ratios. In contrast, Co3DCAE provides a generalizable intermediate feature compression module that preserves key spatial-channel information, supports flexible bandwidth adaptation without retraining, and maintains detection accuracy while reducing communication costs.

III. METHODOLOGY

Due to the inherent sparsity of LiDAR data, intermediate feature maps often exhibit highly sparse feature distributions, with many regions containing little or no valid information. To address this, we propose Co3DCAE, a novel 3D convolutional autoencoder for intermediate feature compression. By encoding feature maps into a compact latent space, our method significantly reduces communication bandwidth while preserving essential feature details.

The overall workflow, shown in Fig. 2, involves four stages: (1) CAVs acquire local point clouds, project them to a unified frame, and extract intermediate features; (2) the features are compressed into a latent representation; (3) the compressed data is transmitted to a network server or vehicle for decoding and fusion; and (4) the detection results are broadcast back to participating CAVs.

A. Problem Description

This work considers a V2V/V2X collaborative perception scenario, where CAVs exchange perception-related messages over wireless vehicular communication links. Here, V2X is used as a general term covering different vehicular communication modes between vehicles and surrounding communication entities, while V2V specifically refers to direct communication among vehicles. The receiver of the collaborative features is abstracted as a fusion node. Depending on the deployment, the fusion node can be implemented either as an infrastructure-side edge server or roadside unit, or as a central vehicle. These deployment cases may involve different communication dynamics, such as channel conditions, scheduling mechanisms, interference, and transmission latency. The proposed method is not tied to a specific physical-layer communication technology, and can be applied to common vehicular communication technologies such as 5th-Generation Mobile Communication Technology (5G) [49], 5G-Advanced [50], Cellular Vehicle-to-Everything (C-V2X) [51], and 5G New Radio Vehicle-to-Everything (5G NR-V2X) [52]. Instead of modeling low-level communication procedures, such as antenna design, channel coding, medium access control, resource scheduling, or retransmission, we abstract the communication capability using the available communication-rate budget B . The main transmitted messages considered in this work are BEV intermediate feature maps and their compressed latent representations. Specifically, each CAV_{*i*} transmits its compressed latent representation φ_i to the designated fusion node for decoding, feature fusion, and detection. Unless otherwise specified, we model this process as a unicast V2X feature-payload transmission from each CAV

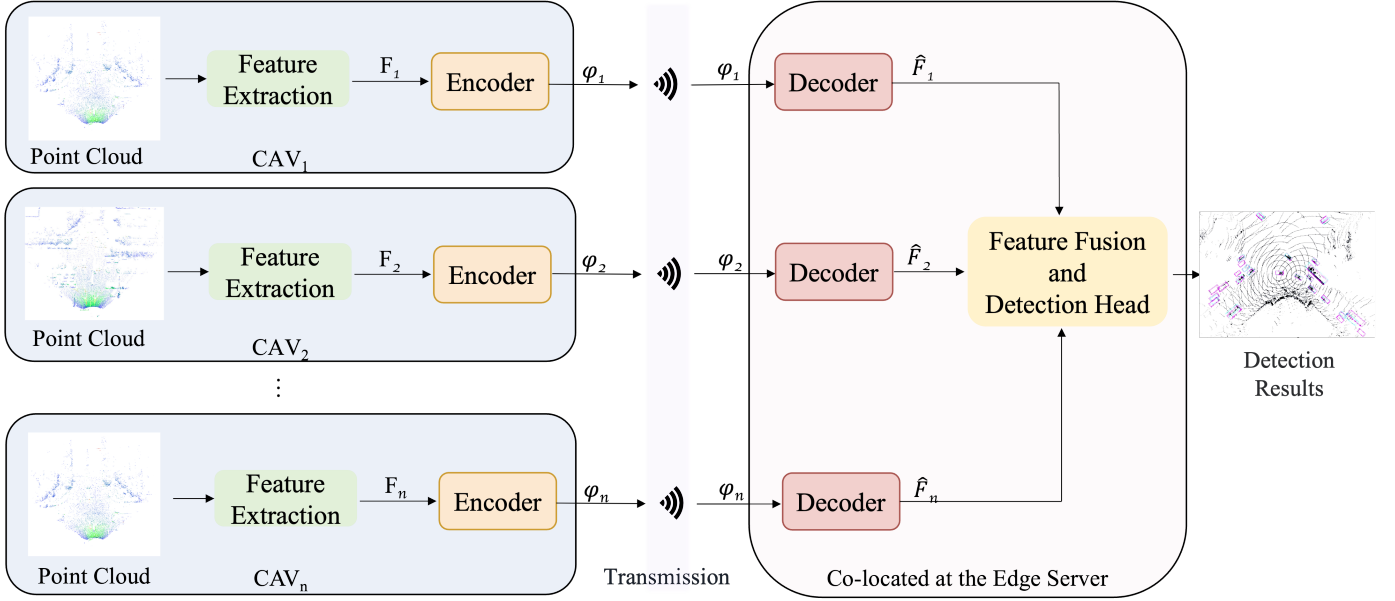


Fig. 2. Overview of the workflow. The raw point cloud data from each CAV is processed to generate intermediate feature maps F_i where i denotes the CAV index. These feature maps are then compressed using the proposed encoder, producing compact latent representations denoted as φ_i . The compressed latent representations are sent to a network server or central vehicle, which decodes them into reconstructed features $\hat{F}_i$. A feature fusion module, followed by a single-stage detection head, performs multi-object detection and sends the detections back to the CAVs. The modules enclosed in the individual CAV_i blocks are co-located on the corresponding CAV, whereas the modules inside the "Edge-server" block are deployed centrally on the server.

TABLE I
SUMMARY OF MAIN NOTATIONS.

Notation	Description
CAV_i	The i -th connected and automated vehicle
n	Number of collaborating CAVs
$\hat{F}_i$	Decompressed intermediate feature of CAV_i
φ_i	Compressed latent representation of CAV_i
$\dim(\varphi_i)$	Number of transmitted latent feature components
$\mathcal{M}_i$	Perception performance score of CAV_i
$\mathcal{M}_{\text{total}}$	Total system perception performance score
B	Available per-second communication bandwidth budget
R_{forward}^i	Forward latent-feature transmission rate
R_{back}^i	Backward detection-result transmission rate

to the fusion node. When the fusion node is an infrastructure-side edge server or roadside unit, this abstraction corresponds to an infrastructure-assisted V2X transmission. When the fusion node is an ego or central vehicle, it corresponds to a V2V unicast transmission. In both cases, our communication-cost analysis focuses on the application-layer feature payload rate required by collaborative perception. Raw point clouds, camera images, protocol-dependent overhead, and final detection results are not the main focus of the communication-cost analysis, since this work focuses on the feature-level communication rate introduced by collaborative perception. To improve readability, the main notations used in the problem formulation and communication-rate analysis are summarized in Table I.

Let us define the collaborative perception performance score of CAV_i as:

$$\mathcal{M}_i = \mathcal{F}(\mathcal{D}_e(\hat{F}_i), g_t) \quad (1)$$

where i indexes the CAVs in the system, $\hat{F}_i$ denotes the decompressed intermediate features received by the server

from CAV_i , $\mathcal{D}_e(\cdot)$ represents the detection model parameters, g_t denotes the ground truth annotations, and $\mathcal{F}(\cdot, \cdot)$ is the evaluation metric function, where a larger value indicates better performance.

The total system perception performance, $\mathcal{M}_{\text{total}}$, is defined as the sum of the evaluation scores across all n CAVs:

$$\mathcal{M}_{\text{total}} = \sum_{i=1}^n \mathcal{M}_i \quad (2)$$

The primary optimization goal is to maximize the total system perception performance while ensuring that the compressed and reconstructed feature maps still support accurate collaborative detection:

$$\arg \max_{\mathcal{D}_e(\cdot), \{\hat{F}_i\}} \mathcal{M}_{\text{total}} \quad (3)$$

Given the available per-second communication bandwidth budget, the total communication rate of all collaborating CAVs must satisfy:

$$\sum_{i=1}^n \frac{\dim(\varphi_i) \cdot b \cdot f}{10^6} \leq B \quad (4)$$

Here φ_i denotes the compressed latent representation transmitted by CAV_i , and $\dim(\varphi_i)$ represents the number of latent feature components transmitted by CAV_i in each frame. b is the bit precision of each latent feature component, and f is the sensor frame rate. Therefore, $\dim(\varphi_i) \cdot b \cdot f$ denotes the transmission rate of CAV_i in bits per second. n is the number of collaborating CAVs, and B denotes the available communication bandwidth budget per second, measured in Mb/s.

Combining the above, the complete optimization problem can be formulated as:

$$\begin{aligned} & \arg \max_{\mathcal{D}_e(\cdot), \{\hat{F}_i\}} \mathcal{M}_{\text{total}} \\ & \text{s.t.} \quad \sum_{i=1}^n \left(\dim \left(\frac{\dim(\varphi_i) \cdot b \cdot f}{10^6} \right) \right) \leq B \end{aligned} \quad (5)$$

The bandwidth consumed by the edge server when transmitting detection results back to CAV_i is much smaller than that of the forward feature transmission. The forward transmission contains the compressed latent feature representation φ_i , with a transmission rate of $R_{\text{forward}}^i = \dim(\varphi_i) \cdot b \cdot f$, where b denotes the number of bits used to represent each latent feature value and f denotes the frame rate, i.e., the number of transmitted frames per second. In contrast, the backward transmission contains only compact detection outputs, such as bounding boxes, confidence scores, and class labels.

