

# S2R-ViT for Multi-Agent Cooperative Perception: Bridging the Gap from Simulation to Reality

Jinlong Li<sup>1</sup>, Runsheng Xu<sup>2</sup>, Xinyu Liu<sup>1</sup>, Baolu Li<sup>1</sup>, Qin Zou<sup>3</sup>, Jiaqi Ma<sup>2</sup>, Hongkai Yu<sup>1\*</sup>

**Abstract**—Due to the lack of enough real multi-agent data and time-consuming of labeling, existing multi-agent cooperative perception algorithms usually select the simulated sensor data for training and validating. However, the perception performance is degraded when these simulation-trained models are deployed to the real world, due to the significant domain gap between the simulated and real data. In this paper, we propose the *first* Simulation-to-Reality transfer learning framework for multi-agent cooperative perception using a novel Vision Transformer, named as S2R-ViT, which considers both the Deployment Gap and Feature Gap between simulated and real data. We investigate the effects of these two types of domain gaps and propose a novel uncertainty-aware vision transformer to effectively relieve the Deployment Gap and an agent-based feature adaptation module with inter-agent and ego-agent discriminators to reduce the Feature Gap. Our intensive experiments on the public multi-agent cooperative perception datasets OPV2V and V2V4Real demonstrate that the proposed S2R-ViT can effectively bridge the gap from simulation to reality and outperform other methods significantly for point cloud-based 3D object detection.

## I. INTRODUCTION

The recent advancement in multi-agent cooperative perception shows potentials to overcome the limitation of single-agent perception that suffers from the challenges of perceiving range and occlusion [1], [2]. By leveraging agent-to-agent communication technology to share information, multi-agent cooperative perception systems can significantly enhance perception performance compared to the single-agent perception [3], [4]. Due to the difficulties of collecting multi-agent data with communication in the real world, it is expensive and not easy to collect enough real data in diverse and complex real-world environments [2]. Furthermore, the ground-truth data labeling and uniform coordinate projection for multi-agent cooperative perception systems is particularly time-consuming. Therefore, many existing multi-agent cooperative perception research works usually select the simulated data for model training and validating [3], [4].

However, when we apply the models trained with simulated data to the real world, the perception performance is typically degraded. This phenomenon is because of the significant domain gap between the simulated and real data. In this paper, our research is focused on utilizing labeled simulated data and unlabeled real-world data as transfer learning to reduce the domain gap for multi-agent cooperative perception. We observe that the domain gap from

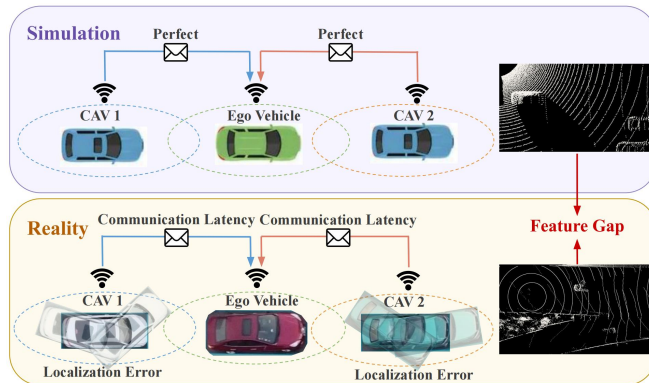


Fig. 1: Illustration of the domain gap (*Deployment Gap*, *Feature Gap*) for multi-agent cooperative perception from simulation to reality. Here we use Vehicle-to-Vehicle (V2V) cooperative perception in autonomous driving as example. CAV indicates the Connected Autonomous Vehicles.

simulation to reality for multi-agent cooperative perception includes the following two perspectives.

- **Deployment Gap:** As shown in Fig. 1, different with the ideal simulation setting, the multiple agents might have localization (positional and heading) errors due to the unavoidable GPS errors and communication latency (time delay) during the real-world agent-to-agent communication.
- **Feature Gap:** As illustrated in Fig. 1, the point cloud feature distribution in real world might differ significantly with that of the simulated data, such as more complex driving scenarios, different LiDAR channel numbers, mixed traffic flow, various point cloud variations, and so on.

In this paper, we propose the first Simulation-to-Reality (S2R) transfer learning framework for multi-agent cooperative perception using a novel Vision Transformer (ViT), named as S2R-ViT, by taking both Deployment Gap and Feature Gap into consideration. We choose the task of Vehicle-to-Vehicle (V2V) Cooperative Perception for the point cloud-based 3D object detection as algorithm development. Specifically, our framework takes the labeled point cloud data from simulation and the unlabeled data from real world as input, so as to largely utilize the simulated data. In machine learning research, this setting is widely called as Unsupervised Domain Adaptation [5] from source domain (simulation) to target domain (reality).

The proposed S2R-ViT comprises two key components:

<sup>1</sup>Cleveland State University, Cleveland Vision & AI Lab. <sup>2</sup>University of California, Los Angeles, UCLA Mobility Lab. <sup>3</sup>Wuhan University, MVR Lab.

