

# Enhancing Perception Systems using Vehicle-To-Vehicle Enabled Sensor Fusion

Ansh S. Gwash

Thesis submitted to the Faculty of the  
Virginia Polytechnic Institute and State University  
in partial fulfillment of the requirements for the degree of

Master of Science

in

Computer Science and Applications

Tim Talty, Chair

Denis Gracanin, Co-chair

Na Meng

June 18, 2025

Blacksburg, Virginia

Keywords: Autonomous Vehicles, V2X, Connected Vehicles, Wireless

Copyright 2025, Ansh S. Gwash

# Enhancing Perception Systems using Vehicle-To-Vehicle Enabled Sensor Fusion

Ansh S. Gwash

(ABSTRACT)

With the surge in popularity of autonomous vehicles that depend on complex perception systems to make safety critical judgments, it is necessary to test and improve them. One of the ways this can be done is through vehicle-to-vehicle communication. This concept has been around for decades but was first standardized in 2010. Since then, there have been many hurdles in the path to applying this technology. Security, reliability, latency, and cost are the main reasons for the slow growth in this space. Another main problem is the lack of compelling applications that make overcoming these limitations worthwhile for industry. Autonomous Vehicles rely on a number of sensor types, with the most common ones being Cameras, Radars, and LiDAR. The detections from these three sensors are fused into a track list that can be used to plan and control the vehicles movements. This thesis proposes a system to introduce data from Vehicle-to-Vehicle messages into this fused track list. This extra information can be beneficial in cases when the onboard sensors are occluded or have low visibility. City and highway driving scenarios and Software-in-the-Loop testing is used to evaluate the proposed fused track list.

# Enhancing Perception Systems using Vehicle-To-Vehicle Enabled Sensor Fusion

Ansh S. Gwash

(GENERAL AUDIENCE ABSTRACT)

As cars become more automated they rely more on sensors to perceive the environment around them. Vehicles with automated features today use a combination of cameras, radars, and LiDAR to do this. Each of these sensor types has its own strengths and weaknesses. However, just like a human driver these sensors can't see everything. Bad weather, parked cars, and large vehicle can block their view, which makes it hard to detect objects in time. This thesis explores a solution to that problem using Vehicle-to-Vehicle (V2V) communication. V2V allows cars to wirelessly share location and speed information with each other. This project tested how adding V2V information to a car's regular sensor system could improve its awareness and safety. Using simulations of realistic city and highway driving scenarios, the study compared three approaches: using only the car's own sensors, using only shared data from other cars, and combining both. The results showed that combining the two made the car better at spotting hidden or distant hazards than sensors alone. Interestingly, in many cases, V2V alone performed even better—highlighting its powerful potential in improving safety on the road. By showing how connected cars can help each other see better, this work supports the idea of cooperative driving and safer autonomous vehicles in the future.

# Dedication

*Dedicated to my parents.*

# Acknowledgments

It has been a pleasure working with Dr. Talty. His insights of the automotive industry and technologies have been crucial in shaping this project. I would also like to thank Dr. Gracanin for providing his expertise in computer science to refine my writing. Lastly, I would also like to thank Dr. Meng for serving on this committee.

# Contents

<b>List of Figures</b>	<b>ix</b>
<b>List of Tables</b>	<b>xi</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Motivations . . . . .	1
1.2 Problem Statement . . . . .	6
1.3 Thesis Structure . . . . .	7
<b>2 Review of Literature</b>	<b>8</b>
<b>3 Methodology</b>	<b>12</b>
3.1 Motivation for Cooperative Perception . . . . .	12
3.2 Simulation Environment . . . . .	13
3.3 Sensor Modeling . . . . .	15
3.4 Fused Tracks Implementation . . . . .	17
3.5 Driving Scenarios . . . . .	22
3.5.1 Highway 1 Traffic Pile Up . . . . .	22
3.5.2 Highway 2 Median Crossover Hazard . . . . .	23

3.5.3	Highway 3 Merge . . . . .	24
3.5.4	Highway 4 Hidden Cut-in . . . . .	25
3.5.5	Highway 5 Emergency Stop on Curve . . . . .	26
3.5.6	City 1 Parked Vehicle Occlusion . . . . .	27
3.5.7	City 2 Intersection Occlusion . . . . .	28
3.5.8	City 3 Roundabout . . . . .	29
3.5.9	City 4 Left With Pedestrian . . . . .	30
3.5.10	City 5 Ego Straight, Vehicle Opposite Takes U-Turn . . . . .	31
3.6	Metrics . . . . .	32
<b>4</b>	<b>Results &amp; Discussion</b>	<b>35</b>
4.1	Overview of trends . . . . .	35
4.2	Highway Results . . . . .	37
4.3	City Results . . . . .	40
4.4	Aggregated Comparison . . . . .	43
<b>5</b>	<b>Conclusions</b>	<b>47</b>
5.1	Approach . . . . .	47
5.2	Key Findings . . . . .	48
5.3	Limitations . . . . .	49
5.4	Future Work . . . . .	50

<b>Bibliography</b>	<b>51</b>
<b>6 Appendix</b>	<b>54</b>

# List of Figures

- 1.1 SAE J3016 . . . . . 2
- 1.2 8 cameras on Tesla Model Y [2] . . . . . 3
- 1.3 All sensor types on Waymo’s Self Driving Taxi [1] . . . . . 3
- 1.4 Vehicle changing lanes is occluded by other cars creating a hazard [6] . . . . . 4
- 1.5 Toyota Vehicle-to-Infrastructure Application [5] . . . . . 5
  
- 3.1 Sensor models in Simulink . . . . . 13
- 3.2 Camera (blue) and RADAR (orange) field of view . . . . . 15
- 3.3 V2V Sim . . . . . 21
- 3.4 Highway 1 Traffic Pile Up . . . . . 23
- 3.5 Highway 2 Median Crossover Hazard . . . . . 24
- 3.6 Highway 3 Highway Merge . . . . . 25
- 3.7 Highway 4 Hidden Cut-in . . . . . 26
- 3.8 Highway 5 Emergency Stop on Curve . . . . . 27
- 3.9 City 1 Parked Vehicle Occlusion . . . . . 28
- 3.10 City 2 Intersection Occlusion . . . . . 29
- 3.11 City 3 Roundabout . . . . . 30
- 3.12 City 4 Left with Pedestrian . . . . . 31

3.13	City 5 U-Turn . . . . .	32
6.1	Hidden vehicles being tracked by V2V . . . . .	54
6.2	Hazard from other side of median detected early . . . . .	55
6.3	Pedestrian detected despite occlusion with truck in a busy urban environment	55

# List of Tables

- 4.1 Breakdown for Highway 1 scenario . . . . . 37
- 4.2 Breakdown for Highway 2 scenario . . . . . 38
- 4.3 Breakdown for Highway 3 scenario . . . . . 39
- 4.4 Breakdown for Highway 4 scenario . . . . . 39
- 4.5 Breakdown for Highway 5 scenario . . . . . 40
- 4.6 Breakdown for City 1 scenario . . . . . 41
- 4.7 Breakdown for City 2 scenario . . . . . 41
- 4.8 Breakdown for City 3 scenario . . . . . 42
- 4.9 Breakdown for City 4 scenario . . . . . 43
- 4.10 Breakdown for City 5 scenario . . . . . 43
- 4.11 Average GOSPA for local, V2V, and fused tracks . . . . . 44
- 4.12 Percentage improvement in GOSPA score compared to local baseline . . . . . 45

# List of Abbreviations

ADAS Advanced Driver-Assistance Systems

BSM Basic Safety Message

DoF Degrees of Freedom

DSD Driving Scenario Designer

EKF Extended Kalman Filter

FOV Field of View

GNSS Global Navigation Satellite System

GOSPA Generalized Optimal Sub-Pattern Assignment

INS Inertial Navigation System

LOS Line of Sight

MIO Most Important Object

OEM Original Equipment Manufacturer

ROI Region of Interest

SiL Software-in-the-Loop

V2V Vehicle-to-Vehicle

V2X Vehicle-to-Everything

# Chapter 1

## Introduction

### 1.1 Motivations

The automotive industry has seen a rapid increase in Advanced Driver Assistance Systems (ADAS) features such as Adaptive Cruise Control (ACC) and Lane Centering Control (LCC). There is also a significant push from the industry to get more autonomous vehicles on the road due to the pursuit of improved safety, convenience, and new mobility-as-a-service business strategies. In an annual survey by the McKinsey Center for Future Mobility it was found that around 50% of consumers are willing to pay up to \$9,999 for autonomous driving features [3]. These safety features and technologies are only possible through Software Defined Vehicles (SDV). SDVs are vehicles in which software plays a central role in controlling its functionality. This is different from traditional vehicles that are based only on hardware systems. Designing vehicles that are primarily controlled by software can have many benefits such as continuous over-the-air updates, modularity and customization, and most importantly autonomy features. These improvements make SDVs a compelling way forward for the automotive industry.

The autonomy features that are enabled with SDVs can become confusing due to the different names that automakers use. For example, Tesla's Autopilot, General Motors' SuperCruise, and Ford's BlueCruise don't give much information about the extent of their abilities. To combat this, SAE has standardized the levels of driving automation with SAE J3016 [4].

Figure 1.1 shows the five levels of automation, with 0-2 being driver support features and 3-5 being automated driving features. Drawing this line is important, as it helps identify what level of human engagement is required.

		SAE LEVEL 0	SAE LEVEL 1	SAE LEVEL 2	SAE LEVEL 3	SAE LEVEL 4	SAE LEVEL 5
What does the human in the driver's seat have to do?		You <b>are</b> driving whenever these driver support features are engaged - even if your feet are off the pedals and you are not steering			You <b>are not</b> driving when these automated driving features are engaged - even if you are seated in "the driver's seat"		
		You must constantly supervise these support features; you must steer, brake or accelerate as needed to maintain safety			When the feature requests, you must drive	These automated driving features will not require you to take over driving	
What do these features do?	These are driver support features			These are automated driving features			
	These features are limited to providing warnings and momentary assistance	These features provide steering OR brake/acceleration support to the driver	These features provide steering AND brake/acceleration support to the driver	These features can drive the vehicle under limited conditions and will not operate unless all required conditions are met	This feature can drive the vehicle under all conditions		
Example Features	<ul style="list-style-type: none"> <li>• automatic emergency braking</li> <li>• blind spot warning</li> <li>• lane departure warning</li> </ul>	<ul style="list-style-type: none"> <li>• lane centering OR</li> <li>• adaptive cruise control</li> </ul>	<ul style="list-style-type: none"> <li>• lane centering AND</li> <li>• adaptive cruise control at the same time</li> </ul>	<ul style="list-style-type: none"> <li>• traffic jam chauffeur</li> </ul>	<ul style="list-style-type: none"> <li>• local driverless taxi</li> <li>• pedals/steering wheel may or may not be installed</li> </ul>	<ul style="list-style-type: none"> <li>• same as level 4, but feature can drive everywhere in all conditions</li> </ul>	
<small>For a more complete description, please download a free copy of SAE J3016: <a href="https://www.sae.org/standards/content/J3016_201806/">https://www.sae.org/standards/content/J3016_201806/</a></small>							

Figure 1.1: SAE J3016

Companies like Waymo have already implemented Level 4 automation through their driverless taxis in select cities in the US. Other OEMs are also continuously developing and improving the automation offered on their vehicles. These features, which can improve safety, efficiency, and driver comfort, rely on a number of sensors to work. The most commonly used sensors are cameras, LiDAR, and RADAR. Different autonomy architectures use different combinations of these sensors to make up their perception suite. Tesla's Full Self-Driving (Level 2) uses an array of 8 cameras as shown in Figure 1.2. Waymo, on the other hand, uses all three types as shown in Figure 1.3. The main reason why Waymo uses a suite of LiDARs,



Figure 1.2: 8 cameras on Tesla Model Y [2]

RADARs, and cameras is that each of these sensors complement each other's weaknesses with their strengths, and vice versa. While the optimal combination of sensor types and placements can be debated, the fundamental purpose of the sensor suite remains the same: to provide the SDV with as much useful information as possible about its surroundings. This way, the vehicle can make better informed decisions about its motion planning and controls. The richer the vehicle's perception of the environment, the better the decisions it takes while navigating it.

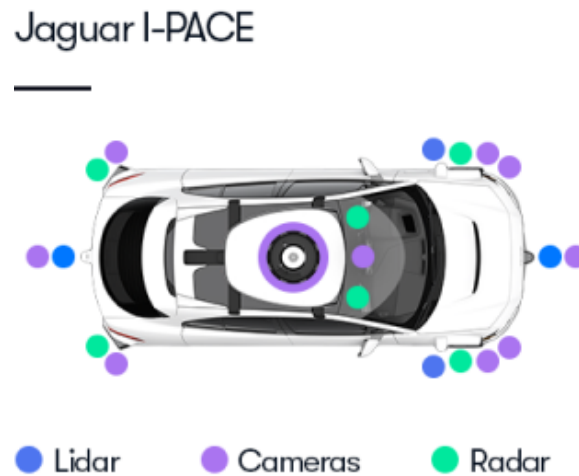


Figure 1.3: All sensor types on Waymo's Self Driving Taxi [1]

However, all of these sensors have one major downside that is often overlooked: point of view. Cameras, LiDAR, and radar can only perceive objects in their physical field of view,

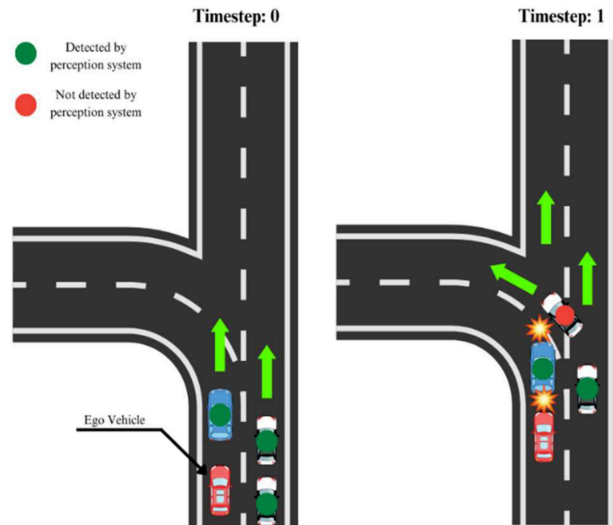


Figure 1.4: Vehicle changing lanes is occluded by other cars creating a hazard [6]

which makes them easily susceptible to occlusions from vehicles, infrastructure, or weather. Figure 1.4 shows a vehicle abruptly merging into the ego vehicle’s lane. However, because the hazard is blocked by the lead vehicle, the ego vehicle is unable to proactively break and prevent a collision. If the goal of autonomy is to advance vehicle safety to the point that it is better than a human driver, vehicles should be able to see beyond what a human can. Moreover, for a vehicle to exceed the situational awareness of a human driver, it must first perceive the environment around and then also predict events beyond its range of physical perception. This is where vehicle connectivity and Vehicle-To-Everything (V2X) technology becomes critical. V2X encompasses three types of connectivity: Vehicle-to-Vehicle (V2V), Vehicle-to-Infrastructure (V2I) and Vehicle-to-Pedestrian (V2P). Some information that can be shared between these nodes (Vehicle, Infrastructure, Pedestrian) includes: Speed, Location, Heading, Brake status, Traffic SPaT (Signal Phase and Timing), Traffic status, Lane locations and rules.

