University of Porto

Faculty of Sciences

CroVe: Cross-Vehicle Collaboration Using RGB Cameras

João Pedro Coelho Monteiro

Candidate: João Pedro Coelho Monteiro, up202108347@fc.up.pt

Scientific Guidance: Ricardo Cruz, rpcruz@fe.up.pt

Scientific Co-Guidance: Celso Pereira, up202200546@fe.up.pt

Company: INESC-TEC

AIDC

Bachelor's in Artificial Intellingence and Data Science

Faculty of Sciences
INESC-TEC

July 3, 2024

Acknowledgements

I would like to express my gratitude to my advisor, Professor Ricardo Cruz, for his guidance, advice and support throughout this project.

I am also thankful to my co-advisor, Celso Pereira, whose expertise and insightful feedback were crucial in shaping this work.

This work would not have been possible without the support and mentorship of both. Their commitment has been inspiring and i am grateful for the opportunities and knowledge they have provided.

Thank you.

This work was supported by Portuguese National Funds, through AICEP, E. P. E., and the Innovation and Digital Transition Program (COMPETE2030). Project ATLAS – Trusted Autonomous Navigation, with Funding Reference 01/RPA/2022-C679908640-00009887.

Abstract

CroVe provides a cross-vehicle collaboration framework using RGB cameras to improve 3D object detection, integrating data from two vehicles to enhance detection accuracy. Two fusion methods are explored: late fusion, which combines detections from individual cameras, and intermediate fusion, which merges feature maps before making final predictions.

Evaluated on the OPV2V dataset, the study demonstrates that collaborative detection outperforms single-vehicle detection. Intermediate fusion achieves the highest Intersection over Union (IoU) with 10.10% opposed to the 9.83% from late fusion and 7.34% from no collaboration.

The code is available at https://github.com/joaopecomonteiro/CroVe.

Keywords: autonomous driving, collaborative perception, semantic segmentation, occupancy estimation

Contents

Chapter 1

Introduction

Collaborative perception [1, 2], also known as cross-vehicle communication, refers to the capability of vehicles to detect and communicate with other vehicles on the road. This involves exchanging information between vehicles in real time (sensor data, velocity, GPS positions, and other data) to collaborate on detecting objects on or near the road and other tasks. There is much focus in the literature on LiDAR information being exchange, this might be because: (i) LiDAR is a main sensor in autonomous driving due to its properties in 3D and night vision; (ii) it is easy to concatenate LiDAR point-cloud information from different sources since each point can be transformed onto a common referential, making the concatenation of points from different point-clouds effortless. However, scaling up systems that rely on LiDAR sensors tend to be very expensive. Another approach is using much cheaper sensors to privilege cost effectiveness over, usually, robustness. Some manufactures, most famously Tesla, insist on using only RGB cameras to simplify the pipeline because human drivers are proof that RGB cameras already capture all information required for autonomous driving – furthermore, color information is important for sign recognition.

Independently of the type of sensor being used, there are three main forms of merging data from different sources: (i) early fusion: where the input is simply concatenated after being transformed onto a common representation, and only then is the input processed by the model; (ii) intermediate or middle fusion: the input is partially processed by the model and the concatenation happens inside of the model

(nowadays, transformers are typically used for this purpose [3]); (iii) late fusion: each model handles each input separately and the decisions are merged in the end using an average or majority voting.

In this work I explored the possibilities of expanding an architecture designed for single vehicle semantic occupancy grid prediction [4] to a collaborative perception paradigm. Occupancy grid maps [5] are extensively utilized in robotics, as they allow easy integration of data captured by various sensors and across different time frames. This grid-based format is particularly suitable for processing by convolutional neural networks, allowing me to leverage advancements in deep learning and, more precisely, computer vision. When it comes to the merging of data, I analyzed both intermediate and late fusion.

To sum up, my contributions are:

- I propose a new collaborative camera-only segmentation framework CroVe, which improves the detection abilities of cameras with multi-agent collaboration.
- I conducted experiments validating: (i) the necessity of multi-agent collaboration in autonomous driving tasks; (ii) the cost effectiveness of RGB cameras in semantic segmentation tasks.

