Collaboration Helps Camera Overtake LiDAR in 3D Detection

Yue Hu[1]   Yifan Lu[1]   Runsheng Xu[2]   Weidi Xie[1,3]   Siheng Chen[1,3*]   Yanfeng Wang[3,1]

[1]Cooperative Medianet Innovation Center, Shanghai Jiao Tong University
[2]University of California, Los Angeles   [3]Shanghai AI Laboratory

[1]{18671129361, yifan_lu, weidi, sihengc, wangyanfeng}@sjtu.edu.cn   [2]{rxx3386}@ucla.edu

Abstract

Camera-only 3D detection provides an economical solution with a simple configuration for localizing objects in 3D space compared to LiDAR-based detection systems. However, a major challenge lies in precise depth estimation due to the lack of direct 3D measurements in the input. Many previous methods attempt to improve depth estimation through network designs, e.g., deformable layers and larger receptive fields. This work proposes an orthogonal direction, improving the camera-only 3D detection by introducing multi-agent collaborations. Our proposed collaborative camera-only 3D detection (CoCa3D) enables agents to share complementary information with each other through communication. Meanwhile, we optimize communication efficiency by selecting the most informative cues. The shared messages from multiple viewpoints disambiguate the single-agent estimated depth and complement the occluded and long-range regions in the single-agent view. We evaluate CoCa3D in one real-world dataset and two new simulation datasets. Results show that CoCa3D improves previous SOTA performances by 44.21% on DAIR-V2X, 30.60% on OPV2V+, 12.59% on CoPerception-UAVs+ for AP@70. Our preliminary results show a potential that with sufficient collaboration, the camera might overtake LiDAR in some practical scenarios. We released the dataset *and* code.

1. Introduction

As a fundamental task of computer vision, 3D object detection aims to localize objects in the 3D physical space given an agent's real-time sensor inputs. It is crucial in a wide range of applications, including autonomous driving [10,33,43], surveillance systems [39], robotics [12] and unmanned aerial vehicles [7]. Depending on sensor setups, there are multiple technical solutions to realize 3D object detection. In this spectrum, one extreme emphasizes raising the upper bound of detection performance, which uses
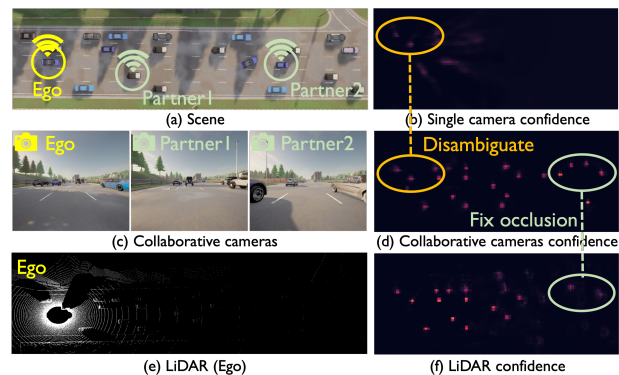

Figure 1. Collaborative camera-only 3D detection can disambiguate the single-view estimated depth, address the long-range and occlusion issues, and approach LiDAR in 3D detection.

high-end LiDAR sensor [2, 3, 10, 21, 28, 42, 45] to collect precise 3D measurements. However, this approach is too expensive to scale up. The other extreme solution emphasizes cost effectiveness, which tries to use thrifty sensor setups, e.g. only using cameras to detect 3D objects in real-time [1, 8, 9, 23, 25, 26, 30, 32, 41, 43]. However, camera-only 3D detection is significantly and consistently worse than LiDAR-based detection in most scenarios [19].

In this paper, we propose an orthogonal direction for improving camera-only 3D detection performances by introducing multi-agent collaborations. Hypothetically, empowered by advanced communication systems, multiple agents equipped only with cameras could share visual information with each other. This would bring three outstanding benefits. First, different viewpoints from multiple agents can largely resolve the depth ambiguity issue in camera-only 3D detection, bridging the gap with expensive LiDARs on depth estimation. Second, multi-agent collaboration avoids inevitable limitations in single-agent 3D detection, such as occlusion and long-range issues, and potentially enables more holistic 3D detection; that is, detecting all the objects existed in the 3D scene, including those beyond visual range. Since LiDAR also suffers from limited field of view, this potentially enables collaborative cameras to outperform LiDAR. Third, the total expense of a large fleet of vehicles is significantly reduced as cameras are much

---

*Corresponding author.

cheaper than LiDAR. However, multi-agent collaboration also brings new challenges. Different from many multi-view geometry problems, here we also have to concern communication bandwidth constraints. Thus, each agent needs to select the most informative cues to share.

Following this design rationale, we propose a novel collaborative camera-only 3D detection framework `CoCa3D`. It includes three parts: i) single-agent camera-only 3D detection, which achieves basic depth estimation and 3D detection for each agent; ii) collaborative depth estimation, which disambiguates the estimated depths by promoting spatial consistency across multiple agents' viewpoints; and iii) collaborative detection feature learning, which complements detection features by sharing key detection messages with each other. Compared to recent collaborative perception methods [11,14] that are dealing with LiDAR, `CoCa3D` specifically designs novel collaborative depth estimation to customize the task of camera-only 3D detection.

To evaluate `CoCa3D`, we conduct comprehensive experiments on one real-world dataset, DAIR-V2X [40], and two new simulation datasets, OPV2V+ and CoPerception-UAVs+, which are extended based on original OPV2V [37] and CoPerception-UAVs [6] with more collaborative agents that cover three types of agents (cars, infrastructures and drones). Our results show that i) with 10 collaborative agents, `CoCa3D` enables camera-only detectors to overtake LiDAR-based detectors on OPV2V+; and ii) `CoCa3D` consistently outperforms previous works in the performance-bandwidth trade-off across multiple datasets by a large margin, improving the previous SOTA performances by 30.60% on OPV2V+, 12.59% on CoPerception-UAVs+, 44.21% on DAIR-V2X for AP@70. To sum up, our contributions are:

• We propose a novel collaborative camera-only 3D detection framework `CoCa3D`, which improves the detection ability of cameras with multi-agent collaboration, promoting more holistic 3D detection.

• We propose core communication-efficient collaboration techniques, which explore the spatially sparse yet critical depth messages and tackle the depth ambiguity, occlusion, and long-range issues by fusing complementary information from different viewpoints, achieving more accurate and complete 3D representation.

• We expand two previous collaborative datasets with more agents, and conduct extensive experiments, validating that i) `CoCa3D` significantly bridges the performance gap between camera and LiDAR on OPV2V+ and DAIR-V2X; and ii) `CoCa3D` achieves the state-of-the-art performance-bandwidth trade-off.