Although the concept of V2X technology has been around since the early 1980s, it was first standardized in 2003 in the IEEE 802.11p standard tailored for vehicular environments.

Since then, there have been a number of attempts to make applications of the technology compelling enough to implement in industry. *Soto et al.* break down these applications into three categories: road safety, traffic efficiency and other. Road safety applications help to protect vehicles, their occupants, and other road users [11]. Traffic efficiency applications aim to improve energy efficiency of vehicles. The other category contains applications for convenience or entertainment. Some examples of V2X applications being integrated in the real world by automakers are: Toyota’s DSRC-based V2V system in Japan, Audi V2I Connected Vehicle Project, and GM’s V2V which was offered on Cadillac CTS sedan.

Toyota’s implementation in Figure 1.2 shows the gauge cluster graphic that would be displayed when a V2X enabled vehicle is turning right at an intersection with a pedestrian crossing.

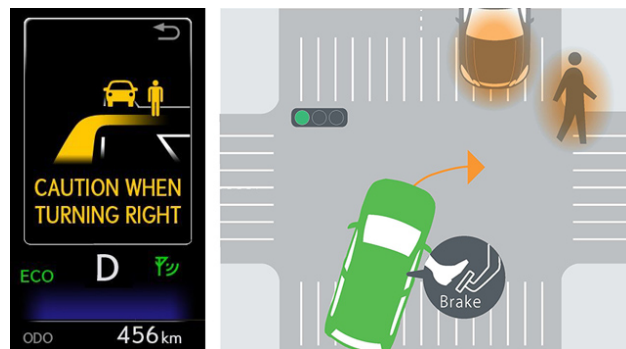


Figure 1.5: Toyota Vehicle-to-Infrastructure Application [5]

So far, V2X technology and applications have faced several challenges that have slowed its adoption in industry. One of the main factors being the uncertainty with regulations regarding spectrum use. Since 2018, several changes to spectrum allocation have been made in the United States. Due to the lack of compelling features and uncertainty in regulations, automakers haven’t been inclined to fit their vehicles with the expensive hardware required to make V2X possible.

However, the potential of V2X remains significant, and recent advancements in communica-

tion technologies have renewed interest in its adoption. The introduction of Cellular-V2X (C-V2X) as an alternative to Dedicated Short Range Communications (DSRC) has provided a more scalable and efficient solution for vehicle communication. C-V2X, which operates on LTE and 5G networks offers a number of advantages. One of these is that C-V2X has lower latency and higher reliability in its direct communication mode (PC5). Another advantage is broader coverage up to 1050 meters, which is larger than the typical DSRC range of 625 meters. The transition to C-V2X has been supported by major industry players including Qualcomm, Ford, and the 5G Automotive Association (5GAA).

## 1.2 Problem Statement

As discussed in the section above, range and field of view remain to be the biggest problems for any local sensors which include on-board cameras, radar, and LiDAR. This thesis aims to propose a new type of sensor fusion strategy that blends camera and radar tracks with V2X tracks to enhance the overall perception of the vehicle, especially in compromised scenarios. This method will allow us to offset the challenges that come with onboard sensors by pooling the sensors in multiple vehicles. Using this strategy, we can increase the performance of perception systems for autonomous vehicles, leading to better safety. In this thesis, we will use Software-in-the-Loop simulations to assess the increase in performance between standard camera and radar vehicle tracks and V2X enabled vehicle tracks in a number of driving scenarios. These scenarios will focus on neglected edge cases where standard camera and radar tracks typically fail to detect and track objects.

## 1.3 Thesis Structure

This thesis is broken up into five key chapters:

*Chapter 2* contains a comprehensive review of existing research on Vehicle-to-Everything communications and applications. It discusses the current research applications and explores relevant limitations of the technology.

*Chapter 3* contains the methodology. This chapter outlines the simulation-based methodology that was designed to evaluate the proposed fused V2X tracks. The research specifically focuses on enhancing vehicle perception. This is done by integrating V2X Basic Safety Messages with conventional ADAS sensors. The choice to use software-in-the-loop (SiL) testing was made due to its efficiency in rapid prototyping, low cost, and safety when compared to hardware or vehicle testing. SiL is essential for research, however, it does have inherent limitations with accuracy and latency. While the impact of these limitations can be reduced through careful modeling of the real-world, it can't be mitigated completely.

*Chapter 4* contains the results and discussion. This chapter explores the trends in GOSPA values as well as individual components such as localization errors, missed target errors, false track errors, and switching errors.

*Chapter 5* contains the conclusions. In this chapter, we will emphasize the problem statement, hypothesis, software design, and overall results. We will also state the challenges and learning from the project.

# Chapter 2

## Review of Literature

Most V2X research related to perception comes under the broad term of cooperative perception (CP). CP is the use of V2X communications to enhance the sensing abilities for automated vehicles. At the core of this is sensor fusion, which is the process of combining information from multiple kinds of sensors. This is done because each type of sensor has unique strengths and weaknesses. A camera can detect objects but it can't accurately measure the distance to another vehicle. A radar or lidar can accurately do this, but aren't able to easily detect objects. The goal of sensor fusion is to combine the information from each sensor modality to minimize their inherent weaknesses and maximize their strengths.

Sensor fusion strategies can be categorized by when the sensor information gets combined: early, intermediate, or late. In early fusion, vehicles transmit unprocessed sensor data to each other. In theory, this would provide the highest accuracy, however, it is impractical with the current bandwidth constraints [14]. In intermediate fusion, vehicles share processed features or maps. This can be practically achieved and is most common in literature. Late fusion includes vehicles sharing high level outputs from perception, which are typically object or track lists. Late fusion has many benefits as it includes the least amount of data. In a paper on enhancing lane detection using collaborative late fusion, *Jahn et al.* found that "late fusion approaches are better suited for realistic and scalable collaborative perception solutions" [8]. Another major benefit is that it is agnostic to the type of perception system a vehicle uses. Early and intermediate stage fusion needs to be more specific which narrows

the number of vehicles that would be compatible. However, the proposed method in this paper can scale to any number of vehicles regardless of the type of sensor suite used.

*Shan et al.* present a framework to fuse local and V2X perception data [10]. Their system uses the European standard ETSI Cooperative Perception Message which consists of perceived objects containers, each with their type, position, speed, heading, and dimensions. The perceived object information is in the coordinate system of the sender so it needs to be transformed before it is useful. They also validate the performance of the fused tracks using numerical and CARLA simulations, which makes it relevant to our research. One key limitation from this paper is that they use LiDAR and camera as the baseline perception system. Our proposed system will use the more mass market radar and camera configuration.

*Cui et al.* indicate that cooperative perception enhances the range of detections beyond the limitations of standalone sensors [7]. This significantly reduces blind spots from occlusions like parked vehicles, buildings, or larger vehicles. *Zhang et al.* further emphasize the reduction of blind spots, especially in inclement weather, or when visibility is compromised [16]. In practice, *Mo et al.* showed that connected vehicles detection information from V2X Road Side Units (RSUs) successfully detected pedestrians and vehicles hidden behind structural occlusions. These studies have proven the potential that V2X has in increasing safety [9].

Despite the obvious advantages, integrating V2X information with onboard sensor data presents unique problems that need to be solved for the technology to be compelling enough (for businesses and consumers) to implement at scale. Latency and synchronization are the two main problems, as shown by *Zhang et al.*, who report delays from 50-200 ms in C-V2X message delivery. These delays can result in spatio temporal misalignment between the onboard sensors data and external V2X inputs. This can result in inaccuracies in fused tracks. Another major limitation is the high bandwidth required to transmit perception information. This has sparked innovations in optimizing data sharing strategies to fit within

the V2X channels [7]. This is the research gap this thesis aims to explore. By using smart late fusion and minimal V2X information through BSMS, it is possible to create a system that provides a significant boost in perception within the constraints of the technology.

Another concern for V2X communications is latency. Traditional predictive approaches to reduce latency use kinematic motion models (constant velocity or acceleration) to project delayed V2X object positions forward. This allows the detections to be synchronized with real-time sensor data [12]. There have also been some ML techniques, such as FFNet [15] proposed by *Yu et al.*. These can predict future states from delayed feature data. A major way of improving tracking robustness comes by using validation by multiple sources before assigning priority to tracks. Systems such as those described by *Mo et al.* maintain high-priority tracks even during temporary occlusions [9].

Simulation-based validation is a popular method of testing V2X systems. Simulation allows researchers to test strategies in a safer controlled environment. CARLA is commonly used along with datasets like DAIR-V2X and OPV2V. These provided realistic test environments to develop and refine fusion algorithms. *Xu et al.* compare traditional single-vehicle to cooperative perception in simulated urban scenarios. Their results show a major improvement in safety when using cooperative perception [13]. Furthermore, hardware-in-the-loop testing using Cohda Wireless units confirm the real-world feasibility. *Mo et al.* shows a latency of approximately 120 ms in real-world V2I perceptions experiments.

Track consistency, measured by Generalized Optimal Subpattern Assignment (GOSPA) metrics, also improved substantially. This showed fewer object tracking errors and improved continuity through prolonged occlusions (*Mo et al.*, *Zhang et al.*). Additionally, V2X fusion reduced false positives by cross-validating sensor detections. This results in redundancy, enhancing the safety and robustness of perception systems.

In conclusion, recent academic advancements demonstrate that integrating V2X data into traditional camera and radar sensor fusion significantly enhances autonomous vehicle perception, particularly in occlusion and low-visibility scenarios. While substantial performance improvements have been achieved in simulations and controlled real-world experiments, practical implementation remains challenged by real-time processing limitations, communication reliability, and security risks. Addressing these issues will be critical in enabling widespread industry adoption and ensuring robust, safe autonomous vehicle operations. Late stage sensor fusion is also chosen for its simplicity and ability to scale in the future.

# Chapter 3

## Methodology

### 3.1 Motivation for Cooperative Perception

A self-driving stack is only as effective as the information it receives through its sensors. However, conventional ADAS sensors like cameras and radar face limitations in occlusion, adverse weather, and range. Even LiDARs sensors, which are gaining traction in autonomous vehicles only have an effective range of about 80 meters. Beyond this range, the resolution of the point cloud (i.e. sensor data) decreases drastically. Thus, it is crucial that we use networking and information sharing between vehicles and infrastructure to expand perception range and covering occluded areas. Cooperative perception aims to address the fundamental limitations of traditional ADAS sensors by leveraging data shared across connected vehicles. This enables an ego vehicle to receive information about objects beyond its direct line of sight, thereby improving situational awareness and decision-making under challenging conditions.

Occlusion is a primary motivator for cooperative perception. Vehicles can share what they see to alert others of hidden obstacles. For example, a car ahead might transmit the position of a pedestrian occluded from the following car's view, preventing a possible collision. Adverse weather is another case where vehicles equipped with different sensors can complement each other (one car's radar can penetrate fog and inform another car whose camera is impaired). Likewise, in sparse sensing situations (e.g. a vehicle with a sensor outage), data from nearby vehicles can provide redundancy. Overall, cooperative perception addresses the "field-of-

view” and “line-of-sight” limitations of single vehicles, creating a more robust collective sensing network. This is increasingly recognized as a path to safer autonomous driving – for instance, Euro NCAP’s 2025 roadmaps consider V2X-based safety features as essential for top safety ratings [1]. It is also important to note that to see benefits from V2V fused tracks, you don’t even need to be in an automated driving mode. Figure 1.5 shows Toyota’s implementation of V2X, where the extra information is presented to the driver.

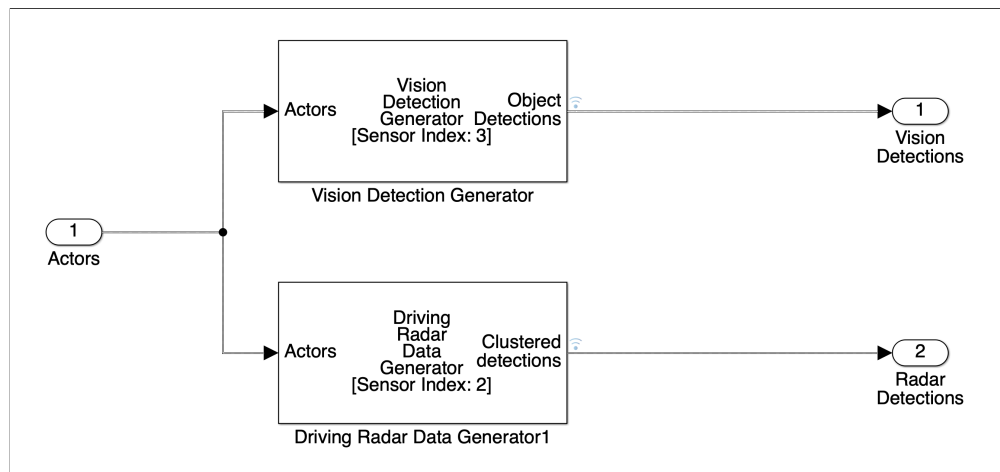


Figure 3.1: Sensor models in Simulink

## 3.2 Simulation Environment

Simulink is widely used in the automotive industry for prototyping and simulating complex systems. Its library of blocks allows for systems to rapidly be built and tested without the need of tinkering. This is why it was chosen as the primary tool. Scenario reader, coordinate transform, transmit and receive block were all developed in Simulink.

Another tool that was used was MATLAB’s Driving Scenario Designer (DSD). This is a tool that can be used to create complex scenarios with multiple actors on custom roads. MATLAB DSD was selected for its ability to generate diverse traffic scenarios, including

complex urban intersections, highway merges, and challenging visibility conditions. Driving Scenario Designer also lets us import custom roads from open source maps, as well as export scenarios to OpenSCENARIO and OpenDRIVE formats, which are both widely used in the automotive industry ensuring that this work will remain relevant.

