

Cooperative Perception in Autonomous Ground Vehicles using a Mobile Robot Testbed

Srivatsan Sridhar

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
Mechanical Engineering

Azim Eskandarian, Chair
John B. Ferris
Saied Taheri

September 14th, 2017
Blacksburg, Virginia

Keywords: Intelligent Vehicles, Autonomous Vehicles,
Connected Vehicles, Cooperative Perception

Copyright 2017, Srivatsan Sridhar

Cooperative Perception in Autonomous Ground Vehicles using a Mobile Robot Testbed

Srivatsan Sridhar

ABSTRACT

With connected and autonomous vehicles, no optimal standard or framework currently exists, outlining the right level of information sharing for cooperative autonomous driving. Cooperative Perception is proposed among vehicles, where every vehicle is transformed into a moving sensor platform that is capable of sharing information collected using its on-board sensors. This helps extend the line of sight and field of view of autonomous vehicles, which otherwise suffer from blind spots and occlusions. This increase in situational awareness promotes safe driving over a short range and improves traffic flow efficiency over a long range.

This thesis proposes a methodology for cooperative perception for autonomous vehicles over a short range. The problem of cooperative perception is broken down into sub-tasks of cooperative relative localization and map merging. Cooperative relative localization is achieved using visual and inertial sensors, where a computer-vision based camera relative pose estimation technique, augmented with position information, is used to provide a pose-fix that is subsequently updated by dead reckoning using an inertial sensor. Prior to map merging, a technique for object localization using a monocular camera is proposed that is based on the Inverse Perspective Mapping technique. A mobile multi-robot testbed was developed to emulate autonomous vehicles and the proposed method was implemented on the testbed to detect pedestrians and also to respond to the perceived hazard. Potential traffic scenarios where cooperative perception could prove crucial were tested and the results are presented in this thesis.

This work was carried out at the Autonomous Systems and Intelligent Machines Laboratory (ASIM) in the Department of Mechanical Engineering of Virginia Tech

Cooperative Perception in Autonomous Ground Vehicles using a Mobile Robot Testbed

Srivatsan Sridhar

GENERAL AUDIENCE ABSTRACT

Perception in Autonomous Vehicles is limited to the field of view of the vehicles' on-board sensors and the environment may not be fully perceivable due to the presence of blind spots and occlusions. To overcome this limitation, Vehicle-to-Vehicle wireless communication could be leveraged to exchange locally sensed information among vehicles within the vicinity. Vehicles may share information about their own position, heading and velocity or go one step further and share information about their surroundings as well. This latter form of cooperative perception extends each vehicle's field of view and line of sight, and helps increase situational awareness. The result is an increase in safety over a short range whereas communication over a long range could help improve traffic flow efficiency. This thesis proposes one such technique for cooperative perception over a short range. The system uses visual and inertial sensors to perform cooperative localization between two vehicles sharing a common field of view, which allows one vehicle to locate the other vehicle in its frame of reference. Subsequently, information about objects in the surroundings of one vehicle, localized using a visual sensor is relayed to the other vehicle through communication. A mobile multi-robot testbed was developed to emulate autonomous vehicles and to experimentally evaluate the proposed method through a series of driving scenario test cases in which cooperative perception could be effective and crucial to the safety and comfort of driving.

Acknowledgments

“If I have seen further than others, it is by standing upon the shoulders of giants.”

— Issac Newton

I’m deeply indebted to my graduate advisor and mentor, Dr.Azim Eskandarian for having provided me, the opportunity to work with him and learn from his years of experience on Intelligent Vehicles Research. I’m grateful for all the insight, both technical and on the general nature of research, and for his constant motivation, patience and optimism, which kept me going. He steered me in the right direction when necessary, but also provided the freedom that empowered me to develop independent ideas.

I thank Dr.John Ferris and Dr.Saied Taheri for serving as my graduate thesis committee members, for their expert feedback and for reviewing this work to completeness. I am also indebted to Dr.John Ferris for his guidance and advice during my time at Virginia Tech.

I’m thankful to all my labmates at the ASIM Lab, especially Dr.Yuan Lin for his contribution in developing the initial indoor test-track and positioning system and Dr.Lirong Wang for her feedback and advice. I’m grateful to my friends and colleagues for all the motivation, technical insight and support in much needed moments. I thank Savio Pereira for the detailed discussions that helped refine the idea. I thank my professors at Virginia Tech for equipping me with the necessary background for carrying out this research. I’m also thankful to the ME Department Staff, who always had the time and a solution, for all my queries, that helped complete this project on time.

Lastly, I’m grateful to my parents for having believed in me. None of this would have seen the light of day, if not for them. They were a source of endless emotional support and their love, assurance and motivation kept me working through thick and thin and succeed in Graduate school.

Contents

1	Introduction	1
1.1	Motivation and Background	1
1.2	Problem Formulation	3
1.3	Goals and Contributions	5
1.4	Scope of Work	5
1.5	Thesis Outline	6
2	Literature Review	7
2.1	Mobile Robotics	7
2.2	Autonomous Vehicles	9
2.3	Connected Vehicles	10
2.4	Perception in Autonomous Vehicles	12
2.5	Cooperative Perception	14
2.5.1	Ground and Aerial Robots	14
2.5.2	Ground Vehicles	15
2.5.3	Advantages and Challenges of Cooperative Perception	16
3	Relative Pose Estimation	18
3.1	Vision-based orientation estimation	18

3.1.1	Feature Correspondence Matching	19
3.1.2	Extraction of Fundamental and Essential Matrix	20
3.2	IMU-based orientation estimation	23
3.3	Cooperative localization system	23
4	Object localization on the Inverse Perspective Map	26
4.1	Inverse perspective mapping	26
4.2	Object Localization	30
4.3	System Overview	32
5	Mobile Robot Testbed Development	34
5.1	Wifibot	34
5.1.1	ROS Architecture	37
5.1.2	Indoor Positioning System (IPS)	39
5.1.3	Tele-operation and low-level control	40
5.1.4	Stop Sign Detection	41
5.1.5	Simple Emergency braking	42
5.1.6	Pedestrian Detection	42
5.1.7	Lane Detection	42
5.2	Cooperative Perception Algorithm	43
6	Experimental Evaluation of Cooperative Perception	48
6.1	Evaluation of approximate object localization using a monocular camera	48
6.2	Evaluation of Relative pose estimation	50
6.3	Evaluation of potential traffic scenarios	52
6.3.1	Scenario-1: Vehicles traveling along the same direction on adjacent lanes	54

