

# Communication-Efficient Collaborative Perception via Information Filling with Codebook

Yue Hu<sup>1,4</sup> Juntong Peng<sup>1,4</sup> Sifei Liu<sup>1,4</sup> Junhao Ge<sup>1,4</sup> Si Liu<sup>3</sup> Siheng Chen<sup>1,2,4</sup>

<sup>1</sup>Cooperative Medianet Innovation Center, Shanghai Jiao Tong University <sup>2</sup>Shanghai AI Laboratory

<sup>3</sup>Beihang University <sup>4</sup>Multi-Agent Governance & Intelligence Crew (MAGIC)

<sup>1</sup>{18671129361, juntong.peng, hiraeth416, cancaries, sihengc}@sjtu.edu.cn <sup>3</sup>{liusi}@buaa.edu.cn

## Abstract

Collaborative perception empowers each agent to improve its perceptual ability through the exchange of perceptual messages with other agents. It inherently results in a fundamental trade-off between perception ability and communication cost. To address this bottleneck issue, our core idea is to optimize the collaborative messages from two key aspects: representation and selection. The proposed codebook-based message representation enables the transmission of integer codes, rather than high-dimensional feature maps. The proposed information-filling-driven message selection optimizes local messages to collectively fill each agent’s information demand, preventing information overflow among multiple agents. By integrating these two designs, we propose CodeFilling, a novel communication-efficient collaborative perception system, which significantly advances the perception-communication trade-off and is inclusive to both homogeneous and heterogeneous collaboration settings. We evaluate CodeFilling in both a real-world dataset, DAIR-V2X, and a new simulation dataset, OPV2VH+. Results show that CodeFilling outperforms previous SOTA Where2comm on DAIR-V2X/OPV2VH+ with 1,333/1,206× lower communication volume. Our code is available at <https://github.com/PhyllisH/CodeFilling>.

## 1. Introduction

Collaborative perception aims to enhance the perceptual ability of each individual agent by facilitating the exchange of complementary perceptual information among multiple agents [1, 10, 12, 15, 16, 19, 22, 24, 30, 31, 43]. It fundamentally overcomes the occlusion and long-range issues in single-agent perception [4, 11, 13, 45]. As the forefront of autonomous systems, collaborative perception shows significant potential in enormous real-world appli-

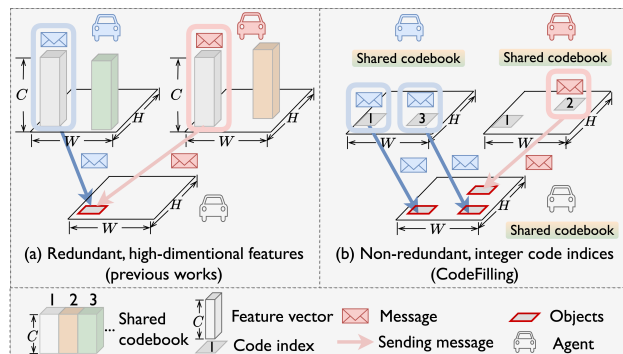


Figure 1. CodeFilling avoids redundant messages and achieves more complete detections by transmitting more critical perceptual information with the compact code index message.

cations, particularly vehicle-to-everything-communication-aided autonomous driving [2, 3, 18, 26, 28–30, 33, 38, 39].

In this emerging field, a central challenge lies in optimizing the trade-off between perception performance and communication cost inherent in agents sharing perceptual data [10, 12, 16, 19, 20, 30, 41, 42]. Given inevitable practical constraints of communication systems, efficient utilization of communication resources is the prerequisite for collaborative perception. To minimize communication overhead, a straightforward solution is late collaboration, where agents directly exchange the perception outputs. However, numerous previous works indicate that late collaboration yields marginal perception improvements and is vulnerable to various noises [10, 21]. To optimize the perception-communication trade-off, most studies consider intermediate collaboration, where the collaborative messages are perceptual features [10, 12, 16, 19, 20, 29, 35, 37, 40]. For example, When2com [19] proposed the handshake strategy to limit the number of collaborators; and Where2comm [10] proposed a pragmatic strategy that only transmits messages about constrained spatial areas. While these methods mitigate certain communication costs, they still necessitate the transmission of high-dimensional feature maps, which incurs substantial communication expenses.

To overcome the limitations of intermediate collabora-

tion, our core idea is to optimize the collaborative messages from two key perspectives: representation and selection. For message representation, we introduce a codebook to standardize the communication among agents, where each code is analogous to a word in the human language dictionary. Based on this shared codebook among all agents, we can use the codes to approximate perceptual features; consequently, only integer code indices need to be exchanged, eliminating the need for transmitting high-dimensional features comprised of floating-point numbers. For message selection, we propose an information filling strategy, akin to piecing together a jigsaw puzzle. In this approach, assuming an agent’s information demand is upper bounded, each of its collaborators performs a local optimization to select non-redundant messages to fill its information gap. This strategy prevents information overflow among multiple agents, further significantly reducing communication cost.

Following the above spirit, we propose *CodeFilling*, a novel communication-efficient collaborative 3D detection system; see Figure 2. The proposed *CodeFilling* includes four key modules: i) a single-agent detector, providing basic detection capabilities; ii) an novel *information-filling-driven message selection*, which solves local optimizations for choosing pertinent messages to optimally fill other agents’ information demands without causing information flow; iii) an novel *codebook-based message representation*, which leverages a task-driven codebook to achieve pragmatic approximation of feature maps, enabling the transmission of integer code indices; and iv) a message decoding and fusion module, which integrates the messages to achieve enhanced collaborative detections.

*CodeFilling* offers two distinct advantages: i) it delivers a substantial advancement in the perception-communication trade-off through the transmission of non-redundant code indices; and ii) it is inclusive to both settings of homogeneous and heterogeneous agents by leveraging a standardized code representation; that is, feature maps obtained from various perception models and diverse sensors can be aligned to the unified feature space provided by the proposed codebook.

To evaluate *CodeFilling*, we conduct extensive experiments on real-world dataset DAIR-V2X, and a new simulation dataset OPV2VH+ under two homogeneous (LiDAR, camera) and one heterogeneous setting. The results show that i) *CodeFilling* achieves superior performance performances than Where2comm, the current SOTA, with  $1333/1206 \times$  less communication cost on DAIR-V2X/OPV2VH+; and ii) *CodeFilling* maintains superior trade-off in both homogeneous and heterogeneous settings, establishing an inclusive collaboration system.

To sum up, our main contributions are three-fold:

- We propose *CodeFilling*, a novel communication-efficient collaborative 3D detection system, which signifi-

cantly improves the perception- communication trade-off and is inclusive to both homogeneous and heterogeneous collaboration settings;

- We propose two novel methods to optimize collaborative messages: codebook-based message representation and information-filling-driven message selection;

- We conduct comprehensive experiments to validate that *CodeFilling* achieves SOTA perception-communication trade-off across varying communication bandwidths, on both real-world and simulation datasets in both homogeneous and heterogeneous settings.

