<table>
<tr><td><b>1. Report No.</b></td><td colspan="3"><b>2. Government Accession No.</b></td><td colspan="3"><b>3. Recipient’s Catalog No.</b></td></tr>
<tr><td colspan="4" rowspan="2"><b>4. Title and Subtitle</b>
Connected Vision for Increased Pedestrian Safety (CVIPS)</td><td colspan="3"><b>5. Report Date</b>
July 31, 2024</td></tr>
<tr><td colspan="3"><b>6. Performing Organization Code</b></td></tr>
<tr><td colspan="4"><b>7. Author(s)</b>
Vijayakumar Bhagavatula, https://orcid.org/ 0000-0001-7126-6381
Aswin Sankaranarayanan, https://orcid.org/ 0000-0003-0906-4046
Dereje Shenkut</td><td colspan="3"><b>8. Performing Organization Report No.</b></td></tr>
<tr><td colspan="4" rowspan="2"><b>9. Performing Organization Name and Address</b>
Carnegie Mellon University, 5000Forbes Avenue, Pittsburgh, PA 15213, USA.</td><td colspan="3"><b>10. Work Unit No.</b></td></tr>
<tr><td colspan="3"><b>11. Contract or Grant No.</b>
Federal Grant No. 69A3552344811</td></tr>
<tr><td colspan="4" rowspan="2"><b>12. Sponsoring Agency Name and Address</b>
Safety21 University Transportation Center
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213</td><td colspan="3"><b>13. Type of Report and Period Covered</b>
Final Report (July 1, 2023-June 30, 2024)</td></tr>
<tr><td colspan="3"><b>14. Sponsoring Agency Code</b>
USDOT</td></tr>
<tr><td colspan="7"><b>15. Supplementary Notes</b></td></tr>
<tr><td colspan="7"><b>16. Abstract</b>
Pedestrian safety has become a critical concern due to the higher risk of serious injury they are prone to in traffic accidents. To improve traffic safety, one promising approach is to advance the level of autonomy in vehicles. This report presents the results of a preliminary investigation of the limitations of single agent-based perception systems for pedestrian detection and the potential of collaborative or cooperative perception to further improve pedestrian detection, and the impact of communication on collaborative perception.</td></tr>
<tr><td colspan="3"><b>17. Key Words</b>
Pedestrian Detectors, Pedestrian Safety, Pedestrians</td><td colspan="4"><b>18. Distribution Statement</b></td></tr>
<tr><td colspan="2"><b>19. Security Classif. (of this report)</b></td><td colspan="3"><b>20. Security Classif. (of this page)</b></td><td><b>21. No. of Pages</b>
46</td><td><b>22. Price</b></td></tr>
</table>

Safety21
INNOVATING SAFETY FOR ALL

US DOT National
University Transportation Center for Safety

Carnegie Mellon University

Penn University of Pennsylvania

The Ohio State University

CCAC

Community College of Philadelphia

The University of Texas Rio Grande Valley

Morgan State University

Connected Vision for Increased Pedestrian Safety (CVIPS)

PI: Vijayakumar Bhagavatula
OrcID: 0000-0001-7126-6381

Co-PI: Aswin Sankaranarayanan
OrcID: 0000-0003-0906-4046

Project Participant: Dereje Shenkut

Contract: # 69A3552344811

Final Project Report - July 31, 2024

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated in the interest of information exchange. This report is funded, partially or entirely, by a grant from the U.S. Department of Transportation's University Transportation Centers Program. The U.S. Government assumes no liability for the contents or use thereof.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Overview

Pedestrian safety has become a critical concern due to the higher risk of serious injury they are prone to in traffic accidents. In the United States, the situation has reached alarming levels, with pedestrian fatalities increasing dramatically over the past decade. According to the Governors Highway Safety Association (GHSA), pedestrian deaths rose from 4,280 in 2010 to an estimated 7,508 in 2022, a staggering 77% increase [1]. This surge far outpaces the 25% rise in total traffic fatalities during the same period, highlighting a disproportionate risk to pedestrians. The severity of this issue is further emphasized by 2022 seeing the highest number of pedestrian deaths since 1981. On average, 20 pedestrians lose their lives daily while engaging in routine activities such as commuting, running errands, or exercising. Intersections, where complex interactions occur between various road users, are particularly dangerous. Approximately one-quarter of all traffic fatalities and nearly one-half of all traffic injuries in the US occur at intersections [2].

To improve traffic safety, one promising approach is to advance the level of autonomy in vehicles. This strategy aims to mitigate human-specific problems such as distracted driving and impaired driving. Autonomous vehicles rely heavily on computer vision algorithms applied to data from sensors (e.g., RGB cameras, radar, and LiDAR) which are employed to detect and identify objects within the driving scene.

Although these algorithms perform well in detecting larger objects like vehicles, they struggle to accurately identify pedestrians and other vulnerable road users. LiDAR provides accurate 3D representations of objects, but falls short in semantic interpretation. LiDAR also may not give enough points for smaller objects such as pedestrians. Cameras offer rich semantic details, but they have limitations in measuring depth accurately. To overcome the deficiencies inherent in single-sensor processing, multisensor fusion techniques are being devel- oped. These methods integrate data from multiple sensing modalities to form a comprehensive and reliable perception system. This fusion aims to combine the depth information of LiDAR with the detailed semantic information from the cameras, thus creating a more complete and nuanced under- standing of the driving environment [3]. Although progress in single-vehicle perception has been notable, it exhibits significant shortcomings, notably in its limited sensing range and susceptibility to occlusions [4]. Single-vehicle perception performance degrades in the presence of occlusions caused by other vehicles or obstacles. Distant objects often provide sparse measurements; for instance, they might only cover a few pixels in images or constitute a small number of points in LiDAR point clouds. In addressing the challenges of occlusions and long-range issues, the concept of collaborative perception (CP) has gained traction within the autonomous driving community. This paradigm extends beyond the single vehicle perception system, allowing for a collective approach through vehicle-to-everything (V2X) communication technologies. Through connected and autonomous vehicles (CAVs) and smart infrastructure, collaborative perception aims to forge a more expansive and integrated sensory network, where vehicles, infrastructure, and other entities in the traffic ecosystem share complementary perception data. CP aims to construct a more complete and dynamic representation of the traffic environment, enhancing the decision-making capabilities of autonomous systems.

In line with the vision of the US DOT for pedestrian safety, as well as for a more connected autonomous vehicle approach, this work focuses on investigating existing pedestrian detection challenges, the potential of collaborative or cooperative perception to further improve pedestrian detection, and the impact of communication on collaborative perception. In Chapter 2, we discuss challenges in pedestrian detection and introduces a collaborative perception approach for improving pedestrian detection. Chapter 3 studies the impact of communication on performance of Lidar based collaborative perception on state of the art collaborative perception method. In Chapter 4, we discuss the potential of state-of-the-art vision language models for better understanding the pedestrian's trajectory and behavior around intersections.

Chapter 2

Pedestrian Detection Via Collaborative Perception

2.1 Introduction

2.1.1 Challenges in Pedestrian Detection

Autonomous driving technology has made significant progress in recent years. However, ensuring the safety of pedestrians through accurate detection remains a significant challenge. Current systems exhibit a notable performance gap in detecting pedestrians and other Vulnerable Road Users (VRUs) compared to larger objects such as vehicles. This performance disparity is illustrated in Figure 2.1, which shows the average AP values for cars, pedestrians, bicycles, and motorcycles for various distance thresholds ranging from 0.5 to 4.0 meters. Average Precision (AP) is calculated based on the precision of object detection at various distance thresholds. Precision is defined as the ratio of True Positives (TP) to the sum of True Positives and False Positives (FP). The definition of a True Positive varies with the distance threshold (d), which in this evaluation is considered at four specific distances: 0.5m, 1m, 2m, and 4m. A detection is considered a True Positive if the center of the predicted bounding box is within the specified distance d from the center of the ground truth bounding box. The AP value reported in the figure is the average of the precision values calculated at these four distance thresholds. This multi-threshold approach provides a comprehensive evaluation of the detection model's performance, balancing the need for precise localization (at 0.5m) with more lenient criteria (up to 4m) that may be relevant in certain autonomous driving scenarios. The figure thus illustrates how different object classes (cars, pedestrians, bicycles, and motorcycles) perform across these varying levels of localization strictness, offering insights into the model's capabilities and limitations in detecting and accurately positioning different types of road users. The final AP indicated shows the average of AP@{0.5,1,2,4} meters. The performance on cars consistently outperforms all other categories across all distance thresholds, with the highest AP of about 0.83 at 4.0 meters. Pedestrians show the second-best performance among VRUs, but their detection accuracy declines more steeply than cars as the distance threshold decreases, i.e., as we demand smaller distance between the estimated object location and ground truth location of that object. Motorcycles and bicycles demonstrate lower AP values overall, with bicycles showing the poorest detection performance across all thresholds.

Several factors contribute to the underperformance of the computer vision methods on pedestrians and other VRUs.

- **Dataset bias:** Existing datasets tend to have more vehicle instances compared to VRUs, skewing detection algorithms towards larger objects. This imbalance results in models that are better trained to recognize and localize vehicles, while underperforming on VRUs. Creating more balanced datasets that accurately represent the diversity and frequency of VRUs in real-world traffic scenarios is important for improving detection algorithms.

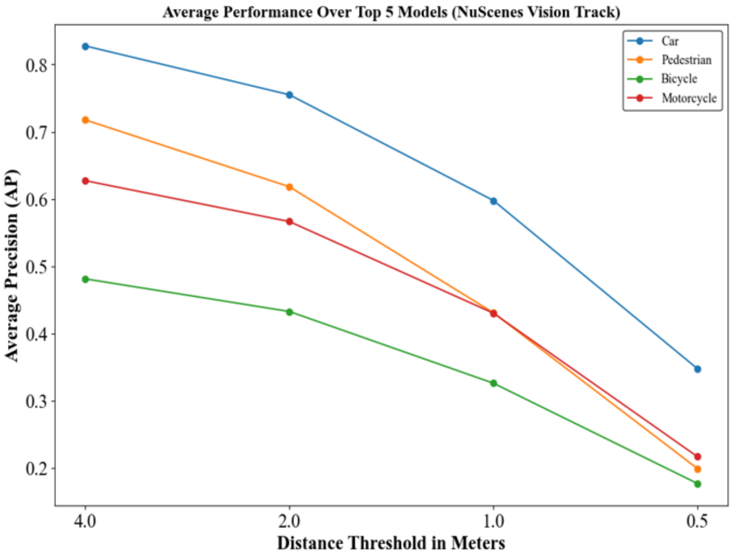

Figure 2.1: Detection performance comparison across object classes. Average Precision (AP) of top 5 models in the nuScenes [3] Vision track challenge as a function of detection distance threshold (as of February 2024).

- **Physical characteristics:** Pedestrians and other VRUs present unique challenges due to their smaller size, which leads to fewer representative pixels in the images and increased susceptibility to occlusion. This makes it difficult for current detection systems to accurately identify and track VRUs, especially in cluttered urban environments. Developing algorithms that can better handle small-scale objects and partial occlusions is essential for improving VRU detection.

- **Movement patterns:** VRUs exhibit less predictable movement compared to vehicles, which typically follow established traffic rules and road layouts. This unpredictability makes it challenging for current systems to anticipate and track VRU movements accurately. Improving trajectory prediction models and incorporating more sophisticated behavior modeling for VRUs could enhance detection and tracking performance.

- **Environmental factors:**Detection performance often degrades significantly under challenging conditions such as poor lighting, sun glare, or extreme obstructions. These conditions are particularly problematic for VRU detection due to their smaller size and variable appearance. Developing robust algorithms that can maintain high performance across a wide range of environmental conditions is crucial for reliable VRU detection in real-world scenarios.

The widening performance gap between vehicles and VRUs at more stringent distance thresholds highlights the challenge of precise localization for smaller, more dynamic objects. This underscores the need for improved detection algorithms and training strategies specifically tailored to enhance the accuracy of VRU detection in autonomous driving systems. Addressing these challenges is crucial for improving pedestrian safety and advancing the overall capabilities of autonomous vehicles. This chapter explores approaches to improve VRU detection, with a focus on collaborative perception techniques that aim to mitigate these limitations and improve pedestrian safety in autonomous driving scenarios.

2.1.2 VRU-Specific Detector Vs Generic Detector

In order to study the need for training algorithms tailored to pedestrians, cyclists, and motorcyclists, we used DeepAccident dataset [5] to compare the baseline model trained with six classes to a VRU-specific specific detector. The baseline DeepAccident model has car, truck, van, cyclist, motorcyclist, and pedestrian classes. A comparison is made with a VRU-specific model trained with 3 classes, namely pedestrian, cyclist, and motorcyclist.

DeepAccident DeepAccident [5] is introduced for end-to-end motion and accident prediction tasks on the autonomous vehicle side, along with various perception tasks in V2X (vehicle-to-everything). It contains a dataset recorded from four vehicles and one infrastructure each with six cameras at the intersection. In our experiment, we used samples with VRU classes (pedestrians, motorcyclists, and cyclists) to compare the performance of a model specifically trained with VRU classes with the performance of a generic baseline model trained with six classes, namely car, truck, van, cyclist, motorcyclist, and pedestrian.