Following the detection-head output format, each 3D bounding box is represented by seven offsets $(x, y, z, w, l, h, \theta)$ together with c class confidence scores, so each detection result contains $d_{\text{det}} = 7 + c$ components. Assuming that N_{obj} objects are detected, the backward transmission rate can be estimated as $R_{\text{back}}^i = N_{\text{obj}} \cdot d_{\text{det}} \cdot b \cdot f$. Since $N_{\text{obj}} \cdot d_{\text{det}} \ll \dim(\varphi_i)$ in typical collaborative perception scenarios, we have $R_{\text{back}}^i \ll R_{\text{forward}}^i$. As a numerical example, when $N_{\text{obj}} = 20$ and $c = 2$ in a two-class detection setting with vehicles and pedestrians, we have $d_{\text{det}} = 7 + 2 = 9$, and thus $N_{\text{obj}} \cdot d_{\text{det}} = 20 \times 9 = 180$. In comparison, a compact transmitted latent feature tensor with size $8 \times 75 \times 75$ contains 45,000 components. Since $180 \ll 45,000$, the backward transmission of detection results has a size much smaller than the forward latent-feature transmission under the same bit precision b and frame rate f .

Therefore, the bandwidth consumed by backward transmission of results is negligible compared with forward transmission of features. We propose a generic intermediate feature autoencoder, agnostic to the specific collaborative perception system, designed to balance communication bandwidth efficiency and perception accuracy.

B. Feature Extraction and Projection

Intermediate features are widely used for sharing information, as they efficiently balance semantic richness and data volume [6], [13], [21]–[23]. Each CAV first processes local point cloud data with a feature extraction network to generate intermediate feature maps. Following SOTA practice, the framework assumes spatially calibrated and synchronized data transmission, and uses a representative CAV to define a unified pose coordinate frame for feature projection.

We generalize the intermediate features extracted from different collaborative CAVs $F_i \in \mathbb{R}^{C \times H \times W}$. Here F_i denotes the BEV intermediate feature map of CAV_i rather than raw sensor data. The BEV representation describes the driving scene from a top-down perspective by projecting sensor observations or encoded voxel features onto a ground-plane tiling, and it is a commonly used representation in many perception systems. The dimensionality of F_i is given by C , the number

of feature channels, while H and W denote the height and width of the BEV tiling grid obtained after projection onto the ground plane. This BEV intermediate feature representation preserves spatial layout information and facilitates feature alignment and fusion among multiple CAVs. The resulting spatially and temporally aligned feature tensor F then serves as the input to the Co3DCAE feature compression module.

C. Feature Compression

The encoder and decoder modules (Fig. 2) form the core of the compression system. To reduce bandwidth consumption during collaborative perception, each CAV independently compresses its intermediate features before transmission. We propose a 3D convolutional residual autoencoder (Co3DCAE) that jointly captures spatial and channel-wise information. Unlike 2D convolution, which processes only spatial dimensions, 3D convolution operates across both spatial and channel dimensions, enabling more comprehensive learning of high-dimensional intermediate feature maps and effectively encoding them across channel and spatial dimensions. The overall architecture is shown in Fig. 3.

The encoder maps the intermediate feature maps $F \in \mathbb{R}^{C \times H \times W}$ to a latent representation $\varphi \in \mathbb{R}^{C_L \times H_L \times W_L}$. H_L and W_L are fixed spatial dimensions of the latent representation, while C_L (the number of channels) may vary during inference. The encoder minimizes the bandwidth usage of φ while preserving critical information between the original features F and the reconstructed features $\hat{F}$.

Co3DCAE incorporates 3D residual blocks (3D Res-blocks) in both the encoding and decoding stages. The channel dimension (C_L) is treated differently from the spatial dimensions (H_L and W_L) because the channel axis encodes semantic abstraction and allows finer control over compression granularity, preserving richer features and maximizing the recovery of spatial and contextual information.

The 3D Res-block design, inspired by ResNet, includes shortcut connections and identity mapping to preserve input information and mitigate degradation in deep networks. The block processes the tensor through three parallel 3D convolutional branches: one for latent feature extraction over the channel-dimension, and two focusing on spatial latent feature extraction along the height and width dimensions (Fig. 4). All extracted latent features are concatenated and passed to subsequent encoding layers.

Co3DCAE can be trained independently or jointly with the collaborative perception system. We employ two training methods: single-stage and multi-stage.

1) *Single-stage training*: This method (Fig. 5) uses a fixed latent representation dimension, $C_L \times H_L \times W_L$, and enables joint training of the entire collaborative system with Co3DCAE. This end-to-end training has a simpler structure and high training efficiency, achieving an optimal balance between intermediate feature compression and perception accuracy in most scenarios. However, the resulting compressed bandwidth is fixed, limiting adaptability without retraining.

2) *Multi-stage training*: This method uses a progressive training scheme along the channel dimension (Fig. 6). In

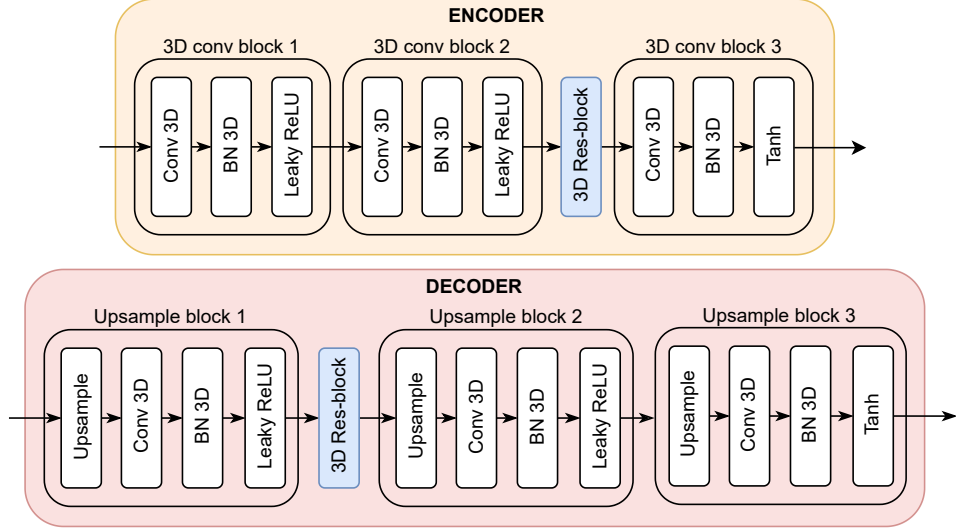


Fig. 3. Architecture overview of Co3DCAE. The network is an autoencoder composed of an encoder and a decoder, each with three 3D-CNN modules and one 3D Res-block (in blue). The encoder maps the input intermediate feature maps $F \in \mathbb{R}^{C \times H \times W}$ to a latent representation $\varphi \in \mathbb{R}^{C_L \times H_L \times W_L}$. The decoder mirrors the encoder structure, with three 3D-CNN upsampling modules and one 3D Res-block.

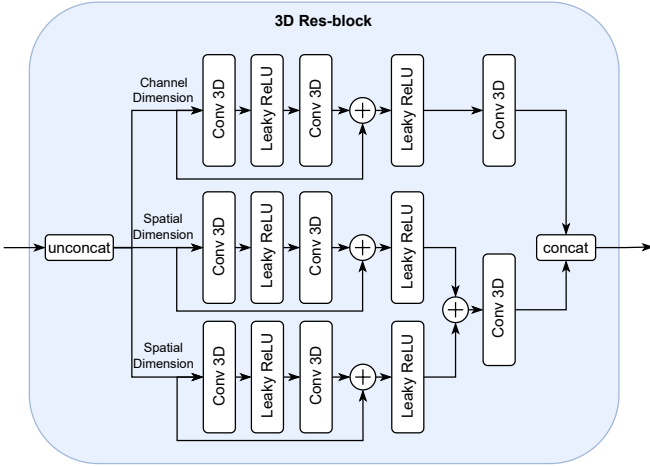


Fig. 4. Architecture of the proposed 3D Res-block. The 3D Res-block module processes the tensor through three parallel 3D convolutional branches: one branch operates along the channel dimension to extract latent features, while the other two branches focus on spatial latent extraction along the height and width dimensions, respectively. All latent features are concatenated and then passed to the subsequent encoder layers for further processing.



Fig. 5. Illustration of the single-stage training method.

the first stage, the model compresses the input feature map $F = F^{(1)}$ into the first latent channel $\varphi^{(1)}$ and reconstructs the entire feature tensor. From the second stage onward, the residual error between the original tensor and the reconstructed tensor from the previous stage $F^{(i+1)} = F^{(i)} - \hat{F}^{(i)}$ is compressed into the next latent channel. This progressive process reduces residual errors and captures inter-channel dependencies while reducing the total number of parameters,

as all stages share encoder and decoder parameters.

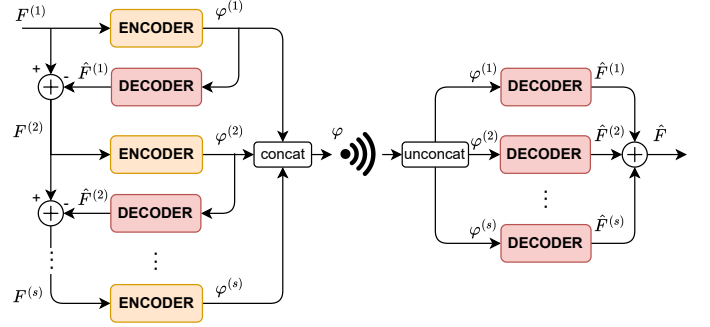


Fig. 6. Illustration of the multi-stage training method.

