

CoDS: Enhancing Collaborative Perception in Heterogeneous Scenarios via Domain Separation

Yushan Han, Hui Zhang, *Member, IEEE*, Honglei Zhang,
Chuntao Ding, *Member, IEEE*, Yuanzhouhan Cao, Yidong Li, *Senior Member, IEEE*

Abstract—Collaborative perception has been proven to improve individual perception in autonomous driving through multi-agent interaction. Nevertheless, most methods often assume identical encoders for all agents, which does not hold true when these models are deployed in real-world applications. To realize collaborative perception in actual heterogeneous scenarios, existing methods usually align neighbor features to those of the ego vehicle, which is vulnerable to noise from domain gaps and thus fails to address feature discrepancies effectively. Moreover, they adopt transformer-based modules for domain adaptation, which causes the model inference inefficiency on mobile devices. To tackle these issues, we propose CoDS, a Collaborative perception method that leverages Domain Separation to address feature discrepancies in heterogeneous scenarios. The CoDS employs two feature alignment modules, i.e., Lightweight Spatial-Channel Resizer (LSCR) and Distribution Alignment via Domain Separation (DADS). Besides, it utilizes the Domain Alignment Mutual Information (DAMI) loss to ensure effective feature alignment. Specifically, the LSCR aligns the neighbor feature across spatial and channel dimensions using a lightweight convolutional layer. Subsequently, the DADS mitigates feature distribution discrepancy with encoder-specific and encoder-agnostic domain separation modules. The former removes domain-dependent information and the latter captures task-related information. During training, the DAMI loss maximizes the mutual information between aligned heterogeneous features to enhance the domain separation process. The CoDS employs a fully convolutional architecture, which ensures high inference efficiency. Extensive experiments demonstrate that the CoDS effectively mitigates feature discrepancies in heterogeneous scenarios and achieves a trade-off between detection accuracy and inference efficiency.

Index Terms—Connected and autonomous vehicle, collaborative perception, 3D object detection, cooperative computing.

I. INTRODUCTION

AS the number of vehicles continues to rise, the rapid advancement of intelligent transportation systems and autonomous driving technologies offers innovative solutions to address challenges in traffic efficiency and road safety. Among these, collaborative perception [1]–[9], which leverages multi-agent interactions to overcome occlusion and long-range limitations faced by individual vehicles, has gained

This work was supported in part by the Fundamental Research Funds for the Central Universities 2023JBZY031 and 2025JBMC018, in part by the National Natural Science Foundation of China under Grant U2268203 and 62203040, and in part by the Beijing Natural Science Foundation L221011.

Yushan Han, Hui Zhang, Honglei Zhang, Yuanzhouhan Cao and Yidong Li are with Key Laboratory of Big Data & Artificial Intelligence in Transportation (Beijing Jiaotong University), Ministry of Education and School of Computer Science and Technology, Beijing Jiaotong University, Beijing 100044, China. E-mail: {yushanhan, huizhang1, honglei.zhang, yzhcao, ydli}@bjtu.edu.cn. (Corresponding author: Yidong Li.)

Chuntao Ding is with the School of Artificial Intelligence, Beijing Normal University, Beijing 100875, China (e-mail: ctding@bnu.edu.cn).

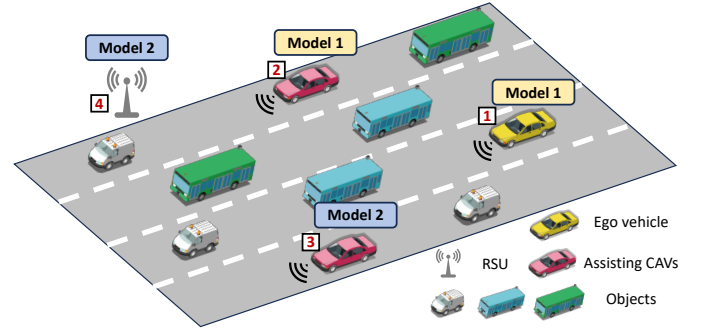


Fig. 1. Comparison between homogeneous and heterogeneous scenarios. In homogeneous scenario, Ego vehicle (No. 1) and CAV (No. 2) employ the identical model, resulting in shared homogeneous features. In heterogeneous scenario, Ego vehicle (No. 1) and CAV (No. 3) or RSU (No. 4) employ distinct models, leading to shared heterogeneous features.

significant attention in the field of autonomous driving. Collaborative perception usually includes vehicle-to-vehicle (V2V) [10], [11] and vehicle-to-everything (V2X) [12], [13] modes, enabling interaction among ego vehicles, assisting connected autonomous vehicles (CAVs) and roadside units (RSUs). Collaborative perception is categorized into early, intermediate and late collaboration, and most methods adopt intermediate collaboration for their fusion flexibility and low bandwidth requirements. Recent studies concentrate on improving communication mechanisms [14]–[16], fusion strategies [17], [18], and mitigating the noise issue caused by communication latency [19], [20] and localization errors [21]–[23].

Despite the notable success, existing methods often focus on homogeneous scenarios, where different agents employ identical encoders to extract features of the same size and distribution, thereby simplifying the feature fusion process. However, in practical applications, heterogeneous scenarios are more prevalent due to variations in hardware and software configurations [24]. As shown in Fig. 1, in heterogeneous scenarios, different encoders deployed in mobile devices extract features with discrepancies in both dimension and distribution, which can be attributed to distinct parameters and inductive biases [25]. For example, 3D object detector encoders with varying architectures and hyperparameters exhibit different sensitivities to fine-grained details [26], resulting in feature misalignment. This misalignment poses challenges for directly applying existing feature fusion methods, leading to performance degradation in collaborative perception and potentially compromising traffic safety. Consequently, enabling connected autonomous vehicles to collaborate effectively in heterogeneous scenarios has become a significant research area.

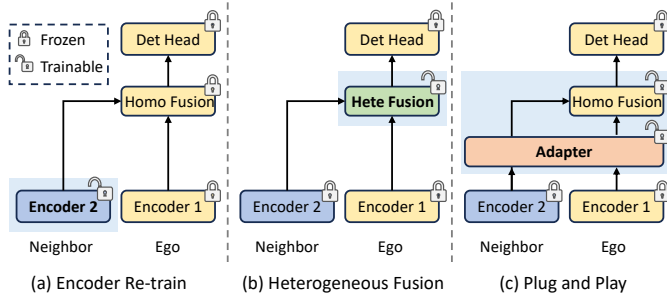


Fig. 2. **Different frameworks for heterogeneous collaboration.** (a) Re-trains encoders for neighbor agents, (b) designs a dedicated heterogeneous feature fusion module, and (c) incorporates a plug-and-play adapter that requires only fine-tuning based on the existing homogeneous feature fusion module.

Some methods have been proposed to address this issue, which can be classified into three categories (Fig. 2). (a) **Encoder re-train**: This type of work [27] re-trains the encoders of neighbor agents to align them with the feature space of the ego vehicle. However, this approach requires access to the original encoder architectures of neighbor agents, which can be challenging when agents come from different companies. (b) **Heterogeneous fusion**: These methods [28] design specialized modules to aggregate features with domain gaps. This approach requires training the new fusion module from scratch. (c) **Plug-and-play adapter**: This category [26], [29]–[31] introduces adapters to align features from heterogeneous encoders. These adapters are non-intrusive to the original agent models and only require fine-tuning on trained homogeneous fusion modules, offering a more flexible solution.

This work focuses on adapter-based methods for their flexibility and scalability. Existing adapter-based approaches [26], [29], [31] typically employ transformers to align the feature distribution from the neighbor to that of the ego vehicle. For instance, MPDA [26] utilizes a cross-domain transformer to convert heterogeneous features from neighbor agents into the feature space of the ego vehicle. Similarly, PnPDA [29] introduces a transformer-based semantic converter to transform the neighbor heterogeneous features into the ego semantic space. PolyInter [31] employs a transformer-based interpreter to project neighbor features, guided by a general prompt and agent-specific prompts. Despite their effectiveness, these methods suffer from two problems. (1) Since different encoders have varying characteristics and capabilities in feature extraction, the forced feature distribution conversion is vulnerable to noise from domain gaps [32] and results in information loss. (2) Furthermore, the high computational cost of the transformer reduces model inference efficiency, which is crucial for autonomous driving applications. Consequently, these methods achieve suboptimal performance in actual deployment.

In heterogeneous scenarios, features obtained from different agents can be considered multiple views of the same scene, containing both task-related and encoder-specific (task-unrelated) information. According to the classic hypothesis [33], valuable information is the one that is shared across multiple views. Substantial evidence from cognitive science and neuroscience [34], [35] also supports the idea that such view-invariant representations are encoded in the brain. There-

fore, task-related information across heterogeneous features is valuable for collaborative perception, while encoder-specific information reflects the inductive bias of the encoder and hinders effective feature fusion. To address distribution discrepancy issues in heterogeneous scenarios, we only need to capture task-related (domain-invariant) information between multiple agents while discarding encoder-specific (domain-specific) information. In other words, we aim to separate domain-invariant features from domain-specific ones.

Building on the above observations, we propose CoDS, a concise and effective collaborative perception method for heterogeneous scenarios. The core idea is to extract task-related information while eliminating encoder-specific information through domain separation. The CoDS comprises two alignment modules, i.e., the Lightweight Spatial-Channel Resizer (LSCR) and Distribution Alignment via Domain Separation (DADS). It also utilizes the Domain Alignment Mutual Information (DAMI) loss to enhance effective feature alignment. Specifically, the LSCR aligns the spatial and channel dimensions of neighbor features using a lightweight convolutional layer. After that, the DADS performs domain separation through two mechanisms: an encoder-specific module that removes domain-dependent information and an encoder-agnostic module that extracts task-related (domain-invariant) information. To improve inference efficiency, we leverage convolutional layers for domain separation, utilizing their parameter-sharing and parallelization capabilities instead of transformers. During training, the DAMI loss maximizes the mutual information between aligned heterogeneous features, ensuring that the features processed by LSCR and DADS preserve only task-related information while discarding encoder-specific content. This enhances both the robustness and effectiveness of feature alignment. In summary, the main contributions are as follows:

- We propose CoDS, a fully convolutional collaborative perception adapter designed to mitigate feature discrepancy issues in heterogeneous scenarios through domain separation. The CoDS enhances collaborative perception performance while ensuring high inference efficiency.
- The LSCR and DADS are proposed to align heterogeneous features. Specifically, the LSCR aligns neighbor features across spatial and channel dimensions using a lightweight convolutional layer. Subsequently, the DADS employs both encoder-specific and encoder-agnostic modules to remove domain-dependent information and capture task-related information effectively.
- The proposed DAMI loss enhances domain separation by maximizing the mutual information between aligned ego features and aligned neighbor features. This ensures that aligned features from multiple views preserve only task-related information in the current scene.
- Extensive experiments on three large-scale collaborative perception datasets (V2V4Real, OPV2V and V2XSet), three classic homogeneous feature fusion modules and five heterogeneous scenarios, demonstrate the superiority of the proposed CoDS in mitigating feature discrepancies while ensuring inference efficiency.

