Multi-Modal Virtual-Real Fusion based Transformer for Collaborative Perception

Hui Zhang †
School of Computer and Information Technology
Beijing Jiaotong University
Beijing, China
huizhang1@bjtu.edu.cn

Guiyang Luo †
the State Key Laboratory of Networking and Switching Technology
Beijing University of Posts and Telecommunications
Beijing, China
luoguiyang@bupt.edu.cn

Yuanzhouhan Cao, Yi Jin, Yidong Li*
School of Computer and Information Technology
Beijing Jiaotong University
Beijing, China
{yzhcao, yjin, ydli}@bjtu.edu.cn

2022 IEEE 13th International Symposium on Parallel Architectures, Algorithms and Programming (PAAP) | 978-1-6654-5218-2/22/$31.00 ©2022 IEEE | DOI: 10.1109/PAAP56126.2022.10010640

***Abstract*—Automobile intelligence and networking have become the inevitable trend in the future development of the automotive industry. Existing intelligent and connected vehicles rely on single-agent intelligence to perform the basic perception, which is still weak in dealing with the problem of accurate recognition and positioning in complex traffic scenes such as small and far away objects. To tackle this issue, we propose a multi-model virtual-real fusion Transformer for collaborative perception. Specifically, to possess the complementary information from both RGB images and LiDAR point clouds, we propose the multi-model virtual-real fusion (MVRF) method, which generates virtual points and compensates for the lack of point information on sparse locations. Furthermore, the heterogeneous graph attention network (HGAN) is constructed to capture the inter-agent interaction and adaptively incorporate multiple agents' features. The HGAN contains a series of encoder layers, each of which has a heterogeneous inter-agent attention module and a multi-scale self-attention module, which motivates to learn different relationships based on various agents' types and simultaneously capture the global and local spatial attention. Extensive experiments demonstrate that the proposed method gains superior performance as compared with state-of-the-art methods.**

***Index Terms*—Collaborative Perception, Intelligent and Connected Vehicle, Multi-Model Fusion**

I. Introduction

Intelligent and connected vehicles (ICV) refer to carrying advanced onboard sensors, controllers, actuators, and other devices, and integrating modern communication and network technologies [1], [2] to achieve intelligent information exchange and sharing between vehicles, roads, people, etc [3], [4]. With complex environmental perception, intelligent decision-making, collaborative control, and other functions, it can realize safe, efficient, comfortable, and energy-saving driving, and finally realize a new generation of vehicles that replace human operation. ICV integrates perception, decision-making, and control. Specifically, it senses the road environment by analyzing the real-time signals of onboard sensors and decides on safe and efficient driving routes, so as to control the steering and speed of vehicles to reach the predetermined location. Therefore, environmental perception [5]–[7] is the basic function of ICV, which is the premise of realizing behavior decision-making and vehicle control.

Existing ICV rely on local multi-source and redundant sensors to realize the perception of the driving environment (single-agent intelligence). For example, Huawei's high-level autonomous vehicles are equipped with various types of sensors, including laser radars, millimeter-wave radars, ultrasonic radars, and cameras [8]. However, due to the limitations of the installation location, detection distance, and angle of view of on-board sensors, it is still difficult to deal with the problem of accurate perception, recognition, and positioning in complex traffic scenes such as busy intersections, bad weather, small and far away objects. Relying on single-agent sensor fusion is difficult to solve the above problems, which seriously restricts and affects the safety and robustness of ICV. For instance, in May 2016, a Tesla Model S collided with a semi-trailer truck turning left at an intersection in Florida, killing one person. The main cause of this accident is that the perception ability of the single-agent camera is affected by complex and changeable environmental factors [9]. Especially in urban road scenes, the range and accuracy of single-agent perception are vulnerable to weather, object distance, and other factors. Depending only on the stacking of a single agent's sensors can not overcome the barrier of insufficient perception ability in complex traffic scenes [10]. Collaborative perception provides an alternative way to extend the vehicle's perception range and improve perception accuracy [11].

Recent collaborative perception focuses on how to utilize visual cues from neighboring vehicles and infrastructure to boost the overall perception performance. Some approaches

† Contributed Equally.

* Corresponding Author.

This paper is supported by the Fundamental Research Funds for the Central Universities (KKRC22002532).

