IEEE Open Journal of Vehicular Technology

Received 18 March 2025; accepted 26 March 2025. Date of publication 1 April 2025;
date of current version 9 May 2025. The review of this article was coordinated by Editor Zhuo Sun.

Digital Object Identifier 10.1109/OJVT.2025.3556381

DelAwareCol: Delay Aware Collaborative Perception

AHMED N. AHMED, SIEGFRIED MERCELIS, AND ALI ANWAR (Member, IEEE)

Faculty of Applied Engineering, imec - IDLab, University of Antwerp, 2000 Antwerp, Belgium

CORRESPONDING AUTHOR: AHMED N. AHMED (e-mail: ahmed.ahmed@uantwerpen.be).

This work was supported by the Research Foundation Flanders (FWO) under Grant 1S90022 N.

ABSTRACT Multi-agent collaborative perception has gained significant attention due to its ability to overcome the challenges stemming from the limited line-of-sight visibility of individual agents that raised safety concerns for autonomous navigation. Despite notable progress in collaborative perception, several persistent challenges hinder optimal performance, such as the size of data being shared, communication delays, computationally expensive collaboration mechanisms, and spatial misalignment. To address these challenges, we propose DelAwareCol, a versatile collaborative perception framework that tackles the transmission delay between connected agents in real-life autonomous driving. Our framework introduces three key modules designed to balance perception performance with communication bandwidth and delay. Firstly, an intra-agent information aggregation module captures valuable semantic cues within the temporal context to enhance the local representation of each ego agent. Secondly, an inter-agent information aggregation module manages inter-agent interactions and spatial relationships, addressing common vehicle-to-vehicle (V2V) and vehicle-to-everything (V2X) issues, such as spatial misalignment, asynchronous information sharing, and pose errors. Thirdly, an adaptive fusion mechanism integrates multi-source representations based on dynamic contributions from different agents. The proposed framework is validated on large-scale simulated and real-life collaborative perception datasets OPV2V, V2XSet, and V2VReal. Our experimental results demonstrate that DelAwareCol achieved state-of-the-art performance in collaborative object detection, maintaining robust performance in the presence of high latency and localization error.

INDEX TERMS Collaborative perception, spatio-temporal modeling, attention, V2X communication, autonomous vehicles, 3D object detection.

I. INTRODUCTION

Autonomous Vehicles (AVs) have the potential to enhance the safety, efficiency, and accessibility of future transportation systems. AVs must operate with robust situational awareness to provide safe and efficient navigation. Despite rapid advancements in sensor technology and perception algorithms, existing solutions still present significant challenges that hinder the widespread deployment of AVs on public roads. A fundamental limitation arises from the inherent constraints of perception capabilities within a single vehicle; such as objects further away from the vehicle are difficult to detect due to sparse data, and limited perceptual range of the sensors. In addition to that, in crowded or complex environments, objects may be obscured by other obstacles, further impairing their detection. Furthermore, perception sensors can experience temporary failures due to internal or external factors, including circuit malfunctions or smudges. Moreover, the perception precision can be affected by lighting conditions and sensor noise. These flaws can result in missed or inaccurate situational awareness, potentially leading to cascading failures in downstream modules (such as planning and control) that are dependent on the results of perception.

Collaborative perception [1] offers a promising approach to overcoming these limitations. Through V2V and V2X communication, connected autonomous vehicles (CAVs) and roadside units (RSUs) can share perception data (for simplicity, we will refer to CAVs and roadside units as agents), thereby achieving more accurate and robust perception that extends beyond the range and line of sight of individual

vehicles, achieving a shared situational awareness through collaboration. Multiple studies have demonstrated that [1], [2] collaborative perception, facilitated by V2X communication, reduces blind spots, increases object detection accuracy, and enhances situational awareness of the agent; thereby preventing collisions and improving overall transportation safety.

Collaborative perception in AV is categorized into three categories based on the type of data being shared among CAVs. Early collaboration [3], [4] involves each agent sharing its raw sensor data. Although this approach provides the most detailed and comprehensive data, however, this approach imposes a significant communication overhead due to the size of the raw data being shared. Late fusion [5], [6] involves sharing fully processed and high-level information between agents. Despite the low communication bandwidth required by this approach, it suffers a potential loss of context, which results in suboptimal detection performance due to insufficiently shared context as the receiving agent relies entirely on the sender's interpretation. In contrast, intermediate fusion [7], [8], [8], [9], [10], [11], [12], [13], which involves sharing partially processed raw sensor data i.e. semantic information. This approach has been demonstrated to balance perception performance-communication overhead trade-off effectively.

In practical scenarios, semantic information exchanged among neighboring agents is subject to multi-level heterogeneity, including varying transmission latencies and differing perspectives of the surrounding environment. This heterogeneity results in asynchronous and misaligned semantic information in both temporal and spatial dimensions. As noted in [14], real-time LTE-V2X communication systems exhibit an average latency of 498 communication periods plus 131.30 ms. Such delays cause collaborative data to reflect past states of the environment, potentially leading to substantial inaccuracies in object detection outputs. Consequently, identifying relevant collaborators that can enhance perception performance becomes challenging due to these spatiotemporal misalignments.

Several studies have addressed various aspects of collaborative perception, with some focusing exclusively on spatial misalignment [7], [8], [9], [10], [11], [12], [13], while others have tackled temporal misalignment [15], [16], [16], [17]. However, these approaches remain suboptimal as they fail to account for temporal misalignment caused by significant transmission delays and communication losses between agents, which are critical for real-world applications. For instance, relying on semantic information received after substantial delays may mislead the ego agent, resulting in erroneous perception outcomes.

In this work, we address the following key research questions:

- How should a collaborative perception model adapt to achieve optimal performance in the event of communication loss?
- If a collaborator is positioned in a location that could enhance the ego agent's detection accuracy, but the data is received with a delay, how should the model incorporate this delayed information effectively?
- During the fusion of collaborator data, how can the model mitigate the impact of noise to ensure robust perception outcomes?

In this work, we propose a framework for intermediate collaborative perception that handles data transmission delays between agents. Our proposed framework is built based on three main modules, which we are going to elaborate on in this paragraph. Multi-agent collaborative approaches [9], [13], [17] utilize only the current frame of the ego agent, this overlooks valuable contextual information that could be available from previous frames. This single-frame approach leads to challenges in precisely detecting objects, especially in scenarios where the connection with neighboring agents is delayed or interrupted, the agent must depend solely on its own perception, potentially diminishing detection precision. To overcome this limitation, we introduce the first module of our framework, the *intra-agent fusion module*, which leverages both historical and real-time data from the ego agent. This module employs spatiotemporal ego feature aggregation, capturing critical temporal cues from prior frames to enhance the integration of contextual semantics available from previous frames of the ego agent. In addition to that, to ensure aggregation, the ego agent aggregates only complementary information from relevant neighbors and guarantees seamless aggregation; we thereby introduce our second module *inter-agent fusion module*. This module accounts for the time delay by encoding it within the feature map of each agent, mitigating temporal misalignment occurring due to such delays. The inter-agent fusion module is designed to aggregate perceptually delayed information received from neighboring agents with that of the ego agent while addressing transmission delays through the inclusion of time encoding during the neighboring agent's feature aggregation process with that of the ego agent. Subsequently, we introduce the third component *adaptive fusion module* to incorporate the contextual features retrieved from the intra and inter-fusion created from diverse representations based on distinctive characteristics of each module i.e. historical context, collaboratively reinforced features respectively. The adaptive fusion module is also enhanced by incorporating a third input i,e. the ego agent's current frame to ego-centered characteristics, and based on the contributions of each module, this module learns to dynamically assign weights to each feature. The main contributions of our proposed framework can be summarized as follows:

- We propose a novel framework for multi-agent collaborative perception, enabling efficient information sharing and feature fusion across agents. This approach achieves an optimal trade-off between performance and bandwidth. Extensive experimental results on collaborative detection tasks demonstrate that our method surpasses previous state-of-the-art approaches.
- We introduce the integration of ego agent temporal context into collaborative perception systems. We propose an intra-agent fusion mechanism that incorporates both

the current frame with the historical frames, capturing valuable temporal information retrieved solely by the ego agent.
- We present an novel spatiotemporal inter-agent fusion module that accounts for the time delay of each neighboring agent within the feature map. This module enables refined semantic information aggregation between agents based on their temporality (time delay) as well as their contextual information, which refines multi-source features for the ego agent, enhancing overall system object detection performance.
- We carry out a comprehensive evaluation of our proposed method on three large-scale widely used collaborative perception datasets OPV2V [12], V2XSet [17], and V2VReal [18]. The experimental results show that DelAwareCol is more effective in improving perception performance than state-of-the-art works, such as V2VNet [9], V2X-ViT [17], Where2comm [13] and Select2Col [15].

The remainder of this paper is organized as follows. Section II reviews the related works on collaborative perception. Section III presents the system model and formulates the problem. Section III elaborates on our proposed framework. We conduct experiments to verify our proposed method in Section IV. We conduct ablation study in Section V to investigate the effect of different modules on the model's performance. Finally, we conclude this article with a summary in Section VI.

II. RELATED WORKS

Intermediate-level multi-agent collaborative perception has recently received significant attention [1]. However, despite the promising advances from previous efforts, several unavoidable challenges that cause performance bottlenecks remain. In this section, we review different strategies of intermediate collaborative perception highlighting their potential and limitations.

