Model-Agnostic Multi-Agent Perception Framework

Runsheng Xu[1*], Weizhe Chen[2*], Hao Xiang[1*], Xin Xia[1], Lantao Liu[2], Jiaqi Ma[1]

***Abstract*— Existing multi-agent perception systems assume that every agent utilizes the same model with identical parameters and architecture. The performance can be degraded with different perception models due to the mismatch in their confidence scores. In this work, we propose a model-agnostic multi-agent perception framework to reduce the negative effect caused by the model discrepancies without sharing the model information. Specifically, we propose a confidence calibrator that can eliminate the prediction confidence score bias. Each agent performs such calibration independently on a standard public database to protect intellectual property. We also propose a corresponding bounding box aggregation algorithm that considers the confidence scores and the spatial agreement of neighboring boxes. Our experiments shed light on the necessity of model calibration across different agents, and the results show that the proposed framework improves the baseline 3D object detection performance of heterogeneous agents. The code can be found at this url.**

I. Introduction

Recent advancements in deep learning have improved the performance of modern perception systems on many tasks, such as object detection [1–3], semantic segmentation [4, 5], and visual navigation [6, 7]. Despite the remarkable progress, single-agent perception systems still have many limitations due to single-view constraints. For instance, autonomous vehicles (AVs) usually suffer from occlusion [8], and such situations are difficult to handle because of the lack of sensory observations of the occluded area. To address this issue, recent studies [9–17] have explored wireless communication technology to enable nearby agents to share the sensory information and collaboratively perceive the surrounding environment.

Although existing fusion frameworks have obtained a significant 3D object detection performance boost, they assume that all the collaborating agents share an identical model with the same parameters. This assumption is hard to satisfy in practice, particularly in autonomous driving. Distributing the model parameters among AVs might raise privacy and confidentiality concerns, especially for vehicles from different automotive companies. Even for AVs from the same company, the detection models can have various versions, depending on the vehicle type and model updating frequency. Without adequately handling the inconsistency, the shared sensory information can have a large domain gap, and the advantage brought by multi-agent perception will be diminished rapidly.

*The first three authors contribute equally.

[1]Runsheng Xu, Hao Xiang, Xin Xia, and Jiaqi Ma are with the University of California, Los Angeles, CA, USA {rxx3386,haxiang,x35xia}@g.ucla.edu, jiaqima@ucla.edu

[2]Weizhe Chen and Lantao Liu are with Indiana University, Bloomington, IN, USA {chenweiz, lantao}@iu.edu

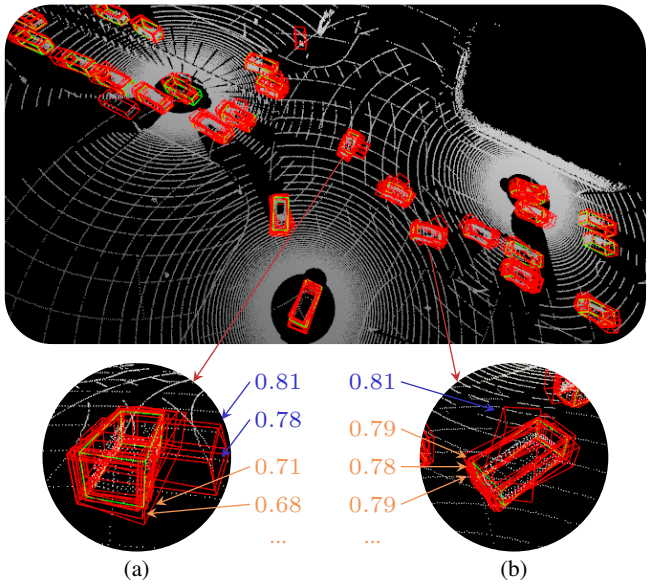


Fig. 1. **Ground truth (green) and bounding box candidates (red) produced by three connected autonomous vehicles**. (a) Some agents have confidence scores that are systematically larger than others, e.g., the blue scores versus the orange scores. However, they might be confidently wrong, which mislead the fusion process. (b) Candidates with slightly lower confidence scores (orange) but higher spatial agreement with neighboring boxes can be better than a singleton with a higher confidence score (blue).

To this end, we propose a model-agnostic multi-agent perception framework to handle the model heterogeneity while maintaining confidentiality. The perception outputs (i.e., detected bounding boxes and confidence scores) are shared to bypass the dependency on the underlying model's detailed information. Due to the distinct models used by the agents, the confidence scores provided by different agents can be systematically misaligned. Some agents may be over-confident, whereas others tend to be under-confident. Directly fusing bounding box proposals from neighboring agents using, for example, Non-Maximum Suppression (NMS)[18] can result in poor detection accuracy due to the presence of over-confident and low-quality candidates.

We propose a simple yet flexible confidence calibrator, called Doubly Bounded Scaling (DBS), to mitigate the misalignment. We also propose a corresponding bounding box aggregation algorithm, named Promote-Suppress Aggregation (PSA), that considers the confidence scores and the spatial agreement of neighboring boxes. Fig. 1 illustrates the importance of these two components. This framework does not reveal model design and parameters, ensuring confidentiality. We evaluate our approach on an open-source large-scale multi-agent perception dataset – OPV2V [12]. Experiments show that in the presence of model discrep-

ancies among agents, our framework significantly improves multi-agent LiDAR-based 3D object detection performance, outperforming the baselines by at least 6% in terms of Average Precision (AP).