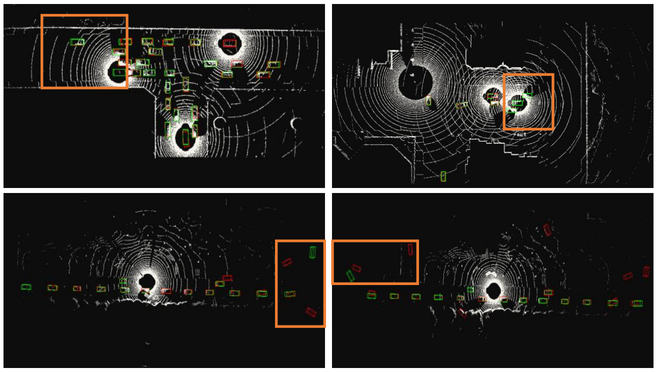

Fig. 1. The failure visualization of the current collaboration detector. The ground truth and predictions are represented in green and red bounding boxes, respectively. As can be seen from the detection inside the orange boxes, the faraway object with quite sparse point clouds cannot be detected precisely using current detector.

utilize raw data fusion to acquire a holistic view [12]. They can fundamentally overcome the problems of small and occluded objects but require large transmission bandwidth. A series of works perform collaboration via late fusion [13], [14]. Although they are bandwidth-efficient, the perceptual output of each agent may be noisy and incomplete, leading to irreversible information loss and perception errors. Intermediate fusion is a more effective way to deal with the trade-off between perception accuracy and communication bandwidth [15]–[17]. F-Cooper [18] incorporates the shared features with the same weights. V2VNet [19] aggregates features by a spatially-aware graph neural network. DiscoNet [20] proposes a teacher-student framework to enhance training by restricting the encoded features to that from early fusion. The student model can learn better representation under the guidance of early fusion supervision. However, it still cannot perform well on faraway small objects. As shown in Fig. 1, the faraway object with quite sparse point clouds cannot be detected precisely. There are some false positives and false negatives in the case of sparse point clouds.

To address this problem, in this paper, we introduce a multi-model virtual-real fusion Transformer for collaborative perception. Firstly, we choose one of the ICV as the ego vehicle to build a heterogeneous graph among the connected agents (vehicle or infrastructure). All other connected agents (exclude ego vehicles) map their LiDAR point clouds into the ego vehicle's coordinate format. In the meanwhile, we propose the multi-model virtual-real fusion (MVRF) method to enhance the information on sparse point cloud locations. Then, the anchor-based PointPillar [23] is adopted to generate the visual features from multiple agents' point clouds. To effectively model inter-agent interaction and adaptively aggregate their features, the heterogeneous graph attention network is proposed, consisting of 3 encoder layer, each of which contains a heterogeneous inter-agent attention (Hiaa) module, a multi-scale self-attention (Mssa) module and a feed-forward network (Ffn). Finally, the aggregated features are further used for classification and bounding box regression.

II. Methodology

In this paper, we treat the V2X perception as the heterogeneous multi-agent collaborative perception. We aim to utilize a specially-designed Transformer based on multi-model information that can effectively achieve multi-agent collaborative feature fusion for dynamic heterogeneous relationship graph, as shown in Fig. 2. Specifically, to employ the complementary information from both RGB images and LiDAR point clouds, we propose the MVRF method to generate the virtual points based on RGB images and perform virtual-real incorporation, so that they could jointly be leveraged to reduce the false negatives and false positives on sparse point locations. Moreover, the HGAN module is built to capture both the long-term global dependencies and local context information.

A. Multi-Model Virtual-Real Fusion

The MVRF depends on 2D detection, 3D detection, and transformation from 2D to 3D space. For 2D detection. we leverage the CenterNet [21] detector, which detects each object as a triplet of keypoints. Specifically, we denote the RGB image by X. The detector takes X as input and predicts the bounding boxes $B_x \in \mathbb{R}^4$ and category scores $S_x \in \mathbb{R}^C$, where C is the number of class. Furthermore, we extend a cascade RoI head [22] on top of the proposal generation network for mask estimation, which generates pixel-wise segmentation $M_{x,b} \in [0,1]^{W \times H}$ for each object. For 3D detection, let $P = \{(u_i, v_i, o_i, r_i)\}$ be the point clouds where (u_i, v_i, o_i) is the 3D coordinates and r_i is the reflectance factor. A 3D detector aims to generate 3D bounding boxes B from the point cloud P. The 3D bounding boxes $B = \{(u_j, v_j, o_j, l_j, w_j, h_j, \theta_j)\}$ consist of the 3D center coordinate (u_j, v_j, o_j), object size (l_j, w_j, h_j) and the yaw rotation along z axis θ_j. In this paper, we process the point clouds P as bird-eye view pillars and extract features with the 2D convolutional backbone. The features from multiple agents' backbones are gathered and processed using the proposed HGAN in the following. In the end, they are adopted to generate the 3D bounding boxes B by the detection head in PointPillars [23].