Chapter 2

Related Work

2.1 Autonomous driving datasets

Dataset	Year	Environment	V2X	RGB Images	Objects	Classes	Locations
Kitti	2012	Real	No	15k	200k	8	Karlsruhe
nuScenes	2019	Real	No	1.4M	1.4M	23	Boston, SG
Argoverse	2019	Real	No	107k	993k	15	2x USA
Waymo Open	2019	Real	No	1M	12M	4	3x USA
OPV2V	2022	Sim	V2V	44k	230k	1	CARLA
V2X-Sim	2022	Sim	V2V + V2I	60K	26.6k	1	CARLA
V2XSet	2022	Sim	V2V + V2I	44K	230k	1	CARLA
DAIR-V2X	2022	Real	V2I	39K	464K	10	Beijing, CN
DOLPHINS	2022	Sim	V2V + V2I	42k	293k	3	CARLA
V2V4Real	2023	Real	V2V	40K	240K	5	Ohio, USA
Rcooper	2024	Real	V2I	50k	158k	10	Beijing, CN

Table 2.1: Comparison between existing autonomous driving datasets. Datasets: Kitti [6], nuScenes [7], Argoverse [8], Waymo Open [9], OPV2V [10], V2X-Sim [11], V2XSet [3], DAIR-V2X [12], DOLPHINS [13], V2V4Real [14], Rcooper [15]

Large open datasets are essential in deep learning. Even though numerous well-established autonomous driving datasets exist, including KITTI [6], nuScenes [7] and Argoverse [8], they prmarily address individual perception and fail to fulfill the demand for collaborative perception. Thankfully, recent advances in artificial intelligence and in collaborative perception benchmarks [10, 11, 12, 14, 15] have

contributed to the acceleration of the development of autonomous driving datasets focused on multi-agent communication. These datasets can: (i) be either real, meaning the data was collected in the real world or simulated, meaning the data was collected in some kind of simulated environment, normally using CARLA; (ii) have no interaction between vehicle, vehicle-to-vehicle (V2V) interaction, or vehicle-to-infrastructure (V2I) interaction, or both V2V and V2I interactions (see Table 2.1).

2.2 Collaborative Perception Methods

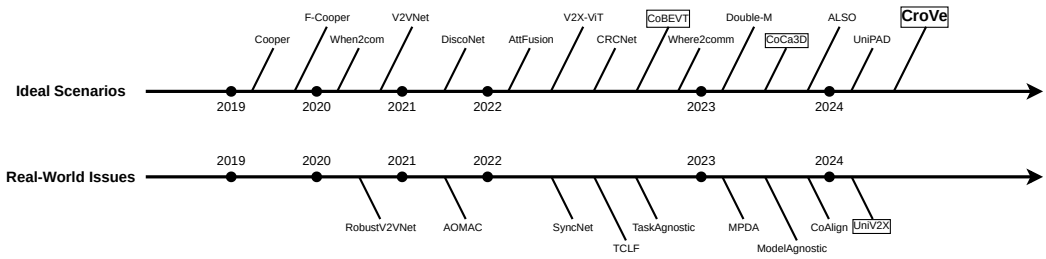


Figure 2.1: Some collaborative perception methods in autonomous driving divided into two categories: the ones that focus collaborative performance and efficiency, and the ones that focus on robustness and safety. The highlighted methods only use RGB cameras. Methods: Cooper [16], F-Cooper [17], When2com [18], V2VNet [19], DiscoNet [20], AttFusion [10], V2X-ViT [3], CRCNet [21], CoBEVT [22], Where2comm [23], Double-M [24], CoCa3D [25], RobustV2VNet [26], AOMAC [27], SyncNet [28], TCLF [12], TaskAgnostic [29], MPDA [30], ModelAgnostic [31], CoAlign [32], ALSO [33], UniPAD [34], UniV2X [35], CroVe

Collaborative perception has been a topic of interest in multiple fields for quite some time. However, the lack of large public datasets has hindered the advancement of collaborative perception. In recent years, with the surge of new datasets and the development of deep learning, there has been a crescent of work being done in this field (see Figure 2.1, which was adapted from [2]). Most of the research focuses on LiDAR sensors, only more recently there has been real developments on collaborative perception using other sensors like RGB cameras. With this in mind, this work aims to expand the options for cross-vehicle collaboration using RGB cameras by introducing another tested architecture.

2.3 Object detection and segmentation

RGB Image Camera-only collaborative perception involves identifying objects within a 3D space using 2D images captured by either vehicles on the road or roadside infrastructure. The most common form of environment representation is

birds's-eye-view (BEV). This happens because of their computation efficiency and the ease of using modern deep learning tools with them.

