IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

IEEE VEHICULAR TECHNOLOGY SOCIETY SECTION

Received 6 September 2024, accepted 22 September 2024, date of publication 26 September 2024, date of current version 9 October 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3468432

RESEARCH ARTICLE

CollabGAT: Collaborative Perception Using Graph Attention Network

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

Imec Research Group, IDLab, Faculty of Applied Engineering, University of Antwerp, 2000 Antwerp, Belgium

Corresponding author: Ahmed N. Ahmed (ahmed.ahmed@uantwerpen.be)

This work was supported by the Research Foundation Flanders (FWO) under Grant 1S90022N.

ABSTRACT Collaborative perception exploits exchanging perception data across multiple agents to enhance situational awareness. To overcome the constraints created by limited network resources, it is necessary to have a reliable collaborative perception that can sustain the trade-off between performance and bandwidth. This research proposes a method to address this issue through the development of a graph attention network (GAT) for intermediate collaborative perception. The proposed approach aims to enhance object detection accuracy by exchanging information among multiple agents, addressing issues such as sensor limitations, and occluded objects, and expanding the sensing range. Additionally, the proposed approach addresses the challenges caused by limited communication bandwidth and inconsistencies in the data shared by different agents by maintaining a balance between performance and bandwidth, fitting real-life application requirements. Our approach aggregates the intermediate features obtained from multiple neighboring agents and introduces a novel attention mechanism to selectively emphasize significant regions within the intermediate feature maps. This attention mechanism operates on both the channel and spatial levels to direct the data aggregation process. This research presents a new method for aggregating features utilizing various attention architectures. We perform quantitative and qualitative assessments of the final results of this method and compare them to other state-of-the-art collaborative perception techniques. The results of this work are evaluated using the V2XSim and OPV2V datasets, and our quantitative and qualitative experiments in multi-agent object detection show that our approach achieves a better performance than the state-of-the-art collaborative perception methods.

INDEX TERMS Collaborative perception, intermediate fusion, graph neural network, attention.

I. INTRODUCTION

Single-vehicle perception has been studied extensively and made remarkable achievements in recent years, e.g., object detection [1], [2], segmentation [3], [4], and tracking [2], [5], [6]. However, despite its great progress, single-vehicle perception often suffers from several shortcomings stemming from its individual perspective. For instance, the perception system can hardly perceive objects that are occluded or are far away due to the limited field of view of perception sensors, making accurate recognition challenging, which leads to uncertain interpolation and poor situational awareness [7]. To address these problems, several compelling research has been focused on collaborative perception [8], [9], [10], [11], [12], which take advantage of sharing the multiple-viewpoints of the same scene with the Vehicle-to-Vehicle (V2V) and Vehicle-to-Everything (V2X) communication. Collaborative perception is a technique that enables the exchange and fusion of visual sensory information among multiple agents (agents: vehicles and infrastructure) to enhance situational awareness for single-agent perception and to overcome the limitations imposed by single-perspective view constraints. Collaborative perception can be divided into three categories, early, intermediate,

The associate editor coordinating the review of this manuscript and approving it for publication was Cheng Huang.

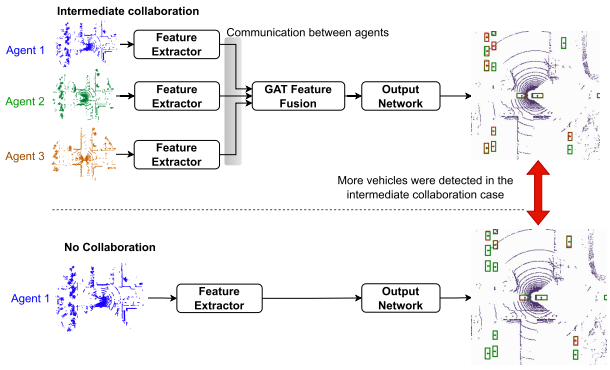


FIGURE 1. An illustration comparing the overall architecture of intermediate collaborative and single vehicle perception. In the intermediate collaborative perception feature maps across the collaborators are fused into a single representation. The resulting output is shown on the right of the diagram, where the green and red boxes denote ground truth and predictions, respectively. It is apparent that the intermediate collaborative perception can perceive more objects than the single-vehicle perception.

and late collaboration. Early collaboration involves aligning and aggregating raw visual data received from neighboring agents to create a holistic perspective, as proposed in [13] and [14]. This approach effectively addresses single-agent perception challenges, but it requires a significant amount of communication bandwidth to broadcast raw sensor data. On the other hand, late collaboration, however, is a more bandwidth-efficient approach, as it fuses the perception outputs obtained from the ego-agent and neighboring agents, as proposed in [15]. Nevertheless, this approach may result in unsatisfactory fusion results due to the presence of noise and incompleteness in each agent's perception output.

To address the trade-off between performance and bandwidth, as well as the relative spatial alignment of output predictions, intermediate collaboration achieves this balance by aggregating the intermediate feature maps received from neighboring agents, as proposed in [10], [11], [16], and [12]. This approach encodes the ego and the received essential perception features into representative compact feature maps, which can be transmitted efficiently, thus enhancing perception ability. An illustration of the intermediate collaborative perception is shown in Fig. 1. But despite that, a poorly designed intermediate collaboration strategy may lead to information loss during feature alignment and fusion, resulting in limited improvement in perception ability. Intermediate collaborative perception poses several challenges that need to be addressed to achieve its full potential in enhancing situational awareness. This work tackles the aggregation methodology of the intermediate features broadcasted by multiple neighboring agents. This is particularly challenging when agents have different poses; requiring an effective fusion strategy that can highlight important and relevant regions between the ego and the received feature maps.

In this work, we address multi-agent collaborative perception using graph neural networks (GNNs) to fuse the ego's and neighboring agent's feature maps. Recent advancements in GNNs have demonstrated significant success in processing graph-structured data [17], [18]. GNNs are particularly effective due to their capability to capture complex relationships and dependencies inherent in graph-structured data, facilitating the learning process from both node and edge features. This capability makes GNNs highly suitable for multi-agent systems, as they enable efficient message propagation and aggregation across nodes to update node features. GraphSAGE [19] learns a function to derive node embeddings by sampling and aggregating information from neighboring nodes. Similarly, GAT [20] introduces a multi-head attention mechanism that dynamically adjusts the weights of neighboring nodes during the aggregation process. Furthermore, GNNs have also been applied effectively in self-driving applications. For example, [21] and [22] propose a spatially-aware GNN and an interaction transformer, respectively, to model interactions between actors in autonomous driving scenarios. Additionally, [23] employs GNNs to estimate value functions for map nodes and to facilitate coordinated route planning by sharing vehicle information.

In this work, we adopt the GAT and design a network capable of capturing the intricate relationships between collaborating agents, where each agent is represented as a node in the graph. The adoption of graphs in our approach is motivated by their capability to efficiently leverage node connectivity for information propagation. This feature is particularly essential in collaborative perception when aggregating intermediate representations from multiple perspectives (received from multiple neighboring agents). In contrast, alternative methods often fail to handle this propagation as seamlessly. Additionally, graphs are well-suited for dynamic environments, where the number of agents and their relationships vary over time. Many traditional approaches face significant challenges in scaling efficiently under such conditions. In our proposed GAT, we incorporate both channel and spatial attention mechanisms, which allow the network to focus on significant features, thereby enabling more intelligent aggregation of intermediate features. The significance of the attention mechanism on feature fusion has been studied extensively in literatures [24] and [25]. Attention mechanisms allow for selective focus on relevant features in the feature map, leading to improved capture of the visual structure. Additionally, attention mechanisms enhance the representation power by assigning higher attention coefficients to essential features and suppressing irrelevant ones. To further improve the ability to extract valuable features, attention-based convolution has been proposed as a method of combining the characteristics of attention mechanisms with convolution operations [26]. This is achieved by applying channel and spatial attention modules, which allow for the learning of ''what'' and ''where'' to attend to in the feature map. As a result, attention-based convolution allows for efficient information flow within the network and the ability to emphasize relevant information and suppress irrelevant ones.