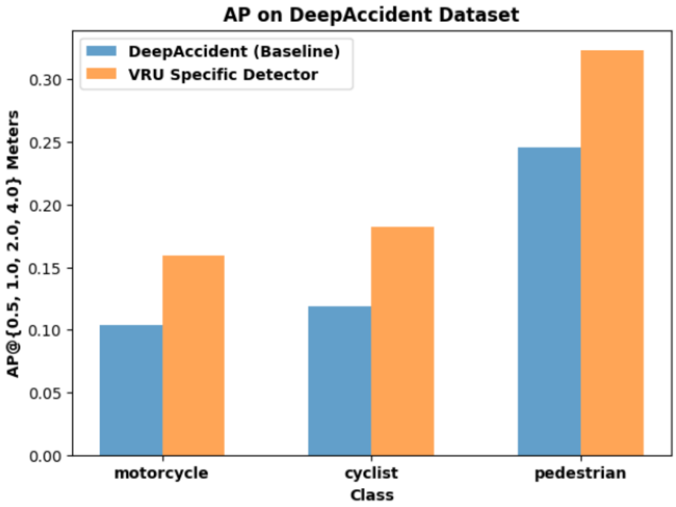

Figure 2.2: Performance of model trained on only three VRU classes versus generic DeepAccident [5] baseline model trained on six classes

Figure 2.2 demonstrates the performance improvement gained by training a VRU-specific model. The model trained with pedestrians, cyclists, and motorcyclists outperformed the baseline model in AP@{0.5,1,2,4} meters for all three classes, suggesting the need to design VRU specific detectors. It also shows that there is a performance difference between the three classes. This can be attributed to the class imbalance problems in the dataset, as the number of pedestrian instances is greater than that of motorcyclists and cyclists.

To address this gap, we investigate a computer vision approach that enhances pedestrian detection through camera-only collaborative perception. This involves using synchronized cameras in both vehicles and infrastructure to cooperatively detect pedestrians at intersections. Due to the unavailability of annotated real-world datasets collected in a collaborative setup, we generate synchronized vehicle and infrastructure-side video using the high-fidelity CARLA simulator. This synthetic dataset is then used to train and evaluate deep learning algorithms for pedestrian detection in a collaborative setting. Our preliminary results demonstrate the potential of CP to significantly improve pedestrian detection. Our main contributions include the following:

- We first discuss in detail the concept of collaborative perception and review related works in detail

- Creating a synchronized dataset from both vehicle and infrastructure perspectives, specifically tailored for pedestrian detection under normal and challenging conditions

- Developing a vision-only collaborative perception technique focused on pedestrian detection

2.2 Toward Collaborative Perception

Collaborative perception has emerged as a vital component in improving vehicle safety and navigation. This paradigm uses collective sensory input from surrounding vehicles and road infrastructure to create a comprehensive understanding of the environment, mitigating limitations such as limited field of view and occlusion. Collaborative perception can be camera-only [6], LiDAR-only [7, 8], or involve fusion of processed image and LiDAR point cloud features. In terms of data sharing and collaboration stage, collaborative perception in autonomous vehicles can be categorized into early fusion, intermediate fusion, and late fusion. Collaborative perception has become an important option for in improving vehicle safety and navigation. This paradigm leverages the collective sensory input of multiple agents, such as vehicles and infrastructure, through vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I) and vehicle-to-everything (V2X) communications, fostering a more comprehensive understanding of the driving environment by improving the capabilities of individual agents by mitigating limitations such as limited field of view and occlusion. There are three types of CP, early fusion, intermediate fusion, and late fusion, based on the data sharing and fusion stage [9]. Fig 2.3 illustrates the three collaborative stages.

2.2.1 Fusion Types

- **Early Fusion:** Early fusion involves the exchange of raw-level data, such as images or LiDAR point clouds. This method requires higher communication bandwidth due to the transmission of unprocessed raw data. While it potentially offers better performance by allowing each agent access to the most complete information, it comes with significant drawbacks. The high bandwidth requirement can be a limiting factor in real-world applications. Additionally, this approach necessitates substantial processing capability at all agents to handle the raw data, which may not always be feasible or cost-effective.

- **Late Fusion:** Late fusion involves transmitting the final perception outputs, such as detection bounding boxes. This approach requires less bandwidth than early fusion, making it more efficient in terms of data transmission. However, it may lead to processing and transmission delays, as the data is processed by each agent before sharing individual agent's perception outputs. While late fusion is bandwidth-efficient, it has its own set of challenges. The transmission delay can be critical in time-sensitive applications, and the need for processing capability at all agents remains a consideration.

- **Intermediate Fusion:** Intermediate fusion represents a balance between early and late fusion methods. Each agent processes the raw data into intermediate features and then compresses them before transmitting. This approach aims to strike a better performance-bandwidth trade-off. By exchanging processed features rather than raw data or final outputs, intermediate fusion can potentially offer a good compromise between the high performance of early fusion and the bandwidth efficiency of late fusion. This method allows for more flexibility in managing the balance between data transmission and processing requirements.

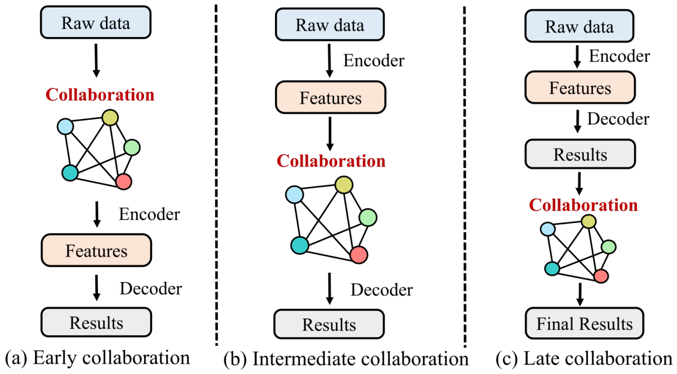

Figure 2.3: Schematic representation of the three fusion types in collaborative perception [10]: (a) Early collaboration, where raw data is shared directly; (b) Intermediate collaboration, where features extracted from raw data are shared; and (c) Late collaboration, where only final results are shared among agents.

2.3 Related Work

2.3.1 Pedestrian Detection

Accurately detecting vulnerable road users (VRUs), such as pedestrians, cyclists, and motorcyclists continues to be a significant challenge for autonomous vehicles (AVs). While the main focus is detecting road vehicles, pedestrians are the most explored among the VRU classes. Works on VRU perception involve a pure camera-based approach, fusing from multiple sources and different V2X-based data exchanges between the VRUs and the vehicles, showcasing a range of approaches from communication technologies to machine learning and computer vision.

In [11], a computer vision-based system is proposed for recognizing VRU hand signals, using CNN for enhanced detection accuracy. [12] introduced machine learning-based movement models for predicting VRU behavior, demonstrating improved trajectory prediction [13] developed a deep generative model for detecting interactions between vehicles and VRUs at intersections, using a conditional variational auto-encoder. [14] conducted an extensive study on the parameters affecting VRU detection in ADAS, highlighting the complexity of VRU appearances and behaviors. The PROSPECT project [15] proposed a method to improve active VRU safety systems by integrating various data sources and developing advanced sensor processing and intervention strategies. [16] emphasized the importance of simulation software in the development of VRU detection systems, combining radar and vision sensing for effective pedestrian and cyclist detection. [17] introduced an approach using mobile phones for vehicle-to-VRU communication, enhancing the detection and safety of VRUs. [18] proposed a multi-sensing and communication approach, leveraging smart city sensors and vehicle and VRU data for predicting potential collisions. [19] evaluated the performance of V2X communications technologies in enhancing VRU safety, particularly in urban intersection scenarios.[20] discussed the effectiveness of messaging protocols in V2X communication for VRU protection, emphasizing the combination of sensor data sharing and active VRU transmissions.

2.3.2 Collaborative Perception

Collaborative perception has emerged as a vital component for enhancing vehicle safety and navigation. This paradigm leverages the collective sensory input from multiple agents, such as vehicles (V2V), vehicle to infrastructure (V2I), and vehicle-to-everything (V2X), to create a comprehensive understanding of the environment by improving the capabilities of individual agents by mitigating limitations such as limited field of view and occlusion.

The majority of studies focus on using one type of sensor for collaboration. Methods like Robust V2V [21], V2VNet [22] and Adversial V2V [23] use point cloud input for detection, prediction, and planning tasks of autonomous vehicle via intermediate (feature-level) collaboration. DiscoNet [10] uses a mix of early and intermediate collaboration with collaborative graph representation. Other LiDAR-only collaborative perception works include AttFuse [24] which introduced attention-based intermediate V2V collaboration, In similar work, V2X-ViT [25] introduced vision-transformer-based collaboration, while SyncNet [26] studied latency-aware collaboration in addition to attention-based fusion. Other research such as Where2comm [27] focused on reducing communication bandwidth needs without affecting performance. Coopernaut [28] explores end-to-end driving via cooperative perception. MPDA [29], and DI-V2X [30] delved into collaboration with unidentical agents. CoAlign [21] introduced a collaborative scheme robust to unknown pose errors, [31], DUSA [32] explores sim2real adaptation in cooperative perception. CO3 [33] studied unsupervised contrastive learning for vehicle-infrastructure point cloud features collaboration. UMC [34] focuses on multi-resolution collaborative learning. SCOPE [35], CORE [36], FFNet [37], CoBEVFlow [38], FF-Tracking [39], and AR2VP [40] contribute to detection and segmentation tasks focusing on vehicles' navigation. While almost all previously listed works focus only on vehicles, AdaFusion [31] gives focus to pedestrian detection as well.

To further improve perception performance in cooperative settings, recent multi-modal intermediate-level fusion approaches explored LiDAR-camera fusion for each agent. CoBEVT [41] and CoBEVFusion [42] demonstrated that bird's-eye view fusion can significantly improve segmentation and detection tasks in a cooperative setting, HM-ViT [43] introduced graph transformer for lidar-camera fusion and between agent's interaction, LAV [44] used multi-modal sensor reading for perception and planning in CARLA driving challenge. Due to the high cost of LiDAR and the ability of the camera-based approach to mimic human-like perception, recent work focuses on camera-only collaborative perception. QUEST [45], CoCa3D [46], V2XFormer [5] use camera-only collaboration. In other application, When2com [47] for collaborative robotic learning from aerial RGB image.

Datasets and simulators have been equally crucial in propelling research in this domain. Emphasis is being given to diverse and high-quality synthetic data generated on CARLA and real cooperative datasets as well to advance this specific field. They provide the realistic scenarios and benchmarks required to train and evaluate collaborative perception models. V2X-Sim [48], DeepAccident [5] and DAIR-V2X [49] are noteworthy contributions, offering a large-scale setting for vehicle-infrastructure cooperative 3D object detection. Similarly, OPV2V has become a standard benchmark for assessing the performance of LiDAR-based multi-agent perception systems enabling researchers to simulate and evaluate complex V2X interactions.

Within the scope of current collaborative perception research, both datasets and algorithms predominantly concentrate on vehicle-related tasks such as detection, tracking, and motion forecasting. However, this focus has inadvertently resulted in less robust perception capabilities for other road users, including pedestrians, cyclists, and motorcyclists. Addressing this disparity is important for a more comprehensive and safer understanding of the road environment.

2.3.3 Perception Uncertainty

Estimating perception uncertainty is critical for AV perception. The estimated uncertainty is used to measure robustness under challenging conditions, fuse perception from multiple sensors (lidar, camera, radar, etc) [50], [51], [52] as well as to exchange perception results from multiple agents. [50] introduced a method that combines multi-source perception fusion and deep ensemble for real-time evaluation in autonomous vehicles. This approach assesses the effectiveness of single-frame perception results and spatial uncertainty of detected objects. Similarly, [51] presented Uncertainty-Encoded Mixture-of-Experts (UMoE) for LiDAR-camera fusion, which uses MC dropout to effectively incorporate single-modal uncertainties into multi-modal fusion, enhancing object detection under various challenging conditions. Additionally, [53] addresses the domain drift problem in autonomous driving with a domain adaptive object detection algorithm based on feature uncertainty. Their approach, which includes a local alignment module and an instance-level alignment module guided by feature uncertainty, shows improved detection performance in unlabeled data. These methods show the purpose of perception uncertainty in autonomous driving, in tackling key challenges of multi-modal fusion, robust detection, and domain adaptation, and paving the way for more reliable and accurate autonomous driving systems.

2.4 Proposed Approach

This section outlines our methodology for generating a comprehensive dataset using the CARLA simulator and employing it for collaborative pedestrian detection. We first describe the dataset generation process, detailing the simulation setup and data collection techniques. Following this, we introduce the collaborative perception method, emphasizing the per-agent detection unit and the subsequent fusion of detection results from multiple agents to enhance accuracy and robustness in complex urban scenarios.

2.4.1 Dataset Generation

An annotated, synchronized vehicle and infrastructure side dataset that covers a wide range of scenarios involving pedestrians is currently not available. Thus, a multi-camera and multi-agent dataset that focuses on pedestrians is generated using the high-fidelity CARLA [54] simulator.

CARLA. CARLA (Car Learning to Act) is an open-source simulator for autonomous driving research. It provides a high-fidelity virtual environment built on the Unreal Engine, offering realistic urban settings with various weather conditions, vehicles, and pedestrians. CARLA, as illustrated in Figure 2.4 allows for the generation of synchronized multi-agent data, including multiple camera views from both vehicles and infrastructure, LiDAR point clouds, depth maps, semantic segmentation, instance segmentation, and bounding boxes for objects. The simulator can be controlled via Python scripts, enabling customized scenario creation and data collection. This flexibility makes CARLA ideal for generating diverse data sets tailored to specific research needs, such as collaborative perception scenarios geared toward pedestrians.