The final latent representation, φ , is formed by concatenating all individual latent channels. Depending on the available bandwidth, the number of latent channels transmitted s can be selected between 1 and C_L :

$$\varphi \in \{\varphi^{(1)}, \varphi^{(2)}, \dots, \varphi^{(s)}\}. \quad (6)$$

The final reconstructed feature is expressed as:

$$\hat{F} = \sum_{i=1}^s \hat{F}^{(i)}, \quad 1 \leq s \leq C_L. \quad (7)$$

Although this approach prevents joint training, it allows flexible adjustment of the latent channel dimensionality s during inference based on practical bandwidth constraints.

D. Feature Decoding and Fusion Mechanism

The decoder reconstructs high-resolution intermediate feature maps from the compressed latent representation. Its structure (Fig. 3) consists of three 3D upsampling modules and one 3D Res-block. The decoding process depends on

the training method: single-stage training uses one trained decoder, whereas multi-stage training employs a shared decoder to reconstruct all s channels sequentially. The 3D Res-block in the decoder mirrors the encoder's structure, ensuring symmetry and effective feature reconstruction. The reconstructed intermediate features are then passed to the collaborative perception framework for further fusion and analysis.

E. Loss Function

The proposed autoencoder can be trained independently or jointly with the full collaborative perception system. In the case of joint training, we also use the independent training as a pretraining strategy.

1) *Independent Training*: Co3DCAE is trained as an independent feature compression module, using the Mean Squared Error (MSE) between the original feature maps F and the reconstructed feature maps $\hat{F}$ as the reconstruction loss. The autoencoder is trained once on the entire feature map tensor, enabling it to learn a unified compression for all channels. The reconstruction loss $\mathcal{L}_D$ is defined as:

$$\mathcal{L}_D = \frac{1}{CHW} \sum_{x=1}^C \sum_{y=1}^H \sum_{z=1}^W \left(F(x, y, z) - \hat{F}(x, y, z) \right)^2 \quad (8)$$

where F and $\hat{F}$ are the original and reconstructed feature maps and C, H, W , their channel, height and width dimensions.

2) *Joint Training*: After pretraining, the model is integrated into an end-to-end collaborative perception framework for joint training, aiming to simultaneously optimize feature compression quality and 3D object detection performance. The overall multi-task loss combines the reconstruction loss with the perception loss of the collaborative perception system:

$$\mathcal{L}_{\text{total}} = \lambda_D \mathcal{L}_D + \mathcal{L}_P \quad (9)$$

$\mathcal{L}_D$ denotes the reconstruction loss measured by MSE, whereas $\mathcal{L}_P$ represents the perception loss of the collaborative system, which typically combines a classification loss for 3D object detection and a regression loss for bounding box localization [53]. λ_D is a hyperparameter that controls the influence of the reconstruction loss. To ensure a fair comparison across different fusion frameworks, we use the same loss $\mathcal{L}_P$ for all methods evaluated in our experiments, including AttFuse [21], AdaFusion [22] and TS-IFF [23].

IV. EXPERIMENTAL RESULTS

We adopt a unified model training strategy without task-specific adjustments and conduct targeted experiments on the proposed compression framework across diverse traffic environments and detection objects, using different SOTA collaborative perception methods. We select publicly available synthetic and real-world datasets and design a series of comparative and ablation studies to comprehensively demonstrate the advantages of the collaborative perception model with feature compression over traditional approaches. The autoencoder is trained on intermediate feature maps extracted from raw LiDAR point clouds by the PointPillars [53] backbone, corresponding to the first feature extraction block in

Fig. 2. These intermediate feature maps serve as a unified input representation across all datasets, ensuring consistent feature compression and reconstruction. The following section describes the characteristics of the OPV2V, CODD, and V2V4Real datasets used for evaluation.

A. Datasets

1) *OPV2V* [21]: This dataset consists of two subsets, Default Towns (DT) and Culver City (CC). The DT subset contains data from eight default towns provided by CARLA [54], with each frame including an average of about 3 collaborative autonomous vehicles (CAVs), with a minimum of 2 and a maximum of 7. The data in this subset is split into a training set (6.7K frames), a validation set (2K frames), and a test set (2.7K frames). The CC subset contains an independent test set of 550 frames, used to evaluate the model's generalization ability in new scenarios. Each vehicle's driving scene lasts approximately 16.4 seconds, captured using a 64-channel LiDAR, that generates about 1.3 million point cloud data points per second. We use this dataset as the main benchmark to validate the recognition performance and bandwidth efficiency of the proposed model in collaborative perception tasks.

2) *CODD* [24]: This dataset contains 108 scene segments from 8 different CARLA towns. Each scene consists of 125 frames, with the first 100 frames used for training and the remaining 25 frames for testing. It includes varying numbers of vehicles and pedestrians, with vehicle counts ranging from 4 to 15 and pedestrian counts from 2 to 8. CODD provides LiDAR data for all vehicles and precise 3D bounding box annotations for both vehicles and pedestrians. Notably, CODD is currently the only collaborative perception dataset that includes pedestrian categories. Therefore, we use it as a supplementary dataset primarily to validate detection performance on small targets such as pedestrians.

3) *V2V4Real* [25]: This dataset is the first large-scale publicly available real-world dataset for V2V collaborative perception, collected in Columbus, Ohio, and includes both highways and urban streets. It contains 19 hours of driving data and 310K frames, from which 67 representative scenarios are selected. LiDAR and RGB frames are sampled at 10Hz, generating 20K LiDAR point clouds and 40K images. The dataset features 32-channel LiDAR point clouds and 240K annotated 3D bounding boxes across 5 categories. In the experiments, 14,210 / 2,000 / 3,986 samples from the V2V4Real dataset are used for model training, validation, and testing, respectively.

B. Training and Inference Details

1) *Co3DCAE Pre-training*: We first pre-train Co3DCAE independently to optimize feature compression and reconstruction. Training is performed entirely using the PyTorch framework [55] on a GeForce RTX 3090 GPU with 24GB RAM, using CUDA v11.1 and cuDNN v8.0. We use the Adam optimizer with parameters $\varepsilon = 0.1$, weight decay 10^{-4} , and momentum in the range [0.85, 0.95]. A multi-step learning rate scheduler is applied with milestones at epochs [20, 30] and a decay factor of 0.1. Early stopping based on validation loss is used to prevent overfitting.

TABLE II

PERFORMANCE EVALUATION OF THE PROPOSED AUTOENCODER INTEGRATED INTO REPRESENTATIVE COLLABORATIVE PERCEPTION ALGORITHMS ON THE OPV2V AND CODD VIRTUAL DATASETS WITH 7 CAVs. BOLD VALUES INDICATE THE BEST RESULTS. $\uparrow$: HIGHER IS BETTER; $\downarrow$: LOWER IS BETTER.

Dataset	OPV2V			CODD	
Method/Index	DT		Comm.	Comm.	
	AP@0.5 ($\uparrow$)	AP@0.7 ($\uparrow$)	(Mb/s) ($\downarrow$)	AP@0.1 ($\uparrow$)	(Mb/s) ($\downarrow$)
AttFuse [21]	90.8	81.5	532.5	38.1	324.0
AttFuse [21] + Co3DCAE	85.5	75.1	66.6	27.4	22.2
AdaFusion [22]	91.6	85.6	699.0	45.2	345.4
AdaFusion [22] + Co3DCAE	88.2	78.8	87.5	33.8	43.2
TS-IFF [23]	94.1	89.3	1200.0	63.8	2320.0
TS-IFF [23] + Co3DCAE	92.0	86.1	302.4	55.1	289.0

2) End-to-End Collaborative Training with Co3DCAE:

After pre-training, the autoencoder is integrated into the full collaborative perception pipeline and fine-tuned jointly with other modules in an end-to-end manner, with the goal of simultaneously optimizing communication compression and perception performance. At this stage, compressed features from multiple CAVs are received by a network server. The features are decompressed and fused, and detections from the fused features are sent back to the CAVs. Driving scenes are randomly selected from the training set, and the number of participating CAVs is uniformly sampled from the range [2, 7]. The range [2, 7] follows the original OPV2V configuration, where each scenario contains 2–7 available CAVs, and is consistent with representative baselines such as AttFuse, AdaFusion, and TS-IFF. For fair comparison, the number of CAVs is kept fixed within each evaluation group. The Adam optimizer is used with the same hyperparameters. The model jointly minimizes both reconstruction loss and detection loss.

The LiDAR point cloud ranges are configured as follows: At each frame, all LiDAR point clouds from the selected CAVs are preprocessed to a common range relative to the ego vehicle coordinate system. Points falling outside this range are clipped, and bounding boxes beyond these limits are discarded, restricting evaluation to the vicinity of the ego vehicle:

- OPV2V and V2V4Real: $x, y, z \in [-140.8 \text{ m}, 140.8 \text{ m}] \times [-40 \text{ m}, 40 \text{ m}] \times [-3 \text{ m}, 1 \text{ m}]$
- CODD: $x, y, z \in [-140.8 \text{ m}, 140.8 \text{ m}] \times [-40 \text{ m}, 40 \text{ m}] \times [-6 \text{ m}, 4 \text{ m}]$

Finally, the point cloud is voxelized into pseudo-images with a resolution of $0.4 \text{ m} \times 0.4 \text{ m}$ in the x and y directions.