II. Related Work

Multi-Agent Perception. Multi-agent perception investigates how to leverage visual cues from neighboring agents through the communication system to enhance the perception capability. There are three categories of existing work according to the information sharing schema: 1) early fusion [8], where raw point clouds are transmitted directly and projected into the same coordinate frame, 2) late fusion [19], where detected bounding boxes and confidence scores are shared, and 3) intermediate fusion [9–12, 14, 20], where compressed latent neural features extracted from point clouds are propagated. Though early fusion has no information loss, it usually requires large bandwidth. Intermediate fusion can achieve a good balance between accuracy and transmission data size, but it requires the complete knowledge of each agent's model, which is non-trivial to satisfy in reality due to intellectual property concerns. On the contrary, late fusion only needs the outputs of the detector without demanding access to the underlying neural networks, which are typically confidential for automotive companies. Therefore, our approach adopts the late fusion strategy but further designs customized new components to address the model discrepancy issue in vanilla late fusion.

3D LiDAR Detection. To tackle the irregular or unstructured data format of point clouds, researchers have come up with point-based, voxel-based, and point-voxel-based methods. Frustum PointNet [21] uses 2D image detection bounding boxes to generate frustums on raw point clouds. Then, we can directly operate the point clouds in the frustums to obtain the final bounding box positions. PointRCNN [22] develops a two-stage framework for 3D detection, which first produces rough bounding box proposals and then fine-tunes them in the second stage. McCraith et al. [23] combines outlier detection [24–26] and PoinNet to make precise predictions. In [1, 27, 28], point clouds are aggregated into voxels and generate latent features per voxel. Such an approach usually follows a one-stage fashion, with less accuracy but lower inference latency than the two-stage methods. [29, 30] integrate both voxel-based network and PointNet-based [31] set abstraction to produce more robust point cloud features, which can keep high learning efficiency while enjoying flexible receptive fields of the PointNet-based networks.

Confidence Calibration. For a probabilistic classifier, the probability associated with the predicted class label should reflect its correctness likelihood. However, many modern neural networks do not have such property [32]. Confidence calibration aims to endow a classifier with such property. Calibration methods can be tightly coupled with the neural networks, such as Bayesian neural networks and regularization techniques [33–35], or serve as a post-processing step. Post-processing methods include histogram binning

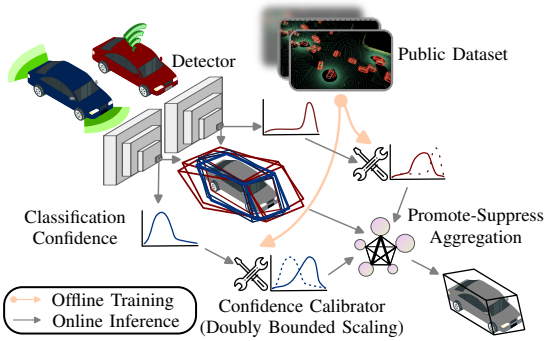


Fig. 2. **Overview of the proposed framework**. Each agent trains its confidence calibrator (i.e., Doubly Bounded Scaling) on the same public dataset offline (orange arrows). Promote-Suppress Aggregation yields the final detection result, considering the spatial information and calibrated confidence of bounding boxes given by connected autonomous vehicles.

methods [36], scaling methods [37, 38], and mixtures[39] that combine the first two branches. Due to the popularity of the Temperature Scaling method [32] which is a single-parameter version of Platt Scaling [37], scaling methods are widely adopted for calibrating neural networks. Our proposed method follows the same fashion.

Bounding Box Aggregation. Object detection models typically require bounding box aggregation to lump the proposals corresponding to the same object. The *de facto* standard post-processing method is Non-Maximum Suppression (NMS) [18, 40], which sequentially selects the proposals with the highest confidence score and then suppresses other overlapped boxes. NMS does not fully exploit information in the proposals because it only uses the relative order of confidence, ignoring the absolute confidence scores and the spatial information hidden in the bounding box coordinates. Several works have been proposed to refine the box aggregation strategies. Soft-NMS [41] softly decays the confidence scores of the proposals proportional to the degree of overlap. In [40] NMS can be learned by a neural network to achieve better occlusion handling and bounding box localization. Adaptive NMS [42] applies a dynamic suppression threshold to an instance according to the target object density. Rothe et al. [43] formulate NMS as a clustering problem and use Affinity Propagation Clustering to solve the problem. The idea of message passing between proposals is related to the aggregation algorithm introduced in Section III-C, but our update rules are simpler and more efficient.

III. Methodology

In this paper, we consider the cooperative perception of a heterogeneous multi-agent system, where agents communicate to share sensing information from different perception models without revealing model information, i.e., model-agnostic collaboration. We focus on a 3D LiDAR detection task in autonomous driving, but the methodology can also be customized and used in other cooperative perception applications. Our goal is to develop a robust framework to handle the heterogeneity among agents while preserving confidentiality. The proposed model-agnostic collaborative perception framework is shown in Fig. 2, which can be

divided into two stages. In the offline stage, we train a model-specific calibrator. During the online phase, real-time on-road sensing information is calibrated and aggregated.

A. Model-Agnostic Fusion Pipeline

Agents with distinct perception models usually generate systematically different confidence. The mismatch in confidence distributions can affect the fusion performance. For instance, an inferior model may be over-confident and dominate the aggregation process, decreasing the accuracy of the final results.