Built upon the 2D detection results, we predict the dense virtual points $V = \{(u_k, v_k, o_k, r_k)\}$ and map them into 3D LiDRA space, as shown in Fig. 3. For implement this, we firstly map the LiDAR point clouds into RGB image coordinates. We denote the capture time of LiDAR sensor and RGB image by t and t', respectively. Let $T_{\text{lidar2car}}$ be the mapping from LiDAR sensor to car reference frame. $T_{t'2t}$ represents the mapping of car from t' to t. T_{car2rgb} represents the mapping from car reference frame to the RGB sensor. Therefore, the mapping from the LiDAR to RGB sensor is denoted by

$$T_{\text{lidar2rgb}} = T_{\text{car2rgb}} T_{t_2 2t_1} T_{\text{lidar2car}}. \tag{1}$$

Then, we get the image coordinates n_i with corresponding depth d_i and gather the mapping points for a single detection b as a cluster $U_b = \{(n_i, d_i) | n_i \in M_{x,b} \forall_i\}$. The cluster only

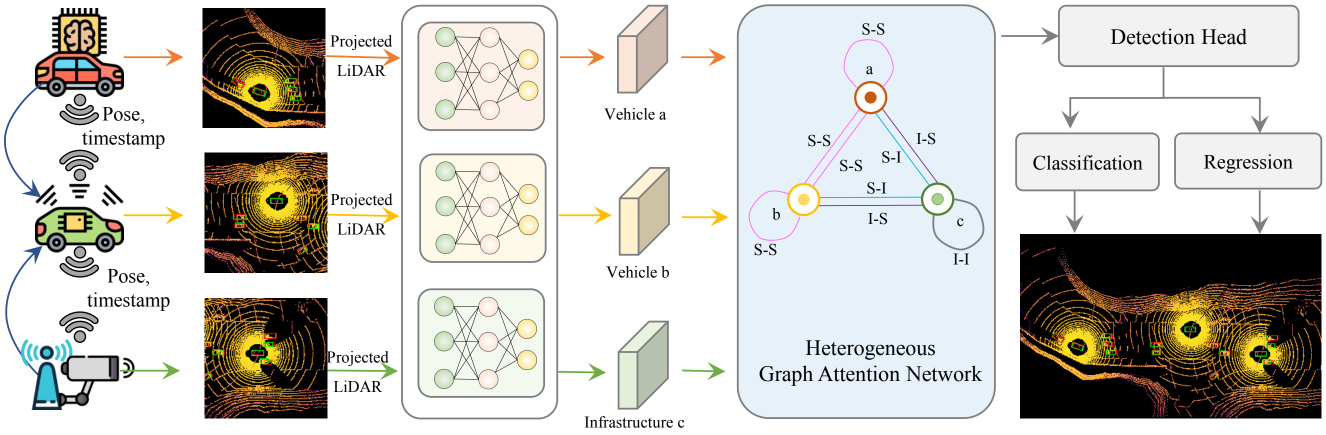

Fig. 2. Overview of the proposed method. Other vehicles firstly map their LiDAR point clouds into the ego vehicle's coordinate format. The predicted virtual points and real points are inputted into the PointPillar detector simultaneously. Meanwhile, the heterogeneous graph attention network is built to capture different agents' interactions and aggregate their features. The aggregated features are inputted into the detection head for classification and box regression.

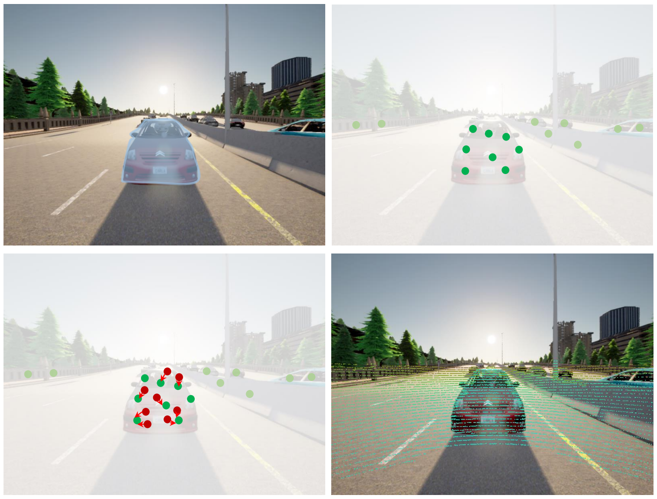

Fig. 3. Overview of the generation of the virtual points. We firstly map the LiDAR point clouds into the RGB image coordinate. Then, the random sampling method is used to generate the virtual points, whose depths are predicted by their nearest neighbor. Finally, the virtual points are mapped into 3D LiDAR space and incorporated with real points to further perform collaborative detection.