2. Related Work

Camera-only 3D object detection. Camera-only 3D detection aims to detect objects in the 3D space given the 2D image through explicit or implicit depth estimation [7, 9, 23, 26, 30, 41]. Recently, bird's-eye-view (BEV) repre-

sentations are widely used for their computation efficiency and comparable performance compared to the 3D voxel features [10]. To get the BEV features, there are two types of methods. The depth-based methods [8, 23, 25] first estimate the depth distribution, and then attentively project the 2D image features along the projection ray to get the 3D voxel features and collapse it to BEV feature. The query-based methods [16, 22, 43] first initialize queries for each BEV grid and then leverage transformer-architecture-based cross-attention to query image features with camera-aware positional embedding. Here, our single-agent camera-only detector follows the simple-yet-effective CaDDN [25].

LiDAR 3D object detection. LiDAR-based 3D detection achieves excellent performance due to the precise 3D measurements of the input data. Two well-known ways to encode LiDAR points include voxel-based [10, 42, 45] and point-based [28,29,38]. Voxel-based methods divide the 3D space into regular voxels [45] or pillars [10], and encode the point inside into feature representations. Point-based methods are usually based on the PointNet [24] series to aggregate the feature of points. Then the point features will be used for the proposal refinement. LiDAR-based 3D detection performs well, but high-quality LiDAR is hard to be adopted in a large scale due to the expensive cost. Here we propose an economic camera-only solution by introducing multi-agent collaboration, whose performance can catch up with LiDAR given a sufficient number of agents.

Collaborative perception. Collaborative perception is an emerging application of multi-agent systems, which promotes perception performance by enabling agents to share information with other agents through communication. In this emerging research field, a surge of high-quality datasets, (V2X-SIM [13], OPV2V [37], DAIR-V2X [40], CoPerception-UAVs [6]), and collaborative methods [5,6,13–15,17,18,20,31,35–37,40] aimed for better performance-bandwidth trade-off have been proposed. Collaborative methods have three types: early, late and intermediate fusion. Compared with early fusion(i.e. transmitting raw data) and late fusion (i.e. transmitting outputs), intermediate fusion(i.e. transmitting and fusing features encoded by deep networks) achieves a better performance-bandwidth trade-off. DiscoNet [14] adopts knowledge distillation to take the advantage of both early and intermediate collaboration. Where2comm [6] optimizes communication efficiency by sharing spatially sparse yet perceptually critical information. However, previous works mainly focus on LiDAR-based 3D object detection, while we investigate collaborative camera-only detection and specifically enhance depth estimation via multi-agent collaboration.

3. Collaborative Camera-Only 3D Detection

This section presents `CoCa3D`, collaborative camera-only 3D detection framework, which enables multiple
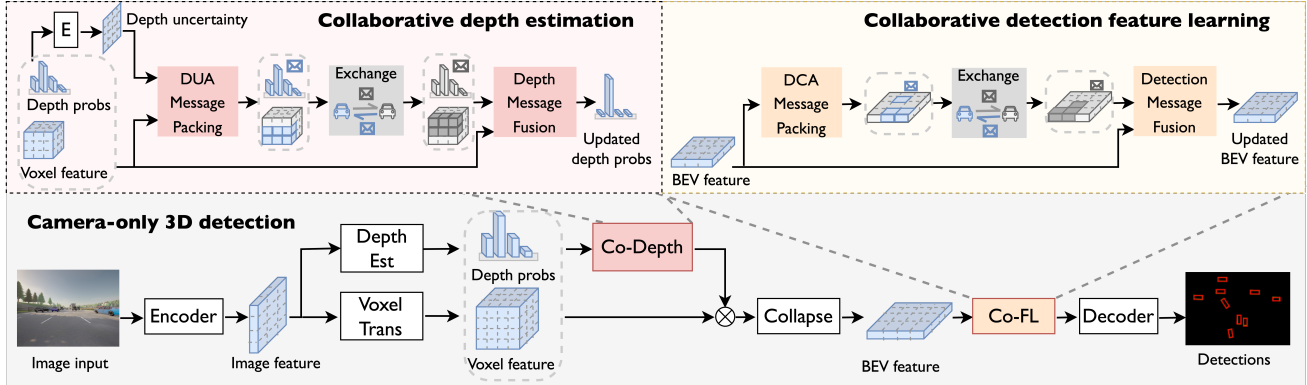
Figure 2. System overview. `CoCa3D` is a camera-only 3D detector integrated with two collaboration modules. Collaborative depth estimation (Co-Depth) enhances the single-agent estimated depth to achieve more accurate 3D feature. Collaborative detection feature learning (Co-FL) complements the single-agent 3D feature to achieve more holistic 3D detection.

agents to share visual information with each other, promoting more holistic 3D detection.

3.1. Problem Formulation

Consider N agents in the scene, let $\mathcal{X}_i$ be the RGB image collected by the ith agent and $\mathcal{O}_i^0$ be the corresponding ground-truth detection. The objective of `CoCa3D` is to maximize detection performances of all agents given certain communication budget B; that is,

$$\max_{\theta, \mathcal{P}} \sum_i g\left(\Phi_\theta\left(\mathcal{X}_i, \{\mathcal{P}_{i \to j}\}_{j=1}^N\right), \mathcal{O}_i^0\right), \text{s.t.} \sum_i |\mathcal{P}_{i \to j}| \le B,$$

where $g(\cdot, \cdot)$ is the detection evaluation metric, $\Phi(\cdot)$ is a detection model with trainable parameter θ, and $\mathcal{P}_{i \to j}$ is the message transmitted from the ith agent to the jth agent. The key challenge is to determine the messages $\mathcal{P}_{i \to j}$, which should be both informative and compact. Our design rational comes from two aspects: First, since the major gap between camera and LiDAR is the depth, the message should include depth information. This would allow different viewpoints from multiple agents to disambiguate the infinite depth possibilities and localize the correct depth candidate. Second, the message should include detection clues to provide complementary detection information, which can fundamentally overcome inevitable limitations of single-agent detection, such as occlusion and long-range issues.

Based on the above intuition, the proposed `CoCa3D` includes a) single-agent camera-only 3D detection, which achieves basic depth estimation and detection ability, and b) multi-agent collaboration, which shares both estimated depth information and detection features to improve 3D representation and detection performances; see Fig. 2.

3.2. Single-agent camera-only 3D detection

Single-agent camera-only 3D object detection network learns to detect 3D objects in the physical space based on the 2D camera inputs. Here we adopt CaDNN [25], an off-the-shelf architecture. The main idea is to project the flat image feature into all the possible depths in the 3D space,

and then attentively aggregate the 3D voxel feature and collapse them into the BEV feature. To mention that, the BEV feature is used because it largely reduces the computation cost, while performing similarly to 3D voxel features [10]. The architecture consists of five modules as detailed below.

Encoder. The encoder extracts features from the input images. For agent i, given its input $\mathcal{X}_i$, the output feature map of encoder is $\mathcal{F}_i = \Phi_{\text{enc}}(\mathcal{X}_i) \in \mathbb{R}^{H \times W \times C}$, where H, W, C are the height, width and channel of the image feature.