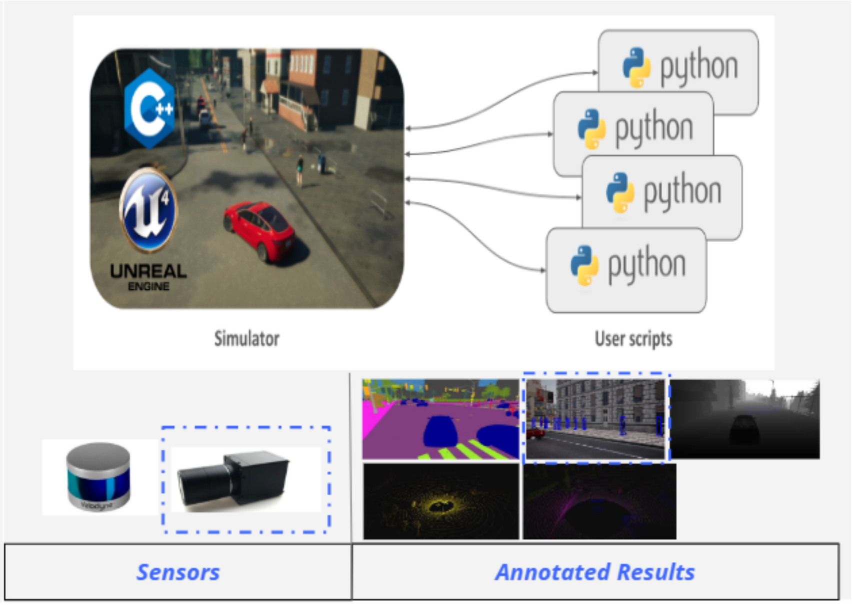

Figure 2.4: CARLA simulator setup for collaborative perception dataset generation. The simulator provides a realistic urban environment, while Python scripts control scenario generation. Various sensor outputs and annotated results are collected, including RGB images, depth maps, semantic segmentation, and bounding boxes.

Agent and sensor setup. Figure 2.5 illustrates an example of how vehicles and infrastructure units with cameras mounted on them are placed at the intersection. Each vehicle and infrastructure unit is equipped with six cameras working at 20 fps frame rate. Each camera has a field of view (FOV) of 70°, except for the back camera, which has an FOV of 110°, following the nuScenes [3] data collection framework. Each image is tagged with timestamp and saved at a 10Hz interval rate. The dataset covers different scenarios such as occlusions and non-line-of-sight situations, crowded pedestrian scenes, diverse weather conditions and different times of the day (noon, sunset, etc.), and varying speeds and profiles of pedestrians.

2.4.2 Camera Only Collaborative Pereption Method

Per Agent Detection Unit

We adopt V2XFormer [5] as birds eye view (BEV)-based 3D detection method for single-agent detection. It involves processing each image sequence with the image view encoder and transforming it into BEV features in each agent. Then, the BEV feature is fed to the detection head, which results in bounding box candidates and a spatial heatmap that serves as detection confidence.

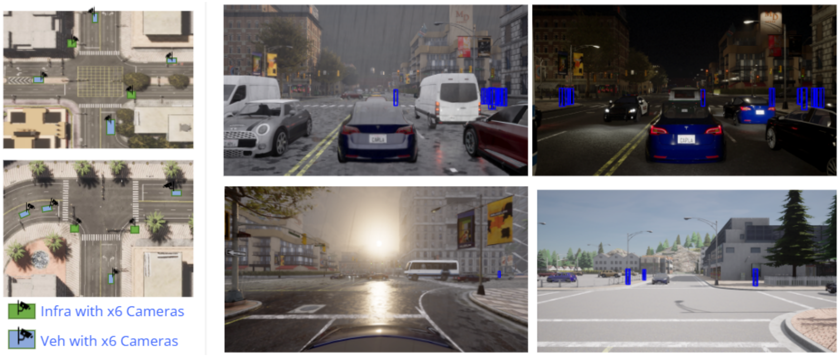

Figure 2.5: **Dataset generation setup and sample images.** **Left:** Bird's-eye views of intersections showing infrastructure (green) and vehicle (blue) agents, each equipped with six cameras. **Right:** Four sample images showcasing diverse scenarios: rainy conditions (top-left), nighttime scene (top-right), sun-glare effect (bottom-left), and sunny daytime traffic (bottom-right). Blue bounding boxes indicate pedestrians. These images demonstrate challenging scenarios including occlusions, varying lighting conditions, and complex urban environments, crucial for training robust collaborative perception models.

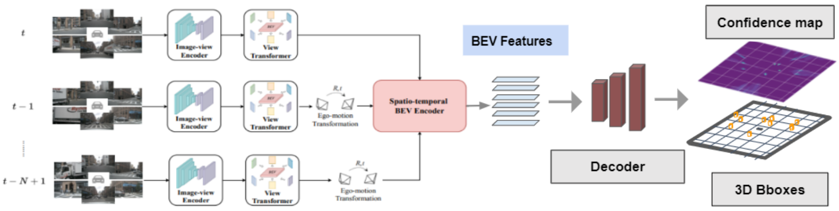

Figure 2.6: Multi-frame image processing pipeline for 3D object detection for each agent, based on [5]. The workflow includes image-view encoding, view transformation, and ego-motion compensation across multiple time frames (t, t-1, ..., t-N+1) for N past frames. A spatio-temporal BEV encoder processes these inputs to generate BEV features. The decoder then produces a confidence map and 3D bounding boxes.

Image Encoding

Given a sequence of T frames from agents N, each equipped with six cameras, every frame from each camera is encoded into a rich and dense representation of features F, $F \in R^{H^{'} \times W^{'} \times C^{'}}$, where $H^{'}, W^{'}, C^{'}$ are the height, width, and channel of the characteristic of the image.

Image Features to BEV Transform

Each image feature of T frames is discretized into a pseudo-density point cloud. Then, the temporal data is encoded, and past features are warped to the current reference frame using a spatiotemporal encoder that extracts spatial and temporal information using 3D convolution, resulting in aligned BEV features for each agent.

Detection & Confidence Estimation Head

The detection head consists of convolutional blocks that generate unfiltered 3D bounding boxes. Additionally, it includes a learnable heatmap prediction block that outputs a Gaussian heatmap representing the detection confidence. The heatmap is created with a Gaussian kernel of radius r and standard deviation σ with the peak at the center of the bounding box. This allows a fine-grained understanding of the detection performance across different spatial regions and fusion based on that.

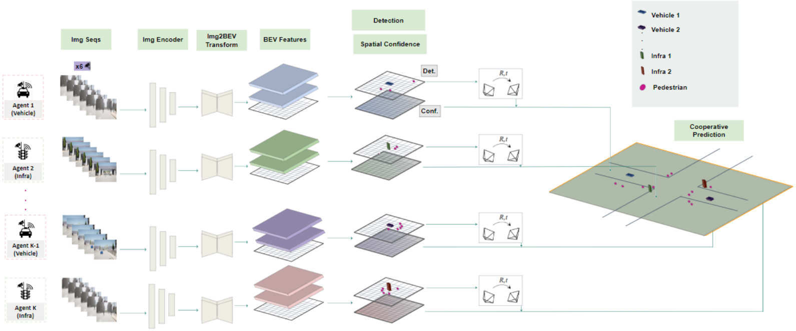

Figure 2.7: **Collaborative perception for pedestrian detection**. Each agent captures sequences of images that are encoded into image features from the six cameras. These features are subsequently converted into Birds-Eye-View (BEV) representations. The BEV features are fed into a 3D detection head that estimates per agent 3D bounding box detection **(Det.)** and detection confidence **(Conf.)** as shown in 2.6 which is then transformed with transformation matrix [**R, t**] into the main unit for cooperative prediction.

2.4.3 Vehicle-Infrastructure Collaborative Detection

The detection results from multiple agents (vehicles and infrastructure) are fused based on confidence estimates in the Gaussian heatmap. Each agent produces a confidence heatmap and a list of 3D bounding boxes that are not filtered with Non-maximum Suppression (NMS). After the confidence map and candidates for the bounding boxes of each agent are transformed into the main cooperative unit, the final bounding box is obtained by choosing the result from the agent with the highest spatial confidence for that bounding box. This fusion process is guided by confidence levels, ensuring that more reliable detections have a greater influence on the combined detection output.

2.5 Numerical Experiments

Collaborative perception setup: Each frame has a resolution of 1600x900 which is resized to 224x224 and fed to the image backbone, resulting in a spatial dimension of 704x256. The image features are then transformed into a BEV feature transformation of grid size 1024x1024, corresponding to an actual ground area of 102.4x102.4 meters around the agent. We first train a single agent detection baseline and then study the impact of adding collaborative agents.

2.5.1 Preliminary Results and Discussions

Figure 2.8 illustrates the average precision (AP) calculated for pedestrian detection in a collaborative setting and compares it with the performance of a single agent. The single agent vehicle or infrastructure side AP is calculated as the average over all six agents' detection results. For collaborative setup, using infrastructure as the main unit, we gradually add one agent at a time and record the AP. Performance has steadily improved as the number of collaborative agents increases going from AP of 0.34 for the single agent case to 0.51 when all the agents participate in the fusion process. This preliminary result suggests that a collaborative vehicle-infrastructure system, where vehicles and infrastructure share their perception results, can considerably improve pedestrian detection performance.

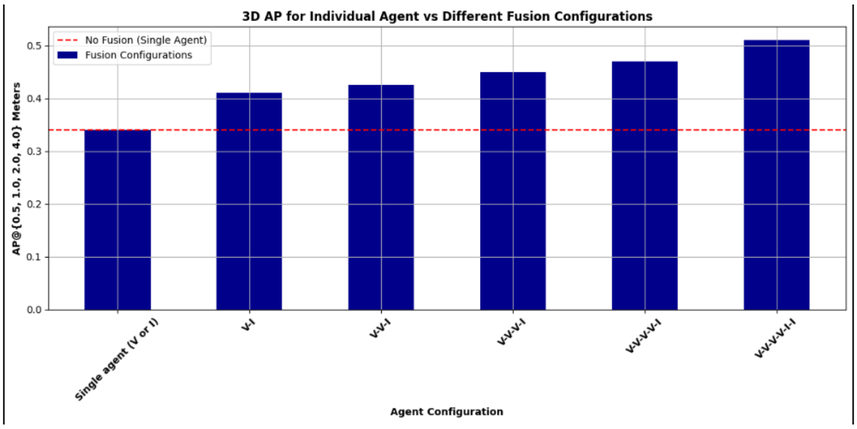

Figure 2.8: Collaborative perception performance. Average Precision (AP) improves as the number of collaborating agents increases. Starting from a single agent or average of vehicle-only or infrastructure-only (AP = 0.34), performance improves with vehicle-infrastructure collaboration (AP = 0.41), reaching the best performance when all 6 agents (4 vehicles and 2 infrastructure units) collaborate (AP = 0.51). V and I represent vehicle and infrastructure, respectively

2.6 Conclusion

This chapter presented a camera only collaborative perception approach to pedestrian detection. We generated a synthetic dataset using the CARLA simulator, designed for collaborative perception scenarios mainly involving pedestrians. This dataset aims to address the current lack of annotated, synchronized vehicle and infrastructure data for pedestrian detection in a collaborative perception set-up. We then proposed a camera-only collaborative perception method that utilizes these multi-agent data. Our preliminary experiments indicated an improvement in Average Precision (AP) when using collaborative perception compared to single-agent detection. While these initial results are promising, further research is needed to fully validate the approach. This work represents a step towards enhancing pedestrian detection in autonomous driving systems, potentially contributing to improved safety for vulnerable road users in urban environments.

Chapter 3

Impact of Communication Limitations on Collaborative Perception

3.1 Introduction

As discussed in Chapter 2, the use of vehicle-to-everything (V2X) communications for sensor data exchange is emerging as a crucial strategy for enhancing pedestrian safety. Collaborative perception (CP) transcends single-vehicle perception systems, enabling a collective approach through V2X communication technologies. By leveraging connected and autonomous vehicles (CAVs) and smart infrastructure, CP aims to create an expansive and integrated sensory network, facilitating the exchange of complementary perception data among vehicles, infrastructure, and other entities within the traffic ecosystem. This allows the construction of a more comprehensive and dynamic representation of the traffic environment, thereby enhancing the decision-making capabilities of autonomous systems.

CP involves a complex multi-agent[1] fusion[2] process, which introduces several practical challenges. For instance, communication latency and interruptions can significantly impact perception performance, necessitating strategies to mitigate the effects of time delays. Efficiency in collaborative perception is crucial, as the system must manage data exchanges within bandwidth constraints without compromising the integrity and utility of shared information. Moreover, collaborative systems are susceptible to adversarial attacks, requiring robust defenses to ensure data reliability. Accurate alignment of data from multiple sensors is also critical for maintaining CP performance, which can be affected by location errors between collaborating agents. Additionally, integrating perception models from different vehicles presents unique challenges, demanding advanced fusion techniques to manage discrepancies and maintain overall system performance. Addressing these challenges requires a multifaceted approach to ensure the seamless integration of collaborative perception into operational systems. Consequently, managing communication limitations such as latency, bandwidth constraints, collaborative agents' location errors, and communication interruptions is becoming increasingly critical. Most CP methods operate under the assumption of ideal communication conditions, focusing primarily on improving perception performance. Despite extensive research in CP, the impact of communication limitations has not been fully explored. Figure 3.1 illustrates the effects of latency and communication interruptions on LiDAR-based detection, underscoring the need to study the impact of communication limitations on CP. This chapter examines the effects of latency, communication interruptions,

[1] "Agent" refers to a vehicle or infrastructure unit with sensing and connection capabilities.

[2] "Collaboration" and "multi-agent fusion" are used interchangeably.