In this work, we propose a collaborative perception framework based on a novel design of graph-attention-based network that fuses intermediate feature fusion using channel and spatial attention mechanisms to weigh the importance of different features from multiple sources before combining them. This helps to ensure our proposed collaborative perception framework focuses on (gives higher attention to) the most relevant regions of the received feature maps which, consequently, reduces the impact of irrelevant or noisy regions of the feature maps. Furthermore, our proposed attention-based feature fusion is designed to be lightweight, and selectively combine features from different agents based on their relative importance. Our results show that this can improve the model's average precision on object detection tasks while maintaining the number of parameter counts. However, it requires careful design to ensure that the attention mechanism is correctly identifying the most informative features and that the feature fusion strategy is effectively combining them. Our proposed unified, graph attention-based collaborative perception framework achieves competitive performance with state-of-the-art intermediate collaboration methods. To summarize, our contributions are as follows:

- We propose a pose-aware graph attention network to model the multi-agent collaborative perception which is adaptive to real-time measurements.
- We propose a novel lightweight channel-spatial attention mechanism for aggregating the intermediate features enhancing the representation power of the resulting feature maps after fusion.
- We conduct experiments on V2XSim [27] to validate our methodology, conduct performance-bandwidth analysis, and examine the different architectures of our proposed channel-spatial attention using the average precision metric on object detection task.

The remainder of this paper is structured as follows: Section. II provides a comprehensive summary and analysis of collaborative perception fusion methods. In section. III we introduce the proposed methodology, elucidating its fundamental components and our feature fusion strategy. The dataset, experimental setup, and evaluation metric are expounded upon in Section IV. Subsequently, Section V presents a quantitative and qualitative analysis of the achieved outcomes, drawing comparisons between our proposed methods and the baseline methods. Finally, in section. VI we present the deductions and implications of this work.

The reason behind adopting GNNs in our method is due to its highly effective ability to leverage local and global node connectivity to propagate information. This is crucial in collaborative perception, where information from multiple perspectives needs to be aggregated efficiently. Other methods might not handle this propagation as naturally. Moreover, GNNs excel in scenarios where the number of agents or their relationships change over time. Many alternative methods struggle to scale efficiently in such dynamic environments.

II. RELATED WORK

Collaborative perception methodologies are divided into three categories: early, intermediate, and late. Early collaboration perception [13], [28] involves sharing raw data among vehicles and infrastructure. Then aggregating those raw data creates a holistic perspective of the surrounding environment. Despite this approach's capability to address occlusion and long-range issues that occur in single-agent perception, the bandwidth required to transmit raw data is limited in actual scenarios due to the needed network bandwidth capacity.

Late collaboration [29] fuses the output predictions received from every agent, which is typically implemented after each agent has processed its observations. Although late cooperation offers benefits in terms of transmission bandwidth, it is particularly susceptible to positioning errors, which can lead to significant estimation noise resulting from insufficient local observations. Lately, there has been extensive study on an emerging solution i.e. intermediate collaborative perception. This technique includes broadcasting a condensed intermediate representation and has been shown to achieve a more favorable balance between perception precision and network bandwidth. Nevertheless, the intermediate cooperative perception technique encounters two significant obstacles: i) How to determine the most advantageous and concise features from the initial measurements for transmission; and ii) How to effectively aggregate the features of other agents to improve each agent's situational awareness.

Intermediate collaboration involves the fusion of intermediate features generated by each agent's encoder model. F-Cooper [16] introduced a feature-level fusion approach that uses the highest value in overlapping areas to represent intermediate features. Another method, V2VNet [10], utilized GNN and proposed multi-round message passing to achieve better perception and prediction performance; however, this method did not account for cross-channel feature significance when fusing the feature maps. DiscoNet [12] exploited a method incorporating a teacher-student model for both early and intermediate collaboration, enabling the knowledge of early collaboration to guide the training of the intermediate fusion model. HP3D-V2V [30] proposed an adaptive feature fusion method that only considers spatial relationships. Though this work achieves promising results, both DiscoNet and HP3D-V2V do not consider the channel-wise significance within the fused feature map. CORE [31] is an intermediate collaboration approach strongly correlated with DiscoNet. However, CORE tackles feature fusion by employing a learning-to-reconstruction approach instead of using teacher-student knowledge distillation. V2X-ViT [32] tackled the feature fusion task using vision transformers, which are known to have a large number of parameters,

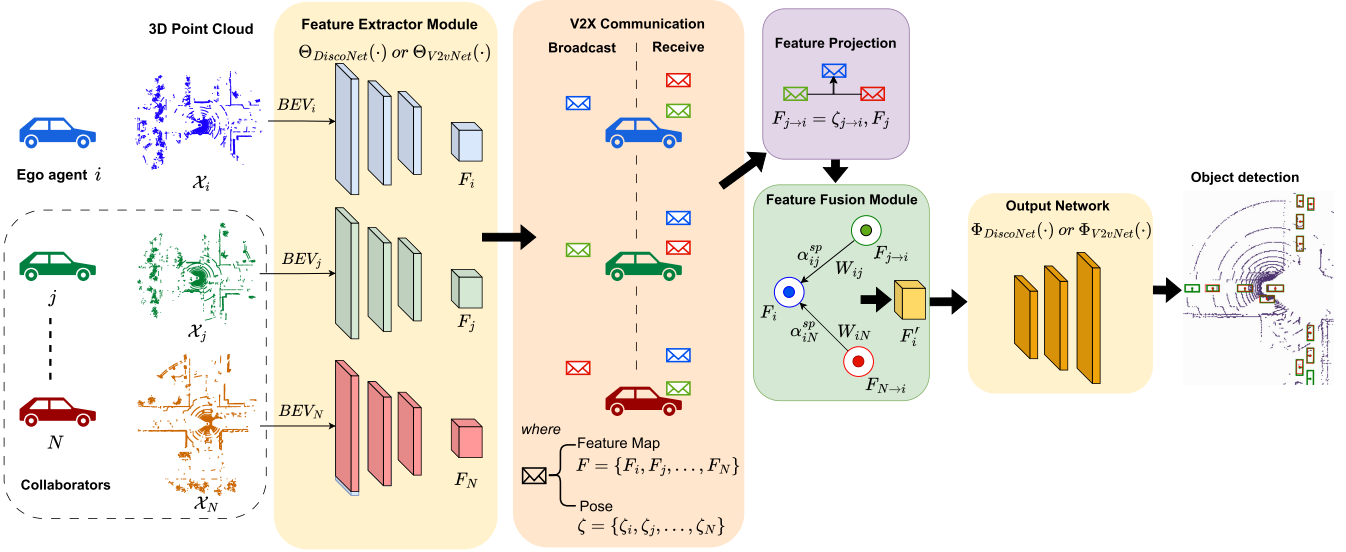


FIGURE 2. The proposed method's overall architecture begins with each agent $\mathcal{X}$ converting its point cloud into a BEV map. The shared feature extractor $\Theta(\cdot)$ processes the BEV map to obtain the feature map F. The received feature maps are then transformed into the ego agent's coordinate system before being aggregated with the ego feature map, utilizing channel and spatial attention mechanisms to produce an updated representation F_i'. The updated representation is then passed to the output network $\Phi(\cdot)$ for object detection.

complicating real-time deployment. The authors in [33] developed a collaborative technique using transformers, which allows for dynamic semantic interaction based on positional correlation. Where2Comm [34] fused camera images with LiDAR point clouds bird's-eye-view (BEV) images received from multiple agents. For feature fusion, Where2Comm employed a confidence-aware pattern to guide agents in focusing on sharing spatially critical information. The confidence feature map is updated iteratively at every communication round, which might be slow and only reliable in use cases like platooning, where the vehicle communicates with other vehicles in the platoon for an extended period. The authors in [35] adopted GAT to fuse features feature relationships between neighboring drones, however, the proposed methodology relied only on spatial attention. Furthermore, the authors in [35] implemented two 2D convolutions to reduce the dimension of the feature map to 1/8 of the original size, which can result in the loss of spatial context, making it harder for the model to understand the structure and layout.

Motivated by the flaws of the current collaborative perception approaches, we incorporate channel and spatial attention within our graph network, to attend only to significant features, facilitating the graph network to aggregate intermediate features intelligently, this yields improved feature selection and provides a more versatile approach. Additionally, it offers a dual perspective of attention, allowing for enhanced interpretation of vital spatial regions and feature-types.