6.3.2	Scenario-2: Vehicle merging onto traffic	56
6.3.3	Scenario-3: Sensor/Detection Failure: Vehicles traveling along the same direction on adjacent lanes	57
6.3.4	Scenario-4: Vehicles traveling on the same lane with one vehicle behind the other vehicle	59
6.3.5	Scenario-5: Vehicles on adjacent lanes with a non-cooperative vehicle in front	61
6.3.6	Scenario-6: Intersection of Roads	63
7	Conclusion	80
7.1	Conclusions	80
7.1.1	Relative pose estimation	81
7.1.2	Object detection and localization	82
7.2	Future Research	83
7.2.1	Relative pose estimation	83
7.2.2	Object localization and map merging	83
7.2.3	Cooperative Perception System	84
	Bibliography	85

List of Figures

1.1	An illustration of Cooperative Perception in vehicles	4
3.1	Image feature points	19
3.2	Feature matching	20
3.3	Robots observed by the overhead camera	24
3.4	Relative camera orientation estimation	25
4.1	Perspective image and Inverse Perspective image	29
4.2	Pedestrian detection (left) and localization (right)	31
4.3	Pipeline for pedestrian localization	32
4.4	Cooperative perception - System overview	33
5.1	Wifibot Robot	35
5.2	Wifibot Hardware Architecture [Source:Nexter Robotics]	37
5.3	Sensors on the Wifibot	38
5.4	Overhead view of the Indoor track (left) and Detected Wifibot Location superimposed on a binary image of the track (right)	39
5.5	Indoor track and LED on the Wifibot	40
5.6	Steering and Joystick - Teleoperation	40
5.7	Front camera image observed by the wifibot	41

5.8	Stop sign recognition	41
5.9	Pedestrian toy (left) and Pedestrian detection result(right)	42
5.10	Inverse perspective map and edge image of the Inverse perspective map	43
5.11	Lane detection	43
5.12	Lane detection based on polynomial fitting on the IPM edge image	44
5.13	Implementation of Cooperative Perception system	45
5.14	Relative Orientation - Initialized by Camera Relative pose and updated by IMU readings	46
5.15	Pedestrian localization	47
5.16	Result of Cooperative perception	47
6.1	Evaluation of the approximate localization system (image not to scale)	49
6.2	Cooperative Perception Interface - Ego Vehicle	53
6.3	Indoor camera-based localization(left) and Ground truth images(right)	54
6.4	Cooperative perception across vehicles on adjacent lanes (left), on a wide road without lanes (center) and pedestrian crossing in front of stationary vehicle 'B' and suddenly emerging onto the future trajectory of the ego-vehicle on the adjacent lane (below)	55
6.5	Experimental Evaluation of Scenario-1	55
6.6	Cooperative perception across vehicles that approach each other in a traffic merging scenario	56
6.7	Scenario-2 in the absence of Cooperative Perception	56
6.8	Experimental Evaluation of Scenario-2	57
6.9	Cooperative perception across vehicles on adjacent lanes (Sensor/Detection Failure)	58
6.10	Scenario-3 in the absence of Cooperative Perception : Ego-vehicle's sensor fails	58
6.11	Experimental Evaluation of Scenario-3: Emergency braking response	58

6.12	Experimental Evaluation of Scenario-3: Emergency lane change response . . .	58
6.13	Cooperative perception across vehicles on the same lane	60
6.14	Scenario-4 in the absence of Cooperative Perception	60
6.15	Experimental Evaluation of Scenario-4	60
6.16	Cooperative perception across vehicles with a non-cooperative vehicle in front	62
6.17	Scenario-5 in the absence of Cooperative Perception	62
6.18	Experimental Evaluation of Scenario-5	62
6.19	Cooperative perception across vehicles at an intersection	64
6.20	Scenario-6 in the absence of Cooperative Perception	64
6.21	Experimental Evaluation of Scenario-6	64
6.22	Experimental Results of Scenario-1	66
6.23	Experimental Results of Scenario-1	67
6.24	Experimental Results of Scenario-2	68
6.25	Experimental Results of Scenario-2	69
6.26	Experimental Results of Scenario-3 (emergency stop case)	70
6.27	Experimental Results of Scenario-3 (emergency stop case)	71
6.28	Experimental Results of Scenario-3 (lane change case)	72
6.29	Experimental Results of Scenario-3 (lane change case)	73
6.30	Experimental Results of Scenario-4	74
6.31	Experimental Results of Scenario-4	75
6.32	Experimental Results of Scenario-5	76
6.33	Experimental Results of Scenario-5	77
6.34	Experimental Results of Scenario-6	78
6.35	Experimental Results of Scenario-6	79

List of Tables

5.1	Camera Specifications	36
5.2	IMU Specifications	36
5.3	Motor Specifications	36
5.4	Infrared Sensors Specifications	37
6.1	Approximate Object Localization Results (distance units are in centimeters)	49
6.2	Relative orientation under different camera exposure settings (units are in degrees for fields 2 through 4)	51
6.3	Relative orientation under different backgrounds (units are in degrees for fields 2 through 4)	51

Chapter 1

Introduction

1.1 Motivation and Background

Autonomous Vehicles are one of the recent advancements in the field of Ground-based Transportation Technology. A fully autonomous vehicle is a vehicle which is capable of navigating from one point to another on its own without the need for manual intervention. A concept that was envisioned as early as the 1920s [1], [2], [3] has gained momentum in the recent past few years due to technological advances. The Automotive Industry is witnessing a major paradigm shift towards automating a task that has been done manually for over a century - Driving. Advanced Driver-Assistance Systems (ADAS) such as Adaptive Cruise Control (ACC) and Lane departure warning system [4] can already be found on modern-day cars. Such contemporary systems on the road are adept in providing assistance to the driver to perform various tasks related to driving as well as in warning the driver of imminent potential threats on the road. However, several challenges have to be solved before a vehicle can be made fully autonomous, through accurate perception of the environment followed by planning and robust control [2].

Any invention is triggered by three factors: the need to change, the ability to change and the willingness to change.