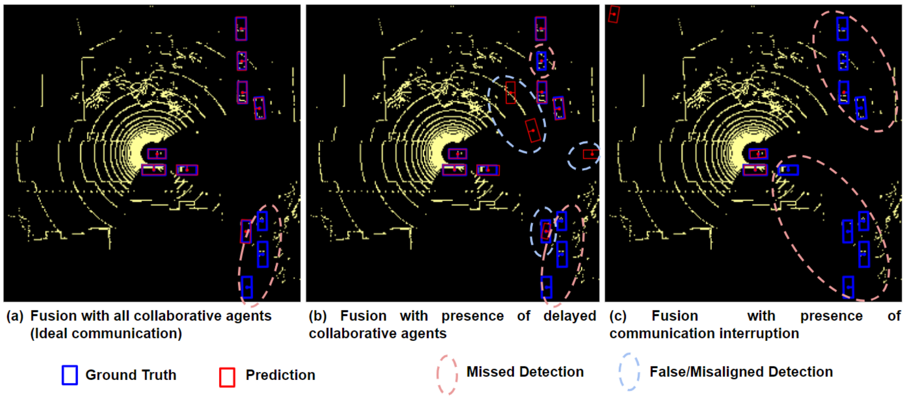

Figure 3.1: **Impact of communication limitations on collaborative detection.** (a) Ideal communication with all agents; (b) Presence of delayed collaborative agents (by 400 ms), resulting in false/misaligned detections; (c) Presence of communication interruptions, leading to more missed detections compared to (a).

and bandwidth constraints on collaborative detection performance using a state-of-the-art CP model. To study these impacts, we conduct the followings:

– An investigation of how different levels of latency influence the performance of LiDAR-based collaborative detection.

– An analysis of the impact of various compression levels during data transmission on collaborative perception systems.

– An evaluation of the effects of random communication interruptions on collaborative detection results.

– A proposed method to mitigate latency and communication interruption using a lightweight spatio-temporal feature prediction model.

The remainder of this chapter is structured as follows: We begin with a brief introduction to CP, followed by a detailed review of existing literature on CP systems. We then describe the graph-based CP framework used to assess the impact of communication limitations. Subsequently, we discuss the results of our experiments and conclude by summarizing our findings and outlining considerations for future work in this domain.

3.2 Related Work in Non-ideal Collaborative Perception

Most collaborative perception (CP) methods discussed in Chapter 2, Section 2.3.2 assume ideal communication scenarios. Some recent work has begun to address the challenges posed by non-ideal conditions in real-world applications. These studies investigate how factors such as latency, bandwidth limitations, pose errors, and the gap between simulated and real environments can significantly impact the performance of CP systems.

In studying latency-aware collaboration, SyncNet [26] has made notable contributions. This work not only studied the effects of communication delays but also integrated attention-based fusion and temporal alignment techniques to mitigate these issues. Similarly, Where2comm [27] tackled the critical challenge of bandwidth limitations in CP systems. Their approach focused on reducing communication bandwidth requirements without compromising the overall performance of the collaborative perception system.

Addressing the challenge of pose errors in CP, RobustV2V [21] proposed a collaborative scheme designed to be robust against unknown sensor location errors. This work is particularly significant as accurate pose information is crucial for effective collaboration between multiple agents. In an another work, V2X-ViT [25] introduced a vision transformer-based collaboration method capable of handling both ideal and noisy localization scenarios, further enhancing the robustness of CP systems in real-world conditions.

The gap between simulated and real environments, a persistent challenge in autonomous driving research, has been addressed by DUSA [32]. This work explored sim2real adaptation techniques in the context of cooperative perception, aiming to improve the transferability of models trained in simulated environments to real-world scenarios. Additionally, CO3 [33] contributed to this area by studying unsupervised contrastive learning for vehicle-infrastructure point cloud feature collaboration, potentially offering a way to reduce the reliance on large amounts of labeled real-world data.

Some researchers have also explored CP in the context of heterogeneous agent networks. MPDA [29] and DI-V2X [30] investigated collaboration strategies for non-identical agents, addressing the reality that different vehicles and infrastructure elements may have varying sensing and processing capabilities. These works contribute to making CP systems more adaptable and robust in diverse real-world settings. Lastly, Coopernaut [28] took a holistic approach by exploring end-to-end driving via cooperative perception. This work potentially bridges the gap between perception and control in autonomous driving systems, considering the challenges of non-ideal conditions throughout the entire driving pipeline.

Although recent studies have advanced collaborative perception (CP) in non-ideal conditions, a significant gap remains between theoretical progress and real-world applications. Many current CP efforts assume ideal communication scenarios, overlooking crucial limitations in autonomous driving environments. Our study addresses this gap by systematically investigating the impact of communication challenges on CP performance. Specifically, we examine the effects of varying latency levels, data compression ratios, and random communication interruptions on LiDAR-based collaborative detection. To mitigate these issues, we propose a lightweight spatio-temporal feature prediction model. This comprehensive approach bridges the gap between theoretical advancements and practical implementations, contributing to the development of more robust and reliable CP systems for real-world autonomous driving scenarios.

3.3 Collaborative Perception Framework

In this section, we cover the CP framework used to study the impact of communication limitations. In this work, we adopt the state-of-the-art open-source LiDAR-based CP method named **coperception** with DiscoNet [10]. This method uses a student-teacher knowledge distillation model in which the teacher uses raw-level fusion and the student uses a graph-based feature-level collaborative method as shown in Figure 3.2. The following sections discuss the chosen method and the components that are used to simulate the communication limitations.

3.3.1 Early Fusion During Training

Early collaboration provides the upper bound of performance due to the aggregation of raw LiDAR data from all collaborating agents. As shown in Figure 3.2, the teacher model uses early collaboration that allows for a comprehensive view of the driving environment. This enables the use of combined data from all agents with the aim of minimizing performance degradation due to issues such as occlusion and perception limitations faced by individual agents. The teacher model processes these aggregated data to guide the learning process in a student model during the inference phase. This model acts as a guide to force the student model to improve CP performance.

The teacher model employs a feature encoder-decoder and output header that are used only during training. For the feature encoding process, the system receives an aggregated 3D point cloud (X_a) from all participating agents $\{X_1, ..., X_k\}$, merging their collected data points within a global coordinate framework. To align the global point cloud X to each agent's reference frame, it is transformed to match the individual coordinate system of the agents, ensuring that the teacher and student models process data within a consistent coordinate system. In the decoding stage, the teacher model feature map is transformed through the feature decoder to produce a bird's-eye-view (BEV)-based feature map. This map then passes through the output header, producing category classifications and bounding box regressions. The training follows the conventional teacher-student methodology, where it is trained independently using binary cross-entropy for category classification and smooth L1 loss for bounding box regression.

The overall loss is a summation of individual loss functions, representing the combined classification and regression errors for each agent's detected ground truth within their perception field.

3.3.2 Graph-Based Intermediate Fusion

Intermediate collaboration is collaboration that focuses on the exchange of intermediate features, rather than raw data or final perception output. This method strikes a balance between bandwidth-heavy early collaboration and potentially noisy late collaboration. Therefore, the student model employs an intermediate-level fusion. Based on [10], a graph-based intermediate collaboration is used to model interactions and data exchange between agents, as illustrated in Figure 3.2. In this graph-based collaboration, the nodes represent an agent, and the edges represent matrix-valued features that are exchanged between the collaboration agents. The strength of collaboration is encoded in these edges, which is learned during training. The collaboration graph facilitates the aggregation of features from different agents, allowing for a more nuanced and efficient fusion of information. This process is designed to adaptively learn the specific contributions of each agent to the overall perception task.

Similar to the teacher model, the student model also includes feature encoding-decoding stages and taskhead. Here, each agent i processes the 3D point cloud input X_i with its feature encoder. The encoder transforms the 3D point cloud into a bird's-eye-view (BEV) map suitable for 2D convolution operations. This BEV map, a 2D representation of the 3D point cloud, undergoes a series of convolutional, batch normalization, and ReLU activation operations to refine and enrich the feature data. The feature maps are then compressed prior to transmission. The collaboration graph allows feature map updates through agent interactions. The collaboration graph process consists of transmission, where agents exchange compressed feature maps and attention network, where each agent computes attention weights to assess the importance of received feature maps; and aggregation, where agents update their own feature maps by integrating received features based on the attention.

Through this intermediate collaboration, agents can share compressed, yet informative, feature maps, reducing the required communication bandwidth while still enhancing the collective perception capability. The effect of bandwidth requirement versus performance is studied by adjusting the compression level of the matrix-valued weights of the edges on the graph. The graph also allows for random communication interruption by removing randomly chosen edges from the graph and studying the impact. To study the effect of latency, a frame delay is introduced in the transmission of features from one agent to another.

Following the collaboration process, each agent uses a decoder to refine the updated BEV feature map. This refinement involves upsampling the feature map through a series of layers, each enhancing the details by merging with corresponding features from earlier stages and reducing channel dimensions via convolution. Subsequently, an output header processes this enhanced map to produce the final detection results, identifying object categories and their bounding boxes through convolutional pathways. This structured approach ensures that each agent can accurately interpret and respond to the collective data gathered during the collaboration.

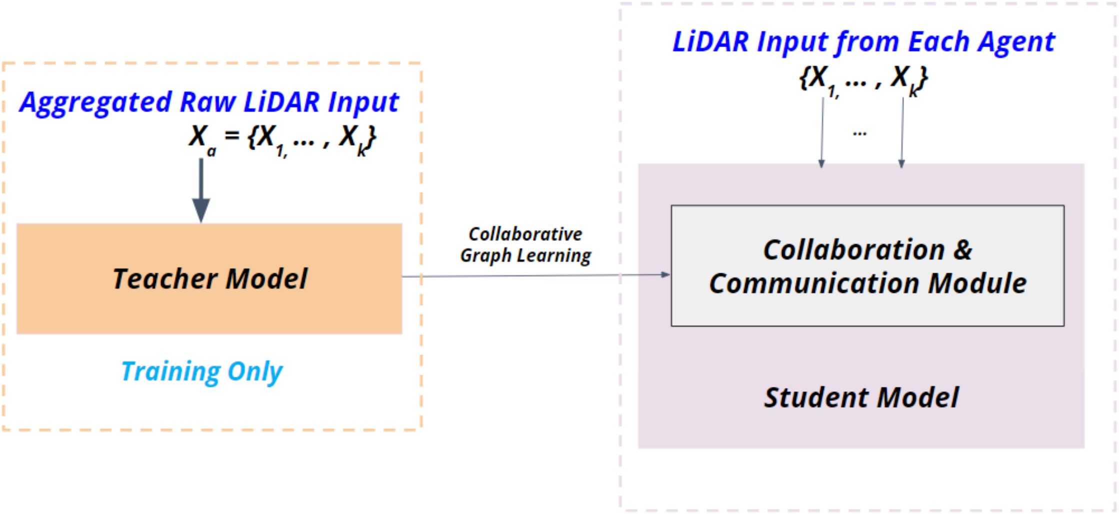
Aggregated Raw LiDAR Input
$X_a = \{X_1, \dots, X_k\}$
Teacher Model
Training Only
Collaborative Graph Learning
LiDAR Input from Each Agent
$\{X_1, \dots, X_k\}$
...
Collaboration & Communication Module
Student Model

Figure 3.2: **Collaborative perception framework.** A LiDAR-based collaborative perception approach utilizing a student-teacher knowledge distillation model [10]. Here, the teacher model employs raw-level fusion, while the student model adopts a graph-based feature-level collaboration method. The collaborative graph is further illustrated in Figure 3.3.

3.3.3 Feature Compression

To study the impact of the size of the information that is being exchanged, each collaborative agent has the ability to compress its feature map (F_s^i) before transmission to reduce the bandwidth requirement. As in most previous works, a 1×1 convolutional filter is used to

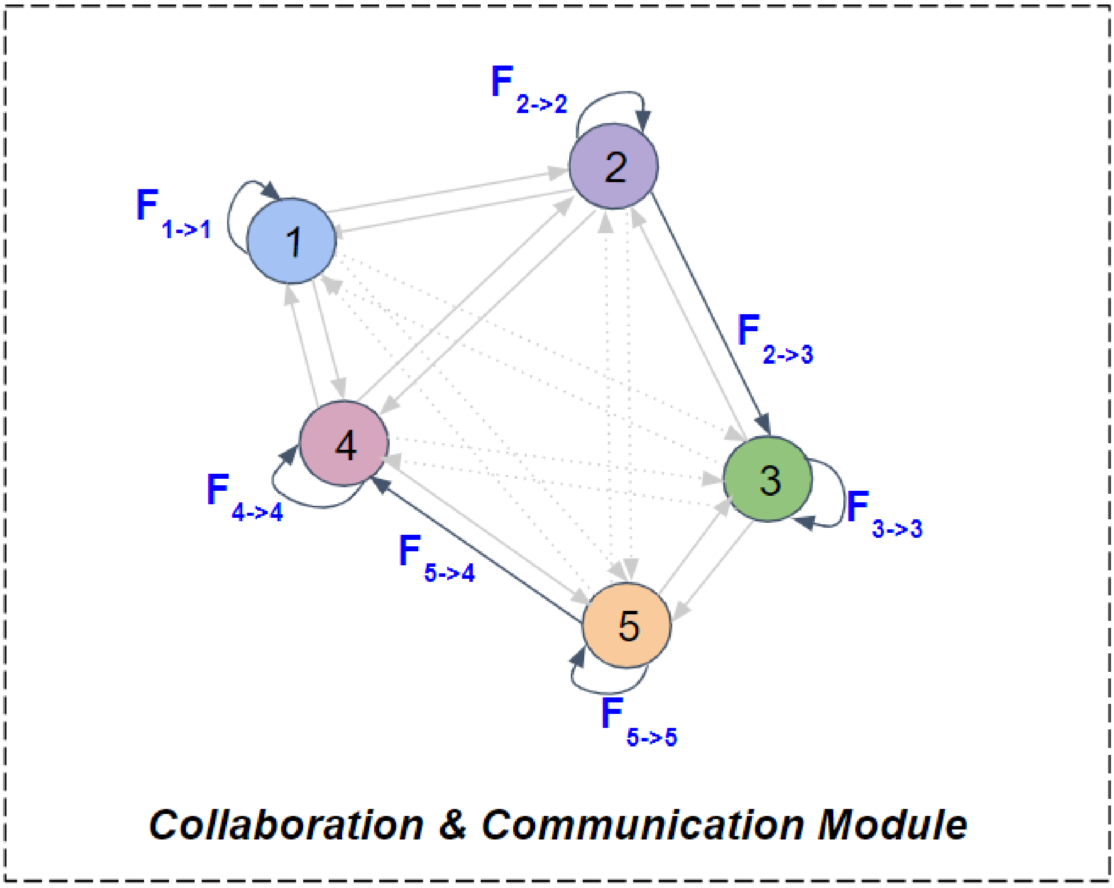

Figure 3.3: **Collaborative and communication graph.** Each node, $\{1, 2, 3, 4, 5\}$ represents one collaborative agent. Each edge $F_{i->j}$ is the transmitted feature from agent i to agent j when i is different from j and its own extracted feature if $i = j$. Using this collaborative graph, different levels of latency, communication interruption, and compression are simulated.

compress the channel dimension. Hence, $B_i = \text{Compress}(F^i_s)$, where B_i is compressed feature map of the i^{th} agent, which is subsequently transmitted to other agents.