III. METHODOLOGY

A. PROBLEM FORMULATION

We assume a system of N agents equipped with an onboard LiDAR sensor and simultaneously perceiving the environment to obtain its local measurement in the form of 3D point clouds X_N, then convert it to BEV. Following mathematically, for the i-th agent, the proposed intermediate collaborative perception works as:

$$F_i = f_{\text{encoder}}(BEV_i) \tag{1a}$$
$$F_{j\to i} = f_{\text{transform}}(\zeta_{j\to i}, F_j) \tag{1b}$$
$$F_i' = f_{\text{fusion}}(F_i, \text{AGG}\{F_j \to i\}_{j=1,2,\ldots,N}) \tag{1c}$$
$$B_i = f_{\text{decoder}}(F_i') \tag{1d}$$

where F_i is the feature extracted from the i-th agent's observation, $\zeta_{j\to i} = (x_i, y_i, z_i, \theta_i, \phi_i, \gamma_i)$ is the 6DoF pose of the i-th agent with $\theta_i, \phi_i, \gamma_i$ the yaw, pitch and roll angle, F_j is the feature map that were transmitted from the j-th agent to the i-th agent. $F_{j\to i}$ is the transformed feature map on the j-th after it's coordinate space got aligned with F_i through pose transformation. Subsequently, F_i' is the aggregated feature of the i-th agent after fusing other agents' messages. B_i is the output of the object detection task.

The mapping function f_{fusion} and aggregation function AGG in shown Eq.1c are the most important stages that differentiate one graph neural network from another. Prior to the introduction of graph attention mechanisms, classical GNNs often treated all nodes equally and, the aggregation function typically often employed average or max pooling. This frequently overlooks the variations in the influence of adjacent nodes' characteristics on the central node during the process of sampling and aggregation. GAT resolved this problem by including attention mechanisms, which learn to allocate weights to adjacent nodes based on their significance to the central node. GAT introduces the concept of attention mechanisms, allowing the aggregation function to adaptively match different neighboring nodes with corresponding weights. GAT learns attention weights

to measure the significance of neighbor j to node i. This method allocates distinct attention weights to surrounding node features depending on their impact on the central node.

B. OVERVIEW OF PROPOSED METHOD

The overview of our proposed methodology is illustrated in Fig. 2. This work introduces a novel channel and spatial-based graph attention network to aggregate the feature maps received from different cooperative agents perceiving the environment from various viewpoints. Our proposed intermediate fusion methodology comprises three main modules. The first module is the "Feature Extraction Module" (section III-C). To implement feature extraction, we convert a 3D point cloud to a BEV image, then extract relevant features from the resulting BEV, creating a feature representation encoding essential features as shown in Fig. 2. The resulting feature map is then utilized by the second module, referred to as the "Feature Fusion Module" (section III-E). This module performs cross-agent feature fusion by aggregating information from multiple agents, including the ego vehicle's intermediate representation, generating an updated feature representation while considering their relative spatial locations and relevancy to each other. Finally, the "Output Network Module" (section III-F) decodes the updated feature map to perform object detection predictions.

C. FEATURE EXTRACTION MODULE

This work employs feature representations in the BEV, where all agents project their own perceptual information onto a single global coordinate system. This approach eliminates the requirement for complex coordinate transformations, promoting improved collaboration among different agents. The BEV representation creates a shared spatial coordinate system, bringing together spatial input from different agents and modalities while effectively retaining both geometric and semantic information. BEV-based approaches are now a rapidly growing field of study, demonstrating exceptional performance in LiDAR-based 3D detection. In addition, state of the art collaborative perception frameworks like V2VNet [10], DiscoNet [12], CORE [31], and Where2Comm [34] all adopt the BEV representation. This work utilizes a constant geometric transformation to transform the characteristics of the 2D pseudo-image to a BEV, aligning with the reliable CaDDN [36] approach.

The purpose of the feature extraction module is to derive informative features from 3D point cloud for each agent separately. Assuming there are N agents perceiving the environment in a scene, with $\mathcal{X}_N$ corresponding to the point clouds raw observation of the Nth agent. The point clouds $\mathcal{X}_N$ are first converted into voxels and transformed into a pseudo-image that represents a 3D voxel lattice. This pseudo-image gives a compact top-down BEV of the surroundings. This serves as a helpful representation of the objects in nearby areas and offers an overall context through a compact top-down 2D map. [37]. Given that the 3D voxel grid can be regarded as a pseudo-image with the height dimension representing the channel dimension, we use an efficient 2D convolution on the BEV instead of the computationally intensive 3D convolution.

With this BEV, we apply feature extraction blocks composed of 2D convolutions, batch normalization, and ReLU activation transforming the BEV into intermediary state space, obtaining the feature map $F_i \leftarrow \Theta(BEV_i)$, where $\Theta(\cdot)$ is the feature encoder shared by all the agents, and the resulting feature map $F_i \in \mathbb{R}^{W \times H \times C}$ with $W \times H$ representing that spatial resolution and C the number of channels. The feature extraction process results in an optimal representation of the input data with respect to the task objective of object detection.

It is crucial to underscore that our primary focus is to enhance the feature map aggregation and collaboration strategy, aiming to optimize the trade-off between communication cost and perception performance.

In our feature extraction, we conduct two experiments one time utilizing the same feature extractor as DiscoNet [12] ($\Theta_{DiscoNet}(\cdot)$), and in the other experiment we use the same feature extractor as V2VNet [10] ($\Theta_{V2VNet}(\cdot)$). This allows us to independently analyze and compare the performance of our proposed novel feature fusion strategy regardless of the feature extractor used. It is noteworthy to mention that both DiscoNet and V2VNet feature extractors are designed to utilize the BEV and leverage CNNs for feature extraction to generate the feature map.

D. FEATURE SHARING AND PROJECTION MODULE

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 70m 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 [38], this metric has also been adopted in [10]. We assume ideal communication, in which each agent successfully receives messages from all its relevant neighbors at each timestep.

Since each agent possesses its unique pose, the feature map from the neighboring agent F_j needs to be transformed into the perspective system of the ego agent i using the transformation matrix $\zeta_{j \to i}$. This is determined based on ζ_i and ζ_j. 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 [39], 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 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 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, sampling it at the grid-specified positions using bi-linear interpolation to manage non-integer positions. This results in $F_{j\to i}$ i.e. the transformed neighboring feature map to the ego agent's perspective. The ego agent repeats this affine transformation process for all received feature maps, and once all feature maps are transformed, the ego and the transformed feature maps are passed to the feature fusion module.

E. ATTENTIVE FEATURE FUSION MODULE

Upon the completion of the feature extraction and projection modules, the feature maps obtained contain only the most relevant spatial information for each agent's perception data, which are projected onto the receiving vehicle's pose. To achieve reliable shared situational awareness, it is essential to design a reliable methodology to integrate and fuse these feature maps. This work designs a channel-spatial attention-based GAT to enhance message-passing aggregation of the feature maps into a single updated feature representation (as demonstrated in Fig. 2).

1) GRAPH ATTENTION STRUCTURE

The primary objective of the proposed graph attention collaboration is to update the feature map through message passing among the connected nodes. The graph G is composed of (V, E), where node $V = V_N, N \in i, j, \ldots, N$ is the set of nodes holding the feature map of each agent $F = F_i, F_j, \ldots, F_N$, and $E = W_{N\to i}$ is the set of trainable edge weight matrices between two connected nodes, modeling the collaboration strength between the feature maps of those nodes. This represents the collaboration importance between the nodes. The edge weights for both directions are distinct and are dynamically adjusted based on the detection task. In the proposed method, each agent maintains its own local graph, and edges are only established with relevant agents (relevant agents are determined based on the metric discussed in Section. III-D). Additionally, we introduce channel and spatial attention to compute the attention coefficients to capture cross-dimensional interaction information, including direction-aware channel and channel-sensitive spatial information, which can help to improve the aggregation of the feature maps. We adopt a pose-aware, fully connected bidirectional architecture, learning the edge weight $W_{N\to i}$, channel attention α^{ch}, and spatial attention α^{sp}. Each node maintains a state representation, the attention map is computed between neighboring nodes, and then node states are updated based on their attention and edge weight matrices. The final output of the GAT is a new ''updated'' state representation (node features) of the ego agent's node F_i', as shown in Fig. 2. The overall feature fusion methodology can be presented in two stages: (1) message attention, where the attention coefficients are computed, and (2) message aggregation.