V2V communication was modeled using the V2V Transmitter and V2V Receiver subsystems based on Mathworks' HelperV2VTrasnmmitter and HelperV2VReceiver objects. The V2V Transmitter is responsible for:

1. Reading actor information from the scenario.
2. Converting Cartesian coordinates to geographic coordinates using the scene origin.
3. Applying noise via an INS/GNSS error model.
4. Formatting the data into Basic Safety Messages (BSMs).
5. Delivering the BSMs to a FIFO entity queue using Simulink messages.

Each BSM contains fields such as GPS position, heading, speed, acceleration, steering angle, and brake status. Only the position and velocity fields are needed to create a fused tracks list.

The V2V Receiver processing messages by:

1. Taking the scene origin, ego pose, and precomputed channel characteristics as inputs.
2. Computing relative distance and estimating throughput using a mask parameter.
3. Comparing throughput to a random threshold to determine successful reception.
4. Outputting the received BSM as a structured Simulink bus signal.

These BSMs are parsed and transformed into the same detection format as the camera and radar signals. This is done so they can be concatenated with detections from the radar and camera sensors.

### 3.3 Sensor Modeling

The sensor setup being used will model what most cars with ADAS use today. This means a long-range radar and camera. The radar provides long-range detection capabilities, however, it struggles with resolution. Cameras offer high-resolution images which are essential for object recognition but can suffer in inclement weather conditions. The field of sensor fusion is vast and its sole aim is to combine the strengths from different types of sensors while reducing their weaknesses.

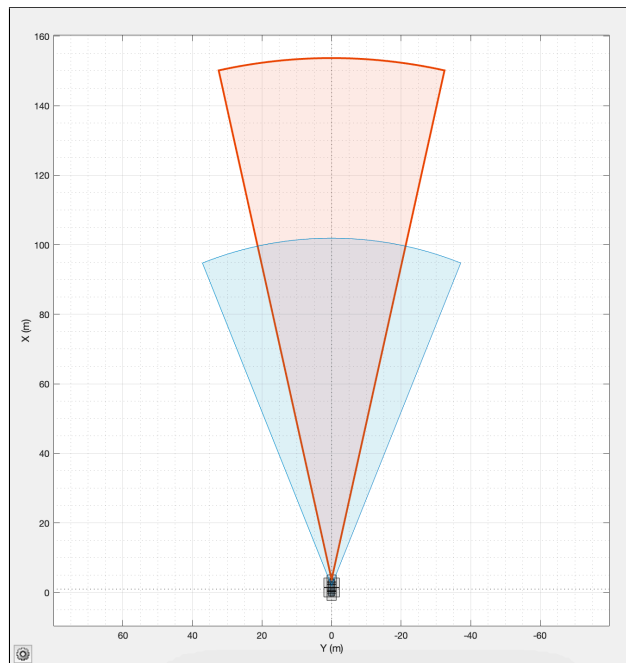


Figure 3.2: Camera (blue) and RADAR (orange) field of view

The ego vehicle's sensors were modeled using Mathworks' Automated Driving Toolbox. The

ego is equipped with a forward-looking camera and radar, corresponding to a typical Level-2 ADAS sensor suite. In Simulink, we use the Vision Detection Generator block (backed by the `visionDetectionGenerator` System object) for the camera, and the Driving Radar Data Generator block (backed by `radarDetectionGenerator`) for the radar. These blocks generate synthetic detections of other actors in the scenario, accounting for each sensor’s field of view, range, and resolution limits. By default, the detection generator objects produce measurements with random noise and can simulate false alarms and missed detections via a built-in statistical model. We enable the “Add noise to measurements” option for the radar and ensure the camera’s `HasNoise` property is true, so that every reported object position/velocity is perturbed by noise covariance appropriate to that sensor.

For the inclement weather scenarios, the magnitude of noise is tuned to represent degraded visibility: for example, we increase the standard deviation of the vision sensor’s range and bearing measurements to mimic the effect of fog reducing camera clarity, and we slightly increase radar measurement noise to reflect reduced radar confidence in heavy rain (though radar is generally robust to weather). We also reduce the camera’s detection probability (e.g. from the default 0.9 to a lower value) to simulate that a camera might fail to detect some objects in dense fog or low-light. The MathWorks documentation suggests that an ideal camera can be modeled with `DetectionProbability = 1` and no noise; conversely, our non-ideal camera uses a detection probability  $< 1$  to force occasional missed detections, and `HasNoise = true` to perturb measurements.

Similarly, the radar model’s false alarm rate can be adjusted to introduce spurious detections under adverse conditions, reflecting how certain radar returns might be misinterpreted during heavy clutter (though we keep this moderate to focus on occlusion-induced errors rather than excessive clutter). All these noise parameters are informed by literature and documentation: cameras degrade in bad weather or low visibility, whereas radars maintain

detection but with reduced discrimination. The sensor models are validated by checking that in clear conditions the fused system performs near-perfectly (high detection probability, low measurement error), whereas in our configured low-visibility mode, the sensors individually show degraded detection performance – for instance, the camera occasionally misses distant vehicles and reports slightly noisy positions. This form of validation is consistent with prior simulation studies of V2X and weather effects, which often use increased noise or reduced detection probabilities to emulate fog or rain. By using the `visionDetectionGenerator` and `radarDetectionGenerator`, we leverage proven sensor models that have been used in MathWorks examples for camera/radar fusion in fog and rain scenarios (e.g. adding noise to emulate sensor uncertainty).

### 3.4 Fused Tracks Implementation

The sensor types used for this project replicates that of most vehicles with SAE Level 2+ features: camera and radar. Each of these has its own strengths. Cameras are great at object detection but perform poorly at longer ranges and inclement weather. Radars perform well in poor weather and at longer ranges. The traditional camera and radar fusion has been successful in a vast majority of vehicles due to this reason.

Each kind of sensor comes with its strengths and limitations. Radars have great range but poor object detections and are noisy. Cameras are good for object detection but perform poorly in inclement weather. LiDARs can accurately get distances to any point but are expensive. More granular differences between sensor types were discussed in [1](#). Sensor fusion is a broad area that comprises of techniques to utilize the strengths of these sensors while minimizing their limitations. Fusing information provided by different types of sensors leads to a safer perception system with built-in redundancies and checks in place to safeguard

against faulty information. There are many different strategies to fuse sensor data. This section will be divided into two parts: fusing camera and radar tracks and fusing V2X tracks.

The fusion system is built around a central multi-object tracking filter that ingests detections from all sources and produces a set of tracked objects (with state estimates and unique IDs). In Simulink, we implement this using the Multi-Object Tracker block (from the Sensor Fusion and Tracking Toolbox). The Multi-Object Tracker is configured for a 3D state  $[x, vx; y, vy]$  (assuming a 2D planar motion with position and velocity in the horizontal plane), using a Constant Velocity Kalman Filter (which is essentially an Extended Kalman Filter since the measurements include nonlinear elements like angle). The tracker internally performs data association and track management; by default it uses a Global Nearest Neighbor (GNN) approach for associating detections to tracks, but we also explore a probabilistic association.

Specifically, we compare GNN with a Joint Probabilistic Data Association (JPDA) tracker to handle cases where multiple detections (e.g., from radar and V2V) pertain to the same object. The MathWorks tracking framework makes it straightforward to switch association algorithms (e.g., by using a `trackerJPDA` object instead of the default GNN tracker). We ensure that the tracker treats the V2V detections appropriately: since a V2V BSM essentially gives the exact position of a remote vehicle, it comes with much lower measurement noise than camera or radar. We assign a sensor index and measurement noise covariance for the V2V “virtual sensor” such that the tracker can weigh it heavily. For example, if camera position uncertainty is 3 m and radar is 1 m, we might set the V2V detection covariance to 0.5 m to indicate its high precision (this accounts for small GPS error).

The fusion logic must also prevent double-counting an object seen by both sensors and V2V. For this, correct data association is critical: the tracker uses gating (based on Mahalanobis distance) to decide if a camera/radar detection and a V2V detection correspond to the same vehicle. Thanks to the unique IDs in BSM, we could alternatively fuse at the track level by

matching IDs, but we treat it as an unassociated detection to remain general (in case one day V2V could also send info about a pedestrian or other object without a direct ID match). Our architecture diagram can be summarized as follows: Scenario Reader feeds ground-truth actor poses to sensor simulators (Radar & Camera blocks) and to the V2V transmitter for each remote vehicle. The ego vehicle’s Sensor Fusion and Tracking subsystem takes three inputs – radar detections, camera detections, and V2V detections. These are merged into a common bus and fed into the Multi-Object Tracker block. The tracker outputs a track list (with fused state estimates for each object). Finally, we connect a visualization (Bird’s Eye Plot and scope) to monitor the fused tracks versus ground truth for verification. We also log the track outputs for offline analysis.

Within this fusion system, time alignment of inputs is addressed as follows. The radar and camera are set to update every simulation timestep (the visionGenerator can be configured with an update interval equal to scenario sample time, e.g. 0.05 s, and radar similarly), so their detections arrive synchronously in our simulation. The V2V messages, however, come at a lower rate and with some latency. We use a Rate Transition block or a buffer to hold the last received V2V detection so that at each tracker update step, a V2V detection is available if one was received in the recent past. Essentially, if the tracker runs at 20 Hz and V2V at 10 Hz, we feed the most recent V2V detection on every tracker step until a new one arrives. This ensures the tracker doesn’t erroneously treat missing V2V updates as the object vanishing. We also timestamp each detection (the detection generators provide timestamps, and we assign one to the BSM-derived detection) so that the tracker can perform out-of-sequence measurement rejection if needed. In practice, because the Multi-Object Tracker block processes one step at a time with all available detections at that step, our buffering aligns detections to the nearest tracker step.

The Constant Velocity Kalman filter (with process noise tuned to allow  $\pm 2$  m/s<sup>2</sup> acceleration

variance) is used for highway and intersection motions. To handle potentially abrupt maneuvers (e.g., a vehicle braking suddenly out of occlusion), we consider an Interacting Multiple Model (IMM) tracker that switches between a constant-velocity and a constant-acceleration model. However, given the focus on perception differences, we keep the motion model uniform and instead rely on the V2V data to promptly inform on sudden changes. The track management parameters (confirmation and deletion thresholds) are set identically for both systems (radar+camera only vs. cooperative) to ensure a fair comparison. For instance, a track is confirmed if it is seen in at least 2 out of 3 consecutive frames, and deleted after 5 misses. These values are in line with typical fusion pipelines and ensure that false alarms are pruned while real objects are retained with minimal lag.

We anticipate that with V2V augmentation, track confirmation may happen faster (since an occluded vehicle that radar/camera barely see might still get confirmed due to V2V), and track deletion might be delayed (the tracker can coast through a few missed detections if V2V data filled in intermittently). To avoid track duplication (where V2V and radar each spawn a track for the same object), we rely on the tracker’s logic to merge them. We verified in a pilot simulation that the tracker successfully merges V2V and radar observations of the same vehicle into one track when the gating threshold is properly set. In ambiguous cases (e.g., two vehicles very close and one BSM coming in), a JPDA tracker helps assign the detections correctly to tracks by considering combined probabilities rather than a hard nearest-neighbor.

Additionally, if implementing track-to-track fusion (an alternative architecture), one must avoid “rumor propagation” where two vehicles keep sharing the same track back and forth. In our detection-level fusion design, this is not an issue: only non-ego vehicles transmit, and only the ego fuses, so there is no feedback loop of tracks. This design decision (centralized fusion in the ego) follows best practices to maintain consistency. Overall, our Simulink model

encapsulates this architecture in a modular way: the Actors and Sensors subsystem generates raw detections (with noise and occlusion effects), the V2V Communication subsystem generates additional detections from remote vehicles' states, and the Fusion and Tracking subsystem combines everything using a multi-sensor tracker. This modular approach mirrors frameworks seen in literature (e.g., multi-sensor data fusion for ADAS) and in MathWorks examples like Sensor Fusion Using Synthetic Radar and Vision Data, extended here with a V2V input.

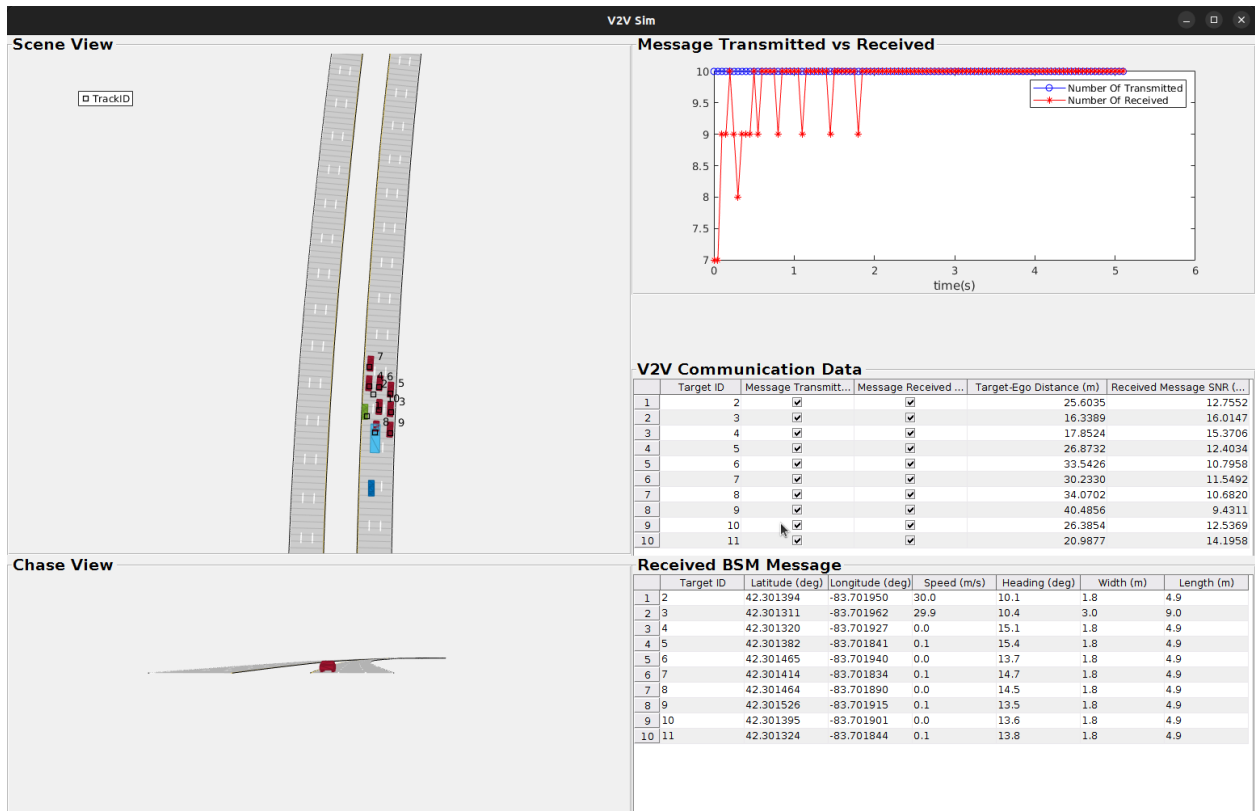


Figure 3.3: V2V Sim

## 3.5 Driving Scenarios

The use of driving scenarios allows us to test and evaluate the fusion algorithms for V2V and local tracks. While there are a vast number of driving scenarios, this project curates a list of scenarios where regular sensor fusion with just camera and radar would perform especially poorly. Through the literature review and initial tests we can hypothesize that the performance of perception systems improves significantly with added V2V data. These scenarios include occlusions, poor weather, and busy highways, where the chances of regular sensors not perceiving objects is higher. In normal scenarios we make the assumption that both camera radar and priority tracks will have the same performance. Now, let's go over each of the scenarios in more detail. Each of these scenarios will have a number of variations to make sure we are getting as much data as possible to later evaluate. These scenarios were created with the help of Mathworks Driving Scenario Designer tool.