Previous studies [22, 25], similar to CroVe, aim to enhance overall performance by extending available architectures and frameworks into the collaborative perception domain. This means finding the best representation of the environment as well as the best form of communication between agents. In CoCa3D [25] there is two moments of communication between agents, one in depth estimation and other on detection feature learning. CroVe tries to achieve similar results but with only one communication moment between the agents, this will be discussed further ahead.

LiDAR LiDAR-based 3D detection performs exceptionally well because it provides accurate 3D measurements of the input data. The two most common ways to encode LiDAR points are voxel-based [36, 37, 38] and point-based [39, 40, 41]. Voxel-based methods partition the 3D space into uniform voxels [38] or pillars [36], then convert the points within each section into feature representations. Point-based methods are usually based on PointNet [42] series to aggregate the features of points. LiDAR-based 3D detections achieves great results, but high-quality LiDAR sensor are very expensive difficulting large scale approaches.

Chapter 3

Cross-Vehicle Collaboration Using RGB Cameras

3.1 Problem formulation

Consider N agents in the scene, let $\mathcal{X}_i$ be the RGB image collected by the ith camera and $\mathcal{O}_i^0$ be the ground truth detection for that scene. The objective of CroVe is to maximize detection performances of the agents, that is:

$$\max_{\theta,\mathcal{I}} \sum_i g(\Phi_\theta(\mathcal{X}_i, \mathcal{I}_{i\to ego}), \mathcal{O}_i^0), \tag{3.1}$$

where $g(\cdot,\cdot)$ is the detection evaluation metric, $\Phi_\theta(\cdot)$ is a detection model with trainable parameter θ, and $\mathcal{I}_{i\to ego}$ is the message containing the information captured by the ith camera to the ego. The design rational comes from some aspects: First, the assumption that the messaging between both agents occurs without any problems (information loss, lag or other problems). This would allow for a much easier transmission of information during and after the training. Second, the assumption that each camera can detect the objects that surround the vehicle with some confidence. This was a fair assumption because of the results obtained from training the single agent model. Additionally, this would remove the need for communication during the depth estimation phase, decreasing the number of needed messages to one, making this architecture simpler, and potentially faster.

3.2 Single agent

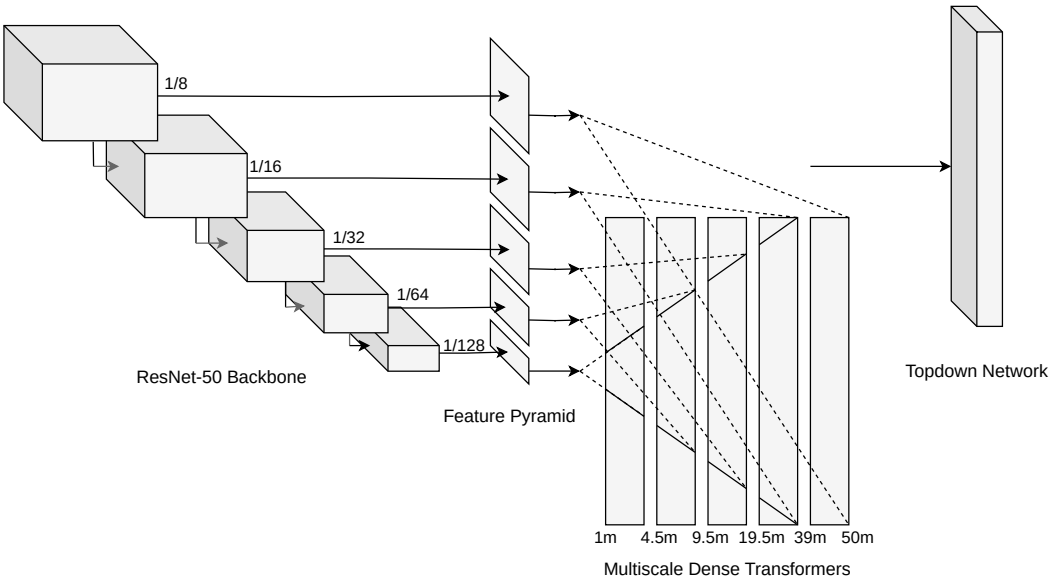


Figure 3.1: Single agent architecture

The single-agent detector follows the Pyramid Occupancy Networks [4] (shown in Figure 3.1). This framework consists of four main stages. First, a backbone feature extractor generates multi-scale semantic and geometric features from the image. This is then passed to an FPN [43]-inspired feature pyramid which upsamples low-resolution feature-maps to provide context to features at higher resolutions. A stack of dense transformer layers together map the image-based features into BEV, which are processed by the topdow network to obtain the final features. At the end, for single vehicle detection, those features pass through a classifier in order to obtain the final semantic occupancy grid probabilities.