2) MESSAGE ATTENTION

The squeeze-and-excitation networks (SENet) [40], which learn channel attention, brought clear performance gains for various deep CNN architectures. Although this method has achieved higher accuracy, it often brings higher model complexity and suffers from a heavier computational burden. Revisiting the channel attention module in SENet, it takes input data and applies global average pooling to each channel individually. Then, two fully connected (FC) layers with non-linearity are utilized, followed by a sigmoid function, to build channel weights. The purpose of the two FC layers is to capture non-linear cross-channel interaction and reduce dimensionality in order to regulate the complexity of the model. While this approach is often employed in future channel attention modules [24], empirical research done by [41] demonstrates that dimensionality reduction has negative consequences on channel attention prediction. Furthermore, it is both wasteful and unnecessary to record dependencies across all channels. In our proposed method, we overcome this problem by implementing a channel 1D point-wise convolution of size 1. From the empirical analysis conducted in the literature [24] and our analytical results (presented in Section V), we see that our channel attention module can learn effective channel relationships, avoiding linear channel dimensionality reduction while capturing cross-channel interactions while still maintaining a lightweight model. We also reinforce our attention map by learning spatial relationships (using spatial attention), which guidea the model on the most important regions in the feature map. Our message attention learning stage is composed of a sequential arrangement of channel and spatial attention to guide the model to the regions to give higher attention to; the proposed attention computation schematic is illustrated in Fig. 3.

3) CHANNEL ATTENTION

Since each channel represents different aspects or features of the feature map (like edges, textures, etc.), the goal of channel attention is to specify the relationship between various channels by adaptively determining each channel's significance during the network training process. The feature maps of the ego agent F_i and the received $F_{j\to i}$ are first aggregated, creating F_{ij} as shown below:

$$F_{ij} = F_i \oplus F_{j\to i} \tag{2}$$

where $\oplus$ (will be elaborated in table 1) is a permutation invariant aggregation operator that accepts an arbitrary number of inputs (e.g., element-wise sum, concatenation, etc.). Meaning that we get the same result regardless of the order of inputs.

The $H \times W$ height and width of the feature map are referred to as the ''spatial dimensions'' as they define

the two-dimensional spatial extent of the feature map, corresponding to the image size in pixels. In our proposed method, the channel attention mechanism compresses the spatial dimensions $H \times W$ into a single pixel, i.e.,$1 \times 1 \times C$, where C represents the number of channels. This compression enables the assessment of the significance of each channel individually by assigning distinct scores to each channel. This incites salient features and suppresses insignificant ones guiding the model on which channels to focus on [24]. To achieve this, spatial the dimension of the aggregated feature map F_{ij} is squeezed using global average pooling yielding in $GAP(F_{ij}) \in \mathbb{R}^{1\times1\times C}$. GAP is selected for this operation as it has shown to be effective in highlighting informative channel regions [42]. Afterward, encoder-decoder pointwise convolution blocks (PwC), and non-linear activation funtion are applied to extract different channel information. The channel attention α_{ij}^{ch} is represented as follows:

$$\alpha_{ij}^{ch} = \sigma(L_{ch}(\delta(G_{ch}(GAP(F_{ij}))))) \tag{3}$$

where G_{ch} is the PwC-based encoder representing the dimensionality reduction layer, and L_{ch} is the PwC-based decoding representing the dimension-increasing layer, and δ is the ReLU. The PwC encodes channel information at each pixel over all spatial locations using a 1×1 kernel PwC that acts as the local channel context aggregator, which only exploits point-wise channel interactions for each spatial obtaining the channel attention map α_{ij}^{ch}. The final attention map is normalized by the sigmoid function σ. Next, for the channel attention α_{ch} to take effect, it is applied to the aggregated features F_{ij} as follows:

$$F_{ij}^{ch} = \alpha_{ij}^{ch} \otimes F_{ij} \tag{4}$$

where $\otimes$ is the dot multiplication operation, and F_{ij}^{ch} is the updated aggregated feature map.

4) SPATIAL ATTENTION

Spatial attention coefficients are learnable parameters utilized to adaptively select specific regions within an input, directing the model's focus on specific regions or locations within the feature map. This enables the model to selectively process key information guiding the model to learn "where to pay attention" [24].

Our strategy for fusing feature maps in V2X systems involves the use of both channel and spatial attention in a sequential manner to emphasize the significance of capturing inter-channel and cross-dimensional relationships during the computation of attention weights, to generate robust feature representations. The authors in [26] explored the impact of the arrangement of the two attention modules (spatial and channel) on the final performance of the model. The authors concluded that a sequential arrangement of the two modules is superior to a parallel arrangement. Moreover, they discovered that the optimal order of the sequential arrangement is channel-first, as this configuration yielded slightly better results compared to a spatial-first arrangement. Based on these findings, the attention module in this work was designed to compute channel attention first, followed by spatial attention as shown in Fig. 3.

Consequently, on the resulting feature map from F_{ij}^{ch} (from Eq. 4) we apply point-wise convolution, ReLU, and batch normalization to learn the optimal spatial attention coefficients α_{sp} which encodes where to emphasize or suppress. The spatial attention is computed as:

$$\alpha_{ij}^{sp} = \sigma(L_{sp}(\delta(G_{sp}(F_{ij}^{ch})))) \tag{5}$$

where α_{ij}^{sp} represents the function of spatial attention calculation, G_{sp} is the PwConv-based encoder representing the dimensionality reduction layer, and L_{sp} is the PwConv-based decoding representing the dimension-increasing layer. After computing the spatial attention coefficients α_{ij}^{sp}, it is used to compute and update the features corresponding to them F_{ij}^{sp} as follows:

$$F_{ij}^{sp} = \alpha_{ij}^{sp} \otimes F_{ij}^{ch} \tag{6}$$

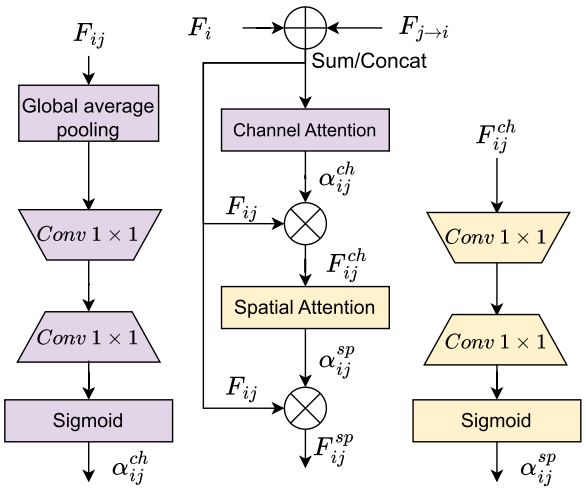


FIGURE 3. Illustration of the attention-based feature fusion module architecture. With channel attention sub-module indicated in purple, and the spatial attention sub-module indicated in yellow.

5) FINAL MESSAGE AGGREGATION

Afterward, all the aggregated features are summed to compute the final output features F_i' for every ego node as follows:

$$F_i' = \sigma\left(\sum_{j\in N} F_{ij}^{sp}\right) \tag{7}$$

Using both attention mechanisms sequentially offers significant advantages in the learning process. The channel attention mechanism plays a crucial role by guiding the model to identify essential channels by assigning varying weights to each channel while considering their spatial relationships. Once this initial feature selection occurs, the spatial attention

starts refining the focus on specific spatial locations within the chosen channels. Consequently, this sequential two-step process enables the model to concentrate on both the most relevant channels and the most informative spatial regions within those significant channels. Additionally, the sequential arrangement of the attention mechanisms leads to a reduction in computational complexity, making the approach more efficient compared to applying them simultaneously. This efficiency is particularly beneficial in resource-constrained applications where computational resources are limited.

Furthermore, our method of channel and spatial attention in the feature map fusion strategy utilizes PwC. This choice is motivated by the computational efficiency of point-wise convolution, as it requires a minimal number of parameters. This is of particular significance when considering the computation of attention weights for each channel or spatial location within the feature map, as this can be computationally demanding for feature maps of large dimensionality.

F. OUTPUT NETWORK MODULE

The objective of this study is to improve the feature fusion methodology and evaluate its performance in comparison to state-of-the-art techniques. Our proposed method employs the same output network as DiscoNet and V2VNet, with regard to the feature extraction component. Following collaboration and aggregation of feature maps, the agent forwards the updated feature map to the output network to perform object detection. The output network involves applying $\Phi(F_i')$, where $\Phi(\cdot)$ consists of convolution layers that are used to categorize the foreground-background categories and predict the bounding boxes. Based on the reasoning discussed in section III-C we employ identical output network employed by DiscoNet $\Phi_{DiscoNet}(\cdot)$ [12] and V2VNet $\Phi_{V2VNet}(\cdot)$ [10] to provide the final detection results.