II. RELATED WORK

A. Collaborative Perception

Collaborative perception [36]–[39] is a vital technique in autonomous driving, enabling mobile vehicles to overcome the limitations of individual perception [40]–[42] through multi-agent interaction. It can be broadly categorized into three types based on the transmitted information: early fusion using raw point clouds, intermediate fusion using bird’s-eye view (BEV) features, and late fusion using detection outputs. Existing methods primarily focus on efficient communication [14]–[16], [43], adaptive feature fusion [17], [44]–[46], and addressing challenges such as time delays [19], [20], [47] and localization errors [21]–[23]. However, most approaches assume that all agents use identical encoders, an assumption that is often unrealistic in practice.

B. Collaborative Perception in Heterogeneous Scenarios

In real-world applications, heterogeneous scenarios are more common, leading to feature discrepancies that hinder effective information fusion. To address this issue, HEAL [27] re-trains the encoders of newly added agents to align with the ego domain. Hetecooper [28] introduces a heterogeneous feature fusion module that directly operates under heterogeneous settings. Furthermore, some methods [26], [29]–[31] align heterogeneous features by introducing lightweight and flexible adapters without re-training the original encoders. Specifically, MPDA [26] uses a cross-domain transformer to project neighbor features into the ego feature domain. PnPDA [29] introduces a semantic converter for feature alignment and a semantic enhancer to enhance ego features. STAMP [30] first trains a protocol network to construct a unified semantic domain, then trains local adapters and reverts for feature alignment. PolyInter [31] uses an interpreter network to project neighbor features into the ego agent’s semantic space, guided by a general prompt and agent-specific prompts for each newly added neighbor. These adapter-based methods typically use transformers to align neighbor and ego feature distributions, but this forced conversion is prone to domain-gap noise and information loss, while the high computational cost of transformers hinders inference efficiency. To address this, we propose a fully convolutional adapter with domain separation to mitigate noise vulnerability.

C. Domain Adaptation

Domain adaptation [48]–[50] addresses domain shift challenges when transferring knowledge across different domains. It encompasses various approaches, including feature-based, instance-based and model-based adaptation. Among these, feature-based adaptation is the most widely used, focusing on identifying domain-invariant features through techniques such as discrepancy minimization, adversarial learning and feature reconstruction. In heterogeneous collaborative perception, the use of distinct encoders introduces discrepancies in feature distributions, hindering effective feature fusion. To address this, MPDA [26] leverages adversarial learning for domain adaptation. Specifically, it introduces a domain classifier tasked

with distinguishing features from different domains, while the adapter aims to align features to confuse the domain classifier. This adversarial learning helps the adapter generate domain-invariant representations. On the other hand, PnPDA [29] employs contrastive learning to extract semantic information from heterogeneous features. It considers features of the same object in two feature maps as positive sample pairs and maximizes their semantic similarity. Unlike previous work, we adopt mutual information maximization for domain adaptation.

D. Mutual Information Estimation

Mutual Information (MI) is an information-theoretic measure that quantifies the dependency and shared information between two variables. Since true probability distributions are often unknown in real-world scenarios, recent studies have introduced neural networks for MI estimation. For instance, MINE [51] and InfoNCE [52] estimate MI by maximizing variational bounds, while Club [53] takes an alternative approach by minimizing variational bounds. Building on these works, some collaborative perception methods employ mutual information estimation for representation learning. For example, CRCNet [54] minimizes mutual information between fused feature pairs to reduce information redundancy from different neighbor agents, while CMiMC [55] maximizes mutual information between individual features and fused features to retain discriminative information from different views. In contrast to these methods, our approach leverages mutual information maximization to achieve domain alignment between heterogeneous features, ensuring effective feature fusion in collaborative perception.

III. PROBLEM FORMULATION

Consider N agents in V2V or V2X collaboration perception scenarios. Let x_i denote the LiDAR data collected by the i -th agent. The ego vehicle receives and fuses features from neighbor agents. The intermediate collaborative detection in a heterogeneous scenario can be formulated as:

$$\begin{aligned} f_i &= \mathbf{F}_{\text{enc}}^{\text{ego}}(x_i), \\ f_j &= \mathbf{F}_{\text{enc}}^{\text{nei}}(x_j), \\ f_{j \rightarrow i} &= \mathbf{F}_{\text{proj}}(\xi_i, (f_j, \xi_j)), \\ \hat{f}_i &= \mathbf{F}_{\text{fuse}}(f_i, f_{j \rightarrow i}), \\ y_i &= \mathbf{F}_{\text{det}}(\hat{f}_i), \end{aligned} \quad (1)$$

where $\mathbf{F}_{\text{enc}}^{\text{ego}}$ and $\mathbf{F}_{\text{enc}}^{\text{nei}}$ denote encoders of ego vehicle and neighbor agents. $\mathbf{F}_{\text{proj}}$, $\mathbf{F}_{\text{fuse}}$ and $\mathbf{F}_{\text{det}}$ represent feature pose projection, feature fusion and detection head, respectively. And the $\xi_i = (x_i, y_i, z_i, \theta_i, \phi_i, \psi_i)$ is the 6-DoF pose of the i -th agent. The fused feature is denoted as $\hat{f}_i$, and the collaborative detection output is denoted as y_i .

Training a collaborative detector is straightforward when encoders are identical, with features of the same size and distribution. However, in heterogeneous scenarios, where the encoders of the ego vehicle and neighbor agents differ, discrepancies in feature dimensions and distributions arise, leading to performance degradation after feature fusion. Our goal is to design a plug-and-play adapter to mitigate feature discrepancies while ensuring inference efficiency.

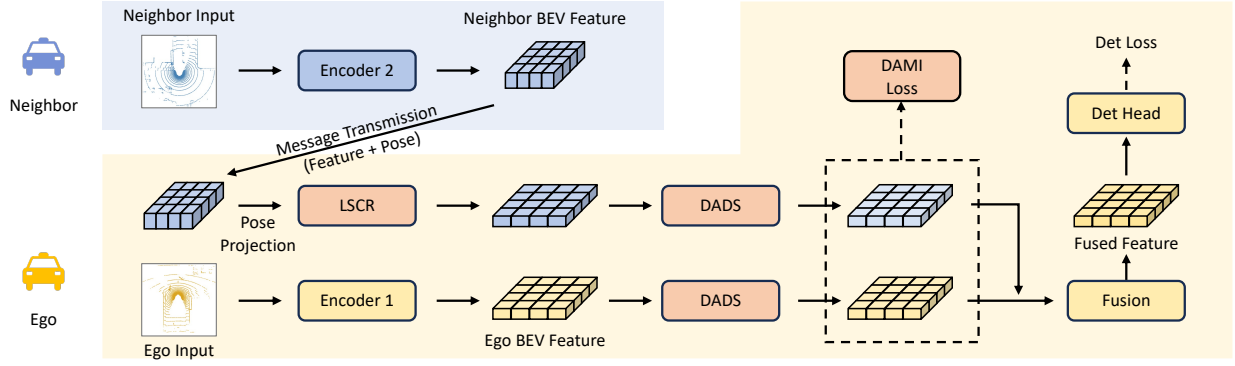


Fig. 3. **Overview of CoDS.** The ego and neighbor agents first extract features using distinct encoders. After receiving heterogeneous neighbor features, the ego applies the *Lightweight Spatial-Channel Resizer* (LSCR) and *Distribution Alignment via Domain Separation* (DADS) to align these features. During training, *Domain Alignment Mutual Information* (DAMI) loss is used to ensure effective feature alignment.

IV. METHODOLOGY

This section proposes a collaborative perception method to alleviate the feature discrepancies in heterogeneous scenarios. We first introduce the framework, followed by the details of key modules and loss functions.

A. Overall Architecture

To address the feature discrepancy issues in heterogeneous scenarios, we propose a collaborative perception method, called CoDS. As illustrated in Fig. 3, the method comprises two alignment components and a loss function, i.e., LSCR module, DADS module and DAMI loss. i) The LSCR adjusts the size of neighbor features in both spatial and channel dimensions. ii) The DADS employs encoder-specific and encoder-agnostic domain separation modules to remove domain-dependent information and capture task-related information. iii) During training, the DAMI loss maximizes mutual information between aligned ego and neighbor features to ensure distribution alignment. Specifically, the proposed components are formulated as:

$$\begin{aligned} \tilde{f}_{j \rightarrow i} &= \mathbf{F}_{\text{LSCR}}(f_{j \rightarrow i}), \\ \tilde{f}_i, \tilde{\tilde{f}}_{j \rightarrow i} &= \mathbf{F}_{\text{DADS}}(f_i, \tilde{f}_{j \rightarrow i}). \end{aligned} \quad (2)$$

Note that the distinct encoders are pre-trained in homogeneous scenarios and remain frozen in heterogeneous scenarios. Only the layers following the encoder are fine-tuned. To address the feature discrepancies issue, we employ the following steps before feature fusion for the ego vehicle. First, the LSCR module $\mathbf{F}_{\text{LSCR}}$ is employed to resize the projected neighbor features $f_{j \rightarrow i}$. Subsequently, the ego feature f_i and the resized neighbor features $\tilde{f}_{j \rightarrow i}$ are jointly passed through the DADS module $\mathbf{F}_{\text{DADS}}$, which effectively aligns their distributions. Finally, the aligned features $\tilde{f}_i$ and $\tilde{\tilde{f}}_{j \rightarrow i}$ are fused.

The proposed CoDS has the following advantages: i) We use domain separation modules to remove encoder-specific information and capture task-related information, thereby avoiding directly converting the domain of one encoder to another. ii) Benefiting from the parameter sharing and parallelization of the convolutional layer, our method is more efficient for training and inference than transformer-based collaborative perception methods.

B. Lightweight Spatial-Channel Resizer

The Lightweight Spatial-Channel Resizer (LSCR) aims to adjust the neighbor features to align with the feature size of the ego vehicle. Given the ego feature $f_i \in \mathbb{R}^{H \times W \times C}$ and neighbor features $f_{j \rightarrow i} \in \mathbb{R}^{H' \times W' \times C'}$, where $H' \neq H, W' \neq W, C' \neq C$, the LSCR will adjust the neighbor features to $\tilde{f}_{j \rightarrow i} = \mathbf{F}_{\text{LSCR}}(f_{j \rightarrow i}) \in \mathbb{R}^{H \times W \times C}$ as follows:

$$\begin{aligned} f_{j \rightarrow i}^0 &= \text{Conv}(f_{j \rightarrow i}), \\ \tilde{f}_{j \rightarrow i} &= \text{BI}(f_{j \rightarrow i}^0), \end{aligned} \quad (3)$$

where we first apply 1×1 convolutional layers for channel alignment and get features $f_{j \rightarrow i}^0 \in \mathbb{R}^{H' \times W' \times C}$. To achieve spatial alignment, we follow [56] to adopt bilinear interpolation (BI) and obtain the resized features $\tilde{f}_{j \rightarrow i} \in \mathbb{R}^{H \times W \times C}$.