A. SPATIAL COLLABORATIVE PERCEPTION.

Numerous well-designed works investigate efficient collaboration mechanisms. In CoBEVT [8] a local-global sparse attention mechanism is implemented that captures intricate spatial interactions across different agents' fields of view views. V2VNet [9] adopts a space-aware graph neural network (GNN) to aggregate information from nearby agents. DiscoNet [10] utilizes a student-teacher model where the student model is trained to mimic the pre-trained teacher model. DiscoNet outperforms V2VNet in terms of detection precision. In our previous works [11], [19] we proposed a graph attention network-based aggregation strategy for intermediate representation fusion in collaborative perception, where both channel and spatial attention are utilized to attend over the ego and neighboring agent's features. Where2Comm [13] introduces an agent-selective strategy that optimizes bandwidth and channel utilization by judiciously selecting which agents to communicate with and what information to exchange. However, eventhough the aforementioned collaborative perception techniques enhanced the object detection performance and focused on reducing communication bandwidth, they rely solely on spatial misalignment neglecting the temporal aspect i.e. information is shared instantaneously across agents. Thereby those methods don't benefit from the historical context of the ego agent.

B. SPATIO-TEMPORAL COLLABORATIVE PERCEPTION.

Spatio-temporal methods are designed to address these extra challenges ensuring effective correction of spatial and temporal discrepancies. For instance, SCOPE [20] adapted Long Short-Term Memory (LSTM) module to extract temporal dependencies from the historical frame of the ego agent to capture temporal semantics. Simultaneously, it ensures that the ego agent integrates the cross-agent information by utilizing a cross-attention module. Similarly, What2comm [21] introduced a spatial integration and a temporal context aggregation module. The spatial integration module extracts spatial information from exclusive representations of collaborators leveraging a cross-attention module. Concurrently, the temporal context module integrates the ego agent's historical features via a gating mechanism, substantially mitigating collaboration noise. Select2col [15] proposes the Historical Prior Hybrid Attention (HPHA) to integrate the features across the spatial and temporal dimensions. It utilizes a spatial-attention weight to indicate the importance of a collaborator's semantic information to generate the aggregated semantic information. Then employs a short-term attention module to capture correlations in semantic information across the temporal dimension using historical features of the ego agent. While spatio-temporal collaborative perception methods have made substantial progress, these approaches remain suboptimal, as they didn't consider temporal misalignment originating due to latency, and lossy communication between agents which is crucial for real-life applications. This leads to significant performance degradation when applied in real-world scenarios with communication imperfections, such as network congestion and signal loss [16].

C. COLLABORATIVE PERCEPTION UNDER COMMUNICATION DELAYS

Communication plays a vital role in achieving pragmatic multi-agent collaborative perception as it directly determines the subsequent collaboration performance. The delay occurring due to broadcasting/transmission speeds and inference models to generate the semantic information, communication delays, and interruptions are critical challenges that influence collaborative perception performance. Since delays are inevitable in the collaborative perception framework, some methods have been developed to address these issues by implementing a time delay compensation module. V2X-ViT [17], for example, introduces a Vision Transformer (ViT) architecture that employs multi-agent self-attention to effectively capture spatial relationships between agents, enabling efficient feature fusion. Besides that it adopts an adaptive

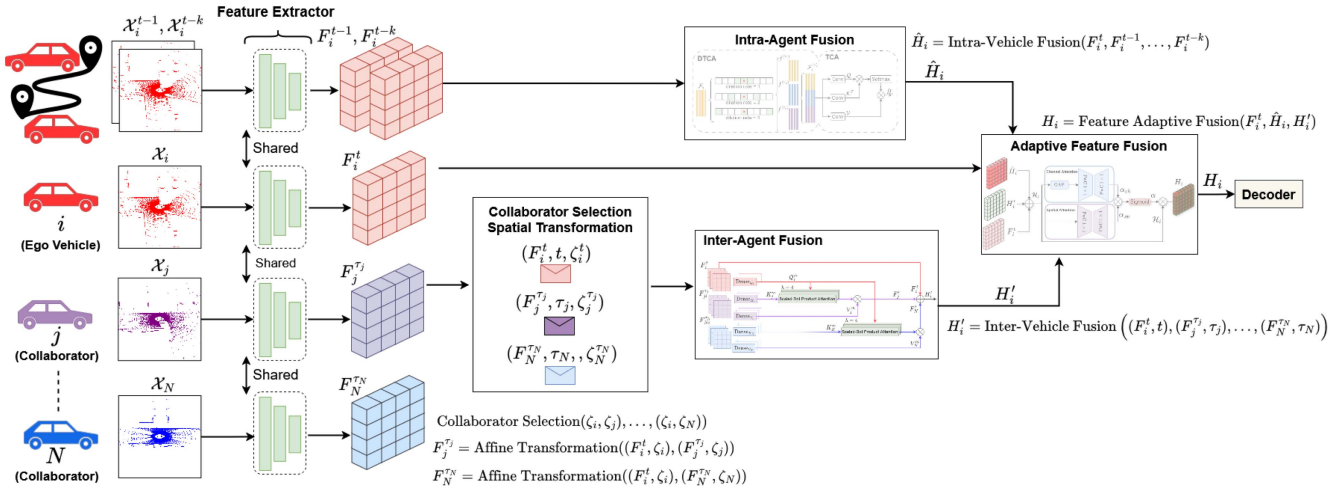

FIGURE 1. The overall architecture of the proposed method. The framework consists of four parts: (i) inter-agent fusion module, (ii) collaborator selection and spatial transformation, (iii) inter-agent fusion module, and (iv) adaptive fusion module. The details of each module are illustrated in Section III.

delay-aware positional encoding for time alignment, compensating for latency by inputting delay time information between collaborators to ensure accurate synchronization despite delays. Furthermore, SyncNet [16] introduces a novel latency-aware collaborative perception system that functions as a plugin for latency compensation by synchronizing perceptual features across agents to a common timestamp while simultaneously estimating real-time features and collaboration attention. This method diverges from traditional techniques by employing historical information for synchronization via a dual-branch pyramid LSTM network, which captures spatial features at multiple scales. Other methods employ flow maps to address latency issues like FFNet [22]. It proposes a self-supervised approach to train a feature flow generator that predicts future features and compares them with the ego vehicle's sensor data. It employs a linear operation for prediction, effectively addressing temporal fusion errors across different latencies and compensating for uncertainties in latency.

In contrast, in our proposed cross-agent collaborative perception we harness spatiotemporal collaborative perception and include dynamically varying time delays to mitigate features spatial and temporal misalignment and ensure that our proposed method demonstrates resilient performance on detection precision in the presence of time delays. In addition to that, we adopt spatio-temporal aggregation within the ego agent's current and historical frames to provide a richer semantic context, and help refine object characteristics that mitigate the impact of inaccurate/noisy inter-agent synchronization, or in case of communication loss.

III. METHODOLOGY

A. OVERVIEW

We consider that there exists an ego agent i performing collaborative perception with a set of neighboring agents N, where $N \in \{j, \ldots, N\}$ perceiving the environment in a scene (e.g., vehicles and/or road infrastructures with perception and communication functionalities). Let $\mathcal{X}_i^t$ be the raw point cloud observation of the i-th agent at timestamp t. This $\mathcal{X}_i$ is then projected onto a bird eye view image (BEV). The BEV of each agent is then passed to a shared encoding network locally f_{enc} that performs perceptual feature extraction from *BEV* resulting in a perceptual feature map as shown in (1a), where $F_i^t \in \mathbb{R}^{C \times H \times W}$. For a more consistent representation of semantic information, we employ the notation $F_j^{\tau_j}$ to denote the semantic information neighboring agent j, where τ_j denotes the time delay between the j-th agent capturing the scene until the i-th agent receives the feature. Note that this work considers that collaboration happens at discrete time stamps and τ is discrete as each agent has a certain sampling rate of observation, and other surrounding circumstances that impact the time delay (will be further discussed in Section. III-E). An illustrated overview of the proposed methodology is shown in Fig. 1, and the overall methodology can be formulated as:

$$F_i^t = f_{\text{enc}}(\text{BEV}_i) \tag{1a}$$

$$\hat{H}_i = \text{Intra-Agent Fusion}(F_i^t, F_i^{t-1}, \ldots, F_i^{t-k}) \tag{1b}$$

$$\text{Collaborator Selection}(\zeta_i, \zeta_j), \ldots, (\zeta_i, \zeta_N)) \tag{1c}$$

$$\text{Affine Transformation}((F_i^t, \zeta_i), (F_j^{\tau_j}, \zeta_j)) \tag{1d}$$

$$H_i' = \text{Inter-Agent Fusion}\left((F_i^t, t), (F_j^{\tau_j}, \tau_j), \ldots, (F_N^{\tau_N}, \tau_N)\right) \tag{1e}$$

$$H_i = \text{Adaptive Fusion}(F_i^t, \hat{H}_i, H_i') \tag{1f}$$

$$Y^{\text{cls}}, Y^{\text{reg}} = f_{\text{dec}}(H_i) \tag{1g}$$

where $\hat{H}_i$ denotes the intra-agent fusion step, which utilizes spatiotemporal attention to aggregate both current and historical features of the ego agent. We assume that each agent can store k frames of historical features in memory. Since not all neighboring agents are relevant to the ego agent we employ the Collaborator Selection function to select only