## 2. Related works

**Collaborative Perception.** Collaborative perception [10, 12, 16, 17, 19–21, 30, 32, 35, 36, 43] is an emerging application of multi-agent communication systems to perception tasks, which promote the crucial perception module through communication-enabled complementary perceptual information sharing. Several high-quality datasets have emerged [10, 12, 17, 35, 43, 44] to aid in the algorithm development. Collaborative perception systems have made remarkable progress in improving perception performance [36, 37] and robustness on practical issues, such as communication bandwidth constraints, pose error [21, 27] and latency [14, 31]. Here, considering that communication efficiency is the bottleneck issue for the scale-up of collaborative perception, we aim to optimize the performance-communication trade-off instead of solely promoting the perception performance regardless of bandwidth costs.

**Communication efficiency in collaborative perception.** To address this bottleneck issue, prior methods have made efforts in two key aspects: message selection and message representation. For message selection, When2com [19] and Who2comm [20] employ a handshake mechanism, to select information from all the relevant collaborators. Furthermore, Where2comm [10] and CoCa3D [12] expand the selection process to incorporate spatial dimensions. For message representation, intermediate feature representation [10, 12, 16, 17, 30, 36, 37] has demonstrated a more balanced performance-communication trade-off. Source coding [30] and channel compression [16] techniques are used to further enhance feature representation efficiency. However, previous methods accumulate redundant information from various collaborators and still transmit high-dimensional feature vectors, incurring high communication costs. Here, we facilitate essential supportive information exchange among agents with compact codebook-based representation, efficiently enhancing detection performance.

**Codebook compression.** Codebook compression, a lossless compression technique, effectively captures the essence of high-dimensional vectors through the combination of these codes [7]. It has diverse applications, ranging from digital image compression [8] to the neural network parameters compression [9]. Recently, task-adaptive codebooks

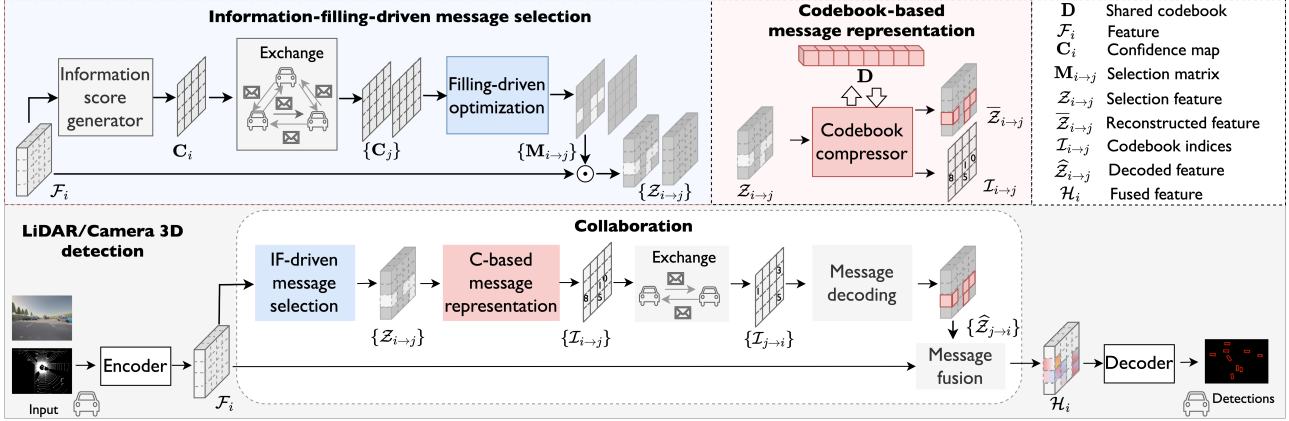


Figure 2. CodeFilling is a novel communication-efficient collaborative 3D detection system. The proposed information-filling-driven message selection and codebook-based message representation contribute to optimizing collaborative messages.

have emerged. Rather than pursuing lossless compression, it drops task-irrelevant information and focuses on essential information for specific downstream tasks, further improving representation efficiency [5, 25]. However, existing task-adaptive codebooks have largely concentrated on 2D classification tasks [25]. Here, we explore the novel realm of collaborative 3D object detection, introducing fresh challenges for codebook compression. This entails preserving an extensive feature set for precise 3D interpretation and adapting to fluctuating communication bandwidths, necessitating versatile codebook configurations.

### 3. Problem Formulation

Consider  $N$  homogeneous or heterogeneous agents in the scene, each has its individual perceptual task and unique sensor setup. To enhance the perception abilities, the agents exchange complementary perceptual information, forming a decentralized mutually beneficial collaboration network. Each agent concurrently acts as both a supporter and a receiver. In their role as supporters, they contribute perceptual information to assist their counterparts. Conversely, as receivers, they gain from the messages provided by others. Such collaborative perception leads to a holistic enhancement of perceptual capabilities. Here we focus on 3D object detection. Let  $\mathcal{X}_i$  be the input collected by the  $i$ th agent's sensor (LiDAR or camera), and  $\mathcal{O}_i^0$  be the corresponding ground-truth detection. The objective is to maximize the detection performances of all agents given certain communication budget  $B$ ; that is,

$$\max_{\theta, \mathcal{P}} \sum_i g\left(\Phi_{\theta}\left(\mathcal{X}_i, \{\mathcal{P}_{j \rightarrow i}\}_{j=1}^N\right), \mathcal{O}_i^0\right), \text{ s.t. } \sum_{i, j, j \neq i} b(\mathcal{P}_{j \rightarrow i}) \leq B, \quad (1)$$

where  $g(\cdot, \cdot)$  is the detection evaluation metric,  $\Phi(\cdot)$  is a detection model with trainable parameter  $\theta$ ,  $\mathcal{P}_{j \rightarrow i}$  is the message transmitted from the  $i$ th agent to the  $j$ th agent, and  $b(\cdot)$  measures the communication cost of the collaborative messages. The key challenge is to determine the messages

$\mathcal{P}_{j \rightarrow i}$ , which should be both informative and compact.

## 4. CodeFilling: Collaborative 3D Detection

To optimize the trade-off between perception ability and communication cost, we present CodeFilling, a novel communication-efficient collaborative 3D detection system; see Fig. 2. It has two parts: i) single-agent 3D detection, which allows an agent to equip basic detection ability, implementing  $\Phi_{\theta}$  in (1), and ii) multi-agent collaboration, enhancing an agent's detection ability through the exchange of efficient perceptual messages  $\mathcal{P}$  in (1).

### 4.1. Single-agent 3D detection

An agent learns to detect 3D objects based on its sensor inputs. It involves an observation encoder and a detection decoder. CodeFilling allows agents to accept multi-modality inputs, including RGB images and 3D point clouds. Each agents with its distinct modality projects its perceptual information to the unified global bird's eye view (BEV) coordinate system, better supporting inter-agent collaboration and more compatible with both homogeneous and heterogeneous settings.