C. Distribution Alignment via Domain Separation

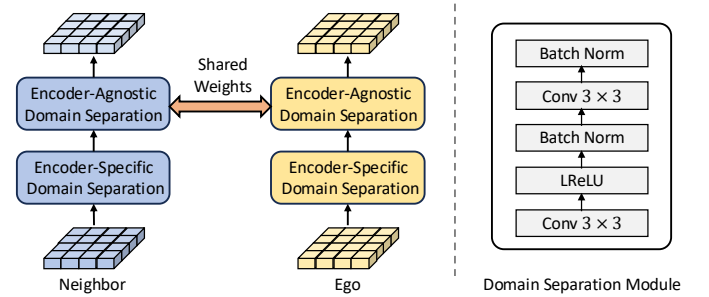


Fig. 4. **The architecture of the DADS.** The encoder-specific domain separation modules remove private information tied to individual encoders, whereas the encoder-agnostic modules capture shared task-related information.

Given features $f_i, \tilde{f}_{j \rightarrow i} \in \mathbb{R}^{H \times W \times C}$ with distribution discrepancy, we denote their marginal distributions as $\mathbb{P}(f_i)$ and $\mathbb{P}(\tilde{f}_{j \rightarrow i})$, where $\mathbb{P}(f_i) \neq \mathbb{P}(\tilde{f}_{j \rightarrow i})$. Extensive research [57], [58] has demonstrated the existence of projection functions in domain adaptation, which effectively maps features from disparate distributions into a common space. Therefore, we propose the Distribution Alignment via Domain Separation (DADS), which employs projection functions for each domain, denoted as $\mathbf{M}_{\text{ego}}(\cdot)$ and $\mathbf{M}_{\text{nei}}(\cdot)$. These functions ensure that the projected features maintain similar marginal distributions, i.e., $\mathbb{P}(\mathbf{M}_{\text{ego}}(f_i)) \approx \mathbb{P}(\mathbf{M}_{\text{nei}}(\tilde{f}_{j \rightarrow i}))$.

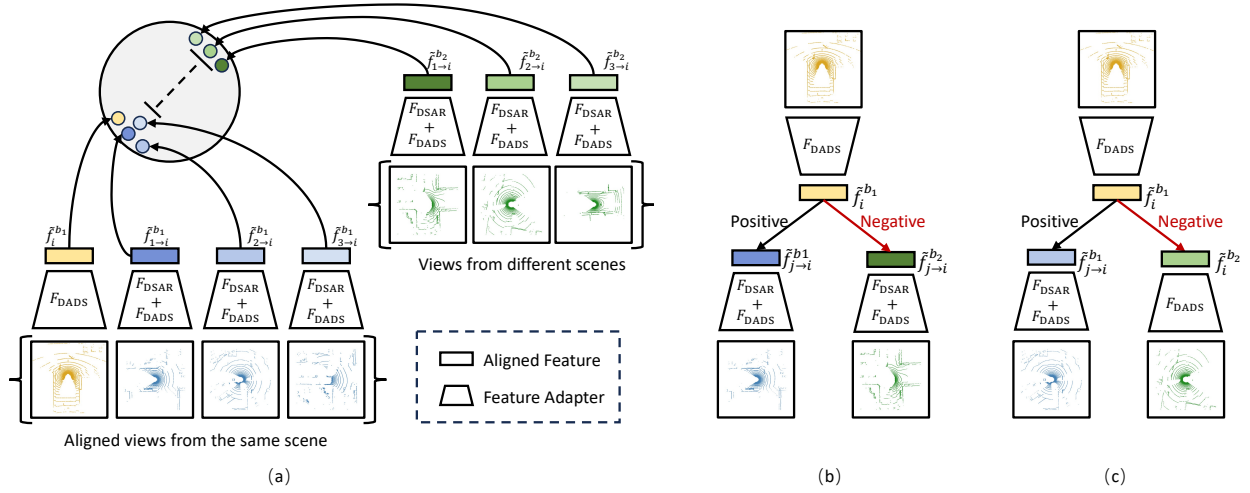


Fig. 5. **Illustration of the DAMI loss.** (a) We maximize mutual information between aligned features by bringing views of the same scene closer together and pushing views of different scenes further apart, which is achieved through contrastive learning on positive and negative pairs. (b) and (c) illustrate the construction of positive and negative samples. In scene b_1 , the j -th aligned neighbor feature $\tilde{f}_{j \rightarrow i}^{b_1}$ serves as a positive sample for the aligned ego feature $\tilde{f}_i^{b_1}$. (b) If the ego in another scene b_2 also has the j -th agent, the negative sample is the aligned neighbor feature $\tilde{f}_{j \rightarrow i}^{b_2}$ from scene b_2 . (c) Otherwise, the negative sample is the aligned ego feature $\tilde{f}_i^{b_2}$ from scene b_2 .

As shown in Fig. 4, the projection functions $M_{\text{ego}}(\cdot)$ and $M_{\text{nei}}(\cdot)$ are implemented using two types of domain separation modules. The first type is the encoder-specific domain separation modules $M^{\text{es}}(\cdot)$, which are constructed independently for each domain and directly achieve domain separation by removing domain-dependent information. The second type is the encoder-agnostic domain-separation modules $M^{\text{ea}}(\cdot)$, which employ a weight-sharing scheme and indirectly achieve domain separation by capturing task-related information (i.e., domain-invariant features) through projection into a common latent feature space. Both types of modules are arranged sequentially within the domain and share an identical structure, comprising two 3×3 convolutional layers, Batch Normalization, and the LeakyReLU activation function. Consequently, the overall projection functions can be expressed as:

$$\begin{aligned} M_{\text{ego}}(\cdot) &= (M_{\text{ego}}^{\text{es}} \circ M_{\text{ego}}^{\text{ea}})(\cdot), \\ M_{\text{nei}}(\cdot) &= (M_{\text{nei}}^{\text{es}} \circ M_{\text{nei}}^{\text{ea}})(\cdot), \end{aligned} \quad (4)$$

where $\circ$ denotes the connection of convolution layers. And features processed by domain separation modules can be expressed as $\tilde{f}_i = M_{\text{ego}}(f_i)$ and $\tilde{f}_{j \rightarrow i} = M_{\text{nei}}(f_{j \rightarrow i})$.

Note that both encoder-specific and encoder-agnostic domain separation modules are indispensable. Using only encoder-specific modules would preserve some private information tied to individual encoders, which hinders complete distribution alignment. Conversely, relying solely on encoder-agnostic modules would be vulnerable to encoder-specific information, thereby impeding the projection of features into a shared space. Further analysis and discussion can be found in the ablation study section.

D. Domain Alignment Mutual Information Loss

In collaborative perception, heterogeneous features from different agents can be viewed as multiple views of the same

scene, with only task-related information accurately representing the environment. To ensure the adapter captures this task-related information while eliminating encoder-specific details, we maximize the mutual information (MI) between aligned ego and neighbor features. This enhances representation consistency across different views and effectively mitigates distribution discrepancies. The MI between aligned ego feature $\tilde{f}_i$ and aligned neighbor feature $\tilde{f}_{j \rightarrow i}$ is defined as:

$$\mathcal{I}(\tilde{f}_i; \tilde{f}_{j \rightarrow i}) = \sum_{x \in \tilde{f}_i} \sum_{y \in \tilde{f}_{j \rightarrow i}} p(x, y) \log \frac{p(x, y)}{p(x)p(y)}. \quad (5)$$

Vanilla MI only measures the dependency between two random variables, which is insufficient for capturing dependencies across aligned features from multiple views. To fill this gap, we define the Domain Alignment Mutual Information (DAMI), which is adaptable to multiple views. Specifically, DAMI first constructs pairwise MI between the aligned ego feature $\tilde{f}_i$ and each aligned neighbor feature $\tilde{f}_{j \rightarrow i}$, then averages these pairwise MIs to form DAMI. Let N_{nei} represent the total number of neighbor agents for the i -th ego vehicle. Then, the DAMI for the i -th ego can be formulated as:

$$\mathcal{I}_{\text{DAMI}} = \frac{1}{N_{\text{nei}}} \sum_{j=1}^{N_{\text{nei}}} \mathcal{I}(\tilde{f}_i; \tilde{f}_{j \rightarrow i}). \quad (6)$$

To mitigate distribution discrepancies between heterogeneous features, the CoDS aims to maximize DAMI, which requires a lower-bound estimation. Following [52], we estimate this MI lower bound using contrastive loss between aligned feature pairs, which can be formulated as:

$$\mathcal{I}(\tilde{f}_i; \tilde{f}_{j \rightarrow i}) \geq \log(k) - \mathcal{L}_{\text{contrast}} = \hat{\mathcal{I}}(\tilde{f}_i; \tilde{f}_{j \rightarrow i}), \quad (7)$$

where the contrastive loss $\mathcal{L}_{\text{contrast}}$ is used to train a discriminator that distinguishes aligned features from different scenes. Specifically, the discriminator minimizes the loss by assigning

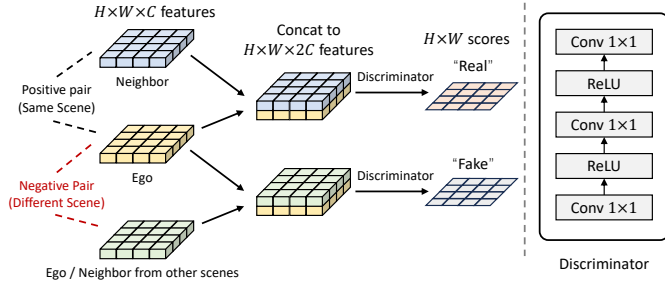


Fig. 6. **The architecture of discriminator.** Positive and negative feature pairs are concatenated separately, and a convolutional discriminator is then applied to score these “real” (positive) and “fake” (negative) feature pairs.

a high score to positive pairs (i.e., aligned ego and neighbor features from the same scene) and a low score to negative pairs (i.e., aligned ego and neighbor features from different scenes). The parameter k represents the number of negative pairs in the sample set, indicating that incorporating more negative samples can enhance the learning of the feature adapter.