To address the issue, we train a calibrator offline for each model, aligning its confidence score with its empirical accuracy on a calibration dataset. First, each model runs its well-trained detector on a public dataset to produce a model-specific dataset with labels and confidence scores. The public dataset, like nuScenes [44] or Waymo open dataset [45], should be independent of the manufacturer and sensor setup, serving only to test the model's performance. The calibration dataset is then used to train the calibrator (see Section III-B for more details). After training, the calibrator is saved locally for each agent.

When the vehicle is driving on-road and making predictions from the sensor measurements, the calibrator will align the predicted confidence score towards the same standard, thus alleviating the aforementioned mismatch. Then the bounding box coordinates and calibrated confidence scores are packed together and transmitted to neighboring agents. The receiving agent (i.e., ego vehicle) will fuse the shared information via the Promote-Suppress Aggregation algorithm (see Section III-C for details) to output the final results. Since each agent learns its calibrator independently in the offline stage and only shares the detection outputs during the online phase, the detector architecture and parameters are invisible to other agents, protecting the intellectual property.

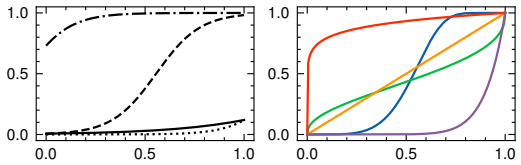
B. Classification Confidence Calibration

To eliminate the bias brought by the system heterogeneity, the models need to be *well-calibrated*. If the confidence scores can imply the likelihood of correct prediction, for example, 80% confidence leads to 80% accurate predictions, this model is *well-calibrated*. Formally, let $\tilde{s}$ be the confidence score produced by the model and $y \in \{0, 1\}$ be the label indicating vehicle or background[1]. A model is *well-calibrated* if its confidence score $\tilde{s}$ matches the expectation of correctly predicting the label:

$$\mathbb{E}[y = 1 \mid \tilde{s}] = \tilde{s}. \tag{1}$$

Scaling-Based Confidence Calibration. The goal of scaling-based confidence calibration is to learn a parametric scaling function (i.e., calibrator) $\mathsf{c}_{\boldsymbol{\theta}}(\tilde{s}) : [0, 1] \mapsto [0, 1]$ on a calibration dataset to transform the uncalibrated confidence scores $\tilde{s}$ into well-calibrated ones s. Given a calibration set $\mathcal{D} \triangleq \{(\tilde{s}_n, y_n)\}_{n=1}^{N}$ containing the model-dependent confidence scores $\tilde{s}$ and ground-truth labels y, we optimize

[1]We discuss binary classification here for simplicity but the proposed framework can be generalized to the multi-class case.



(a) Logistic (b) Kumaraswamy

Fig. 3. **Scaling functions with various parameters** that follow (a) the logistic form and (b) the Kumaraswamy CDF. Note that, in (b), the "inverse-sigmoid" shape (green curve, $a = 0.4, b = 0.4$) and the identity map (orange curve, $a = 1, b = 1$) are not in the logistic family.

the parameters $\boldsymbol{\theta}$ of the calibrator $\mathsf{c}_{\boldsymbol{\theta}}(\tilde{s})$ by gradient descent on the binary cross entropy loss

$$\ell_{CE} = -y_n \log(s_n) - (1 - y_n) \log(1 - s_n), \tag{2}$$

where $s_n = \mathsf{c}_{\boldsymbol{\theta}}(\tilde{s}_n)$. Training a parametric function by optimizing Eq. (2) is similar to standard binary classification, however, extra constraints are required on the scaling function for confidence calibration . Designing a suitable calibrator for our application requires satisfying three conditions: (a) The scaling function needs to be *monotonically non-decreasing* as a higher confidence score is supposed to indicate a higher expected accuracy; (b) The scaling function should be relatively smooth to avoid over-fitting to the calibration set; (c) The scaling function is supposed to be *doubly bounded*, meaning that it maps a confidence interval $[0, 1]$ to the same $[0, 1]$ range. In the following subsections, we will explain why the commonly used calibration methods do not meet all these conditions, which motivates the development of our proposed calibrator.

Platt Scaling and Temperature Scaling. The most popular scaling methods are arguably Platt Scaling [37] and Temperature Scaling [32]. Platt Scaling uses the logistic family as the calibrator:

$$\mathsf{c}_{\text{Platt}}(\tilde{s}; a, b) = \frac{1}{1 + \exp\left(-(a \times \tilde{s} + b)\right)}, \tag{3}$$

where a, b are parameters with $a \geq 0$ to ensure that the calibration map is *monotonically non-decreasing*. Temperature Scaling is a special case of Eq. (3) where b is fixed to 0. Fig. 3 shows several scaling functions from this family. Platt Scaling can fail if its parametric assumptions are not met [46]. For example, we cannot learn an "inverse-sigmoid" (see the green curve in Fig. 3b) scaling function within this family. Furthermore, the identity function is also not a member of the logistic family. In addition to the aforementioned limitations, the logistic family is also not a function family that can naturally map $[0, 1]$ to $[0, 1]$ as its input domain is $\mathbb{R}$, therfore, these popular choices are not our ideal candidates.