In order to narrow down a list of scenarios, we will first look at the Operational Design Domain that the proposed fused track will be used in. Operational Design Domains or ODDs are situations where the proposed method would be used. For most autonomous vehicle features, there are mainly two ODDs: highway and city. We looked at crash data in each of these domains to create scenarios where V2V could have the most safety impact. Five scenarios for each ODD are created.

### 3.5.1 Highway 1 Traffic Pile Up

This scenario models one of the most common type of crashes: rear-end collisions. The ego vehicle is driving behind a box truck there is a multi-chain braking even down the road. Due to the height of the truck, the ego vehicle's camera and radar sensors are completely occluded. As such, the pile up remains invisible to ego's sensors until the truck in front slams

on the breaks. This limits the reaction time. V2V communication allows for the ego vehicle to track the vehicles way ahead of time, potentially increasing the time it has to react, as well as the perception system's performance.

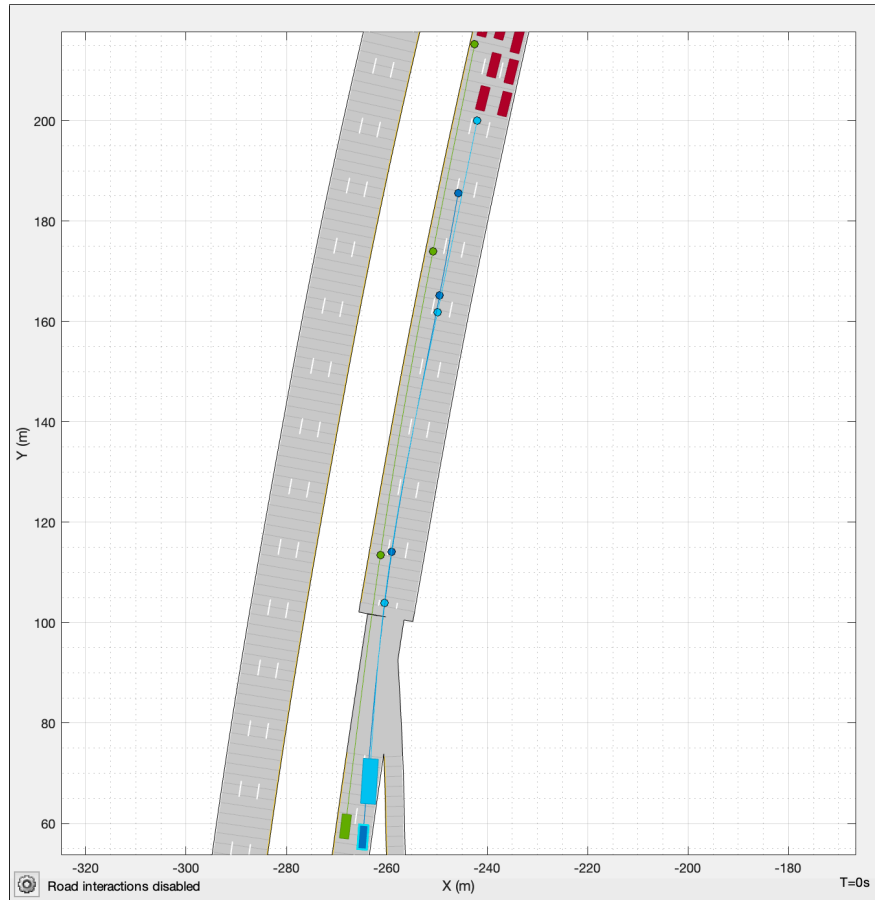


Figure 3.4: Highway 1 Traffic Pile Up

### 3.5.2 Highway 2 Median Crossover Hazard

In this scenario, a vehicle traveling in the opposite direction on a divided highway loses control, crosses the median, and enters the ego vehicle's travel lanes. Traditional perception methods are severely challenged due to obstructions like guardrails and median vegetation. This can result in a delayed detection until the hazard vehicle breaches the median barrier.

By leveraging V2V connectivity, the hazard vehicle is tracked throughout. The ego vehicle thus gains early and reliable knowledge of the impending crossover, enabling proactive hazard tracking and timely evasive action, greatly reducing collision risk.

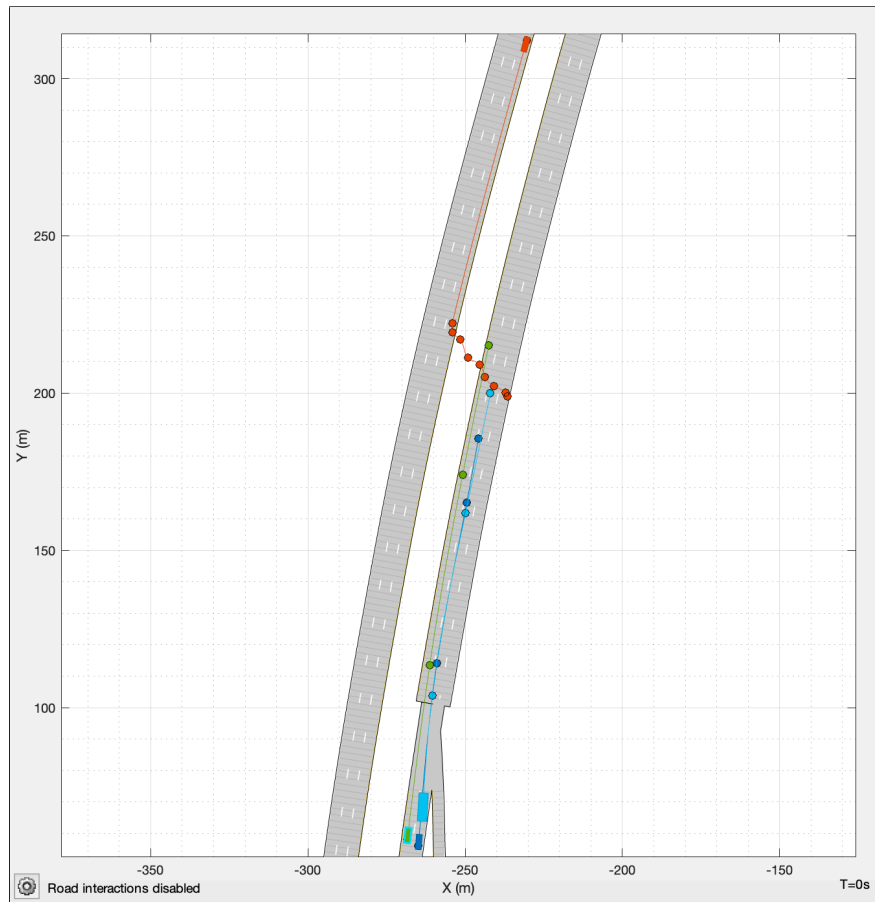


Figure 3.5: Highway 2 Median Crossover Hazard

### 3.5.3 Highway 3 Merge

This scenario shows our ego vehicle attempting to merge onto the highway from an on-ramp while truck is blocking its view. There is a vehicle behind the truck that isn't detected by the camera and radar sensor suite. As the ego vehicle begins merging, it is unaware of the rapidly closing gap, which results in a high-risk maneuver.

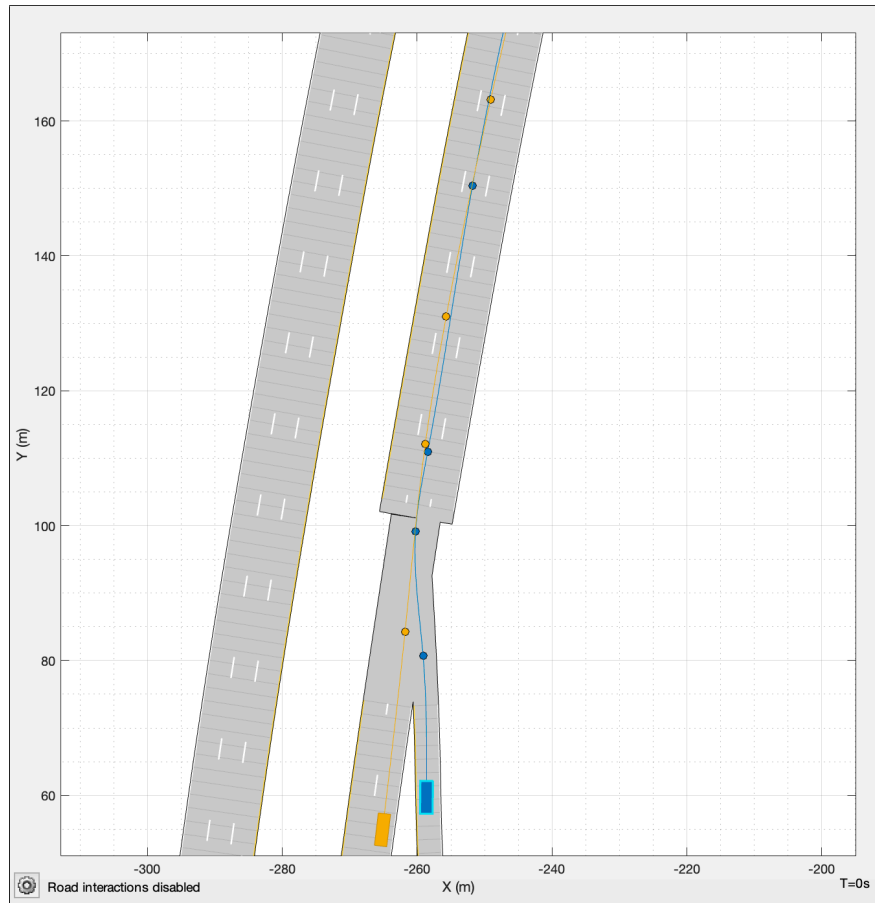


Figure 3.6: Highway 3 Highway Merge

### 3.5.4 Highway 4 Hidden Cut-in

In this scenario, the ego vehicle is driving closely behind a truck which obscures a V2V-connected vehicle traveling in an adjacent lane. The hidden vehicle initiates a sudden lane change into the ego vehicle's lane at close range. Local sensor will likely not detect this lane change and only perceive the lead vehicle slowing down abruptly. However, the integration of V2V BSM data means that the ego vehicle gets an early indication of the hidden vehicle's position and velocity, enabling early track initiation and safer, smoother maneuvers to accommodate the cut-in vehicle.

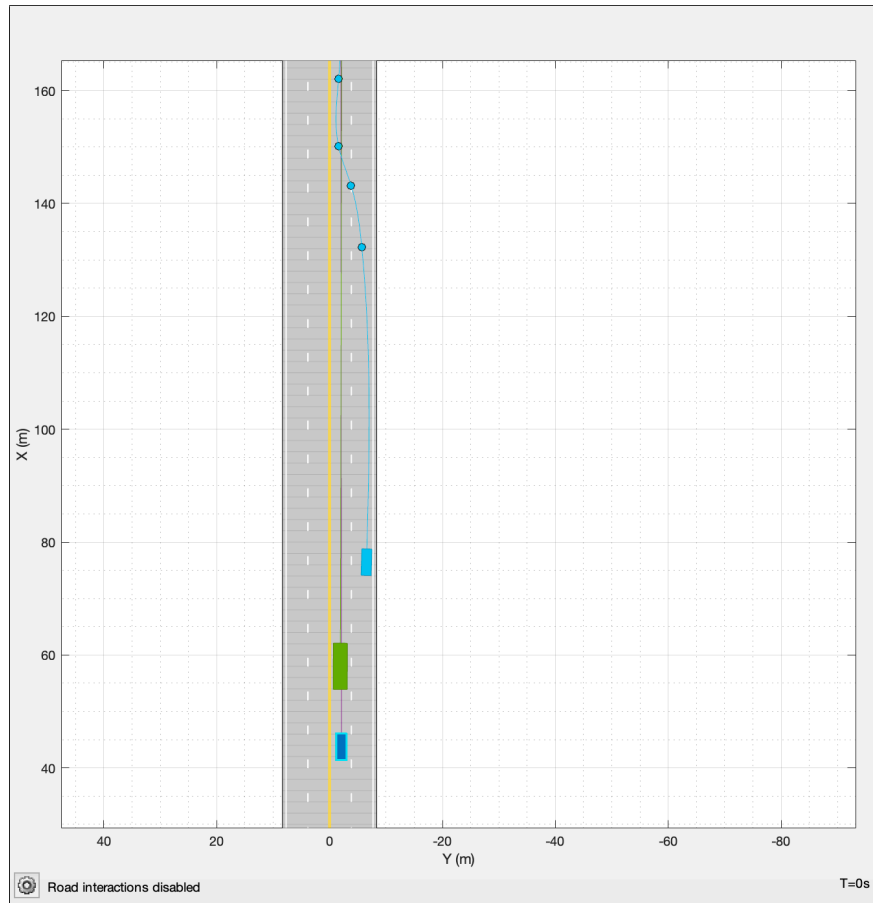


Figure 3.7: Highway 4 Hidden Cut-in

### 3.5.5 Highway 5 Emergency Stop on Curve

In this scenario, the ego vehicle is traveling along a highway curve at about 65 mph. A disabled vehicle is stopped in the same lane just beyond the curve, putting it outside the ego sensors' line-of-sight. Without V2X, the stopped vehicle is detected only once it enters the field of view at around 15 meters. This results in emergency braking being required. With V2X, the stopped vehicle's location is known in advance through BSMs, and the ego vehicle begins slowing down before it reaches the curve.