IV. IMPLEMENTATION DETAILS

A. DATASET

We use two popular cooperative driving datasets for evaluation, namely, V2XSim [27] and OPV2V [43]. **V2XSim** is a V2X dataset, constructed by integrating SUMO [44] and Carla [45]. V2XSim utilizes SUMO to generate realistic traffic flow data, while Carla is used to collect sensory information such as LiDAR from multiple agents within the same geographical region. The dataset contains 10,000 frames of LiDAR point clouds, we follow the split of [27], partitioning the dataset into training, validation, and testing sets with ratios of 8,000, 1,000, and 1,000 frames respectively. This dataset provides a wide range of scenarios with different levels of complexity, which helps in the development and evaluation of perception models that utilize vehicle-to-everything communication.

OPV2V OPV2V is a large-scale V2V perception dataset it was generated by utilizing CARLA [45] and the cooperative driving automation tool OpenCDA [46]. The dataset consists of 11,464 frames, which includes LiDAR point clouds and RGB images. OPV2V is divided into two subsets: the default CARLA towns and the Culver City digital town. The default CARLA towns subset has a total of 10,914 frames. These frames are divided into train/val/test splits of 6,764/1,980/2,170 frames, respectively. This subset offers a broad spectrum of scenarios characterized by varying levels of complexity, thereby facilitating the training and evaluation of collaborative perception models. In contrast, the Culver City subset consists 550 frames and is especially designed to evaluate the ability of the model to generalize. This subset accurately simulates a real-world urban environment, with a wide range of objects and structures that test the perceptive abilities of the models.

B. EVALUATION

To supervise foreground-background classification, we used the binary cross-entropy loss L_{cls}, as follows:

$$L_{cls}(y, \hat{y}) = y \cdot \log(\hat{y}) + (1 - y) \cdot \log(1 - \hat{y}) \tag{8}$$

where y is the true binary label, which is either 0 or 1, and $\hat{y}$ is the predicted probability that the input belongs to class 1 (the positive class). The binary cross-entropy loss penalizes the model more when its predicted probability is far from the true label. When the true label y is 1, the loss becomes $-log(\hat{y})$, and when the true label is 0, the loss becomes $-log(1 - \hat{y})$. This ensures that the model is encouraged to predict high probabilities for the true positive class and low probabilities for the true negative class.

For the bounding-box regression loss L_{reg}, we use a weighted smooth L_1 loss overall regression targets utilized as shown in Eq.9.

$$L_{reg}(x_n, y_n) = \sum_{n \in \{x,y,w,h\}} smooth_{L_1}(x_n - y_n) \tag{9}$$

in which:

$$smooth_{L_1} = \begin{cases} 0.5 \cdot (x_n - y_n)^2, & \text{if } |x_n - y_n| < 1 \\ |x_n - y_n| - 0.5, & \text{otherwise} \end{cases} \tag{10}$$

where x_n and y_n are predicted and target values, respectively. $smooth_{L_1}$ helps prevent large gradients and makes the loss function more robust to outliers compared to the standard L_2 loss (mean squared error) or L_1 loss (mean absolute error), therefore, it is often preferred in tasks where there may be noisy or imprecise annotations. The total loss can be represented as follows:

$$L_{total} = \beta_{cls} L_{cls} + \beta_{reg} L_{reg} \tag{11}$$

where β_{cls}, β_{reg} denotes the weight contribution of the classification, and regression losses, respectively.

Our approach's results are compared to the baseline methods, and we use Average Precision, a widely adopted metric, to evaluate the performance of the presented object detection methods [47]. For our evaluations, we consider Intersection over Union (IoU) thresholds of 0.5 and 0.7 to

calculate average precision scores. The primary focus of our detection efforts is on the car category, and the reported results are based on the test set. The Intersection over Union (IoU) value represents the extent of overlap between the predicted result and the ground truth.

1) COMMUNICATION VOLUME

Our communication volume is the same as DiscoNet [12], with the sole disparity arising from the logarithmic base selection. In our communication volume calculation, the logarithmic base is 2, whereas DiscoNet employs a base 10. Consequently, our metric is approximately 3.32 times greater than theirs. The rationale behind adopting a base of 2 is rooted in the conformity with the bit/byte metric, wherein communication volume denotes message size in a logarithmic scale with a base of 2. The logarithmic base of 10 is used as it corresponds more closely to the natural understanding of numerical magnitudes, making it easier to interpret results. Mathematically, for the designated sparse feature map F_j, the expression can be formulated as follows:

$$\log_2(|F_j| \times C \times 32/8) \tag{12}$$

where $|.|$ signifies the application of the $L0$ norm, which enumerates the non-zero elements within the binary selection matrix. This quantification corresponds to the total spatial grids necessitated for transmission. Furthermore, for each feature point, the symbol C denotes the channel dimension. The factor of 32 is introduced due to the utilization of the float32 data type for representing each numerical value, while division by 8 is undertaken to account for the metric byte employed in the measurement.

C. TRAINING SETUP

During the training phase, one agent is chosen at random as the ego vehicle. However, during the testing phase, we assess a consistent ego vehicle for all models being compared. The effective communication range of each agent is defined as 70m as [38]. Any agent located outside of this radius from ego vehicle is disregarded. We initiate training using the Adam optimizer [48], with an initial learning rate of 10^{-3}. We then gradually decrease the learning rate every 10 epochs by a factor of 0.1 for a total of 100 epochs. All models are trained on NVIDIA Tesla V100 GPU with a batch size of 4.

V. RESULTS

A. BASELINES

To compare our proposed framework, we have considered a diverse range of baseline methodologies. Firstly, we examine the scenario of no collaboration, where only the LiDAR point clouds of the ego-vehicle are utilized for perception, i.e., single-vehicle perception. Secondly, we consider late collaboration, where the detected outputs from all agents are fused to generate the final results. Thirdly, we evaluate early collaboration, which involves the direct aggregation of raw LiDAR point clouds from nearby agents. Finally, for each dataset, we assess our proposed methodologies against state-of-the-art approaches for the intermediate fusion strategy.

B. QUANTITATIVE RESULTS

Table 1 presents a comprehensive overview of the architectural details, average precision (AP) scores at IoU thresholds of 0.5 and 0.7, and the model sizes of the proposed approaches in this work. As discussed in Section III-E, we employ a convolutional encoder-decoder to calculate the attention map, which enables the model to identify significant regions during the feature maps fusion process. The ''Aggregation Operation'', described in Equation 2 (denoted as $\oplus$), represents two aggregation operations: the concatenation (Cat) and summation (Sum) of the feature maps. These operations are used to investigate their respective impacts on the fusion methodology. The ''Depths'' column presents two depths of the convolutional encoding-decoding dimensions utilized in our experiments, specifically (256, 128, 64) and (256, 128, 64, 32). These variations allow us to examine the influence of a deeper encoder-decoder network on the learned attention weights, overall model performance, and the resulting model size. The ''Experiment Name'' column serves as a unique identifier for each experiment; for instance, Exp_D with $\_Sum64$ indicates the usage of the DiscoNet feature extractor ($\Theta_{DiscoNet}(\cdot)$), as well as the output network ($\Phi_{DiscoNet}(\cdot)$) (as discussed in section III-C) with summation operation and convolution encoding-decoding depth (for learning the attention map) of 64 (i.e. 256, 128, 64).

1) COMPARISON OF THE PROPOSED METHODS WITH BASELINES V2XSIM

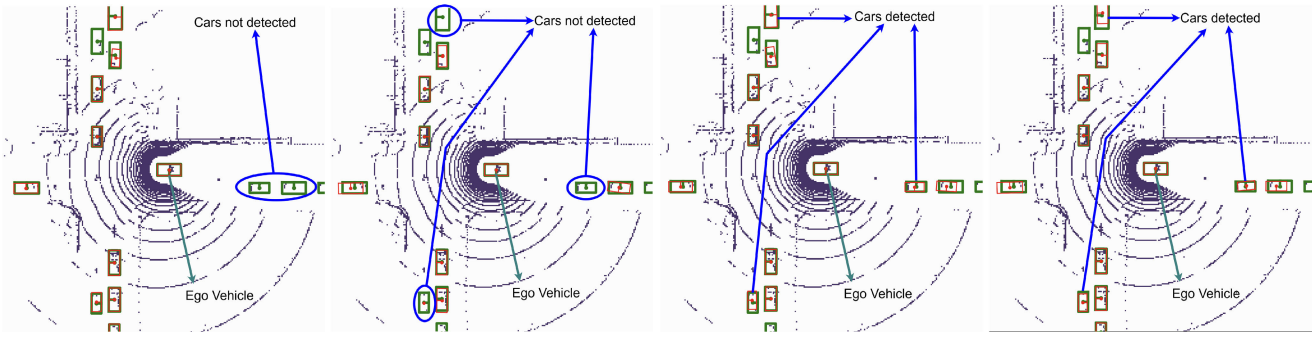
Table. 1, clearly demonstrates that all cooperation approaches offer improved perception outcomes when tested with RSU (V2I) compared to without RSU (V2V alone). This is attributed to the availability of more data to train the model. Additionally, RSU can always provide support in key scenes such as intersections and crosswalks as it has elevated positioned sensors which mitigates the shortcomings of vehicle perception such as limited field-of-view and potentially frequent occlusion. It is important to note that RSU data undergoes the same processing as vehicle data, as all point cloud data is transformed into a BEV. This transformation unifies the visual data representation across all agents, whether from vehicles or infrastructure, thereby maintaining spatial consistency by preserving the spatial arrangement of objects in the environment. An analysis of the AP@IoU 0.5 and 0.7 values in Table 2 reveals that collaboration generally improves performance across all cases compared to the absence of cooperation, with the exception of late collaboration. This exception occurs because late collaboration is adversely affected by the transmitted detection results from other agents, leading to an increase in false positives. Among the collaboration methods, the early collaboration outperforms all other methods (in many cases), however, it poses challenges with the shared data size (as shown