Depth estimation. The depth estimation module predicts pixel-wise categorical depth distribution over a set of pre-defined depth bins to accurately locate image information in 3D space. Treating the estimation as a classification problem can capture the inherent depth estimation uncertainty to reduce the impact of erroneous depth estimates, as stated in CaDNN [25]. To achieve this, a parametric depth estimation network $\Phi_{\text{Depth}}(\cdot)$ is used, and the pixel-wise categorical depth distribution is obtained as

$$\mathcal{D}_i = \Phi_{\text{Depth}}(\mathcal{F}_i) \in \mathbb{R}^{H \times W \times D}, \quad (1)$$

whose (h, w)th element $\mathcal{D}_i(h, w) \in \mathbb{R}^D$ refers to the estimated depth distribution of the (h, w)th image pixel with D as the number of discretized depth bins.

Voxel transformation. Voxel transformation projects the image feature to the 3D space, given all the possible depth candidates and the known image calibration matrix, resulting in a 3D voxel representation $\widehat{\mathcal{V}}_i \in \mathbb{R}^{X \times Y \times Z \times C}$. The camera projection matrix $\mathbf{P} \in \mathbb{R}^{3 \times 4}$ defines the mapping between the 3D voxel coordinate to the image pixel coordinate, this is, $[u, v, d] = \mathbf{P} \cdot [x, y, z, 1]$. Similarly, the depth probabilities can be transformed to the 3D voxel space, resulting in $\widehat{\mathcal{D}}_i \in \mathbb{R}^{X \times Y \times Z}$, where each element indicates the confidence that the feature pixel belongs to the voxel.

Collapse. The voxel feature is collapsed to a single height plane to generate the bird's-eye-view (BEV) feature as $\overline{\mathcal{B}}_i = \Phi_{\text{clp}}(\widehat{\mathcal{D}}_i \odot \widehat{\mathcal{V}}_i) \in \mathbb{R}^{X \times Y \times C}$, where $\odot$ is the element-wise multiplication. Symbol $\cdot / \widehat{\cdot} / \overline{\cdot}$ reflect 2D image / 3D voxel / BEV spaces, respectively. The collapse network $\Phi_{\text{clp}}(\cdot)$

flattens the 3D voxel feature along the Z-axis and then applies 1×1 convolution to reduce the channel dimension.

Decoder. The decoder takes the BEV feature as input, and outputs the objects including class and regression. Here, we adopt CenterNet [44], an off-the-shelf detector. Given the BEV feature $\overline{\mathcal{B}}_i$, the detection decoder $\Phi_{\text{dec}}(\cdot)$ generate the dense heatmap of ith agent by $\overline{\mathcal{O}}_i = \Phi_{\text{dec}}(\overline{\mathcal{B}}_i) \in \mathbb{R}^{X \times Y \times 7}$, where each location of $\overline{\mathcal{O}}_i$ represents a rotated box with class $(c, x, y, h, w, \cos \alpha, \sin \alpha)$, denoting class confidence, position residual, size and angle. Non-maximum suppression (NMS) is applied to the dense predictions and generate the final sparse output of the 3D detection system.

Discussion. Admittedly, single-agent camera-only 3D detection makes localizing objects in the 3D space given 2D image input possible through numerous ingenious algorithm designs. The 3D detection performance is still far from perfect due to the ambiguous single-agent depth estimation and limited visual range. Tackling these two issues are challenging for a single agent, but from a perspective of multi-agent collaboration, sharing complementary visual clues makes depth estimation and visual range enlargement much more natural. This motivates our following designs.

3.3. Multi-agent collaboration

Multi-agent collaboration includes two parts: i) collaborative depth estimation, which enables the sharing of depth information estimated by individual agents to produce more accurate 3D representation and less aliased BEV feature; and ii) collaborative detection feature learning, which enables the sharing of detection features of individual agents to achieve more holistic 3D detection.

3.3.1 Collaborative depth estimation

Collaborative depth estimation (**Co-Depth**) targets to disambiguate infinite depth possibilities in single-agent camera-only depth estimation and localize the correct depth candidate through multi-view consistency. The intuition is, for a correct depth candidate, its corresponding 3D location should be spatially consistent from multiple agents' viewpoints. To achieve this, each agent can exchange the depth information through communication. Meanwhile, we promote communication efficiency by selecting the most critical and unambiguous depth information. Accordingly, Co-Depth includes: a) depth uncertainty aware message packing, which packs compact messages with unambiguous depth information; and b) depth message fusion, which enhances depth estimation with the received depth messages.

Depth uncertainty aware message packing. Depth uncertainty-aware (DUA) message packing packs the most critical depth information used for multi-view consistency into the to-be-sent message based on the depth uncertainty. The depth message includes: i) the voxel feature, which is used for the multi-view visual similarity measurement; and ii) the depth probability, which indicates the confidence that

the feature pixel belongs to the voxel and is used for the multi-view candidate selection.

The spatially sparse features of ith agent are selected based on its spatial depth uncertainty map in the 2D image space $\mathbf{U}_i \in \mathbb{R}^{H \times W}$, that can be represented by the pixel-wise entropy from corresponding depth distribution,

$$\mathbf{U}_i(h, w) = \mathrm{H}(\mathcal{D}_i(h, w)) \in \mathbb{R},$$

where $\mathrm{H}(\cdot)$ is the entropy function and $\mathcal{D}_i(h, w)$ is the depth distribution at the (h, w)th image pixel, following from (1). The intuition is, for the locations with low uncertainty score, an agent is confident about which depth bin the feature pixel belongs to. Thus, sharing information at these locations can help with depth estimation accuracy. On the contrary, for the locations with high uncertainty score, an agent is hard to tell which depth bin the feature pixel belongs to. Transmitting these spatial features would introduce ambiguity, and even degrade the single-agent depth estimation.

Specifically, a binary selection matrix is used to represent whether each location is selected or not, where 1 denotes selected, and 0 elsewhere. Let $\mathbf{M}_{i \to j}^d \in \{0, 1\}^{H \times W}$ be a selection matrix to determine the depth message sent from agent i to j. Its (h, w)th element is

$$\mathbf{M}_{i \to j}^d(h, w) = I(\mathbf{U}_i(h, w) < u_{\text{thre}}),$$

where $I(\cdot)$ is an indicator function and u_{thre} is a predefined threshold. This reflects that a specific spatial area will be selected when its depth uncertainty is below a threshold.