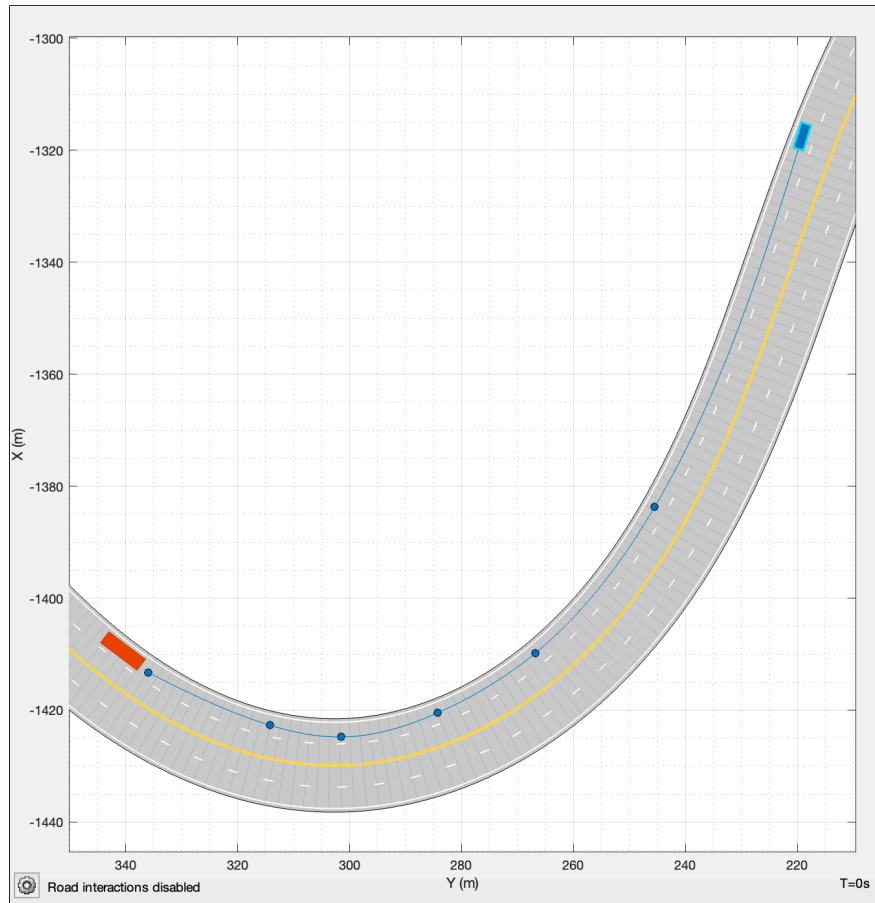


Figure 3.8: Highway 5 Emergency Stop on Curve

### 3.5.6 City 1 Parked Vehicle Occlusion

In this scenario, the ego vehicle is traveling straight while parked vehicles line the roadside. There is a pedestrian crossing the street that is hidden due to parked cars and cars waiting to turn right. It is important to note that in most modern cars AEB (Automatic Emergency Braking) would be triggered. However, these systems are a last resort and tend to violently break. With vehicles sharing perception information about the pedestrian, our ego vehicle can break much sooner.

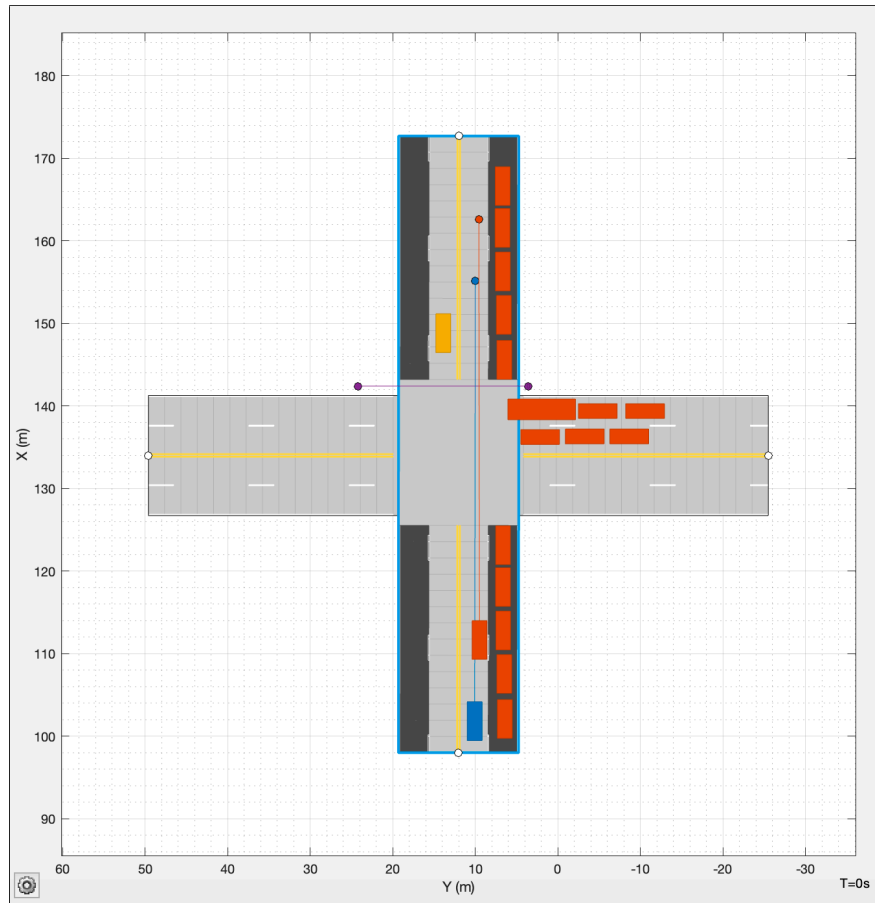


Figure 3.9: City 1 Parked Vehicle Occlusion

### 3.5.7 City 2 Intersection Occlusion

This scenario includes a vehicle trying to make an unprotected left turn at an intersection. While the ego vehicle approaches the intersection, a car with right of way is occluded by the lead vehicle. Once the lead vehicle begins turning, the ego vehicle also initiates the left turn. However, the ego vehicle's sensor fail to detect the oncoming vehicle which result in a close call situation.

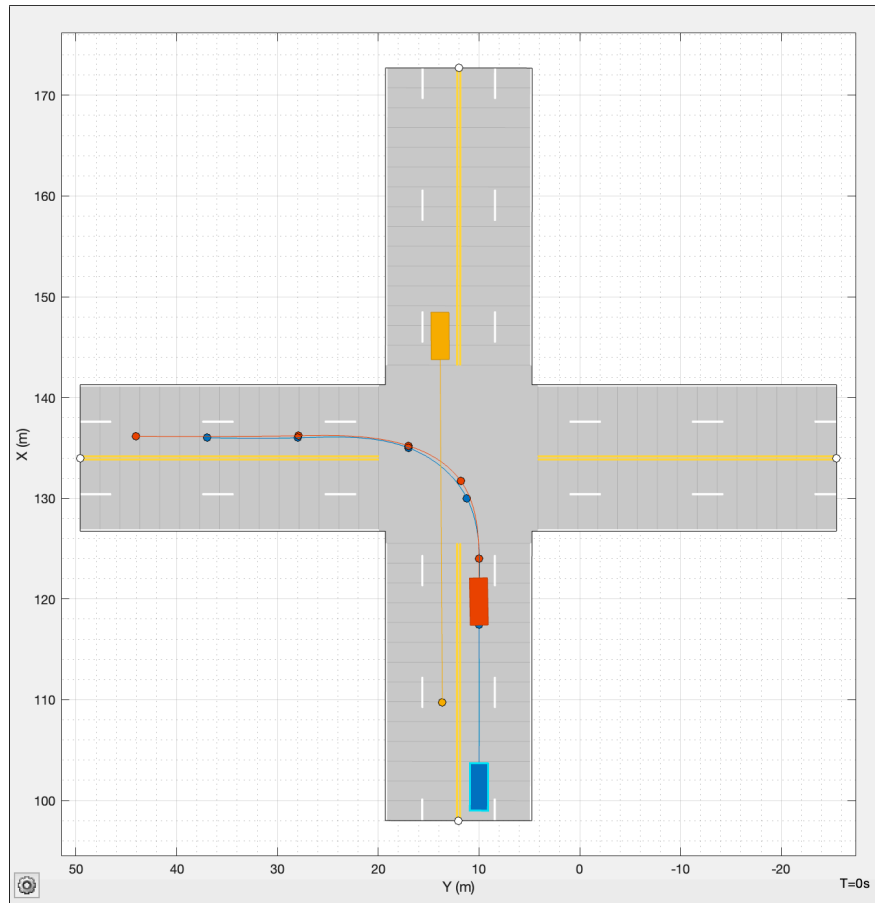


Figure 3.10: City 2 Intersection Occlusion

### 3.5.8 City 3 Roundabout

This scenario takes place at a single-lane roundabout where the ego vehicle is preparing to enter. A large SUV in the roundabout occludes a smaller car that's following closely behind. The ego vehicle detects the SUV and assumes the space behind it is clear, initiating entry into the roundabout. The car behind the SUV only appears after the maneuver has been initiated by the ego vehicle, which forces Automatic Emergency Braking. With V2X, the car's presence is known in advance and the ego vehicle can safely wait for an appropriate gap.

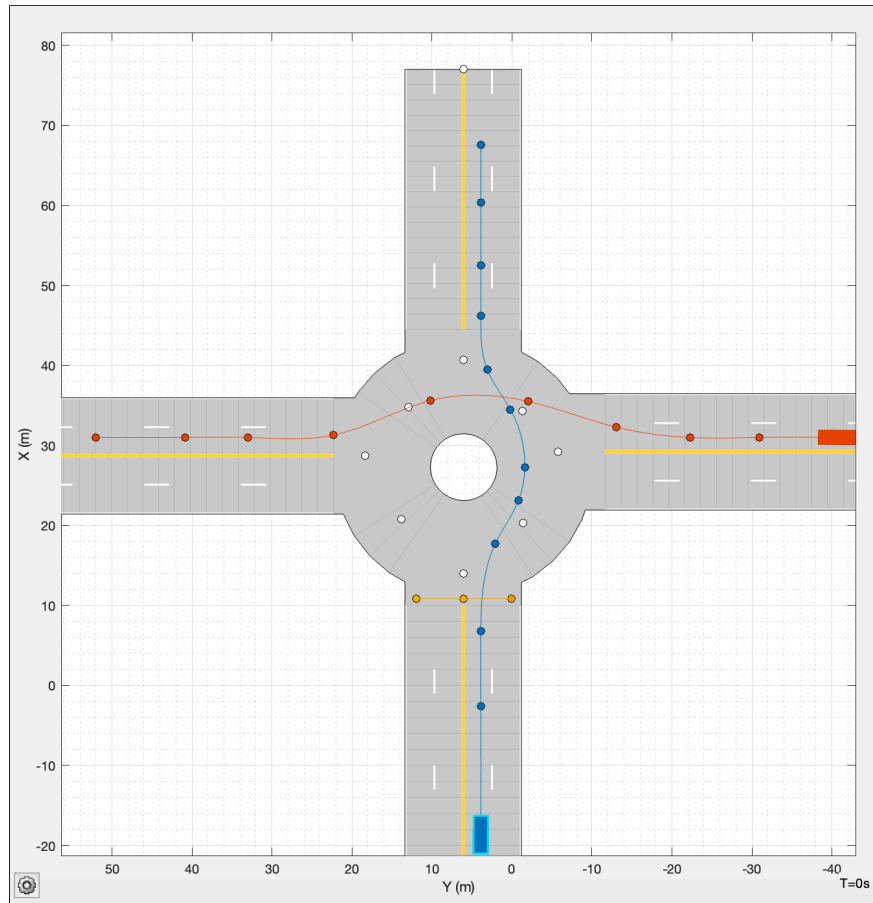


Figure 3.11: City 3 Roundabout

### 3.5.9 City 4 Left With Pedestrian

This scenario is designed to test the ego vehicle’s reaction to a sudden braking event in front of it, essentially mirroring an emergency halt on a highway. The ego vehicle is traveling at 65 mph with a 60-meter headway. The lead vehicle abruptly braking to a stop due to a pile up ahead. Using only camera and radar, the ego vehicle detects the braking at approximately 35 meters and must rely on emergency braking. With V2x, the lead vehicle broadcasts its braking intent and deceleration profile, allowing the ego vehicle to initiation braking earlier and more gradually.