Minimizing the objective $\mathcal{L}_{\text{contrast}}$ effectively maximizes the lower bound on the mutual information $\hat{\mathcal{I}}(f_i; f_{j \rightarrow i})$. Therefore, the task of DAMI maximization is transformed into a contrastive learning minimization problem, and we can use $\mathcal{L}_{\text{contrast}}$ to represent the DAMI loss $\mathcal{L}_{\text{DAMI}}$:

$$\begin{aligned}
 \mathcal{L}_{\text{DAMI}} &= \max \frac{1}{N_{\text{nei}}} \sum_{j=1}^{N_{\text{nei}}} \hat{\mathcal{I}}(f_i; f_{j \rightarrow i}) \\
 &= \max \frac{1}{N_{\text{nei}}} \sum_{j=1}^{N_{\text{nei}}} [\log(k) - \mathcal{L}_{\text{contrast}}] \\
 &= \min \frac{1}{N_{\text{nei}}} \sum_{j=1}^{N_{\text{nei}}} [\mathcal{L}_{\text{contrast}} - \log(k)]. \quad (8)
 \end{aligned}$$

To implement contrastive loss, we first construct positive and negative pairs. As shown in Fig. 5(a), to achieve effective domain separation and enhance representation consistency across heterogeneous features, we treat aligned ego and neighbor features from the same scene as positive pairs, while aligned ego features and those from other scenes are treated as negative pairs. In each training iteration, we sample B scenes, denoted as $\mathcal{B} = \{b_1, b_2, \dots, b_n\}$. In a scene b_1 , the aligned ego feature serves as the anchor, with each aligned neighbor feature as a positive sample for the anchor. As shown in Fig. 5(b), for the j -th aligned neighbor feature, the negative sample is the j -th aligned neighbor feature from another scene b_2 . The different scenes and distinct encoders between the anchor and the negative sample ensure that negative pairs are pushed apart. However, not every collaborative scene has j neighbor agents. To address this issue, we use the aligned ego feature as the negative sample when other scenes lack sufficient neighbor agents, as illustrated in Fig. 5(c). The structure of the discriminator is shown in Fig. 6. For each positive and negative feature pair, we first concatenate the anchor with the positive or negative sample along the channel dimension. The concatenated features are then fed into the discriminator, which outputs $H \times W$ score maps used to calculate the $\mathcal{L}_{\text{contrast}}$. The details of constructing positive

Algorithm 1: Calculate contrastive loss $\mathcal{L}_{\text{contrast}}$

Input : Aligned ego feature $\tilde{f}_i$ and neighbor features $\{\tilde{f}_{j \rightarrow i}\}_{j=1}^{N_{\text{nei}}}$ in one iteration. Discriminator D .

```

1 ego_anchor  $\leftarrow []$ ;
2 pos_sample  $\leftarrow []$ ;
3 pos_num  $\leftarrow []$ ;
4 // Traverse all scenes in current iteration
5 for  $b = b_1, b_2, \dots, b_n$  do
6   pos_num  $\leftarrow N_{\text{nei}}^b$ ;
7   ego_anchor  $\leftarrow \tilde{f}_i^b$ ;
8   pos_sample  $\leftarrow \{\tilde{f}_{j \rightarrow i}^b\}_{j=1}^{N_{\text{nei}}^b}$ ;
9  $\mathcal{L}_{\text{cont}} \leftarrow []$ ;
10 for  $k = 1, 2, \dots, \max(\text{pos\_num})$  do
11   ego_anchor_k  $\leftarrow []$ ;
12   pos_sample_k  $\leftarrow []$ ;
13   neg_sample_k  $\leftarrow []$ ;
14   // The  $k$ -th positive sample pairs
15   for  $v = 1, 2, \dots, \text{len}(\text{pos\_num})$  do
16     if pos_num[v] >  $k$  then
17       ego_anchor_k  $\leftarrow \text{ego\_anchor}[v]$ ;
18       pos_sample_k  $\leftarrow \text{pos\_sample}[v][k]$ ;
19   // Find the  $k$ -th negative sample
20   if len(pos_sample_k) > 1 then
21     for  $p = 2, \dots, \text{len}(\text{pos\_sample\_k})$  do
22       neg_sample_k  $\leftarrow \text{pos\_sample\_k}[p]$ 
23     neg_sample_k  $\leftarrow \text{pos\_sample\_k}[1]$ ;
24   else
25     repeat
26        $q \leftarrow \text{Random}([1, \text{len}(\text{ego\_anchor})])$ ;
27     until  $q \neq v$ ;
28     neg_sample_k  $\leftarrow \text{ego\_anchor}[q]$ ;
29   // Calculate contrastive loss
30    $\mathcal{L}_{\text{cont}} \leftarrow D(\text{ego\_anchor\_k}, \text{pos\_sample\_k}, \text{neg\_sample\_k})$ ;
31  $\mathcal{L}_{\text{contrast}} \leftarrow \text{mean}(\mathcal{L}_{\text{cont}})$ ;
Output:  $\mathcal{L}_{\text{contrast}}$  in current iteration

```

and negative sample pairs and calculating contrastive loss are illustrated in Algorithm 1.

E. Overall Loss Function

The total loss $\mathcal{L}$ for training CoDS is summarized as follows:

$$\begin{aligned}
 \mathcal{L} &= \mathcal{L}_{\text{det}} + \beta_{\text{DAMI}} \mathcal{L}_{\text{DAMI}}, \\
 \mathcal{L}_{\text{det}} &= \alpha_{\text{cls}} \mathcal{L}_{\text{cls}} + \alpha_{\text{reg}} \mathcal{L}_{\text{reg}} + \alpha_{\text{dir}} \mathcal{L}_{\text{dir}},
 \end{aligned}$$

where $\mathcal{L}_{\text{det}}$ denotes detection loss, which includes focal loss for classification $\mathcal{L}_{\text{cls}}$, smooth-L1 loss for regression $\mathcal{L}_{\text{reg}}$ and softmax classification loss for direction $\mathcal{L}_{\text{dir}}$. Specifically, β_{DAMI} , α_{cls} , α_{reg} and α_{dir} are the weights for loss functions.

V. EXPERIMENTAL RESULTS

A. Datasets and Evaluation Metrics

We validate the proposed CoDS method on the task of LiDAR-based collaborative 3D object detection using three large-scale collaborative perception datasets.

V2V4Real [10] is the first large-scale, real-world V2V dataset collected using two vehicles. It includes 20K frames of point clouds, with 6,958 frames for training, 1,993 for validation, and 1,993 for testing.

OPV2V [11] is a simulated V2V perception dataset where 2 to 7 collaborative vehicles, each equipped with a LiDAR and four cameras. It comprises 11,464 frames of 3D point clouds and 230K annotated 3D boxes, split into 6,374 training frames, 1,980 validation frames and 2,170 testing frames.

V2XSet [13] is another simulated dataset designed for V2X applications, featuring both roadside units and autonomous vehicles. It contains 6,694 training frames, 1,920 validation frames and 2,834 testing frames.

We evaluate different methods on the testing sets of three datasets, assessing both accuracy and efficiency. For accuracy, we use 3D detection performance measured by average precision (AP) at Intersection-over-Union (IoU) thresholds of 0.50 and 0.70. For efficiency, we evaluate the frames per second (FPS) to measure the processing speed of models.

B. Experimental Setups

1) *Implementation Details*: The collaborative detector in the heterogeneous scenario is fine-tuned from the homogeneous scenario. Initially, we train collaborative detectors with various encoders in the homogeneous scenario. Next, we load and freeze distinct pre-trained encoders for different agents and fine-tune adapters and fusion modules. This work simplifies the setting and only considers two different encoders. To train the model, we use the Adam optimizer with a learning rate of 0.002. The weight of the DAMI loss is set to $\beta_{\text{DAMI}} = 1$. For the detection loss, we adopt the same weight settings as PointPillars [59]: $\alpha_{\text{cls}} = 1$, $\alpha_{\text{reg}} = 2$ and $\alpha_{\text{dir}} = 0.2$. All models are trained on NVIDIA RTX 4090. For quantitative comparison, we select three classic feature fusion modules, AttFusion [11], DiscoNet [60] (student model only) and CoBEVT [61]. Specifically, DiscoNet [60] is selected to assess efficiency performance and to conduct ablation studies.

2) *Distinct Encoder*: We select PointPillars [59], SECOND [62] and VoxelNet [63] as the detection encoders. The half LiDAR range (X&Y), voxel resolution of the encoders, and feature size (C×H×W) are summarized in Table I. The detection accuracy of collaborative detectors in homogeneous scenarios is listed in Table II. In the subsequent experiments, p_0 and p_1 represent PointPillars with different voxel parameters. Similarly, s_1 and v_1 refer to SECOND and VoxelNet, respectively. We consider two heterogeneous settings: i) The ego agent is equipped with the pre-trained p_0 , while neighbor agents are equipped with pre-trained p_1 , s_1 or v_1 . ii) The ego agent is equipped with the pre-trained s_1 or v_1 , while neighbor agents are equipped with pre-trained p_0 .

TABLE I
DETAILED PARAMETERS OF HETEROGENEOUS ENCODERS.

Abbr	Encoder Setting			Feature Size		
	Encoder	LiDAR Range	Voxel Size	V2V4Real	OPV2V	V2XSet
p_0	PointPillars	140.8, 38.4	0.4, 0.4	256, 96, 352	256, 96, 352	256, 96, 352
p_1	PointPillars	153.6, 38.4	0.6, 0.6	256, 64, 256	256, 64, 256	256, 64, 256
s_1	SECOND	140.8, 40	0.1, 0.1	512, 100, 352	256, 100, 352	512, 100, 352
v_1	VoxelNet	140.8, 40	0.8, 0.8	512, 100, 352	256, 100, 352	256, 100, 352

3) *Baselines*: Since HEAL [27] and Hetecooper [28] require retraining the encoder or designing a new feature fusion module, we only compare the CoDS with plug-and-play adapter-based methods [26], [29]–[31] for a fair comparison. Additionally, we consider a simple baseline HETE, which uses a naive resizer without domain adapters. Note that STAMP [30] and PolyInter [31] were originally designed for open heterogeneous scenarios, where new types of agents with previously unseen models may join the collaboration. Since our focus is on the general heterogeneous setting, where the neighbor agents are fixed but their models are distinct, we adapt and reproduce STAMP and PolyInter accordingly to ensure a fair and consistent comparison.

- **HETE**: It utilizes direct bilinear interpolation and channel slices (or padding) for feature resizing.
- **MPDA** [26]: It employs a learnable feature resizer to resize neighbor features, and a cross-domain transformer to convert the domain of features to the ego domain.
- **PnPDA** [29]: It uses a semantic converter to transform neighbor heterogeneous features into the ego domain and a semantic enhancer to strengthen the representation of ego features. Both the converter and enhancer share the same transformer-based architecture.
- **STAMP** [30]: It first trains a protocol network, then trains ConvNeXt [64] based local adapters and reverts for feature alignment.
- **PolyInter** [31]: It projects neighbor features into the ego domain using an interpreter network guided by a general prompt and agent-specific prompts.

C. Quantitative Evaluation

1) *Accuracy Comparison*: Table III shows the detection accuracy in different datasets, collaborative detectors and heterogeneous scenarios, where p_0 denotes collaborative detectors in homogeneous scenarios. In contrast, p_0+p_1 indicates the ego is equipped with the p_0 encoder while neighbor agents are equipped with the p_1 encoder.

In V2V4Real, when collaborative detectors are AttFusion or DiscoNet, the accuracy of HETE decreases by approximately 10 in AP@0.70 compared to homogeneous scenarios. For CoBEVT, this drop increases to about 20 in AP@0.70. Adapter-based methods such as MPDA, PnPDA, STAMP, PolyInter and our CoDS enhance the accuracy of collaborative detectors in heterogeneous scenarios, with CoDS demonstrating the most consistent improvements. In particular, combining DiscoNet with CoDS yields an average gain of 20.32 in AP@0.50 and 11.39 in AP@0.70 compared to HETE. In addition, CoDS enables collaborative detectors in heterogeneous

TABLE II
COLLABORATIVE DETECTION RESULTS (AP@0.50/AP@0.70) IN HOMOGENEOUS SCENARIOS.