\*Corresponding Author: h.yu19@csuohio.edu

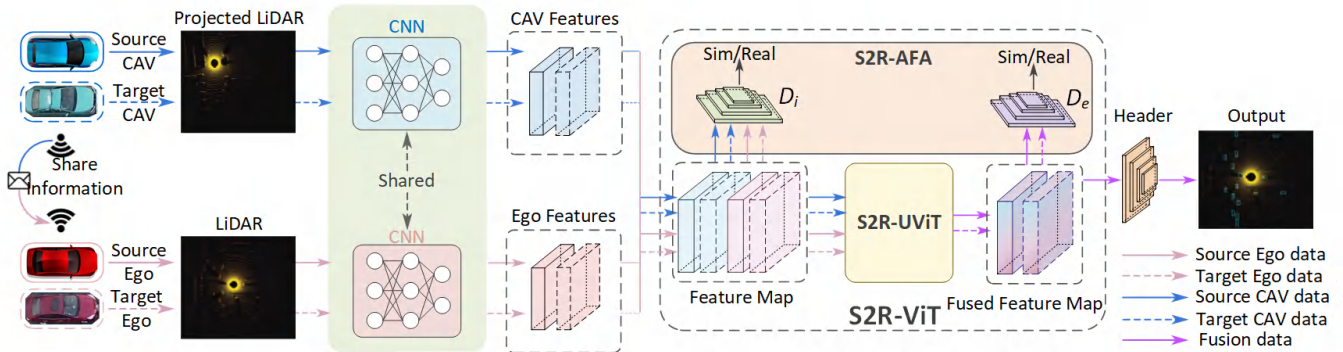


Fig. 2: Overview of the proposed **S2R-ViT** for multi-agent cooperative perception from simulation to reality, which leverages *S2R-UViT* module to handle uncertainty in Deployment Gap and tackles the Feature Gap through *S2R-AFA* module including the inter-agent discriminator  $D_i$  and the ego-agent discriminator  $D_e$ . Source Domain: labeled simulated data, Target Domain: unlabeled real-world data. Best viewed in color.

(1) *S2R-UViT*: a novel S2R Uncertainty-aware Vision Transformer to effectively relieve the *uncertainties brought by the Deployment Gap*. Specifically, *S2R-UViT* includes a Local-and-Global Multi-head Self Attention (LG-MSA) module to enhance feature interactions across all agents' spatial positions to tolerate the uncertainty drawbacks and also a Uncertainty-Aware Module (UAM) to enhance the ego-agent features by considering the shared other-agent features of different uncertainty levels. (2) *S2R-AFA*: S2R Agent-based Feature Adaptation to reduce the *Feature Gap*. *S2R-AFA* utilizes the inter-agent and ego-agent discriminators to extract domain-invariant features to bridge the Feature Gap. Finally, we conducted extensive experiments on two public datasets, namely simulated OPV2V [4] and real V2V4Real [2], to justify the effectiveness of our proposed method. Our contributions are summarized as follows.

- To the best of our knowledge, we propose the **first research**, named *S2R-ViT*, on multi-agent cooperative perception from simulation to reality by investigating two types of domain gaps, *i.e.*, Deployment Gap and Feature Gap, for point cloud-based 3D object detection.
- We propose a novel Uncertainty-aware Vision Transformer (*S2R-UViT*) to effectively relieve the uncertainties brought by the Deployment Gap.
- We design an Agent-based Feature Adaptation (*S2R-AFA*) module that includes inter-agent and ego-agent discriminators to effectively reduce the Feature Gap between simulation and reality.
- We evaluate our proposed method on the large-scale simulated OPV2V dataset and the real V2V4Real dataset, whose experiments demonstrate our superior performance in point cloud-based 3D object detection.

## II. RELATED WORK

**Multi-Agent Perception.** Multi-agent perception system can overcome the challenges of occlusion and short-range perceiving via agent-to-agent communication technology to achieve the large-range perceiving, which has attracted the attention of many researchers. Rather than exchanging raw

sensing data or detected outputs, contemporary approaches typically share intermediate features extracted by neural networks. This strategy offers an optimal balance between accuracy and bandwidth requirements [1], [6]–[8]. V2VNet [3] utilizes a graph neural network to aggregate features from individual agents. When2com [8] employs a spatial confidence-aware communication strategy aimed at enhancing performance by reducing communication overhead. OPV2V [4] proposes an attention fusion framework for cooperative perception system. SyncNet [9] introduces a latency compensation module for time-domain synchronization. V2X-ViT [6] and CoBEVT [10] propose transformer or axial attention based methods to improve the performance. Although these methods have demonstrated impressive performance, all of them are implemented in the simulated data, while this paper aims to bridge the gap from simulation to reality.