TABLE 1. The overall performances and architectures of the proposed approaches in this work, with each approach indicated with a unique experiment name. Where "Aggregation Operation" represents the "⊕" featured in Eq. 2. "Depths" represents the feature map dimensionality reduction to computer the attention weights. The performance is evaluated using the AP@IoU 0.5 and 0.7 for both w/o RSU and w/RSU. Δ_{size}% indicates the model size differences, where ↑ and ↓ denote whether the proposed method's model size is higher or lower, respectively, than V2VNet and DiscoNet. Note results in **red, blue,** denoting the 1st and 2nd highest AP results.

Feature Extractor	Experiment Name	Aggregation Operation	Depths	AP@IoU=0.5		AP@IoU=0.7		Model Size (MB)	Δ_{size}% V2VNet	Δ_{Size}% DiscoNet
				w/o RSU	w/RSU	w/o RSU	w/ RSU			
DiscoNet	Exp_D_Sum64	Sum	256, 128, 64	68.46	74.04	62.42	72.62	122.6	↓ 32.4	↑ 1.0
	Exp_D_Sum32	Sum	256, 128, 64, 32	68.03	74.41	62.10	72.71	123.5	↓ 32.0	↑ 1.7
	$Exp_D\_Concat64$	Concat	512, 256, 128, 64	**68.64**	**74.62**	**62.32**	**72.81**	127.5	↓ 29.7	↑ 5.0
	$Exp_D\_Concat32$	Concat	512, 256, 128, 64, 32	**69.67**	**75.57**	**63.72**	**73.87**	127.2	↓ 30.0	↑ 4.8
V2VNet	$Exp_V\_Concat64$	Concat	512, 256, 128, 64	67.31	72.78	62.26	71.14	127.1	↓ 30.0	↑ 4.7
	$Exp_V\_Concat32$	Concat	512, 256, 128, 64, 32	68.94	70	63.23	68.37	127.2	↓ 30.0	↑ 4.8

TABLE 2. Quantitative results of BEV detection on V2X-Sim 2.0 [27]. AP denotes average perception w/o RSU and w/RSU at IoU of 0.5 and 0.7. Note results in **red, blue, green** denoting the 1st,2nd and 3rd highest AP results.

Method	Collaboration Method			AP@IoU=0.5		AP@IoU=0.7	
	Early	Intermediate	Late	w/o RSU	w/ RSU	w/o RSU	w/RSU
When2com [49]	✗	✓	✗	44.02	46.39	39.89	40.32
When2com* [49]	✗	✓	✗	45.35	48.28	40.45	41.43
Who2com [50]	✗	✓	✗	44.02	46.39	39.89	40.32
Who2com* [50]	✗	✓	✗	45.35	48.28	40.45	41.13
V2VNet [10]	✗	✓	✗	68.35	72.08	62.83	65.85
DiscoNet [12]	✗	✓	✗	69.03	72.87	63.44	66.4
CORE [31]	✗	✓	✗	**70.0**	-	**64.9**	-
$Exp_D\_Concat64$ **(Ours)**	✗	✓	✗	68.64	**74.62**	62.32	**72.81**
$Exp_D\_Concat32$ **(Ours)**	✗	✓	✗	**69.67**	**75.57**	**63.72**	**73.87**
No Collaboration [27]	✗	✗	✗	49.9	46.96	44.21	42.33
Late Collaboration [27]	✗	✗	✓	43.99	42.98	39.1	38.26
Early Collaboration [27]	✓	✗	✗	**70.43**	**77.08**	**67.04**	**72.57**



(a)DiscoNet (b)V2VNet (c)$Exp_D\_Concat64$ (d)$Exp_D\_Concat32$

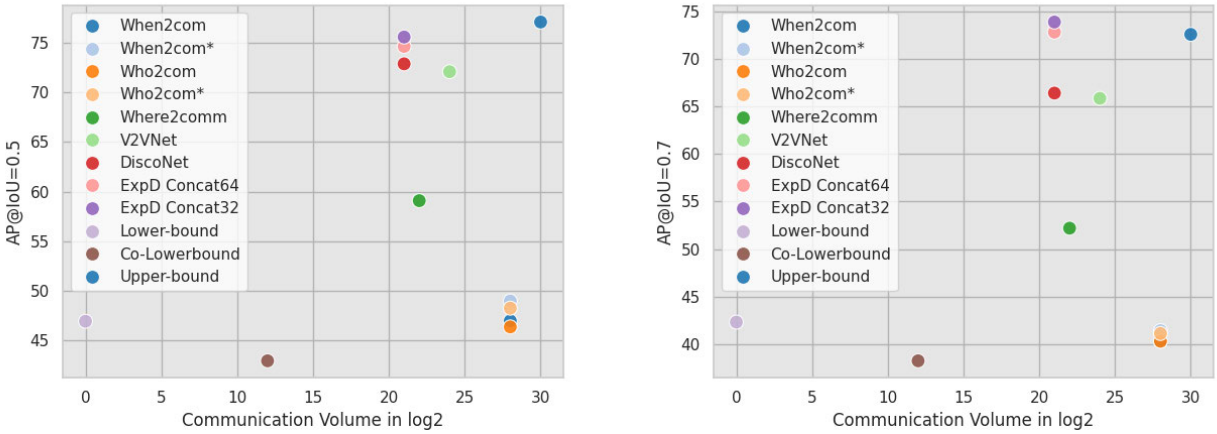
FIGURE 4. Visualizations comparing the BEV object detection on V2X-Sim of the intermediate baseline models (i.e. (a) DiscoNet, (b) V2VNet), and our proposed methods (i.e. (c) $Exp_D\_Concat64$ and (d) $Exp_D\_Concat32$). The green and red boxes denote ground truth and predictions, respectively. The blue lines indicate the vehicles that were not detected by (a) and/or (b) but were detected in (c) and/or (d).

in Fig.5) when compared to the intermediate collaboration methods. Among the collaboration methods, our proposed $Exp_D\_Concat64$ and $Exp_D\_Concat32$ achieve similar and in some cases superior AP among the intermediate models.

Our proposed methods surpass the performance of the state-of-the-art DiscoNet [12] and V2VNet [10], due to our proposed graph attention network architecture that enabled the model to selectively focus on relevant regions within the feature maps received from neighboring vehicles which facilitates flexible feature fusion by dynamically adjusting the contribution (through attention weights) of each feature map based on its relevance while preserving the spatial relationships between those feature maps. Fig.5 compares the proposed $Exp_D\_Concat64$ and $Exp_D\_Concat32$ with the baseline methods in terms of the trade-off between detection performance (AP@IoU = 0.50 and 0.7) and communication bandwidth. In plots (a) and (b), represent the lower-bound with no collaboration shown on the x-axis line since the

TABLE 3. Quantitative results of BEV detection on OPV2V [43]. The results in red, blue, green denoting the 1^{st},2^{nd} and 3^{rd} highest AP results.