gathers the mapping points n_i that fall into the detection mask $M_{x,b}$. Next, the random sampling method is used to generate the virtual points $V \in M_{x,b}$ in the 2D image space. For each virtual point V_k, we predict the depth with its nearest neighbor in the cluster $U_b : d = \arg\min_{d_i} ||n_i - V_k||$. r_k is generated by the objectness scores in the 2D detector. Given the 2D coordinates and predicted depth, we can map the virtual points back into the 3D LiDAR space. The virtual points and original real points are incorporated to further perform collaborative detection.

B. Heterogeneous Graph Attention Network

The sensor information obtained from vehicles and infrastructure may maintain different properties. The LiDAR at the infrastructure is usually installed at a higher position, which can perceive objects in a wider range. The LiDAR at vehicles can perceive the surrounding environment within a certain range. To deal with this heterogeneity, we build a Transformer-based HGAN to capture both the long-term global dependencies and local context information, as shown in Fig. 4. The directed graph is denoted by $G = (P, E)$, where P is the node set consisting of infrastructure I and vehicles S and E contains four types of edges, that is $E = \{I - I, I - S, S - I, S - S\}$. For agent a, its node type is expressed as $P_a \in \{I, S\}$. The Transformer-based HGAN consists of 3 encoder layers, each of which contains a Hiaa module, a Mssa module, and a Ffn module.

The Hiaa module is proposed to incorporate information from distinct agents adaptively, expressed as

$$H_a = \underset{\forall b \in N(a)}{\text{Linear}_{P_a}}(\text{MultiHead}(a,b)\text{FeatAggre}(a,b)), \qquad (2)$$

which contains a linear network Linear_{P_a}, a heterogeneous multi-head attention MultiHead, and a feature aggregation network FeatAggre. $N(a)$ is the neighbor agent set whose distance from the main agent is less than a fixed threshold and P_a is the node type. Specifically, Linear_{P_a} contains a set of linear layer conditioned on the relevant node type P_a. $\text{MultiHead}(a,b)$ generates the attention weights between pairs of nodes, which is expressed as

$$\begin{aligned}\text{MultiHead}(a,b) =& [\text{head}^1_{multi}(a,b), \text{head}^2_{multi}(a,b), \\ & \cdots, \text{head}^h_{multi}(a,b)],\end{aligned} \qquad (3)$$

$$\begin{aligned}\text{head}^j_{multi}(a,b) =& \text{softmax}(\frac{1}{\sqrt{C}}(W^K_{P_b} H_b \times \\ & W^{multi,j}_{E(a,b)} \times (W^Q_{P_a} H_a)^T)),\end{aligned} \qquad (4)$$

where h is the total number of heads and j is current head number. $W^K_{P_b}$, $W^Q_{P_a}$, and $W^{multi,j}_{E(a,b)}$ have distinct parameters

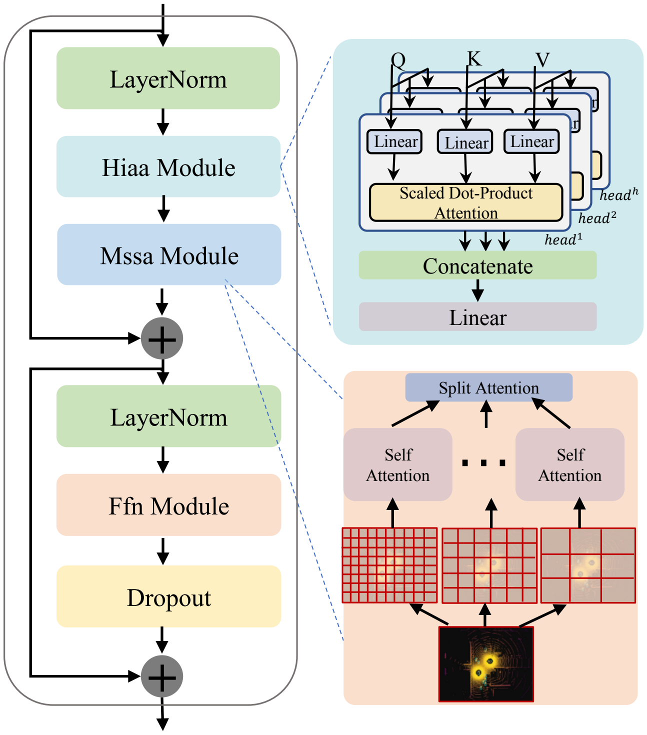

Fig. 4. The architecture of the encoder in Transformer-based HGAN, which mainly contains a Hiaa module, a Mssa module, and a Ffn module.