3.3.4 Communication Interruption

Communication interruption is a critical factor that can significantly affect the performance of collaborative perception systems. Using the collaboration graph $G(V, E)$ shown in Figure 3.3, where V represents the agents and E the communication links between them, we introduce random interruptions in the communication links between agents to simulate the unreliability of real-world networks. To simulate communication interruptions, we randomly disable certain edges E between pairs of agents from all possible pairs (i, j), where each pair represents direct communication between two agents. This method allows us to examine the impact of network interruptions on the system's ability to collaboratively perceive the environment. By altering the number of disrupted edges E in various tests, we can test how effectively the CP system can operate amidst realistic communication interruptions.

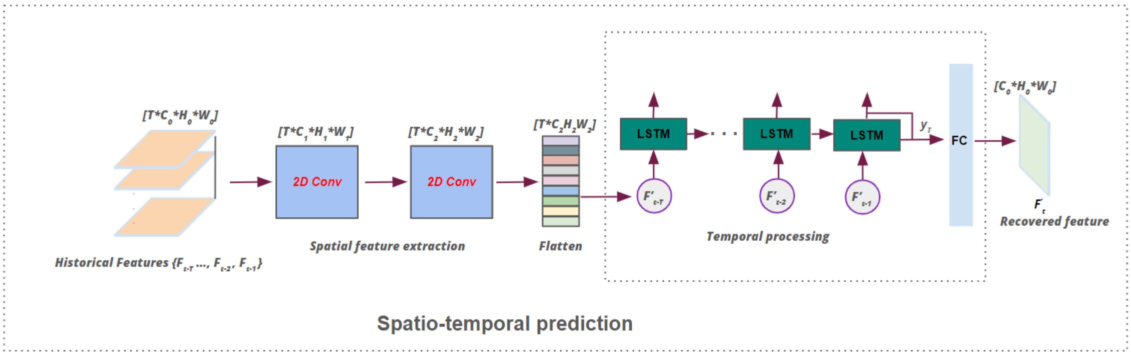

Figure 3.4: **Spatio-temporal prediction module for handling latency and communication interruption** – Historical features undergo sequential 2D convolution to extract spatial features, followed by LSTM layers to capture temporal dynamics and then passed through a fully connected layer, which ensures accurate feature recovery, compensating for any data loss due to communication limitations.

3.3.5 Latency

Latency significantly influences the performance of collaborative perception systems. In our model, we incorporate latency directly into the communication graph to assess its impact on data exchange between agents. Each edge $(i, j) \in E$ is associated with a latency τ_{ij}, which represents the delay encountered in the transmission of information from agent i to agent j. In doing so, we can analyze the impact of various latency scenarios on the overall effectiveness of the CP system, evaluating how well the system can maintain detection accuracy when there is a delay.

3.3.6 Recovery using spatio-temporal prediction module

Under ideal conditions, the aggregate feature at node i at time t, denoted F_t^a, is computed by fusing features from node j to node i, where j is an element of the set K containing all collaborating nodes. i.e.

$$F_t^a = \phi_{fuse}(\{F_t^{j\to i}\}) \quad \text{for } i, j \in K \tag{3.1}$$

Communication interruptions and latency occur when node j is unable to send or sends delayed features to node i, rendering the data $F_t^{j\to i}$ unusable at time t. In this case, the missing information is estimated from the historical feature information through the missing information recovery process.

$$\hat{F}_t^{j\to i} = \text{predict}(F_{(t-T)}^{j\to i}, F_{(t-T+1)}^{j\to i}, \dots, F_{(t-1)}^{j\to i}) \tag{3.2}$$

Where $\hat{F}_{j\to i}^t$ denotes the recovered feature at node i coming from node j at time t, using the spatio-temporal prediction module based on the features from the previous T timesteps.

This process predicts the current state of the features from accumulated past T historical information. Figure 3.4 illustrates the architecture used for spatio-temporal feature prediction. Historical features $F_{t-T}, \dots, F_{t-1}$ are input to a series of 2D convolutional layers to first extract spatial patterns in the feature space. Then, the sequence of flattened features is processed through LSTM layers. The output from the final LSTM layer, denoted as y_T, is passed to a fully connected (FC) layer, which maps it to the recovered feature space.

3.4 Experiments

3.4.1 Dataset

V2X-Sim [48]. V2X-Sim is a synthetic dataset designed for collaborative perception in autonomous driving in V2X scenarios. It is generated with CARLA and SUMO co-simulation to simulate realistic scenarios [54]. It includes 100 scenes, each lasting 20 seconds with recordings at 5Hz. Each scene includes 100 frames. The dataset provides synchronized sensor data from multiple vehicles and one road-side unit (RSU) with a maximum of five collaborating vehicles per scene. Each agent records LiDAR points with annotation for detection, tracking, and segmentation tasks.

Dataset split. We used the V2X-Sim version 2.0 split which comprises 10,000 synchronized total frames, divided into 8,000 for training, and 1,000 each for validation and testing. In total, the dataset contains 37,200 training samples and 5,000 samples each for validation and testing.

3.4.2 Benchmark models studied

We studied the impact of communication limitations on early, intermediate, and late fusion models with the single agent model as a baseline.

Single-agent: This baseline model processes point-cloud data independently for each agent without any collaborative inputs.

When2com [47]: This technique introduces an attention-based system to determine the formation of communication groups and when to communicate it with focus on minimizing bandwidth usage while maximizing the perception performance.

V2VNet [22]: V2VNet uses a convolutional neural network to generate and transmit compressed intermediate representations of LiDAR data, which are then fused using a

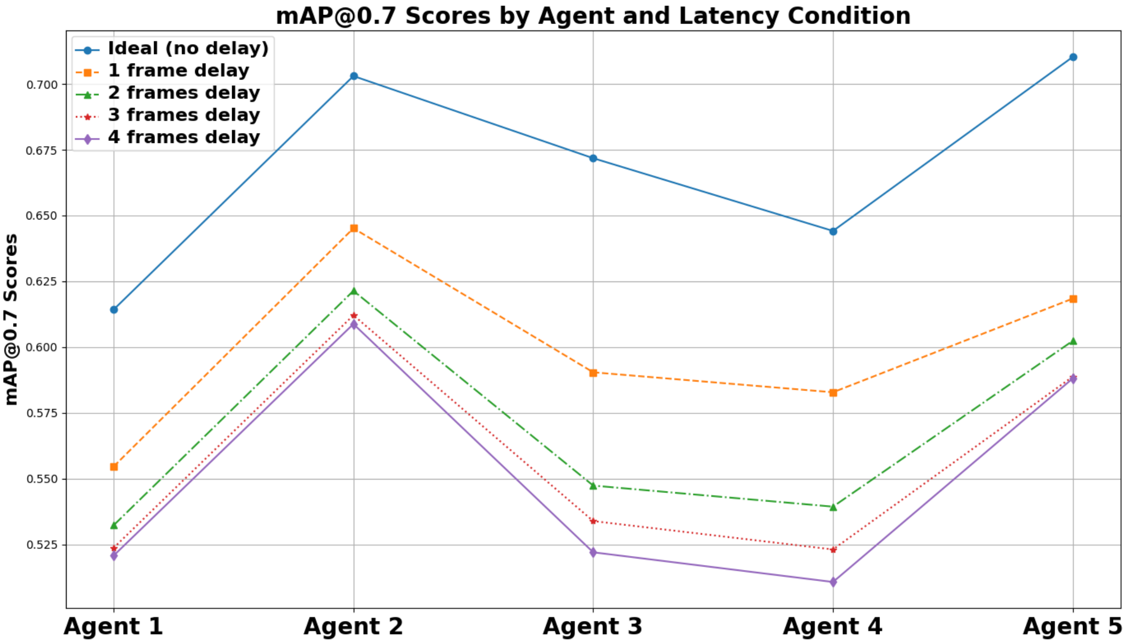

Figure 3.5: **Impact of latency on collaborative detection**, showcasing the detection performance for five agents under varying latency conditions.

spatially-aware graph neural network.
DiscoNet [10]: This method constructs a directed collaboration graph with matrix-valued weights on the edges, which extracts useful spatial areas using a knowledge distillation learning mechanism. During inference, only the small student model is used for prediction.
Late Fusion: In this approach, the final results of individual agents are combined and shared with each other.

3.4.3 Metrics

We quantify the detection performance using the mean average precision at a given intersection over the union (IoU) threshold. For instance, for a 0.7 *IoU* threshold,

$$\text{mAP@IoU=0.7} = \frac{1}{N}\sum_{i=1}^{N} \text{AP}_i\text{@}(IoU = 0.7) \tag{3.3}$$

Where $\text{AP}_i\text{@}(IoU = 0.7)$ represents the average precision for the i-th class among a total of N classes. The IoU threshold of 0.7 means that for a detection to be considered true positive, the overlap between the predicted bounding box and the ground truth bounding box must be at least 70%.

3.4.4 Results and Discussions

Impact of latency

Figure 3.5 shows the mAP@IoU=0.7 scores for five distinct agents on detection performance for different latency scenarios, ranging from the ideal case with no delay to a maximum of four frames of delay, $\tau_{ij} = \{1, 2, 3, 4\}$. A single frame delay represents 200 ms. There is an immediate and significant decrease in performance after a delay of one frame. As latency increases, a further, albeit smaller, decrease in performance is observed. This trend culminates in the lowest performance at a four-frame delay, indicating that increased delays disrupt collaborative detection capabilities.

Impact of communication interruption

The impact of communication interruption is illustrated in Figure 3.6. As the number of agents experiencing communication interruptions increases, the corresponding mAP decreases for each of the five agents.

This suggests that the effectiveness of the CP system is notably sensitive to interruption in the collaboration graph. All agents show a steep decline in mAP even with a single interruption, highlighting its dependence on uninterrupted data flow. The detection performance gets worse when a second agent is disconnected from the graph.

Effect of compression

Figure 3.7 illustrates the impact of data compression on the performance of agents in a collaborative detection system. Increasing compression from 2x to 16x leads to a clear decrease in mAP, with some agents such as Agent 4 and Agent 5 being more adversely affected than others. This highlights a trade-off between compression for efficient communication

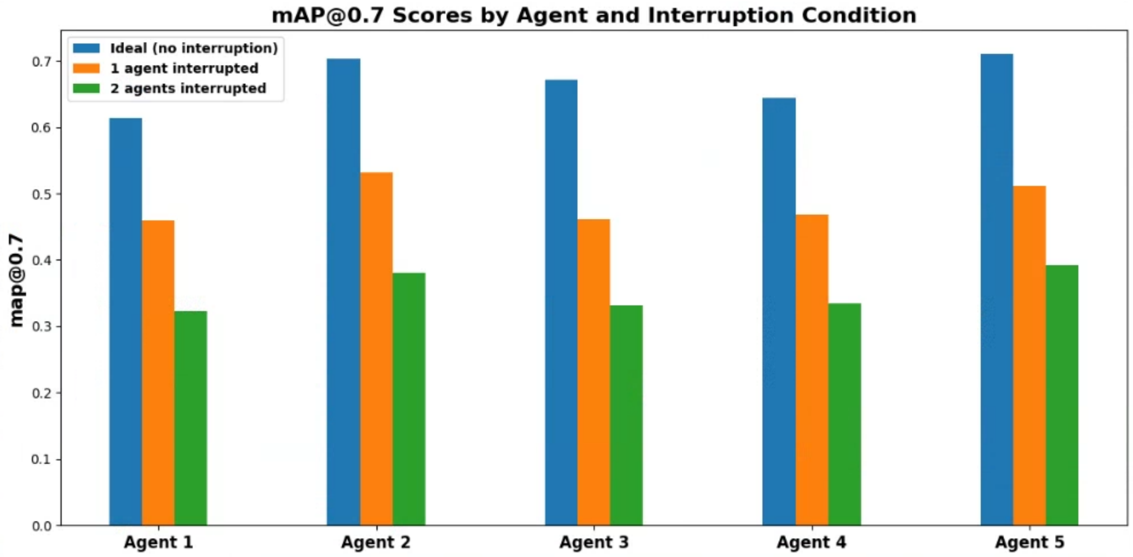

Figure 3.6: **Communication interruption effect on collaborative perception**. Comparison of the performance of agents operating without interruptions where one or two agents are not part of the collaborative group. The columns represent per agent detection as the levels of interruptions increases.