- A need to reduce the number of traffic accidents is one of the key motivators to develop autonomous vehicles. Traffic accidents in the US and around the world cause alarming numbers of fatalities each year. According to a report [5], there were close to 32,744

fatalities and 2.34 million people injured in the year 2014 from crashes on US roadways. The number rose to 35,092 fatalities and 2.44 million injured in 2015. A large number of traffic accidents are attributed to human error. These errors include inability of the driver to evade a potential threat due to negligence, lack of alertness or distraction and lack of information available for decision making such as blind spots. Other factors that fuel the need for autonomous vehicles include possibility of reducing road congestion and the time-saving factor as the time spent in driving could be devoted to other productive tasks [2]

- The ability to drive this change comes through advancements in technology. With developments in the capabilities of sensors such as Cameras and LIDAR, the environment can be sensed and perceived to a high degree of accuracy. Similarly, processors such as GPUs are increasingly developed to support computations for decision-making in autonomous vehicles. and These advancements have paved way to making the concept of self-driving vehicles, practical
- The willingness to change towards automated driving among the general public is dependent however on several factors. In an internet-based user acceptance survey [6] conducted with 63 questions with 5000 responses collected from 109 countries, 33% responded that fully automated driving would be favorable and 69% said that fully automated driving will reach a 50% market share between now and 2050. Main concerns among public included software hacking, legal issues, safety and transmission of vehicle data.

In a common traffic scenario with multiple vehicles, communication among the vehicles could result in an increased situational awareness about the environment. This wireless information sharing among vehicles is proposed to be implemented using Dedicated Short Range Communication Equipment(DSRC) [4] or through cellular communication technology [7] across a local area. This form of information sharing can happen among the vehicles (V2V), as well as information could be exchanged with sensors placed on infrastructure adjoining the roads (V2I). The information shared by a vehicle can pertain to the state of either the vehicle itself or to its surroundings.

The common autonomous vehicle perceives the immediate world around it, through sensors such as Cameras, LIDAR (Light Detection and Ranging) and RADAR(Radio Detection and Ranging). The objective of perception is to successfully develop a contextual scene

understanding of the environment. LIDAR and RADAR provide range-based data in the form of distances to objects around the vehicle and are insensitive to environmental illumination whereas cameras provide visual information that aid in the classification of objects in the surroundings, but are inconsistent in different illumination conditions. Both types of information are complementary have a unique role to play in environmental perception and both sensors have their own advantages and disadvantages [2]. The fact that data from one sensor alone might not always be enough to form a complete picture of the environment necessitates a concept called sensor-fusion. Sensor-fusion techniques [8] are an active area of research where the data obtained from multiple sensors are fused to derive an estimate of the state to a greater level of accuracy than the estimate obtained by any single sensor alone.

On-board vehicle sensors have a limited field of view. Obstacles on the road such as buildings and other vehicles, prevent a vehicle's sensor from completely perceiving the nearby environment. A technology that expands this field of view and perception of autonomous vehicles would be highly beneficial as it would empower the vehicle with more information that could be utilized towards making better driving decisions [7]. Communicating on-board sensor information across multiple vehicles through Vehicle-to-Vehicle Communication, 'Cooperative Perception' is achieved. The goal of this work is to develop and test a system that helps obtain information about the environment that is otherwise unavailable to the sensor on-board a vehicle in specific scenarios by turning every vehicle's sensor on the road into a source of information for every other vehicle. A framework is needed for such effective fusion of information that is computationally light for real-time processing, as well as feasible for communication. The challenges and advantages of such a technology are highlighted in this thesis.

1.2 Problem Formulation

In the context of Intelligent vehicles, the concept of Cooperative perception aims to enable vehicles to share local perception data with other vehicles and infrastructure through wireless communication. Since each vehicle has multiple sensors, each with its own coordinate system for measurement, combined with the fact that the state of the vehicle and environment is dynamically changing, a framework is needed to communicate and utilize such gathered information. One of the prime motivators for this technology is the need to maximize the

line of sight and field of view of every vehicle. Each vehicle has a limited field of view, which is compounded by occlusions and blind spots. Cooperative Perception aims to expand this field of view. Such an increasing situational awareness results in better driving decisions.

The problem of Cooperative perception is broken down into two sub-problems namely estimating the relative position and orientation of the various information gathering agents i.e vehicles, and a means to localize objects in the field of view of the vehicle that detects them. Thus, each of the two sub-problems, namely relative pose estimation and visual object localization using a monocular camera, needs to be individually addressed to form the entire system. In this work, the relative pose estimation problem is addressed through monocular camera relative pose estimation technique based on feature-points matching [9], [10]. The object localization problem is solved by placing the detected objects in real-world coordinates using knowledge of perspective projection through a technique called inverse perspective mapping [11]. GPS information is proposed to be augmented with the relative pose estimate for better localization. This pose estimate is used for coordinate transformation of object positions in the new vehicles' field of view, thus providing information that would not have been possibly gathered by the sensors on the vehicles' own field of view.

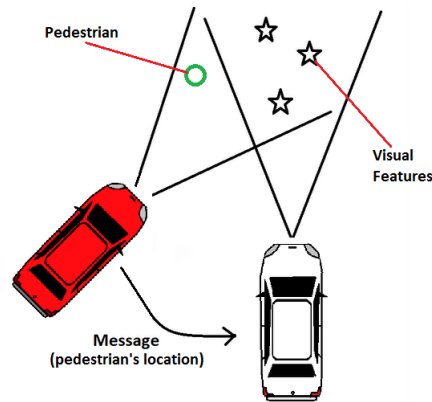


Figure 1.1: An illustration of Cooperative Perception in vehicles

1.3 Goals and Contributions

The goal of this thesis is to develop a method for Cooperative Perception for Mobile Robots that emulate a driving environment and experimentally evaluate the performance of the developed method through a series of driving scenario test cases in which cooperative perception could prove effective to the safety and comfort of driving. The goals of this research work have been realized through the following contributions of this project:

- A study of existing literature and research work in the field of cooperative perception in autonomous vehicles and a compilation of potential advantages and challenges of such a system
- An algorithm that achieves cooperative localization in specific scenarios using visual and inertial data from other vehicles
- A technique to localize objects using a monocular camera using inverse perspective mapping
- Experimental results obtained from a multi-robot testbed used to test the proposed algorithms, followed by results and inferences

1.4 Scope of Work

The following points thus clearly outline the scope of this work.

- This work deals with the concept of cooperative localization through vision-based sensors and inertial sensors. It is to be noted that the same may also be achieved using range-based sensors such as LIDAR and RADAR through other techniques [12]
- Cooperative Perception extends the field of view for vehicles. Cooperative driving is achieved as a result of this extended line of sight. [13]. The main focus of this work is cooperative perception, although the experiments conducted to demonstrate the working of the method, also include suitable response generation in the case of a potential hazard

- The communication of such information and fusion into another vehicle’s field of view is beneficial to both manually driven vehicles as well as autonomous vehicles. In Autonomous vehicles, increased situational awareness is crucial to safe driving.