in different heads. FeatAggre(a, b) is adopted to aggregate different agents' features, which is denoted by

$$\begin{aligned}\text{FeatAggre}(a, b) =& [\text{head}^1_{aggre}(a, b), \text{head}^2_{aggre}(a, b), \\ & \cdots, \text{head}^h_{multi}(a, b)],\end{aligned} \tag{5}$$

$$\text{head}^j_{aggre}(a, b) = \text{linear}^j_{P_b}(H_b) \times W^{aggre,j}_{E(a,b)}, \tag{6}$$

where $\text{linear}^j_{P_b}(H_b)$ represents a linear layer which is indexed by both node type P_b and the current head number j.

The Mssa module is utilized to capture both the global and local spatial feature interaction. It adopts various window sizes to capture distinct attention ranges, which can boost the 3D detection performance greatly. Suppose one of the window size is $Z_j \times Z_j$ and the input feature is $H_a \in \mathbb{R}^{H\times W\times C}$, the output of Mssa can be formulated as

$$\text{head}^j = [h^1, h^2, \cdots, h^{HW/Z_j^2}], \tag{7}$$

$$\bar{\text{head}}^i_m = \text{softmax}(\frac{h^i W^Q_m \times (h^i W^K_m)^T}{\sqrt{d}} + L) h^i W^V_m, \tag{8}$$

$$O_m = [\bar{\text{head}}^1_m, \bar{\text{head}}^2_m, \cdots, \bar{\text{head}}^{HW/Z^2}_m], \tag{9}$$

$$O^j = [O_1, O_2, \cdots, O_{h_j}], \tag{10}$$

where head^j is the split feature set for branch j whose window size is $Z_j \times Z_j$. h^i is one of the element in head^j, where $i \in \{1, 2, \cdots, HW/Z_j^2\}$. W^Q_m, W^K_m, and W^V_m represent the query, key, and value mapping matrices. L is the relative positional encoding. O_m is the output of the m-th head for branch j, where $m = \{1, 2, \cdots, h_j\}$. h_j is the attention head number. The output of h_j heads are connected to generate the final output O^j for the current branch. To capture distinct attention ranges, we leverage different window sizes, each of which corresponds to one branch. We incorporate the output O^j from all the branches by a split attention module [24] to adaptively fuse multi-scale local and global context information.

III. Experiments

A. Experimental Setup

We choose the public-available V2X perception dataset V2XSet [25]. It is generated by CARLA [26] and OpenCDA [27]. CARLA provides real traffic scene modeling and sensor simulation and OpenCDA is used for simultaneous control of a series of vehicles and vehicle network communication protocols. The dataset collects 73 traffic scenes overlaying 5 distinct roadway types and 8 towns in CARLA. There are at most 25 seconds for each scene. Besides, the number of agents that can communicate with each other is between 2 and 5. Each agent is equipped with 32-channel LiDAR and has 120 meters data range. The infrastructure sensors are deployed in the intersection, mid-block, and entrance ramp at the height of 14 feet. The dataset contains 6694 training samples, 1920 validation samples, and 2833 testing samples in total.

In the training, the ego vehicles will be selected randomly among vehicles, while a fixed ego vehicle is set for all compared methods in the testing. The voxel size is set to [0.4m, 0.4m, 4m] along the X, Y, and Z axis. The detection range is set as [-140.8, 140.8]m, [-38.4, 38.4]m, [-3, 1]m along the X, Y, and Z axis. The Transformer network contains 3 encoder layers and there are 3 types of window size in the Mssa module, that is, $Z_j \in \{4, 8, 16\}$. The Adam optimizer is applied to train the proposed method in an end-to-end manner. The initial learning rate is set as 10^{-3} with a factor of 0.1 decay every 10 epochs.