**Observation encoder.** The observation encoder extracts feature maps from the sensor data. For the  $i$ th agent, given its input  $\mathcal{X}_i$ , the BEV feature map is  $\mathcal{F}_i = \Phi_{\text{enc}}(\mathcal{X}_i) \in \mathbb{R}^{H \times W \times C}$ , where  $\Phi_{\text{enc}}(\cdot)$  is the encoder and  $H, W, C$  are its height, weight and channel. For image inputs,  $\Phi_{\text{enc}}(\cdot)$  is followed by an additional warping function that transforms the extracted front-view feature to BEV. The BEV feature is output to the decoder, and also message selection and fusion modules when collaboration is established.

**Detection decoder.** The detection decoder decodes features into objects, including class and regression output. Given the feature map  $\mathcal{F}_i$ , the detection decoder  $\Phi_{\text{dec}}(\cdot)$  generate the detections of  $i$ th agent by  $\mathcal{O}_i = \Phi_{\text{dec}}(\mathcal{F}_i) \in \mathbb{R}^{H \times W \times 7}$ , where each location of  $\mathcal{O}_i$  represents a rotated box with class  $(c, x, y, h, w, \cos \alpha, \sin \alpha)$ , denoting class confidence, position, size and angle.

## 4.2. Multi-agent collaboration

In the proposed multi-agent collaboration, each agent acts in the dual role of supporter and receiver. As a supporter, each agent employs two novel modules, including information-filling-driven message selection and codebook-based message representation, to determine compact, yet supportive collaboration messages to help others. These two proposed modules enhance communication efficiency in both spatial and channel dimensions of a feature map, respectively. As a receiver, each agent employs a message decoding and fusion module to integrate supportive messages from other agents, improving its perceptual performance.

### 4.2.1 Information-filling-driven message selection

To efficiently select compact collaborative messages that support other agents, each agent employs a novel information-filling-driven message selection method. The key idea is to enable each agent to restrainedly select pertinent messages to share with other agents; then collectively, these pieces of non-redundant messages mutually fulfill each other's information demands. For example, in occlusion scenarios, extra information from supporters helps an agent detect missed objects. However, overfilled information from multiple supporters wastes communication resources. Thus, collective coordination is essential to avoid redundancy and enable more beneficial information. To achieve this, the proposed selection has two key steps: information disclosure, wherein agents mutually share their awareness of available information within specific spatial areas, and filling-driven optimization, wherein each agent locally optimizes the supportive messages for others.

**Information disclosure.** In information disclosure, each agent: i) employs an information score generator to create its information score map from its feature map, reflecting its available information at each spatial area, and ii) broadcasts this map to all other agents, promoting a mutually thorough awareness of all the available support.

The information score map is implemented with the detection confidence map. Intuitively, the areas containing an object are likely to offer more useful information for revealing missed detections and, therefore, should be assigned higher information scores. Specifically, given a BEV feature map  $\mathcal{F}_i$ , its spatial information score map is

$$\mathbf{C}_i = \Phi_{\text{generator}}(\mathcal{F}_i) \in [0, 1]^{H \times W}, \quad (2)$$

where  $\Phi_{\text{generator}}(\cdot)$  is implemented by detection decoder. When information score map is generated, each agent broadcasts it to other agents. This initial communication is efficient because of lightweight information score maps.

**Filling-driven optimization.** In the role of a supporter, each agent gathers the other agents' information score maps and determines who needs perceptual information at which spatial areas by locally solving a filling-driven optimization. Here, filling the information demands means that an agent

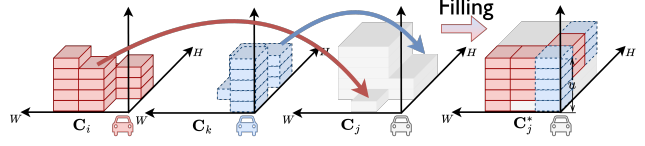


Figure 3. The information-filling-driven message selection fulfills the information demand with non-redundant information.

only requires the necessary information at certain spatial areas for precise detection, as extra information no longer provides significant benefits. This requires each supporter to prioritize non-redundant and informative spatial regions with higher scores to assist others and halts the selection once the receiver's information demands are fulfilled.

Specifically, the optimization is formulated as a proxy-constrained problem and obtains a binary selection matrix for each agent to support each receiver. Let  $\mathbf{M}_{i \rightarrow j} \in \{0, 1\}^{H \times W}$  be the binary selection matrix supported on the BEV map. Each element in the matrix indicates whether Agent  $i$  should send the information to Agent  $j$  at a specific spatial location (1 for sending information, and 0 for not sending). To solve for the binary selection matrix, the proxy-constrained problem is formulated as follows,

$$\{\mathbf{M}_{i \rightarrow j}^*\}_{i,j} = \underset{\mathbf{M}}{\operatorname{argmax}} \sum_{j=1}^N f_{\min} \left( \mathbf{C}_j + \sum_{i=1, i \neq j}^N \mathbf{M}_{i \rightarrow j} \odot \mathbf{C}_i, u \right), \quad (3a)$$

$$\text{where } \sum_{i,j=1, j \neq i}^N \mathbf{M}_{i \rightarrow j} \leq b, \mathbf{M}_{i \rightarrow j} \in \{0, 1\}^{H \times W}. \quad (3b)$$

Here  $\odot$  denotes element-wise multiplication, and the scalar  $u$  is a hyper-parameter to reflect the upper bound of information demand. The function  $f_{\min}(\cdot, \cdot)$  computes the element-wise minimum between a matrix and a scalar.

In (3a),  $\mathbf{C}_j + \sum_{i=1, i \neq j}^N \mathbf{M}_{i \rightarrow j} \odot \mathbf{C}_i$  indicates that each receiver  $j$  accumulates the information transmitted from all supporters, combined with its own information, at each location,  $f_{\min}(\mathbf{C}_j + \sum_{i=1, i \neq j}^N \mathbf{M}_{i \rightarrow j} \odot \mathbf{C}_i, u)$  denotes the utility for each receiver, linearly increasing with the accumulated information scores until reaching the information demand  $u$ . Note that: i) (3a) is solved at the supporter side for preparing messages to a receiver; ii) the sum-based utility motivates supporters to collectively meet the receiver's demand and focus on higher-scoring regions, and iii) the cutoff point leads to halting selection to prevent redundancy.

Equation (3a) transforms the feature-based collaboration utility in (1) as the sum of the information scores. This is based on the assumption that the accumulation of information scores mirrors the benefits of feature aggregation. Equation (3b) addresses the bandwidth limitation in (1) by quantifying the total number of selected regions. This approach simplifies the objective in (1) into a proxy-constrained problem in (3a) and (3b). The optimized selection solution derived from (1) is expected to yield a superior outcome in the final feature-based collaboration.