The selection matrix can be projected in the 3D voxel space, resulting in $\widehat{\mathbf{M}}_{i \to j}^d \in \mathbb{R}^{X \times Y \times Z}$. Then, the selected voxel feature map $\widehat{\mathcal{Z}}_{i \to j}$ and the selected depth distribution $\widehat{\mathcal{E}}_{i \to j}$ sent from the ith agent to the jth agent are obtained as

$$\widehat{\mathcal{Z}}_{i \to j} = \widehat{\mathbf{M}}_{i \to j}^d \odot \widehat{\mathcal{V}}_i \in \mathbb{R}^{X \times Y \times Z \times C},$$

$$\widehat{\mathcal{E}}_{i \to j} = \widehat{\mathbf{M}}_{i \to j}^d \odot \widehat{\mathcal{D}}_i \in \mathbb{R}^{X \times Y \times Z},$$

where $\widehat{\mathcal{V}}_i$ and $\widehat{\mathcal{D}}_i$ are the full 3D voxel feature map and depth distribution. Overall, the depth message sent from the ith agent to the jth agent is $\mathcal{P}_{i \to j}^d = (\widehat{\mathcal{E}}_{i \to j}, \widehat{\mathcal{Z}}_{i \to j})$, where both entries are spatially sparse and communication efficient.

Depth message fusion. Depth message fusion targets to enhance the depth estimation given the received depth messages through different viewpoints of multiple agents. The intuition is that for a correct depth candidate, the visual features at the same 3D point observed by multiple agents should be similar. To achieve this, we introduce multi-view depth consistency weighting. Let $\widehat{\mathcal{S}}_i(x, y, z) \in \mathbb{R}$ be the matching score between agent i and its neighbors $\mathcal{N}_i$, those who share messages with current agent, at coordinate (x, y, z). The multi-view matching score is obtained as

$$\widehat{\mathcal{W}}_{j \to i}(x, y, z) = I(\widehat{\mathcal{E}}_{j \to i}(x, y, z) > p_{\text{thre}}) \in \{0, 1\},$$

$$\widehat{\mathcal{S}}_i(x, y, z) = \sum_{j \in \mathcal{N}_i} \widehat{\mathcal{W}}_{j \to i}(x, y, z) \left\langle \widehat{\mathcal{V}}_i(x, y, z), \widehat{\mathcal{Z}}_{j \to i}(x, y, z) \right\rangle,$$

where $\widehat{\mathcal{W}}_{j \to i}(x, y, z)$ is a binary weight used to filter out the

neighboring view that has a low depth confidence, below a threshold p_{thre} and $\langle \cdot, \cdot \rangle$ is the inner product that measures the visual feature consistency across two views. This means that either the depth candidate is wrong or it is not visible in that view (e.g. due to occlusion), where the multi-view matching score should not be considered.

Given the multi-view matching score, the collaborative depth distribution is obtained as

$$\widehat{\mathcal{D}}_i' = \Phi_{\text{CoDepth}}\left(\left[\widehat{\mathcal{D}}_i; \widehat{\mathcal{S}}_i\right]\right) \in \mathbb{R}^{X \times Y \times Z},$$

where the $[;]$ denotes concatenation, $\Phi_{\text{CoDepth}}(\cdot)$ is the collaborative depth estimation network implemented with 1×1 convolutions followed by a sigmoid activation. Each element of the collaborative depth distribution $\widehat{\mathcal{D}}_i'$ reflects the confidence of the feature locates in the corresponding voxel. Note that i) the collaboratively updated depth probability is decided on both the single-view confidence and the multi-view consistency; ii) the depth consistency weighting discards the candidates with low single-view depth probability to avoid ambiguity and make the estimation more robust.

Given this collaborative depth distribution, each agent can weigh the 3D voxel feature and then collapse into a BEV feature with a collapse network $\Phi_{\text{clp}}(\cdot)$, this is, $\overline{B}_i = \Phi_{\text{clp}}(\widehat{\mathcal{D}}_i' \odot \widehat{\mathcal{V}}_i) \in \mathbb{R}^{X \times Y \times C}$. The collapse network is implemented with 1×1 convolution to reduce channel dimension.

3.3.2 Collaborative detection feature learning

Collaborative depth estimation carefully refines the depth and promotes more accurate 3D representations for each single agent. However, the physical limitations of single agent, such as, limited field of view, occlusion, and long-range issues still remain. To achieve more holistic 3D detection, each agent should be able to exchange 3D detection feature and make use of the complementary information. Meanwhile, we promote communication efficiency by selecting the most perceptually critical information. Therefore, collaborative detection feature learning (**Co-FL**) includes: a) detection confidence aware message packing, which packs spatially sparse yet perceptually critical 3D features with the guidance of the detection confidence; and ii) detection message fusion, which enhances the 3D feature with the received detection messages.

Detection confidence aware message packing. The detection confidence aware (DCA) message packing targets to pack the complementary perceptual information into a compact message. The core idea is to explore the spatial heterogeneity of perceptual information. The intuition is that the areas that contain objects are more critical than background areas. During collaboration, areas with objects could help recover the miss-detected objects due to the limited view; and background areas could be omitted to save the precious bandwidth. To achieve this, we implement the spatial confidence map with the detection confidence map by

$$\overline{\mathbf{C}}_i = \Phi_{\text{cls}}(\overline{\mathcal{B}}_i) \in \mathbb{R}^{X \times Y},$$

where $\Phi_{\text{cls}}(\cdot)$ denotes the standard classification network in the single-agent decoder $\Phi_{\text{det}}(\cdot)$, $\overline{\mathcal{B}}_i$ is the BEV feature generated with the collaboratively estimated depth. Based on this map, agents decide which spatial area to communicate; that is, each agent offers spatially sparse, yet critical features to support other agents.

Specifically, a binary selection matrix is used to represent each location is selected or not. Let $\overline{\mathbf{M}}_{i \to j}^f \in \{0,1\}^{X \times Y}$ be a selection matrix to determine the detection message sent from agent i to j. Its (x, y)th element is

$$\overline{\mathbf{M}}_{i \to j}^f(x, y) = I(\overline{\mathbf{C}}_i(x, y) > c_{\text{thre}}),$$

where $I(\cdot)$ is an indicator function and c_{thre} is a predefined threshold. This reflects that a specific spatial area will be selected when its detection confidence is above a threshold. Overall, the detection message sent from the ith agent to the jth agent is $\mathcal{P}_{i \to j}^f = \overline{\mathbf{M}}_{i \to j}^f \odot \overline{\mathcal{B}}_i$. Only selected features and their indices are packed, and they can be recovered at the receiver. This greatly reduces communication cost.

Detection message fusion. Here we augment the detection feature of each agent by aggregating the received detection messages from other agents. We implement this with simple but effective non-parametric point-wise maximum fusion. Specifically, for the ith agent, after receiving the jth agent's message $\mathcal{P}_{j \to i}^f$. We also include the ego feature map in fusion and denote $\mathcal{P}_{i \to i}^f = \overline{\mathcal{B}}_i$ to make the formulation simple and consistent, where $\mathcal{P}_{i \to i}^f$ might not be sparse. The fused BEV feature is obtained as

$$\overline{\mathcal{B}}_i' = \max_{j \in \mathcal{N}_i \cup \{i\}} (\mathcal{P}_{j \to i}^f) \in \mathbb{R}^{X \times Y \times C},$$

where $\max(\cdot)$ maximizes the corresponding features from multiple agents at each individual spatial location. Note that attention fusion is not permutation invariant, as attention weights vary with the ordering of key and query. Here we simply use the max operator to avoid this permutation variant issue. The fused BEV feature is output to the decoder to generate the final detection.