B. Experimental Results

In this paper, we take the NoFusion method, which only leverages the point clouds from ego vehicle without collaborative perception, as the baseline. We also make a comparison with the LateFusion and EarlyFusion mehod. LateFusion merges the recognition results of multiple agents and utilizes the non-maximum suppression to generate the final detection results. EarlyFusion gathers raw sensor data of multiple agents to achieve global perception of the surrounding environment. IntermediateFusion aggregates the encoded features from each agent and leverages the aggregated features to achieve collaborative detection. We compare with five state-of-the-art IntermediateFusion methods: F-Cooper [18], OPV2V [28], V2VNet [19], DiscoNet [20], and V2X-ViT [25].

Table I demonstrates the performance comparisons with previous state-of-the-art methods on the V2XSet dataset. On the perfect setting, the methods with collaborative detection achieve significant performance improvement over the baseline NoFusion. Our proposed method outperforms it by 28.3% and 31.3% in AP@0.5 and AP@0.7, respectively. Compared

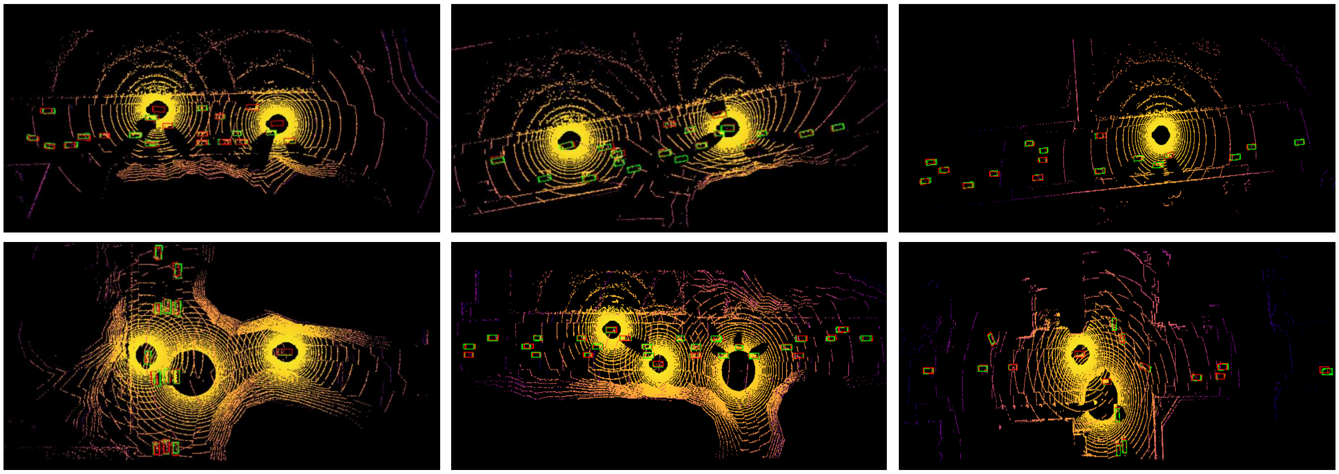

Fig. 5. Qualitative results of our proposed method on V2XSet dataset. The ground truth and predictions are represented in green and red bounding boxes, respectively.

with EarlyFusion, we have a 7.0% AP increase in AP@0.5 and a 0.5% AP increase in AP@0.7. F-Cooper [18] leverages feature-level fusion and sticks out the important features with the help of maxout strategy. the performance drops by 4.9% in AP@0.5 and 3.5% in AP@0.7. OPV2V [28] realizes a single-head self-attention fusion mechanism to aggregate the feature among multiple agents. Our proposed method outperforms it by a large margin, i.e., 8.2% in AP@0.5 and 5.1% in AP@0.7. V2VNet [19] utilizes a spatially-aware graph neural network to parse and aggregate different agents' features. Compare with that, our method increases 4.4% AP in AP@0.5 and 3.8% AP in AP@0.7. DiscoNet [20] proposes a distilled collaboration graph, incorporating the knowledge distillation and matrix-valued edge weight, to adaptively model inter-agent attention and enhance the performance of collaborative detection. Our proposed method outperforms it by 4.5% and 2.0% in AP@0.5 and AP@0.7, respectively. V2X-ViT [25] proposes the multi-agent self-attention and multi-scale window self-attention module to capture inter-agent interaction and intra-agent spatial relationship. It achieves 88.2% AP in AP@0.5 and 71.2% in AP@0.7, whose performance decreases by 0.7% and 0.3% in AP@0.5 and AP@0.7, respectively.