Table 3.1: Performance Comparison between single-agent baseline and collaborative methods under interruption (**Inter.**) and latency (**Lat.**)

	mAP@IoU=0.50			**mAP@IoU=0.70**		
Single Agent	0.47			0.42		
Fusion Type	**Ideal**	**Inter.**	**Lat.**	**Ideal**	**Inter.**	**Lat.**
Late Fusion	0.58	0.43	0.39	0.54	0.39	0.34
When2com [47]	0.48	0.31	0.40	0.41	0.25	0.35
V2VNet [22]	0.72	0.51	0.64	0.65	0.47	0.56
DiscoNet [10]	0.73	0.37	0.65	0.66	0.35	0.56
DiscoNet + STP	-	**0.67**		-	**0.61**	

and perception fidelity, emphasizing the need for careful calibration of compression levels in collaborative perception tasks.

Table 3.1 presents the performance of different collaborative methods under ideal communication conditions, 400ms latency, and communication interruption of two agents. Under ideal conditions, all collaborative methods outperform the single-agent baseline, with early fusion achieving the highest mAP values. However, the introduction of latency and communication interruptions causes notable performance degradation across all methods. The spatiotemporal prediction method with DiscoNet [10] (DiscoNet + STP) demonstrates resilience, maintaining higher mAP values compared to other approaches in both latency and interruption scenarios, highlighting its robustness in non-ideal communication environments.

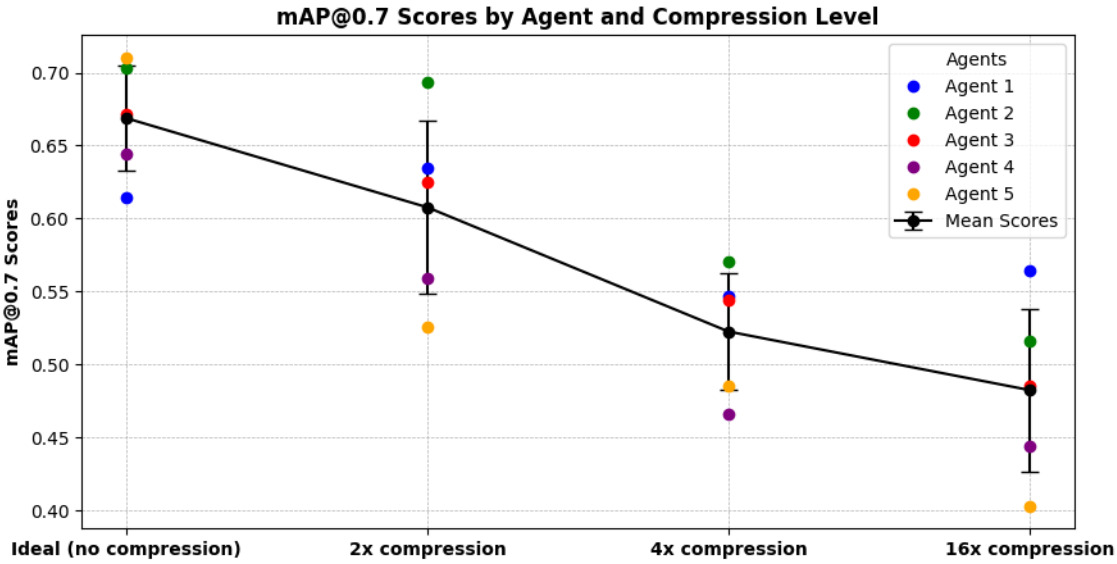

Figure 3.7: **Collaborative perception under different compression level**: $mAP@IoU = 0.7$ for five different agents under varying levels of feature compression. The compression levels evaluated include no compression (ideal scenario), 2x, 4x, and 16x compression, showing the trade-off between feature size and detection performance.

Qualitative discussion

In Figure 4.2, we present a qualitative evaluation of latency and communication interruption on collaborative detection performance.

Ideal communication. For comparison, Figure 8(a) shows the ideal communication scenario. In this case, the agents exchange uninterrupted and synchronized perception results, allowing for accurate detection and minimal false or missed detections. The predicted boxes align well with the ground-truth boxes, indicative of a high-confidence consensus among the agents. This scenario is used as a reference to compare the impact of latency and interruption.

Latency. Figure 4.2(b) shows the performance of the method when there is a delay in collaboration between agents. Latency leads to an increase in false positives where agents incorrectly identify objects based on outdated information and negative detections, where current objects are missed due to the absence of timely data exchange indicating that timely data sharing is critical for maintaining system performance. In addition, there is a misalignment in detection as agents are unable to accurately reconcile the temporal disparity.

Interruption The effect of communication interruption is shown in Figure 4.2(c). There are more missed detections compared to the ideal-case scenario shown in Figures 4.2(a) and even 4.2(b) with latency, as the lack of data exchange yields a considerable increase in missed detections, and agents are unable to compensate for the information void. However, communication interruption shows fewer false detections compared to fusion with latency in as 4.2(b), implying that the absence of information can be less detrimental than inaccurate information in certain contexts.

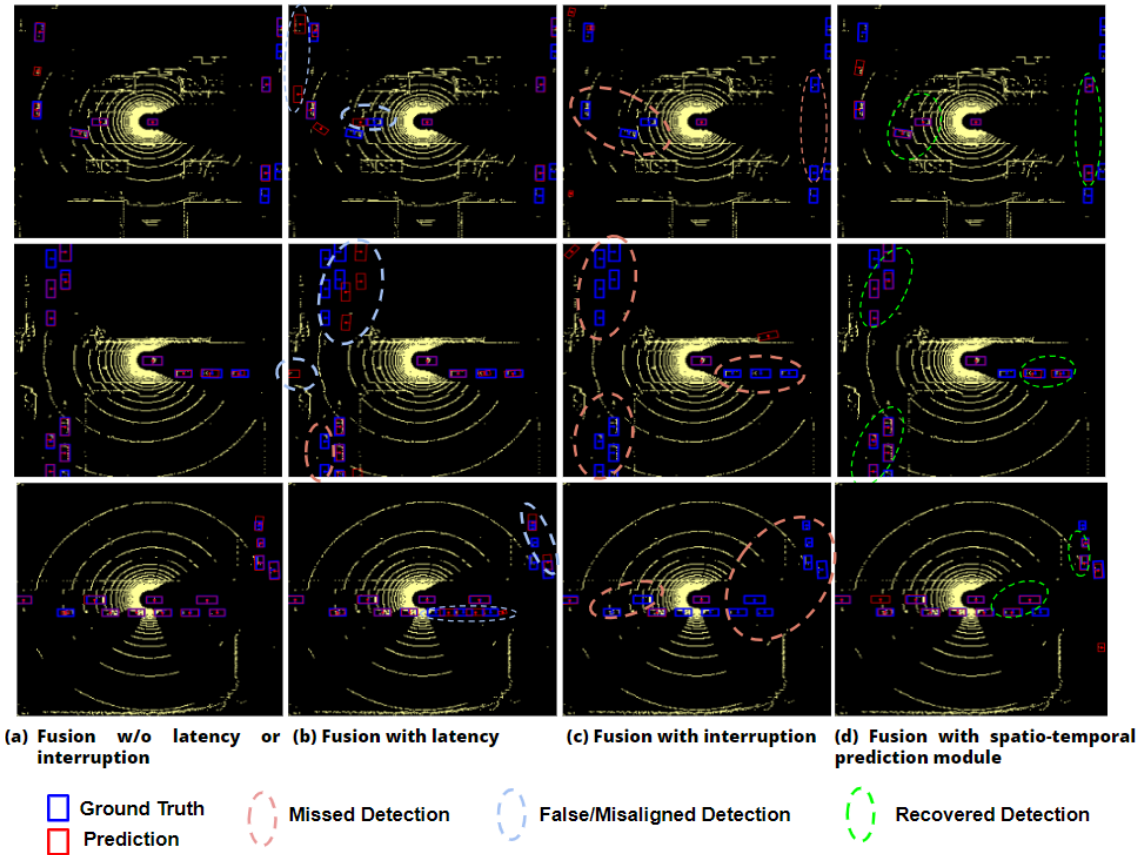
(a) Fusion w/o latency or interruption
(b) Fusion with latency
(c) Fusion with interruption
(d) Fusion with spatio-temporal prediction module

Figure 3.8: **Visualization of the effects of latency and communication interruptions on detection**. Blue and red boxes represent ground truth and predictions, respectively. Different rows represent different scenes. (a) displays results with uninterrupted and delay free communication between agents; (b) demonstrates the detection degradation due to latency; (c) highlights the impact of communication interruption; and (d) presents the detection recovery through the spatio-temporal prediction network.

Information recovery via prediction

Figure 4.2(d) demonstrates the benefit of handling latency and interruption using spatio-temporal prediction method. The recovery process helped align the detections that are misaligned due to latency. In addition, storing historical frames and adding the prediction module during interruption enabled partially recovering some of the missed detection indicated by green dashed ellipses. The recovered detections highlight the module's ability in utilizing historical and contextual data to ensure reliability and robustness in collaborative detection systems, compensating for temporal and spatial data loss caused by communication impacts.

3.5 CONCLUSION

This chapter has examined collaborative perception systems in the context of communication challenges, focusing on the effects of latency, communication interruption, and bandwidth limitations. Our numerical experiments, conducted using the V2X-Sim dataset, reveal significant performance degradation under non-ideal conditions: latency of 200ms (one frame delay) caused a significant decrease in mAP, while communication interruptions led to a steep decline in performance even with a single agent disconnected. Compression for saving bandwidth shows a clear trade-off between communication efficiency and detection accuracy, with 16x compression significantly reducing mAP. The proposed spatio-temporal prediction (STP) method demonstrates resilience, achieving an mAP of 0.67 and 0.61 at IoU thresholds of 0.5 and 0.7 respectively under non-ideal conditions, outperforming other methods including DiscoNet (0.65 and 0.56) and V2VNet (0.64 and 0.56). These results underscore the critical need for robust and adaptive algorithms in collaborative perception systems that can maintain performance under varying real-world communication conditions. Future work should focus on further improving the resilience of the system to communication challenges and validating these approaches in diverse real-world scenarios, particularly for applications that involve vulnerable road users.

Chapter 4

Vision Language Model For Pedestrian Trajectory Estimation

4.1 Introduction

Accurate prediction of pedestrian trajectories is crucial for pedestrian safety. As autonomous vehicles become more prevalent, the ability to anticipate and respond to pedestrian movements has become a critical challenge in ensuring the safety of pedestrians. Pedestrian trajectory prediction presents inherent complexities stemming from the dynamic nature of urban environments and the diverse, often unpredictable behavior of pedestrians. The task requires interpreting subtle visual cues and contextual information while meeting the demands of real-time processing in safety-critical situations. These factors collectively contribute to the challenge of developing accurate and reliable prediction models for autonomous driving systems.

Earlier pedestrian trajectory prediction methods have relied on recurrent neural networks (RNNs) [55] and long-short-term memory (LSTM) networks [56] to process temporal data for this task. These approaches have shown promise in capturing sequential patterns in pedestrian movements. However, they often struggle to fully incorporate the rich visual and contextual information present in real-world scenarios. More recent methods have explored the use of advanced architectures to use both the context from the video frames and ego-vehicle attributes. For instance, [57] proposed a future person localization method for first-person videos, [58] developed an egocentric vision-based future vehicle localization system for intelligent driving assistance. These approaches have made significant strides in improving prediction accuracy, but still face challenges in integrating diverse sources of information and reasoning about complex scenarios.

Recent advances in vision language models (VLMs) offer new possibilities to improve the accuracy, interpretability, and robustness of pedestrian trajectory prediction. VLMs excel in joint visual-textual understanding, providing rich pre-trained representations and enhanced context interpretation through multi-modal reasoning. To leverage these advantages, we propose a vision language reasoning approach on the Pedestrian Intention Estimation (PIE) [59] benchmark. This novel approach, dubbed PieVLM (Pedestrian Intention and trajectory Estimation using Vision Language Model), harnesses the power of VLMs for pedestrian trajectory estimation. PieVLM aims to address current limitations in the following ways:

- Utilize VLMs' capability to process complex visual and semantic information jointly
- Incorporate textual descriptions and annotations seamlessly into the prediction process

- Enhance context understanding through multi-modal reasoning
- Offer more interpretable predictions through natural language explanations
- Improve the overall accuracy and robustness of pedestrian trajectory estimation

This chapter introduces the PieVLM architecture and demonstrates its potential to enhance pedestrian safety in autonomous driving scenarios.

4.2 PieVLM: Vision Language Model for Pedestrian Trajectory Prediction

PieVLM leverages vision language models (VLMs) for enhanced pedestrian trajectory prediction through two main techniques. The first employs a two-stage approach: pre-training with visual-linguistic supervision followed by fine-tuning for trajectory prediction, enabling rich contextual understanding. The second explores end-to-end prediction by framing the task as image-text to text, directly utilizing VLMs' language modeling capabilities. These methods aim to improve prediction accuracy and interpretability in autonomous driving scenarios by integrating advanced language understanding with spatial reasoning.

4.2.1 Pre-training followed by Task-specific Finetuning

The pre-training approach leverages Paligemma [60], a large-scale vision-language model, for pedestrian trajectory prediction through a two-stage process. Paligemma, which combines the SigLIP [61] vision encoder and the Gemma-2B [62] language model, is designed to handle a variety of tasks through a simple image-text in, text out approach. We begin by further pretraining the Paligemma on a dataset of scenes with pedestrians as illustrated in the upper part of Figure 4.1. This pretraining stage utilizes image-text pairs where the text describes the pedestrian's location, actions, and other attributes, enhancing Paligemma's understanding of pedestrian behavior in urban contexts. Following this, we fine-tune the pre-trained model specifically for trajectory prediction.