1.5 Thesis Outline

This thesis has been divided into seven chapters. Chapter-1 defines the problem statement after providing a basic background to the area of research and concept, explains the motivation to pursue the research, as well as outlines the scope of the work. Chapter-2 is a review of contemporary literature and state-of-the-art research in the field of Cooperative Perception, where several benefits and challenges of the technology are discussed. Chapter-3 explains the pipeline for inter-camera relative orientation estimation based on feature-matching and epipolar geometry, followed by IMU based orientation measurement, both of which are used to achieve cooperative localization in terms of orientation. Chapter-4 describes the technique based on inverse perspective mapping to obtain an approximate location of an obstacle detected using a monocular camera. Chapter-4 also includes the system-level overview of the proposed cooperative perception algorithm. Chapter-5 illustrates the capabilities of the multi-robot mobile testbed developed to test the proposed system. Chapter-6 presents corresponding experiments designed, along with results and inferences. Chapter-7 concludes the work and lists possibilities for improvement and future direction of the research.

Chapter 2

Literature Review

The previous chapter introduced the idea of Cooperative perception, provided a background and defined the problem statement and the scope of this work. This chapter provides an overview of Mobile Robotics, Autonomous Vehicles and Connected Vehicles followed by a survey of contemporary research in the field of Cooperative Perception. The methods proposed by various works are compared and contrasted. The end of this chapter presents a list of benefits and challenges of this technology.

2.1 Mobile Robotics

A Robot is defined as a machine that is designed to carry out a specific set of tasks based on a program. Robots may be classified based on several criteria, but one such classification involves a division based on whether the robot is fastened to the ground or if it is mobile. Mobile robots may again be classified based on whether they are legged mobile robots or wheeled mobile robots. This work pertains to the specific category of wheeled mobile robots. These robots sense and perceive the environment, take appropriate decisions based on preprogrammed instructions conditioned on sensor data, and finally execute actions based on decisions taken. Each of the above tasks poses several challenges, some of which are explained below. [14], [15] and [16] offer a holistic compendium of research in the field of Mobile Robotics.

Robotic tasks can be broadly classified as Perception, Localization, Planning, Learning

and Control. In order to navigate through its environment safely, a robot needs to know about its immediate surroundings and the objects in it. Sensing the state of the nearby environment is commonly referred to as robotic perception. Several sensors are available to obtain information about the environment. Common Sensors employed include Vision Sensors or Cameras, RADAR (Radio Detection and Ranging), LIDAR (Light Detection and Ranging) and Infrared Sensors. Sensor-Fusion is achieved by combining the output from multiple sensors to produce a better estimate of the quantity being measured. Perception can be related to the term Robot Mapping, in which the robot creates a model of the environment or a map which representing its knowledge of what the environment looks like.

While the robot makes an attempt to estimate the state of the environment, it is essential to measure its own state comprising of its position and orientation, along with their derivatives within the map. This is termed as localization. In the process of estimating the pose i.e position and orientation, one option is to rely on pose fixing systems which provide the pose directly like the GPS (Global Positioning System) or by measuring distances to known landmarks, and another approach is to rely on sensors such as the IMU (Inertial Measurement Unit) which provide the derivatives of the pose and this process is called dead reckoning. Both methods are combined to provide an accurate estimate of the ground truth. Since Localization and Mapping are dependent upon one another, a technique called SLAM (Simultaneous Localization and Mapping) aims to create a map of the environment though available localization estimates as well as improve localization using developed maps. As the measurements from the localization and mapping data contain noise, several filters are used in state estimation such as Kalman Filter and its variants, Particle Filter, Grid-based methods and Information Filter, depending on the application.

The activity of planning helps consider multiple courses of action based on cost and also predict the consequences of such actions using a model of the robot and the environment. The problem of navigation from the starting point to the destination can be broken down into multiple problems, such as go-to-goal and avoiding obstacles on the way. A Controller is employed to control the actuators to achieve the motion plan. The robot may employ several types of linear and non-linear controllers depending on the application. In wheeled mobile robots, it is necessary to maintain both longitudinal and lateral control for a faithful following of the desired trajectory, and the controller may take into account just the kinematics or in some cases the dynamics of the robot for this purpose.

With the growing complexity of tasks assigned to robots, it is not feasible to program

responses to every situation and therefore the robot must be able to learn from its experiences, and with the progress of time, acquire capacity to become better at its tasks, in a way similar to how humans learn. Machine Learning is a sub-field within Artificial Intelligence that deals with such techniques, where the system becomes better on some metrics at particular tasks based on its past experiences, thus mitigating the need to program responses to every possible scenario.

Mobile Robots have been extensively used in Defense based applications in the form of Unmanned Ground Vehicles in unstructured terrains, as well as they have found applications as rovers for surveying different planets and satellites in space. They are being designed to serve as house assistants to help with chores in the house, and in the commercial domain they are used in manufacturing plants and industries to automate the production process as well as navigate and work in conditions which would be termed as harmful to humans such as mining and excavation. This work however, deals with a specific growing application of mobile robots, namely Autonomous Vehicles.

2.2 Autonomous Vehicles

Autonomous Vehicles face similar challenges as of mobile robots such as environment perception, localization and mapping, planning and decision-making, and motion control. [17] provides an insight and background in common Autonomous Vehicle Algorithms. However, the dynamics of such vehicles on the road is complex compared to ordinary ground robots. Also, the speed at which vehicles move is considerably higher and vehicles have specific objectives and have to conform to traffic laws. Advancements in technology in recent years have empowered the change to autonomous driving. Several automotive manufacturers and technology firms, along with several universities are pioneering research on Autonomous vehicles [2]. [17] also contains an account of software architectures of the Autonomous Vehicles used in the DARPA Grand Urban Challenge of 2007 [18].

[1] contains an account of the history of autonomous vehicles. The modern-day autonomous vehicles are capable of driving completely on their own without manual intervention, however in specific scenarios. Occasionally, such vehicles indicate that the driver take over, when the situation cannot be clearly assessed. Environmental perception of the immediate surroundings is achieved through sensors such as LIDAR, RADAR, Ultrasonic sensors

and Cameras. Sensor-fusion is employed to form a unified belief about the state of the environment by utilizing information from more than one sensor. Perception tasks specific to Autonomous vehicles on the road include pedestrian and vehicle detection and tracking, traffic sign and signal recognition, lane detection and lane marker detection. Perception in autonomous vehicles is explained in detail in a later section within this chapter. Localization is primarily done through the Inertial Navigation System (INS) which fuses estimates from the GPS and IMU. Wheel encoders are also used. Superimposing locally perceived data using on-board sensors onto the global map also paves way for improving the global map. Path-planning and Decision-making happens in multiple stages where mission/route planning helps in obtaining a path from the start point to the destination minimizing distance or time. The mid-level planner is a behavioral planner that deals with lane-keeping, lane-changing, parking, obstacle avoiding and recovering from any deviations to course [17]. The low-level module is the motion controller which implements the plan created by the mid-level planner, and is responsible for longitudinal and lateral control of the vehicle and ensures that the vehicle faithfully follows generated trajectories. Thus, a combination of sensors, decision-making algorithms and actuators help in replacing the human driver and enable the vehicle to navigate on its own.