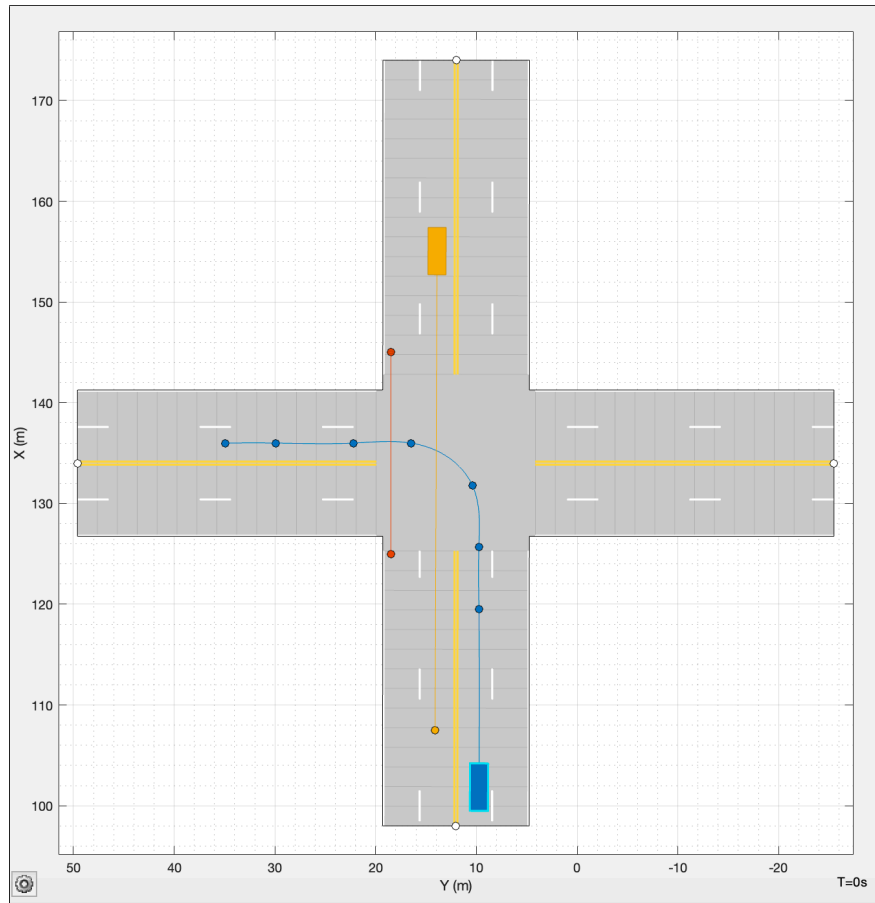


Figure 3.12: City 4 Left with Pedestrian

### 3.5.10 City 5 Ego Straight, Vehicle Opposite Takes U-Turn

The ego vehicle proceeds straight through a signalized intersection at moderate speed while a V2V-connected vehicle traveling in the opposite direction initiates a U-turn. The turning vehicle begins its maneuver from the far side of the intersection limiting the effectiveness of onboard sensors. As the U-turn progresses, the turning vehicle enters the ego lane with limited time for the ego to respond, creating a high-risk conflict. Traditional perception systems often detect such threats too late due to occlusion and limited field of view, resulting in abrupt braking or potential collisions. However, with V2V communication, the ego vehicle

receives real-time BSMs from the turning vehicle, enabling early prediction of its trajectory. This advanced situational awareness allows the ego system to initiate smoother deceleration or evasive actions, reducing collision risk and improving intersection safety.

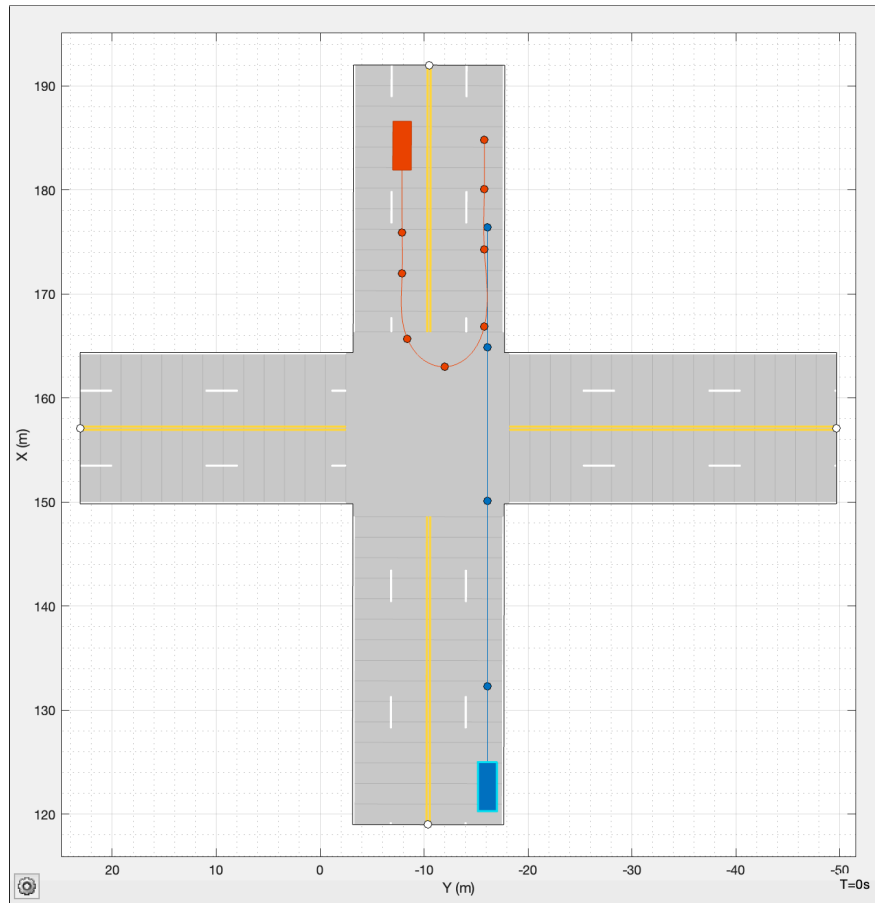


Figure 3.13: City 5 U-Turn

## 3.6 Metrics

To rigorously evaluate the perception performance of the baseline versus the V2V-enhanced system, we employ the Generalized Optimal Subpattern Assignment (GOSPA) metric. GOSPA is a multi-object tracking performance metric that captures localization errors, missed detections, false tracks, and track identity switches in a single unified measure. It is well-suited for

our evaluation because it provides a single scalar score at each time step, while also allowing breakdown into components, reflecting how well the ego perceives the full set of objects. In our analysis, we compute the GOSPA metric for each simulation run of each scenario, comparing the output tracks from the sensor fusion to the ground truth actor states.

We integrate the GOSPA Metric block from the Sensor Fusion and Tracking Toolbox into our Simulink model. This block accepts the current set of tracks (from the Multi-Object Tracker) and the ground truth (from the Scenario Reader or from platformPoses of the drivingScenario) and outputs the GOSPA value along with optional components. We configure the block to output all components: Localization error, Missed target error, False track error, and the Switching error (which penalizes track ID swaps). By enabling these, we can analyze not only the total GOSPA score but also why one system might be better – e.g., a system with V2V might have fewer missed targets (lower missed-target error) at the cost of perhaps a few more false tracks if it momentarily double-detects a vehicle. The cutoff distance and order parameters of GOSPA are set to standard values (e.g., cutoff=20 meters, order=2) as commonly used in literature for automotive tracking. The output of the GOSPA block is logged for each time step of simulation.

The trackGOSPAMetric usage allows flexibility to compute statistics across time or multiple runs. For instance, we compute the average GOSPA over the duration of a scenario as an aggregate performance measure for that run, and we examine the GOSPA time-series to see how quickly each system detects each object (a delayed detection of an occluded object will show up as a high missed-target penalty initially, then dropping once the object is tracked). In addition to GOSPA, other conventional metrics were also considered like detection recall (percentage of time the object was tracked) and localization RMSE. However, GOSPA provides an integrated view that is sensitive to both detection and tracking quality. A lower GOSPA value indicates better performance, as it means fewer misses/false tracks and

more accurate localization. We expect the radar+camera system to sometimes lose tracks (e.g., the occluded pedestrian might not be tracked at all, incurring a large missed-target error), whereas the cooperative system should maintain tracks on more objects, reducing missed-target error significantly.

We referenced related experiments in the literature: for example, Shan et al. (2022) report that adding V2X data in a fusion framework improved object tracking consistency and accuracy, which would manifest as improved multi-object tracking metrics. Similarly, an IEEE IV study on cooperative perception noted improved detection of vehicles in complex traffic, aligning with the expectation that the V2V-enabled system yields better perception scores. Our evaluation captures this by comparing GOSPA results: the cooperative system should yield lower GOSPA scores overall, particularly in scenarios with heavy occlusion or noise, demonstrating more robust perception.

In an intersection scenario with an occluded car, the baseline might have a high missed-target error until the car comes into view, whereas the V2V system, receiving BSMS, would show near-zero missed-target error (it never lost the car) but might have a small localization error (depending on GPS noise). By computing the time-averaged GOSPA and the maximum GOSPA during each encounter, we quantify improvements in both steady-state tracking and worst-case perception delay. All GOSPA calculations are done with the same parameters across systems to ensure fairness. The use of this metric aligns with best practices in multi-object tracking research, where metrics like OSPA/GOSPA are commonly used to evaluate and compare tracking performance. This quantitative evaluation is complemented by qualitative observation via the Bird’s Eye Plot animations to ensure that the numerical improvements correspond to intuitively better behavior (e.g., seeing that the ego car’s tracker picks up the hidden vehicle earlier with V2V).

# Chapter 4

## Results & Discussion

The test bench ran each scenario through three perception pipelines and gathered the localization error, missed targets, false tracks, and switching errors to calculate the GOSPA score at each timestamp. In the following sections we will compare the performance of three perception pipelines – local-only (camera and radar), V2V-only (BSMs), and fused (local and V2V)

### 4.1 Overview of trends

Across both city and highway scenarios, the V2V-only pipeline consistently achieved the lowest overall GOSPA scores, followed by the fused and local pipelines respectively. In highway scenarios, the fused pipeline demonstrated a 30% improvement over local in average GOSPA, while in city scenarios, this improvement was 15%. Figure 4.11 shows that the V2V only GOSPA was significantly lower than the fused GOSPA.

These results reveal a nontrivial insight: while fusion improves perception performance over onboard sensing alone, it does not outperform the V2V-only pipeline. This is counterintuitive since fusion is generally expected to leverage the strengths of both data sources. However, this behavior can be explain by looking at how the multi-object tracker works. The V2V-only pipeline uses highly accurate noise-free position information that comes straight from an actor’s ground truth. Although there is minor INS/GNSS error simulated, the measurements

are still very accurate. When these consistent low covariance measurements are sent to the tracker, it minimizes the localization and missed detection components.

Another trend throughout all the scenarios is the zero localization error with the local-only (camera and radar) pipeline. In all the simulation logs, localization error for local-only was exactly zero, while missed detection and false track error were much higher. This behavior can be explained by the way the GOSPA metric works. The localization error component is only calculated for detections that have successfully been matched to ground truth objects. However, in local-only perception the occlusions and poor range lead to frequent tracker failures. This can happen due to a track not being confirmed or detections that are mismatched. This gets exacerbated in crowded scenarios. Therefore, the zero values can be misleading as they mean a failure to track and not that the tracking was precise.

Fusion of V2V data with local sensors improves robustness but results in higher localization error when compared to just V2V. Across all scenarios, the fused pipeline consistently produced higher localization error than the V2V-only baseline. This can be explained by the noise in radar/camera detections which can degrade the estimated track position. However, the tracker still successfully matches fused tracks to ground truth, so GOSPA registers a localization penalty. This trend shows that fusion quality can be highly sensitive to quality of the detections as well as the data associate method used.

The V2V detections are treated as low-noise virtual sensors in this fusion pipeline. However, when noisy radar or camera detections are within the tracker's gating region, they are combined with V2V inputs in state updates. If the assigned covariances are not sufficiently differentiated, the tracker gives undue weight to lower-quality detections, pulling the track estimate away from the more reliable V2V position. This mis-weighting is especially impactful in scenarios with partial occlusion or measurement ambiguity. Proper tuning of sensor noise models and validation of gating thresholds can mitigate these effects in future

implementations.

## 4.2 Highway Results

The V2V-only configuration produced the lowest total GOSPA score across all the highway scenarios. This means that on paper it outperformed both fused and local-only pipelines. While this wasn't the hypothesized result, there are many insights we can gain from this. The reason that V2V-only performed the best is due to the BSM-derived position and velocities, which are directly generation from the actor's ground truth. As stated in Chapter 3, there is INS/GNSS noise that is simulated. However, the V2V tracks are still able to maintain clean, long-duration tracks with high precision. The GOSPA score is calculated by comparing the outputs of the perception pipeline with ground truths. Since all the vehicles in the simulated scenarios communicate their positions using Basic Safety Messages, the values are nearly identical to ground truth.

<b>Component</b>	<b>Local</b>	<b>V2V</b>	<b>Fused</b>
GOSPA	71.909	10.502	28.94
Switching	0	0	0
Localization	0	6.5779	9.9001
Missed	67.082	4.0196	3.1739
False Track	24.57	3.0246	25.471

Table 4.1: Breakdown for Highway 1 scenario

In the first highway scenario, the ego vehicle is following a tall truck that occludes vehicles ahead while a chain-reaction braking is taking place. This scenario portrays the inherent limitations of local sensing in high-occlusion and/or low-visibility scenarios. The local pipeline fails to track the slowing vehicles that are hidden in front of the box truck, which leads to the high missed detection penalty of  $\sim 67$ . V2V is able to boost the performance by tracking the

occluded vehicles early via BSMS. However, the fused pipeline suffers from over-association and track duplication. This can be attributed to the conflict between delayed or noise local sensor inputs and the highly accurate V2V positions.

The second highway scenario shows a vehicle crossing the highway median from the opposite direction and entering the ego vehicle’s lane. This is a high-speed cross-traffic scenario where the local sensor are blocked by guardrails on the highway. Local-only perception only detects the intruding vehicle when it is dangerously close, making a collision highly likely. V2V performs well and is able to continuously track the hazard using BSMS. This results in a  $\sim 78\%$  reduction in the missed targets. Figure 4.2 also shows that the fused system again struggled with conflicting inputs, creating 27 false tracks. This suggests that in scenarios with sudden lateral intrusion, especially in an opposing direction of travel, fusion can incorrectly interpret the motion path.

<b>Component</b>	<b>Local</b>	<b>V2V</b>	<b>Fused</b>
GOSPA	43.52	15.744	31.443
Switching	0	0	0
Localization	0	4.7202	6.662
Missed	36.742	8.0539	7.716
False Track	22.975	8.0334	27.776

Table 4.2: Breakdown for Highway 2 scenario

The third scenario has an ego vehicle trying to merge onto highway while being blocked by a truck when another vehicle approaches from behind. This creates a potential conflict point. This is a simple scenario and involves a partial occlusion with low vehicle density. Local sensors fail to detect the approaching vehicle due to the reduced field of view. Local sensors fail to detect the approaching vehicle due to the occluding truck, resulting in total track loss. V2V enables clean early detection, yielding the lowest GOSPA of all scenarios. The fused pipeline performs nearly as well, indicating no fusion degradation. This validates that in clean, low-noise scenes, fusion is both safe and beneficial—even slightly improving

robustness without incurring association penalties.

<b>Component</b>	<b>Local</b>	<b>V2V</b>	<b>Fused</b>
GOSPA	21.213	2.3312	3.817
Switching	0	0	0
Localization	0	2.3312	3.817
Missed	21.213	0	0
False Track	0	0	0

Table 4.3: Breakdown for Highway 3 scenario

In the fourth scenario, the ego vehicle is driving closely behind a truck when another vehicle abruptly cuts into its lane. This scenario stresses the latency and spatial precision of fusion during fast cut-ins. Local-only fails entirely, both missing the cut-in and incorrectly tracking the lead truck. V2V handles the cut-in well, tracking the adjacent vehicle even before it moves. However, the fusion system again suffers from a high number of false tracks—indicative of track duplication caused by inconsistent camera/radar and V2V detections. This reinforces that in rapid lateral maneuvers, where motion models are strained and observations can get misaligned, fusion must be managed more carefully.