During fine-tuning, we provide sequences of images and corresponding text descriptions as input, training the model to generate accurate trajectory predictions as output after the features are processed using temporal model. This two-stage process allows us to adapt Paligemma's powerful vision-language capabilities to the specific task of pedestrian trajectory prediction, potentially improving both accuracy and interpretability of the predictions in complex urban environments.

4.2.2 End-to-end Trajectory Prediction With VLM

For the end-to-end PieVLM, we adopt Florence-2 [63], a powerful vision-language model that advances unified representations across vision and language tasks. Florence-2 is designed to handle a variety of tasks through a prompt-based sequence-to-sequence framework, with robust pre-trained features leveraging a vast dataset of 5.4 billion visual annotations for pre-training. This approach frames pedestrian trajectory prediction as a text-to-text task, leveraging the model's ability to process and integrate both visual and textual inputs. The

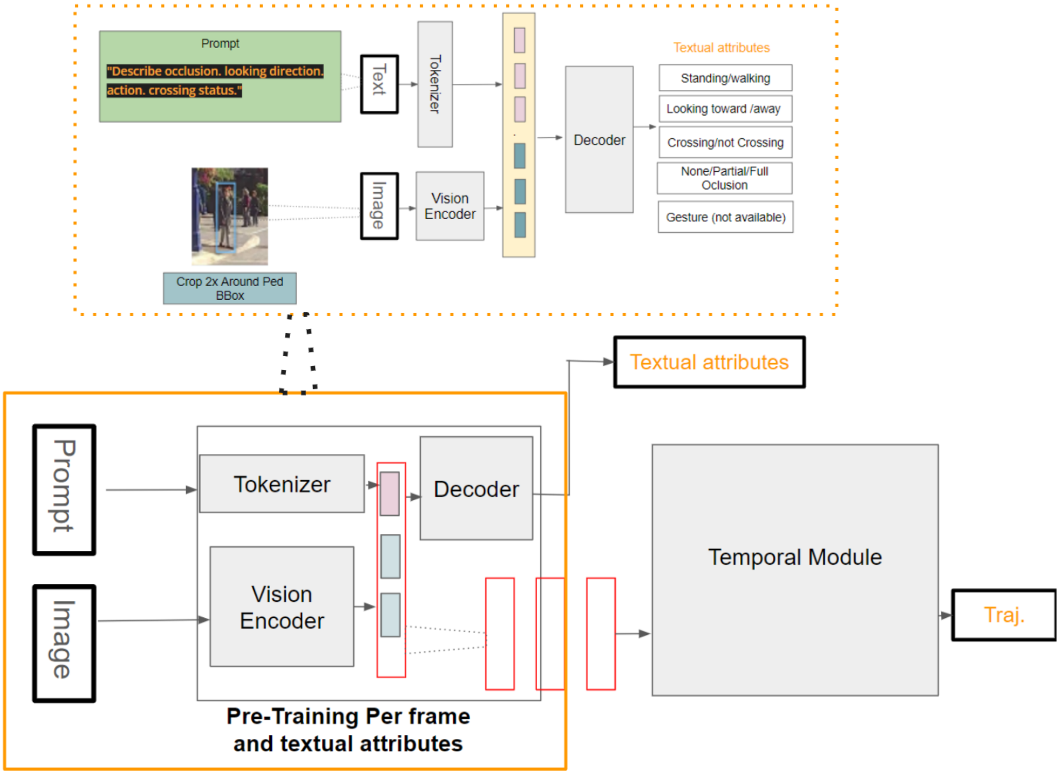

Figure 4.1: Two stage PieVLM architecture for pedestrian trajectory prediction. The upper section illustrates the pre-training phase, where image-text pairs are processed to learn pedestrian attributes and contexts. The lower section shows the fine-tuning stage, integrating pre-trained features with a temporal module to predict trajectories.

input to the model consists of two primary components that provide a rich, multimodal representation of the scene and pedestrian behavior.

The first component is a structured textual prompt that encapsulates detailed information about the scene and pedestrian status. This prompt begins with a frame identifier, such as "<PIE_PREDICT> Frame 1:", followed by the pedestrian's current location encoded as a series of coordinates, for example, "<loc_656><loc_681><loc_670><loc_766>". The prompt also includes critical contextual information such as the pedestrian's occlusion status (e.g., "is full occluded from ego vehicle view"), relevant traffic elements like traffic light locations and states (e.g., " <loc_531><loc_591><loc_539><loc_618> Type: regular State: green"), and additional frames of pedestrian location data to capture temporal dynamics.

The second input component is an image frame that provides the visual context of the scene. This image is processed through a Vision Encoder, which extracts relevant visual features that complement the textual information. The image typically shows the pedestrian and their immediate surroundings, offering visual cues that may influence trajectory prediction.

Florence-2 processes these inputs through several sophisticated stages. Initially, the textual prompt undergoes tokenization and embedding, transforming the structured text into a format the model can efficiently process. Concurrently, the Vision Encoder converts the input image frame into a rich set of visual features. The model then combines these multimodal inputs – the embedded text and visual features – leveraging its deep architecture to interpret the complex relationships between textual descriptions and visual cues.

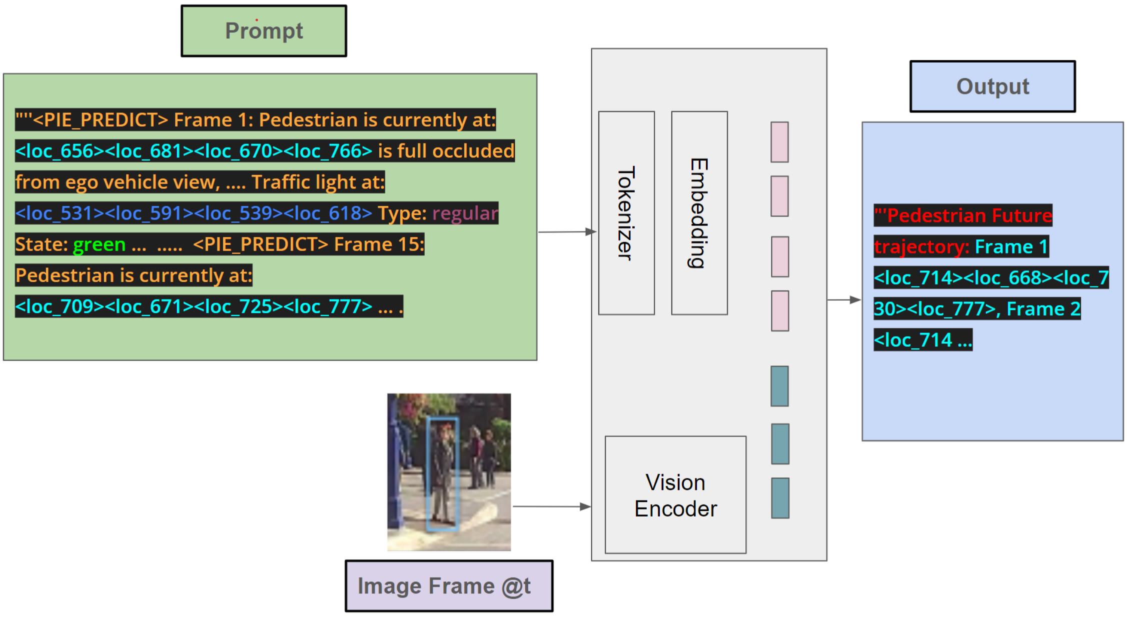
Prompt
""<PIE_PREDICT> Frame 1: Pedestrian is currently at:
<loc_656><loc_681><loc_670><loc_766> is full occluded
from ego vehicle view, ... Traffic light at:
<loc_531><loc_591><loc_539><loc_618> Type: regular
State: green ... . <PIE_PREDICT> Frame 15:
Pedestrian is currently at:
<loc_709><loc_671><loc_725><loc_777> ... .
Tokenizer
Embedding
Vision Encoder
Image Frame @t
Output
"Pedestrian Future
trajectory: Frame 1
<loc_714><loc_668><loc_7
30><loc_777>, Frame 2
<loc_714 ...

Figure 4.2: End-to-End PieVLM architecture for pedestrian trajectory prediction. The system integrates structured text prompts (top left) containing spatial and contextual information with image frames (bottom left). These inputs are processed through a vision-language model comprising a tokenizer, embedding layer, and vision encoder which are then concatenated. The model then generates predictions of future pedestrian trajectories (right), after the fusion of textual and visual features.

The output of this end-to-end process is generated in a structured text format, predicting the "Pedestrian Future Trajectory" across multiple frames. Each frame in the prediction includes the expected future location coordinates of the pedestrian, formatted similarly to the input (e.g., "<loc_714><loc_668><loc_730><loc_777>"). This format allows for precise spatial predictions while maintaining the text-to-text paradigm.

By framing trajectory prediction in this manner, PieVLM can directly map from rich, multimodal inputs to detailed trajectory predictions. This approach potentially captures intricate relationships between visual elements, textual descriptions of the scene, and future pedestrian movements that might be challenging to model using traditional computer vision techniques alone. The end-to-end nature of this method, combined with the powerful Florence-2 architecture, offers a novel and potentially more nuanced approach to understanding and predicting pedestrian movements in complex urban environments.

4.3 Numerical Experiments

4.3.1 Datasets

For this study, the Pedestrian Intention Estimation (PIE) dataset [59] is used. The PIE dataset contains over 6 hours of egocentric driving videos, comprising 911k frames from six urban locations. It features annotations for 300k frames, including 2.1 million bounding boxes for 1,842 unique pedestrian samples, as well as annotations for vehicles, traffic lights, and signs. Each frame provides detailed spatial, temporal, and behavioral information, including pedestrian actions, attributes, and occlusion levels. The dataset also includes synchronized ego-vehicle movement data and pedestrian intention probability annotations, ranging from 0 to 1, enhancing its utility for predictive modeling in autonomous driving applications.

4.3.2 Metrics

We present the results for a 0.5-second past observation period and a 1-second future prediction, corresponding to 15 past frames and 30 future frames at a rate of 30 frames per second. The trajectory prediction metrics, all reported in pixels, are as follows:

- **Average Displacement Error (ADE)**: The average Euclidean distance between predicted and ground truth centers over all prediction time steps.

  $$ADE = \frac{1}{T}\sum_{t=1}^{T}\sqrt{(x_t^p - x_t^g)^2 + (y_t^p - y_t^g)^2} \tag{4.1}$$

  where (x_t^p, y_t^p) is the predicted center position of the bounding box containing the pedestrian and (x_t^g, y_t^g) is the ground truth center position at time step t, and T is the total number of prediction time steps.

- **Final Displacement Error (FDE)**: The Euclidean distance between the predicted final center position and the ground truth final center position.

  $$FDE = \sqrt{(x_T^p - x_T^g)^2 + (y_T^p - y_T^g)^2} \tag{4.2}$$

  where T is the final prediction time step.

- **Average Rotated Bbox (ARB)**: The average RMSE of bounding box coordinates over all prediction time steps.

  $$ARB = \frac{1}{T}\sum_{t=1}^{T}\sqrt{\frac{1}{4}\sum_{i=1}^{4}[(x_{t,i}^p - x_{t,i}^g)^2 + (y_{t,i}^p - y_{t,i}^g)^2]} \tag{4.3}$$

  where $(x_{t,i}^p, y_{t,i}^p)$ and $(x_{t,i}^g, y_{t,i}^g)$ are the predicted and ground truth coordinates of the i-th corner of the bounding box at time step t, respectively.

- **Final Rotated Bbox (FRB)**: The RMSE of bounding box coordinates at the final prediction time step.

  $$FRB = \sqrt{\frac{1}{4}\sum_{i=1}^{4}[(x_{T,i}^p - x_{T,i}^g)^2 + (y_{T,i}^g - y_{T,i}^g)^2]} \tag{4.4}$$

  where $(x_{T,i}^p, y_{T,i}^p)$ and $(x_{T,i}^g, y_{T,i}^g)$ are the predicted and ground truth coordinates of the i-th corner of the bounding box at the final time step T, respectively.

All metrics are computed based on pixel coordinates, with lower values indicating better performance.

4.4 Preliminary Results

Initial experiments with PieVLM have shown promising results compared to baseline methods, although comprehensive evaluations are still ongoing. Using the pre-training followed by the finetuining method (PieVLM-I) as indicated in Table 4.1 shows promising result in accurately predicting the pedestrian's trajectory. The Florence-2 version (PieVLM-II) demonstrated an Average Displacement Error (ADE) of 15.42 pixels and Final Dislacement Error (FDE) of 35.84 pixels for trajectory prediction, suggesting competitive performance in estimating pedestrian movements as shown in Table 4.1. Although these initial results are encouraging, more metrics and more extensive evaluations are needed to fully assess the performance of PieVLM across various scenarios and compared to state-of-the-art methods.