2.3 Connected Vehicles

With the advent of autonomy in vehicles, communication among the agents on the road has been perceived to be crucial for decision making. Short-range communication helps safety related to navigation whereas wide range communication helps traffic management. This form of vehicle-to-vehicle communication results in connected vehicles. Through wireless communication, vehicles are expected to be able to transmit information about their own state such as position and heading as well as information about their surroundings over a short range. A brief account of the proposed systems for inter vehicular communication is presented from literature as follows.

[4] explains vehicular communication systems in detail, along with their requirements and challenges. Any safety concern of the vehicle is usually associated with the immediate proximity or vicinity of the vehicle, whereas traffic concerns might involve a larger range. In dense traffic, even this vicinity might pertain to nearly a hundred vehicles on the road. Sessions need to be established and changed at high speeds which might prove challenging.

[4] mentions the concept of 'Spheres of Influence' in which multiple spheres of communication might exist among vehicles at different ranges, for different priorities. The sphere of influence of interest in this thesis is the local sphere where safe navigation is of concern. [4] mentions that regional communication using fixed terminals does not propose the best solution, and instead, localized broadcast transmissions could be used in a non-networked scenario. Each terminal would not have a network address, but messages are simply transmitted and can be received by any other agent that deems it relevant. However, the sender has no way of ensuring that the receiver has received it. [4] discusses density and bandwidth effects as well. Vehicular communication is compared and contrasted with regular cellular phone communications. The latter had a low bandwidth requirement for voice and infrequent users during any given time, however with the rise of smartphones, the requirements have risen. Vehicular communication faces similar problems of a large number of users requiring the bandwidth of the communicating channel at the same time. A rough image of vehicle density is presented as 100 vehicles per kilometer per lane. Wireless systems can have varying ranges, for example Wi-fi (802.11 a/b/g) can have a range less than 50 meters, DSRC (Dedicated Short Range Communication Systems) can have a range of about 500 meters (although reliably between 100-200 meters) and 3G cellular can have from 1000 to 5000 meters. DSRC is mostly preferred as it provides 10 Mbps with a bandwidth of 6 and 156 kbps per vehicle [4]. The interval after which a vehicle can repeat its message comes to about 26ms to 0.6s. [7] explains various technologies considered as potential candidates for vehicular communication systems which can be broadly categorized as Cellular (3G HSDPA and 4G LTE) and Wi-Fi (802.11n and 802.11g). An experiment was conducted by [7] with 2 nodes at a distance 10 meters apart which is the usual inter-vehicular distance and these candidates were evaluated for performance. Wi-Fi had a lesser delay but performance was found to decrease with increasing distance. 3G and 4G had relatively greater delay but the effect of distance was absent. Communication delay was however found to become uncertain in both cases, as the size of data increased. [7] mentions that 802.11n Wi-Fi works well in urban roads up to a distance of 30 meters and that future work would be aimed at evaluating 802.11p. It also mentions problems in inter-vehicle communication such as the 'Vehicle Identification Problem' and the 'Circular Reasoning Problem'.

[4] proposes a few examples in which DSRC might be useful, such as communicating road hazards such as potholes ahead, communicating vehicle state information to road-side units or infrastructure which can then pass this onto a central repository and distributed to other subscribers, and also communication of traffic signal information where the vehicle

gets to know in advance, how long it would be before a traffic signal at an approaching intersection changes its state. [4] also provides an overview of cooperative driving in which it is proposed that vehicles can share information about their current state to other vehicles. This can even include their future trajectories such that each vehicle ensures that it is not intersecting with another vehicle's future course. Another class of communications is proposed where vehicles receive information and then rebroadcast it to other vehicles. Also, [4] mentions how high-level requirements for vehicular communication such as range, amount of data to be sent, timing requirements and reliability define physical characteristics of the communication channel.

This work explores such a scenario of short-range communication of vision-based information which helps localize vehicles within a vicinity and communicate information about perceived entities along with their locations to nearby vehicles. Means of perception of these entities is described in detail in the next section.

2.4 Perception in Autonomous Vehicles

The sensors used for perceiving the environment, that are particular to Autonomous Vehicles are Vision-Sensors, LIDAR and RADAR. Each of these sensors have their own unique position in sensing. Infrared sensors suffer from a serious disadvantage of not being able to differentiate between objects that irradiate similar thermal energies. The readings would also depend upon the colors, shades and surfaces, even though the range may remain the same. It is also difficult to use under the sun, as sunlight contains infrared light as such. It is also sensitive to weather conditions and lacks accuracy. With passive infrared, if the human body temperature matches that of the ambient temperature, then detection falters. RADAR provides only limited lateral spatial information as well as it's resolution is low at large distances. LIDAR could be employed to construct detailed 3D maps of the environment but it has its own limitations. LIDAR units are too costly to be fitted onto every vehicle, has low ability of discrimination, possesses recognition latency and would at times result in clustering errors as well [17]. LIDAR is also prone to failure in detecting transparent objects. RADAR and LIDAR both suffer from interference issues when there is a unit placed on top of every vehicle, when the signal from one system would be captured by another system leading to erroneous measurements. Owing to several such limitations, vision-based sensors are gaining preference to shoulder most perception tasks.

Computer-Vision has evolved a separate branch of research that involves interpretation of images to derive meaningful information. Computer Vision Research is fast evolving in recent times due to the improving computational capability and introduction of GPUs. This branch includes various sub-fields such as image processing, perspective geometry, object detection and tracking, semantic scene understanding and in recent times has evolved to include advanced topics such as action recognition and machine learning.

Cameras capture visual information about objects in the environment that aid object detection and classification. Such classification is crucial for decision making. Vision sensors suffer from their own set of limitations including more computational effort, low sensing depth and a lack of availability of spatial information. In spite of these disadvantages, their unique position in the sensing process is indisputable.

[19] lists several advances made in several areas of computer vision related to autonomous vehicles including stereo vision, visual odometry, object detection and tracking. A subset of object detection is pedestrian detection for which a number of techniques exist. The introduction of Deep Learning and Convolution Neural Networks has improved object detection standards by a great margin. [20] illustrates computer vision research related to autonomous vehicles including lane detection, vehicle detection, traffic sign detection and recognition and also outlines a few open problems. [21] and [22] provide the necessary background in the field of computer vision.