3) *Inference and Evaluation*: Consistent with existing SOTA methods, we use a confidence threshold of 0.2 and apply NMS with an IoU threshold of 0.2 during inference. For evaluation, we report AP@0.5 and AP@0.7 for vehicle detection, and AP@0.1 for pedestrian detection. The reported results follow the dataset-specific cooperative perception settings: OPV2V and CODD use seven participating CAVs, while V2V4Real uses two CAVs for communication-rate calculation.

The communication metric Comm. in Table II and subsequent tables follows the bandwidth constraint in Eq. (4). All communication rates reported in this paper represent the aggregate feature-level transmission rate of all participating vehicles. Specifically, the total transmission rate is computed as $\sum_{i=1}^n \dim(\varphi_i) \cdot b \cdot f$, where $\dim(\varphi_i) = C_L \times H_L \times W_L$.

The reported communication values are calculated using the corresponding latent channel number C_L under each setting. An additional factor of $\frac{1}{10^6}$ is applied to convert bits/s to Mb/s.

To further analyze processing latency, we estimate the transmission delay using an assumed available V2X link capacity of 10 Mb/s. This value is used only as an illustrative bandwidth setting for transmission-delay estimation and is not intended to represent the capacity of a specific V2X technology or deployment condition. The assumed 10 Mb/s link capacity is used only for transmission-delay estimation and is distinct from the Comm. values defined above, which quantify the communication-rate demand in the performance evaluation. An uncompressed collaborative perception method such as V2VNet [56] transmits about 1.07 Mb of data per frame on the OPV2V dataset. Under the assumed 10 Mb/s link capacity, this corresponds to approximately 107 ms of transmission delay. In contrast, our channel-wise autoencoder compression reduces the transmitted data size to about 0.106–0.117 Mb per frame while maintaining comparable detection accuracy, corresponding to only 10.6–11.7 ms of transmission delay under the same link-capacity setting.

In addition, the feature compression time of Co3DCAE is about 13 ms on an NVIDIA GeForce RTX 3090 GPU with 24 GB of memory. These results indicate that the proposed method effectively reduces communication latency while introducing only moderate processing overhead. Unless otherwise stated, all inference results are obtained using the end-to-end trained model. Additional training configurations and network settings follow the open-source implementations provided by OPV2V [21], CODD [24], and V2V4Real [25].

C. Quantitative Evaluation

Building on the previous evaluations, we further assess the generalization capability of Co3DCAE by integrating it into three representative intermediate feature fusion frameworks: AttFuse [21], AdaFusion [22], and TS-IFF [23]. The experiments are conducted on the OPV2V and CODD datasets to evaluate vehicle and pedestrian detection performance under communication bandwidth constraints.

As shown in Table II, Co3DCAE consistently reduces the communication rate when integrated with different collaborative perception frameworks. On the OPV2V dataset, the communication rates of AttFuse and AdaFusion decrease from

532.5 Mb/s and 699.0 Mb/s to 66.6 Mb/s and 87.5 Mb/s, respectively, corresponding to approximately an $8\times$ reduction. For TS-IFF, the communication rate is reduced from 1200.0 Mb/s to 302.4 Mb/s, while still maintaining high detection accuracy. Although the AP values decrease slightly after compression, the compressed models preserve competitive performance, indicating that Co3DCAE effectively removes redundant feature information while retaining key semantic cues for vehicle detection.

A similar trend is observed on the Codd dataset. After integrating Co3DCAE, the communication rates of AttFuse, AdaFusion, and TS-IFF are reduced from 324.0, 345.4, and 2320.0 Mb/s to 22.2, 43.2, and 289.0 Mb/s, respectively. This demonstrates that Co3DCAE also generalizes to pedestrian detection scenarios. Among the evaluated frameworks, TS-IFF combined with Co3DCAE achieves the best AP@0.1 of 55.1 while reducing the communication rate by approximately $8\times$. Overall, these results show that Co3DCAE provides a consistent and effective trade-off between perception accuracy and communication efficiency across different datasets and baseline fusion frameworks.

Co3DCAE achieves substantial communication bandwidth reductions ranging from 75% to over 93% across different collaborative perception methods, with controllable accuracy degradation, and demonstrates particularly robust performance when integrated with TS-IFF. Specifically, TS-IFF with Co3DCAE reduces bandwidth by 75% on OPV2V with only 2.23% (AP@0.5) and 3.59% (AP@0.7) relative performance drops, achieving a favorable balance between detection accuracy and bandwidth efficiency. Similar trends are also observed for AttFuse and AdaFusion, where significant bandwidth compression leads to only limited performance degradation, as illustrated in Fig. 7.

Fig. 7 presents the evaluation results of Co3DCAE under a wider range of communication bandwidth constraints, highlighting its generalization capability. The curves show the detection performance when Co3DCAE is incorporated into three representative compression-based collaborative perception frameworks and indicate the average precision achieved under different bandwidth conditions. This demonstrates that the proposed Co3DCAE can be effectively integrated into different collaborative perception algorithms to compress exchanged features under varying communication bandwidth constraints.

We further compare Co3DCAE with several SOTA collaborative perception and communication-efficient methods on the OPV2V dataset. In this evaluation, Co3DCAE is integrated with the TS-IFF fusion framework, and different numbers of latent channels represent different communication bandwidth levels. As shown in Table III, Co3DCAE consistently achieves a favorable balance between detection accuracy and communication cost on both the DT and CC subsets. When the latent channel number is set to 24, Co3DCAE achieves the best overall detection performance, with AP@0.5/AP@0.7 of 92.0/86.1 on DT and 88.2/79.9 on CC, while requiring only 302.4 Mb/s communication bandwidth. Compared with CORE, which reports 90.9/85.8 on DT and 87.7/78.1 on CC with 336.0 Mb/s, Co3DCAE achieves higher detection

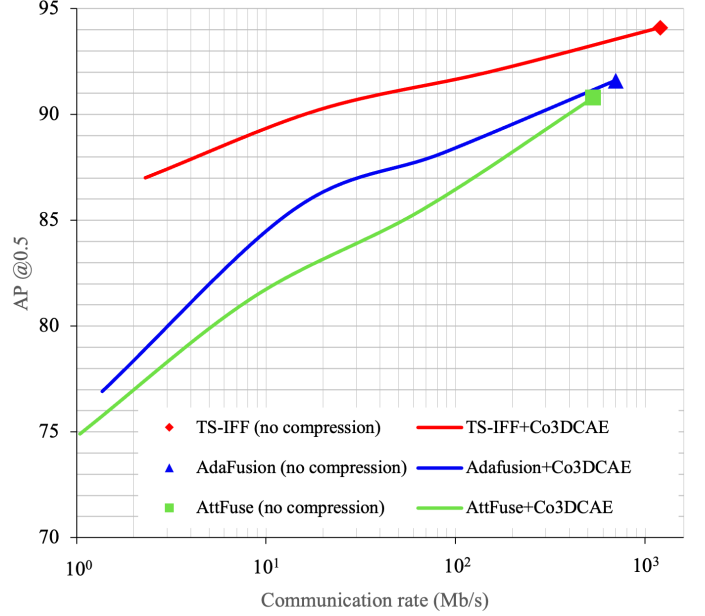


Fig. 7. Co3DCAE optimizes the trade-off between bandwidth and average precision across three representative intermediate feature fusion frameworks with 7 CAVs, and demonstrates clear advantages in achieving a balanced compromise between detection performance and bandwidth efficiency. For visual comparison, the communication rate is plotted on a logarithmic scale. Viewing in color is recommended for improved clarity.

accuracy with lower communication cost. Compared with V2VFormer, Co3DCAE also obtains better AP results while reducing the bandwidth from 565.9 Mb/s to 302.4 Mb/s.

Under more constrained bandwidth settings, Co3DCAE with 16 and 8 latent channels still maintains competitive detection accuracy while reducing the communication rate to 201.6 Mb/s and 100.8 Mb/s, respectively. This demonstrates that the proposed compression module can flexibly adapt to different bandwidth budgets by adjusting the latent channel size. The results in Table III show that Co3DCAE not only achieves SOTA or comparable perception accuracy, but also reduces communication cost, confirming its effectiveness in balancing perception quality and bandwidth efficiency.

To further evaluate the robustness of Co3DCAE across different dataset types, Table IV reports comparison results on the simulated Codd dataset and the real-world V2V4Real dataset. Unlike Table III, which focuses on the OPV2V simulated benchmark, Table IV further examines whether Co3DCAE can maintain its bandwidth-accuracy trade-off under more diverse scenarios. On the real-world V2V4Real dataset, Co3DCAE achieves the best AP@0.5/AP@0.7 scores of 67.8/32.8 while requiring only 75.7 Mb/s communication bandwidth. Compared with V2VFormer, CPSC, and CORE, Co3DCAE not only improves detection accuracy but also substantially reduces the communication rate. For example, compared with CPSC, Co3DCAE improves AP@0.5 from 66.3 to 67.8 and AP@0.7 from 29.6 to 32.8, while reducing communication from 366.0 Mb/s to 75.7 Mb/s. On the Codd dataset, Co3DCAE obtains the best AP@0.1 of 55.1 with only 289.0 Mb/s, outperforming the competing methods in both pedestrian detection accuracy and communication efficiency.

TABLE III

COMPARISON OF Co3DCAE WITH SOTA METHODS ON THE OPV2V VIRTUAL DATASET IN TERMS OF 3D OBJECT DETECTION PERFORMANCE AND BANDWIDTH. Co3DCAE IS EVALUATED WITH DIFFERENT NUMBER OF LATENT CHANNELS (C_L), CORRESPONDING TO DIFFERENT COMMUNICATION BANDWIDTHS. BOLD VALUES INDICATE THE BEST RESULTS, UNDERLINED VALUES THE SECOND-BEST. $\uparrow$: HIGHER IS BETTER; $\downarrow$: LOWER IS BETTER.