3.3 Agent collaboration

For the fusion between the collaborative vehicles' data, two options were explored, late and intermediate.

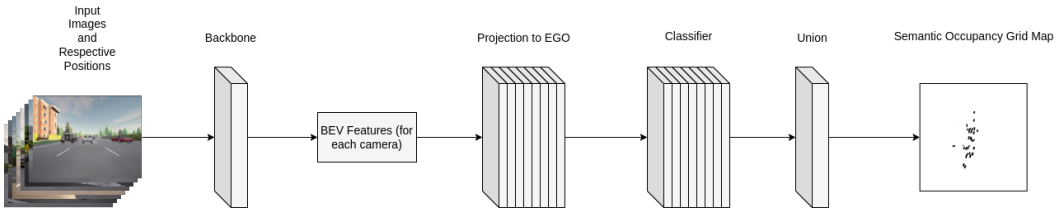


Figure 3.2: Late fusion

Late Fusion The late fusion architecture only required a transformation that went from the camera's coordinates to the ego's coordinates. The scene's prediction ($\mathcal{Y}$) would be the union of the true coordinates (points in the real world that contain a vehicle) captured by each camera, that is:

$$\mathcal{Y} = \bigcup_{x=0}^{i} \Phi_\theta(\mathcal{X}_i, \mathcal{P}_{i \to ego}),$$

where $\mathcal{X}_i$ is the ith camera, $\Phi_\theta(\cdot)$ is the detection model trainable with parameter θ, and $\mathcal{P}_{i \to ego}$ is the message containing the prediction of the ith camera to the ego (see Figure 3.2). This approach has an evident issue when two cameras have two different predictions for the same car, causing the scene occupancy to have some shapes that were visibly bigger than they were supposed to be. This happens because the union doesn't take into consideration that a specific car might appear in a slightly different position in other's camera estimation. The bigger the number of collaborative vehicles, the more noticeable this error is.

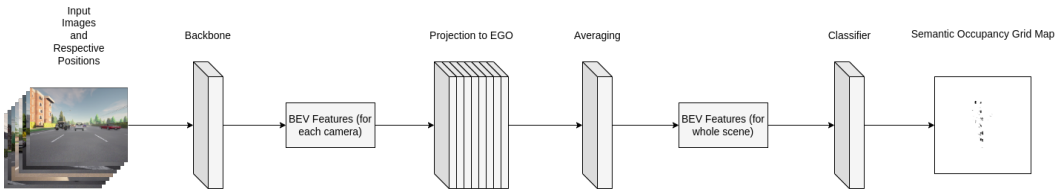


Figure 3.3: Intermediate fusion

Intermediate Fusion Following the same logic as the previous fusion method, the merging would revolve around transforming each point in space from each camera's coordinates to the ego's coordinates. The value for each point was then added to a final features matrix. This would happen after the topdown network from the single agent architecture. After every camera was processed, each point in the scene's features matrix was divided by the number of cameras that could see that specific point.

$$\mathcal{Y} = \sum_{i=0}^{n} \mathcal{F}_i \odot \mathcal{M},$$

where $\sum_{i=0}^{n} F_i$ is the scene's features matrix and $\mathcal{M} = \frac{1}{\sum_i m_i}$ is a matrix of the sum of each ith camera's matrix m_i where each element is 1 or 0 depending if the that point in space is visible by camera i or not (see Figure 3.3). In theory, this approach would solve the problem that late fusion has, because the model would learn the small errors between different cameras.