<b>Component</b>	<b>Local</b>	<b>V2V</b>	<b>Fused</b>
GOSPA	39.016	3.3367	28.044
Switching	0	0	0
Localization	0	3.3367	7.2549
Missed	30	0	0.98666
False Track	24.728	0	26.322

Table 4.4: Breakdown for Highway 4 scenario

In the last highway scenario, the ego vehicle approaches a curve at a high speed when a disabled vehicle is stopped just beyond its line of sight. This event tests curvature-induced occlusion, where ego cannot see a stopped object until it’s dangerously close. Local sensing again fails, missing the stopped vehicle until it’s nearly within braking distance. V2V maintains clean tracking throughout the curve. Fusion performs well here—only a

minor increase in localization and false tracks.

<b>Component</b>	<b>Local</b>	<b>V2V</b>	<b>Fused</b>
GOSPA	24.03	2.4649	7.8936
Switching	0	0	0
Localization	0	2.4649	2.8396
Missed	21.213	0	0
False Track	5.58	0	5.7071

Table 4.5: Breakdown for Highway 5 scenario

### 4.3 City Results

Urban environments pose unique challenges for autonomous perception systems due to frequent occlusions, complex interactions at intersections, and dynamic road users such as pedestrians and turning vehicles. Unlike highways, where occlusions are typically longitudinal and short-lived, city scenarios feature lateral occlusion, tight spacing, and multi-agent interaction, all of which test the limits of traditional camera and radar perception. The use of V2V communication in these scenarios enables early awareness of objects beyond line-of-sight, while sensor fusion attempts to combine this with local data. This section analyzes the perception performance across five representative urban scenarios, each designed to stress a specific perception failure mode.

In the first city scenario, the ego vehicle drives past a line of parked vehicles, behind which a pedestrian begins crossing. Local sensors are heavily occluded until the pedestrian is nearly in front of the vehicle. V2V dramatically improves perception performance, reducing the GOSPA score from 97.40 (local-only) to 23.67 by enabling early detection of the occluded pedestrian. The local-only pipeline suffers 92 missed detections, showing it fails to detect the pedestrian until dangerously late. Fusion reduces the missed detection count but introduces a significant false track penalty (27.23), likely due to mismatched detections between the

delayed local sensors and accurate V2V reports. This demonstrates how fusion can recover continuity, but when local sensors lag behind, it can also confuse the tracker and increase false positives.

<b>Component</b>	<b>Local</b>	<b>V2V</b>	<b>Fused</b>
GOSPA	97.399	23.67	36.567
Switching	0	0	0
Localization	0	10.063	10.466
Missed	92.466	21.213	21.701
False Track	29.794	0	27.234

Table 4.6: Breakdown for City 1 scenario

The second city scenario shows an ego vehicle attempting a left turn, with an oncoming vehicle being occluded by a lead car until the last second. The V2V pipeline again performs best, achieving a low GOSPA of 5.12 and missing only 1.66 detections. Local-only misses 30 detections and generates nearly 23 false tracks, confirming that intersection occlusion is a critical weakness for camera and radar. The fused pipeline reduces missed detections to 11.33 but introduces the highest false track count of all city scenarios (29.32). This reflects difficulty in associating asynchronous local detections with V2V updates, especially during a brief, head-on occlusion. This scenario highlights how cross-path occlusions challenge fusion more than V2V-only perception, especially without temporal alignment or ID-level track association.

<b>Component</b>	<b>Local</b>	<b>V2V</b>	<b>Fused</b>
GOSPA	38.522	5.1239	35.149
Switching	0	0	0
Localization	0	2.7737	7.5327
Missed	30	1.6593	11.325
False Track	22.805	1.9494	29.315

Table 4.7: Breakdown for City 2 scenario

In the third city scenario, a close-following car in a roundabout is occluded by a larger SUV.

The ego vehicle misjudges the available gap and enters the roundabout prematurely. Surprisingly, V2V performs poorly here relative to other city scenarios: GOSPA is 21.37, with over 21 missed detections. This may stem from close proximity between the occluding and following vehicles, making BSM data difficult to separate spatially. Fusion also underperforms, with 24 missed detections and 13.96 false tracks, barely improving over local-only. This scenario exposes the limitations of V2V in tight, low-speed clusters and shows that fusion adds noise when local sensors fail to provide corroborating detections. It underscores the need for more discriminative association logic in urban environments with tight spacing.

<b>Component</b>	<b>Local</b>	<b>V2V</b>	<b>Fused</b>
GOSPA	33.042	21.367	32.947
Switching	0	0	0
Localization	0	2.0318	9.133
Missed	30	21.213	24.209
False Track	8.8956	0	13.959

Table 4.8: Breakdown for City 3 scenario

The last city scenario features an oncoming vehicle initiating a U-turn from across the intersection and entering the ego lane. V2V-only perception performs exceptionally well here, with a GOSPA of 2.08 and zero missed detections. In contrast, the local pipeline misses 21 detections and generates 25.43 false tracks, due to the turning vehicle being far outside the local sensor FOV at maneuver onset. Fusion reduces missed detections to 4.61 but adds nearly 28 false tracks, suggesting that the motion arc of the turning vehicle creates conflicting detections across time from different sensors. This scenario showcases the power of V2V in anticipating complex maneuvers, but also shows that fusion must be carefully tuned when motion is nonlinear or spans wide angles, to prevent fragmentation and redundant tracking.

<b>Component</b>	<b>Local</b>	<b>V2V</b>	<b>Fused</b>
GOSPA	31.383	31.713	32.9
Switching	0	0	0
Localization	0	2.3922	3.1558
Missed	30	25.704	24.858
False Track	3.4748	15.779	19.005

Table 4.9: Breakdown for City 4 scenario

<b>Component</b>	<b>Local</b>	<b>V2V</b>	<b>Fused</b>
GOSPA	33.434	2.084	29.837
Switching	0	0	0
Localization	0	2.084	2.8423
Missed	21.213	0	4.6116
False Track	25.428	0	27.881

Table 4.10: Breakdown for City 5 scenario

## 4.4 Aggregated Comparison

Table 4.11 shows the average GOSPA score for each of the three perception systems in all ten driving scenarios. The comprehensive evaluation across ten scenarios reveals some counterintuitive findings that challenge conventional assumptions about sensor fusion in autonomous vehicle perception. One example of this is that the V2V-only pipeline consistently achieved the lowest GOSPA scores across both highway and city environments. The fused pipeline came in second and the local-only pipeline third with a considerable margin.

The quantitative results demonstrate that while fusion provides substantial improvements over local-only perception—with average GOSPA reductions of 19.9 points in highway scenarios and 13.3 points in city scenarios—it fails to match the performance of V2V-only perception by significant margins of 13.2 and 16.7 points respectively. This gap stems fundamentally from the quality disparity between data sources. The V2V pipeline benefits from near-ground-truth position and velocity information with minimal INS/GNSS noise, while local sensors introduce measurement uncertainty, occlusion-induced gaps, and range

Scenario	Local GOSPA	V2V GOSPA	Fused GOSPA
Highway 1	72.083	10.502	28.94
Highway 2	43.679	15.744	31.443
Highway 3	21.213	2.3312	3.817
Highway 4	38.99	3.3367	28.044
Highway 5	24.078	2.4649	7.8936
City 1	97.352	23.67	36.567
City 2	38.511	5.1239	35.149
City 3	33.025	21.367	32.947
City 4	31.373	31.713	32.9
City 5	33.47	2.084	29.837

Table 4.11: Average GOSPA for local, V2V, and fused tracks

limitations that degrade the overall system performance when fused.

The fusion system’s suboptimal performance can be attributed to improper weighting of heterogeneous sensor inputs during track updates. When noisy radar or camera detections fall within the tracker’s gating region, they are combined with high-quality V2V measurements according to their assigned covariances. If these covariance models fail to adequately differentiate between sensor qualities, the tracker gives undue influence to lower-fidelity detections, pulling track estimates away from more reliable V2V positions. This phenomenon is particularly pronounced in scenarios involving partial occlusions or measurement ambiguity, where local sensor reliability is already compromised.

An important finding emerges from the localization error patterns across the three pipelines. The local-only system consistently reported zero localization error across all scenarios, which initially appears to suggest perfect tracking precision. However, this result is misleading and reflects a fundamental limitation in how GOSPA calculates localization penalties. The metric only computes localization error for tracks that have been successfully matched to ground truth objects. In the local-only pipeline, frequent tracking failures due to occlusions and poor sensor range result in unconfirmed tracks or detection mismatches, leading to high missed

detection penalties rather than localization errors.

Scenario	% $\Delta$ V2V vs. Local	% $\Delta$ Fused vs. Local
Highway 1	85%	60%
Highway 2	64%	28%
Highway 3	89%	82%
Highway 4	91%	28%
Highway 5	90%	67%
City 1	75%	62%
City 2	87%	8%
City 3	35%	0%
City 4	-1%	-5%
City 5	93%	11%

Table 4.12: Percentage improvement in GOSPA score compared to local baseline

In contrast, both V2V-only and fused pipelines consistently reported non-zero localization errors, indicating successful track-to-ground-truth associations despite some position uncertainty. The fused pipeline systematically produced higher localization errors than V2V-only across all scenarios, reinforcing the conclusion that incorporating noisy local sensor data degrades position accuracy even when tracks remain viable.

The impact of environmental complexity on fusion performance varies significantly between highway and urban scenarios. Highway environments primarily present longitudinal occlusion challenges from large vehicles and curvature-induced line-of-sight limitations. In these scenarios, fusion generally performed more predictably, with false track generation remaining moderate except in cases of rapid lateral intrusion or cut-in maneuvers.

Urban environments proved more challenging for fusion systems due to their inherently complex multi-agent interactions, lateral occlusions, and non-linear motion patterns. City scenarios consistently generated higher false track penalties for the fused pipeline, with particularly severe degradation in intersection scenarios and complex turning maneuvers. The worst fusion performance occurred in City 2, where a left-turn scenario with oncoming

vehicle occlusion resulted in 29.32 false tracks—the highest count in any scenario tested.

The fusion system’s struggle with track association became most apparent in scenarios involving rapid state changes or conflicting sensor inputs. Highway scenarios featuring sudden lane changes or cross-traffic intrusion consistently produced high false track counts, suggesting that the temporal misalignment between local sensor detections and V2V updates creates ambiguity in the data association process. This problem is exacerbated when motion models cannot adequately predict rapid lateral movements, leading to track fragmentation and redundant track creation.

The severity of association problems correlates strongly with scenario complexity. Simple scenarios with predictable motion patterns and minimal occlusion (such as Highway 3 and Highway 5) showed fusion performance nearly matching V2V-only results, with minimal false track penalties. This indicates that in clean environments where sensor inputs remain consistent and predictable, fusion can indeed provide benefits without significant degradation.

Despite its overall superior performance, the V2V-only pipeline exhibited notable limitations in specific scenarios. The most significant degradation occurred in City 3, where closely spaced vehicles in a roundabout scenario resulted in a GOSPA score of 21.37 with over 21 missed detections. This represents the worst V2V performance in any urban scenario and highlights a fundamental limitation of BSM-based perception: spatial resolution challenges when multiple vehicles operate in close proximity.

# Chapter 5

## Conclusions

The aim of software-defined vehicles and autonomous driving software is to provide groundbreaking improvements on road safety for everyone. Their performance relies on robust perception systems. The majority of innovation in this space focuses on onboard sensors like cameras, radars, and LiDAR. Each of these have their strengths and limitations. However, they all share the two main drawbacks: limited field of view and occlusions. This thesis explored a novel method of fusing data from Basic Safety Messages with traditional camera and radar tracks to improve the perception systems robustness and performance. The main aim was to create a system to achieve higher fusion quality and expand the range and operational readiness for autonomous vehicles.

### 5.1 Approach

To rigorously assess the effectiveness of V2V-enhanced sensor fusion, this research implemented a comprehensive Software-in-the-Loop (SiL) simulation environment using MATLAB's Simulink and Driving Scenario Designer tools. A representative selection of realistic driving scenarios—including both highway and urban environments—was developed to specifically challenge traditional perception systems. These scenarios featured critical edge cases like sudden lane changes, highway median crossovers, hidden pedestrians, intersection occlusions, and tight roundabouts, all aimed at exposing weaknesses in onboard sensors.

Three perception pipelines were systematically tested across these scenarios:

1. Local-only, relying exclusively on onboard radar and camera sensors.
2. V2V-only, utilizing precise location and velocity information transmitted via Basic Safety Messages (BSMs).
3. Fused, integrating both local sensor detections and V2V communications into a unified track list using multi-object tracking algorithms with Kalman filters and data-association techniques.

To objectively quantify the performance of each pipeline, the Generalized Optimal Subpattern Assignment (GOSPA) metric was employed. This metric allowed for detailed breakdowns of localization accuracy, missed detections, false tracks, and track identity errors, providing a nuanced understanding of each method’s strengths and limitations.

## 5.2 Key Findings

The results demonstrated significant performance differences among the three perception pipelines tested. Overall, the V2V-only pipeline consistently outperformed both the local-only and fused systems, achieving substantially lower GOSPA scores across nearly all highway and urban scenarios. Specifically, V2V data dramatically reduced missed detections, particularly in situations involving occlusions and limited visibility. This outcome underscores the inherent advantage of V2V’s precise and continuous communication of positional data, enabling accurate tracking even when local sensors fail due to obstructions.

The fused pipeline—while consistently superior to the local-only system—did not match the performance of the pure V2V pipeline. Fusion effectively decreased the number of missed

detections relative to the local-only pipeline, especially in scenarios involving sudden hazards and lateral intrusions. However, it introduced additional localization errors and false tracks, largely resulting from challenges in accurately associating noisy local sensor inputs with the more precise V2V detections. This issue was most prominent during scenarios featuring rapid maneuvers and short-duration occlusions, highlighting the need for careful calibration of track association methods.

Another significant observation was the local-only system’s inability to reliably detect critical hazards, particularly in urban scenarios involving complex interactions and occluded pedestrians. Frequent missed detections and high false track counts confirmed that purely sensor-based perception systems are fundamentally limited, reinforcing the potential value of V2V integration.