**Challenges in Multi-Agent Perception.** Multi-Agent perception system also introduces some new challenges, *e.g.*, localization error, communication latency, adversarial attacks. These challenges might diminish the benefits of collaborations [6], [7], [11]. To ensure robustness for Multi-Agent perception, V2X-ViT [6] proposes a ViT based multi-agent perception system, showcasing robust performance under communication delay. [12] introduces a pose regression module and consistency module prior to feature aggregation to rectify pose errors. [8] presents the first latency-aware multi-agent perception framework, achieving feature-level synchronization. [13] explores adversarial attacks in multi-agent perception and devises a novel transfer attack approach within the intermediate collaborative perception. [11] proposes a lossy communication-aware repair network to ensure the robustness of multi-agent perception amidst lossy communication data. To promote multi-agent perception research, several large-scale cooperative perception datasets are publicized, *e.g.*, simulation based OPV2V [4] and V2XSet [6], real-world V2V4Real [2]. Because collecting the multi-agent sensing data in the wild and annotating its ground truth are time and labor consuming, many existing research works are based on simulated data. In this paper, we

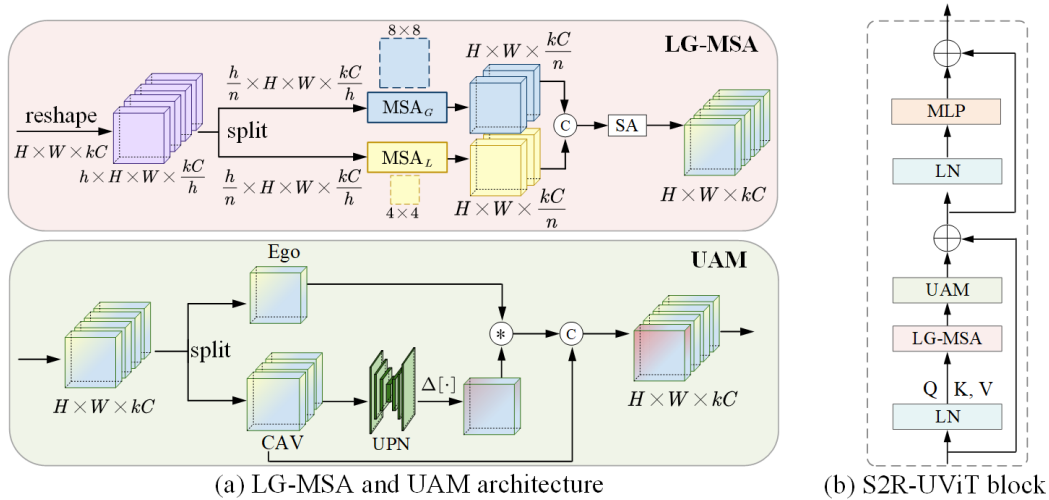


Fig. 3: Architecture of the proposed S2R-UViT: Simulation-to-Reality Uncertainty-aware Vision Transformer.

focus on improving point cloud-based cooperative 3D object detection in real world by largely utilizing labelled simulated data and just unlabelled real-world data.

**Domain Adaptation for Perception.** Domain adaptation involves adapting the machine learning model trained on source domain to the target domain. Many domain adaptation works are mainly focused on the RGB camera data [14]–[17], while more domain adaptation works have been proposed to solve this problem in LiDAR data [7], [18]–[20]. Specifically, Semantic Point Generation (SPG) [19] is presented to enhance the reliability of LiDAR detectors against domain gaps to generates semantic points at the 3D objects. CoS-Mix [20] is designed for 3D LiDAR segmentation, aiming to alleviate the domain gap by generating two intermediate domains comprising composite point clouds. This is achieved through the application of a special mixing strategy at the input level. [7] proposes a domain adaption framework to bridge the domain gap of shared data in communication for collaborative perception. In this paper, an agent-based feature adaptation module are proposed to reduce the feature gap between simulated to real data for multi-agent perception.

### III. METHODOLOGY

As illustrated in Fig. 2, the proposed S2R work is an end-to-end unified deep learning pipeline, including 1) V2V metadata sharing, 2) Feature extraction and sharing, 3) S2R-ViT, and 4) Detection header.

#### A. Overview of Architecture

**1) V2V metadata sharing.** We designate one of the CAVs as the ego vehicle and construct a spatial graph around it, with each node representing a CAV within the communication range. After receiving the ego vehicle’s relative pose and extrinsic information, all other nearby CAVs then project their LiDAR data to the coordinate frame of the ego vehicle. **2) Feature extraction and sharing.** We utilize the Point-Pillar backbone [21] to extract intermediate features from LiDAR data due to its low inference latency and refined

memory usage. Each CAV possesses its own feature extraction module. Following feature extraction by each CAV, the ego vehicle receives the visual features of neighboring CAVs via communication. **3) S2R-ViT.** The intermediate features aggregated from other surrounding CAVs are fed into our major component named S2R-ViT, which consists of S2R-UViT and S2R-AFA modules. Upon receiving the final fused features, we utilize a prediction header for 3D object classification and localization.

#### B. S2R-UViT: Simulation-to-Reality Uncertainty-aware Vision Transformer

The Deployment Gap from simulation to reality brings different uncertainties to both ego and neighboring agents, *e.g.*, spatial bias by GPS errors, spatial misalignment in the coordinate projection because of communication latency. *How to effectively reduce the degradation effects of these uncertainty drawbacks is an essential problem and open question to the S2R multi-agent perception research.* In this paper, we propose to answer this question from two perspectives: uncertainties can be relieved by enhancing (1) the feature interactions across all agents’ spatial positions more comprehensively and (2) the ego-agent features by considering the shared other-agent features of different uncertainty levels. These two perspectives motivate us to develop the novel Local-and-Global Multi-head Self Attention (LG-MSA) Module and Uncertainty-Aware Module (UAM) respectively. The Fig. 3 presents the two major modules of the proposed S2R-UViT.