TABLE 1. Notations and Explanation

Notation	Explanation
General Terms	
$\mathcal{X}_i^t$	Point cloud data of agent i
f_{enc}	Encoder that extracts semantic information from the BEV
F_i^t	Feature map of agent i at time t
ζ_i	Pose of ego agent i
f_{dec}	Convolution based detection decoder
Intra-Vehicle Feature Fusion	
$F_i^{(t-k)}$	Feature map of agent i with previous frames k
$\text{DilatedConv}_{1\times1}^{r_n}$	Dilated convolution operation
$\mathcal{F}_i$	Concatenated current and historical frames $F_i^t, F_i^{t-1}, \ldots, F_i^{t-k}$
f^{r_n}	Resulting feature map after applying dilated convolution
$\mathcal{F}_i^{(r)}$	Aggregated Feature representation resulting from the dilated convolution
$\mathcal{Q}$	Query within the TSA
$\mathcal{K}$	Keys within the TSA
$\mathcal{V}$	Values within the TSA
$\hat{H}_i$	Intra-vehicle fusion feature representation
Inter-Vehicle Feature fusion	
$\zeta_{j\to i}$	Transformation matrix to transform the feature map of agent j to agent i coordinate
τ_j^{asyn}	Inter-agent asynchronous overhead
τ_j^{ext}	Time required for feature extraction
τ_j^{trans}	Collaborative message transmission latency
τ_j^{idle}	Synchronization between the perception system and communication system
b_{ji}	Agent j's transmission bandwidth to agent i
p_{ji}^{noise}	Agent j's transmission noise power to agent i
p_{ji}^{loss}	Agent j's transmission path loss to agent i
τ_j	Total time delay of agent j
$F_j^{\tau_j}$	Feature map of j shared to agent i with time delay τ_j
TE_p	Temporal token encoding
$F_{ji}^{\tau_j}$	Agent j's feature after including the temporal token encoding
H_i'	Inter-vehicle fusion feature representation
Adaptive Feature Fusion	
$\mathcal{H}_i$	Aggregated features of the F_i, $\hat{H}_i$, and H_i'
α_{ch}	Channel attention cofficient
α_{sp}	Spatial attention cofficient
H_i	Adaptive fusion feature representation

relevant neighbors. Consequently, we utilize the Affine Transformation to transform the selected collaborator's semantic information to the ego agent's perceptive. The term H_i' to represent the inter-agent fusion output. Given that the ego agent receives perceptual features from other agents with varying latencies τ, both current and collaborative information from relevant agents are fused. The aggregated feature of the i-th agent, denoted as H_i, is obtained by combining the ego feature F_i^t, the intra-agent fusion $\hat{H}_i$, and the inter-agent collaboration information H_i'. Finally, H_i is fed to the decoder network f_{dec} to obtain the predicted outputs Y^{cls} and Y^{reg} for object classification and bounding box regression, respectively.

B. FEATURE EXTRACTOR

The feature extractor extracts semantic information from raw point cloud data. Each agent N extracts BEV features from local point clouds via a parameter-shared feature encoder $f_{\text{enc}}(\cdot)$. We adopt the backbone of PointPillar [23] due to its low inference latency and efficient memory usage, consistent with the literature [9], [13], [15], [17]. The encoder converts the raw point clouds into the stacked pillars and scatters them into a 2D pseudo-image. A feature pyramid network processes the pseudo-image and outputs the final features. Given the point cloud $\mathcal{X}_N$ of k-th agent at timestamp t, the BEV feature is $F_N^t = f_{\text{enc}}(\mathcal{X}_N) \in \mathbb{R}^{C\times H\times W}$, where C, H, W stand for the channel, height, and width.

C. INTRA-AGENT FUSION MODULE

Intra-agent fusion operates exclusively on combining the ego vehicle's current and historical frames. This is advantageous, especially during high transmission delays, or interruptions between agents, as this enhances the ego agent's situational awareness ability by relying on its historical context without the need to be dependent on other neighboring agents. As illustrated in Fig. 2, the proposed intra-agent fusion model comprises two core components. The first is the *Dilated Temporal Convolution* (DTCA) module, which is designed to enhance and reinforce the acquisition of both historical and current frame context. The dilated convolution enables large receptive fields and consequently captures a long memory, which is not possible with causal convolutions alone as they require a very deep neural network architecture [24]. The second module is the *Temporal Self-Attention* (TSA), which deploys an attention mechanism to direct the model to learn and capture temporal dependencies across the frames, thereby enabling the extraction of more intricate spatial information which enriches the capabilities of the intra-agent fusion. Temporal Convolutional Networks (TCNs) are a group of temporal processing methods [25], which replace the recurrence of recurrent neural networks (RNNs) with convolutions over temporal data. TCNs use dilated convolutions to process the entire input sequence in parallel, rather than sequentially as in RNNs. This enhances memory utilization and efficiency during training as less memory is required for storing intermediate results during training compared to RNNs. The use of dilated convolutions results in a large receptive field, meaning that the network can capture contextual dependencies while keeping the computational cost low. Additionally, the dilation processes the sequences/frames in parallel, making them faster to train than RNNs. Therefore, we adopt dilated temporal convolution to capture the spatiotemporal dependencies, along with contextual information from the input sequence, at low computational cost, and efficient memory.

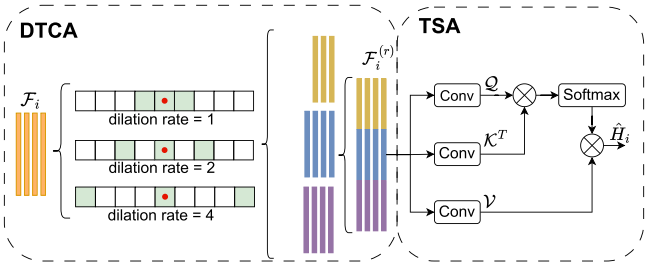

FIGURE 2. The overview of the intra-agent fusion module.

Our proposed DTCA acts as a temporal filter across frames, fusing the historical context of the ego agent with the current frame. Our proposed DTCA assigns temporal attention weights at multiple scales, unlike standard TCNs that uniformly weigh input features or frames within the kernel. This is advantageous as events or patterns in time often operate at multiple levels of granularity. In any time-dependent process, meaningful information can exist in both short-term interactions (small scale) and long-term trends or dependencies (large scale) that require varying scales across different frames. To achieve this, we employ multiple dilation rates to the convolution network, enabling the model to learn dependencies across diverse temporal scales. This

approach increases the receptive field size without introducing additional parameters or computational overhead, thereby improving detection accuracy while maintaining efficient inference time. Moreover, extending the receptive field in the temporal dimension enhances the extraction of long-range temporal features. The use of multi-scale temporal information allows the network to capture motion at varying scales, with larger receptive fields detecting slower movements and smaller ones responding to rapid changes. This multi-scale integration enables the network to effectively model dynamic object motion over time.

Our method utilizes three parallel learnable 1×1 dilated convolutions with fixed kernels, enabling it to process feature maps exclusively in the temporal domain, preserving spatial information. As shown in Fig. 2, we utilize three different dilation rates to model different spatiotemporal information. A small dilation rate of 1 is used to capture short-range temporal information, while the larger dilation rates of 2 and 4 are used to capture longer-range temporal information. The adopted dilation factor is 2in this work to achieve an exponential receptive field. The dilated convolution is computed as follows:

$$\mathcal{F}_i = \text{concat}(F_i^t, F_i^{t-1}, \ldots, F_i^{t-k}) \tag{2a}$$

$$f^{r_n} = \text{DilatedConv}_{1\times1}^{r_n}(\mathcal{F}_i) \tag{2b}$$

where $\mathcal{F}_i$ is the concatenated current and historical frames, i.e., $\{F_i^t, F_i^{t-1}, \ldots, F_i^{t-k}\}$, where k is the historical frame number. $f^{(r_n)}$ is the resulting feature map after applying the DilatedConv$^{(r)}$, and n is the number of parallel dilated convolution branches. DilatedConv is the fixed convolutional kernel 1×1, r denotes the dilated convolution with a dilation rate indicating the temporal stride for sampling frame features. In our work, n is fixed to 3 branches, increasing progressively from 1 to 2 to 4 to enlarge the temporal receptive fields. Afterward, we concatenate the outputs from the dilated convolution branches (i.e., 3 branches) resulting in the updated temporal feature $\mathcal{F}_i^{(r)}$, which is computed as:

$$\mathcal{F}_i^{(r)} = \text{concat}\left(f^{(r_1)}, f^{(r_2)}, \ldots, f^{(r_n)}\right) \tag{3}$$

Local features extracted by the DTCA layer are fed to the TSA to attend to features across the entire feature map $\mathcal{F}_i^{(r)}$, enabling the model to handle dependencies in both time and spatial domains. First the concatenated feature $\mathcal{F}_i^{(r)}$ is projected into query $\mathcal{Q}$, keys $\mathcal{K}$, and values $\mathcal{V}$ using a 1×1 convolution operation, as shown in Fig. 2. Consequently, we apply scaled dot product attention similar to [26]. The attention weights are used to compute the intra-agent fusion feature representation $\hat{H}_i$. This gives us the attended output for the $\mathcal{Q}$, effectively aggregating information from different parts of the input sequence, the TSA operation is computed as follows:

$$\hat{H}_i = \text{Atten}(\mathcal{Q}, \mathcal{K}, \mathcal{V}) = \text{softmax}\left(\frac{\mathcal{Q}\mathcal{K}^T}{\sqrt{d_k}}\right)\mathcal{V} \tag{4}$$

By attending to relevant parts of $\mathcal{F}_i^{(r)}$ dynamically, TSA helps models understand dependencies in sequences better than fixed-window approaches like recurrent networks.

D. COLLABORATOR SELECTION AND SPATIAL TRANSFORMATION

At each timestep, the neighboring agent j broadcasts its pose ζ_j and feature map F_j to its neighboring agents. Subsequently, the ego agent i employs a pre-defined relevancy metric to evaluate whether agent j is pertinent to its current situation. This metric considers the j-th agent relevant if it lies within a 70 m radius or a heading intersection of 70 degrees from the ego agent. This relevancy metric range is based on existing dedicated short-range communications (DSRC) standards [27], this metric has also been adopted in [9].