On the noisy setting, the accuracy of LateFusion dramatically decreases to 54.9% and 30.7% in AP@0.5 and AP@0.7, respectively, even worse than the NoFusion baseline. The performance of EarlyFusion achieves 38.4% in AP@0.7, which is still worse than the NoFusion baseline. In contrary, our method achieves 84.3% in AP@0.5 and 61.9% in AP@0.7, which has a significant improvement that the baseline, i.e., 23.7% higher in AP@0.5 and 21.7% higher in AP@0.7. For intermediate fusion, our method significantly outperforms F-Cooper [18] and OPV2V [28] in both AP@0.5 and AP@0.7. Specifically, F-Cooper achieves 46.9% in AP@0.7 and OPV2V achieves 48.7% while our accuracy is 61.9% AP. We further compare our results with V2VNet [19], which directly adopts a spatial-aware graph network to incorporate features. Our improvement over it is 5.2% in AP@0.5 and 12.6% in AP@0.7. Moreover, our method achieves a 4.5% improvement in AP@0.5 and a 7.8% improvement in AP@0.7 compared with DiscoNet [20].

TABLE I
COMPARISONS WITH PREVIOUS METHODS ON V2XSET DATASET. WE SHOW THE AVERAGE PRECISION (AP) AT IOU=0.5, AND 0.7 ON PERFECT AND NOISY SETTINGS, RESPECTIVELY. ALL NUMBER ARE IN %.

Models	Perfect		Noisy	
	AP@0.5	A@P0.7	AP@0.5	AP@0.7
NoFusion	60.6	40.2	60.6	40.2
LateFusion	72.7	62.0	54.9	30.7
EarlyFusion	81.9	71.0	72.0	38.4
F-Cooper [18]	84.0	68.0	71.5	46.9
OPV2V [28]	80.7	66.4	70.9	48.7
V2VNet [19]	84.5	67.7	79.1	49.3
DiscoNet [20]	84.4	69.5	79.8	54.1
V2X-ViT [25]	88.2	71.2	83.6	61.4
Ours	**88.9**	**71.5**	**84.3**	**61.9**

Fig. 5 shows some qualitative detection results. It can be observed that the proposed method can localize objects well, even for faraway objects. For example, in the third image of first row, although the point clouds on the far left are quite sparse, the objects can be detected precisely using the proposed method. The same conditions can also be seen in the last two images of the second row. The vehicles can be detected accurately even though it is far way from the ego vehicle. These demonstrate the effectiveness of our proposed method.

IV. CONCLUSION

In this paper, we propose a multi-model virtual-real fusion Transformer for collaborative perception. The multi-model virtual-real fusion method is adopted to generate the virtual points on sparse locations, to possess the complementary information from both RGB images and LiDAR point clouds.

We propose the heterogeneous graph attention network to capture the inter-agent interaction and adaptively aggregate multiple agents' features. The aggregated feature is further leveraged for classification and bounding box regression. The significance and superiority of our method are validated with extensive experiments as compared with the state-of-the-art methods.

References

[1] G. Luo et al., ”Software-Defined Cooperative Data Sharing in Edge Computing Assisted 5G-VANET,” in IEEE Transactions on Mobile Computing, 2021, pp. 1212-1229.

[2] G. Luo, Q. Yuan, J. Li, S. Wang and F. Yang, ”Artificial Intelligence Powered Mobile Networks: From Cognition to Decision,” in IEEE Network, 2022, pp. 136-144.

[3] Abdelkader G, Elgazzar K, Khamis A. “Connected vehicles: Technology review, state of the art, challenges and opportunities.” Sensors, 2021, 21(22): 7712.

[4] Rios-Torres J, Malikopoulos A A. “A survey on the coordination of connected and automated vehicles at intersections and merging at highway on-ramps.” IEEE Transactions on Intelligent Transportation Systems, 2016, 18(5): 1066-1077.

[5] H. Zhang, Y. Tian, K. Wang, W. Zhang and F. -Y. Wang, ”Mask SSD: An Effective Single-Stage Approach to Object Instance Segmentation,” in IEEE Transactions on Image Processing, 2020, pp. 2078-2093.

[6] H. Zhang, K. Wang, Y. Tian, C. Gou and F. -Y. Wang, ”MFR-CNN: Incorporating Multi-Scale Features and Global Information for Traffic Object Detection,” in IEEE Transactions on Vehicular Technology, 2018, pp. 8019-8030.

[7] G. Luo, H. Zhang, Q. Yuan, J. Li and F. -Y. Wang, ”ESTNet: Embedded Spatial-Temporal Network for Modeling Traffic Flow Dynamics,” in IEEE Transactions on Intelligent Transportation Systems, 2022.