This optimization problem has an analytical solution; see the theoretical derivation in the appendix. The solving pro-

cess incurs a computational cost of  $O(\log(m))$ , where  $m$  denotes the number of spatial region candidates. By focusing on the extremely sparse foreground areas, we effectively reduce the cost to a negligible level, enabling each agent to provide more targeted support for others with minimal cost.

Based on the optimized selection matrix  $\{\mathbf{M}_{i \rightarrow j}^*\}_{j=1}^N$ , each agent supports each collaborator with a sparse yet informative feature map  $\mathcal{Z}_{i \rightarrow j} = \mathbf{M}_{i \rightarrow j}^* \odot \mathcal{F}_i$ , promising superior perception improvements given the limited communication budget. These selected sparse feature maps are then output to the message representation module.

The proposed message selection offers two key benefits: i) it avoids redundancy from multiple supporters via collective selection, and ii) it adapts to varying communication conditions by adjusting information demand, lower demand for efficiency in limited budgets, and higher demand for superior performance in ample budgets. Compared to existing selection methods [10, 12, 19, 20], which are based on individual supporter-receiver pairs, our collective optimization further reduces redundancy across various supporters.

#### 4.2.2 Codebook-based message representation

To efficiently transmit the selected feature map  $\mathcal{Z}_{i \rightarrow j}$ , each agent leverages a novel codebook-based message representation, reducing communication cost along the channel dimension. The core idea is to approximate a high-dimensional feature vector by the most relevant code from a task-driven codebook; as a result, only integer code indices need to be transmitted, rather than the complete feature vectors composed of floating-point numbers.

**Codebook learning.** Analogous to a language dictionary used by humans, our task-driven codebook is shared among all agents to standardize their communication for achieving the detection task. This codebook consists of a set of codes, which are learned to pragmatically approximate possible perceptual features present in the training dataset. Here the pragmatic approximation refers to each code serving as a lossy approximation of a feature vector, while retaining essential information necessary for the downstream detection task within that vector. Specifically, let  $F = \{\mathcal{F}^{(i,s)}\}_{i=1, s=1}^{N,S}$  be the collective set of BEV feature maps extracted by the observation encoders of all  $N$  agents across all  $S$  training scenes. Let  $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_{n_L}] \in \mathbb{R}^{C \times n_L}$  be the codebook, where  $\mathbf{D}_{[\ell]} = \mathbf{d}_\ell \in \mathbb{R}^C$  is the  $\ell$ th code and  $n_L$  is the number of codes.

The task-driven codebook is learned through feature approximation at each spatial location; that is,

$$\mathbf{D}^* = \arg \min_{\mathbf{D}} \sum_{F \in \mathcal{F}} \sum_{h,w} \min_{\ell} \left( \Psi(\mathbf{D}_{[\ell]}) + \|\mathcal{F}_{[h,w]} - \mathbf{D}_{[\ell]}\|_2^2 \right), \quad (4)$$

where  $\Psi(\cdot)$  denotes the resulting detection performance achieved by substituting  $\mathbf{D}_{[\ell]}$  for  $\mathcal{F}_{[h,w]}$ . The first term pertains to the requirements of the downstream detection

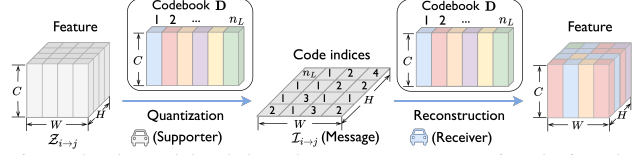


Figure 4. The codebook-based message representation depicts the original feature vector with the most relevant codes.

task and the second term reflects the reconstruction error between the original feature vector and the code. This approximation is lossy for reconstruction while lossless for the perceptual task, enabling the reduction of communication cost without sacrificing perceptual capacity.

**Code index representation.** Based on the shared codebook  $\mathbf{D}$ , each agent can substitute the selected sparse feature map  $\mathcal{Z}_{i \rightarrow j}$  by a series of code indices  $\mathcal{I}_{i \rightarrow j}$ . For each BEV location  $(h, w)$ , the code index is obtained as,

$$(\mathcal{I}_{i \rightarrow j})_{[h,w]} = \arg \min_{\ell} \left\| (\mathcal{Z}_{i \rightarrow j})_{[h,w]} - \mathbf{D}_{[\ell]} \right\|_2^2. \quad (5)$$

The codebook offers versatility in its configuration by adjusting both the codebook size  $n_L$  and the quantity of codes  $n_R$  used for representing the input vector. Equation (5) demonstrates a specific instance where  $n_R = 1$ , chosen for simplicity in notation. When  $n_R$  is larger, the representation involves a combination of multiple codes.

Overall, the final message sent from the  $i$ th agent to the  $j$ th agent is  $\mathcal{P}_{i \rightarrow j} = \mathcal{I}_{i \rightarrow j}$ , conveying the required complementary information with compact code indices. Agents exchange these packed messages with each other.

This codebook-based representation offers three advantages: i) efficiency for transmitting lightweight code indices; ii) adaptability to various communication resources via adjusting code configurations (smaller for efficiency, larger for superior performance), and iii) extensibility by providing a shared standardized representation. New heterogeneous agents can easily join the collaboration by adding its effective perceptual feature basis to the codebook.

#### 4.2.3 Message decoding and fusion

Message decoding reconstructs the supportive features based on the received code indices and the shared codebook. Given the received message  $\mathcal{P}_{j \rightarrow i} = \mathcal{I}_{j \rightarrow i}$ , the decoded feature map's  $\hat{\mathcal{Z}}_{j \rightarrow i} \in \mathbb{R}^{H \times W \times C}$  element located at  $(h, w)$  is  $(\hat{\mathcal{Z}}_{j \rightarrow i})_{[h,w]} = \mathbf{D}_{[\mathcal{I}_{j \rightarrow i}[h,w]]}$ . Subsequently, message fusion aggregates these decoded feature maps to augment individual features, implementing by the non-parametric point-wise maximum fusion. For the  $i$ th agent, given the reconstructed feature  $\hat{\mathcal{Z}}_{j \rightarrow i}$ . The enhanced BEV feature is obtained as  $\mathcal{H}_i = \max_{j \in \mathcal{N}_i} (\mathcal{F}_i, \hat{\mathcal{Z}}_{j \rightarrow i}) \in \mathbb{R}^{H \times W \times C}$  where  $\mathcal{N}_i$  is  $i$ -th agent's connected collaborators and  $\max(\cdot)$  maximizes the corresponding features from multiple agents at each individual spatial location. The enhanced feature  $\mathcal{H}_i$  is decoded to generate the upgraded detection  $\hat{\mathcal{O}}_i$ .