3.4. Losses

To train the overall system, we supervise two tasks: categorical depth estimation and 3D object detection. The overall loss is $L = L_{\text{dep}}\left(\mathcal{D}_i, \mathcal{D}_i^0\right) + L_{\text{det}}\left(\overline{\mathcal{O}}_i, \mathcal{O}_i^0\right)$, where $\mathcal{D}_i^0$ and $\mathcal{O}_i^0$ is the ith agent's ground-truth depth category and objects, respectively, L_{dep} and L_{det} is the depth classification loss [25] and detection loss [44], respectively.

4. Experimental Results

4.1. Datasets and experimental settings

We cover three datasets, both real-world and simulation scenarios, and multiple agent types. Metric Average Precision (AP) at Intersection-over-Union (IoU) threshold of $0.30, 0.50, 0.70$, and 0.80 are used. Metric multi-class classification accuracy is used for depth accuracy. The communication volume follows the standard setting as [6, 36, 37]
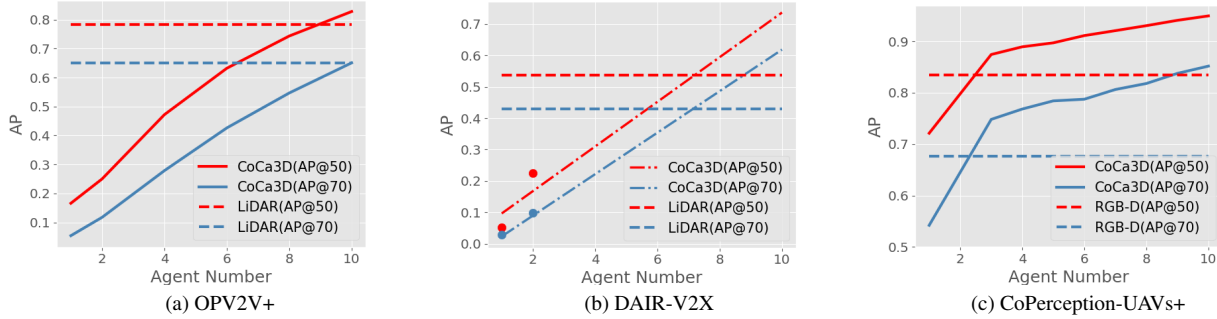
Figure 3. CoCa3D steadily improves 3D detection performance as the number of agents grows. In OPV2V+, around 10 collaborative agents enable collaborative cameras to catch up with LiDAR.

Table 1. CoCa3D significantly outperforms the previous SOTAs, e.g. improve by 30.60% on OPV2V, 12.59% on CoPerception-UAVs, 44.21% on DAIR-V2X for metric AP@70.

Method	OPV2V+			CoPerception-UAVs+			DAIR-V2X		
	AP@30	AP@50	AP@70	AP@50	AP@70	AP@80	AP@30	AP@50	AP@70
No Collaboration	0.2748	0.2041	0.0853	0.6956	0.4900	0.2309	0.0977	0.0524	0.0305
Late Fusion	0.6501	0.6198	0.5109	0.7206	0.5372	0.2597	0.2060	0.1078	0.0455
When2com (CVPR'20)	0.4853	0.4211	0.3737	0.8219	0.6705	0.4102	0.1957	0.0984	0.0459
V2VNet (ECCV'20)	0.6246	0.5042	0.3852	0.9093	0.7177	0.3804	0.1640	0.0847	0.0512
DiscoNet (NeurIPS'21)	0.7300	0.6009	0.4179	0.9054	0.7079	0.3564	0.1836	0.1262	0.0683
V2X-ViT (ECCV'22)	0.8346	0.6659	0.3946	0.9094	0.7143	0.3525	0.1862	0.1075	0.0490
Where2comm (NeurIPS'22)	0.8191	0.7089	0.4741	0.9102	0.7383	0.3676	0.1754	0.1025	0.0547
CoCa3D	**0.8642**	**0.8260**	**0.6675**	**0.9497**	**0.8502**	**0.5835**	**0.3522**	**0.2260**	**0.0985**

that counts the message size by byte in log scale with base 2. To compare communication results straightforwardly and fair, we do not consider any extra compression.

OPV2V+. The original OPV2V [37] is a large-scale vehicle-to-vehicle collaborative perception dataset, co-simulated by OpenCDA [34] and CARLA [4]. Here we introduce an extended version, OPV2V+, which includes more collaborative agents ($\approx$ 10). Each agent is equipt with 4 cameras and 4 depth sensors with resolution 600×800. The detection range is $281.6m \times 80m$. Our single-agent camera-only detector follows CaDDN [25] with 50 depth categories in linear-increasing spacing mode.

DAIR-V2X. DAIR-V2X [40] is the only public **real-world** collaborative perception dataset. It contains two agents: a vehicle and a road-side-unit with image resolution 1080×1920. The perception range is $100m \times 79m$. Our single-agent camera-only detector also follows CaDDN [25].

CoPerception-UAVs+. The original CoPerception-UAVs [6] is a large-scale UAV-based collaborative perception dataset, co-simulated by AirSim [27] and CARLA [4]. Here we introduce an extended version, CoPerception-UAVs+, which includes more collaborative agents ($\approx$ 10). Each agent equips with 1 camera and 1 depth sensor with resolution 450×800. The detection range is $192m \times 96m$. Our single-agent camera-only detector follows DVDET [?] with 10 depth categories in uniform spacing mode.

4.2. Quantitative evaluation

Collaborative camera-only 3D detection overtakes LiDAR. Fig. 3 investigates how the collaborative camera-only 3D detection performance changes with the number of collaborative agents on multiple datasets. To set up a goal, we consider a LiDAR-based detector implemented by the widely used PointPillar [10]. Fig. 3 (a) shows that: i) with 10 collaborative agents, CoCa3D outperforms the LiDAR 3D detection at both AP@0.5/0.7 on OPV2V+! ii) the detection performance positively increases as the number of collaborative agents. And we then use the slope of OPV2V+ to fit the function of collaborative detection performance with agent number on DAIR-V2X as it only has two agents available. Fig. 3 (b) shows that camera-only 3D detection is expected to outperform LiDAR 3D detection with 7 collaborative vehicles in the real scenario. As LiDAR is not a commonly available equipment for drones, we set a high-quality RGB-D camera instead. Fig. 3 (c) shows that with 3 collaborative drones, camera-only 3D detection outperforms ground-truth depth-based 3D detection. Furthermore, the steadily increasing detection performance with the number of collaborative agents encourages the agents to actively collaborate and achieve consistent improvement.