Each agent possesses its unique pose, therefore feature map $F_j^{\tau_j}$ received from the neighboring agent j needs to be transformed to the ego agent's i perspective using the i-th and the neighboring agent's j poses ζ_i^t, $\zeta_j^{\tau_j}$, receptively. In this work, we adopt the affine transformation due to its ability to preserve parallel lines and distance during rotations. The affine transformation adopted in this work is closely aligned with the method proposed in [28], with the key distinction being the absence of a localization network, as each agent broadcasts its pose along with the feature map. The transformation operates on the entire $F_j^{\tau_j}$ in a non-local manner in two stages: (1) grid generator and (2) grid sampler. The transformation matrix $\zeta_{j\to i}$ generates a sampling grid, which determines the points where the $F_j^{\tau_j}$ will be sampled. This grid, created within the grid generator stage, defines the transformation such as rotation and translation that needs to be carried out. Afterward, the grid sampler applies the sampling grid to $F_j^{\tau_j}$, sampling it at the grid-specified positions using bi-linear interpolation to manage non-integer positions. This results in the feature map $F_j^{\tau_j}$ of agent j being transformed to the ego agent's perspective. The ego agent repeats this affine transformation process for all the received feature maps, and once all feature maps are transformed, the ego and the transformed feature maps are passed to the inter-agent fusion module.

E. INTER-AGENT FUSION MODULE

Although the positional misalignment is captured by the spatial warping matrix $\zeta_{j\to i}$, another type of misalignment, arising due to transmission delay also needs to be considered i.e. the time between the frame was captured by the neighboring agent until it was received by the ego agent. Therefore, in order to counter this issue, the temporal delay information needs to be considered and learned within the network. To achieve this we first need to represent the time delay in a form that can be embedded within the neighboring agent's feature map. In this work, we define τ_j to represent the total time delay i.e. the time needed for the neighboring agent to capture the scene, until it is received by ego agent. Previous work [13], [16], [17], [20] adopted a fixed time delay. However, we argue that time delay cannot be fixed as it depends on multiple

TABLE 2. Experimental Parameters

Parameter	Value
Carrier frequency f_c	5.9 Ghz
Total bandwidth	20 M
Transmit power p_{ji}^{trans}	23 dbm
Sensor asynchronous overhead τ_j^{asyn}	-95 dbm -110 dbm
Semantic extraction time τ_j^{ext}	-100ms 100 ms
Sensor sampling interval	100 ms
Number of historical prior frames k	2

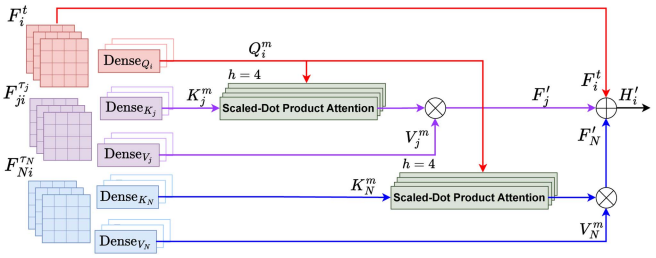

FIGURE 3. Overview of the Attention-based inter-agent Feature Fusion component.

factors. Similar to [15], we compute τ_j including realistic metrics. In our work τ_j is defined as the summation of:

- inter-agent asynchronous overhead τ_j^{asyn},
- time required for feature extraction τ_j^{ext}
- collaborative message transmission latency τ_j^{trans}
- synchronization between the perception system and communication system τ_j^{idle} [17].

This is expressed as:

$$\tau_j = \tau_j^{\text{asyn}} + \tau_j^{\text{ext}} + \tau_j^{\text{trans}} + \tau_j^{\text{idle}} \tag{5}$$

According to the specification 3GPP TR 38.901 [29], network transmission latency τ_{ji}^{trans} can be approximately derived from a path-loss driven channel as follows:

$$\tau_{ji}^{\text{trans}} = \frac{\text{size}(F_j^{\tau_j})}{b_{ji}\log_2(1 + 10^{0.1(p_{ji}^{\text{trans}} - p_{ji}^{\text{loss}} - p_{ji}^{\text{noise}})})} \tag{6}$$

where $\text{size}(F_j^{\tau_j})$ denotes a function that calculates the size of the transmitted feature map from agent j. b_{ji}, p_{ji}^{trans}, p_{ji}^{noise}, and p_{ji}^{loss} represents agent j's transmission bandwidth, transmission power, transmission noise power, and transmission path loss to agent i, respectively. With being derived as $p_{ji}^{\text{loss}} = 28.0 + 22\log_{10}(d_{ji}) + 20\log_{10}(f_c)$. Where, d_{ji} denotes the distance between agent j and i in meters, and f_c represents the center frequency in GHz. The parameters b_{ji}, p_{ji}^{trans}, p_{ji}^{noise}, τ_j^{asyn}, τ_j^{ext} are set as range values presented in Table 2). LTE-V2V for the cooperative awareness of connected vehicles [24] (6) is further adopted for a more realistic calculation of the transmission time, where the related parameters are consistent with the literature [34]. Notably, we allocate equal bandwidth to each individual agent.

Temporal encoding: The temporal encoding within our proposed inter-agent fusion scheme is illustrated in Fig. 3(a). As discussed earlier in this section our model employs variable time delays as presented in (6) and parameters in Table 2 to mimic real-life situations. Therefore to include this temporal misalignment, we encode these varying time delay values associated with every selected collaborator to incorporate it within the feature map prior to the inter-feature aggregation. With each agent, we adopt a relative temporal encoding (RTE) [30] which is inspired by Transformer's positional encoding method [31]. Given the corresponding timestamps of the selected collaboratorτ_j and ego agent t_i the relative time delay is represented as $(\Delta\tau_{ij} = \tau_j - t_i)$ as an index to get a relative temporal encoding $RTE(\Delta(\tau ij))$. The variable time delay typically ranges between 80–110 ms, therefore another benefit of encoding the time delay within our inter-agent fusion scheme, is that during training not all possible time delays are covered as the communication parameters shown in Table 2 are selected randomly with every iteration. Thus the RTE should enable generalizing to unseen times and time delays. To encode the time delay, we adopt a fixed set of sinusoid functions as a basis conditioned on the time delay τ_{ij} and channel $c \in [1, C]$ as follows:

$$\text{Base}(\Delta\tau_{ij}) = \begin{cases} \sin\left(\frac{\tau_j}{10000^{\frac{2c}{C}}}\right), & c = 2k \\ \cos\left(\frac{\tau_j t}{10000^{\frac{2c}{C}}}\right), & c = 2k+1 \end{cases}$$

A tunable linear projection $T - linear : \mathbb{R}^C \to \mathbb{R}^C)$ is then applied to further warp the learnable embedding so it can generalize better for unseen time delays [30]:

$$RTE(\Delta\tau_{ij}) = \text{T-Linear}(\text{Base}(\Delta\tau_{ij})) \tag{7}$$

Subseqquently the projected relative temporal embedding $RTE(\Delta\tau_{ij})$ is added to each agent's feature F_{ji} as follows:

$$F_{ji}^{\tau_j} = F_{ji}^{\tau_j} + RTE(\Delta\tau_{ij}) \tag{8}$$

In this way, the temporal augmented representation $F_{ji}^{\tau_j}$ will capture the relative temporal information of selected collaborator j. This process is repeated for all the selected collaborators.

Attention-based Inter-Agent Feature Fusion: Given an ego agent i, and all the selected collaborators $j \in N$, we want to calculate their mutual importance to each other and attend to the significant regions within their feature maps. Towards this goal, we utilize the attention mechanism, inspired by the architecture design of Transformer [26], to extract attention maps to attend to spatial regions within the feature maps of agents i, and j, the attention-based feature fusion is expressed as follows:

$$F_j' = \text{Attention}(i, j) \cdot \text{Message}(j) \tag{9}$$

where **Attention** estimates the attention weight of each selected collaborator in regard to the ego agent, and **Message** is the message aggregator. We further refine the attention map

the attention map learning process by utilizing multi-head attention, which enables the model to simultaneously attend to information across different subspaces. We project the ego agent into a query Q_i^m, whereas the selected collaborator into a key and value K_j^m, and V_j^m respectively, with m representing the current head number and h is the total number of heads i.e. 4 performs the attention function in parallel. This is expressed as follows:

$$Q_i^m = \text{Dense}_i^m(\mathbf{F_i^t}), \tag{10a}$$

$$K_j^m = \text{Dense}_j^m(\mathbf{F_{ji}^{\tau_j}}), \tag{10b}$$

$$V_j^m = \text{Dense}_j^m(\mathbf{F_{ji}^{\tau_j}}) \tag{10c}$$

The linear aggregator is denoted by Dense : $\mathbf{R^C} \rightarrow \mathbf{R^{\frac{C}{h}}}$ $\frac{C}{h}$ is the vector dimension per head. Each Dense is indexed by it's respective agent to model the distribution differences maximally. Next, we calculate the attention between the query vector Q_i^m and the key vector K_j^m as follows:

$$\text{Attention}(i,j) = \underset{j\in\mathcal{N}}{\sigma}\left(\Big\|_{m\in[1,h]} \text{head}^m_{\text{Attention}}(i,j)\right) \tag{11a}$$

$$\text{head}^m_{\text{Attention}}(i,j) = \left(\frac{Q_i K_j^\top}{\sqrt{C}}\right) \tag{11b}$$

where $\|$ denotes concatenation and σ represents softmax. The resulting attention map signifies the relevance of a collaborator's semantic information. Note that eventhough the multi-head attention adds extra complexity to the model the overall computational cost remains comparable to that of the single-head attention [26].