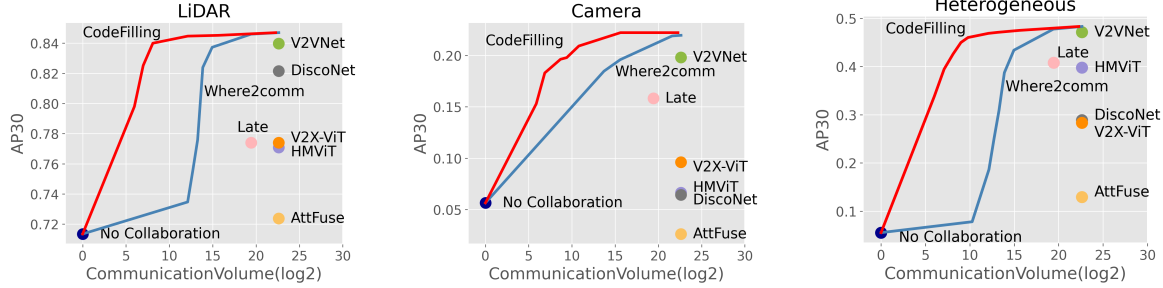


Figure 5. In DAIR-V2X, CodeFilling achieves the best perception-communication trade-off in *homogeneous & heterogeneous* settings.

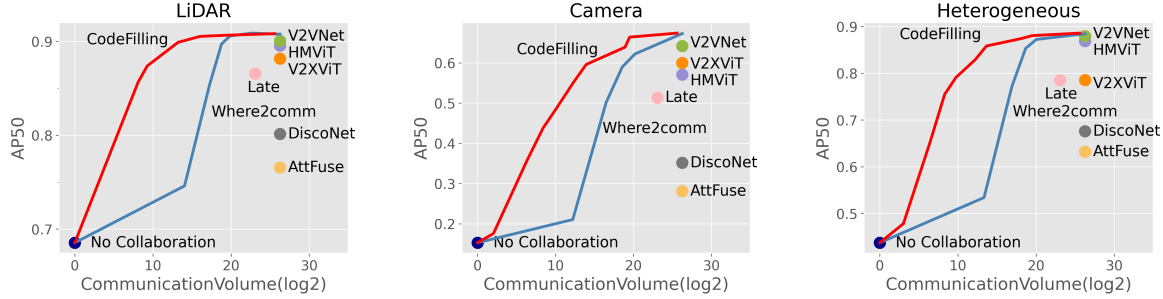


Figure 6. In OPV2VH+, CodeFilling achieves the best perception-communication trade-off in *homogeneous & heterogeneous* settings.

### 4.3. Loss functions

To train the overall system, we supervise three tasks: information score map generation, object detection, and codebook learning. The information score map generator reuses the parameters of the detection decoder. The overall loss is defined as,  $L = \sum_i^N L_{\text{det}}(\hat{\mathcal{O}}_i, \mathcal{O}_i^0) + \|\mathcal{F}_i - \bar{\mathcal{F}}_i\|_2^2$ , where  $L_{\text{det}}(\cdot)$  denotes the detection loss [45],  $\mathcal{O}_i^0$  and  $\hat{\mathcal{O}}_i$  represents the ground-truth and predicted objects, and  $\mathcal{F}_i$  and  $\bar{\mathcal{F}}_i$  denote the  $i$ -th agent’s original feature map and the one approximated by codes. During the optimization, the network parameters and the codebook are updated simultaneously.

## 5. Experimental Results

Our experiments cover two datasets, both real-world and simulation scenarios, two types of sensors (LiDAR and cameras), and both homogeneous and heterogeneous settings. Specifically, we conduct 3D object detection in the setting of V2X-communication-aided autonomous driving on DAIR-V2X dataset [43] and the extended large-scale OPV2VH+ dataset. The detection results are evaluated by Average Precision (AP) at Intersection-over-Union (IoU) thresholds of 0.30 and 0.50. The communication volume follows the standard setting as [10, 12, 33, 35] that counts the message size by byte in log scale with base 2.

### 5.1. Datasets and experimental settings

**DAIR-V2X** [43] is a widely-used **real-world** collaborative perception dataset. Each scene contains two agents: a vehicle and a road-side-unit. Each agent is equipped with a LiDAR and a camera. The perception range is 204.8m×102.4m. **OPV2VH+** is an extended large-scale version of the original vehicle-to-vehicle camera-only col-

laborative perception dataset OPV2V+ [12] with a larger array of collaborative agents (a total of 10) and additional LiDAR sensors, co-simulated by OpenCDA [34] and CARLA [6]. Each agent has a LiDAR, 4 cameras, and 4 depth sensors. The detection range is 281.6m × 80m.

**Implementation.** We adopt PointPillar [13] and CaDDN [23] for the LiDAR and camera detector, respectively. Regarding the heterogeneous setup, agents are randomly assigned either LiDAR or camera, resulting in a balanced 1:1 ratio of agents across the different modalities.

**Communication volume.** Specifically, for feature representation, given a selection matrix  $\mathbf{M}$ , the bandwidth is calculated as  $\log_2(H \times W \times |\mathbf{M}| \times C \times 32/8)$ . Here, 32 represents the float32 data type and 8 converts bits to bytes. For code index representation, given codebook  $\mathbf{D} \in \mathbb{R}^{C \times n_L}$ , comprised of  $n_L$  codes and each vector constructed using  $n_R$  codes, the bandwidth given the selection matrix  $\mathbf{M}$  is calculated as  $\log_2(H \times W \times |\mathbf{M}| \times \log_2(n_L) \times n_R/8)$ . Here,  $\log_2(n_L)$  signifies the data amount required to represent each code index integer, decided by the codebook size.

### 5.2. Quantitative evaluation

**Benchmark comparison.** Fig. 6 and 5 compare the proposed CodeFilling with previous methods in terms of the trade-off between detection performance and communication bandwidth for DAIR-V2X and OPV2VH+ datasets under homogeneous and heterogeneous settings, respectively. Detailed values can also be found in the appendix. Baselines include no collaboration ( $\mathcal{O}_i$ ), Where2comm [10], HMViT [33], V2VNet [30], DiscoNet [16], V2X-ViT [37], AttFuse [35] and late fusion, where agents exchange the detected 3D boxes directly. Note that HMViT is specifically designed for heterogeneous set-

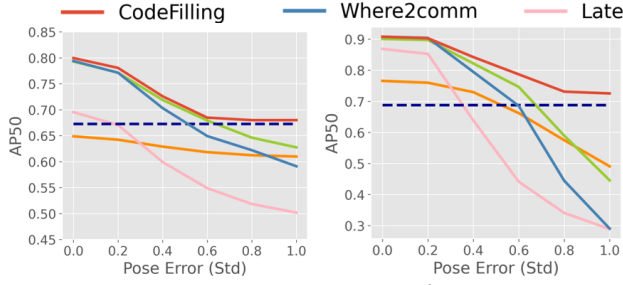


Figure 7. CodeFilling is robust to pose error issue.