3.4 Losses

Similarly to [4], both the single agent and collaborative networks were trained using a combination of two loss functions. The binary cross entropy loss encourages the predicted semantic probabilities $p(m_i^c|z_t)$ to match the ground truth occupancies $\hat{m}_i^c$. Given that the cars may only occupy a small portion of the scene, a balanced variant of this loss was used which up-weights occupied cells belonging to the vehicle class (c) by a constant factor α^c:

$$\mathcal{L}_{\text{xent}} = \alpha^c \hat{m}_i^c \log p(m_i^c|z_t) + (1 - \alpha^c) \log(1 - p(m_i^c|z_t))$$

Neural networks are known for frequently predicting high probabilities, even when they are very uncertain about the outcome. To encourage the networks to predict high uncertainty in regions which are known to be ambiguous, a second loss was introduces, which maximises the entropy of the predictions, encouraging them to fall close to 0.5:

$$\mathcal{L}_{\text{uncert}} = 1 - p(m_i^c|z_t) \log_2 p(m_i^c|z_t)$$

This maximum entropy loss is applied only to grid cells which are no visible to the network because they fall outside the field of view of every image. Cross entropy loss is ignored in these regions. The overall loss is given by the sum of the two loss functions:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{xent}} + \lambda \mathcal{L}_{\text{uncert}}$$

where λ is a constant weighting factor.

Chapter 4

Experimental Setup and Results

4.1 Dataset

Due to its simplicity and availability, the experiments were conducted using the OPV2V [10] dataset. OPV2V's data was collected in CARLA and it focus on V2V interaction. It has around 44,000 images and around 230,000 annotated objects that are only vehicles. Each split is divided in scenes that can contain between 2 and 7 collaborative vehicles. Unfortunately, because of GPU limitations, only 2 of these collaborative vehicles were considered. With this in mind, the first vehicle of each scene was chosen to be the ego, the main vehicle that receives the data from the other collaborative vehicle, and the second vehicle of each scene was chosen to be the helping vehicle, if it was under 50m away from the ego, otherwise, only the ego was considered. Despite that, the pipeline works with any number of collaborative agents and with batches with scenes that have a different number of collaborative agents.

4.2 Data Generation

The OPV2V dataset provide ground truth annotations in the form of 3D bounding boxes. The bounding boxes of the vehicles that were under 51m away from any collaborative vehicle were converted into a semantic occupancy map that has the ego vehicle in its center. Since part of the resulting labels lie outside of the field

of view of every camera, a binary mask was generated indicating whether each grid cell is within the field of view of any camera or not.

The grid size for a single camera was 50m forward and 25m to each side, with a resolution of 25cm per pixel, and the grid size for the whole scene was 100m in every direction with the same resolution.

4.3 Architecture and training details

The backbone of both intermediate and late fusion is similar to [4]. A pretrained FPN network [43], incorporating a ResNet-50 [44] frontend, was used for the feature pyramid components. The topdown network consists of a stack of 8 residual blocks, including a transposed convolutional layer which upsamples the birds-eye-view features from a resolution of 0.5m to 0.25m per pixel.

A single camera model was trained using the backbone explained previously. This model was useful for for: (i) Providing the single camera predictions in the late fusion paradigm; (ii) Starting the intermediate fusion training with a pretrained backbone, so the training time was reduced and the performance was better.

The balanced loss weight α^c was 5.2 in single camera training and in intermediate fusion training. The uncertainty loss δ was 0.0001. All networks were trained until convergence using SGD with a learning rate of 0.1 (learning rates of 0.01, 0.001 and 0.0001 were also tested), batch size of 12 for single camera training and 2 for intermediate fusion training and a momentum of 0.9.

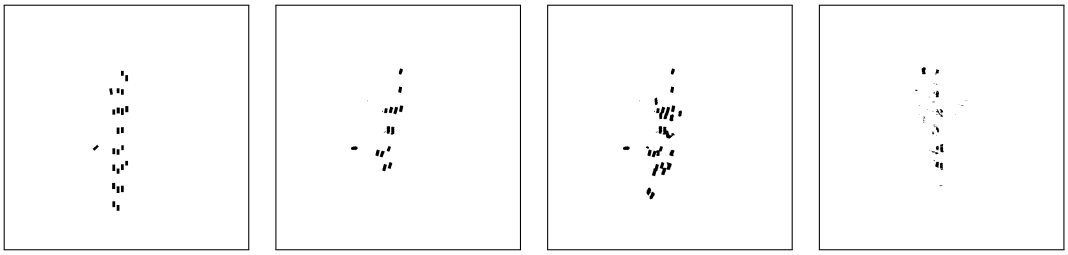
4.4 Evaluation

The evaluation metric used was the Intersection over Union (IoU) score. To compute this, the predictions were converted to binary form based on a Bayesian decision threshold, where predictions with a probability $p(m_i^c|z_t)$ greater than 0.5 were considered positive.