**1) Local-and-Global Multi-head Self Attention (LG-MSA):** In order to enhance the feature interactions across all agents’ spatial positions more comprehensively, we propose LG-MSA to promote both local and global feature interactions across all agents’ spatial positions. In the proposed LG-MSA, a local feature-based attention is utilized to focus on the local details of spatial features, while a global feature-based attention is used to pay attention on the wide range of spatial features. Specifically, the fused features of the

ego agent and other agents are spilt into two branches (*i.e.*, local branch and global branch) to process local-based and global-based attentions, respectively, then these two features are concatenated together and fed into a Self Attention (SA) module to further capture local and global information.

Inspired by [22], [23], after retrieving the whole features  $F_{e,o} \in \mathbb{R}^{H \times W \times kC}$  with spatial dimension  $H \times W$  from all agents ( $e$ : ego-agent,  $o$ : other agents,  $k$ : number of all agents), we reshape it into  $\mathbb{R}^{h \times H \times W \times \frac{kC}{h}}$ . After that, we take  $h = 8$  as head number and  $n = 2$  as window-type number, the multi-head  $h$  of standard multi-head Self Attention Module (MSA) [24] is evenly divided into two groups with different window sizes, *i.e.*,  $4 \times 4$  for the local branch and  $8 \times 8$  for the global branch. In the *local branch*, the spilt feature  $F_{e,o}^l \in \mathbb{R}^{\frac{h}{n} \times H \times W \times \frac{kC}{h}}$  is fed into the  $MSA_L$  with a small window size  $4 \times 4$  to enhance the local details of spatial features. In the *global branch*, another spilt feature  $F_{e,o}^g \in \mathbb{R}^{\frac{h}{n} \times H \times W \times \frac{kC}{h}}$  is fed into the  $MSA_G$  with a large window size  $8 \times 8$  to capture global spatial feature information. Then we concatenate these two-branch output features and implement the feature interactions by a Self Attention (SA) module to obtain the promoted feature  $F_{e,o}^p \in \mathbb{R}^{H \times W \times kC}$ . As show in Fig. 3(a), the proposed LG-MSA computation can be defined as

$$F_{e,o}^p = \text{SA}(\text{Concat}(MSA_L(F_{e,o}^l), MSA_G(F_{e,o}^g))). \quad (1)$$

2) *Uncertainty-Aware Module (UAM)*: To enhance the ego-agent feature based on the shared other-agent features, their different uncertainty levels should not be neglected. When receiving the feature  $F_{e,o}^p$  for all agents, we split them into ego-agent feature  $F_e^p \in \mathbb{R}^{H \times W \times C}$  and other agent shared feature  $F_o^p \in \mathbb{R}^{H \times W \times (k-1)C}$ . The shared other-agent feature  $F_o^p$  will be fed into an Uncertainty Prediction Network (UPN) to predict the uncertainty levels on spatial features, which generates a uncertainty-level map  $M \in \mathbb{R}^{H \times W \times (k-1)C}$  with the same spatial size of input. The UPN is an encoder-decoder based neural network simplified from [25], which can be learned during the end-to-end training of our whole architecture. Inspired by the natural selection in the mechanism of evolution, the feature values in the predicted uncertainty-level map  $M$  with high uncertainty levels (*i.e.*, low confidences) are reset as 1 using the median as threshold. It results in a new uncertainty-level map  $M_t$ . Then,  $M_t$  is multiplied with  $F_e^p$ , by taking the shared other-agent features of different uncertainty levels into consideration, so as to produce the enhanced ego feature. In other words, only the low-uncertainty spatial features of other agents (related to non-ones in  $M_t$ ) will contribute to enhance the ego-agent feature during the matrix multiplication. Finally, the enhanced ego feature and other agent shared feature are concatenated together to obtain the combined feature  $F_{e,o}^h \in \mathbb{R}^{H \times W \times kC}$ . As show in Fig. 3(a), the computation of the proposed UAM can be formulated as

$$F_{e,o}^h = \text{Concat}(\Delta[\text{UPN}(F_o^p)] \otimes F_e^p, F_o^p), \quad (2)$$

where  $\Delta[\cdot]$  represents the threshold process and  $\otimes$  denotes the matrix dot product. Combining these local and global attention branches with standard Transformer designs [22], [23], such as Layer Normalization (LN), MLPs, and skip-connections, results in our proposed S2R-UViT block, as depicted in Fig. 3(b). The S2R-UViT block can be expressed as:

$$F_{e,o}^h = \text{S2RAttn}(\text{LN}(F_{e,o})) + F_{e,o}, \quad (3)$$

$$\hat{F}_{e,o}^h = \text{MLP}(\text{LN}(F_{e,o}^h)) + F_{e,o}^h, \quad (4)$$

where  $F_{e,o}^h$  and  $\hat{F}_{e,o}^h$  denote the output features of our S2RAttn module (*i.e.*, the proposed LG-MSA and UAM) and MLP module.