Benchmark comparison. Tab. 1 compares the proposed CoCa3D with previous collaborative methods. We consider single-agent detection without collaboration (No collaboration), When2com [17], V2VNet [31], DiscoNet [14], V2X-ViT [36], Where2comm [6] and late fusion, where agents

Table 2. `CoCa3D` outperforms single-agent camera-only 3D detection with GT depth on OPV2V+ and CoPerception-UAVs+.

Co -Depth	Co -FL	OPV2V+			CoPerception-UAVs+		
		AP@30	AP@50	AP@70	AP@50	AP@70	AP@80
-	-	0.2748	0.2041	0.0853	0.7213	0.5421	0.2846
GT	-	0.3454	0.2553	0.0973	0.8347	0.6764	0.4120
-	✓	0.8201	0.7191	0.4756	0.9084	0.7256	0.4028
GT	✓	**0.9120**	**0.8805**	**0.7434**	**0.9505**	0.8398	0.5504
✓	✓	0.8642	0.8260	0.6675	0.9495	**0.8518**	**0.5849**

directly exchange the detected 3D boxes. We see that the proposed `CoCa3D` significantly outperforms previous state-of-the-arts, improves the SOTA performance by 30.60% on OPV2V+, 12.59% on CoPerception-UAVs+, 44.21% on DAIR-V2X for AP@70. The reason is that collaborative depth estimation promotes more accurate 3D features per single agent, and the enhanced single agent features further facilitate collaborative detection feature learning, while previous collaborative methods do not specifically consider depth ambiguity and collaborate over single-agent features.

Multi-agent collaboration evaluation. Tab. 2 assess the effectiveness of the proposed multi-agent collaboration on OPV2V+ and CoPerception-UAVs+ datasets. Ground-truth (GT) depth is included to provide the upper bound of collaborative camera-only 3D detection. We see that: i) both collaborations can consistently improve the performance; ii) camera-only 3D detection with collaboratively estimated depth significantly outperforms that with single-agent estimated depth and even approaches the upper bound with ground truth depth.

4.3. Qualitative evaluation

Visualization of depth and uncertainty. Fig. 5 shows that collaborative depth estimation outperforms single-agent depth estimation and approaches the ground truth depth. We see that: i) the single-agent depth estimation can estimate the relative depth while not able to precisely localize the depth candidate, for example, the vehicles are higher than the plane it sits while this plane is not grounded to the right category; ii) by introducing multi-view geometry, the collaboratively estimated depth can smoothly and accurately ground the plane; iii) the depth uncertainty is larger for the long-range and background areas. The reason is that the distant regions are hard to localize as they occupy too few image pixels, and the background regions are hard to localize due to the texture-less surfaces.

Visualization of detection results. Fig. 6 shows that compared to single LiDAR 3D detection, cameras with collaborations (Co-Depth and Co-FL) are able to achieve holistic and accurate detection results. The reason is that the single agent has some fundamental physical limitations such as long-range and occlusion issues. Fig. 6(a-c) shows that augmenting a single agent sensor going from just a cam-
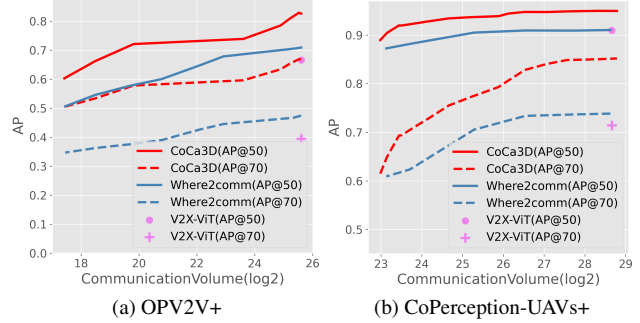

(a) OPV2V+ (b) CoPerception-UAVs+

Figure 4. `CoCa3D` achieves superior detection performance and communication cost trade-off over various bandwidths.

Table 3. Co-Depth significantly improves depth accuracy.

	Full plane			Foreground	
Single	Collaboration	Gain	Single	Collaboration	Gain
0.6167	0.7781	26.17% ↑	0.7586	0.8133	7.21% ↑

Table 4. Ablation of collaborative depth estimation on CoPerception-UAVs+. Uniform/Linear denotes uniform/linear-increasing spacing. Dense and sparse supervision denote applying depth supervision over the full image plane and the object regions.

Spacing	Supervision	AP@50	AP@70	AP@80
Uniform	Dense	**0.9502**	0.8444	0.5746
Uniform	Sparse	0.9330	0.8042	0.4829
Linear	Dense	0.9484	0.8365	0.5580
Linear	Sparse	0.9495	**0.8518**	**0.5849**

era to including a depth sensor to LiDAR can improve the detection but still not able to achieve holistic detection. Fig. 6(d)/(b) and (a)/(e) compares the detection results with and without collaborative feature learning, we see that collaboration can help detect lots of the missed objects in the single-agent detection. Fig. 6(d,e,f) show that: i) both collaborations, Co-FL and Co-Depth, consistently improve the detection performance; ii) with collaboration, collaborative camera-only agents can outperform LiDAR in 3D detection.

4.4. Ablation studies

Trade-off between detection performance and communication cost. Fig. 4 compares the proposed `CoCa3D` with the previous communication-efficient solution Where2comm [6] in terms of the trade-off between detection performance (AP@IoU=50/70) and communication bandwidth. We see that: i) `CoCa3D` can adapt to varying bandwidths by adjusting the depth uncertainty threshold and the detection confidence threshold; ii) `CoCa3D` achieves superior detection performance and communication cost trade-off to Where2comm over varying communication bandwidth. The gain mainly comes from Co-Depth's improvement in depth estimation, which improves single-agent BEV features. With enhanced single BEV features, the Co-FL can generate better-augmented BEV features, resulting in improved detection performance.
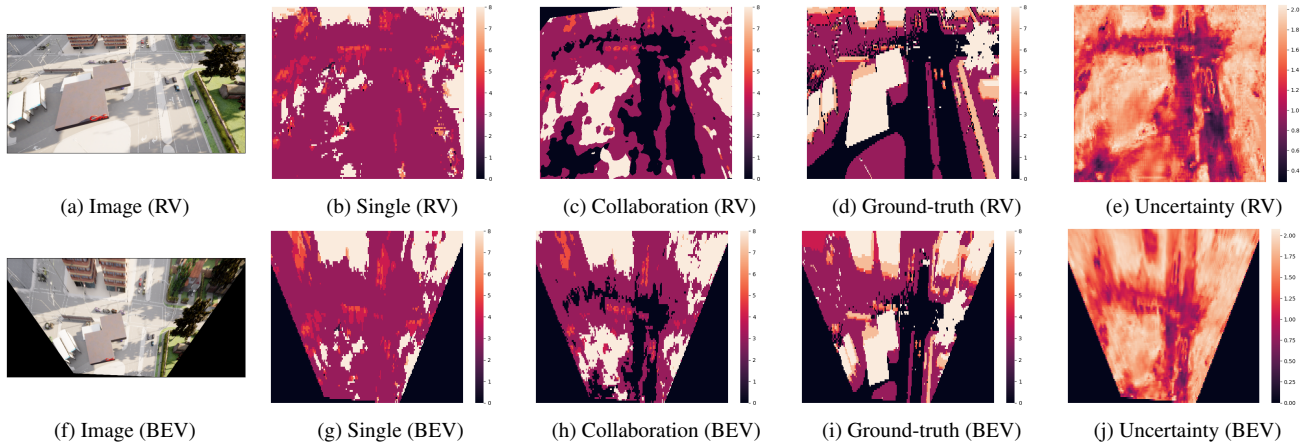
(a) Image (RV)   (b) Single (RV)   (c) Collaboration (RV)   (d) Ground-truth (RV)   (e) Uncertainty (RV)