Doubly Bounded Scaling (DBS). We propose to use the Kumaraswamy Cumulative Density Function (CDF) [47] that meets all the three constraints while being sufficiently flexible. To the best of our knowledge, this is the first time that this function family has been adopted in confidence calibration. Specifically, we learn a scaling function with the

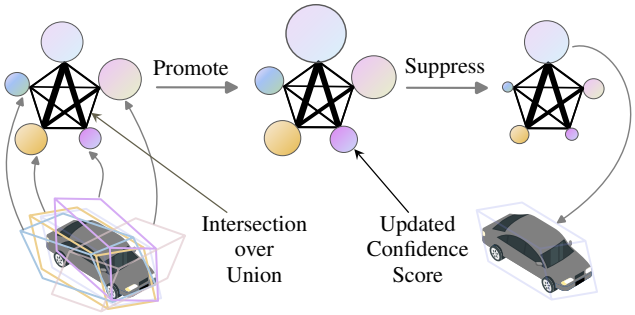


Fig. 4. **Illustration of Promote-Suppress Aggregation**. The size of a node indicates the confidence score of the bounding box and the edge width represents the Intersection-over-Union of two boxes.

following form

$$\mathtt{c}(\tilde{s}; a, b) = 1 - (1 - \tilde{s}^a)^b \,, \tag{4}$$

where $a > 0$ and $b > 0$ are the parameters. Scaling functions that follow Eq. (4) are *monotonically non-decreasing*, *smooth*, and *doubly bounded*, hence the name. we can see that DBS is more flexible than the logistic form by comparing Fig. 3a and Fig. 3b. For each detector, we optimize the a and b on a calibration dataset by minimizing Eq. (2).

C. Promote-Suppress Aggregation (PSA)

Detection models typically output a bunch of overlapped bounding box candidates for the same detected object, thus we need a post-processing step to select from these candidates. In most of the detection algorithms, the optimization objective function is a summation of a bounding box regression loss and a classification loss. The detector can express its "confidence" by assigning high classification scores to the promising bounding boxes or allocating more bounding boxes to the region that it finds relevant features. To select the high-score bounding boxes with many confident neighbors, we propose Promote-Suppress Aggregation (PSA), which takes into account both the regression and classification confidences.

Fig. 4 illustrates the idea of PSA. We first construct a spatial graph of bounding box candidates based on Intersection-over-Union (IoU) values and the confidence scores. In the promotion step, the IoU weighted confidence scores are propagated to the neighboring nodes. We design the propagation rule to meet the following desiderata:

- A candidate should be promoted if many other candidate boxes have *large intersections* with it;
- A candidate with many *high-score neighbors* should be promoted;
- If possible, the update rules should be *parallelizable* and *permutation-invariant*. Namely, the propagation order does not change the result.

In the suppression step, the candidate with the highest updated score will softly suppress the scores of other candidates. Finally, we select one or more bounding boxes that rank in the first (few) places. The idea of soft suppression and selecting more than one candidate is akin to soft-NMS [41], which is beneficial when the bounding box of a small object is within the box of a large object. Below we formally describe the PSA algorithm.

Algorithm 1 Promote-Suppress Aggregation

Arguments: bounding boxes $\mathbf{B} = [\mathbf{b}_1, \ldots, \mathbf{b}_N]^\intercal$,
confidence score vector $\mathbf{s} = [s_1, \ldots, s_N]^\intercal$,
soft selection parameters ε, and threshold ϕ

1: Initialize selected box indices to an empty set $\mathcal{I} = \emptyset$
2: Compute IoU matrix $\mathbf{U} \in [0,1]^{N \times N}$ using $\mathbf{B}$
3: Find vertex indices of connected components $\mathcal{C} \triangleq \{\mathbf{c}_m\}_{m=1}^{M}$
4: **for** each $\mathbf{c}_m \in \mathcal{C}$ **do**
5: Extract IoU sub-matrix $\mathbf{U}_m \in [0,1]^{N_m \times N_m}$ via $\mathbf{c}_m$
6: Extract score sub-vector $\mathbf{s}_m \in [0,1]^{N_m}$ via $\mathbf{c}_m$
7: $\hat{\mathbf{s}}_m = \mathbf{U}_m \mathbf{s}_m$ ▷ Promote
8: $\bar{\mathbf{s}}_m = \mathtt{softmax}(\hat{\mathbf{s}}_m / \varepsilon)$ ▷ Suppress
9: $\mathcal{I} = \mathcal{I} \cup \{c_m^{(n)} \mid \bar{s}_m^{(n)} > \phi, n = 1, \ldots, N_m\}$ ▷ Select
10: **return** selected candidate indices $\mathcal{I}$

Definition 1 (Bounding Box Graph)**.** Let $\mathcal{G}$ be a weighted graph with a set of edges $\mathcal{E}$ and a set of nodes/vertices $\mathcal{V}$, where each vertex $v \in \mathcal{V}$ represents a bounding box candidate b with an associated confidence score s after calibration. The edge weigh w_{ij} between vertex v_i and v_j is defined as the Intersection-over-Union value $\mathtt{IoU}(b_i, b_j) \triangleq \cap(b_i, b_j) / \cup(b_i, b_j)$. An edge connects vertex v_i and v_j if the edge weight is non-zero.

Definition 2 (Connected Components)**.** The graph consists of a number of *connected components* in which every pair of nodes is connected via a sequence of edges.