Dataset	OPV2V					
Method/Index	Channels (C_L)	DT		CC		Comm. (Mb/s) $\downarrow$
		AP@0.5 $\uparrow$	AP@0.7 $\uparrow$	AP@0.5 $\uparrow$	AP@0.7 $\uparrow$	
Where2comm [8]	–	88.9	75.5	82.2	68.0	477.4
FPV-RCNN [29]	–	82.0	77.3	76.3	72.0	508.2
CPSC [13]	–	89.1	77.9	–	–	401.3
V2Vformer [6]	–	89.5	83.6	84.4	78.6	565.9
CNN-Autoencoder [46]	–	–	83.9	–	77.5	400.0
CORE [5]	–	<u>90.9</u>	<u>85.8</u>	<u>87.7</u>	78.1	336.0
Co3DCAE	8	90.5	83.8	86.5	78.1	100.8
Co3DCAE	16	90.7	84.1	87.5	<u>78.8</u>	<u>201.6</u>
Co3DCAE	24	92.0	86.1	88.2	79.9	302.4

Overall, the results in Table IV demonstrate that Co3DCAE generalizes well to both simulated and real-world scenarios. It achieves the best detection performance on both datasets while maintaining the lowest communication cost, confirming its effectiveness for bandwidth-efficient collaborative perception in complex and real driving environments.

TABLE IV

PERFORMANCE COMPARISON BETWEEN Co3DCAE AND SOTA METHODS ON V2V4REAL AND CODD. THE BEST RESULTS ARE IN BOLD, AND THE SECOND-BEST ARE UNDERLINED. $\uparrow$: HIGHER / $\downarrow$: LOWER IS BETTER.

Method/Index	V2V4Real		CODD	
	AP@0.5/@0.7 ($\uparrow$)	Comm. ($\downarrow$)(Mb/s)	AP@0.1 ($\uparrow$)	Comm. ($\downarrow$)(Mb/s)
V2Vformer [6]	63.1/30.7	439.0	41.5	821.9
CPSC [13]	<u>66.3</u> /29.6	366.0	29.4	<u>696.8</u>
CORE [5]	61.0/ <u>31.7</u>	<u>328.0</u>	<u>44.2</u>	783.1
Co3DCAE	67.8/32.8	75.7	55.1	289.0

D. Qualitative Evaluation

Figs. 8 present qualitative experimental results on the OPV2V dataset involving only two CAVs. Although most methods can detect surrounding vehicles (highlighted with red circles in the figure) false positives and missed detections still occur in scenarios with communication bandwidth limitations. To address this issue, we apply varying levels of compression using Co3DCAE when transmitting intermediate features from the CAVs. After decoding and fusion at the edge server, the proposed method achieves accurate object detection and bounding box prediction while minimizing communication bandwidth usage. In complex real-world traffic scenarios (e.g., intersections), as shown in Fig. 9, our model maintains robust and consistent perception performance across consecutive timesteps. Despite dynamic changes in the environment, such as moving vehicles, varying traffic densities, and occlusions, the system is still able to deliver accurate object detection results with only minor delays after network bandwidth compression using Co3DCAE. The overall real-time performance loss remains within acceptable limits, ensuring that the system can meet the low-latency requirements of autonomous driving.

E. Ablation Study

To validate the model, we conduct a series of ablation experiments focusing on the impact of four key factors on perception performance: compression ratio, compression methods, autoencoder training strategy, and the number of collaborating CAVs. Each experiment is designed to isolate and evaluate the effect of a single factor, thereby verifying the reliability and robustness of our approach.

Compression Ratio: To examine the effect of the compression, we vary the number of filters in the last 3D convolution block of the encoder, which determines the channel size of the compressed latent representation. Table V reports TS-IFF results with and without Co3DCAE on the V2V4Real dataset. The original TS-IFF uses 128 channels and requires 605.6 Mb/s of communication bandwidth, achieving 68.2/40.1 AP@0.5/AP@0.7. After applying Co3DCAE, the communication cost is reduced. With 64 channels, the communication rate drops to 75.7 Mb/s, while AP@0.5 remains close to the baseline at 67.8. With 32 channels, the communication rate is further reduced to 9.5 Mb/s, and the model still achieves 66.3/38.0 AP@0.5/AP@0.7. When the number of channels is reduced to 16 or 8, the communication cost becomes extremely low, reaching 1.2 Mb/s and 0.2 Mb/s, respectively, but detection accuracy decreases noticeably.

These results illustrate how latent channel size affects the balance between communication cost and detection accuracy. Increasing the number of channels allows more feature information to be retained, which generally improves AP but also increases bandwidth consumption. Conversely, very small channel settings greatly reduce transmitted data throughput, but the loss of feature detail becomes more evident. Among the tested settings, 32 and 64 channels provide a favorable operating range, where communication bandwidth is reduced by a large margin while detection performance remains close to the uncompressed baseline.

Training Strategy: Table VI evaluates how different autoencoder training strategies affect the performance–communication trade-off on the OPV2V benchmark. This ablation study examines whether the proposed autoencoder can provide scalable latent representations under different channel budgets, while avoiding repeated retraining for each

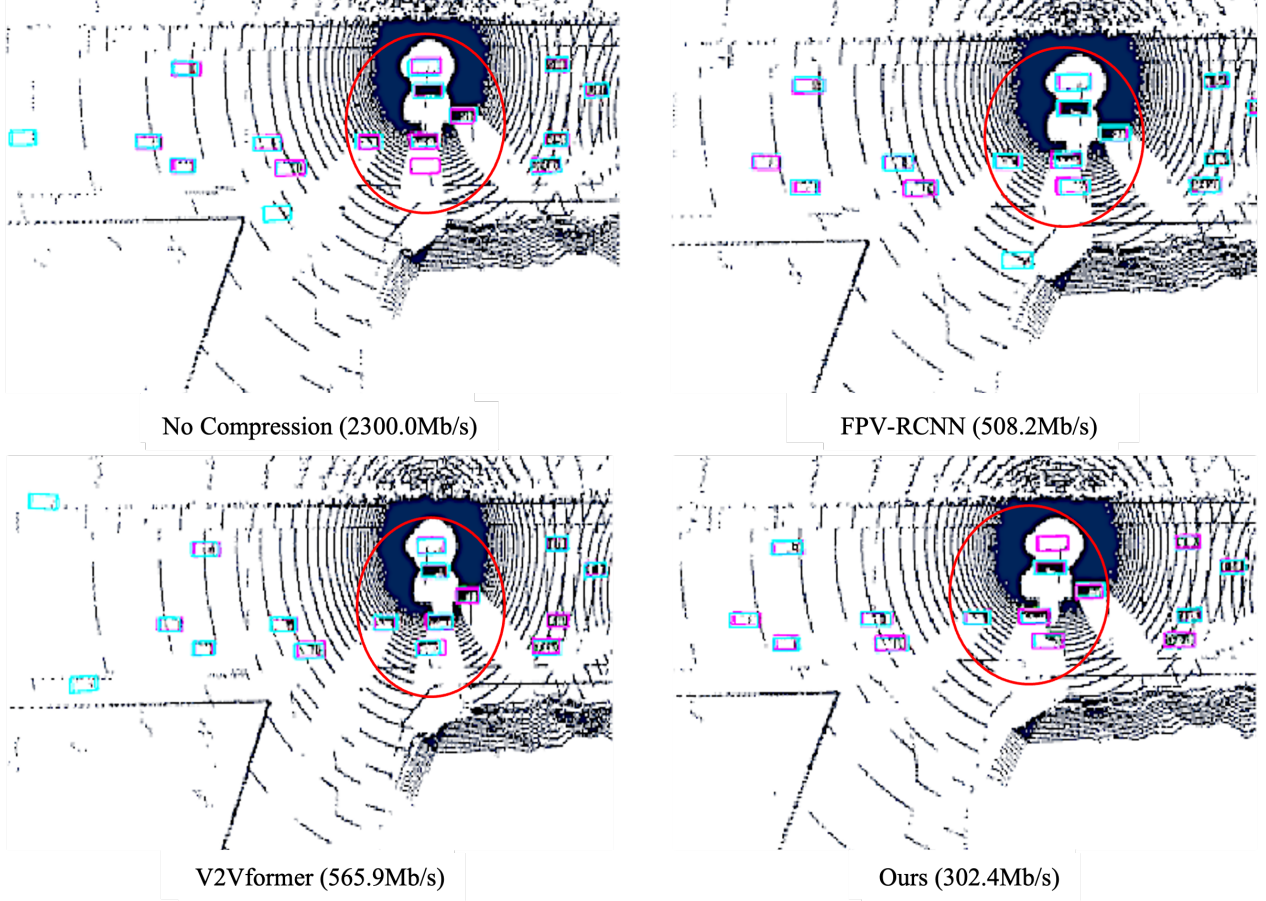


Fig. 8. Qualitative comparison of visualization results from OPV2V. Magenta boxes indicate ground truth annotations, and blue boxes indicate detection results. The visualization has been color-inverted to enhance the clarity of 3D detection effects. These images are best viewed in color.