### 5.3 Limitations

Despite demonstrating clear benefits, this research faced several inherent limitations and practical challenges. Primarily, the evaluation relied entirely on Software-in-the-Loop (SiL) simulations. While SiL provides efficient prototyping and safety benefits compared to hardware testing, it introduces simplified models of real-world factors such as communication latency, sensor noise, and GNSS accuracy. Thus, performance results obtained through simulations might differ from real-world deployments, particularly regarding latency and noise impacts on tracking quality.

Moreover, the simulations utilized idealized V2V communication scenarios, with each actor reliably broadcasting Basic Safety Messages. Real-world conditions—including packet loss, varying communication ranges, and cybersecurity vulnerabilities—were not fully accounted for. Additionally, the fusion pipeline encountered frequent false-track generation due to

challenges in accurately associating asynchronous, noisy local sensor inputs with precise V2V messages, underscoring the need for more sophisticated tracking methods.

Another notable challenge was the assumption that all other vehicles were V2V-enabled. In practice, V2V adoption will initially be sparse, limiting its immediate effectiveness. Consequently, sensor fusion strategies must be robust enough to handle mixed scenarios, requiring careful balancing and calibration between onboard sensors and V2V data.

## 5.4 Future Work

The work in this thesis aimed to build a fusion framework that would be sensor suite agnostic and use minimal BSM messages to improve perception performance. The simulation shows this to be a success with an average 28% improvement in GOSPA. However, the real world performance of such a system is still untested. In most projects, simulation can only go so far. After this, Hardware-in-the-Loop and Vehicle-in-the-Loop testing needs to take place. This would be the next logical step to further strengthen the findings of this thesis. Hardware-in-the-loop testing could involve using an on-board V2V unit to transmit and receive BSMS to and from other devices. This would result in real-world noise and wireless communication interferences.

# Bibliography

- [1] How our cars drive - Waymo One Help, . URL <https://support.google.com/waymo/answer/9190838?hl=en>.
- [2] Model Y Owner’s Manual, . URL [https://www.tesla.com/ownersmanual/modely/en\\_us/GUID-682FF4A7-D083-4C95-925A-5EE3752F4865.html](https://www.tesla.com/ownersmanual/modely/en_us/GUID-682FF4A7-D083-4C95-925A-5EE3752F4865.html).
- [3] Perceptions of Advanced Driver Assistance Systems (ADAS) | McKinsey, . URL <https://www.mckinsey.com/industries/automotive-and-assembly/our-insights/hands-off-consumer-perceptions-of-advanced-driver-assistance-systems>.
- [4] SAE J3016 automated-driving graphic, . URL <https://www.sae.org/site/news/2019/01/sae-updates-j3016-automated-driving-graphic>. Last Modified: 2020-05-15T14:03:15-04:00.
- [5] Toyota Bringing Advanced ITS Technology to Mass-market Models | Toyota Motor Corporation Official Global Website, . URL <https://global.toyota/en/detail/9676551>.
- [6] Thomas Billington, Ansh Gwash, Aadi Kothari, Lucas Izquierdo, and Timothy Talty. Enhancing Track Management Systems with Vehicle-To-Vehicle Enabled Sensor Fusion, April 2024. URL <http://arxiv.org/abs/2404.17697>. arXiv:2404.17697 [cs].
- [7] Guangzhen Cui, Weili Zhang, Yanqiu Xiao, Lei Yao, and Zhanpeng Fang. Cooperative Perception Technology of Autonomous Driving in the Internet of Vehicles Environment: A Review. *Sensors*, 22(15):5535, January 2022. ISSN 1424-8220. doi:

- 10.3390/s22155535. URL <https://www.mdpi.com/1424-8220/22/15/5535>. Number: 15 Publisher: Multidisciplinary Digital Publishing Institute.
- [8] Lennart Lorenz Freimuth Jahn, Seongjeong Park, Yongseob Lim, Jinung An, and Gyeungho Choi. Enhancing lane detection with a lightweight collaborative late fusion model. *Robotics and Autonomous Systems*, 175:104680, May 2024. ISSN 0921-8890. doi: 10.1016/j.robot.2024.104680. URL <https://www.sciencedirect.com/science/article/pii/S0921889024000630>.
- [9] Yanghui Mo, Roshan Vijay, Raphael Rufus, Niels de Boer, Jungdae Kim, and Minsang Yu. Enhanced Perception for Autonomous Vehicles at Obstructed Intersections: An Implementation of Vehicle to Infrastructure (V2I) Collaboration. *Sensors*, 24(3):936, January 2024. ISSN 1424-8220. doi: 10.3390/s24030936. URL <https://www.mdpi.com/1424-8220/24/3/936>. Number: 3 Publisher: Multidisciplinary Digital Publishing Institute.
- [10] Mao Shan, Karan Narula, Stewart Worrall, Yung Fei Wong, Julie Stephany Berrio Perez, Paul Gray, and Eduardo Nebot. A Novel Probabilistic V2X Data Fusion Framework for Cooperative Perception. In *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*, pages 2013–2020, October 2022. doi: 10.1109/ITSC55140.2022.9922251. URL <https://ieeexplore.ieee.org/document/9922251/?arnumber=9922251>.
- [11] Ignacio Soto, Maria Calderon, Oscar Amador, and Manuel Urueña. A survey on road safety and traffic efficiency vehicular applications based on C-V2X technologies. *Vehicular Communications*, 33:100428, January 2022. ISSN 2214-2096. doi: 10.1016/j.vehcom.2021.100428. URL <https://www.sciencedirect.com/science/article/pii/S2214209621000978>.

- [12] Li Wang, Xinyu Zhang, Ziyang Song, Jiangfeng Bi, Guoxin Zhang, Haiyue Wei, Liyao Tang, Lei Yang, Jun Li, Caiyan Jia, and Lijun Zhao. Multi-Modal 3D Object Detection in Autonomous Driving: A Survey and Taxonomy. *IEEE Transactions on Intelligent Vehicles*, 8(7):3781–3798, July 2023. ISSN 2379-8904. doi: 10.1109/TIV.2023.3264658. URL [https://ieeexplore.ieee.org/abstract/document/10093116?casa\\_token=DUFesEw1R5AAAAA:68K508Pqayl1gTVIyj\\_5LRGjN7sZDKp2yU3Lsngi4YfpP7B\\_cYhCmXLTPbTs8YP-xKp50KCREg](https://ieeexplore.ieee.org/abstract/document/10093116?casa_token=DUFesEw1R5AAAAA:68K508Pqayl1gTVIyj_5LRGjN7sZDKp2yU3Lsngi4YfpP7B_cYhCmXLTPbTs8YP-xKp50KCREg). Conference Name: IEEE Transactions on Intelligent Vehicles.
- [13] Runsheng Xu, Hao Xiang, Xin Xia, Xu Han, Jinlong Li, and Jiaqi Ma. OPV2V: An Open Benchmark Dataset and Fusion Pipeline for Perception with Vehicle-to-Vehicle Communication, June 2022. URL <http://arxiv.org/abs/2109.07644>. arXiv:2109.07644 [cs].
- [14] Melih Yazgan, Thomas Graf, Min Liu, Tobias Fleck, and J. Marius Zoellner. A Survey on Intermediate Fusion Methods for Collaborative Perception Categorized by Real World Challenges, April 2024. URL <http://arxiv.org/abs/2404.16139>. arXiv:2404.16139 [cs].
- [15] Haibao Yu, Yingjuan Tang, Enze Xie, Jilei Mao, Jirui Yuan, Ping Luo, and Zaiqing Nie. Vehicle-Infrastructure Cooperative 3D Object Detection via Feature Flow Prediction, March 2023. URL <http://arxiv.org/abs/2303.10552>. arXiv:2303.10552 [cs].
- [16] Yuxuan Zhang, Bing Chen, Jie Qin, Feng Hu, and Jie Hao. CooPercept: Cooperative Perception for 3D Object Detection of Autonomous Vehicles. *Drones*, 8(6):228, June 2024. ISSN 2504-446X. doi: 10.3390/drones8060228. URL <https://www.mdpi.com/2504-446X/8/6/228>. Number: 6 Publisher: Multidisciplinary Digital Publishing Institute.

# Chapter 6

## Appendix

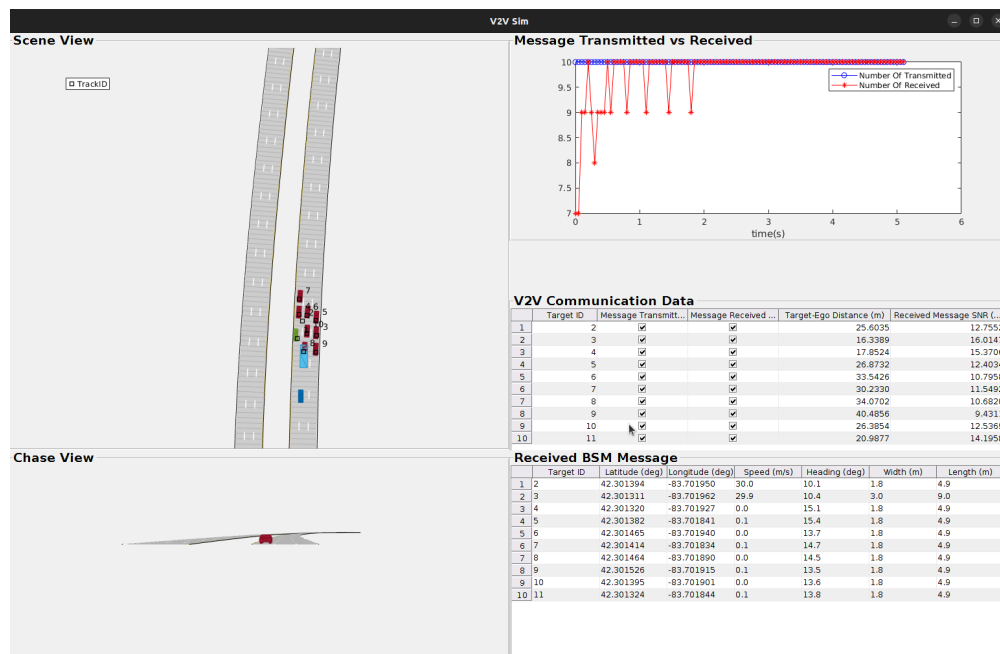


Figure 6.1: Hidden vehicles being tracked by V2V

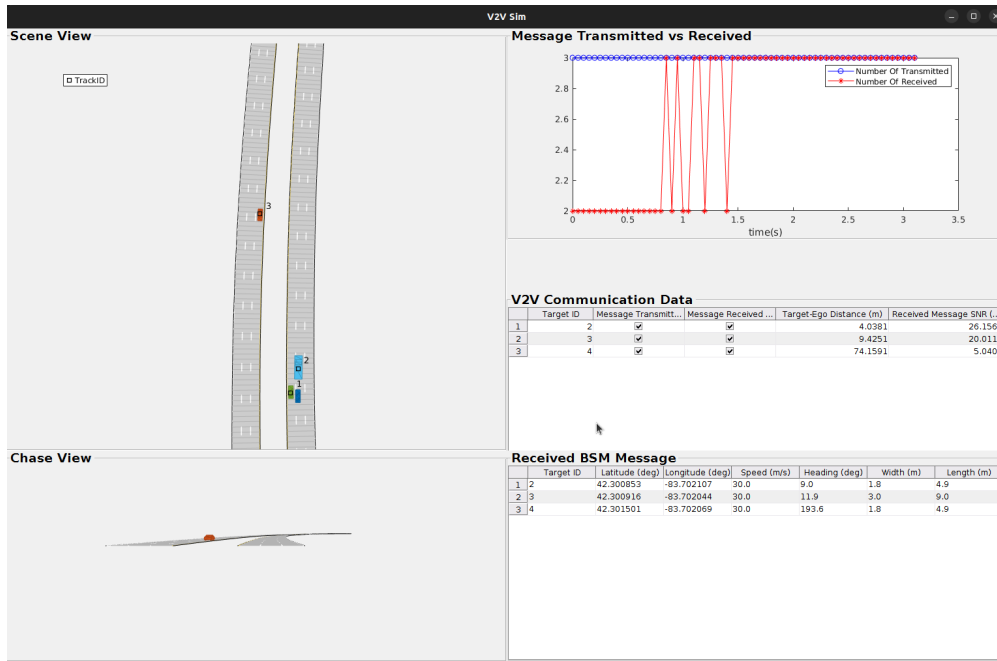


Figure 6.2: Hazard from other side of median detected early

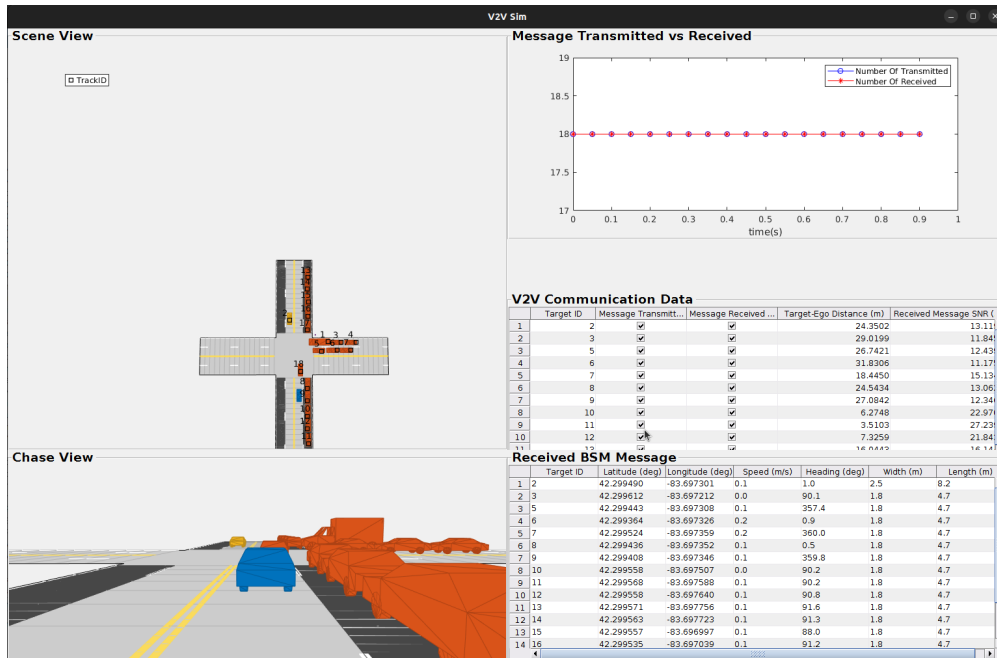


Figure 6.3: Pedestrian detected despite occlusion with truck in a busy urban environment