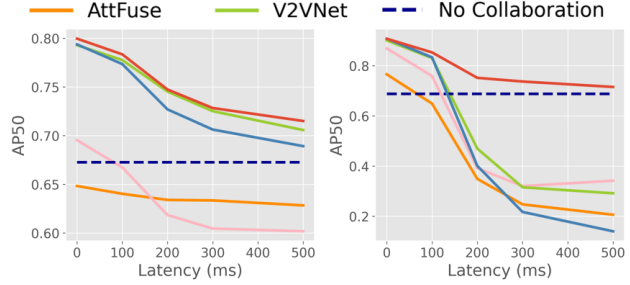


Figure 8. CodeFilling is robust to communication latency issue.

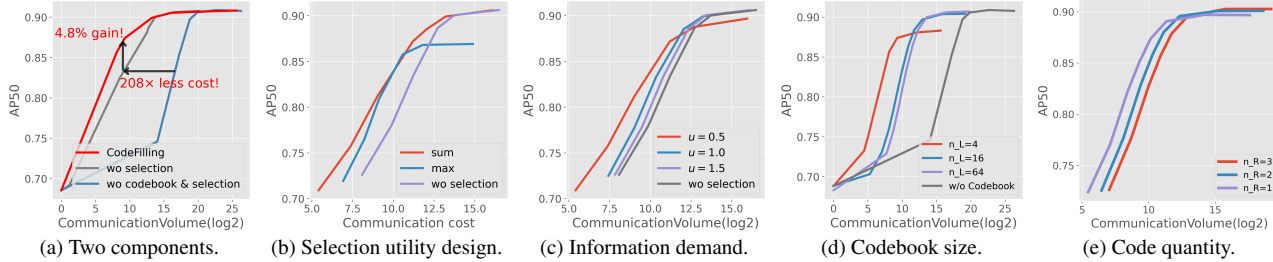


Figure 9. Both the proposed information-filling-driven message selection and codebook-based representation are effective.

tings by using domain adaption, while CodeFilling is naturally compatible with heterogeneous settings without additional cost. We see that CodeFilling: i) achieves a far-more superior perception-communication trade-off across all the communication bandwidth choices and various collaborative perception settings, including camera-only, lidar-only, and heterogeneous 3D detection; ii) significantly improves the detection performance, especially under extremely limited communication bandwidth, improves the SOTA performance by 11.093/5.271/38.357%, 14.75/28.516/28.372% for LiDAR/camera/heterogeneous on DAIR-V2X and OPV2VH+ even when the bandwidth is constrained by a factor of 100K; and iii) outperforms previous communication-efficient SOTA, Where2comm, with significantly reduced communication cost: 1333/115/863, 1206/1078/252 times less on DAIR-V2X and OPV2VH+.

Furthermore, for inference speed, CodeFilling (36/99ms) is comparable to Where2comm (34/94ms), and significantly faster than HMViT (90/1266ms) on DAIR-V2X/OPV2VH+. This communication efficiency ensures that agents are able to actively collaborate with each other.

### Robustness to pose error and communication latency.

We validate the robustness against pose error and communication latency on both OPV2VH+ and DAIR-V2X. The pose error setting follows CoAlign [21] using Gaussian noise with a mean of 0m and standard deviations ranging from 0m to 1.0m. The latency setting follows SyncNet [14], varying from 0ms to 500ms. Figs. 7 and 8 show the detection performances as a function of pose error and latency, respectively. We see: i) while perception performance generally declines with increasing levels of pose error and latency, CodeFilling consistently outperforms baselines under all imperfect conditions; ii) CodeFilling consis-

tently surpasses No Collaboration, whereas baselines fail when pose error exceeds 0.6m and latency surpasses 100ms.

### 5.3. Ablation studies

#### Effectiveness of our message selection and representation.

Fig. 9a compares CodeFilling, the one without message selection, and the one without codebook and message selection. We see that: i) applying code index representation reduces the communication cost by 208 times while maintaining the same detection performance, as it bypasses the high channel dimension typical of feature vectors, and an integer index requires less data than the original floating-point numbers, and ii) applying information-filling-driven message selection achieves 4.8% higher detection performance with the same communication cost, as it reallocates the bandwidth wasted in redundant information to more beneficial information.

#### Ablation of information-filling-driven message selection.

Fig. 9b compares different utility designs in information-filling optimization: sum, max, and scenarios without selection. The max utility design favors selecting collaborators with the highest score, leading to no selection if the ego agent has the highest score. We see that the sum-based utility outperforms both the max and no selection in the perception-communication trade-off across all communication bandwidth conditions. This superiority is due to the optimized combination of information from different collaborators, which has proved to be more effective than relying solely on the best-performing single agent. This approach encourages agents to participate in collaboration; even the top-performing agents can benefit from collaboration.

Fig. 9c evaluates three different information demands  $u$  (0.5, 1.0, 1.5). We see that: i) lower  $u$  demonstrates better efficiency under limited communication budgets; ii) higher

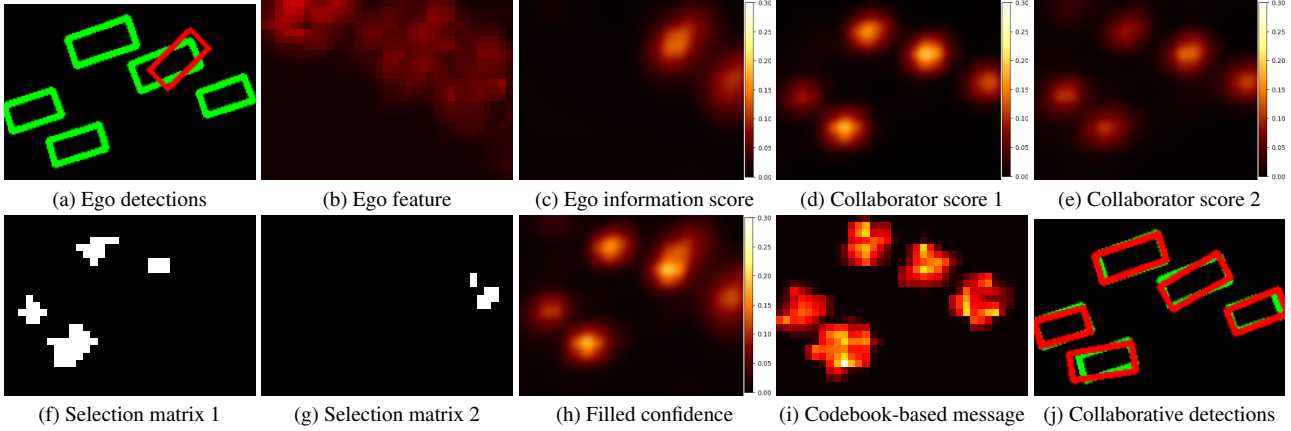


Figure 10. Visualization of collaboration in CodeFilling. Green and red denote ground truth and detection, respectively.