$$\text{Message}(j) = \Big\|_{m\in[1,h]} V_j^m \tag{12}$$

Finally, the inter-agent feature H_i' is computed as:

$$H_i' = \sum_{j\in N}(F_j') + \mathbf{F_i^t} \tag{13}$$

F. ADAPTIVE FEATURE FUSION

Previous work on collaborative perception approaches [9], [10], [11], [17], [19] has achieved remarkable object detection performance relying heavily on the received perceptual information from collaborators. However, relying only on the collaborator's perceptual information invariably introduces noise interferences caused by asynchronous measurement, imperfect localization sensors, and transmission delays, leading to sub-optimal detection performance. In this work, we look at collaboration from a different perspective, i.e. in addition to the inter-agent feature H_i', we further incorporate purely ego-centered features to counteract and refine the noisy information (such asynchronous measurement, feature misalignment noise, localization errors, etc) that may be introduced due to cross-agent feature fusion. The intuition is that the historical semantic information of the ego contains rich object information, and the ego's current representation

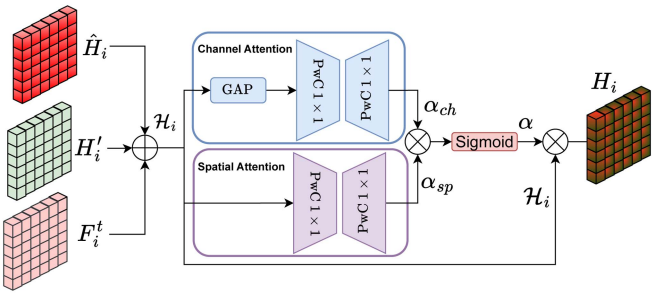

FIGURE 4. The architecture of the proposed Adaptive Fusion module.

embodies the natural perception advantages, like: the right perspective view (no spatial transformation required), correct location of the agent, and spontaneous availability even during network congestion. To this end, we propose an adaptive fusion to fuse multi-source features retrieved from the previously mentioned modules based on their complementary contributions. Our proposed adaptive fusion modulates and integrates three informative features, including the intra-agent feature $\hat{H}_i$, the collaboration features H_i' and the ego agent feature F_i^t. Fig. 4 illustrates the overview of the adaptive fusion scheme. The aggregation of the three features is drafted as follows:

$$\mathcal{H}_i = \text{AGG}(F_i^t, H_i', \hat{H}_i) \tag{14}$$

where AGG represents the summation of the three features, resulting in the aggregated representation $\mathcal{H}_i \in \mathbb{R}^{C\times H\times W}$. Subsequently, we utilize our work developed in [32] to direct the model focus on relevant information by learning the attention map. This guides the model on "where" and "what" to focus within the $\mathcal{H}_i$. We leverage both channel and spatial attention to exploit their respective benefits to highlight the significant regions within $\mathcal{H}_i$. This strengthens the representation power of the adaptive feature fusion. This attention mechanism guides the model to focus on important features for fusion while suppressing irrelevant ones, and adaptively highlighting significant regions. Since each channel represents different aspects or features of the feature map (like edges, textures, etc.), the goal of channel attention is to identify the relationship between various channels by adaptively determining each channel's significance during the network training process.Channel attention maps important channels based on the features within each channel, prioritizing relevant channels and leading to improved representation power. The channel attention map $\alpha_{ch} \in \mathbb{R}^{C\times H\times W}$ is learned as follows:

$$\alpha_{ch} = \Gamma_{ch}(\text{GAP}(\mathcal{H}_i)) \tag{15}$$

where Γch and channel point-wise convolution (PwC) and GAP represents the global average pooling, respectively.

In addition to channel attention, in parallel, we incorporate spatial attention. Spatial attention complements channel attention by focusing on the spatial dimension of the feature map $H \times W$. Similar to channel attention we employ PwConv

autoencoder to generate the spatial attention map α_{sp}. Because PwConv only operates on a single spatial location it is computed very efficiently, even for large feature maps. This makes PwConv well-suited for use in spatial attention mechanisms, where attention weights need to be computed for each spatial location in the feature map generating a spatial attention map $\alpha_{sp} \in \mathbb{R}^{C\times H\times W}$ which is computed as follows:

$$\alpha_{sp} = \Gamma_{sp}(\mathcal{H}_i) \tag{16}$$

Subsequently, the distinct attention scores, denoted as α_{sp} and α_{sp}, are element-wise summed ($\oplus$) to obtain the final attention score α. Following this, α is element-wise multiplied ($\otimes$) with the aggregated feature $\mathcal{H}_i$ to generate the final feature representation $H_i \in \mathbb{R}^{C\times H\times W}$, as expressed by the following equation:

$$\alpha = \sigma(\alpha_{ch} \oplus \alpha_{sp}) \tag{17a}$$

$$H_i = \alpha \otimes \mathcal{H}_i \tag{17b}$$

G. DETECTION DECODER

We adopt two convolution-based detection decoders [23], denoted as $f^{\text{reg}}_{\text{dec}}(\cdot)$, $f^{\text{cls}}_{\text{dec}}(\cdot)$, to transform the fused feature H_i into the prediction results. The regression results encompass information about the position, size, and yaw angle of the predefined bounding box at each location, expressed as $Y^{\text{reg}} = f^{\text{reg}}_{\text{dec}}(H_i) \in \mathbb{R}^{7\times H\times W}$. In addition, the classification results are given by $Y^{\text{cls}} = f^{\text{cls}}_{\text{dec}}(H_i) \in \mathbb{R}^{2\times H\times W}$, providing the confidence scores of each bounding box to be an object or background.

IV. EXPERIMENTS

A. EXPERIMENTAL SETTINGS

All experiments in this article are conducted on open collaborative perception datasets **OPV2V** [12], **V2XSet** [17] and **V2VReal** [18]. OPV2V is a large-scale simulated dataset for multi-agent V2V perception generated using CARLA [33] and SUMO [34], comprising 10,914 LiDAR point cloud frames with 3D annotation. The training/validation/testing splits include 6,764, 1,981, and 2,169 frames, respectively. On the other hand, V2XSet is a simulated dataset supporting V2X perception, generated employing Carla [33] and OpenCDA [35]. It includes 73 representative scenes with 2 to 5 connected agents and 11,447 3D annotated LiDAR point cloud frames. The training/validation/testing sets are 6,694, 1,920, and 2,833 frames, respectively. The V2VReal dataset comprises collaborative perception data collected in real-world driving scenarios involving two vehicles operating simultaneously in Columbus, Ohio. The vehicles maintained a maximum separation of 150 meters to ensure overlapping fields of view. The dataset encompasses 19 hours of driving, covering a total of 347 km on highways and 63 km on city roads. It includes 20,000 frames of LiDAR point cloud data, which are partitioned into training, validation, and test sets containing 14,210, 2,000, and 3,986 frames, respectively. As presented in Section. III-G the loss function consists of a classification and a regression loss. In line with PointPillars [23], we utilize focal loss [36] for the classification loss and smooth L1 loss [37] for the regression loss.

TABLE 3. Overall Perception Performance

Dataset	Methods	Average Precision (%)		
		IoU 0.3	IoU 0.5	IoU 0.7
OPV2V	No Fusion	79.81	77.70	62.12
	V2VNet	82.71	80.38	52.96
	V2X-ViT	87.09	85.89	75.56
	Where2comm	86.71	85.20	71.35
	Select2Col	89.80	88.51	77.65
	DelAwareCol	**91.13**	**89.50**	**78.64**
V2XSet	No Fusion	75.55	71.17	47.12
	V2VNet	77.85	72.27	46.99
	V2X-ViT	81.54	78.76	62.36
	Where2comm	82.70	79.05	57.16
	Select2Col	87.18	84.25	64.92
	DelAwareCol	**90.80**	**88.51**	**68.71**
V2VReal	No Fusion	46.77	37.23	15.55
	V2X-Vit	55.08	47.78	23.74
	Where2comm	60.28	48.32	20.55
	Select2Col	62.20	50.19	24.89
	DelAwareCol	**65.30**	**52.73**	**27.53**

To comprehensively evaluate DelAwareCol we focus on LiDAR-based object detection. The detection performance is measured using the average precisions (AP) metric at an intersection over union (IoU) thresholds of 0.3, 0.5, and 0.7. We utilize the Adam optimizer [38] with an initial learning rate of 10^{-3} and steadily decay it every 10 epochs using a factor of 0.1. All experiments are conducted on a station equipped with Intel i7-11700 @2.50GHz NVIDIA Tesla V100 GPU. We comaprare our proposed DelAwareCol againstthe state-of-the-art collaborative perception frameworks: V2VNet [9], V2X-ViT [17], Where2comm [13], and Select2Col [15]. As presented in Table 2, the key parameters employed in our study are in line with V2VNet [9], V2X-Vit [17], Where2comm [13], and Select2Col [15]. In addition, (6) is further adopted for a more realistic calculation of the transmission time, where the related parameters are consistent with the literature [39]. Notably, we allocate equal bandwidth to each individual agent.

B. QUANTITATIVE ANALYSIS

We present a comprehensive quantitative analysis to evaluate the detection performance, computational efficiency, and robustness under localization errors and transmission delays of the proposed DelAwareCol method. Our evaluation is conducted on multiple datasets, and the results are benchmarked against state-of-the-art methods.