Problem 1 (Bounding Box Aggregation)**.** Given the Intersection-over-Union matrix $\mathbf{U} \in [0,1]^{N \times N}$ among N bounding box candidates $\mathbf{B} = [\mathbf{b}_1, \ldots, \mathbf{b}_N]^\intercal$ and their confidence scores $\mathbf{s} = [s_1, \ldots, s_N]^\intercal$, our goal is to compute an index set $\mathcal{I}$ to select/filter candidates that best match the ground-truth bounding boxes.

Algorithm 1 shows how PSA computes the index set. Given the IoU adjacency matrix, we can find out the indices of each component and put them into a component set $\mathcal{C} = \{\mathbf{c}_m\}_{m=1}^{M}$, where M is the number of components and $\mathbf{c}_m$ contains the indices of N_m vertices (line 3). For each component, we extract the IoU matrix $\mathbf{U}_m \in [0,1]^{N_m \times N_m}$ and confidence score vector $\mathbf{s}_m \in [0,1]^{N_m}$ corresponding to this component (line 5-6). Then, we perform the promotion step $\hat{\mathbf{s}}_m = \mathbf{U}_m \mathbf{s}_m$ where each vertex updates its score to be the IoU-weighted sum of scores from other vertices in the component (line 7). In the suppression step, we normalize the updated scores back to $[0,1]$ and distill the winning candidate via $\bar{\mathbf{s}}_m = \mathtt{softmax}(\hat{\mathbf{s}}_m / \varepsilon)$ (line 8). In the end, indices with updated scores larger than a threshold are added to the set $\mathcal{I}$ (line 9). We can select multiple candidates if $\varepsilon \in (0,1]$ is large and ϕ is small. In our application, however, one component typically contains only one object/vehicle, so we use a small ε and $\phi = 0.5$. Overall, PSA is highly

TABLE I

OBJECT DETECTION PERFORMANCE. AVERAGE PRECISION (AP) AT IOU=0.7 ON *Homo*, *Hetero1*, AND *Hetero2* SETTING.

Methods	Homo ↑AP@0.7	Hetero1 ↑AP@0.7	Hetero2 ↑AP@0.7
No fusion	0.602	0.602	0.602
Intermediate w/o calibration	**0.815**	0.677	0.571
Late fusion w/o calibration	0.781	0.691	0.723
Our method	0.813	**0.750**	**0.784**

TABLE II

COMPONENT ABLATION STUDY.

Components DBS	PSA	Hetero1 ↑AP@0.7	Hetero2 ↑AP@0.7
		0.691	0.723
✓		0.734	0.776
✓	✓	**0.750**	**0.784**

parallelizable as each component operates independently and each step only requires simple linear search or small matrix-vector multiplication.

IV. EXPERIMENTS

A. Dataset

We evaluate the proposed framework on a large-scale open-source multi-agent perception dataset OPV2V [12], which is simulated using the high-fidelity simulator CARLA [48] and a cooperative driving automation simulation framework OpenCDA [49]. It includes 73 scenarios with an average of 25 seconds duration. In each scene, various numbers (2 to 7) of Autonomous Vehicles (AVs) provide LiDAR point clouds from their viewpoints. The train/validation/test splits are $6764/1981/2169$ frames, respectively. For details of the dataset, please refer to [12].

B. Experiment Setup

Evaluation metric. Following [49], we evaluate the detection accuracy in the range of $x \in [-140, 140]$m and $y \in [-40, 40]$m, centered at the ego-vehicle coordinate frame. The detection performance is measured with Average Precision (AP) at $IoU = 0.7$.

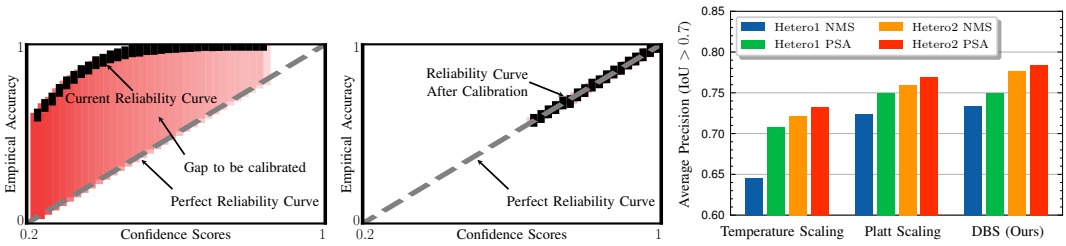
Evaluation setting. We evaluate our method under three different settings: 1) *Homo Setting*, where the detectors of agents are homogeneous with the same architecture and trained parameters. This setting has no confidence distribution gap and is used to demonstrate the performance drop when taking heterogeneity into account; 2) *Hetero Setting 1*, where the agents have the same model architecture but different parameters; 3) *Hetero Setting 2*, where the detector architectures are disparate. For *Homo Setting*, we select pre-trained Pointpillar [50] as the backbone for all the AVs. For *Hetero Setting 1*, the ego vehicle employs the same pre-trained Pointpillar model as in *Homo Setting*, whereas other AVs pick the parameters of Pointpillar from a different epoch during training. Likewise, in the *Hetero Setting 2*, the ego vehicle utilizes Pointpillar while other AVs use SECOND [28] for detection. As intermediate fusion requires equal feature map resolution, we apply simple bi-linear interpolation under this setting. The ego vehicle uses the identical model with the same parameters across all settings for the *No Fusion* and *Late Fusion*. To compare with existing calibrators, we use the same calibration method for all agents, but the parameters are agent-specific. The proposed framework should also work even when the calibration methods across agents are heterogeneous, as long as the prediction bias is effectively reduced.