### C. S2R-AFA: Simulation-to-Reality Agent-based Feature Adaptation

To reduce the Feature Gap from simulated feature  $F_s$  to real feature  $F_r$  within both inter-agent and ego-agent features, we design two domain discriminators/classifiers before and after fusion as shown in Fig. 2, where all agent feature  $F_s$  and  $F_r$  before fusion are fed into the inter-agent discriminator  $D_i$  to classify whether it belongs to simulation or reality, and the fused ego features  $F_s^e$  and  $F_r^e$  after fusion are also classified into simulation or reality by the ego-agent discriminator  $D_e$ . The binary cross entropy loss is used for binary domain (Sim/Real) classification. These two discriminators are adversarially optimized by the Agent-based Feature Adaptation loss  $\mathcal{L}_{AFA}$ :

$$\min_{G_m} \max_{D_i, D_e} \mathcal{L}_{AFA} = \mathbb{E}_{s,r}[D_i(F_s, F_r)] + \mathbb{E}_{s,r}[D_e(F_s^e, F_r^e)], \quad (5)$$

where  $\mathbb{E}_{s,r}$  indicates the domain classification error in simulation and reality respectively,  $G_m$  is our whole model (backbone, S2R-UViT, and detection header) which can be thought as the generator of Generative Adversarial Network [26]. Because of S2R-AFA, our generator model  $G_m$  will have the capability of extracting domain-invariant features of simulation and reality.

We use the focal loss [27] and smooth  $L_1$  loss as the detection loss for our model training. Our final loss is the combination of detection loss and the Agent-based Feature Adaptation loss as follows,

$$\mathcal{L}_{total} = w_1 \mathcal{L}_{det} + w_2 \mathcal{L}_{AFA}, \quad (6)$$

where  $w_1$  and  $w_2$  are balance weights with sum as 1.

## IV. EXPERIMENT

### A. Dataset

We conduct experiments on two public benchmark datasets (OPV2V [4], V2V4Real [2]) for the V2V cooperative perception task. **OPV2V** is a large-scale *simulated* dataset for V2V cooperative perception, which is collected by CARLA simulator and OpenCDA [28]. It contains 73 divergent scenes with various numbers of connected vehicles ([1, 5]), and its training/validation/testing set is split into 6,764, 1,981,

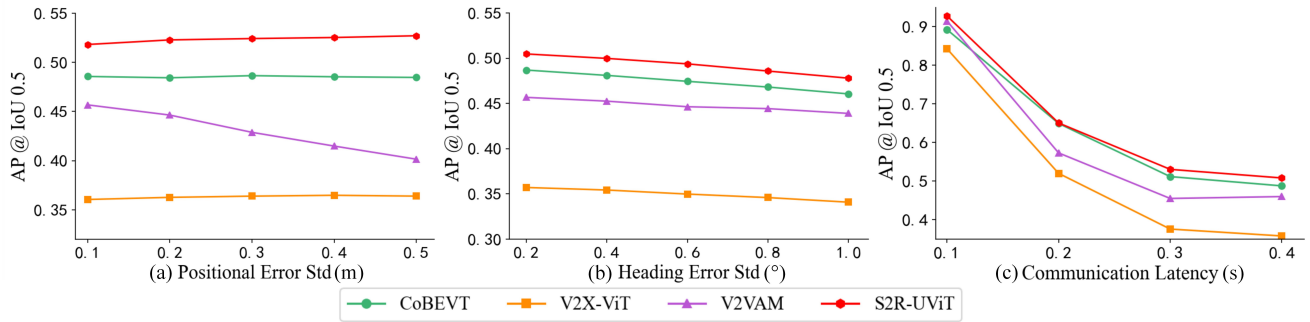


Fig. 4: Robustness in *Deployment-Gap Scenario* including GPS (positional, heading) errors and communication latency on the CARLA Towns testing set of OPV2V [4] with the simulated *Noisy Setting*.

TABLE I: 3D detection performance on two OPV2V testing sets under *Deployment-Gap Scenario* with *Perfect Setting* and *Noisy Setting*. All methods are trained on *Perfect Setting*.

Models	Setting	V2V CARLA Towns		V2V Culver City	
		AP@0.5	AP@0.7	AP@0.5	AP@0.7
AttFuse [4]	Perfect	0.921	0.804	0.887	0.716
	Noisy	0.851	0.472	0.865	0.565
V2VNet [3]	Perfect	0.915	0.828	0.884	0.757
	Noisy	0.845	0.413	0.868	0.617
F-cooper [1]	Perfect	0.907	0.810	0.893	0.746
	Noisy	0.842	0.479	0.881	0.624
V2X-ViT [6]	Perfect	0.902	0.792	0.903	0.764
	Noisy	0.801	0.395	0.882	0.606
CoBEVT [10]	Perfect	0.925	0.852	0.904	0.776
	Noisy	0.862	0.519	0.889	0.665
V2VAM [11]	Perfect	0.916	0.849	0.903	<b>0.794</b>
	Noisy	0.876	0.507	0.883	0.663
S2R-UViT	Perfect	<b>0.928</b>	<b>0.867</b>	<b>0.912</b>	0.768
	Noisy	<b>0.879</b>	<b>0.579</b>	<b>0.900</b>	<b>0.681</b>

and 2,719 frames, respectively. **V2V4Real** is a *real-world*, large-scale dataset with diverse driving scenarios, which is collected by two CAVs driving simultaneously in Columbus, Ohio, USA. It’s split into the train/validation/test set with 14,210/2,000/3,986 frames, respectively.