Method	Collaboration Method			Default		Culver	
	Early	Intermediate	Late	AP@0.5	AP@0.7	AP@0.5	AP@0.7
F-Cooper [16]	✗	✓	✗	61.7	49.8	53.7	44.5
Who2Com [50]	✗	✓	✗	62.0	50.5	54.1	44.2
AttFuse [43]	✗	✓	✗	62.8	50.8	54.0	46.3
V2VNet [10]	✗	✓	✗	63.3	51.6	54.5	45.8
HP3D-V2V [30]	✗	✓	✗	**67.4**	**56.5**	**58.8**	**50.5**
$Exp_D\_Concat64$ **(Ours)**	✗	✓	✗	**67.6**	**57.1**	**59.1**	**50.7**
$Exp_D\_Concat32$ **(Ours)**	✗	✓	✗	**68.4**	**58.3**	**60.0**	**51.8**
No Collaboration [30]	✗	✗	✗	49.1	38.3	40.6	26.7
Late Collaboration [30]	✗	✗	✓	59.6	42.5	49.4	39.7
Early Collaboration [30]	✓	✗	✗	52.3	40.6	42.5	35.3



(a) Scatterplot at AP@IoU 0.5

(b) Scatterplot at AP@IoU 0.7

FIGURE 5. Performance-bandwidth trade-off of the collaborative perception methods evaluated in Table. 2. $Exp_D\_Concat32$ achieves consistently the best performance-bandwidth trade-off on all the collaborative perception datasets.

communication volume is zero. The communication results quantify the size of the message in bytes using a logarithmic scale with a base of 2, as defined in Eq.12. Among the collaboration methods, the early collaboration (upper-bound) outperforms all other methods (in many cases), however, it poses challenges with the shared data size when compared to the intermediate collaboration methods. Notably, $Exp_D\_Concat32$ exhibits a notable advancement in achieving a superior precision-communication trade-off across diverse communication bandwidth configurations and collaborative perception tasks, slightly surpassing the performance of DiscoNet, whose feature extractor is utilized in our study.

On another note, all of our proposed methods consistently result in much lower parameter budgets when compared to V2VNet, as illustrated by "Δ_{size}% V2VNet" column in Table 1, which shows a maximum and minimum delta size of 32.4% and 30.0%. Compared to DiscoNet, our models are slightly bigger ranging between 1% to 5%. The communication volume for the features maps resulting from Exp_D and the Exp_V is the same as DiscoNet and V2VNet, respectively, as we have utilized the same feature extractor to guarantee that the enhancement in the AP is purely due to our proposed feature fusion method (as discussed earlier in section. III-C).

2) OPV2V

As shown in Table. 2, all cooperative methods perform better than no fusion, demonstrating the benefits of a multi-agent perception system. Among all fusion models, our $Exp_D\_Concat32$ consistently achieves the best IoU scores for all four categories in both the default and Culver City settings, outperforming the second-best method, HP3D-V2V [30]. These results highlight the effectiveness of $Exp_D\_Concat32$ in improving the performance of intermediate collaboration, and its superior results in Culver City demonstrate its strong generalization ability.

C. QUALITATIVE RESULTS

Fig. 4 shows two different scenes of the detection results on V2XSim to compare the proposed $Exp_D\_Concat32$ and

$Exp_D\_Concat64$ with the baselines of intermediate collaboration methods (DiscoNet and V2VNet). We see that $Exp_D\_Concat32$ and $Exp_D\_Concat64$ qualitatively outperform the state-of-the-art methods and are able to detect more objects. Comparing (a)/(b) with (c)/(d), we see that the missed or incorrectly detected vehicles in (a)/(b) (indicated in blue), are correctly detected in (c)/(d). This shows that our methodology achieves the best compensation and precisely recovers the true position of vehicles, which can give the downstream planning system more information to plan a safe maneuver better. The reason is that V2VNet employs a scalar to denote the agent-to-agent attention, which cannot distinguish the informative regions; while DiscoNet can adaptively find the beneficial regions in a cell-level resolution, however, the results show that our attention method was able to perform better.

D. ABLATION STUDY

Incorporating both the channel-spatial attention module for attention map generation yields an increase in AP. This enhancement is attributed to the channel attention mechanism, which enables the model to concentrate on crucial features within individual channels, thereby augmenting the representation of pertinent information. Simultaneously, spatial attention empowers the model to selectively attend to specific spatial locations in the feature maps, facilitating the capture of spatial dependencies and correlations among disparate regions. Consequently, this refinement contributes to improved spatial accuracy and localization capabilities within the model. The adoption of this integrated approach significantly enhances overall perception performance by selectively attending to informative features. As previously indicated in section. III-C, we leverage the feature extractors from DiscoNet and V2VNet denoted as $\Theta_{DiscoNet}$ and Θ_{V2VNet}, respectively. It has been empirically validated that the incorporation of the proposed channel-spatial attention module results in further advancements in AP, with a marginal increase in model size compared to DiscoNet. Notably, the proposed method exhibits a substantial reduction in model size when compared with V2VNet. This underscores the efficacy of the proposed methodology in achieving improved performance metrics.

1) EFFECT OF AGGREGATION OPERATION

Table. 1 shows the contribution of individual components in our proposed framework for detection. It is evident that the $Exp_D\_Concat32$ setup achieves the highest detection precision while keeping a low model parameter count; while the $Exp_D\_Concat64$ model performs second best in terms of overall performance. The overall quantitative analysis of the proposed methodology, reveals that concatenation archives slightly better results when compared to summation. This is because concatenation allows for more flexible feature fusion by preserving both the collective information and the individual characteristics of the input feature maps. The concatenation allows the model to capture a richer set of features by increasing the number of channels for instance if there are F_i and F_j of shape $H \times W \times C$ after concatenation the shape becomes $H \times W \times 2C$. Additionally, the concatenation facilitates the integration of information from different sources without losing any individual feature maps. This is beneficial when providing the model access to different types of information captured by the input features. This comprehensive integration is the reason behind the slight improvement when compared to summation as it leads to a more nuanced understanding of the aggregated feature enhancing the attention map learning. However, concatenation increases the dimensionality of the feature maps, which can offer a broader representation space but also requires additional computational resources compared to summation.

2) EFFECT OF DEPTHS ON ATTENTION

For learning channel and spatial attention coefficients presented in Eqs. 3 and 5, it s evident, as shown in Table. 1, that a deeper encoder-decoder architecture leads to higher AP. The reason for this is our proposed attention network is based on a PwC-based network, where the deeper layers of the network learn higher-level, and more abstract representations of the input. This enables the network to learn more intricate patterns and correlations between features. This is because the deeper layers can combine features learned in earlier layers in order to create higher-level representations that capture more complex features of the input data which is crucial for learning the attention weights. However, we also found that going beyond the depth that was examined led to a decrease in AP due to the vanishing gradient, which occurs when the gradient signal is too small to propagate through multiple layers.

VI. CONCLUSION

This work focuses on LiDAR-based intermediate collaborative perception, limited to single sensor modality and vehicle detection tasks. We propose a novel intermediate-collaboration perception approach that fuses intermediate features received from multiple agents. The effectiveness of our proposed method is demonstrated using the V2XSim and compared against the baseline state-of-the-art methods. The results of our proposed methods achieve superior performance over the state-of-the-art intermediate collaboration methods. This is because the core component of our methodology is based on GAT, where we develop and utilize both channel and spatial attention for feature fusion. Our proposed attention mechanism guides the model to highlight significant regions within the feature maps of the two neighboring nodes. This trains the model to determine ''where'' and ''what'' to pay attention to, leading to a more reliable feature fusion. Our future work will address the latency challenges encountered when sharing feature maps in V2X perception. In particular, we plan to incorporate transmission delay, as it adversely affects the performance of collaborative perception.

REFERENCES

[1] S.-y. Wang, Z. Qu, and L.-y. Gao, "Multi-spatial pyramid feature and optimizing focal loss function for object detection," *IEEE Trans. Intell. Vehicles*, pp. 1–13, 2023.

[2] J. E. Hoffmann, H. G. Tosso, M. M. D. Santos, J. F. Justo, A. W. Malik, and A. U. Rahman, "Real-time adaptive object detection and tracking for autonomous vehicles," *IEEE Trans. Intell. Vehicles*, vol. 6, no. 3, pp. 450–459, Sep. 2021.

[3] G. Xian, C. Ji, L. Zhou, G. Chen, J. Zhang, B. Li, X. Xue, and J. Pu, "Location-guided LiDAR-based panoptic segmentation for autonomous driving," *IEEE Trans. Intell. Vehicles*, vol. 8, no. 2, pp. 1473–1483, Feb. 2023.

[4] T.-H. Chen and T. S. Chang, "RangeSeg: Range-aware real time segmentation of 3D LiDAR point clouds," *IEEE Trans. Intell. Vehicles*, vol. 7, no. 1, pp. 93–101, Mar. 2022.