Datasets	V2V4Real			OPV2V			V2XSet		
Abbr	AttFusion	DiscoNet	CoBEVT	AttFusion	DiscoNet	CoBEVT	AttFusion	DiscoNet	CoBEVT
p_0	73.06/46.24	72.46/43.33	70.70/41.59	93.11/82.05	92.61/82.29	93.52/78.63	88.29/73.41	87.92/74.09	89.05/74.15
p_1	66.63/35.87	67.90/38.13	65.11/36.14	92.81/80.08	92.54/81.33	94.17/80.10	87.74/68.12	88.14/70.53	85.86/56.30
s_1	69.03/41.87	73.25/50.02	55.90/32.07	81.24/71.21	80.93/71.55	90.19/82.10	85.82/72.19	86.39/73.42	81.93/70.23
v_1	67.26/40.64	69.09/40.53	50.98/31.26	76.59/65.25	49.77/40.96	86.31/73.87	80.26/66.88	81.56/68.07	77.71/64.01

TABLE III
DETECTION PERFORMANCE OF COLLABORATIVE DETECTORS. p_0 , s_1 AND v_0 DENOTES THE COLLABORATIVE DETECTOR IN HOMOGENEOUS SCENARIOS, p_0+p_1 , p_0+s_1 , p_0+v_1 , s_1+p_0 AND v_1+p_0 DENOTE DIFFERENT HETEROGENEOUS SCENARIOS. RESULTS ARE REPORTED IN AP@0.50/AP@0.70.

Datasets		V2V4Real			OPV2V			V2XSet		
Encoder	Method	AttFusion	DiscoNet	CoBEVT	AttFusion	DiscoNet	CoBEVT	AttFusion	DiscoNet	CoBEVT
p_0	-	73.06/46.24	72.46/43.33	70.70/41.59	93.11/82.05	92.61/82.29	93.52/78.63	88.29/73.41	87.92/74.09	89.05/74.15
p_0+p_1	HETE	55.41/36.59	50.96/31.19	33.19/5.85	83.86/70.07	86.64/76.28	66.63/42.72	80.56/66.42	81.15/65.43	79.06/50.78
	MPDA	59.34/36.92	58.45/35.51	60.34/35.31	88.22/77.55	87.69/77.31	79.13/69.99	83.04/68.96	81.48/67.11	85.46/70.87
	PnPDA	63.99/32.63	60.08/38.77	63.53/35.61	87.31/74.18	76.65/55.98	66.24/55.71	82.21/67.27	69.06/50.04	74.05/56.20
	STAMP	64.08/40.93	63.59/37.17	60.07/35.24	85.42/61.36	80.78/64.63	86.53/75.09	82.42/63.59	81.89/58.70	85.67/71.76
	PolyInter	56.35/37.79	59.32/37.81	55.57/27.75	88.07/77.53	84.63/74.49	88.66/79.07	82.83/68.94	81.70/69.14	69.16/58.19
	CoDS (Ours)	60.85/37.21	60.99/37.89	61.62/40.30	88.81/77.55	87.97/77.49	86.84/74.48	84.03/69.39	82.71/67.11	86.24/72.37
p_0+s_1	HETE	55.45/36.58	41.21/28.84	12.10/6.51	87.96/76.30	86.44/76.03	62.86/33.63	82.87/68.21	81.31/67.81	76.32/45.46
	MPDA	59.61/37.40	67.24/40.05	65.94/35.10	91.99/79.57	91.18/77.72	91.47/77.87	87.42/73.11	87.04/71.32	83.54/65.83
	PnPDA	68.38/39.60	68.90/44.98	64.82/37.24	83.34/71.99	75.72/56.36	92.85/79.48	77.33/62.06	76.03/64.39	59.96/47.64
	STAMP	68.53/43.30	70.59/42.05	65.94/38.43	79.54/62.61	90.59/72.51	86.01/75.86	87.31/60.06	87.99/68.16	88.04/73.96
	PolyInter	65.17/36.25	66.83/42.31	65.17/36.25	92.50/81.47	90.53/78.18	92.47/82.72	89.09/75.61	88.91/76.51	82.15/68.50
	CoDS (Ours)	65.37/44.21	71.27/46.52	69.00/38.58	92.83/79.86	92.11/81.76	90.13/77.49	89.14/74.61	90.12/78.33	88.63/76.06
p_0+v_1	HETE	55.42/36.60	33.93/24.13	46.33/16.71	87.95/76.30	86.89/76.37	90.09/78.45	82.87/68.21	79.38/66.59	80.12/52.34
	MPDA	59.39/38.08	65.08/37.49	67.96/37.57	89.96/77.68	89.96/77.68	90.18/80.59	85.78/70.47	86.88/73.33	50.83/38.89
	PnPDA	68.26/43.38	65.58/35.97	67.14/38.03	74.07/64.44	77.11/61.95	79.45/65.13	55.09/45.43	78.43/61.56	49.52/42.44
	STAMP	69.92/45.09	69.25/42.55	67.28/38.10	87.26/56.37	90.99/70.79	91.51/76.91	86.57/63.08	87.15/67.29	87.98/73.76
	PolyInter	68.53/43.30	66.67/39.08	61.29/32.32	91.97/79.50	89.25/73.70	90.14/79.84	88.20/73.29	87.25/74.70	78.47/65.47
	CoDS (Ours)	66.04/44.81	69.78/43.41	68.38/39.87	93.63/82.45	93.35/84.07	90.24/80.00	88.69/76.11	87.16/73.73	88.32/73.81
s_1	-	69.03/41.87	73.25/50.02	55.90/32.07	81.24/71.21	80.93/71.55	90.19/82.10	85.82/72.19	86.39/73.42	81.93/70.23
s_1+p_0	HETE	54.17/34.77	55.36/40.73	51.77/27.97	81.49/71.39	74.68/65.85	88.34/78.51	81.07/68.74	73.91/64.53	83.08/66.38
	MPDA	63.82/36.32	68.58/42.85	67.46/34.74	89.73/75.21	87.05/70.64	89.56/81.04	82.01/70.24	80.34/69.35	86.03/73.13
	PnPDA	60.73/37.92	60.21/38.10	62.58/31.12	61.18/51.43	79.97/48.95	92.69/79.10	87.35/72.63	78.41/68.53	86.34/72.59
	STAMP	62.18/40.89	67.77/37.41	66.97/33.92	83.55/67.35	82.51/62.19	93.19/83.29	80.02/70.32	60.55/41.87	38.60/34.12
	PolyInter	63.35/41.97	65.36/45.99	62.28/34.50	90.85/77.08	86.97/71.05	66.63/60.26	86.42/73.19	73.92/66.06	80.44/69.41
	CoDS (Ours)	69.79/46.58	66.07/45.34	67.47/36.49	86.76/77.44	90.59/78.34	93.57/84.01	83.40/72.66	85.91/71.46	86.92/73.52
v_1	-	67.26/40.64	69.09/40.53	50.98/31.26	76.59/65.25	49.77/40.96	86.31/73.87	80.26/66.88	81.56/68.07	77.71/64.01
v_1+p_0	HETE	51.23/33.48	52.45/32.47	40.98/18.70	78.19/65.90	77.45/63.82	82.86/65.11	74.34/62.95	76.15/62.38	52.19/40.39
	MPDA	65.21/39.51	64.27/36.45	62.07/34.95	88.59/70.59	79.85/66.90	90.96/77.78	83.92/67.84	78.25/66.66	70.31/49.14
	PnPDA	57.69/34.23	63.07/32.60	54.13/29.72	62.43/50.76	86.15/64.21	85.39/73.98	86.42/70.72	82.18/69.59	84.71/68.52
	STAMP	66.78/39.23	65.75/27.94	62.99/37.66	84.02/52.75	82.27/60.98	91.68/79.20	83.71/64.88	77.97/57.55	85.01/68.57
	PolyInter	64.30/40.73	63.83/38.20	58.72/33.35	91.45/79.51	87.38/68.85	90.51/76.73	86.98/73.92	86.36/73.19	85.02/69.53
	CoDS (Ours)	67.00/42.60	67.41/41.26	62.13/35.87	89.13/74.41	92.06/77.34	91.97/79.49	84.70/65.83	85.26/70.27	85.52/69.71

TABLE IV
INFERENCE SPEED (FPS) UNDER DIFFERENT COLLABORATION NUMBERS.

Datasets		V2V4Real		OPV2V			V2XSet			
Agent Numbers		2 Agents	2 Agents	3 Agents	4 Agents	5 Agents	2 Agents	3 Agents	4 Agents	5 Agents
p_0+p_1	1. MPDA	28.74	30.69	21.64	17.20	14.78	31.19	18.52	15.77	13.59
	2. PnPDA	29.58	34.29	27.21	24.35	22.15	40.36	31.10	25.81	23.53
	3. STAMP	39.35	39.38	32.60	28.94	26.01	40.20	29.90	26.94	24.36
	4. PolyInter	39.59	40.87	28.96	24.28	20.86	41.19	25.56	21.70	18.97
	5. CoDS (Ours)	46.99	47.58	39.67	35.01	33.06	48.99	37.87	34.43	31.97
p_0+s_1	1. MPDA	28.88	27.33	18.56	14.86	12.72	29.59	17.23	14.66	12.75
	2. PnPDA	33.76	29.63	23.36	20.65	18.66	40.00	33.38	30.26	27.42
	3. STAMP	41.54	36.19	29.05	25.48	22.70	40.69	31.24	28.39	26.03
	4. PolyInter	41.50	36.39	26.05	21.75	18.72	40.75	26.24	22.70	20.00
	5. CoDS (Ours)	50.19	43.56	35.47	31.17	28.16	48.82	38.17	35.82	33.44
p_0+v_1	1. MPDA	29.03	29.76	20.44	16.87	14.20	29.94	17.40	14.75	12.87
	2. PnPDA	33.58	35.32	28.64	25.58	21.52	42.65	33.04	30.94	28.22
	3. STAMP	40.78	40.91	34.20	30.77	28.06	41.39	31.61	28.77	26.45
	4. PolyInter	41.49	41.35	30.20	25.51	22.18	41.47	26.63	23.05	20.35
	5. CoDS (Ours)	39.55	49.02	40.56	39.39	35.86	49.59	38.43	37.77	35.06
s_1+p_0	1. MPDA	22.76	22.66	16.30	13.47	11.35	22.87	13.48	11.76	10.12
	2. PnPDA	21.93	30.71	25.58	21.24	17.76	29.31	24.01	20.52	16.56
	3. STAMP	16.72	29.61	24.45	21.64	19.16	28.67	24.55	22.34	19.27
	4. PolyInter	18.61	30.09	22.32	18.93	16.11	29.87	22.32	17.23	15.10
	5. CoDS (Ours)	24.03	33.43	28.49	25.74	22.96	34.04	28.78	25.09	22.04
v_1+p_0	1. MPDA	22.64	23.64	16.53	13.33	11.42	22.71	13.84	11.81	10.30
	2. PnPDA	21.98	28.23	23.82	22.40	19.68	29.16	23.36	20.62	18.82
	3. STAMP	13.45	27.67	23.37	21.03	18.67	27.77	21.51	19.59	17.34
	4. PolyInter	14.82	28.77	21.56	18.61	16.01	28.63	19.20	16.82	14.50
	5. CoDS (Ours)	24.21	31.59	28.46	26.00	22.87	34.24	26.35	24.75	20.27

scenarios (s_0+p_0 and v_0+p_1) to achieve higher AP@0.70 than those in homogeneous scenarios (s_1 and v_0).