### B. Experiments Setup

**Evaluation Metrics.** We assess the perception performance based on the final 3D vehicle detection accuracy. Following [4], [6], we define the evaluation range as  $x \in [-140, 140]$  meters and  $y \in [-40, 40]$  meters, encompassing all CAVs within this spatial range in the experiment. Accuracy is calculated as Average Precision (AP) at two Intersection-over-Union (IoU) thresholds (0.5 and 0.7).

**Experiment Settings.** In this work, we want to address the deployment gap and feature gap on LiDAR-based object detection and assess our model under two distinct settings:

- 1) **Deployment-Gap Scenario:** During the training, all models are trained in the perfect simulated OPV2V training set, then all of them are evaluated on OPV2V CARLA Towns and Culver City testing sets under two different setting (e.g., *Perfect* and *Noisy*), respectively.

We implement the *Noisy Setting* following [6] to simulate the GPS errors, the positional and heading noises of the transmitter are sampled from a Gaussian distribution with a standard deviation of 0.2 m and 0.2° respectively, and the communication latency is set as 100 ms for all the evaluated models. This *Deployment-Gap Scenario* only includes the *Deployment Gap*.

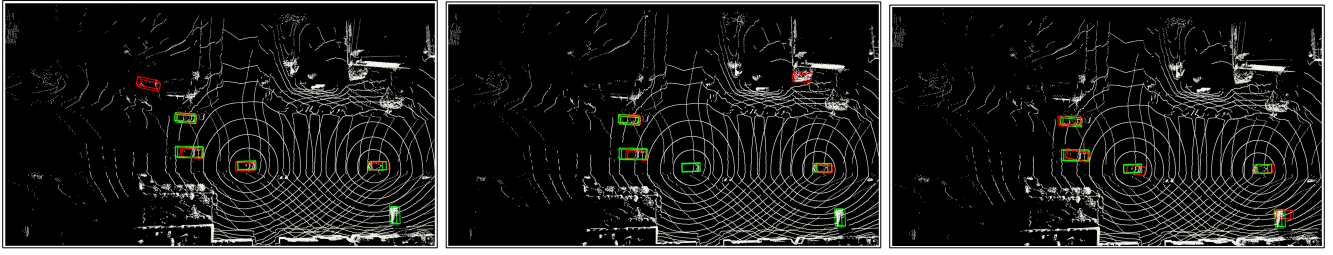
- 2) **Sim2Real Scenario:** We set the labeled training set of simulation dataset OPV2V [4] as the source domain and the unlabeled training set of the real-world dataset V2V4Real [2] as the target domain for all models during the training, by following the same setting in [2]. Then, all trained models are evaluated on the testing set of V2V4Real to report the performance. This *Sim2Real Scenario* includes both the *Deployment Gap* and *Feature Gap* from simulation to reality.

All methods utilize PointPillar [29] as the point cloud encoder/backbone. We employ the Adam optimizer [30] with an initial learning rate of  $10^{-3}$  and gradually decay it every 10 epochs based a factor of 0.1. Following the hyperparameters outlined in V2X-ViT [6], we train all models on two RTX 3090 GPU cards.

**Compared Methods.** Six state-of-the-art V2V methods are evaluated, all of which employ *Intermediate Fusion* as the primary strategy: AttFuse [4], V2VNet [3], F-Cooper [1], V2X-ViT [6], CoBEVT [10], and V2VAM [11]. We first train these methods on the perfect setting of OPV2V training set, and then these methods are evaluated on the *Noisy Setting* of OPV2V testing set and V2V4Real testing set to assess their performance. In addition, to show the effectiveness of reducing *Feature Gap*, two domain adaptation methods *i.e.*, gradient reverse layer (GRL) [31] and adversarial gradient reverse layer (AdvGRL) [15] are utilized to back-propagate the reversed gradient to adversarially guide the model for generating domain-invariant features by one feature-level domain classifier (after fusion) and one object/proposal-level domain classifier (in detection header).

### C. Quantitative Evaluation

**Performance in Deployment-Gap Scenario.** Table I shows the performance comparison on *Deployment-Gap Scenario*, where all methods are evaluated on *Perfect Setting* and *Noisy*



(a) V2X-ViT w/ AdvGRL

(b) V2VAM w/ AdvGRL

(c) S2R-ViT

Fig. 5: Visualization example of point cloud-based 3D object detection on V2V4Real testing set under the *Sim2Real Scenario*. Green and red 3D bounding boxes represent ground truth and prediction respectively. Best viewed in color.

TABLE II: 3D detection performance on V2V4Real testing set under *Sim2Real Scenario*. All methods with domain adaptation are trained following the setting in *Sim2Real Scenario*. S2R-UViT w/ S2R-AFA indicates our S2R-ViT.