[5] K. Samal, H. Kumawat, P. Saha, M. Wolf, and S. Mukhopadhyay, "Task-driven RGB-LiDAR fusion for object tracking in resource-efficient autonomous system," *IEEE Trans. Intell. Vehicles*, vol. 7, no. 1, pp. 102–112, Mar. 2022.

[6] P. Karle, F. Fent, S. Huch, F. Sauerbeck, and M. Lienkamp, "Multi-modal sensor fusion and object tracking for autonomous racing," *IEEE Trans. Intell. Vehicles*, pp. 1–13, 2023.

[7] K. Wang, T. Zhou, X. Li, and F. Ren, "Performance and challenges of 3D object detection methods in complex scenes for autonomous driving," *IEEE Trans. Intell. Vehicles*, vol. 8, no. 2, pp. 1699–1716, Feb. 2023.

[8] 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 Trans. Intell. Vehicles*, pp. 1–13, 2023.

[9] C. Chang, J. Zhang, K. Zhang, W. Zhong, X. Peng, S. Li, and L. Li, "BEV-V2X: Cooperative birds-eye-view fusion and grid occupancy prediction via V2X-based data sharing," *IEEE Trans. Intell. Vehicles*, pp. 1–18, 2023.

[10] 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. Eur. Conf. Comput. Vis.* Cham, switzerland: Springer, 2020, pp. 605–621.

[11] E. E. Marvasti, A. Raftari, A. E. Marvasti, Y. P. Fallah, R. Guo, and H. Lu, "Feature sharing and integration for cooperative cognition and perception with volumetric sensors," 2020, *arXiv:2011.08317*.

[12] 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.*, vol. 34, 2021, pp. 29541–29552.

[13] 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. (ICDCS)*, Jul. 2019, pp. 514–524.

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

[15] J. Shi, W. Wang, X. Wang, H. Sun, X. Lan, J. Xin, and N. Zheng, "Leveraging spatio-temporal evidence and independent vision channel to improve multi-sensor fusion for vehicle environmental perception," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 591–596.

[16] 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 *Proc. 4th ACM/IEEE Symp. Edge Comput.*, Nov. 2019, pp. 88–100.

[17] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, "A comprehensive survey on graph neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 1, pp. 4–24, Jan. 2021.

[18] J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, "Graph neural networks: A review of methods and applications," *AI Open*, vol. 1, pp. 57–81, Jan. 2020.

[19] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.

[20] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," 2017, *arXiv:1710.10903*.

[21] S. Casas, C. Gulino, R. Liao, and R. Urtasun, "SpAGNN: Spatially-aware graph neural networks for relational behavior forecasting from sensor data," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2020, pp. 9491–9497.

[22] L. L. Li, B. Yang, M. Liang, W. Zeng, M. Ren, S. Segal, and R. Urtasun, "End-to-end contextual perception and prediction with interaction transformer," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2020, pp. 5784–5791.

[23] Q. Sykora, M. Ren, and R. Urtasun, "Multi-agent routing value iteration network," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 9300–9310.

[24] M.-H. Guo, T.-X. Xu, J.-J. Liu, Z.-N. Liu, P.-T. Jiang, T.-J. Mu, S.-H. Zhang, R. R. Martin, M.-M. Cheng, and S.-M. Hu, "Attention mechanisms in computer vision: A survey," *Comput. Vis. Media*, vol. 8, pp. 1–38, Mar. 2022.

[25] Y. Dai, F. Gieseke, S. Oehmcke, Y. Wu, and K. Barnard, "Attentional feature fusion," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2021, pp. 3559–3568.

[26] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2018, pp. 3–19.

[27] Y. Li, D. Ma, Z. An, Z. Wang, Y. Zhong, S. Chen, and C. Feng, "V2X-sim: Multi-agent collaborative perception dataset and benchmark for autonomous driving," *IEEE Robot. Autom. Lett.*, vol. 7, no. 4, pp. 10914–10921, Oct. 2022.

[28] E. Arnold, M. Dianati, R. de Temple, and S. Fallah, "Cooperative perception for 3D object detection in driving scenarios using infrastructure sensors," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 3, pp. 1852–1864, Mar. 2022.

[29] A. Miller, K. Rim, P. Chopra, P. Kelkar, and M. Likhachev, "Cooperative perception and localization for cooperative driving," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2020, pp. 1256–1262.

[30] H. Chen, H. Wang, Z. Liu, D. Gu, and W. Ye, "HP3D-V2V: High-precision 3D object detection Vehicle-to-Vehicle cooperative perception algorithm," *Sensors*, vol. 24, no. 7, p. 2170, Mar. 2024.

[31] B. Wang, L. Zhang, Z. Wang, Y. Zhao, and T. Zhou, "Core: Cooperative reconstruction for multi-agent perception," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 8710–8720.

[32] 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 *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2022, pp. 107–124.

[33] C. Lin, D. Tian, X. Duan, J. Zhou, D. Zhao, and D. Cao, "V2VFormer: Vehicle-to-vehicle cooperative perception with spatial-channel transformer," *IEEE Trans. Intell. Vehicles*, vol. 9, no. 2, pp. 3384–3395, Feb. 2024.

[34] Y. Hu, S. Fang, Z. Lei, Y. Zhong, and S. Chen, "Where2comm: Communication-efficient collaborative perception via spatial confidence maps," in *Proc. 36th Conf. Neural Inf. Process. Syst. (Neurips)*, Nov. 2022, pp. 4874–4886.

[35] Y. Zhou, J. Xiao, Y. Zhou, and G. Loianno, "Multi-robot collaborative perception with graph neural networks," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 2289–2296, Apr. 2022.

[36] C. Reading, A. Harakeh, J. Chae, and S. L. Waslander, "Categorical depth distribution network for monocular 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 8551–8560.

[37] P. Wu, S. Chen, and D. N. Metaxas, "MotionNet: Joint perception and motion prediction for autonomous driving based on bird's eye view maps," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11382–11392.

[38] J. B. Kenney, "Dedicated short-range communications (DSRC) standards in the United States," *Proc. IEEE*, vol. 99, no. 7, pp. 1162–1182, Jul. 2011.

[39] M. Jaderberg, K. Simonyan, and A. Zisserman, "Spatial transformer networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 28, 2015, pp. 1–11.

[40] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.

[41] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "ECA-Net: Efficient channel attention for deep convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11531–11539.

[42] N. Komodakis and S. Zagoruyko, "Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer," in *Proc. ICLR*, 2017.

[43] R. Xu, H. Xiang, X. Xia, X. Han, J. Li, and J. Ma, "OPV2 V: An open benchmark dataset and fusion pipeline for perception with vehicle-to-vehicle communication," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2022, pp. 2583–2589.

[44] D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker, "Recent development and applications of sumo-simulation of urban mobility," *Int. J. Adv. Syst. Meas.*, vol. 5, nos. 3–4, 2012.
[45] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in *Proc. 1st Annu. Conf. Robot Learn.*, vol. 78, S. Levine, V. Vanhoucke, and K. Goldberg, Eds., Nov. 2017, pp. 1–16.
[46] 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. (ITSC)*, Sep. 2021, pp. 1155–1162.
[47] R. Padilla, S. L. Netto, and E. A. B. da Silva, "A survey on performance metrics for object-detection algorithms," in *Proc. Int. Conf. Syst., Signals Image Process. (IWSSIP)*, Jul. 2020, pp. 237–242.
[48] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, 2015.
[49] Y.-C. Liu, J. Tian, N. Glaser, and Z. Kira, "when2com: Multi-agent perception via communication graph grouping," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 4105–4114.
[50] Y.-C. Liu, J. Tian, C.-Y. Ma, N. Glaser, C.-W. Kuo, and Z. Kira, "who2com: Collaborative perception via learnable handshake communication," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2020, pp. 6876–6883.

AHMED N. AHMED (Member, IEEE) received the master's degree in sustainable automotive engineering from the Faculty of Applied Engineering, University of Antwerp, in 2020. He is currently pursuing the Ph.D. degree with the Imec Research Group, IDLab, University of Antwerp. 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, in December 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 of the 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 received 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 Harbin Institute of Technology, Harbin, China, in 2019. Since 2020, he has been a Principle Research Fellow with Imec Research Group, IDLab, University of Antwerp, where he is 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 serves in 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 serves regularly as a Reviewer in journals, including IEEE Transactions on Industrial Electronics, IEEE Transactions on Industrial Informatics, IEEE/ASME Transactions on Mechatronics, and IEEE Access.