In OPV2V and V2XSet, the accuracy of HETE in heterogeneous scenarios decreases by approximately 10 to 30 AP@0.70 compared to homogeneous scenarios. However, in OPV2V, when the collaborative detectors are AttFusion and DiscoNet, HETE achieves higher accuracy in the v_1+p_0 setting compared to v_1 . This is because the feature discrepancies in these scenarios are minimal, allowing heterogeneous features to still achieve effective complementarity. In heterogeneous scenarios, MPDA, PnPDA, STAMP and PolyInter may exhibit unstable performance, occasionally performing worse than HETE. In contrast, our CoDS consistently outperforms previous methods across most settings, highlighting the effectiveness of domain separation in addressing feature discrepancies.

2) *Efficiency Comparison*: We evaluate the inference efficiency of adapter-based methods (MPDA, PnPDA, STAMP, PolyInter and CoDS) in heterogeneous scenarios and examine the impact of the number of agents on inference efficiency. Specifically, we use 2 agents in V2V4Real and 2 to 5 agents in OPV2V and V2XSet.

The results in Table IV indicate that with a small number of collaborators, the collaborative detectors with CoDS exhibit significantly higher inference speeds than previous methods. Specifically, when there is only one neighbor agent, the CoDS

outperforms MPDA and PnPDA by over 30% in FPS.

As the number of agents increases, the FPS of collaborative detectors decreases due to the additional computational requirements for aligning and fusing more features. Despite this, CoDS maintains a significant inference advantage over previous methods. Specifically, when the maximum number of agents reaches five, CoDS achieves an FPS improvement of over 100% compared to MPDA and more than 20% compared to PnPDA, STAMP and PolyInter. These improvements are largely attributed to the fully convolutional architecture of CoDS, which ensures relatively low inference costs.

TABLE V
PARAMETER SIZES (M) OF DIFFERENT ADAPTER-BASED MODELS.

Methods	MPDA	PnPDA	STAMP	PolyInter	CoDS (Ours)
Params (M)	6.12	3.29	4.81	46.22	3.67

We also report the parameter sizes of different adapter-based models in Table V. The results show that our CoDS requires only 3.67M parameters, significantly smaller than PolyInter (46.22M) and competitive with other efficient adapter-based methods, such as PnPDA (3.29M).

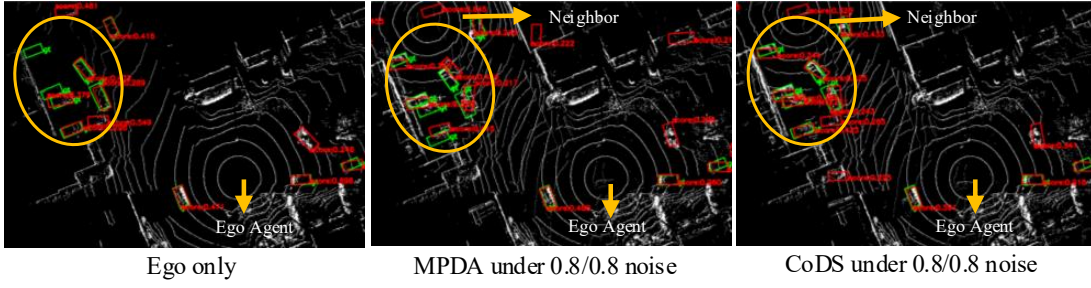


Fig. 7. **Impact of pose error in heterogeneous scenarios.** Large localization noise causes severe bounding box misalignments in collaborative perception relative to the ego-only baseline.

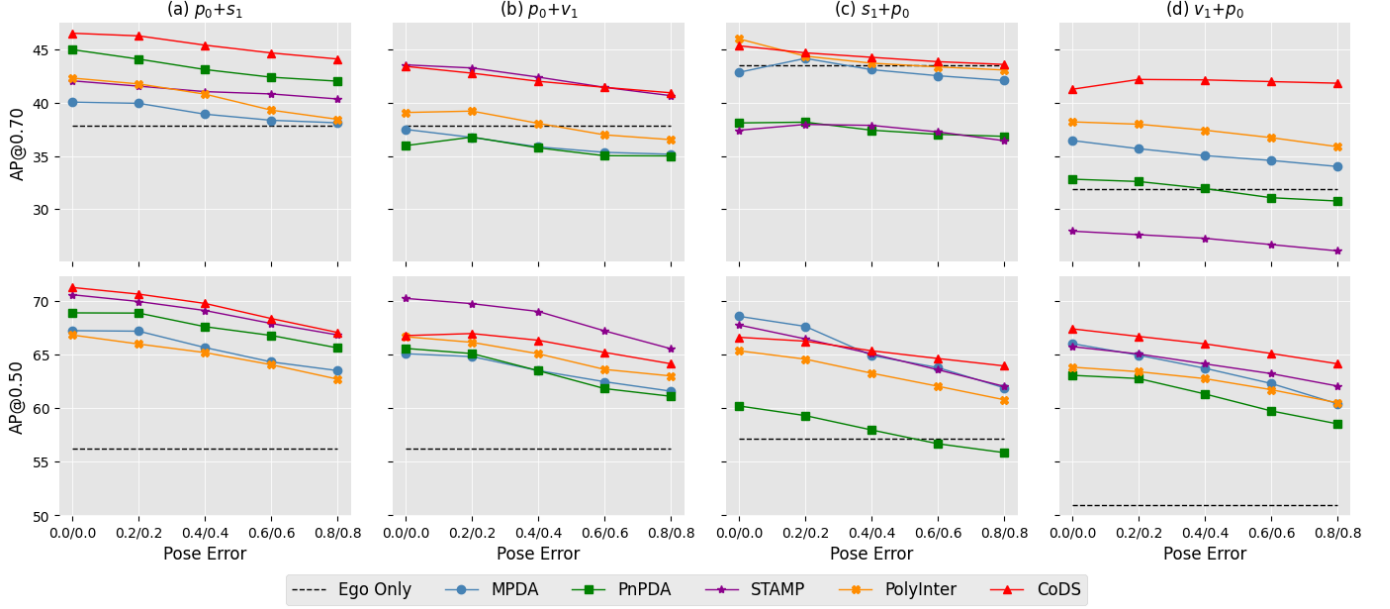


Fig. 8. **Robust Experiment to pose error on V2V4Real.** Pose noise is set to $\mathcal{N}(0, \sigma_p^2)$ on the x , y position and $\mathcal{N}(0, \sigma_r^2)$ on the yaw angle. CoDS achieves state-of-the-art performance under various noisy conditions, consistently surpassing individual perception (ego only).

3) *Localization Error Robustness:* To effectively share valid information, multiple agents require accurate poses to synchronize their individual data within a consistent spatial coordinate system. However, the 6-DoF poses estimated by each agent’s localization module are not always perfect in practice, leading to relative pose inaccuracies. As shown in Fig. 7, adapter-based methods (MPDA and CoDS) detect more objects than the ego-only (no fusion) baseline but suffer from severe bounding box misalignments, with some predictions even deviating farther from the ground-truth than the ego vehicle alone. Therefore, we further evaluate the performance of CoDS and other adapter-based methods in heterogeneous scenarios with localization errors, as illustrated in Fig. 8. To simulate localization errors, Gaussian noise $\mathcal{N}(0, \sigma_p^2)$ is added to the 2D center coordinates x and y , and $\mathcal{N}(0, \sigma_r^2)$ is added to the yaw angle θ , where x , y and θ represent the accurate global pose parameters.

When there is no localization noise, all methods achieve a high AP@0.50. As localization errors increase, the performance of all methods declines, occasionally falling below the accuracy of individual perception (ego only). This will affect

the safety of autonomous driving. However, CoDS consistently outperforms the other methods and maintains higher performance than individual perception. This is because, under the guidance of DAMI loss, CoDS is still able to capture task-related information even in the presence of localization errors.

D. Qualitative Evaluation

1) *Visualization of Feature Maps:* Fig. 9 illustrates the feature maps before and after alignment by CoDS. Before alignment, there are substantial differences in the original semantics of ego and neighbor features. For PointPillars, the foreground regions on the feature map exhibit relatively higher values, whereas for SECOND, the foreground regions on the feature map exhibit relatively lower values.

However, after processing with CoDS, the patterns of ego and neighbor features become noticeably more similar, exhibiting consistent color characteristics. All aligned features emphasize the object regions, demonstrating that CoDS effectively removes encoder-specific information while capturing task-related information. This highlights the effectiveness of CoDS in addressing distribution discrepancies.

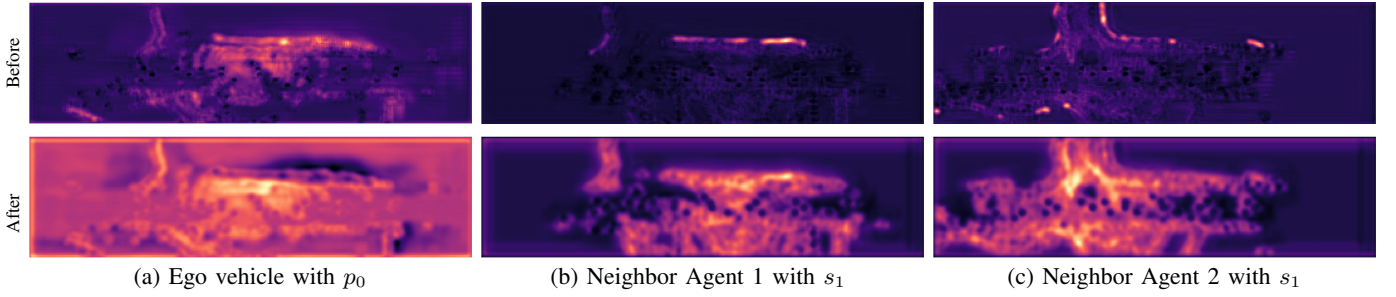


Fig. 9. **Visualization of intermediate features before and after alignment.** Ego and neighbor agents use PointPillars and SECOND encoders, respectively. After processing by CoDS, the heterogeneous features exhibit similar semantic characteristics.