(f) Image (BEV)   (g) Single (BEV)   (h) Collaboration (BEV)   (i) Ground-truth (BEV)   (j) Uncertainty (BEV)

Figure 5. Visualization of the depth and uncertainty in image range-view (RV) and bird's eye view (BEV) on CoPerception-UAVs+. Collaboratively estimated depth improves the single-agent estimated depth and approaches ground-truth depth.
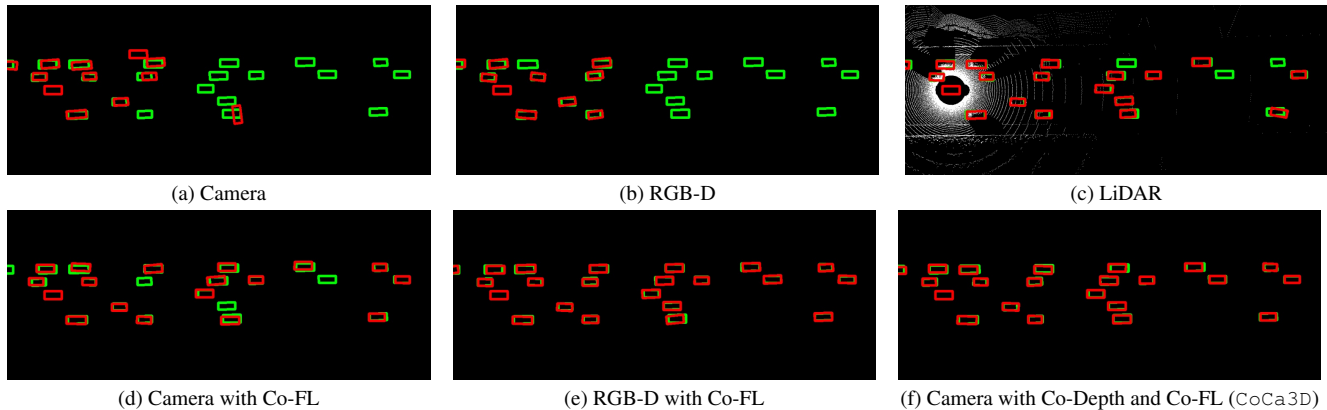

(a) Camera   (b) RGB-D   (c) LiDAR

(d) Camera with Co-FL   (e) RGB-D with Co-FL   (f) Camera with Co-Depth and Co-FL (`CoCa3D`)

Figure 6. `CoCa3D` outperforms LiDAR detection on OPV2V+ with 10 agents. Green and red boxes denote GT and detection respectively.

Effect of components in collaborative depth estimation. Tab. 3 assesses the gain of collaborative depth estimation (Co-Depth) over single-agent depth estimation (S-Depth) in the depth accuracy metric. We see that: i) Co-Depth steadily improves S-Depth over the full plane; ii) the depth accuracy of the foreground objects is higher than the full plane, as the texture-less backgrounds have fewer cues to localize. Tab. 4 assesses the effects of the depth spacing and supervision choices. We see that: i) overall, the Co-Depth is robust to the different spacing and supervision choices on AP@50/70; ii) linear-increasing spacing is robust to the various supervision choices that uniform spacing which fails the stricter metric AP@80 without dense supervision. The reason is that the linearly increasing spacing takes into account the prior depth distribution and assigns more depth candidates to frequently occurring depth ranges, which contributes to more balanced and easier depth learning.

5. Conclusion and Limitation

We propose `CoCa3D`, a novel collaborative camera-only 3D detection that approaches holistic 3D detection. The core idea is to introduce multi-agent collaboration to improve the detection ability of cameras. Meanwhile, the communication cost is optimized, and each agent carefully selects the spatially sparse yet critical depth messages to share. Extensive experiments covering both real-world and simulation scenarios, and multi-type agents (cars, drones, and infrastructures) show that `CoCa3D` not only achieves state-of-the-art perception-bandwidth trade-off, but overtakes LiDAR-based detectors with a sufficient number of collaborative agents on OPV2V+.

Limitation and future work. It is expensive to collect a real-world multi-agent perception dataset. So far, DAIR-V2X is the sole public real-world dataset, which only has one vehicle and one roadside unit. This work mainly leverages simulation data to validate the proposed novel methods and sketch a promising research direction. We advocate more resources for real-world data collection.

References

[1] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020. 1

[2] Siheng Chen, Baoan Liu, Chen Feng, Carlos Vallespi-Gonzalez, and Carl K. Wellington. 3d point cloud processing and learning for autonomous driving: Impacting map creation, localization, and perception. *IEEE Signal Processing Magazine*, 38:68–86, 2021. 1

[3] Jiajun Deng, Shaoshuai Shi, Peiwei Li, Wengang Zhou, Yanyong Zhang, and Houqiang Li. Voxel r-cnn: Towards high performance voxel-based 3d object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 1201–1209, 2021. 1

[4] 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. 6

[5] Peng Gao, Rui Guo, Hongsheng Lu, and Hao Zhang. Regularized graph matching for correspondence identification under uncertainty in collaborative perception. *Robotics: Science and Systems XVI*, 2020. 2

[6] Yue Hu, Shaoheng Fang, Zixing Lei, Yiqi Zhong, and Siheng Chen. Where2comm: Communication-efficient collaborative perception via spatial confidence maps. *Advances in Neural Information Processing Systems*, 2022. 2, 5, 6, 7

[7] Yue Hu, Shaoheng Fang, Weidi Xie, and Siheng Chen. Aerial monocular 3d object detection. *IEEE Robotics and Automation Letters*, 8:1959–1966, 2022. 1, 2

[8] Junjie Huang and Guan Huang. Bevdet4d: Exploit temporal cues in multi-camera 3d object detection. *arXiv preprint arXiv:2203.17054*, 2022. 1, 2