4.5 Results

No Collaboration	Late Fusion	Intermediate Fusion
7.34	9.83	10.10

Table 4.1: Intersection over Union scores (%)



(a) Ground Truth (b) No Collaboration (c) Late (d) Intermediate

Figure 4.1: Example of results

The baseline of the experiments was the performance of a single vehicle, the ego. The performance of late and intermediate fusion were also evaluated. The results are showed in Table 4.1.

Based on the IoU scores and the result images (Figure 4.1), several conclusions can be drawn. Firstly, a clear benefit is observed when vehicles collaborate (over 2% mean IoU). This is easily explained by the fact that vehicles outside the ego's field of view can be seen by the collaborative vehicle and therefore, detected by the model.

Late Fusion As expected, there is an overlap in the collaborative vehicles' predictions. This results in the appearance of more cars in the merged prediction that there are in reality. A way this could be solved is training a model specifically for late fusion. This did not happen as the late fusion model is a combination of all collaborative vehicles individual predictions. It is also visible that, although it can detect the cars on the road, it has some trouble predicting their exact position and angle.

Intermediate Fusion Even though this approach had the best performance, it still has a lot of room for improvement. For instance, the loss weight for the vehicle class, mentioned in chapter 4, was probably too low. There was a considerate difference in performance between training and testing, indicating some overfitting. This might have happened because of the lack of augmentation and only using one vehicle as the ego in each scene. Restricting the dataset to just two collaborative vehicles adversely affected the performance of this fusion method.

Looking at the results of state of the art works [10, 22], it becomes clear that the results obtained are not as bad as they seem at first sight. The performance of tasks like CroVe's greatly increase as the number of collaborative agents increase. Given the necessity to reduce the number of agents, the results are understandable.

Chapter 5

Conclusion

I propose CroVe, a cross-vehicle collaboration framework utilizing RGB cameras to achieve comprehensive 3D object detection. By integrating insights from previous studies and extending existing architectures, CroVe aims to enhance camera-based detection capabilities through the collective efforts of multiple vehicles. Experiments conducted on the OPV2V dataset demonstrate the significant potential of this approach. As research in this field continues to evolve, further refinements in model architectures and training methodologies are expected to drive ongoing advancements in collaborative perception for autonomous systems.

5.1 Limitations and Future Work

3D object detection and segmentation using cameras are both highly resource-intensive tasks. In the current setup, the GPU used for training could only accommodate two collaborative agents due to its limited capacity. Looking ahead, it would be advantageous to scale up the number of agents involved in the training process. This could be achieved by employing a multi-GPU approach, which would distribute the computational load more effectively. Alternatively, optimizing the pipeline to enhance data storage efficiency could also enable the training of more agents. These improvements would not only accelerate the training process but also potentially enhance the performance and accuracy of the detection and segmentation tasks.

This work leverages simulation data to validate the proposed method, providing an initial proof of concept. To further demonstrate its effectiveness and robustness, it would be essential to test the method with real-world datasets. Such testing would offer critical insights into its practical applicability and performance in real-world scenarios, ensuring that the method can handle the complexities and variabilities of actual environments.

An intriguing enhancement would be to utilize OPV2V's map data to predict road layouts by developing a dedicated class specifically for this purpose. This addition could improve the accuracy of the model in recognizing and navigating road structures, ultimately enhancing its overall performance in real-world scenarios.

References

[1] S. Malik, M. J. Khan, M. A. Khan, and H. El-Sayed, "Collaborative Perception—The Missing Piece in Realizing Fully Autonomous Driving," *Sensors*, vol. 23, no. 18, p. 7854, 2023.

[2] Y. Han, H. Zhang, H. Li, Y. Jin, C. Lang, and Y. Li, "Collaborative Perception in Autonomous Driving: Methods, Datasets, and Challenges," *IEEE Intelligent Transportation Systems Magazine*, 2023.

[3] R. Xu, H. Xiang, Z. Tu, X. Xia, M.-H. Yang, and J. Ma, "V2X-ViT: Vehicle-to-Everything Cooperative Perception with Vision Transformer," in *European conference on computer vision*, pp. 107–124, Springer, 2022.

[4] T. Roddick and R. Cipolla, "Predicting Semantic Map Representations From Images Using Pyramid Occupancy Networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11138–11147, 2020.