Table 4.1: Trajectory Prediction Results on PIE Dataset

Method	ADE	FDE	ARB	FRB
FOL [58]	73.87	164.53	78.16	143.69
FPL [57]	56.66	132.23	-	-
B-LSTM [56]	27.09	66.74	37.41	75.87
PIE$_{traj}$ [59]	21.82	53.63	27.16	55.39
PIE$_{full}$ [59]	19.50	45.27	24.40	49.09
BiPed [64]	15.21	35.03	19.62	39.12
PedFormer [65]	13.08	30.35	15.27	32.79
PieVLM-I	18.82	60.39	35.16	67.17
PieVLM-II	15.42	35.84	21.11	47.21

4.5 Conclusion and Future Work

PieVLM represents a novel approach to pedestrian trajectory estimation by leveraging the power of vision language models. The use of pre-trained VLMs, combined with task-specific fine-tuning and temporal modeling, shows promise in capturing complex visual and contextual information for more accurate predictions. The preliminary results suggest that this approach has the potential to advance the state of the art in pedestrian trajectory prediction. An end-to-end approach where visual-linguistic input and text output are used has also shown a promising result in improving pedestrian trajectory prediction.

Here's the revised version with the dataset link as a hyperlink:

PUBLICATION AND PRODUCT

Paper: Shenkut, D. Vijaya Kumar, B.V.K. Impact of Latency and Bandwidth Limitations on the Safety Performance of Collaborative Perception (2024 IEEE International Conference on Computer Communications and Networks (ICCCN)): URL Pending

Dataset: CVIPS Dataset, check dataset section

Code: `https://github.com/cvips/cvips`

REFERENCES

[1] Governors Highway Safety Association. *Pedestrian Traffic Fatalities by State: 2022 Preliminary Data*. Tech. rep. Accessed: July 31, 2023. 2022. URL: https://www.ghsa.org/resources/Pedestrians23.

[2] United States Department of Transportation, Federal Highway Administration. *The Safe System Paradigm: Reducing Fatalities and Injuries at the Nation's Intersections*. Tech. rep. Accessed July 22, 2023. 2023. URL: https://highways.dot.gov/public-roads/winter-2022/03.

[3] H. Caesar et al. "nuScenes: A Multimodal Dataset for Autonomous Driving". In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020, pp. 11618–11628.

[4] S. Liu et al. "Towards Vehicle-to-everything Autonomous Driving: A Survey on Collaborative Perception". In: *arXiv preprint arXiv:2308.16714* (2023). URL: https://arxiv.org/abs/2308.16714.

[5] T. Wang et al. "DeepAccident: A Motion and Accident Prediction Benchmark for V2X Autonomous Driving". In: *arXiv* (2023). URL: https://arxiv.org/abs/2304.01168.

[6] Y. Hu et al. "Collaboration Helps Camera Overtake LiDAR in 3D Detection". In: *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2023.

[7] D. Qiao and F. Zulkernine. "Adaptive Feature Fusion for Cooperative Perception using LiDAR Point Clouds". In: *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. 2023, pp. 1186–1195.

[8] Y. Ma et al. "MACP: Efficient Model Adaptation for Cooperative Perception". In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. Jan. 2024, pp. 3373–3382.

[9] Y. Han et al. "Collaborative Perception in Autonomous Driving: Methods, Datasets, and Challenges". In: *IEEE Intelligent Transportation Systems Magazine* 15.6 (Nov. 2023), pp. 131–151. URL: http://dx.doi.org/10.1109/MITS.2023.3298534.

[10] Y. Li et al. "Learning Distilled Collaboration Graph for Multi-Agent Perception". In: *Thirty-fifth Conference on Neural Information Processing Systems (NeurIPS 2021)*. 2021.

[11] S. Ashtekar, S. Dhalwar, and A. Pasupathy. "Computer Vision Based Vulnerable Road Users Hand Signal Recognition". In: *IEEE International Conference on Intelligent and Robust Computing and Applications*. Sept. 2021.

[12] M. Goldhammer et al. "Intentions of Vulnerable Road Users—Detection and Forecasting by Means of Machine Learning". In: *IEEE Transactions on Intelligent Transportation Systems* (Mar. 2018).

[13] H. Cheng et al. "Interaction Detection Between Vehicles and Vulnerable Road Users: A Deep Generative Approach with Attention". In: *arXiv preprint arXiv:2105.03891* (May 2021).

[14] D. Charlebois et al. "The ideal vulnerable road user – a study of parameters affecting VRU detection". In: *Traffic Injury Prevention* (Apr. 2023).

[15] A. Aparicio et al. *Advancing active safety towards the protection of vulnerable road users: the prospect project*. Tech. rep. PROSPECT Project.

[16] V. Ruiz Garate, R. Bours, and K. Kietlinski. "Numerical modeling of ADA system for vulnerable road users protection based on radar and vision sensing". In: *IEEE Intelligent Vehicles Symposium*. June 2012.

[17] S. Y. Gelbal, B. Aksun-Guvenc, and L. Guvenc. "Vulnerable Road User Safety Using Mobile Phones with Vehicle-to-VRU Communication". In: *Electronics* (Jan. 2024).

[18] P. Teixeira et al. "A Sensing, Communication and Computing Approach for Vulnerable Road Users Safety". In: *IEEE Access* ().

[19] M. Karoui, V. Berg, and S. Mayrargue. "Assessment of V2X Communications For Enhanced Vulnerable Road Users Safety". In: *IEEE Vehicular Technology Conference*. June 2022.

[20] S. C. Lobo, A. Festag, and C. Facchi. "Enhancing the Safety of Vulnerable Road Users: Messaging Protocols for V2X Communication". In: *IEEE Vehicular Technology Conference*. Sept. 2022.

[21] Y. Lu et al. "Robust collaborative 3D object detection in presence of pose errors". In: *2023 IEEE International Conference on Robotics and Automation (ICRA)*. 2023, pp. 4812–4818.

[22] T.-H. Wang et al. "V2VNet: Vehicle-to-Vehicle Communication for Joint Perception and Prediction". In: (2020). URL: https://arxiv.org/abs/2008.07519.

[23] J. Tu et al. "Adversarial Attacks On Multi-Agent Communication". In: (2021). URL: https://arxiv.org/abs/2101.06560.

[24] R. Xu et al. "OPV2V: An Open Benchmark Dataset and Fusion Pipeline for Perception with Vehicle-to-Vehicle Communication". In: *2022 IEEE International Conference on Robotics and Automation (ICRA)*. 2022.

[25] R. Xu et al. "V2X-ViT: Vehicle-to-Everything Cooperative Perception with Vision Transformer". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2022.

[26] Z. Lei et al. "Latency-Aware Collaborative Perception". In: (2022). URL: https://arxiv.org/abs/2207.08560.

[27] Y. Hu et al. “Where2comm: Communication-Efficient Collaborative Perception via Spatial Confidence Maps”. In: *Thirty-sixth Conference on Neural Information Processing Systems (Neurips)*. Nov. 2022.

[28] J. Cui et al. “Coopernaut: End-to-End Driving with Cooperative Perception for Networked Vehicles”. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022.

[29] R. Xu et al. “Bridging the Domain Gap for Multi-Agent Perception”. In: *arXiv preprint arXiv:2210.08451* (2023).

[30] Li Xiang et al. *DI-V2X: Learning Domain-Invariant Representation for Vehicle-Infrastructure Collaborative 3D Object Detection*. 2023. arXiv: 2312.15742 [cs.CV]. URL: https://arxiv.org/abs/2312.15742.

[31] D. Qiao and F. Zulkernine. “Adaptive Feature Fusion for Cooperative Perception using LiDAR Point Clouds”. In: (2023). URL: https://arxiv.org/abs/2208.00116.

[32] X. Kong et al. “DUSA: Decoupled Unsupervised Sim2Real Adaptation for Vehicle-to-Everything Collaborative Perception”. In: *Proceedings of the 31st ACM International Conference on Multimedia*. 2023, pp. 1943–1954.

[33] R. Chen et al. “CO^3: Cooperative Unsupervised 3D Representation Learning for Autonomous Driving”. In: *arXiv preprint arXiv:2206.04028* (2022).

[34] T. Wang et al. “UMC: A Unified Bandwidth-efficient and Multi-resolution based Collaborative Perception Framework”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2023.

[35] K. Yang et al. “Spatio-Temporal Domain Awareness for Multi-Agent Collaborative Perception”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. Oct. 2023, pp. 23383–23392.

[36] B. Wang et al. “CORE: Cooperative Reconstruction for Multi-Agent Perception”. In: (2023). URL: https://arxiv.org/abs/2307.11514.

[37] H. Yu et al. “Flow-Based Feature Fusion for Vehicle-Infrastructure Cooperative 3D Object Detection”. In: *Advances in Neural Information Processing Systems*. 2023.

[38] MediaBrain SJTU et al. “Robust Asynchronous Collaborative 3D Detection via Bird’s Eye View Flow”. In: (2023). URL: https://openreview.net/forum?id=UHIDdtxmVS.

[39] AIR THU et al. “V2X-Seq: The Large-Scale Sequential Dataset for the Vehicle-Infrastructure Cooperative Perception and Forecasting”. In: (2023). URL: https://arxiv.org/abs/2305.05938.

[40] Jiayao Tan et al. *Dynamic V2X Autonomous Perception from Road-to-Vehicle Vision*. 2023. arXiv: 2310.19113 [cs.CV]. URL: https://arxiv.org/abs/2310.19113.

[41] R. Xu et al. “CoBEVT: Cooperative Bird’s Eye View Semantic Segmentation with Sparse Transformers”. In: *Conference on Robot Learning (CoRL)*. 2022.

[42] D. Qiao and F. Zulkernine. “CoBEVFusion: Cooperative Perception with LiDAR-Camera Bird’s-Eye View Fusion”. In: (2023). URL: https://arxiv.org/abs/2310.06008.

[43] H. Xiang, R. Xu, and J. Ma. “HM-ViT: Hetero-modal Vehicle-to-Vehicle Cooperative perception with vision transformer”. In: *arXiv preprint arXiv:2304.10628* (2023).

[44] D. Chen and P. Krähenbühl. "Learning from All Vehicles". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 17222–17231.

[45] S. Fan et al. "QUEST: Query Stream for Practical Cooperative Perception". In: *arXiv preprint arXiv:2308.01804* (2023). [Online].

[46] Y. Hu et al. "Collaboration Helps Camera Overtake LiDAR in 3D Detection". In: *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2023.

[47] Y.-C. Liu et al. "When2com: Multi-Agent Perception via Communication Graph Grouping". In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2020.

[48] Yiming Li et al. *V2X-Sim: Multi-Agent Collaborative Perception Dataset and Benchmark for Autonomous Driving*. 2022. arXiv: `2202.08449 [cs.CV]`. URL: `https://arxiv.org/abs/2202.08449`.

[49] Haibao Yu et al. *DAIR-V2X: A Large-Scale Dataset for Vehicle-Infrastructure Cooperative 3D Object Detection*. 2022. arXiv: `2204.05575 [cs.CV]`. URL: `https://arxiv.org/abs/2204.05575`.

[50] M. Yang, K. Jiang, J. Wen, et al. "Real-Time Evaluation of Perception Uncertainty and Validity Verification of Autonomous Driving". In: *Sensors* (2023). URL: `https://dx.doi.org/10.3390/s23052867`.

[51] Y. Lou, Q. Song, Q. Xu, et al. "Uncertainty-Encoded Multi-Modal Fusion for Robust Object Detection in Autonomous Driving". In: *arXiv* (2023). URL: `https://dx.doi.org/10.48550/arXiv.2307.16121`.

[52] Y. Zhou, L. Liu, H. Zhao, et al. "Towards Deep Radar Perception for Autonomous Driving: Datasets, Methods, and Challenges". In: *Sensors* (2022). URL: `https://dx.doi.org/10.3390/s22114208`.

[53] Y. Zhu, R. Xu, C. Tao, et al. "An Object Detection Method Based on Feature Uncertainty Domain Adaptation for Autonomous Driving". In: *Applied Sciences* (2023). URL: `https://dx.doi.org/10.3390/app13116448`.

[54] A. Dosovitskiy et al. "CARLA: An Open Urban Driving Simulator". In: *arXiv preprint arXiv:1711.03938* (2017).

[55] A. Rasouli, I. Kotseruba, and J. K. Tsotsos. "Pedestrian action anticipation using contextual feature fusion in stacked RNNs". In: *Proceedings of the British Machine Vision Conference (BMVC)*. 2019.

[56] A. Bhattacharyya, M. Fritz, and B. Schiele. "Long-term on-board prediction of people in traffic scenes under uncertainty". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018.

[57] T. Yagi et al. "Future person localization in first-person videos". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018.

[58] Y. Yao et al. "Egocentric vision-based future vehicle localization for intelligent driving assistance systems". In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. 2019.

[59] A. Rasouli et al. "PIE: A large-scale dataset and models for pedestrian intention estimation and trajectory prediction". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019.

[60] Lucas Beyer et al. *PaliGemma: A versatile 3B VLM for transfer*. 2024. arXiv: 2407.07726 [cs.CV]. URL: https://arxiv.org/abs/2407.07726.

[61] Xiaohua Zhai et al. *Sigmoid Loss for Language Image Pre-Training*. 2023. arXiv: 2303.15343 [cs.CV]. URL: https://arxiv.org/abs/2303.15343.

[62] Gemma Team et al. *Gemma: Open Models Based on Gemini Research and Technology*. 2024. arXiv: 2403.08295 [cs.CL]. URL: https://arxiv.org/abs/2403.08295.

[63] Bin Xiao et al. *Florence-2: Advancing a Unified Representation for a Variety of Vision Tasks*. 2023. arXiv: 2311.06242 [cs.CV]. URL: https://arxiv.org/abs/2311.06242.

[64] A. Rasouli, M. Rohani, and J. Luo. "Bifold and semantic reasoning for pedestrian behavior prediction". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021.

[65] Amir Rasouli and Iuliia Kotseruba. *PedFormer: Pedestrian Behavior Prediction via Cross-Modal Attention Modulation and Gated Multitask Learning*. 2022. arXiv: 2210.07886 [cs.CV]. URL: https://arxiv.org/abs/2210.07886.