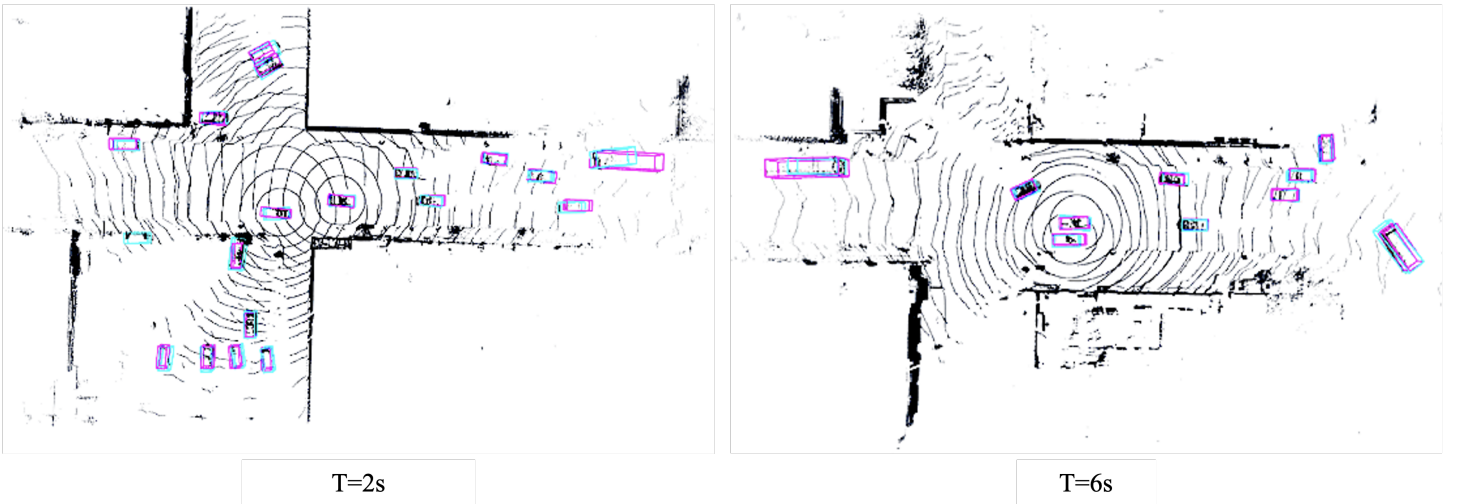


Fig. 9. Real-world visualization results of Co3DCAE on the V2V4Real test set demonstrate the perception performance of two CAVs at an 8 $\times$ bandwidth compression ratio across different timesteps in a busy intersection scenario. Ground truth is shown as magenta rectangles, and predictions are shown as blue rectangles. These images are best viewed in color.

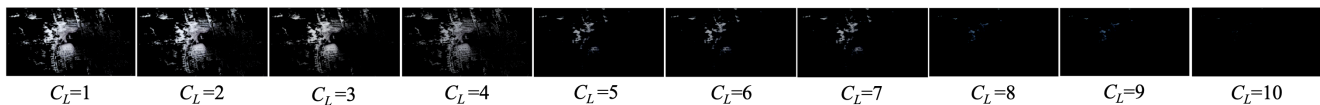


Fig. 10. Visualization of the energy distribution across latent representation channels.

TABLE V
PERFORMANCE COMPARISON BETWEEN Co3DCAE AND THE BASELINE
ON V2V4REAL AT VARIOUS COMPRESSION LEVELS.

Method/Index	Channels (C_L)	AP @0.5/@0.7	Comm. (Mb/s)
TS-IFF (w/o Co3DCAE)	128	68.2 / 40.1	605.6
TS-IFF (w/ Co3DCAE)	64	67.8 / 32.8	75.7
	32	66.3 / 38.0	9.5
	16	55.7 / 31.8	1.2
	8	43.5 / 22.9	0.2

compression ratio.

TABLE VI
ABLATION STUDY ON AUTOENCODER TRAINING STRATEGIES

Dataset	OPV2V		
Method/Index	Channels (C_L)	mAP	Comm.(Mb/s)
Baseline 1	4	55.4	1.2
Baseline 2	4	70.8	1.2
Co3DCAE single-stage	4	69.2	1.2
	8	82.1	100.8
	16	83.4	201.6
	24	86.1	302.8
Co3DCAE multi-stage	4	70.4	1.2
	8	83.8	100.8
	16	84.1	201.6
	24	86.1	302.8

We first compare two baseline configurations using a four-channel latent representation. Baseline 1 performs stepwise compression by padding unused channels with zeros, which leads to a relatively low mAP of 55.4 under a communication bandwidth of 1.2 Mb/s. Baseline 2 removes the zero-padding operation and directly encodes features into the available channels. This modification substantially improves the mAP to 70.8 without increasing the communication cost, indicating that direct channel-wise feature encoding is more effective than padding-based compression. Based on this observation, we further compare two training strategies for the proposed Co3DCAE:

- Single-stage training*: A separate autoencoder is trained for each target number of latent channels. When the target compression ratio changes, the model must be retrained for the corresponding channel dimension.
- Multi-stage training*: A single autoencoder is trained in stages so that different number of latent channels can

be activated at inference time according to the required communication budget.

The results show that increasing the number of latent channels (C_L) improves detection accuracy for both strategies. For single-stage training, the mAP increases from 69.2 at $C_L = 4$ to 86.1 at $C_L = 24$, indicating that a larger latent representation preserves more collaborative perception information. However, this improvement comes with a higher communication cost, which raises from 1.2 Mb/s to 302.8 Mb/s. In comparison, multi-stage training achieves comparable or slightly better accuracy across most C_L settings while requiring only one trained model. Specifically, it improves the mAP from 70.4 at $C_L = 4$ to 86.1 at $C_L = 24$, matching the best performance of single-stage training at the largest channel budget. At intermediate C_L settings, multi-stage training also achieves higher mAP than single-stage training: 83.8 versus 82.1 at $C_L = 8$ and 84.1 versus 83.4 at $C_L = 16$. These results demonstrate that the proposed multi-stage training strategy can flexibly adapt to different communication budgets while maintaining the accuracy of separately trained single-stage models, providing a more practical and efficient solution for scalable collaborative perception.

Multi-Stage Training: We visualize the energy distribution of the latent representation features φ by analyzing the energy associated with each channel index. Specifically, the energy for each channel is computed as follows:

$$E(\varphi) = \frac{1}{C_L \times H_L \times W_L} \sum_{x=1}^{C_L} \sum_{y=1}^{H_L} \sum_{z=1}^{W_L} |\varphi(x, y, z)|^2 \quad (10)$$

As shown in Fig. 10, the energy of the latent representation is not uniformly distributed across different channels. Instead, channels generated at later stages generally exhibit lower energy, indicating that they contain relatively less dominant information for feature reconstruction. This observation suggests that the proposed multi-stage training strategy naturally learns an ordered latent representation, where earlier channels preserve more essential, high-energy information, while later channels mainly provide supplementary details.

Based on this property, the encoder can flexibly select different numbers of latent channels according to the available communication bandwidth during inference. When the communication condition is limited, the low-energy channels located at later stages can be pruned with only a minor impact on reconstruction quality. Conversely, when sufficient bandwidth is available, more channels can be transmitted to preserve richer feature details and further improve perception performance. Therefore, the multi-stage training strategy enables an adaptive compression mechanism. By dynamically adjusting the number of transmitted latent channels, the pro-

posed autoencoder can achieve secondary compression beyond the initial feature encoding process.

From the results shown in Fig. 11, varying the number of latent feature channels during training reveals that the autoencoder trained once with 32 channels achieves a lower MSE than autoencoders trained multiple times. We speculate that this advantage stems from the continuity and consistency of the training process, which helps the network better capture correlations across channels and maintain more stable gradient updates. Moreover, this multi-stage training approach allows flexible selection of any number of channels for encoding according to specific compression requirements, enabling adaptive compression that balances reconstruction quality and resource constraints. This flexibility also facilitates efficient pruning of low-energy channels, further improving compression efficiency without significant performance degradation.

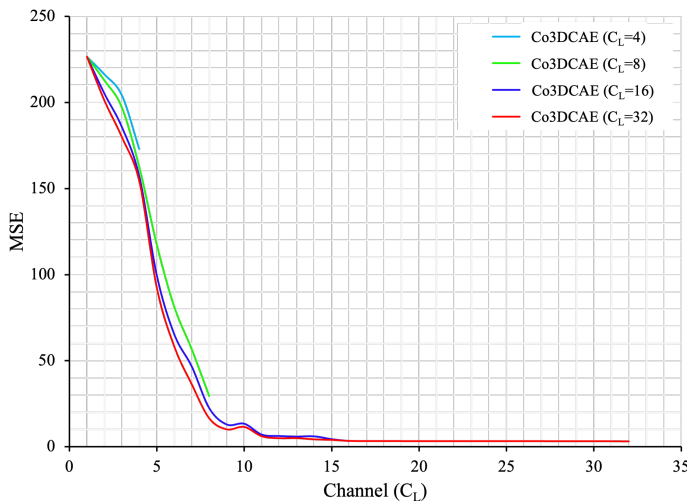


Fig. 11. Ablation study of channel-wise compression under different training strategies. We select encoders with four representative numbers of latent channels to encode intermediate feature maps.