Method	V2V4Real Testing	
	AP@0.5	AP@0.7
AttFuse [4]	0.225	0.094
AttFuse w/ GRL	0.356	0.139
AttFuse w/ AdvGRL	0.366	0.137
V2VNet [3]	0.268	0.108
V2VNet w/ GRL	0.376	0.122
V2VNet w/ AdvGRL	0.358	0.103
F-Cooper [1]	0.236	0.091
F-Cooper w/ GRL	0.372	0.116
F-Cooper w/ AdvGRL	0.364	0.135
V2X-ViT [6]	0.274	0.103
V2X-ViT w/ GRL	0.395	0.156
V2X-ViT w/ AdvGRL	0.404	0.157
CoBEVT [10]	0.334	0.130
CoBEVT w/ GRL	0.398	0.163
CoBEVT w/ AdvGRL	0.393	0.163
V2VAM [11]	0.332	0.120
V2VAM w/ GRL	0.390	0.146
V2VAM w/ AdvGRL	0.401	0.161
S2R-UViT	<b>0.367</b>	<b>0.138</b>
S2R-UViT w/ GRL	0.414	0.141
S2R-UViT w/ AdvGRL	0.414	0.157
S2R-UViT w/ S2R-AFA	<b>0.441</b>	<b>0.170</b>

setting, respectively. Under the *Perfect Setting*, all cooperative perception methods achieve outstanding performance. Nevertheless, when these methods deployed on the *Noisy Setting* scenario, which have the deployment gap with the *Perfect Setting* scenario, the V2X-ViT [6], CoBEVT [10], and V2VAM [11] drop 39.7%, 33.3%, and 34.2% on for AP@0.7 on V2V CARLA Towns testing set. It indicates the highly negative impacts by V2V deployment gap. While our proposed S2R-UViT achieves the mostly best performance under both *Perfect Setting* and *Noisy Setting* scenarios, which is highlighted in Table I. To assess the models' sensitivity

on different *Deployment-Gap Scenario*, we conduct the experiments on the V2V CARLA Towns testing set of OPV2V dataset with the difference of *Noisy Setting*. The Fig. 4 depicts the higher robustness of our S2R-UViT against other fusion methods in the deployment gap.

**Performance in Sim2Real Scenario.** The 3D object detection result on the real-world V2V4Real testing set of *Sim2Real Scenario* is presented in Table II. Among all intermediate fusion methods without domain adaptation, our proposed S2R-UViT achieves the best performance by eliminating the deployment gap in the real V2VReal testing set. After applying domain adaptation methods *e.g.*, GRL [31] and AdvGRL [15], all methods have improved performance. For example, V2X-ViT is improved by 12.1%/5.3% for AP@0.5/0.7 with GRL, and 13.0%/5.4% for AP@0.5/0.7 with AdvGRL. Our S2R-ViT with S2R-AFA achieves the 44.1%/17.0% for AP@0.5/0.7 as the best performance with improvement of 7.4%/3.2% than S2R-UViT. The examples of 3D object detection results on V2V4Real testing set under *Sim2Real Scenario* are visualized in Fig. 5, where our S2R-ViT generates more accurate 3D detection results.

**Ablation Study.** As Table II depicts, all of the proposed components within S2R-ViT have contributed to enhanced detection performance. Specifically, our S2R-UViT model achieves the highest detection performance among all state-of-the-art V2V methods, which is 3.3% and 0.8% higher than the second-best performance model CoBEVT in AP@0.5 and AP@0.7 respectively. Adding our S2R-AFA, our proposed S2R-ViT achieves 44.1% and 17.0% in AP@0.5 and AP@0.7, with the improvement of 7.4% in AP@0.5 and 3.2% in AP@0.7.

## V. CONCLUSIONS

This paper is the first work that investigates the domain gap on multi-agent cooperation perception from simulation to reality, specifically focusing on the deployment gap and feature gap in point cloud-based 3D object detection. Based on the analysis, we present the first Simulation-to-Reality transfer learning framework using a novel Vision Transformer, named S2R-ViT, to mitigate these two types of domain gaps, which mainly contain an Uncertainty-aware Vision Transformer and an Agent-based Feature Adaptation module. The experiment shows the effectiveness of S2R-ViT. This research presents a significant step forward in the multi-agent cooperation perception from simulation to reality.