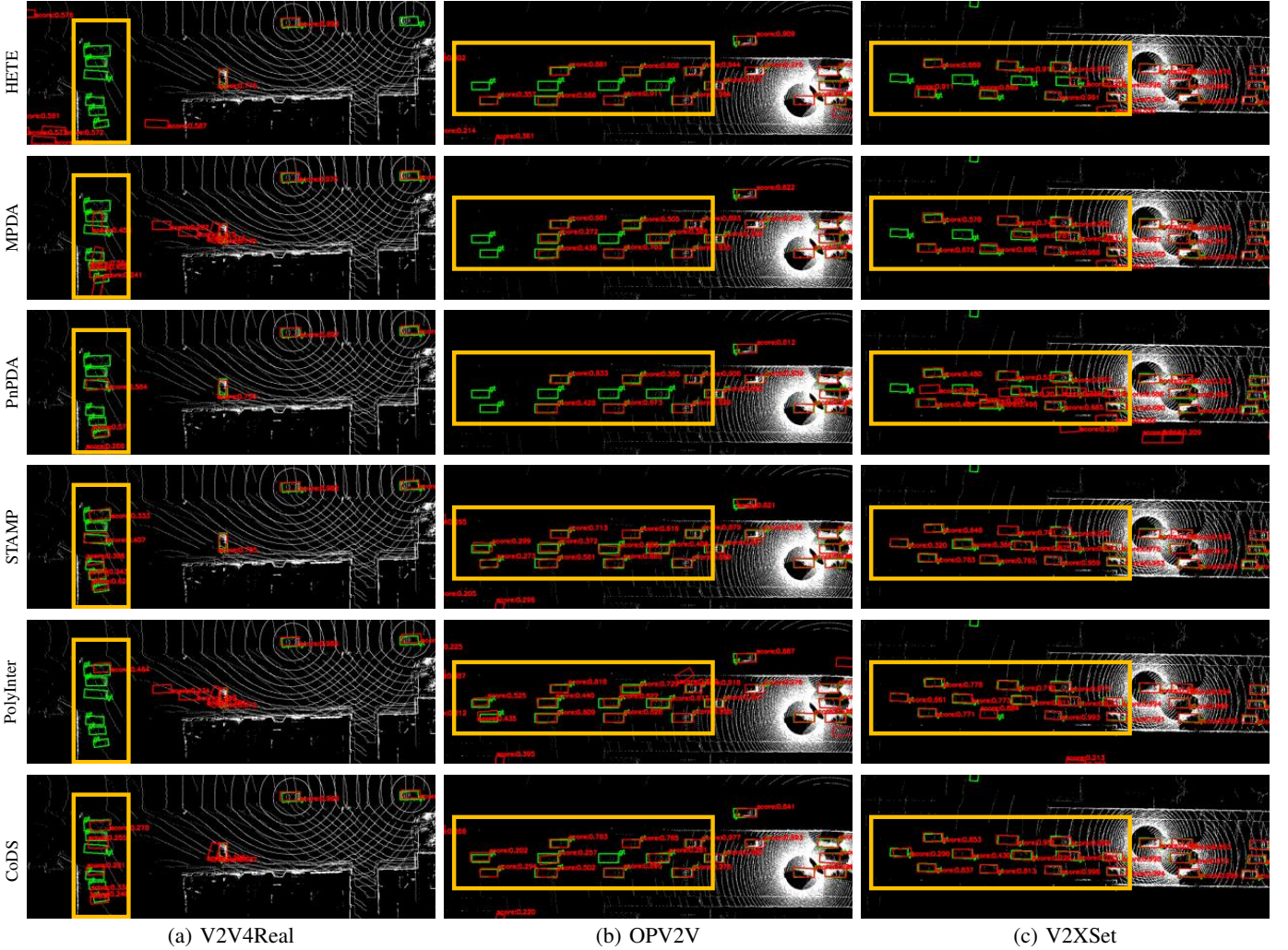


Fig. 10. **Visualization of detection results using different collaborative methods.** The green and red boxes represent the ground truth and the detection results by different collaboration methods, respectively. The proposed CoDS achieves the most detection performance across various datasets.

2) *Visualization of Detection Results:* We visualize the detection results of different methods across three datasets, where the ego and neighbor agents utilize PointPillars and SECOND encoders, respectively. As shown in Fig. 10, when HETE is directly applied for collaboration, the detector misses numerous objects and even produces significant false detections in V2V4Real. This highlights the negative impact of feature discrepancies on intermediate collaboration.

Collaborative detectors employing various adapter-based methods demonstrate improved object detection. However,

due to ambiguities in feature fusion caused by distribution differences, the detectors using MPDA and PnPDA fail to identify certain regions, while STAMP and PolyInter tend to produce false detections. In contrast, our CoDS method significantly reduces these missed detections, demonstrating its effectiveness in addressing distribution discrepancies.

E. Ablation Studies

1) *Contribution of Components:* We conduct ablation studies to evaluate the effectiveness of the proposed LSCR module

TABLE VI
ABLATION STUDY OF THE COMPONENTS ON V2V4REAL.

Encoder	F_{LSCR}	F_{DADS}	$\mathcal{L}_{DAMI}$	AP@0.50	AP@0.70
p_0+p_1	-	-	-	50.96	31.19
	✓	-	-	59.96	36.99
	✓	✓	-	60.38	34.51
	✓	-	✓	60.83	37.89
	✓	✓	✓	60.99	37.89
p_0+s_1	-	-	-	41.21	28.84
	✓	-	-	69.41	39.63
	✓	✓	-	68.32	37.95
	✓	-	✓	68.42	37.63
	✓	✓	✓	71.27	46.52
p_0+v_1	-	-	-	33.93	24.13
	✓	-	-	69.57	39.61
	✓	✓	-	68.24	42.50
	✓	-	✓	69.23	39.68
	✓	✓	✓	69.78	43.41

F_{LSCR} , DADS module F_{DADS} and DAMI loss $\mathcal{L}_{DAMI}$. We take HETE as a baseline and gradually incorporate each component. As shown in Table VI, using F_{LSCR} effectively addresses the dimension discrepancy issue and improves the AP@0.70 over the baseline over 18%. However, when F_{LSCR} is combined with F_{DADS} without $\mathcal{L}_{DAMI}$, the performance decreases due to a lack of distribution alignment guidance. When F_{LSCR} is combined with $\mathcal{L}_{DAMI}$, the resizer learns domain-invariant features, but it does not fully address the distribution issues. Only by combining F_{LSCR} , F_{DADS} and $\mathcal{L}_{DAMI}$ can we alleviate the discrepancy issue, which results in the highest improvement in AP@0.70.

2) *Contribution of Domain Separation Modules:* We also analyze the effectiveness of the encoder-specific and encoder-agnostic domain separation modules in DADS. As shown in Table VII, using either encoder-agnostic or encoder-specific modules alone does not yield satisfactory results. This is because the encoder-agnostic modules fail to project features into a common space due to interference from encoder-specific information. Similarly, the encoder-specific modules alone cannot completely eliminate domain-dependent information, which may hinder distribution alignment. Furthermore, utilizing two encoder-specific modules does not achieve comparable performance to the combination of encoder-specific and encoder-agnostic modules, indicating the necessity of weight sharing among the second modules. Finally, we examine whether additional encoder-specific modules would help remove domain-dependent information. The results show that a single encoder-specific module is sufficient to remove such information, while additional encoder-specific modules may lead to overfitting and decreased performance. Furthermore, additional encoder-agnostic modules will not improve feature alignment. Because deeper weight-sharing layers over-compress information and over-smooth features, which hampers fine-grained detection.

TABLE VII
ABLATION STUDY OF DOMAIN SEPARATION MODULES ON V2V4REAL.

Encoder	DADS modules	AP@0.50	AP@0.70
p_0+p_1	M^{ea}	2.61	1.15
	M^{es}	58.91	35.48
	$M^{es}+M^{es}$	48.47	27.75
	$M^{es}+M^{ea}$	60.99	37.89
	$2*M^{es}+M^{ea}$	58.85	31.14
	$M^{es}+2*M^{ea}$	0.46	36.88
p_0+s_1	M^{ea}	26.03	3.67
	M^{es}	54.23	29.88
	$M^{es}+M^{es}$	56.91	27.68
	$M^{es}+M^{ea}$	71.27	46.52
	$2*M^{es}+M^{ea}$	56.52	33.86
	$M^{es}+2*M^{ea}$	67.26	40.58
p_0+v_1	M^{ea}	47.61	11.01
	M^{es}	62.67	39.35
	$M^{es}+M^{es}$	58.92	31.14
	$M^{es}+M^{ea}$	69.78	43.41
	$2*M^{es}+M^{ea}$	66.62	34.37
	$M^{es}+2*M^{ea}$	67.02	36.01

VI. CONCLUSION

In this paper, we propose CoDS, a fully convolutional collaborative perception adapter to address feature discrepancies in heterogeneous scenarios through domain separation. Specifically, the CoDS incorporates the LSCR to align feature dimensions, followed by the DADS module, which removes encoder-specific information while preserving task-relevant information. During training, CoDS employs DAMI loss to further enhance the domain separation process. Extensive experiments on the V2V4Real, OPV2V and V2XSet datasets demonstrate that CoDS effectively mitigates feature discrepancies and consistently achieves an optimal balance between detection accuracy and inference efficiency.

REFERENCES

- [1] X. Ye, K. Qu, W. Zhuang, and X. Shen, "Accuracy-aware cooperative sensing and computing for connected autonomous vehicles," *IEEE Transactions on Mobile Computing (TMC)*, vol. 23, no. 8, pp. 8193–8207, 2023.
- [2] Z. Fang, S. Hu, H. An, Y. Zhang, J. Wang, H. Cao, X. Chen, and Y. Fang, "PACP: priority-aware collaborative perception for connected and autonomous vehicles," *IEEE Transactions on Mobile Computing (TMC)*, vol. 23, no. 12, pp. 15 003–15 018, 2024.
- [3] Z. Xiao, J. Shu, H. Jiang, G. Min, J. Liang, and A. Iyengar, "Toward collaborative occlusion-free perception in connected autonomous vehicles," *IEEE Transactions on Mobile Computing (TMC)*, vol. 23, no. 5, pp. 4918–4929, 2023.
- [4] X. Zhou, C. Wang, Q. Xie, and T. Qiu, "V2I-Coop: Accurate object detection for connected automated vehicles at accident black spots with v2i cross-modality cooperation," *IEEE Transactions on Mobile Computing (TMC)*, pp. 1–14, 2024.
- [5] L. Zhao, E. Zhang, S. Wan, A. Hawbani, A. Y. Al-Dubai, G. Min, and A. Y. Zomaya, "MESON: a mobility-aware dependent task offloading scheme for urban vehicular edge computing," *IEEE Transactions on Mobile Computing (TMC)*, vol. 23, no. 5, pp. 4259–4272, 2023.
- [6] H. Qian, L. Zhao, A. Hawbani, Z. Liu, K. Yu, Q. He, and Y. Bi, "Collaborative overtaking strategy for enhancing overall effectiveness of mixed connected and connectionless vehicles," *IEEE Transactions on Mobile Computing (TMC)*, vol. 23, no. 12, pp. 13 556–13 572, 2024.