[5] A. Elfes, "Occupancy Grids: A Stochastic Spatial Representation for Active Robot Perception," *arXiv preprint arXiv:1304.1098*, 2013.

[6] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *2012 IEEE conference on computer vision and pattern recognition*, pp. 3354–3361, IEEE, 2012.

[7] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuScenes: A Multimodal Dataset for Autonomous Driving," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11621–11631, 2020.

[8] M.-F. Chang, J. Lambert, P. Sangkloy, J. Singh, S. Bak, A. Hartnett, D. Wang, P. Carr, S. Lucey, D. Ramanan, *et al.*, "Argoverse: 3D Tracking and Forecasting With Rich Maps," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8748–8757, 2019.

[9] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine, *et al.*, " Scalability in Perception for Autonomous Driving: Waymo Open Dataset," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2446–2454, 2020.

[10] R. Xu, H. Xiang, X. Xia, X. Han, J. Li, and J. Ma, "OPV2V: An Open Benchmark Dataset and Fusion Pipeline for Perception with Vehicle-to-Vehicle Communication," in *2022 International Conference on Robotics and Automation (ICRA)*, pp. 2583–2589, IEEE, 2022.

[11] Y. Li, D. Ma, Z. An, Z. Wang, Y. Zhong, S. Chen, and C. Feng, "V2X-Sim: Multi-Agent Collaborative Perception Dataset and Benchmark for Autonomous Driving," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 10914–10921, 2022.

[12] H. Yu, Y. Luo, M. Shu, Y. Huo, Z. Yang, Y. Shi, Z. Guo, H. Li, X. Hu, J. Yuan, *et al.*, "DAIR-V2X: A Large-Scale Dataset for Vehicle-Infrastructure Cooperative 3D Object Detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21361–21370, 2022.

[13] R. Mao, J. Guo, Y. Jia, Y. Sun, S. Zhou, and Z. Niu, "DOLPHINS: Dataset for Collaborative Perception enabled Harmonious and Interconnected Self-driving," in *Proceedings of the Asian Conference on Computer Vision*, pp. 4361–4377, 2022.

[14] R. Xu, X. Xia, J. Li, H. Li, S. Zhang, Z. Tu, Z. Meng, H. Xiang, X. Dong, R. Song, *et al.*, "V2V4Real: A Real-World Large-Scale Dataset for Vehicle-to-Vehicle Cooperative Perception ," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13712–13722, 2023.

[15] R. Hao, S. Fan, Y. Dai, Z. Zhang, C. Li, Y. Wang, H. Yu, W. Yang, J. Yuan, and Z. Nie, "RCooper: A Real-world Large-scale Dataset for Roadside Cooperative Perception ," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22347–22357, 2024.

[16] Q. Chen, S. Tang, Q. Yang, and S. Fu, "Cooper: Cooperative Perception for Connected Autonomous Vehicles Based on 3D Point Clouds," in *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, pp. 514–524, IEEE, 2019.

[17] Q. Chen, X. Ma, S. Tang, J. Guo, Q. Yang, and S. Fu, "F-Cooper: Feature based Cooperative Perception for Autonomous Vehicle Edge Computing System Using 3D Point Clouds," in *Proceedings of the 4th ACM/IEEE Symposium on Edge Computing*, pp. 88–100, 2019.

[18] Y.-C. Liu, J. Tian, N. Glaser, and Z. Kira, "When2com: Multi-Agent Perception via Communication Graph Grouping," in *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pp. 4106–4115, 2020.

[19] T.-H. Wang, S. Manivasagam, M. Liang, B. Yang, W. Zeng, and R. Urtasun, "V2VNet: Vehicle-to-Vehicle Communication for Joint Perception and Prediction," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pp. 605–621, Springer, 2020.

[20] Y. Li, S. Ren, P. Wu, S. Chen, C. Feng, and W. Zhang, "Learning Distilled Collaboration Graph for Multi-Agent Perception," *Advances in Neural Information Processing Systems*, vol. 34, pp. 29541–29552, 2021.

[21] G. Luo, H. Zhang, Q. Yuan, and J. Li, "Complementarity-Enhanced and Redundancy-Minimized Collaboration Network for Multi-agent Perception," in *Proceedings of the 30th ACM International Conference on Multimedia*, pp. 3578–3586, 2022.

[22] R. Xu, Z. Tu, H. Xiang, W. Shao, B. Zhou, and J. Ma, "CoBEVT: Cooperative Bird's Eye View Semantic Segmentation with Sparse Transformers," *arXiv preprint arXiv:2207.02202*, 2022.