Detection Performance Comparison: In this experiment, we evaluate the overall perception performance of our proposed method on OPV2V V2XSet datasets. Table 3 provides a performance comparison of the proposed method

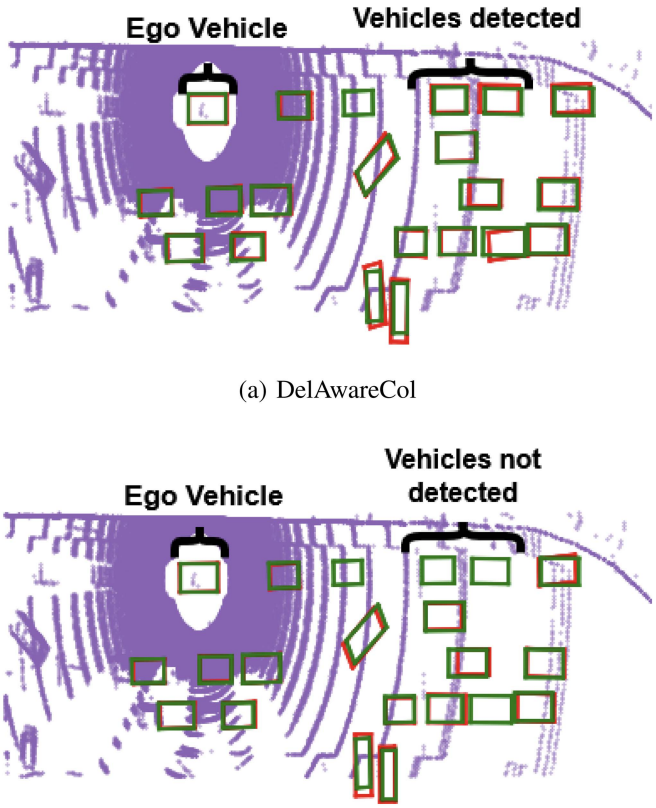

FIGURE 5. BEV object detection visualizations on the V2X-Set dataset, comparing detection results of DelAwareCol and Select2Col. Ground truth and detected vehicles are represented by green and red bounding boxes, respectively. The results demonstrate that DelAwareCol detects 2 extra vehicles compared to Select2Col, as shown by the parenthesis.

with the existing SOTA methods, V2VNet [9], V2X-ViT [17], Where2comm [13] and Select2Col [15]. Intuitively our method outperforms V2VNet [9], V2X-ViT [17], Where2comm [13], and Select2Col [15] by large margins across all datasets, demonstrating the superiority of our perception paradigm. For example, at an IoU threshold of 0.7, our method achieves gains of 37.55%, 9.69%, 18.35%, and 5.67% AP performance against V2VNet, V2X-ViT, Where2comm, and Select2col respectively, on the V2XSet dataset. It results demonstrate that DelAwareCol extra vehicles compared to Select2Col. On the OPV2V dataset, DelAwareCol proves to be highly effective and robust by achieving higher detection performance against Select2Col, and remarkable gains compared to V2VNet, V2X-ViT, and Where2comm. Additionally, Fig. 5 illustrates the detection visualizations on the V2X-Set dataset comparing detection results of DelAwareCol and Select2Col. Furthermore, on V2V4Real, the real-world dataset, DelAwareCol proves to be highly effective and reliable obtaining detection performance gains compared to the benchmarked methods. The detection gains achieved by DelAware are attributed to the introduction of multiple fusion modules.

TABLE 4. Overall Perception Performance Under Localization Errors

Dataset	Methods	Average Precision (%)		
		IoU 0.3	IoU 0.5	IoU 0.7
OPV2V	No Fusion	79.81	77.70	62.12
	V2VNet	82.24	78.33	44.72
	V2X-ViT	86.76	85.41	74.51
	Where2comm	86.47	84.30	66.52
	Select2Col	89.73	88.20	74.54
	DelAwareCol	**90.54**	**88.63**	**76.12**
V2XSet	No Fusion	75.55	71.17	47.12
	V2VNet	76.12	68.54	36.23
	V2X-ViT	80.77	77.02	56.90
	Where2comm	80.74	74.89	51.15
	Select2Col	84.90	78.74	58.41
	DelAwareCol	**90.23**	**86.94**	**64.92**

First, incorporating the time delay within the inter-agent fusion enhances the collaboration between the ego and the collaborating agents, mitigates misalignment occurring due to temporal delays. Second, the inclusion of the ego agent's historical data further enhances the final fused semantic information's representation power from the temporal dimension, providing temporal context. Finally, the adaptive fusion serves as a module selection mechanism, focusing only on relevant semantic information produced by the different fusion modules and filtering out features that would degrade detection performance.

Detection Robustness under Localization Errors: Notably, in real-life, localization errors can be widely observed due to various factors that can degrade GPS signal accuracy or disrupt their transmission. To evaluate the robustness of our proposed method, we follow localization noise similar to [15], [17] to the test data, which follows a Gaussian distribution with a standard deviation of 0.2 for both position and heading. From Table 4, localization noise deteriorates the perception performance of all approaches, and the reduction is more noticeable when the IoU threshold is larger. This indicates that high-precision detection is more vulnerable to noise. The performance of V2VNet, V2X-ViT, Where2comm, and Select2Col is significantly degraded, approaching the performance of the No Fusion method in the case of V2XSet. In contrast, our approach consistently outperforms these SOTA models, demonstrating robustness against localization errors. The resilience to localization errors are attributed to the utilization of ego-centered features, which is achieved through two distinct modules: **(i)** the intra-agent fusion (discussed in Section 2), which captures temporal correlations between the ego agent's current and historical features. By maintaining this temporal record, the ego agent constructs an internal model of the environment that captures its dynamics and evolution, thereby mitigating feature misalignment occurring in the inter-agent fusion due to localization noise; and **(ii)** the incorporation of pure ego features in the adaptive feature fusion module (discussed in Section 4), which remains slightly

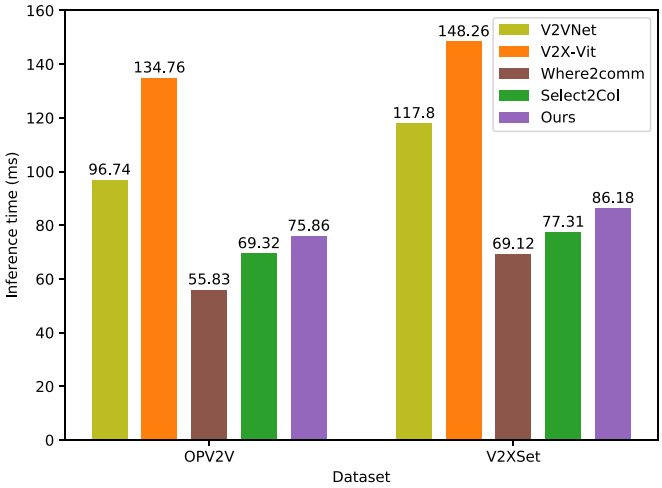

FIGURE 6. Comparison of the inference time on the OPV2V and V2XSet datasets.

effected by localization related noise arising due to GPS errors and orientation misalignment. Together, these two approaches significantly reduce the adverse impact of localization errors.

Robustness to Transmission Delay: This experiment aims to evaluate the influence of transmission delay on detection performance. Unlike the Select2Col framework, which adds delay for only one collaborator, we argue that since all collaborators share the same feature extractor and decoder network, any additive delay is attributed to network's characteristics degradation which impacts the transmission for all connected agents. This degradation affects all agents connected to the network. Fig. 7 illustrates the detection results at varying IoU thresholds under different delay conditions. Consequently, for a more realistic evaluation, we maintain the neighboring agents' latency τ_j as constant values ranging between 0-1000ms, as shown on the x-axis of Fig. 7. The overall results indicate that larger latencies negatively affect detection performance for all models. However, our DelAwareCol demonstrates superior detection performance and the highest resilience to latency across all tested approaches, showing stability during high-latency conditions. The experimental results confirm that collaborators with significant latency contribute minimally or even negatively to enhancing the perception performance of the ego agent.

This is attributed to: **i)** including the temporal delay in the collaborator's feature map (through the temporal encoding) within inter-fusion module guarantees the elimination of collaborators with huge time delays and asynchronisity by giving that collaborator a smaller weight during the aggregation process as it harms the detection performance. In contrast, other methods either overlook collaborator selection or rely solely on spatial criteria, resulting in deteriorating perception performance as latency increases. This underscores the importance of considering both the both temporal and spatial dimensions to achieve optimal inter-agent feature fusion to mitigate delay's negative impact. **ii)** the inclusion of the pure ego as well the intra-features within the adaptive fusion. Which allows the agent to leverage both its current and historical frames, enabling resilience to high delays by relying on its own undelayed data. Thus, our approach ensures stable perception performance under significant latency, proving to be more robust than existing methods.

TABLE 5. Number of Model Parameters

Methodology	Number of Parameters (M)
V2VNet	14.6122
V2X-ViT	12.4554
Where2Comm	8.0574
Select2Col	8.2875
DelAwareCol	**8.5634**

TABLE 6. Impact of Historical Frame Number

No. of historical frames (k)	V2XSet			OPV2V		
	AP@0.3/0.5/0.7			AP@0.3/0.5/0.7		
t-1	89.65	87.17	67.94	89.96	88.21	77.93
t-1, t-2 (Default)	**90.80**	**88.51**	**68.71**	**91.13**	**89.50**	**78.64**
t-1, t-2, t-3	90.29	87.97	68.34	90.35	88.67	78.12

TABLE 7. Impact of Core Modules

Significance of proposed modules	V2XSet			OPV2V		
	AP@0.3/0.5/0.7			AP@0.3/0.5/0.7		
w/o Intra-Vehicle Fusion ($\hat{H}_i$)	89.75	86.90	67.63	90.10	87.93	75.45
w/o Inter-Vehicle Fusion (H_i')	68.23	64.63	41.51	68.78	65.81	52.13
w/o Ego Feature (F_i^t)	89.28	87.45	68.0	88.56	88.61	77.84
Default	**90.80**	**88.51**	**68.71**	**91.13**	**89.50**	**78.64**

Computational Efficiency Comparison: Table 5 compares the parameter counts of our proposed method with SOTA methodologies. Our model has a significantly lower parameter count than V2VNet and V2X-ViT while being marginally higher than Where2comm and Select2Col. Specifically, our model exhibits a 3.33% increase in parameter count compared to Select2Col, which achieves the second-highest AP, as shown in Table 3. Furthermore, as illustrated in Fig. 6, the inference time of our method is considerably shorter than that of V2VNet and V2X-ViT, though slightly longer than Where2comm and Select2Col. Nevertheless, our approach surpasses both Where2comm and Select2Col in detection performance. When it comes to the parameter count, DelAwareCol has slughly higeher count when compared to Where2Comm and Select2Col. This slight increase is due to incorporating complex architectural design within the intra-agent and inter-agent fusion modules that boosted the detection performance. Overall, DelAwareCol achieves superior detection performance even during localization error while maintaining computational efficiency comparable to other SOTA techniques.