- [7] C. Wang, J. Peng, L. Cai, H. Peng, W. Liu, X. Gu, and Z. Huang, "Ai-enabled spatial-temporal mobility awareness service migration for connected vehicles," *IEEE Transactions on Mobile Computing (TMC)*, vol. 23, no. 4, pp. 3274–3290, 2023.
- [8] X. Zhou, S. Ge, P. Liu, and T. Qiu, "Dag-based dependent tasks offloading in mec-enabled iot with soft cooperation," *IEEE Transactions on Mobile Computing (TMC)*, vol. 23, no. 6, pp. 6908–6920, 2023.
- [9] Q. Xie, X. Zhou, T. Hong, W. Hu, W. Qu, and T. Qiu, "Towards communication-efficient cooperative perception via planning-oriented feature sharing," *IEEE Transactions on Mobile Computing (TMC)*, pp. 1–14, 2024.
- [10] R. Xu, X. Xia, J. Li, H. Li, S. Zhang, Z. Tu, Z. Meng, H. Xiang, X. Dong, R. Song *et al.*, "V2V4Real: A real-world large-scale dataset for vehicle-to-vehicle cooperative perception," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 13 712–13 722.
- [11] 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 *IEEE International Conference on Robotics and Automation (ICRA)*, 2022, pp. 2583–2589.
- [12] H. Yu, Y. Luo, M. Shu, Y. Huo, Z. Yang, Y. Shi, Z. Guo, H. Li, X. Hu, J. Yuan *et al.*, "DAIR-V2X: A large-scale dataset for vehicle-infrastructure cooperative 3d object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 21 361–21 370.
- [13] 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 (ECCV)*, 2022, pp. 107–124.
- [14] 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 (NeurIPS)*, vol. 35, 2022, pp. 4874–4886.
- [15] J. Zhang, K. Yang, Y. Wang, H. Wang, P. Sun, and L. Song, "ERMVP: Communication-efficient and collaboration-robust multi-vehicle perception in challenging environments," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024, pp. 12 575–12 584.
- [16] D. Yang, K. Yang, Y. Wang, J. Liu, Z. Xu, R. Yin, P. Zhai, and L. Zhang, "How2comm: Communication-efficient and collaboration-pragmatic multi-agent perception," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 36, 2024.
- [17] Z. Chen, Y. Shi, and J. Jia, "Transiff: An instance-level feature fusion framework for vehicle-infrastructure cooperative 3d detection with transformers," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023, pp. 18 205–18 214.
- [18] X. Li, J. Yin, W. Li, C. Xu, R. Yang, and J. Shen, "DI-V2X: Learning domain-invariant representation for vehicle-infrastructure collaborative 3d object detection," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 38, no. 4, 2024, pp. 3208–3215.
- [19] S. Wei, Y. Wei, Y. Hu, Y. Lu, Y. Zhong, S. Chen, and Y. Zhang, "Asynchrony-robust collaborative perception via bird's eye view flow," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 36, 2024.
- [20] H. Yu, Y. Tang, E. Xie, J. Mao, P. Luo, and Z. Nie, "Flow-based feature fusion for vehicle-infrastructure cooperative 3d object detection," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 36, 2024.
- [21] Y. Lu, Q. Li, B. Liu, M. Dianati, C. Feng, S. Chen, and Y. Wang, "Robust collaborative 3d object detection in presence of pose errors," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 4812–4818.
- [22] J. Gu, J. Zhang, M. Zhang, W. Meng, S. Xu, J. Zhang, and X. Zhang, "Feaco: Reaching robust feature-level consensus in noisy pose conditions," in *Proceedings of the ACM International Conference on Multimedia (MM)*, 2023, pp. 3628–3636.
- [23] Z. Huang, S. Wang, Y. Wang, W. Li, D. Li, and L. Wang, "RoCo: Robust cooperative perception by iterative object matching and pose adjustment," in *Proceedings of the ACM International Conference on Multimedia (MM)*, 2024, pp. 7833–7842.
- [24] D. Feng, A. Harakeh, S. L. Waslander, and K. Dietmayer, "A review and comparative study on probabilistic object detection in autonomous driving," *IEEE Transactions on Intelligent Transportation Systems (TITS)*, vol. 23, no. 8, pp. 9961–9980, 2021.
- [25] Z. Hao, J. Guo, K. Han, Y. Tang, H. Hu, Y. Wang, and C. Xu, "One-for-all: Bridge the gap between heterogeneous architectures in knowledge distillation," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 36, 2024.
- [26] R. Xu, J. Li, X. Dong, H. Yu, and J. Ma, "Bridging the domain gap for multi-agent perception," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 6035–6042.
- [27] Y. Lu, Y. Hu, Y. Zhong, D. Wang, Y. Wang, and S. Chen, "An extensible framework for open heterogeneous collaborative perception," in *International Conference on Learning Representations (ICLR)*, 2024.
- [28] C. Shao, G. Luo, Q. Yuan, Y. Chen, Y. Liu, K. Gong, and J. Li, "Hete-cooper: Feature collaboration graph for heterogeneous collaborative perception," in *European Conference on Computer Vision (ECCV)*, 2024, pp. 162–178.
- [29] T. Luo, Q. Yuan, G. Luo, Y. Xia, Y. Yang, and J. Li, "Plug and play: A representation enhanced domain adapter for collaborative perception," in *European Conference on Computer Vision (ECCV)*, 2024, pp. 287–303.
- [30] X. Gao, R. Xu, J. Li, Z. Wang, Z. Fan, and Z. Tu, "STAMP: Scalable task-and model-agnostic collaborative perception," in *International Conference on Learning Representations (ICLR)*, 2025.
- [31] Y. Xia, Q. Yuan, G. Luo, X. Fu, Y. Li, X. Zhu, T. Luo, S. Chen, and J. Li, "One is Plenty: A polymorphic feature interpreter for immutable heterogeneous collaborative perception," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025, pp. 1592–1601.
- [32] K. Bousmalis, G. Trigeorgis, N. Silberman, D. Krishnan, and D. Erhan, "Domain separation networks," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 29, 2016.
- [33] Y. Tian, D. Krishnan, and P. Isola, "Contrastive multiview coding," in *European Conference on Computer Vision (ECCV)*, 2020, pp. 776–794.
- [34] L. Smith and M. Gasser, "The development of embodied cognition: Six lessons from babies," *Artificial life*, vol. 11, no. 1-2, pp. 13–29, 2005.
- [35] H. E. Den Ouden, P. Kok, and F. P. De Lange, "How prediction errors shape perception, attention, and motivation," *Frontiers in psychology*, vol. 3, p. 548, 2012.
- [36] Y. Han, H. Zhang, H. Li, Y. Jin, C. Lang, and Y. Li, "Collaborative perception in autonomous driving: Methods, datasets, and challenges," *IEEE Intelligent Transportation Systems Magazine (ITSM)*, vol. 15, no. 6, pp. 131–151, 2023.
- [37] H. Zhang, G. Luo, Y. Li, and F.-Y. Wang, "Parallel vision for intelligent transportation systems in metaverse: Challenges, solutions, and potential applications," *IEEE Transactions on Systems, Man, and Cybernetics: Systems (TSMC)*, vol. 53, no. 6, pp. 3400–3413, 2022.
- [38] Y. Han, H. Zhang, H. Zhang, and Y. Li, "SSC3OD: Sparsely supervised collaborative 3d object detection from lidar point clouds," in *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2023, pp. 1360–1367.
- [39] Y. Han, H. Zhang, H. Zhang, J. Wang, and Y. Li, "CoDTS: Enhancing sparsely supervised collaborative perception with a dual teacher-student framework," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 39, no. 3, 2025, pp. 3366–3373.
- [40] E. J. Roh, H. Baek, D. Kim, and J. Kim, "Fast quantum convolutional neural networks for low-complexity object detection in autonomous driving applications," *IEEE Transactions on Mobile Computing (TMC)*, pp. 1–12, 2024.
- [41] D. Li, J. Xu, Z. Yang, Q. Ma, L. Zhang, and P. Chen, "Leovr: Motion-inspired visual-lidar fusion for environment depth estimation," *IEEE Transactions on Mobile Computing (TMC)*, vol. 23, no. 6, pp. 7499–7516, 2023.
- [42] H. Zhang, Y. Tian, K. Wang, W. Zhang, and F.-Y. Wang, "Mask SSD: An effective single-stage approach to object instance segmentation," *IEEE Transactions on Image Processing (TIP)*, vol. 29, pp. 2078–2093, 2019.
- [43] S. Hong, Y. Liu, Z. Li, S. Li, and Y. He, "Multi-agent collaborative perception via motion-aware robust communication network," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024, pp. 15 301–15 310.
- [44] H. Xiang, R. Xu, and J. Ma, "HM-ViT: Hetero-modal vehicle-to-vehicle cooperative perception with vision transformer," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023, pp. 284–295.
- [45] B. Wang, L. Zhang, Z. Wang, Y. Zhao, and T. Zhou, "CORE: Cooperative reconstruction for multi-agent perception," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023, pp. 8710–8720.
- [46] K. Yang, D. Yang, J. Zhang, M. Li, Y. Liu, J. Liu, H. Wang, P. Sun, and L. Song, "Spatio-temporal domain awareness for multi-agent collaborative perception," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023, pp. 23 383–23 392.
- [47] Z. Lei, S. Ren, Y. Hu, W. Zhang, and S. Chen, "Latency-aware collaborative perception," in *European Conference on Computer Vision (ECCV)*, 2022, pp. 316–332.

- [48] 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 (TPAMI)*, 2023.
- [49] S. Chen, M. Harandi, X. Jin, and X. Yang, “Domain adaptation by joint distribution invariant projections,” *IEEE Transactions on Image Processing (TIP)*, vol. 29, pp. 8264–8277, 2020.
- [50] H. Zhang, G. Luo, Y. Cao, X. Wang, Y. Li, and F.-Y. Wang, “Scale-disentangled and uncertainty-guided alignment for domain-adaptive object detection,” *IEEE Transactions on Intelligent Transportation Systems (TITS)*, pp. 1–15, 2024.
- [51] M. I. Belghazi, A. Baratin, S. Rajeshwar, S. Ozair, Y. Bengio, A. Courville, and D. Hjelm, “Mutual information neural estimation,” in *International conference on machine learning (ICML)*. PMLR, 2018, pp. 531–540.
- [52] A. v. d. Oord, Y. Li, and O. Vinyals, “Representation learning with contrastive predictive coding,” *arXiv preprint arXiv:1807.03748*, 2018.
- [53] P. Cheng, W. Hao, S. Dai, J. Liu, Z. Gan, and L. Carin, “Club: A contrastive log-ratio upper bound of mutual information,” in *International conference on machine learning (ICML)*. PMLR, 2020, pp. 1779–1788.
- [54] G. Luo, H. Zhang, Q. Yuan, and J. Li, “Complementarity-enhanced and redundancy-minimized collaboration network for multi-agent perception,” in *Proceedings of the ACM International Conference on Multimedia (MM)*, 2022, pp. 3578–3586.
- [55] W. Su, L. Chen, Y. Bai, X. Lin, G. Li, Z. Qu, and P. Zhou, “What makes good collaborative views? contrastive mutual information maximization for multi-agent perception,” in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 38, no. 16, 2024, pp. 17550–17558.
- [56] H. Talebi and P. Milanfar, “Learning to resize images for computer vision tasks,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 497–506.
- [57] Y. Yao, X. Li, Y. Zhang, and Y. Ye, “Multisource heterogeneous domain adaptation with conditional weighting adversarial network,” *IEEE Transactions on Neural Networks and Learning Systems (TNNLS)*, vol. 34, no. 4, pp. 2079–2092, 2021.
- [58] H. Zhang, G. Luo, J. Li, and F.-Y. Wang, “C2FDA: Coarse-to-fine domain adaptation for traffic object detection,” *IEEE Transactions on Intelligent Transportation Systems (TITS)*, vol. 23, no. 8, pp. 12633–12647, 2021.
- [59] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, “PointPillars: Fast encoders for object detection from point clouds,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 12697–12705.
- [60] Y. Li, S. Ren, P. Wu, S. Chen, C. Feng, and W. Zhang, “Learning distilled collaboration graph for multi-agent perception,” in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 34, 2021, pp. 29541–29552.
- [61] 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 (CoRL)*, 2022.
- [62] Y. Yan, Y. Mao, and B. Li, “SECOND: Sparsely embedded convolutional detection,” *Sensors*, vol. 18, no. 10, p. 3337, 2018.
- [63] Y. Zhou and O. Tuzel, “Voxelnet: End-to-end learning for point cloud based 3d object detection,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 4490–4499.
- [64] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, “A convnet for the 2020s,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, 2022, pp. 11976–11986.