## REFERENCES

- [1] 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 *ACM/IEEE Symposium on Edge Computing*, 2019, pp. 88–100.
- [2] R. Xu, X. Xia, J. Li, H. Li, S. Zhang, Z. Tu, Z. Meng, H. Xiang, X. Dong, R. Song, *et al.*, “V2v4real: A real-world large-scale dataset for vehicle-to-vehicle cooperative perception,” in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 13 712–13 722.
- [3] T.-H. Wang, S. Manivasagam, M. Liang, B. Yang, W. Zeng, and R. Urtasun, “V2vnet: Vehicle-to-vehicle communication for joint perception and prediction,” in *European Conference on Computer Vision*. Springer, 2020, pp. 605–621.
- [4] R. Xu, H. Xiang, X. Xia, X. Han, J. Li, and J. Ma, “Opv2v: An open benchmark dataset and fusion pipeline for perception with vehicle-to-vehicle communication,” in *International Conference on Robotics and Automation*. IEEE, 2022, pp. 2583–2589.
- [5] S. Song, H. Yu, Z. Miao, J. Fang, K. Zheng, C. Ma, and S. Wang, “Multi-spectral salient object detection by adversarial domain adaptation,” in *AAAI Conference on Artificial Intelligence*, vol. 34, no. 07, 2020, pp. 12 023–12 030.
- [6] R. Xu, H. Xiang, Z. Tu, X. Xia, M.-H. Yang, and J. Ma, “V2x-vit: Vehicle-to-everything cooperative perception with vision transformer,” in *European Conference on Computer Vision*. Springer, 2022, pp. 107–124.
- [7] R. Xu, J. Li, X. Dong, H. Yu, and J. Ma, “Bridging the domain gap for multi-agent perception,” in *International Conference on Robotics and Automation*, 2023.
- [8] Y. Hu, S. Fang, Z. Lei, Y. Zhong, and S. Chen, “Where2comm: Communication-efficient collaborative perception via spatial confidence maps,” in *Advances in Neural Information Processing Systems*, 2022.
- [9] Z. Lei, S. Ren, Y. Hu, W. Zhang, and S. Chen, “Latency-aware collaborative perception,” in *European Conference on Computer Vision*. Springer, 2022, pp. 316–332.
- [10] R. Xu, Z. Tu, H. Xiang, W. Shao, B. Zhou, and J. Ma, “Cobevt: Cooperative bird’s eye view semantic segmentation with sparse transformers,” in *Conference on Robot Learning*, 2022.
- [11] J. Li, R. Xu, X. Liu, J. Ma, Z. Chi, J. Ma, and H. Yu, “Learning for vehicle-to-vehicle cooperative perception under lossy communication,” *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 4, pp. 2650–2660, 2023.
- [12] N. Vadivelu, M. Ren, J. Tu, J. Wang, and R. Urtasun, “Learning to communicate and correct pose errors,” in *Conference on Robot Learning*. PMLR, 2021, pp. 1195–1210.
- [13] J. Tu, T. Wang, J. Wang, S. Manivasagam, M. Ren, and R. Urtasun, “Adversarial attacks on multi-agent communication,” in *IEEE International Conference on Computer Vision*, 2021, pp. 7768–7777.
- [14] J. Li, Z. Xu, L. Fu, X. Zhou, and H. Yu, “Domain adaptation from daytime to nighttime: A situation-sensitive vehicle detection and traffic flow parameter estimation framework,” *Transportation Research Part C: Emerging Technologies*, vol. 124, p. 102946, 2021.
- [15] J. Li, R. Xu, J. Ma, Q. Zou, J. Ma, and H. Yu, “Domain adaptive object detection for autonomous driving under foggy weather,” in *IEEE Winter Conference on Applications of Computer Vision*, 2023, pp. 612–622.
- [16] P. Oza, V. A. Sindagi, V. V. Sharmini, and V. M. Patel, “Unsupervised domain adaptation of object detectors: A survey,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.
- [17] L. Hoyer, D. Dai, H. Wang, and L. Van Gool, “Mic: Masked image consistency for context-enhanced domain adaptation,” in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 11 721–11 732.
- [18] L. Yi, B. Gong, and T. Funkhouser, “Complete & label: A domain adaptation approach to semantic segmentation of lidar point clouds,” in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 15 363–15 373.
- [19] Q. Xu, Y. Zhou, W. Wang, C. R. Qi, and D. Anguelov, “Spq: Unsupervised domain adaptation for 3d object detection via semantic point generation,” in *IEEE/CVF International Conference on Computer Vision*, 2021, pp. 15 446–15 456.
- [20] C. Saltori, F. Galasso, G. Fiameni, N. Sebe, E. Ricci, and F. Poiesi, “Cosmix: Compositional semantic mix for domain adaptation in 3d lidar segmentation,” in *European Conference on Computer Vision*. Springer, 2022, pp. 586–602.
- [21] A. H. Lang, S. Vora, H. Caesar, L. Zhou, and O. Beijbom, “Pointpillars: Fast encoders for object detection from point clouds,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.
- [22] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, “Swin transformer: Hierarchical vision transformer using shifted windows,” in *IEEE/CVF International Conference on Computer Vision*, 2021, pp. 10 012–10 022.
- [23] P. Ren, C. Li, G. Wang, Y. Xiao, Q. Du, X. Liang, and X. Chang, “Beyond fixation: Dynamic window visual transformer,” in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 11 987–11 997.
- [24] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, “An image is worth 16x16 words: Transformers for image recognition at scale,” in *International Conference on Learning Representations*, 2021.
- [25] B. Mildenhall, J. T. Barron, J. Chen, D. Sharlet, R. Ng, and R. Carroll, “Burst denoising with kernel prediction networks,” in *IEEE conference on Computer Vision and Pattern Recognition*, 2018, pp. 2502–2510.
- [26] A. Radford, L. Metz, and S. Chintala, “Unsupervised representation learning with deep convolutional generative adversarial networks,” *arXiv preprint arXiv:1511.06434*, 2015.
- [27] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal loss for dense object detection,” in *IEEE International Conference on Computer Vision*, 2017, pp. 2980–2988.
- [28] R. Xu, Y. Guo, X. Han, X. Xia, H. Xiang, and J. Ma, “Openca: An open cooperative driving automation framework integrated with co-simulation,” in *IEEE Intelligent Transportation Systems Conference*, 2021.
- [29] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, “Pointpillars: Fast encoders for object detection from point clouds,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 12 697–12 705.
- [30] I. Loshchilov and F. Hutter, “Decoupled weight decay regularization,” in *International Conference on Learning Representations.*, 2017.
- [31] Y. Ganin and V. Lempitsky, “Unsupervised domain adaptation by backpropagation,” in *International Conference on Machine Learning*. PMLR, 2015, pp. 1180–1189.