Number of CAVs: We conduct an ablation study to investigate the effect of the number of participating CAVs on the proposed Co3DCAE model. This experiment is performed on the jointly considered DT and CC evaluation subsets of the OPV2V dataset under the feature compression setting with $C_L = 24$. The two subsets are jointly reported in this ablation study to provide an overall trend across representative OPV2V scenarios, rather than to compare the subset-specific performance of DT and CC. The results are shown in Fig. 12. As the number of participating CAVs increases from 2 to 7, collaborative perception accuracy consistently improves across different threshold settings. However, involving more CAVs also increases the communication bandwidth requirement. When seven CAVs participate in collaborative perception, the model achieves the best perception performance while maintaining lightweight bandwidth usage. This experiment further examines the sensitivity of the proposed model to the number of participating CAVs under representative OPV2V scenarios. The overall benchmark-level results on OPV2V and CODD are reported in the main experimental tables.

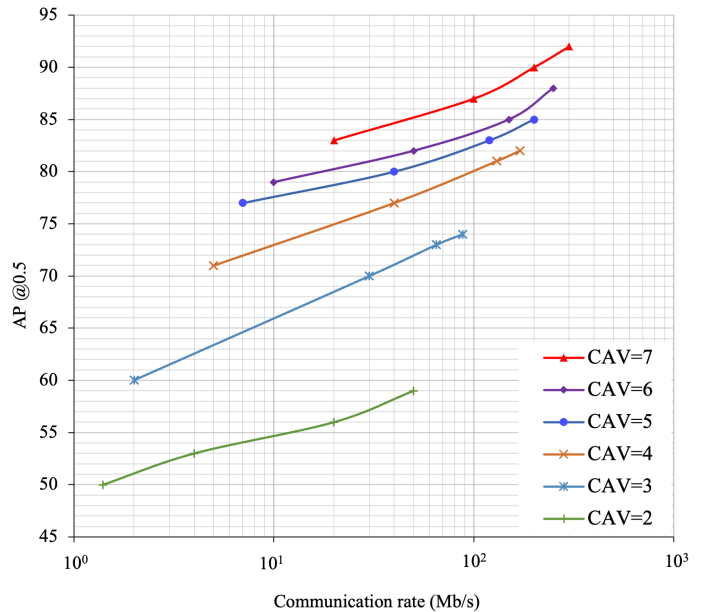


Fig. 12. Ablation study on the effect of the number of participating CAVs on the proposed Co3DCAE model. The results are evaluated on the combined DT and CC evaluation subsets of the OPV2V dataset under the same feature-compression setting with $C_L = 24$. The communication rate denotes the total feature-level communication rate required to transmit the compressed latent representations from the participating CAVs to the fusion node.

V. CONCLUSION AND OUTLOOK

In this paper, we present a novel 3D convolutional autoencoder-based collaborative perception framework that addresses the trade-off between perception accuracy and communication bandwidth in V2X autonomous driving scenarios. The framework introduces three key innovations: (i) a 3D residual convolutional autoencoder for efficient bandwidth compression, (ii) a channel-wise encoder training strategy to improve feature compactness, and (iii) a collaborative mechanism that fuses intermediate feature maps to enable effective information integration across multiple agents.

Extensive evaluation on real-world and simulated LiDAR-based collaborative perception benchmarks shows that the proposed framework achieves superior object detection accuracy while substantially reducing communication bandwidth, highlighting its potential for large-scale deployment in connected autonomous vehicle systems. These results confirm that intermediate-level collaborative fusion, combined with efficient feature compression and scalable training strategies, provides a robust and bandwidth-efficient solution for future V2X perception systems. In future work, we plan to explore real-world deployment through physical experiments and system-level integration. Further research will also focus on enhancing the framework's robustness under dynamic traffic conditions and real-time communication constraints.

REFERENCES

- [1] J. Fang, D. Zhou, F. Yan, T. Zhao, F. Zhang, Y. Ma, L. Wang, and R. Yang, "Augmented lidar simulator for autonomous driving," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1931–1938, 2020.

- [2] A. Caillot, S. Ouerghi, P. Vasseur, R. Boutteau, and Y. Dupuis, "Survey on cooperative perception in an automotive context," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 9, pp. 14 204–14 223, 2022.
- [3] Y. Xing, C. Lv, D. Cao, and P. Hang, "Toward human-vehicle collaboration: Review and perspectives on human-centered collaborative automated driving," *Transportation research part C: emerging technologies*, vol. 128, p. 103199, 2021.
- [4] L. Song, W. Valentine, Q. Yang, H. Wang, H. Fang, and Y. Liu, "Bb-align: A lightweight pose recovery framework for vehicle-to-vehicle cooperative perception," in *2024 IEEE 44th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2024, pp. 1016–1026.
- [5] B. Wang, L. Zhang, Z. Wang, Y. Zhao, and T. Zhou, "Core: Cooperative reconstruction for multi-agent perception," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 8710–8720.
- [6] C. Lin, D. Tian, X. Duan, J. Zhou, D. Zhao, and D. Cao, "V2vformer: Vehicle-to-vehicle cooperative perception with spatial-channel transformer," *IEEE Transactions on Intelligent Vehicles*, 2024.
- [7] Y.-C. Liu, J. Tian, N. Glaser, and Z. Kira, "When2com: Multi-agent perception via communication graph grouping," in *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, 2020, pp. 4106–4115.
- [8] Y. Hu, S. Fang, Z. Lei, Y. Zhong, and S. Chen, "Where2comm: Communication-efficient collaborative perception via spatial confidence maps," *Advances in neural information processing systems*, vol. 35, pp. 4874–4886, 2022.
- [9] Y.-C. Liu, J. Tian, C.-Y. Ma, N. Glaser, C.-W. Kuo, and Z. Kira, "Who2com: Collaborative perception via learnable handshake communication," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 6876–6883.
- [10] Z. Yang, Y. Sun, S. Liu, X. Shen, and J. Jia, "Std: Sparse-to-dense 3d object detector for point cloud," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 1951–1960.
- [11] Q. Jiang, C. Hu, B. Zhao, Y. Huang, and X. Zhang, "Scalable 3d object detection pipeline with center-based sequential feature aggregation for intelligent vehicles," *IEEE Transactions on Intelligent Vehicles*, vol. 9, no. 1, pp. 1512–1523, 2023.
- [12] C. M. R. Carletti, C. Casetti, J. Härrä, and F. Risso, "Platoon-local dynamic map: Hierarchical cooperative perception for platooning vehicles," *IEEE Transactions on Vehicular Technology*, 2025.
- [13] J. Zhang, K. Yang, H. Wang, P. Sun, and L. Song, "Efficient vehicular collaborative perception based on spatial-temporal feature compression," *IEEE Transactions on Vehicular Technology*, 2024.
- [14] Z. Bai, G. Wu, M. J. Barth, H. Qiu, Y. Liu, E. A. Sisbot, and K. Oguchi, "Pillar attention encoder for adaptive cooperative perception," *IEEE Internet of Things Journal*, 2024.
- [15] S. Shi, C. Guo, L. Jiang, Z. Wang, J. Shi, X. Wang, and H. Li, "Pv-rcnn: Point-voxel feature set abstraction for 3d object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 10 529–10 538.
- [16] Y. Li, S. Ren, P. Wu, S. Chen, C. Feng, and W. Zhang, "Learning distilled collaboration graph for multi-agent perception," *Advances in Neural Information Processing Systems*, vol. 34, pp. 29 541–29 552, 2021.
- [17] M. Dong, Y. Fu, C. Li, M. Tian, F. R. Yu, and N. Cheng, "Task offloading and resource allocation in vehicular cooperative perception with integrated sensing, communication, and computation," *IEEE Transactions on Intelligent Transportation Systems*, 2025.
- [18] H. Chen, C. Zhai, Y. Li, and B. Vucetic, "Cooperative strategies for wireless-powered communications: An overview," *IEEE Wireless Communications*, vol. 25, no. 4, pp. 112–119, 2018.
- [19] Z. Meng, X. Xia, R. Xu, W. Liu, and J. Ma, "Hydro-3d: Hybrid object detection and tracking for cooperative perception using 3d lidar," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 8, pp. 4069–4080, 2023.
- [20] X. Xia, E. Hashemi, L. Xiong, and A. Khajepour, "Autonomous vehicle kinematics and dynamics synthesis for sideslip angle estimation based on consensus kalman filter," *IEEE Transactions on Control Systems Technology*, vol. 31, no. 1, pp. 179–192, 2022.
- [21] R. Xu, H. Xiang, X. Xia, X. Han, J. Li, and J. Ma, "Opv2v: An open benchmark dataset and fusion pipeline for perception with vehicle-to-vehicle communication," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 2583–2589.
- [22] D. Qiao and F. Zulkernine, "Adaptive feature fusion for cooperative perception using lidar point clouds," in *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 2023, pp. 1186–1195.
- [23] M. Wang, D. Li, J. R. Casas, and J. Ruiz-Hidalgo, "Adaptive fusion of lidar features for 3d object detection in autonomous driving," *Sensors*, vol. 25, no. 13, 2025.
- [24] E. Arnold, S. Mozaffari, and M. Dianati, "Fast and robust registration of partially overlapping point clouds," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 1502–1509, 2021.
- [25] R. Xu, X. Xia, J. Li, H. Li, S. Zhang, Z. Tu, Z. Meng, H. Xiang, X. Dong, R. Song *et al.*, "V2v4real: A real-world large-scale dataset for vehicle-to-vehicle cooperative perception," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 13 712–13 722.
- [26] K. Cai, T. Qu, B. Gao, and H. Chen, "Consensus-based distributed cooperative perception for connected and automated vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 8, pp. 8188–8208, 2023.
- [27] S. Su, Y. Li, S. He, S. Han, C. Feng, C. Ding, and F. Miao, "Uncertainty quantification of collaborative detection for self-driving," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 5588–5594.
- [28] F. Qingyun and W. Zhaokui, "Cross-modality attentive feature fusion for object detection in multispectral remote sensing imagery," *Pattern Recognition*, vol. 130, p. 108786, 2022.
- [29] Y. Yuan, H. Cheng, and M. Sester, "Keypoints-based deep feature fusion for cooperative vehicle detection of autonomous driving," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 3054–3061, 2022.
- [30] H. Yu, Y. Luo, M. Shu, Y. Huo, Z. Yang, Y. Shi, Z. Guo, H. Li, X. Hu, J. Yuan *et al.*, "Dair-v2x: A large-scale dataset for vehicle-infrastructure cooperative 3d object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 21 361–21 370.
- [31] Z. Lei, S. Ren, Y. Hu, W. Zhang, and S. Chen, "Latency-aware collaborative perception," in *European Conference on Computer Vision*. Springer, 2022, pp. 316–332.
- [32] Q. Yang, S. Fu, H. Wang, and H. Fang, "Machine-learning-enabled cooperative perception for connected autonomous vehicles: Challenges and opportunities," *IEEE Network*, vol. 35, no. 3, pp. 96–101, 2021.
- [33] Q. Chen, X. Ma, S. Tang, J. Guo, Q. Yang, and S. Fu, "F-cooper: Feature based cooperative perception for autonomous vehicle edge computing system using 3d point clouds," in *Proceedings of the 4th ACM/IEEE Symposium on Edge Computing*, 2019, pp. 88–100.
- [34] J. Guo, D. Carrillo, S. Tang, Q. Chen, Q. Yang, S. Fu, X. Wang, N. Wang, and P. Palacharla, "Coff: Cooperative spatial feature fusion for 3-d object detection on autonomous vehicles," *IEEE Internet of Things Journal*, vol. 8, no. 14, pp. 11 078–11 087, 2021.
- [35] R. Xu, H. Xiang, Z. Tu, X. Xia, M.-H. Yang, and J. Ma, "V2x-vit: Vehicle-to-everything cooperative perception with vision transformer," in *European conference on computer vision*. Springer, 2022, pp. 107–124.
- [36] R. Xu, Z. Tu, H. Xiang, W. Shao, B. Zhou, and J. Ma, "Cobevt: Cooperative bird's eye view semantic segmentation with sparse transformers," in *Conference on Robot Learning*. PMLR, 2023, pp. 989–1000.
- [37] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 10 012–10 022.
- [38] W. Yuan and W. Xu, "Msst-net: A multi-scale adaptive network for building extraction from remote sensing images based on swin transformer," *Remote Sensing*, vol. 13, no. 23, p. 4743, 2021.
- [39] A. Masood, U. Naseem, and I. Razzak, "Multi-scale swin transformer enabled automatic detection and segmentation of lung metastases using ct images," in *2023 IEEE 20th international symposium on biomedical imaging (ISBI)*. IEEE, 2023, pp. 1–5.
- [40] F. Chi, Y. Wang, P. Nasiopoulos, and V. C. Leung, "Parameter-efficient federated cooperative learning for 3d object detection in autonomous driving," *IEEE Internet of Things Journal*, 2025.
- [41] Q. Xie, X. Zhou, T. Hong, T. Qiu, and W. Qu, "Swisscheese: Fine-grained channel-spatial feature filtering for communication-efficient cooperative perception," *IEEE Transactions on Intelligent Transportation Systems*, 2024.
- [42] M. K. Abdel-Aziz, C. Perfecto, S. Samarakoon, M. Bennis, and W. Saad, "Vehicular cooperative perception through action branching and federated reinforcement learning," *IEEE Transactions on Communications*, vol. 70, no. 2, pp. 891–903, 2021.
- [43] B. Lu, X. Huang, Y. Wu, L. Qian, S. Zhou, and D. Niyato, "Joint optimization of compression, transmission and computation for cooperative perception aided intelligent vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 74, no. 5, pp. 8201–8214, 2025.