Compared methods. We regard *No Fusion* as the baseline, which only takes the ego vehicle's LiDAR data as input and omits any collaboration. Ideally, the multi-agent system should at least outperform this baseline. To validate the necessity of the calibration, we compare our method with naive late fusion and intermediate fusion that ignore calibrations. The naive late fusion gathers all detected bounding box positions and confidence scores together and simply applies NMS to produce the final results. The intermediate fusion method is the same as the one in [12]. We exclude the early fusion in the comparison as it requires large bandwidth, which leads to high communication delay thus is impractical to be deployed in the real world. Moreover, we also compare the proposed Doubly Bounded Scaling (DBS) with two other commonly used scaling-based calibrators: Temperature Scaling (TS) [32] and Platt Scaling (PS) [37].

C. Quantitative Evaluation

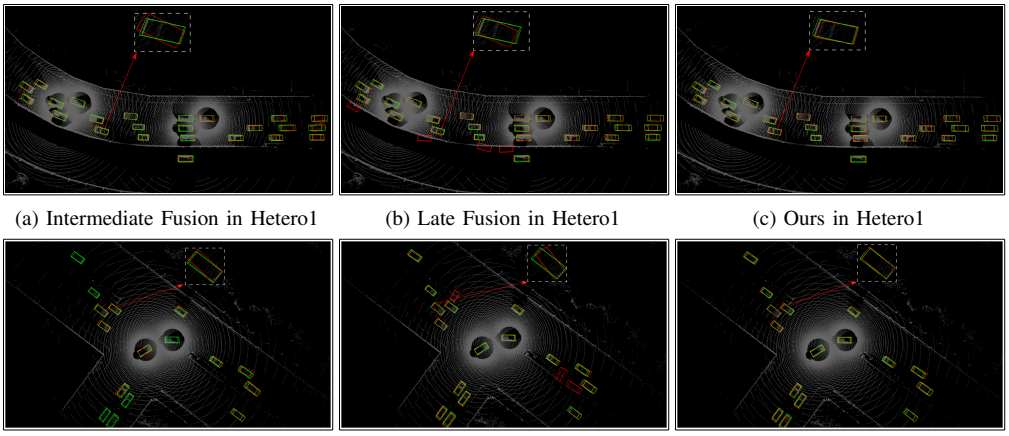
Main performance analysis. Table I describes the performance comparisons of different methods under *Homo*, *Hetero1*, and *Hetero2 Setting*. In the unrealistic *Homo* setting, all methods exceed the baseline remarkably while intermediate fusion and our method have very close performance (0.2% difference). However, when we consider the realistic model discrepancy factor, our method outperforms the classic late fusion and intermediate fusion significantly by 5.9%, 7.3% under *Hetero1 Setting*, and by 6.1%, 21.3% under *Hetero2 Setting*, respectively. The classic late fusion and intermediate fusion suffer from the model discrepancy, leading to clear accuracy decreases. In the *Hetero2 Setting*, the intermediate fusion even becomes lower than the baseline. On the contrary, our method only drops around 6% and 3% under the two realistic settings, indicating the effectiveness of the proposed calibration for the heterogeneity of the multi-agent perception system. Note that although the design essence of our framework aims to handle the heterogeneous situations, we also obtain performance boost under the *Homo Setting* compared with the standard late fusion that shares detection proposals. We attribute this gain to PSA and the filtering operation of low-confidence proposals after confidence calibration that removes some potential false positives.

Major component analysis. Here we investigate the contribution from each component by incrementally adding DBS and PSA. Table II reveals that both modules are beneficial for the performance boost, while the calibration exhibits more contributions – increasing the AP by 4.3% and 5.3% .



(a) Reliability Diagram before Calibration (b) Calibrated by Doubly Bounded Scaling (c) Performance of Different Calibrators

Fig. 5. The reliability diagrams in (a) and (b) reveal that Doubly Bounded Scaling method can effectively calibrate the classification confidence scores. In (c), the proposed Doubly Bounded Scaling outperforms Temperature Scaling and Platt Scaling under various experiment setups and aggregation algorithms.



(a) Intermediate Fusion in Hetero1 (b) Late Fusion in Hetero1 (c) Ours in Hetero1

(d) Intermediate Fusion in Hetero2 (e) Late Fusion in Hetero2 (f) Ours in Hetero2

Fig. 6. **Qualitative comparison in a busy freeway and a congested intersection.** Green and red 3D bounding boxes represent the groun truth and prediction respectively. Our method yields more accurate detection results.

Confidence calibration evaluation. Fig. 5a show the reliability diagram of Pointpillar used by the ego vehicle, in which a perfect calibration will produce a diagonal reliability curve, indicating the real accuracy matches the predictive confidence score. Reliability curves under or above the diagonal line represent over-confident or under-confident models, respectively. Pointpillar has much higher empirical accuracy than its reported confidence score. When using NMS to fuse the predictions of Pointpillar with that of another inaccurate but over-confident detector, the under-estimated confidence will result in the removal of Pointpillar's good predictions. After being calibrated by DBS, in Fig. 5b, the reliability curve of Pointpillar lies on the diagonal line.