[23] is a comprehensive collection of the latest literature on computer vision for automated driving. Beginning with a review of the various computer vision datasets available for training, it discusses advances in literature made on image representation (such as superpixels and stixels), object detection and tracking, semantic image segmentation, reconstruction, motion and pose estimation, scene understanding and end-to-end learning. [2] is another comprehensive survey on techniques used in perception, planning, control and coordination of autonomous vehicles. The following subsection provides insight into literature on a specific area of perception research in connected and autonomous vehicles, namely cooperative perception.

2.5 Cooperative Perception

In a multi-robot environment, information and decision sharing among the robots could help extend field of view of each robot. A framework needs to exist for such information sharing in order to for the information to be relevant. There are three types of sensor fusion as proposed by Durrant-Whyte based on the relation between input data sources [8]: Competitive/Redundant, Complementary and Cooperative. Competitive sensors measure the same property and are fused to increase the confidence in the measurement. Complementary sensors measure different properties that are uniquely available to those sensors. Cooperative sensors use information from multiple sensors to derive information that would not have been possibly obtained by any single sensor alone. The research discussed in this work possesses features of complementary and cooperative sensor fusion, although it has been termed as cooperative perception in literature [7].

In the context of autonomous vehicles on the road, these cooperative perception techniques need to be refined to suit on-road applications. Mathematically, cooperative perception is a way of map merging following pose estimation [24]. Having illustrated the growing importance of vision-based sensors over sensors of other modalities, this work aims to achieve cooperative perception using vision-based sensors. The following sections contain a review of certain cooperative perception techniques employed in robots and vehicles. Subsequently presented, is a list of benefits and challenges of cooperative perception as a technology.

2.5.1 Ground and Aerial Robots

A record of Cooperative Perception techniques in aerial vehicles is presented in [25]. Cooperative perception techniques primarily evolved with increasing applications of the Unmanned Aerial Vehicles (UAV). UAVs are used in search and rescue operations and are usually employed in large numbers as swarms. In such applications, it is beneficial to collaborate, share information and accomplish the task together. Cooperative perception, primarily being sensor fusion across multiple robots may be chosen to be centralized or decentralized or a hybrid of both. Such a decision is taken based on factors such as physical communication layer of communication, local processing power, level of autonomy, structure of the environment and the nature of the tasks to be accomplished. Each robot maintains its own belief about the state of the environment. Either raw sensor information or processed belief

may be communicated. [26] mentions, how the prime requirement for building a unified map would be a good localization of the agents with respect to each other. [26] makes a successful attempt to perform cooperative localization using a camera and an IMU. The camera has been used to perform a 6 Degree of Freedom pose estimation using feature correspondences up to a scale in translation and by using IMU estimates, we can recover the actual translation, thereby obtaining the exact relative pose between the two cameras. The method assumes that the camera-IMU calibration is known and that the cameras on both robots have considerable and sufficient overlapping fields of view.

2.5.2 Ground Vehicles

In the area of cooperative perception among unmanned ground vehicles pertaining to autonomous vehicles, an idea was proposed in [27] and an in-depth analysis has been presented in [24]. This paper proposes multi-modal cooperative perception relying on multiple sensors including vision-based sensors and LIDAR. In that work, different sensory modalities are used to fuse information pertaining to different aspects of the environment. Lane markings are merged after applying a technique known as Inverse Perspective Mapping (IPM), as they are on the ground. Any other object which has a non-zero height is shared through scan matching techniques of laser points. These laser scan points are also used for relative pose estimation. More information about the study on cooperative perception from the same author can be found in the following references. The impact of cooperative perception is explained in the article [7]. Similarly, [13] and [28] explain a mirror-neuron based approach to cooperative driving, utilizing the cooperative perception system developed in the earlier work. [29] offers further insight on the same study and [30] is a related work using the same technique. [12] is one of the most recent surveys on cooperative perception techniques relevant to autonomous vehicles and describes the fundamentals and feasibility of the different approaches through comparison and contrast. [12] delves into a comprehensive analysis of vehicle relative positioning methods, and also includes a review of vehicular communication systems.

2.5.3 Advantages and Challenges of Cooperative Perception

Cooperative Perception has several advantages. The concept by itself, aims to extend line of sight and field of view to obtain more information from the environment than that can be achieved by a single sensor alone. Such an extended field of view may be beneficial at times where there is an occlusion preventing the availability of a complete view of the environment to a vehicle. These occlusions may be buildings and infrastructure, people or other vehicles themselves. Owing to this extended line of sight, there is an increase in situational awareness, which would lead to better overtaking decisions. Also, blind spots are reduced and hidden obstacles can be avoided. There is an improvement in traffic safety, traffic routing efficiency and long-term path planning. As multiple vehicles view the same object, an attempt to recreate the 3D structure of the object can be made, which can aid object detection and classification.

Having highlighted some of the advantages of such a collaborative approach to perception, some of the challenges of such a technology identified from literature are highlighted below:

- Localization through data association and map merging across a long range through multiple agents leads to an error propagation that makes the measurement drift from its true value by a huge margin.
- Rumor propagation is another challenge that could prove detrimental. When a false information is communicated by one vehicle through a network, it propagates through multiple possible channels of vehicles and gets amplified by the time it reaches the vehicle under study. Having obtained the same information from multiple sources, the vehicle could become confident in the belief of false information [25]
- Complexity grows with the fact that the external world has dynamic entities such as vehicles and pedestrians and any estimate of the state of the environment is useful instantaneously or over an immediate short span of time, however grows obsolete over time given the fact that the map keeps changing
- Cooperative perception systems also need to deal with conflicting information. The sensor arrangement needs to be chosen to be complementary where all sensors are needed to form the correct picture of the environment, competitive where multiple sensors

obtain the information about the same entity and the most trusted sensor reading is accepted or collaborative where they might obtain mutually exclusive information to fuse together to form a better estimate

- Since cooperative perception relies on inter-vehicle communication, the limitations of communication such as latency and bandwidth need to be considered
- The right level of information sharing needs to be addressed in such a collaborative network. The frequency of information sharing is thus a deciding factor where at a low frequency, the system is not robust and at a high frequency, messages are lost
- Sensor configuration on each vehicle is not the same as one another. Thus, this has to be taken into account each time information from another vehicle is fused into the frame of reference of the vehicle under consideration. Not just the sensor configuration but also the fusion technique has the possibility of being different for different vehicles. As a result, each vehicle will have its own version of the ground truth map, resulting in varying driving decisions [25]
- For range-based sensors, the outputs are scan points. For the same object, the scanned points will be different for the data obtained by different vehicles from different sides of the object and so matching these points becomes difficult
- For vision-based map merging, an overlapping field of view and accurate data association are essential. It is necessary to ensure that when two robots communicate about entities observed, they are referring to the same part of the world [25]. This is also termed as data association or image registration
- Another challenge that is specific to vision-based cooperative perception is that range-based sensors provide data about the state of the environment as spatial coordinate point clouds whereas vision sensors provide perspective views of the environment as pixel intensity information. Map merging is easier when spatial information about the map is available and therefore map merging of visual information is more challenging. Vision-based map merging also suffers from the fact that vision data processing is time-consuming.