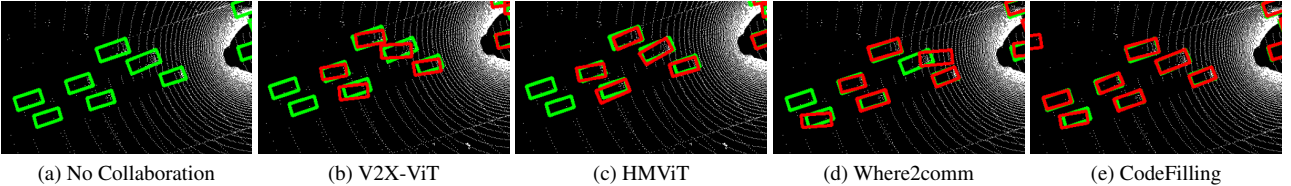


Figure 11. CodeFilling achieves more accurate detections with 1256 times less communication cost. Green and red boxes denote ground-truth and detection, respectively.

$u$  demonstrates superior performance under ample communication budgets. By adjusting  $u$ , CodeFilling consistently maintains superior performance-communication trade-off across all communication conditions.

**Ablation of codebook-based message representation.** Fig. 9d and Fig. 9e explore different codebook configurations: codebook size and code quantity. We see that: i) all the codebook configurations demonstrate a superior perception-communication trade-off under highly constrained communication conditions, showing the effectiveness and robustness of the codebook-based representation; and ii) larger codebook sizes and quantities yield better performance, while smaller sizes and quantities offer greater communication efficiency. By adapting the configuration, CodeFilling maintains superior performance-communication trade-off across all communication budgets.

#### 5.4. Qualitative evaluation

**Visualization of message selection and representation.** Fig. 10 showcases the efficient collaboration in CodeFilling. The scene features one ego agent and two collaborators - one with LiDAR and the other with a camera, with all visualizations presented from the ego’s perspective. Fig. 10 (a) and (j) compare the detection results before and after collaboration. We see that through collaboration, the ego agent successfully uncovers detections missed in its individual view. Fig. 10 (d-g) displays the collaborators’ information score maps and their corresponding selection matrices. We see that the redundant information in the overlapped regions is avoided, promoting communication efficiency. Fig. 10 (c) and (h) illustrate the evolution of the information score before and after filling, highlight-

ing that the information demands have been met, which is indicative of improved perception performance. Fig. 10 (f) presents the codebook-represented message, revealing that: i) it is spatially sparse, and ii) the selected information from heterogeneous collaborators is unified in a common format, facilitating extensibility.

**Visualization of detection results.** Fig. 11 compares CodeFilling with previous SOTAs. We see that CodeFilling qualitatively outperforms previous SOTAs with 1256 times less communication cost. The reason is that CodeFilling avoids redundant information from different collaborators and employs efficient code index representation, thereby transmitting more critical perceptual information even with less communication cost.

## 6. Conclusions

We propose CodeFilling, a novel communication-efficient collaborative 3D detection system with two novel designs: codebook-based message representation and information-filling-driven message selection. Extensive experiments covering both real-world and simulation scenarios show that CodeFilling not only achieves state-of-the-art perception-communication trade-off under various modalities, including LiDAR, camera, and heterogeneous settings but also is robust to pose error and latency issues.

**Limitation and future work.** We plan to explore the temporal dimension and determine critical time stamps.

**Acknowledgments.** This research is supported by the National Key R&D Program of China under Grant 2021ZD0112801, NSFC under Grant 62171276 and the Science and Technology Commission of Shanghai Municipal under Grant 21511100900 and 22DZ2229005.



## References

- [1] Ebtehal Turki Alotaibi, Shahad Saleh Alqefari, and Anis Koubaa. Lsar: Multi-uav collaboration for search and rescue missions. *IEEE Access*, 7:55817–55832, 2019. [1](#)
- [2] Qi Chen. F-cooper: feature based cooperative perception for autonomous vehicle edge computing system using 3d point clouds. *Proceedings of the 4th ACM/IEEE Symposium on Edge Computing*, 2019. [1](#)
- [3] Runjian Chen, Yao Mu, Runsen Xu, Wenqi Shao, Chenhan Jiang, Hang Xu, Yu Qiao, Zhenguo Li, and Ping Luo. CO3: Cooperative unsupervised 3d representation learning for autonomous driving. In *The Eleventh International Conference on Learning Representations*, 2023. [1](#)
- [4] 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](#)
- [5] Tianshi Chen, Zidong Du, Ninghui Sun, Jia Wang, Chengyong Wu, Yunji Chen, and Olivier Temam. Diannao: a small-footprint high-throughput accelerator for ubiquitous machine-learning. *Proceedings of the 19th international conference on Architectural support for programming languages and operating systems*, 2014. [3](#)
- [6] Alexey Dosovitskiy, Germán Ros, Felipe Codevilla, Antonio M. López, and Vladlen Koltun. Carla: An open urban driving simulator. In *Conference on Robot Learning*, 2017. [6](#)
- [7] Allen Gersho and Robert M. Gray. Vector quantization and signal compression. In *The Kluwer International Series in Engineering and Computer Science*, 1991. [2](#)
- [8] Allen Gersho and Bhaskar Ramamurthi. Image coding using vector quantization. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1982. [2](#)
- [9] Song Han, Huizi Mao, and William J. Dally. Deep compression: Compressing deep neural network with pruning, trained quantization and huffman coding. *Computer Vision and Pattern Recognition*, 2015. [2](#)
- [10] 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*, 35:4874–4886, 2022. [1](#), [2](#), [5](#), [6](#)
- [11] Yue Hu, Shaoheng Fang, Weidi Xie, and Siheng Chen. Aerial monocular 3d object detection. *IEEE Robotics and Automation Letters*, 8:1959–1966, 2022. [1](#)
- [12] Yue Hu, Yifan Lu, Runsheng Xu, Weidi Xie, Siheng Chen, and Yanfeng Wang. Collaboration helps camera overtake lidar in 3d detection. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. [1](#), [2](#), [5](#), [6](#)
- [13] 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, 2018. [1](#), [6](#)
- [14] Zixing Lei, Shunli Ren, Yue Hu, Wenjun Zhang, and Siheng Chen. Latency-aware collaborative perception. *ECCV*, 2022. [2](#), [7](#)
- [15] Jinlong Li, Runsheng Xu, Xinyi Liu, Baolu Li, Qin Zou, Jiaqi Ma, and Hongkai Yu. S2r-vit for multi-agent cooperative perception: Bridging the gap from simulation to reality. *ArXiv*, abs/2307.07935, 2023. [1](#)
- [16] 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. [1](#), [2](#), [6](#)
- [17] Yiming Li, Ziyang An, Zixun Wang, Yiqi Zhong, Siheng Chen, and Chen Feng. V2X-Sim: A virtual collaborative perception dataset for autonomous driving. *IEEE Robotics and Automation Letters*, 7, 2022. [2](#)
- [18] Yiming Li, Qi Fang, Jiamu Bai, Siheng Chen, Felix Juefei-Xu, and Chen Feng. Among us: Adversarially robust collaborative perception by consensus. *ICCV*, 2023. [1](#)
- [19] 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. [1](#), [2](#), [5](#)
- [20] 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. [1](#), [2](#), [5](#)
- [21] Yifan Lu, Quanhao Li, Baoan Liu, Mehrdad Dianat, Chen Feng, Siheng Chen, and Yanfeng Wang. Robust collaborative 3d object detection in presence of pose errors. *IEEE International Conference on Robotics and Automation (ICRA)*, 2023. [1](#), [2](#), [7](#)
- [22] Yifan Lu, Yue Hu, Yiqi Zhong, Dequan Wang, Siheng Chen, and Yanfeng Wang. An extensible framework for open heterogeneous collaborative perception. In *The Twelfth International Conference on Learning Representations*, 2024. [1](#)
- [23] Cody Reading, Ali Harakeh, Julia Chae, and Steven L. Waslander. Categorical depth distribution network for monocular 3d object detection. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8551–8560, 2021. [6](#)
- [24] Jürgen Scherer, Saeed Yahyanejad, Samira Hayat, Evsen Yanmaz, Torsten Andre, Asif Khan, Vladimir Vukadinovic, Christian Bettstetter, Hermann Hellwagner, and Bernhard Rinner. An autonomous multi-uav system for search and rescue. In *Proceedings of the First Workshop on Micro Aerial Vehicle Networks, Systems, and Applications for Civilian Use*, pages 33–38, 2015. [1](#)
- [25] Saurabh Singh, Sami Abu-El-Haija, Nick Johnston, Johannes Ballé, Abhinav Shrivastava, and George Toderici. End-to-end learning of compressible features. In *2020 IEEE International Conference on Image Processing (ICIP)*, pages 3349–3353. IEEE, 2020. [3](#)
- [26] Sanbao Su, Yiming Li, Sihong He, Songyang Han, Chen Feng, Caiwen Ding, and Fei Miao. Uncertainty quantification of collaborative detection for self-driving. *ICRA*, 2023. [1](#)