Comparison with other calibration methods. Fig. 5c describes the comparison between our DBS calibration and other calibration methods, including TS and PS. Our DBS achieves better performance than others under both heterogeneous settings. Moreover, PSA can also improve the accuracy of different calibrators and experimental settings, showing the generalized capability to refine the prediction results.

D. Qualitative Results

Fig. 6 shows the detection results of intermediate fusion, classic late fusion, and our method under *Hetero1* and *Hetero2 Setting*. Our method can identify more objects while keeping very few false positives. The zoom-in examples show that our method can regress the bounding box positions more accurately, indicating the robustness against the model discrepancy in multi-agent perception systems.

V. Conclusions

In the context of cooperative perception, agents from different stakeholders have heterogeneous models. For the sake of confidentially, information related to the models and parameters should not be revealed to other agents. In this work, we present a model-agnostic collaboration framework that addresses two critical challenges of the vanilla late fusion strategy. First, we propose a confidence calibrator to align the classification confidence distributions of different agents. Second, we present a bounding box aggregation algorithm that takes into account both the calibrated classification confidence and the spatial congruence information given by bounding box regression. Experiments on a large-scale cooperative perception dataset shed light on the necessity of model calibration across heterogeneous agents. The results show that combining the two proposed techniques can improve the state-of-the-art for cooperative 3D object detection when different agents use distinct perception models.

References

[1] Yin Zhou and Oncel Tuzel. VoxelNet: End-to-end learning for point cloud based 3d object detection.

In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4490–4499, 2018.

[2] Yantao Lu, Xuetao Hao, Shiqi Sun, Weiheng Chai, Muchenxuan Tong, and Senem Velipasalar. Raanet: Range-aware attention network for lidar-based 3d object detection with auxiliary density level estimation, 2021.

[3] Zhaoxin Fan, Yazhi Zhu, Yulin He, Qi Sun, Hongyan Liu, and Jun He. Deep learning on monocular object pose detection and tracking: A comprehensive overview. *arXiv preprint arXiv:2105.14291*, 2021.

[4] Bowen Pan, Jiankai Sun, Ho Yin Tiga Leung, Alex Andonian, and Bolei Zhou. Cross-view semantic segmentation for sensing surroundings. *IEEE Robotics and Automation Letters*, 5(3):4867–4873, 2020.

[5] Peixi Xiong, Xuetao Hao, Yunming Shao, and Jerry Yu. Adaptive attention model for lidar instance segmentation. In *International Symposium on Visual Computing*, pages 141–155. Springer, 2019.

[6] Heming Du, Xin Yu, and Liang Zheng. Learning object relation graph and tentative policy for visual navigation. In *European Conference on Computer Vision*, pages 19–34. Springer, 2020.

[7] Anwesan Pal, Yiding Qiu, and Henrik Christensen. Learning hierarchical relationships for object-goal navigation. In *Conference on Robot Learning*, pages 517–528. PMLR, 2021.

[8] Qi Chen, Sihai Tang, Qing Yang, and Song Fu. Cooper: Cooperative perception for connected autonomous vehicles based on 3D point clouds. In *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, pages 514–524, 2019.

[9] Tsun-Hsuan Wang, Sivabalan Manivasagam, Ming Liang, Bin Yang, Wenyuan Zeng, and Raquel Urtasun. V2VNet: Vehicle-to-vehicle communication for joint perception and prediction. In *European Conference on Computer Vision (ECCV)*, pages 605–621. Springer, 2020.

[10] Qi Chen, Xu Ma, Sihai Tang, Jingda Guo, Qing Yang, and Song Fu. F-Cooper: Feature based cooperative perception for autonomous vehicle edge computing system using 3d point clouds. In *Proceedings of the 4th ACM/IEEE Symposium on Edge Computing*, page 88–100, 2019.

[11] Yiming Li, Shunli Ren, Pengxiang Wu, Siheng Chen, Chen Feng, and Wenjun Zhang. Learning distilled collaboration graph for multi-agent perception. *Advances in Neural Information Processing Systems (NeurIPS)*, 34, 2021.

[12] Runsheng Xu, Hao Xiang, Xin Xia, Xu Han, Jinlong Li, and Jiaqi Ma. OPV2V: An open benchmark dataset and fusion pipeline for perception with vehicle-to-vehicle communication. In *2022 IEEE International Conference on Robotics and Automation (ICRA)*, 2022.

[13] Yiming Li, Ziyan An, Zixun Wang, Yiqi Zhong, Siheng Chen, and Chen Feng. V2x-sim: A virtual collaborative perception dataset for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, 2021.

[14] Runsheng Xu, Hao Xiang, Zhengzhong Tu, Xin Xia, Ming-Hsuan Yang, and Jiaqi Ma. V2x-vit: Vehicle-to-everything cooperative perception with vision transformer. *arXiv preprint arXiv:2203.10638*, 2022.

[15] R. Song and A. Festag. Analysis of existing approaches for information sharing in cooperative intelligent transport systems – V2X messaging and SENSORIS. In *2021 38th FISITA World Congress*, 2021.

[16] Hao Xiang, Runsheng Xu, Xin Xia, Zhaoliang Zheng, Bolei Zhou, and Jiaqi Ma. V2xp-asg: Generating adversarial scenes for vehicle-to-everything perception. *arXiv preprint arXiv:2209.13679*, 2022.

[17] Runsheng Xu, Jinlong Li, Xiaoyu Dong, Hongkai Yu, and Jiaqi Ma. Bridging the domain gap for multi-agent perception. *arXiv preprint arXiv:2210.08451*, 2022.