Chapter 3

Relative Pose Estimation

The proposed cooperative perception technique is a two step process. The first step towards sharing such information across vehicles would be to establish relative localization so that the information obtained in the reference frame of the sender may be transformed and represented in the reference frame of the receiver, whose field of view has now been extended due to cooperative perception. This chapter describes a technique to suit the application, comprising of a vision-based pipeline to extract Euler angles which achieves pose-fixing, followed by IMU-based dead reckoning. Throughout the process, position information as on the ground plane is assumed to be known to a high degree of accuracy.

3.1 Vision-based orientation estimation

Visual sensors are used in this system to achieve pose fixing. A standard computer vision technique exists for obtaining relative camera pose, given images from two cameras that share a common field of view. The process is briefly described as follows: Feature points are extracted from the images obtained using the two cameras, following which local descriptors are computed. Using these descriptors vectors, the features are matched between the two images automatically through a suitable feature matching algorithm. Given sufficient number of feature matches, a technique such as RANSAC could be used to eliminate outliers resulting in a few good matches. Using these good matches of feature points, the Fundamental Matrix is extracted. If the camera intrinsic parameters are known, then the Fundamental Matrix could be used to compute the Essential matrix. The Essential matrix may be decomposed to

obtain the feasible combination of rotation matrix and translation vector. It is to be noted that translation can be obtained only up to scale. This means that it is not possible to say whether the two cameras are close by and looking at a small object or far apart and looking at a large object. However, the rotation matrix is obtained without ambiguity. This rotation matrix could be used to obtain the Euler angles in turn. The Euler angle for yaw is assumed to provide the relative orientation between the two cameras, or in this context, between the two vehicles. The subsequent subsections explain this process in detail

3.1.1 Feature Correspondence Matching

In an image, a feature point is a point of interest. There are certain regions within an image which can be uniquely distinguished for computational purposes. Corners of objects in images make good features. A flat patch region within an image, on the other hand is not a good feature as it cannot be uniquely identified across multiple images of the scene or object. An edge is also indistinguishable along one direction. A corner is however unique. Similarly, in computer vision, there are several types of feature points. The process of detecting such points in an image is called feature detection. A good feature is one that is invariant to image rotation, scaling and brightness. In the current system, a feature known as SURF (Speeded up Robust Features) is used, which is a robust and fast version of the popular SIFT feature (Scale Invariant Feature Transform). To find scale space, SIFT uses the Laplacian of Gaussians whereas SURF uses a simple box filter, and is thus faster.

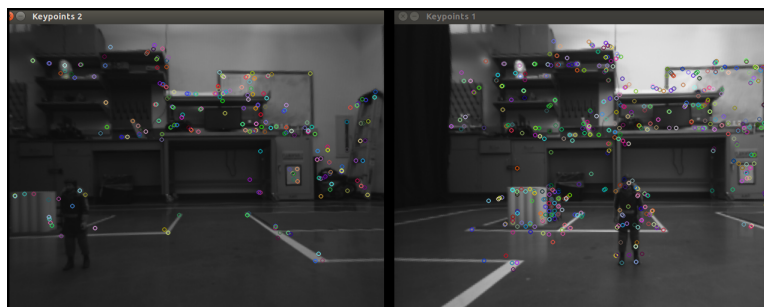


Figure 3.1: Image feature points

Extracting such features from an image is followed by computing a local descriptor for the feature. The descriptor is an identifier for the feature. Neighborhood pixel information around the feature is extracted and a feature descriptor vector is formed. The size of this

vector depends on the type of descriptor used. But the descriptor can be thought of as a lower dimensional representation of a higher dimensional entity. Image features of an object could be used to identify the object in different images, by comparing these descriptors. Some of the popular feature detectors and descriptors in literature are SURF, SIFT, BRISK, ORB, MSER and FAST [22].

Features in two images can be matched using vectorial distance measures applied to their corresponding descriptors. Common feature matching techniques include Brute Force (BF) matcher and FLANN (Fast Library for Approximate Nearest Neighbors) based matcher. Subsequently, a technique such as RANSAC could help eliminate outliers, resulting in select good matches which are used in estimating the Fundamental Matrix. The absence of outliers is crucial to accurate estimation of the Fundamental Matrix.

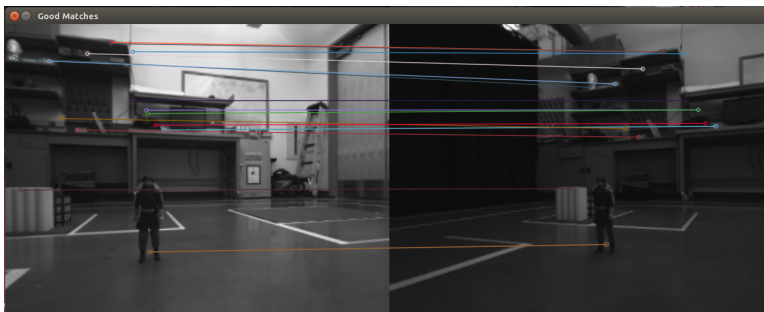


Figure 3.2: Feature matching

3.1.2 Extraction of Fundamental and Essential Matrix

The relationship between points in the 3D world and their corresponding representation on an image in the pixel space is given by the following camera equation:

$$\mathbf{x} = \mathbf{K}[\mathbf{R}|\mathbf{t}]\mathbf{X} \quad (3.1)$$

where \mathbf{x} is the homogeneous image coordinates represented by a vector and \mathbf{X} represents the vector of Cartesian coordinates of the 3D point in the real-world. \mathbf{K} is the Intrinsic Camera matrix. \mathbf{R} is the 3x3 Rotation matrix and \mathbf{t} is the 3x1 translation vector and together, they are augmented to form the Extrinsic matrix. There are three co-ordinate spaces: the

real-world space, the camera space and the image space. The Extrinsic matrix takes a point from the real-world 3D space to the camera space. The next operation is a transformation from the camera space to the image space which is accomplished through a multiplication by the K matrix. K is given by

$$K = \begin{bmatrix} \alpha_x & \beta & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.2)$$

where α_x and α_y represent the lens scaling factors along x and y directions respectively, β is the skewness factor and x_0 and y_0 are the image center coordinates. Homogeneous coordinates are used in projective geometry instead of Cartesian coordinates. Using homogeneous coordinates, even points at infinity can be represented with finite coordinates. In homogeneous coordinates, all points in the real-world that lie on a light ray that reaches the camera center are represented using the same set of coordinates and form a single pixel on the image plane.