- [44] C. Zhao, D. Ding, C. Lei, S. Wang, Y. Ji, and Y. Du, "Safety field-based vehicle-infrastructure cooperative perception for autonomous driving using 3d point clouds," *IEEE Transactions on Intelligent Transportation Systems*, 2025.
- [45] X. Wang, J. Li, J. Wu, S. Wu, and L. Li, "Dadan: Dynamic-augmented and density-aware network for accurate 3d object detection with 4d radar," *IEEE Sensors Journal*, 2025.
- [46] C. Liu, Y. Chen, J. Chen, R. Payton, M. Riley, and S.-H. Yang, "Co-operative perception with learning-based v2v communications," *IEEE Wireless Communications Letters*, vol. 12, no. 11, pp. 1831–1835, 2023.
- [47] M. Usman, M. Zaka-Ud-Din, and Q. Ling, "Enhanced encoder-decoder architecture for visual perception multitasking of autonomous driving," *Expert Systems with Applications*, vol. 246, p. 123249, 2024.
- [48] R. Abbasi, A. K. Bashir, H. J. Alyamani, F. Amin, J. Doh, and J. Chen, "Lidar point cloud compression, processing and learning for autonomous driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 1, pp. 962–979, 2022.
- [49] J. G. Andrews, S. Buzzi, W. Choi, S. V. Hanly, A. Lozano, A. C. K. Soong, and J. C. Zhang, "What will 5g be?" *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1065–1082, 2014.
- [50] X. Lin, "An overview of 5g advanced evolution in 3gpp release 18," *IEEE Communications Standards Magazine*, vol. 6, no. 3, pp. 77–83, 2022.
- [51] S. Gyawali, S. Xu, Y. Qian, and R. Q. Hu, "Challenges and solutions for cellular based v2x communications," *IEEE Communications Surveys Tutorials*, vol. 23, no. 1, pp. 222–255, 2021.
- [52] M. H. C. Garcia, A. Molina-Galan, M. Boban, J. Gozalvez, B. Coll-Perales, T. Şahin, and A. Kousaridas, "A tutorial on 5g nr v2x communications," *IEEE Communications Surveys Tutorials*, vol. 23, no. 3, pp. 1972–2026, 2021.
- [53] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "Pointpillars: Fast encoders for object detection from point clouds," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 12 697–12 705.
- [54] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "Carla: An open urban driving simulator," in *Conference on robot learning*. PMLR, 2017, pp. 1–16.
- [55] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.
- [56] T.-H. Wang, S. Manivasagam, M. Liang, B. Yang, W. Zeng, and R. Urtasun, "V2vnet: Vehicle-to-vehicle communication for joint perception and prediction," in *European conference on computer vision*. Springer, 2020, pp. 605–621.



Mingrui Wang received his M.S. degree in Control Theory and Control Engineering from Harbin University of Science and Technology, Harbin, China. He is currently a joint Ph.D. candidate at Harbin University of Science and Technology and the Polytechnic University of Catalonia. His research interests include computer vision, data fusion, and autonomous driving technology.



Judit Salavedra Pujol received the B.Sc. degree in Telecommunications Technologies and Services, specializing in Audiovisual Systems, from the Universitat Politècnica de Catalunya (UPC) and the M.Sc. degree in Computer Vision from the Universitat Autònoma de Barcelona (UAB). She is currently a Ph.D. student at UPC and is involved in the European project 6G-EWOC. Her research interests include computer vision, multimodal 3D perception, and autonomous driving systems using single-vehicle and crowdsourced data.



Dongjie Li was born in Weifang, China. She received the B.S., M.S. and Ph.D. degree in mechanical engineering and automation from Harbin University of Science and Technology, Harbin, China, in 2004, 2007, and 2009, respectively. From 2010 to 2013, she worked as a Post-doctoral Fellow with the State Key Laboratory of Robotics and System, Harbin Institute of Technology. From 2016 to 2017, she was a visiting scholar in School of Aeronautics and Astronautics, Georgia Institute of Technology, USA. She is currently a professor and doctoral supervisor at Harbin University of Science and Technology, Harbin, China. Her research interests include micro/nano robotics, intelligent mechatronics, and machine vision.



Josep R. Casas graduated in Telecommunications Engineering in 1990 and received the PhD in 1996, both from UPC in Barcelona. He was a visiting researcher at CSIRO Mathematics & Information Sciences in Canberra from 2000 to 2001. He has been principal investigator of projects MuViPro and PROVEC (Plan Nacional de I+D+i, 2007-2014) and led SGR projects of UPC [Image Processing Group](#) from 2014 to 2021. He is currently involved in European project [6G-EWOC](#) and contributed to (or led) other European projects (EUOfusion, FascinatE, ACTIBIO, CHIL, SCHEMA, ADVISOR) and industry-sponsored projects (InPercept, VISION, HESPERIA, D'Ocon) since 1995. His current research interests focus on multi-view and multimodal data fusion approaches for 3D scene analysis in fields such as automotive, biomedical and human interaction.



Javier Ruiz-Hidalgo graduated in Telecommunications Engineering at the Universitat Politècnica de Catalunya (UPC), Barcelona, Spain in 1997. From 1998 to 1999, he developed an MSc by Research on the field of Computer Vision by the University of East Anglia (UEA) in Norwich, UK. During 1999 he joined the Image Processing Group at UPC working on image and video indexing in the context of the MPEG-7 standard. In 2006, he received his PhD. in the field of image processing. Since 2001 he is an Associate Professor at the Universitat Politècnica de Catalunya. He is currently lecturing on the area of digital signal and systems, image processing and computer vision. His current research interests include 3D video coding and analysis, graph neural networks, conditional generative networks and super-resolution.