[9] Junjie Huang, Guan Huang, Zheng Zhu, and Dalong Du. Bevdet: High-performance multi-camera 3d object detection in bird-eye-view. *arXiv preprint arXiv:2112.11790*, 2021. 1, 2

[10] 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*, pages 12697–12705, 2019. 1, 2, 3, 6

[11] Zixing Lei, Shunli Ren, Yue Hu, Wenjun Zhang, and Siheng Chen. Latency-aware collaborative perception. *ECCV*, 2022. 2

[12] Jing Li, Rui Li, Jiehao Li, Junzheng Wang, Qingbin Wu, and Xu Liu. Dual-view 3d object recognition and detection via lidar point cloud and camera image. *Robotics Auton. Syst.*, 150:103999, 2022. 1

[13] Yiming Li, Dekun Ma, Ziyan An, Zixun Wang, Yiqi Zhong, Siheng Chen, and Chen Feng. V2x-sim: Multi-agent collaborative perception dataset and benchmark for autonomous driving. *IEEE Robotics and Automation Letters*, 2022. 2

[14] 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*, 34:29541–29552, 2021. 2, 6

[15] Yiming Li, Juexiao Zhang, Dekun Ma, Yue Wang, and Chen Feng. Multi-robot scene completion: Towards task-agnostic collaborative perception. In *Conference on Robot Learning*, 2022. 2

[16] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Qiao Yu, and Jifeng Dai. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. *arXiv preprint arXiv:2203.17270*, 2022. 2

[17] Yen-Cheng Liu, Junjiao Tian, Nathaniel Glaser, and Zsolt Kira. When2com: Multi-agent perception via communication graph grouping. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pages 4106–4115, 2020. 2, 6

[18] Yen-Cheng Liu, Junjiao Tian, Chih-Yao Ma, Nathan Glaser, Chia-Wen Kuo, and Zsolt Kira. Who2com: Collaborative perception via learnable handshake communication. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6876–6883. IEEE, 2020. 2

[19] Zhijian Liu, Haotian Tang, Alexander Amini, Xinyu Yang, Huizi Mao, Daniela Rus, and Song Han. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. *ArXiv*, abs/2205.13542, 2022. 1

[20] Yifan Lu, Quanhao Li, Baoan Liu, Mehrdad Dianati, Chen Feng, Siheng Chen, and Yanfeng Wang. Robust collaborative 3d object detection in presence of pose errors. *ICRA*, 2023. 2

[21] Jiageng Mao, Minzhe Niu, Haoyue Bai, Xiaodan Liang, Hang Xu, and Chunjing Xu. Pyramid r-cnn: Towards better performance and adaptability for 3d object detection. *ICCV*, 2021. 1

[22] Lang Peng, Zhirong Chen, Zhangjie Fu, Pengpeng Liang, and Erkang Cheng. Bevsegformer: Bird's eye view semantic segmentation from arbitrary camera rigs. *arXiv preprint arXiv:2203.04050*, 2022. 2

[23] Jonah Philion and Sanja Fidler. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *European Conference on Computer Vision*, pages 194–210. Springer, 2020. 1, 2

[24] 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. 2

[25] Cody Reading, Ali Harakeh, Julia Chae, and Steven L. Waslander. Categorical depth distribution network for monocular 3d object detection. *CVPR*, 2021. 1, 2, 3, 5, 6

[26] Thomas Roddick, Alex Kendall, and Roberto Cipolla. Orthographic feature transform for monocular 3d object detection. *arXiv preprint arXiv:1811.08188*, 2018. 1, 2

[27] Shital Shah, Debadeepta Dey, Chris Lovett, and Ashish Kapoor. Airsim: High-fidelity visual and physical simulation for autonomous vehicles. In *Field and service robotics*, pages 621–635. Springer, 2018. 6

[28] 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. 1, 2

[29] 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. 2

[30] Tai Wang, Jiangmiao Pang, and Dahua Lin. Monocular 3d object detection with depth from motion. In *European Conference on Computer Vision (ECCV)*, 2022. 1, 2

[31] 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*, pages 605–621. Springer, 2020. 2, 6

[32] Yue Wang, Vitor Campagnolo Guizilini, Tianyuan Zhang, Yilun Wang, Hang Zhao, and Justin Solomon. Detr3d: 3d object detection from multi-view images via 3d-to-2d queries. In *Conference on Robot Learning*, pages 180–191. PMLR, 2022. 1

[33] Pengxiang Wu, Siheng Chen, and Dimitris N. Metaxas. Motionnet: Joint perception and motion prediction for autonomous driving based on bird's eye view maps. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11382–11392, 2020. 1

[34] 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. 6

[35] Runsheng Xu, Zhengzhong Tu, Hao Xiang, Wei Shao, Bolei Zhou, and Jiaqi Ma. CoBEVT: Cooperative bird's eye view semantic segmentation with sparse transformers. *CoRL*, 2022. 2

[36] Runsheng Xu, Hao Xiang, Zhengzhong Tu, Xin Xia, Ming-Hsuan Yang, and Jiaqi Ma. V2x-vit: Vehicle-to-everything cooperative perception with vision transformer. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXIX*, pages 107–124. Springer, 2022. 2, 5, 6

[37] 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 International Conference on Robotics and Automation (ICRA)*, pages 2583–2589. IEEE, 2022. 2, 5, 6

[38] Zetong Yang, Yanan Sun, Shu Liu, and Jiaya Jia. 3dssd: Point-based 3d single stage object detector. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11040–11048, 2020. 2

[39] Tianwei Yin, Xingyi Zhou, and Philipp Krähenbühl. Center-based 3d object detection and tracking. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11779–11788, 2021. 1

[40] Haibao Yu, Yizhen Luo, Mao Shu, Yiyi Huo, Zebang Yang, Yifeng Shi, Zhenglong Guo, Hanyu Li, Xing Hu, Jirui Yuan, and Zaiqing Nie. Dair-v2x: A large-scale dataset for vehicle-infrastructure cooperative 3d object detection. In *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, June 2022. 2, 6

[41] Yunpeng Zhang, Zheng Zhu, Wenzhao Zheng, Junjie Huang, Guan Huang, Jie Zhou, and Jiwen Lu. Beverse: Unified perception and prediction in birds-eye-view for vision-centric autonomous driving. *arXiv preprint arXiv:2205.09743*, 2022. 1, 2

[42] Wu Zheng, Weiliang Tang, Sijin Chen, Li Jiang, and Chi-Wing Fu. Cia-ssd: Confident iou-aware single-stage object detector from point cloud. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 3555–3562, 2021. 1, 2

[43] Brady Zhou and Philipp Krähenbühl. Cross-view transformers for real-time map-view semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13760–13769, 2022. 1, 2

[44] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. In *arXiv preprint arXiv:1904.07850*, 2019. 4, 5

[45] 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*, pages 4490–4499, 2018. 1, 2