[18] Alexander Neubeck and Luc Van Gool. Efficient non-maximum suppression. In *18th International Conference on Pattern Recognition (ICPR'06)*, volume 3, pages 850–855. IEEE, 2006.

[19] Zaydoun Yahya Rawashdeh and Zheng Wang. Collaborative automated driving: A machine learning-based method to enhance the accuracy of shared information. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pages 3961–3966. IEEE, 2018.

[20] Sanbao Su, Yiming Li, Sihong He, Songyang Han, Chen Feng, Caiwen Ding, and Fei Miao. Uncertainty quantification of collaborative detection for self-driving. *arXiv preprint arXiv:2209.08162*, 2022.

[21] Charles R Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J Guibas. Frustum pointnets for 3d object detection from rgb-d data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 918–927, 2018.

[22] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3d object proposal generation and detection from point cloud. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 770–779, 2019.

[23] Robert McCraith, Eldar Insafutdinov, Lukas Neumann, and Andrea Vedaldi. Lifting 2d object locations to 3d by discounting lidar outliers across objects and views. *arXiv preprint arXiv:2109.07945*, 2021.

[24] Zheng Li, Yue Zhao, Xiyang Hu, Nicola Botta, Cezar Ionescu, and George H Chen. Ecod: Unsupervised outlier detection using empirical cumulative distribution functions. *arXiv preprint arXiv:2201.00382*, 2022.

[25] Yue Zhao, Ryan Rossi, and Leman Akoglu. Automatic unsupervised outlier model selection. *Advances in Neural Information Processing Systems*, 34, 2021.

[26] Weizhe Chen and Lantao Liu. Informative planning in the presence of outliers. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2022.

[27] Alex H. Lang, Sourabh Vora, Holger Caesar, Lubing

Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12689–12697, 2019.

[28] Yan Yan, Yuxing Mao, and Bo Li. Second: Sparsely embedded convolutional detection. *Sensors (Basel, Switzerland)*, 18, 2018.

[29] Shaoshuai Shi, Chaoxu Guo, Li Jiang, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. Pv-rcnn: Point-voxel feature set abstraction for 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10529–10538, 2020.

[30] Shaoshuai Shi, Li Jiang, Jiajun Deng, Zhe Wang, Chaoxu Guo, Jianping Shi, Xiaogang Wang, and Hongsheng Li. Pv-rcnn++: Point-voxel feature set abstraction with local vector representation for 3d object detection. *arXiv preprint arXiv:2102.00463*, 2021.

[31] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017.

[32] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International Conference on Machine Learning*, pages 1321–1330. PMLR, 2017.

[33] Wesley J Maddox, Pavel Izmailov, Timur Garipov, Dmitry P Vetrov, and Andrew Gordon Wilson. A simple baseline for bayesian uncertainty in deep learning. *Advances in Neural Information Processing Systems*, 32, 2019.

[34] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059. PMLR, 2016.

[35] Sunil Thulasidasan, Gopinath Chennupati, Jeff A Bilmes, Tanmoy Bhattacharya, and Sarah Michalak. On mixup training: Improved calibration and predictive uncertainty for deep neural networks. *Advances in Neural Information Processing Systems*, 32, 2019.

[36] Bianca Zadrozny and Charles Elkan. Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers. In *Icml*, volume 1, pages 609–616. Citeseer, 2001.

[37] John Platt et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3): 61–74, 1999.

[38] Bianca Zadrozny and Charles Elkan. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 694–699, 2002.

[39] Ananya Kumar, Percy S Liang, and Tengyu Ma. Verified uncertainty calibration. *Advances in Neural Information Processing Systems*, 32, 2019.

[40] Jan Hosang, Rodrigo Benenson, and Bernt Schiele. Learning non-maximum suppression. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4507–4515, 2017.

[41] Navaneeth Bodla, Bharat Singh, Rama Chellappa, and Larry S Davis. Soft-nms–improving object detection with one line of code. In *Proceedings of the IEEE international conference on computer vision*, pages 5561–5569, 2017.

[42] Songtao Liu, Di Huang, and Yunhong Wang. Adaptive nms: Refining pedestrian detection in a crowd. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6459–6468, 2019.

[43] Rasmus Rothe, Matthieu Guillaumin, and Luc Van Gool. Non-maximum suppression for object detection by passing messages between windows. In *Asian conference on computer vision*, pages 290–306. Springer, 2014.

[44] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuScenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11621–11631, 2020.

[45] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2446–2454, 2020.

[46] Meelis Kull, Telmo Silva Filho, and Peter Flach. Beta calibration: a well-founded and easily implemented improvement on logistic calibration for binary classifiers. In *Artificial Intelligence and Statistics*, pages 623–631. PMLR, 2017.

[47] Ponnambalam Kumaraswamy. A generalized probability density function for double-bounded random processes. *Journal of hydrology*, 46(1-2):79–88, 1980.

[48] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. Carla: An open urban driving simulator. In *Conference on robot learning*, pages 1–16. PMLR, 2017.

[49] Runsheng Xu, Yi Guo, Xu Han, Xin Xia, Hao Xiang, and Jiaqi Ma. Opencda: an open cooperative driving automation framework integrated with co-simulation. In *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, pages 1155–1162. IEEE, 2021.

[50] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12697–12705, 2019.