[8] Yi Z, Dan-ya Yao, Li L, et al. “Technologies and Applications for Intelligent Vehicle-infrastructure Cooperation Systems.” Journal of Transportation Systems Engineering and Information Technology, 2021, 21(5): 40.

[9] National Transportation Safety Board Office of Highway Safety Washington, D.C., Vehicle Automation Report. [Online]. Available: https://www.documentcloud.org/documents/6540547-629713.html.

[10] H. Zhang, G. Luo, J. Li and F. -Y. Wang, ”C2FDA: Coarse-to-Fine Domain Adaptation for Traffic Object Detection,” in IEEE Transactions on Intelligent Transportation Systems, 2022, pp. 12633-12647.

[11] Guiyang Luo, Hui Zhang, Quan Yuan, Jinglin Li. ” Complementarity-Enhanced and Redundancy-Minimized Collaboration Network for Multi-agent Perception,” ACM Multimedia, 2022.

[12] Chen Q, Tang S, Yang Q, et al. “Cooper: Cooperative perception for connected autonomous vehicles based on 3D point clouds.” IEEE 39th International Conference on Distributed Computing Systems. IEEE, 2019, pp. 514-524.

[13] Rauch A, Klanner F, Rasshofer R, et al. “Car2x-based perception in a high-level fusion architecture for cooperative perception systems.” IEEE Intelligent Vehicles Symposium. IEEE, 2012, pp. 270-275.

[14] Rawashdeh Z Y, Wang Z. “Collaborative automated driving: A machine learning-based method to enhance the accuracy of shared information.” International Conference on Intelligent Transportation Systems. IEEE, 2018, pp. 3961-3966.

[15] Vadivelu N, Ren M, Tu J, et al.“ Learning to communicate and correct pose errors.” arXiv preprint arXiv:2011.05289, 2020.

[16] Liu Y C, Tian J, Ma C Y, et al. “Who2com: Collaborative perception via learnable handshake communication.” IEEE International Conference on Robotics and Automation. IEEE, 2020, pp. 6876-6883.

[17] Liu Y C, Tian J, Glaser N, et al. “When2com: Multi-agent perception via communication graph grouping.” IEEE/CVF Conference on computer vision and pattern recognition. 2020, pp. 4106-4115.

[18] Chen Q, Ma X, Tang S, et al. “F-Cooper: Feature based cooperative perception for autonomous vehicle edge computing system using 3D point clouds.” ACM/IEEE Symposium on Edge Computing. 2019, pp. 88-100.

[19] Wang T H, Manivasagam S, Liang M, et al. “V2VNet: Vehicle-to-vehicle communication for joint perception and prediction.” European Conference on Computer Vision. Springer, Cham, 2020, pp. 605-621.

[20] Li Y, Ren S, Wu P, et al. “Learning distilled collaboration graph for multi-agent perception.” Advances in Neural Information Processing Systems, 2021, pp. 29541-29552.

[21] Duan K, Bai S, Xie L, et al. “Centernet: Keypoint triplets for object detection.” IEEE/CVF international conference on computer vision. 2019, pp. 6569-6578.

[22] Cai Z, Vasconcelos N. “Cascade R-CNN: Delving into high quality object detection.” IEEE conference on computer vision and pattern recognition. 2018, pp. 6154-6162.

[23] Lang A H, Vora S, Caesar H, et al. “Pointpillars: Fast encoders for object detection from point clouds.” IEEE/CVF conference on computer vision and pattern recognition. 2019, pp. 12697-12705.

[24] Zhang H, Wu C, Zhang Z, et al. “ResNeSt: Split-attention networks.” IEEE/CVF conference on computer vision and pattern recognition. 2022, pp. 2736-2746.

[25] Xu R, Xiang H, Tu Z, et al. “V2X-ViT: Vehicle-to-everything cooperative perception with vision transformer.” arXiv preprint arXiv:2203.10638, 2022.

[26] Dosovitskiy A, Ros G, Codevilla F, et al. “CARLA: An open urban driving simulator.” Conference on robot learning. PMLR, 2017, pp. 1-16.

[27] Xu R, Guo Y, Han X, et al. “OpenCDA: an open cooperative driving automation framework integrated with co-simulation.” IEEE International Intelligent Transportation Systems Conference. IEEE, 2021, pp. 1155-1162.

[28] Xu R, Xiang H, Xia X, et al. “OPV2V: An open benchmark dataset and fusion pipeline for perception with vehicle-to-vehicle communication.” International Conference on Robotics and Automation. IEEE, 2022, pp. 2583-2589.