- [27] Nicholas Vadivelu, Mengye Ren, James Tu, Jingkang Wang, and Raquel Urtasun. Learning to communicate and correct pose errors. In *Conference on Robot Learning*, 2020. 2
- [28] Binglu Wang, Lei Zhang, Zhaozhong Wang, Yongqiang Zhao, and Tianfei Zhou. Core: Cooperative reconstruction for multi-agent perception. *ICCV*, 2023. 1
- [29] Tianhang Wang, Guang Chen, Kai Chen, Zhengfa Liu, Bo Zhang, Alois Knoll, and Changjun Jiang. Umc: A unified bandwidth-efficient and multi-resolution based collaborative perception framework. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1
- [30] 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. 1, 2, 6
- [31] Sizhe Wei, Yuxi Wei, Yue Hu, Yifan Lu, Yiqi Zhong, Siheng Chen, and Ya Zhang. Asynchrony-robust collaborative perception via bird’s eye view flow. In *Advances in Neural Information Processing Systems*, 2023. 1, 2
- [32] Hao Xiang, Runsheng Xu, Xin Xia, Zhaoliang Zheng, Bolei Zhou, and Jiaqi Ma. V2xp-asg: Generating adversarial scenes for vehicle-to-everything perception. *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3584–3591, 2022. 2
- [33] Hao Xiang, Runsheng Xu, and Jiaqi Ma. Hm-vit: Heteromodal vehicle-to-vehicle cooperative perception with vision transformer. *ICCV*, 2023. 1, 6
- [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. *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, pages 1155–1162, 2021. 6
- [35] Runsheng Xu, Hao Xiang, Xin Xia, Xu Han, Jinlong Liu, and Jiaqi Ma. Opv2v: An open benchmark dataset and fusion pipeline for perception with vehicle-to-vehicle communication. *2022 International Conference on Robotics and Automation (ICRA)*, pages 2583–2589, 2021. 1, 2, 6
- [36] 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
- [37] Runsheng Xu, Hao Xiang, Zhengzhong Tu, Xin Xia, Ming-Hsuan Yang, and Jiaqi Ma. V2X-ViT: Vehicle-to-everything cooperative perception with vision transformer. *ECCV*, 2022. 1, 2, 6
- [38] Runsheng Xu, Jinlong Li, Xiaoyu Dong, Hongkai Yu, and Jiaqi Ma. Bridging the domain gap for multi-agent perception. *ICRA*, 2023. 1
- [39] Runsheng Xu, Xin Xia, Jinlong Li, Hanzhao Li, Shuo Zhang, Zhengzhong Tu, Zonglin Meng, Hao Xiang, Xiaoyu Dong, Rui Song, Hongkai Yu, Bolei Zhou, and Jiaqi Ma. V2v4real: A real-world large-scale dataset for vehicle-to-vehicle cooperative perception. In *The IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*, 2023. 1
- [40] Dingkang Yang, Kun Yang, Yuzheng Wang, Jing Liu, Zhi Xu, Rongbin Yin, Peng Zhai, and Lihua Zhang. How2comm: Communication-efficient and collaboration-pragmatic multi-agent perception. *Advances in Neural Information Processing Systems*, 2023. 1
- [41] Kun Yang, Dingkang Yang, Jingyu Zhang, Mingcheng Li, Y. Liu, Jing Liu, Hanqi Wang, Peng Sun, and Liang Song. Spatio-temporal domain awareness for multi-agent collaborative perception. *Proceedings of the 31st ACM International Conference on Multimedia*, 2023. 1
- [42] Kun Yang, Dingkang Yang, Jingyu Zhang, Hanqi Wang, Peng Sun, and Liang Song. What2comm: Towards communication-efficient collaborative perception via feature decoupling. *Proceedings of the 31st ACM International Conference on Multimedia*, 2023. 1
- [43] Haibao Yu, Yizhen Luo, Mao Shu, Yiyi Huo, Zebang Yang, Yifeng Shi, Zhenglong Guo, Hanyu Li, Xing Hu, Jirui Yuan, et al. DAIR-V2X: A large-scale dataset for vehicle-infrastructure cooperative 3d object detection. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition (CVPR)*, 2022. 1, 2, 6
- [44] Haibao Yu, Wenxian Yang, Hongzhi Ruan, Zhenwei Yang, Yingjuan Tang, Xu Gao, Xin Hao, Yifeng Shi, Yifeng Pan, Ning Sun, Juan Song, Jirui Yuan, Ping Luo, and Zaiqing Nie. V2x-seq: A large-scale sequential dataset for vehicle-infrastructure cooperative perception and forecasting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. 2
- [45] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. In *arXiv preprint arXiv:1904.07850*, 2019. 1, 6