[23] Y. Hu, S. Fang, Z. Lei, Y. Zhong, and S. Chen, "Where2comm: Communication-Efficient Collaborative Perception via Spatial Confidence Maps," *Advances in neural information processing systems*, vol. 35, pp. 4874–4886, 2022.

[24] S. Su, Y. Li, S. He, S. Han, C. Feng, C. Ding, and F. Miao, "Uncertainty Quantification of Collaborative Detection for Self-Driving," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5588–5594, IEEE, 2023.

[25] Y. Hu, Y. Lu, R. Xu, W. Xie, S. Chen, and Y. Wang, "Collaboration Helps Camera Overtake LiDAR in 3D Detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9243–9252, 2023.

[26] N. Vadivelu, M. Ren, J. Tu, J. Wang, and R. Urtasun, "Learning to Communicate and Correct Pose Errors," in *Conference on Robot Learning*, pp. 1195–1210, PMLR, 2021.

[27] J. Tu, T. Wang, J. Wang, S. Manivasagam, M. Ren, and R. Urtasun, "Adversarial Attacks on Multi-Agent Communication," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7768–7777, 2021.

[28] Z. Lei, S. Ren, Y. Hu, W. Zhang, and S. Chen, "Latency-Aware Collaborative Perception," in *European Conference on Computer Vision*, pp. 316–332, Springer, 2022.

[29] Y. Li, J. Zhang, D. Ma, Y. Wang, and C. Feng, "Multi-Robot Scene Completion: Towards Task-Agnostic Collaborative Perception," in *Conference on Robot Learning*, pp. 2062–2072, PMLR, 2023.

[30] R. Xu, J. Li, X. Dong, H. Yu, and J. Ma, "Bridging the Domain Gap for Multi-Agent Perception," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6035–6042, IEEE, 2023.

[31] R. Xu, W. Chen, H. Xiang, X. Xia, L. Liu, and J. Ma, "Model-Agnostic Multi-Agent Perception Framework," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1471–1478, IEEE, 2023.

[32] Y. Lu, Q. Li, B. Liu, M. Dianati, C. Feng, S. Chen, and Y. Wang, "Robust Collaborative 3D Object Detection in Presence of Pose Errors," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4812–4818, IEEE, 2023.

[33] A. Boulch, C. Sautier, B. Michele, G. Puy, and R. Marlet, "ALSO: Automotive Lidar Self-Supervision by Occupancy Estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13455–13465, 2023.

[34] H. Yang, S. Zhang, D. Huang, X. Wu, H. Zhu, T. He, S. Tang, H. Zhao, Q. Qiu, B. Lin, *et al.*, "UniPAD: A Universal Pre-training Paradigm for Autonomous Driving," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15238–15250, 2024.

[35] H. Yu, W. Yang, J. Zhong, Z. Yang, S. Fan, P. Luo, and Z. Nie, "End-to-End Autonomous Driving through V2X Cooperation," *arXiv preprint arXiv:2404.00717*, 2024.

[36] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "PointPillars: Fast Encoders for Object Detection From Point Clouds," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12697–12705, 2019.

[37] W. Zheng, W. Tang, S. Chen, L. Jiang, and C.-W. Fu, "CIA-SSD: Confident IoU-Aware Single-Stage Object Detector From Point Cloud," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, pp. 3555–3562, 2021.

[38] Y. Zhou and O. 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*, pp. 4490–4499, 2018.

[39] S. Shi, C. Guo, L. Jiang, Z. Wang, J. Shi, X. Wang, and H. 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*, pp. 10529–10538, 2020.

[40] S. Shi, L. Jiang, J. Deng, Z. Wang, C. Guo, J. Shi, X. Wang, and H. Li, "Pv-rcnn++: Point-voxel feature set abstraction with local vector representation for 3d object detection," *International Journal of Computer Vision*, vol. 131, no. 2, pp. 531–551, 2023.

[41] Z. Yang, Y. Sun, S. Liu, and J. Jia, "3DSSD: Point-Based 3D Single Stage Object Detector," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11040–11048, 2020.

[42] C. R. Qi, H. Su, K. Mo, and L. 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*, pp. 652–660, 2017.

[43] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature Pyramid Networks for Object Detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2117–2125, 2017.

[44] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.