V. ABLATION STUDY

We perform thorough ablation studies on all datasets to understand the necessity of the different modules and

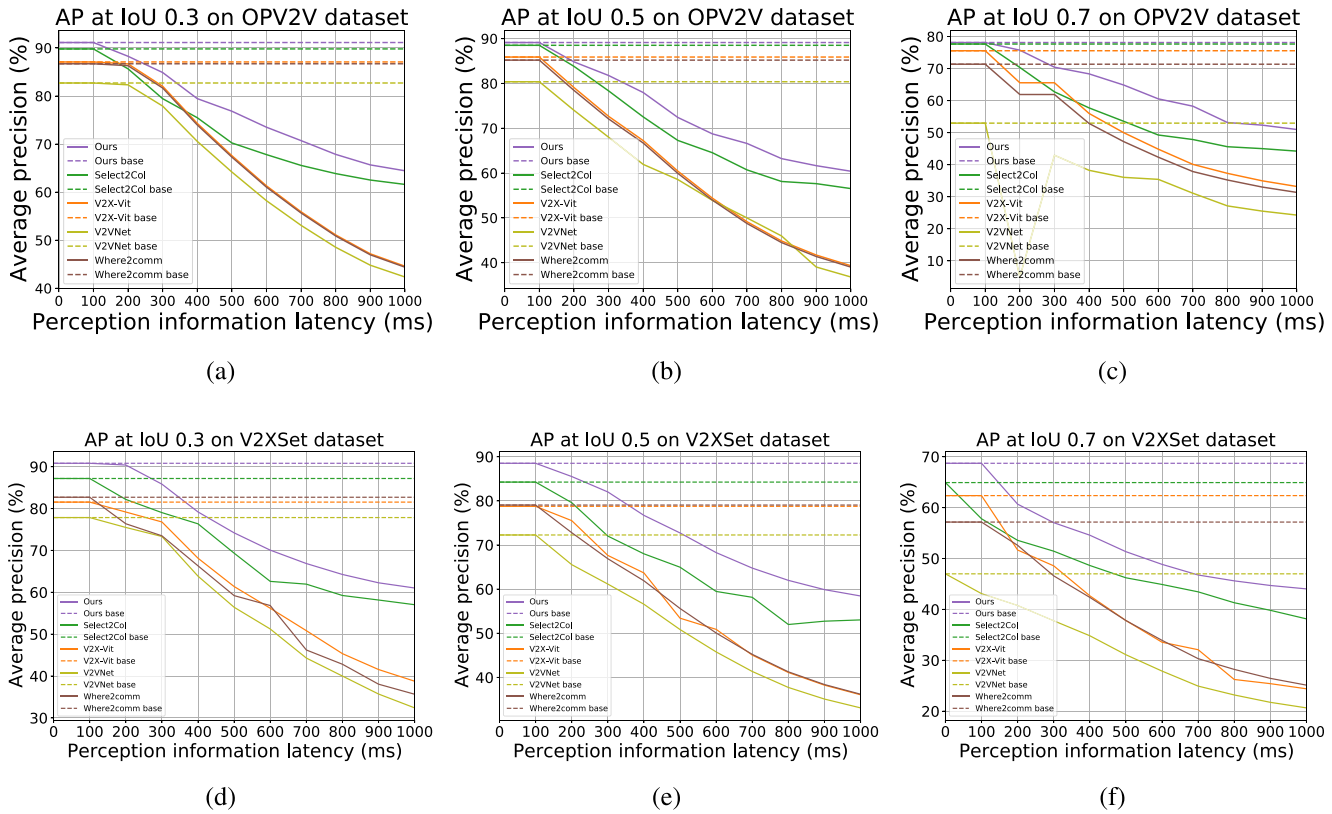

FIGURE 7. Detection performance and robustness to the transmission delay on the OPV2V and V2XSet datasets.

TABLE 8. Impact of Attention in Adaptive Fusion Module

Effect of Fusion Strategies	V2XSet AP@0.3/0.5/0.7			OPV2V AP@0.3/0.5/0.7		
w/ Summation	83.13	81.24	62.23	83.92	83.34	72.87
w/ Concatenation	84.10	82.52	62.78	84.65	84.48	73.52
w/ Adaptive Fusion (Default)	**90.80**	**88.51**	**68.71**	**91.13**	**89.50**	**78.64**

design considerations. Tables 7, 6, and 8 show the following observations.

Impact of Historical Frame Number: Table 6 presents the effects of incorporating varying levels of temporal context from the ego agent, with frame numbers set to 1, 2, and 3, respectively. The results indicate that utilizing two historical frames effectively captures meaningful temporal cues, leading to improved performance. For example, on the V2XSet dataset with an AP of 0.3, incorporating two historical frames results in improvements of 1.28% and 0.56% compared to using a single frame and three frames, respectively. This finding highlights the advantage of leveraging two historical frames for enhanced learning of temporal correlations. However, extending the detection beyond two frames results in reduced detection performance, primarily attributable to the dynamic nature of driving environments, where the relative positions of objects undergo continuous change with respect to the ego vehicle. In such scenarios, objects may undergo significant displacement or disappear entirely in successive frames, which ultimately degrades object detection performance.

Impact of Core Modules: In this experiment, we analyze the impact of the core modules on detection performance. We investigate the impact by excluding intra-agent $\hat{H}_i$, inter-agent H_i', and ego-agent features inclusion from the adaptive feature module F_i^t.

The result of every experiment is shown in Table 7, overall, the removal of each module yields a decrease in the detection performance when compared to the default setting, which proves that each module contributes to achieving robust collaboration. The key findings of our experimental analysis is summarized as follows (Note that, to compare the performance of each exclusion quantitatively we will use the detection performance of V2XSet at AP of 0.5 shown in Table 7 as an example):

- *Excluding the intra-agent feature $\hat{H}_i$,* that encodes ego-agent contextual information causes the detection performance to be degraded sightly by 1.85%. Though this is a slight degradation, it underscores the critical role of learned temporal correlations as a vital information resource. These correlations enhance the ego-agent's perception robustness, particularly in challenging scenarios where communication with neighboring agents is disrupted or unreliable. In safety-critical applications, even small performance losses can have significant consequences, making the preservation of intra-agent features essential for maintaining system resilience and accuracy.

- *Excluding the inter-agent feature* H_i', i.e. relying solely on the ego-agent perception (ego centric feature and intra-agent feature, results in significant performance degradation of 36.94% in detection performance. This highlights the important role of multi-agent information dominating various core modules.
- *Excluding the ego-agent in the adaptive feature fusion* F_i^t leads to a degradation of 1.2% in detection performance. While this may appear minor at first glance, it underscores the critical role of incorporating purely ego features to mitigate the impact of noisy information introduced during inter-agent collaboration. In applications like autonomous driving, where even small improvements in detection accuracy can significantly enhance safety and reliability, this degradation is far from negligible. Addressing this issue ensures robust performance and highlights the importance of balancing ego-centric and collaborative features for optimal system behavior.

Impact of Attention in Adaptive Fusion Module: The impact of the proposed adaptive fusion strategy for aggregating features F_i^t, $\hat{H}_i$, and H_i' is demonstrated. To assess this impact, the adaptive fusion module is replaced with summation and concatenation operations. In the summation approach, features are combined via pixel-wise addition, whereas in the concatenation operation, features are concatenated and reshaped to meet the dimensional requirements of the decoder network. As shown in Table 8, both summation and concatenation fail to address the complexity of multi-source feature fusion effectively. For example, on the V2XSet dataset with an AP at IoU of 0.7, the default attention-based adaptive fusion demonstrates detection performance improvements of 10.41% and 9.45% compared to summation and concatenation, respectively. In contrast, the proposed adaptive fusion module outperforms these methods by utilizing both channel and spatial attention mechanisms. This dominance is attributed to the fact that channel and spatial attention assigns greater importance to significant semantic features, enhancing detection performance by exploiting the complementary contributions of each feature.

VI. CONCLUSION

In this paper, we propose a deep learning-based framework to address challenges in multi-agent collaborative perception using an end-to-end approach. Our method captures temporal semantics from ego agents by leveraging an intra-vehicle fusion module to extract valuable contextual cues. Additionally, we introduce a technique to integrate time delays in neighboring features via temporal token encoding, enabling comprehensive information exchange between agents in the inter-vehicle fusion module. Furthermore, we propose an adaptive fusion module that intelligently aggregates contextual information from the ego agent with collaborative features and real-time data. Extensive experiments on multiple benchmark collaborative perception datasets demonstrate the superiority of our proposed method and validate the effectiveness of our proposed components. A limitation of DelAwareCol that will be tackled in future work is that, in the proposed methodology, we assume accurate pose being retrieved for each agent, which could be improved by methods like [40]. Working on this limitation would further enhance adaptability to real Scenarios, such limitation was obvious when testing on V2VReal dataset where data was retrieved from real-life situations.

REFERENCES

[1] M. Yazgan, T. Graf, M. Liu, T. Fleck, and J. M. Zöllner, "A survey on intermediate fusion methods for collaborative perception categorized by real world challenges," in *Proc. IEEE Intell. Veh. Symp.*, 2024, pp. 2226–2233.

[2] M. Yazgan, M. V. Akkanapragada, and J. M. Zöllner, "Collaborative perception datasets in autonomous driving: A survey," in *Proc. IEEE Intell. Veh. Symp.*, 2024, pp. 2269–2276.

[3] Q. Chen, S. Tang, Q. Yang, and S. Fu, "Cooper: Cooperative perception for connected autonomous vehicles based on 3D point clouds," in *Proc. IEEE 39th Int. Conf. Distrib. Comput. Syst.*, 2019, pp. 514–524.

[4] A. N. Ahmed, I. Ravijts, J. de Hoog, A. Anwar, S. Mercelis, and P. Hellinckx, "A joint perception scheme for connected vehicles," in *Proc. 2022 IEEE Sensors*, 2022, pp. 1–4.

[5] A. Rauch, F. Klanner, R. Rasshofer, and K. Dietmayer, "Car2X-based perception in a high-level fusion architecture for cooperative perception systems," in *Proc. IEEE Intell. Veh. Symp.*, 2012, pp. 270–275.

[6] Z. Y. Rawashdeh and Z. Wang, "Collaborative automated driving: A machine learning-based method to enhance the accuracy of shared information," in *Proc. 21st Int. Conf. Intell. Transp. Syst.*, 2018, pp. 3961–3966.

[7] J. Guo et al., "CoFF: Cooperative spatial feature fusion for 3-D object detection on autonomous vehicles," *IEEE Internet Things J.*, vol. 8, no. 14, pp. 11078–11087, Jul. 2021.

[8] Y. Lu et al., "Robust collaborative 3D object detection in presence of pose errors," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2023, pp. 4812–4818.

[9] 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 *Proc. 16th Eur. Conf. Comput. Vis.–ECCV*, 2020, pp. 605–621.

[10] Y. Li, S. Ren, P. Wu, S. Chen, C. Feng, and W. Zhang, "Learning distilled collaboration graph for multi-agent perception," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, vol. 34, pp. 29541–29552.

[11] A. N. Ahmed, S. Mercelis, and A. Anwar, "Graph attention based feature fusion for collaborative perception," in *Proc. 2024 IEEE Intell. Veh. Symp.*, 2024, pp. 2317–2324.

[12] 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 *Proc. Int. Conf. Robot. Automat.*, 2022, pp. 2583–2589.

[13] Y. Hu, S. Fang, Z. Lei, Y. Zhong, and S. Chen, "Where2comm: Communication-efficient collaborative perception via spatial confidence maps," in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, vol. 35, pp. 4874–4886.

[14] K. Lee, J. Kim, Y. Park, H. Wang, and D. Hong, "Latency of cellular-based V2X: Perspectives on TTI-proportional latency and TTI-independent latency," *IEEE Access*, vol. 5, pp. 15800–15809, 2017.

[15] Y. Liu et al., "Select2col: Leveraging spatial-temporal importance of semantic information for efficient collaborative perception," *IEEE Trans. Veh. Technol.*, vol. 73, no. 9, pp. 12556–12569, Sep. 2024.

[16] Z. Lei, S. Ren, Y. Hu, W. Zhang, and S. Chen, "Latency-aware collaborative perception," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 316–332.

[17] R. Xu, H. Xiang, Z. Tu, X. Xia, Ming-H. Yang, and J. Ma, "V2X-ViT: Vehicle-to-Everything cooperative perception with vision transformer," in *Proc. Euro. Conf. Comput. Vis.*, 2022, pp. 107–124.

[18] R. Xu et al., "V2v4real: A real-world large-scale dataset for Vehicle-to-Vehicle cooperative perception," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 13712–13722.

[19] A. N. Ahmed, S. Mercelis, and A. Anwar, "CollabGAT: Collaborative perception using graph attention network," *IEEE Access*, vol. 12, pp. 142380–142393, 2024.

[20] K. Yang et al., "Spatio-temporal domain awareness for multi-agent collaborative perception," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 23383–23392.
[21] K. Yang, D. Yang, J. Zhang, H. Wang, P. Sun, and L. Song, "What2comm: Towards communication-efficient collaborative perception via feature decoupling," in *Proc. 31st ACM Int. Conf. Multimedia*, 2023, pp. 7686–7695.
[22] H. Yu et al., "Vehicle-infrastructure cooperative 3D object detection via feature flow prediction," 2023, *arXiv:2303.10552*.
[23] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "Pointpillars: Fast encoders for object detection from point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 12697–12705.
[24] Y. Lin, I. Koprinska, and M. Rana, "Temporal convolutional attention neural networks for time series forecasting," in *Proc. Int. Joint Conf. Neural Netw.*, 2021, pp. 1–8.
[25] W. Luo, Y. Li, R. Urtasun, and R. Zemel, "Understanding the effective receptive field in deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, 2016, pp. 4898–4906.
[26] A. Vaswani, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.
[27] J. B. Kenney, "Dedicated short-range communications (DSRC) standards in the United States," *Proc. IEEE*, vol. 99, no. 7, pp. 1162–1182, Jul. 2011.
[28] M. Jaderberg et al., "Spatial transformer networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, vol. 28, pp. 2017–2025.
[29] 3rd Generation Partnership Project (3GPP), "Study on channel model for frequencies from 0.5 to 100 GHz," 3GPP, Sophia Antipolis, France, Tech. Rep. TR 38.901 V17.0.0, Dec. 2021, Version 17.0.0.
[30] Z. Hu, Y. Dong, K. Wang, and Y. Sun, "Heterogeneous graph transformer," in *Proc. Web Conf.* 2020, 2020, pp. 2704–2710.
[31] P. Shaw, J. Uszkoreit, and A. Vaswani, "Self-attention with relative position representations," 2018, *arXiv:1803.02155*.
[32] Ahmed N. Ahmed, S. Mercelis, and A. Anwar, "GIFFf: Graph iterative attention based feature fusion for collaborative perception," in *Proc. 20th Int. Joint Conf. Comput. Vis., Imag. Comput. Graph. Theory Appl.*, 2025, pp. 820–829.
[33] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "Carla: An open urban driving simulator," in *Proc. Conf. Robot Learn.*, 2017, pp. 1–16.
[34] P. A. Lopez et al., "Microscopic traffic simulation using Sumo," in *Proc. 21st Int. Conf. Intell. Transp. Syst.*, 2018, pp. 2575–2582.
[35] R. Xu, Y. Guo, X. Han, X. Xia, H. Xiang, and J. Ma, "Opencda: An open cooperative driving automation framework integrated with co-simulation," in *Proc. IEEE Int. Intell. Transp. Syst. Conf.*, 2021, pp. 1155–1162.
[36] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2980–2988.
[37] A. R. Sutanto and D.-K. Kang, "A novel diminish smooth l1 loss model with generative adversarial network," in *Proc. Intell. Hum. Comput. Interaction, 12th Int. Conf.*, 2021, pp. 361–368.
[38] D. P. Kingma, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[39] A. Bazzi, B. M. Masini, A. Zanella, and I. Thibault, "On the performance of IEEE 802.11 p and LTE-V2V for the cooperative awareness of connected vehicles," *IEEE Trans. Veh. Technol.*, vol. 66, no. 11, pp. 10419–10432, Nov. 2017.
[40] N. Vadivelu, M. Ren, J. Tu, J. Wang, and R. Urtasun, "Learning to communicate and correct pose errors," in *Proc. Conf. Robot Learn.*, 2021, pp. 1195–1210.

AHMED N. AHMED received the master's degree in sustainable automotive engineering from the Faculty of Applied Engineering, University of Antwerp, Antwerp, Belgium, in 2020. He is currently working toward the Ph.D. degree with IDLab, a research group of the University of Antwerp and IMEC. His research interests include shared situational awareness for ITS, cooperative perception, and autonomous navigation.

SIEGFRIED MERCELIS received the master's degree in music production and engineering (electronics and ICT), and the Ph.D. degree in applied engineering from the University of Antwerp, Antwerp, Belgium, in 2016. From 2012 to 2016, he was with Van den Berghe Research and Development under a Baekeland Ph.D. mandate on the subject of optimizing and parallelizing real-time media applications. He is currently an Assistant Professor with the University of Antwerp, where he is also an Assistant Professor and a Program Manager with AI Applications Team. His team of more than 30 researchers is committed to bridging the gap between academic AI research and industry in domains, such as chemical process control, autonomous shipping, smart buildings, logistics, and mobility. He was the recipient of the VIK Award for his master's thesis on parallel data structures.

ALI ANWAR (Member, IEEE) received the Ph.D. degree in control science and engineering from the Harbin Institute of Technology, Harbin, China, in 2019. Since 2020, he has been the Principal Research Fellow with IDLab, an IMEC research group with the University of Antwerp, Antwerp, Belgium, where he currently leads a team on context-aware control systems. His research interests include autonomous vessel navigation, safe reinforcement learning, cooperative perception, and generative modeling in computer vision. He is with the IEEE Industrial Electronics and Systems, Man and Cybernetics Society, where he is part of the technical committees on motion control, and control, robotics, and mechatronics. He is a reviewer in journals including IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS,IEEE/ASME TRANSACTIONS ON MECHATRONICS, and IEEE ACCESS.