When there are two cameras observing an object with overlapping fields of view, the geometric relations can be described with a specific geometry called epipolar geometry. This is used in stereo vision, where there are two cameras, usually separated by a fixed distance called baseline. Consider two cameras observing a common scene. The position of one of the camera's optical center in the other camera's image plane is called an epipole. The line joining any point in the 3D world and the optical center of a camera, forms a line on the image plane of the other camera and is called an epipolar line. The plane containing the 3D point and the optical centers of the two cameras is called the epipolar plane. [22] and [21] offer comprehensive insight into epipolar geometry.

The Fundamental matrix is a 3x3 matrix of rank 2, which captures the geometry between two views. If a point in 3D space X is captured as x in one camera's image and x' in the other camera's image, then the image points satisfy

$$\mathbf{x}'^T F \mathbf{x} = 0 \quad (3.3)$$

It is to be noted that each x and x' are in homogeneous coordinates such that $\mathbf{x}=(x, y, 1)^T$ and $\mathbf{x}'=(x', y', 1)^T$. The Fundamental matrix is calculated from solving the above shown

equation for a set of n point matches. Properties of the Fundamental Matrix can be found in [22] and [21]. Multiple algorithms exist for determining F including the normalized 8-point algorithm and the 7-point algorithm. [21] has a list of recommendations of using particular algorithms for specific scenarios. Thus, the Fundamental matrix is computed.

In Epipolar geometry, a Matrix called the Essential Matrix E relates the 3D point in one camera's coordinate system to another. The Essential matrix E is computed from the Fundamental Matrix and Camera Intrinsic Matrix K as follows:

$$E = K'^T F K \quad (3.4)$$

where K and K' are the camera intrinsic matrices of the two cameras. If the intrinsic camera parameters such as focal length and image center are the same between the two cameras, then $K=K'$.

The Essential matrix is primarily a product of the skew symmetric translation matrix and the rotation matrix.

$$E = [t]_x R \quad (3.5)$$

It is to be noted that the Essential matrix has only five degrees of freedom although it's constituents R and t have three degrees of freedom each. This is due to ambiguity in scale. A singular value decomposition of the Essential matrix can be used to obtain the rotation matrix R and the translation vector t . E can be decomposed as:

$$E = S R \quad (3.6)$$

where S is a skew-symmetric matrix and R is the rotation matrix

Four possible combinations of S and R will be obtained. From these, the pair of S and R for which the the reconstructed points are in front of the two cameras is chosen as the valid pair of translation and rotation. Finally, the rotation matrix R can be converted into corresponding Euler angles, of which our interest lies in the yaw.

3.2 IMU-based orientation estimation

An IMU consists of accelerometers, gyroscopes and magnetometers. The accelerometer measures specific force or linear acceleration, the gyroscope measures angular velocity and the magnetometer acts like a compass and gives the direction of magnetic north. The Gyroscope outputs angular velocities or rates at which the body is rotating along each of the three axis. A mere integration of the angular rates would not result in the Euler angles. This is because there are multiple coordinate frames with which each angular rotation happens and each angular rate has to be transformed to that particular frame in which it is valid. The following equation gives the transformation from the gyroscope outputs i.e angular rates to the Euler angle rates.

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin(\phi)\tan(\theta) & \cos(\phi)\tan(\theta) \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi)/\cos(\theta) & \cos(\phi)/\cos(\theta) \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (3.7)$$

where p,q and r are the outputs of the gyroscope, i.e the body-frame angular rates and ϕ , θ and ψ are the Euler angle rates. At this point, just integrating the Euler angle rates would give the Euler angles and the parameter of interest is the yaw Euler angle. If this Euler angle were received at every subsequent time-step following the pose fixing, dead reckoning would be possible, as the biases between the two IMU have been zeroed out now.

3.3 Cooperative localization system

The system proposed works as follows: Each vehicle transmits its GPS and IMU data as well as image features and descriptors identified from its frontal field of view. [7] and [24] explain about the feasibility of transmitting low sized compressed images over V2V. In this case, only the features are transmitted. Every vehicle receives this information sent by every other vehicle in the vicinity. Each data message is timestamped. The ego vehicle attempts to check for sufficient number of matches between received features and self-observed features captured at the same time instant. If sufficient matches are found, then the relative orientation is obtained through the standard computer-vision pipeline described in this chapter. The relative position is assumed to be available through absolute position

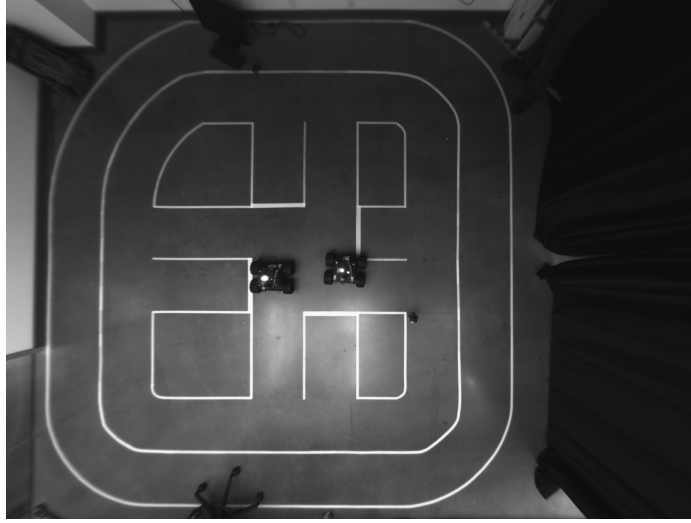


Figure 3.3: Robots observed by the overhead camera

estimates of both vehicles from GPS and an absolute orientation estimate of the ego-vehicle in the global frame. Thus, a relative pose-fix is established, in which both relative position and orientation are known. This ensures that the IMU is tared and the biases are zeroed out. Subsequently, the vehicle listens to IMU information from the other vehicle, along with GPS information and updates its estimate of the other vehicle's pose. The drift in the inertial sensors is not expected to affect the orientation estimate over a short timespan.

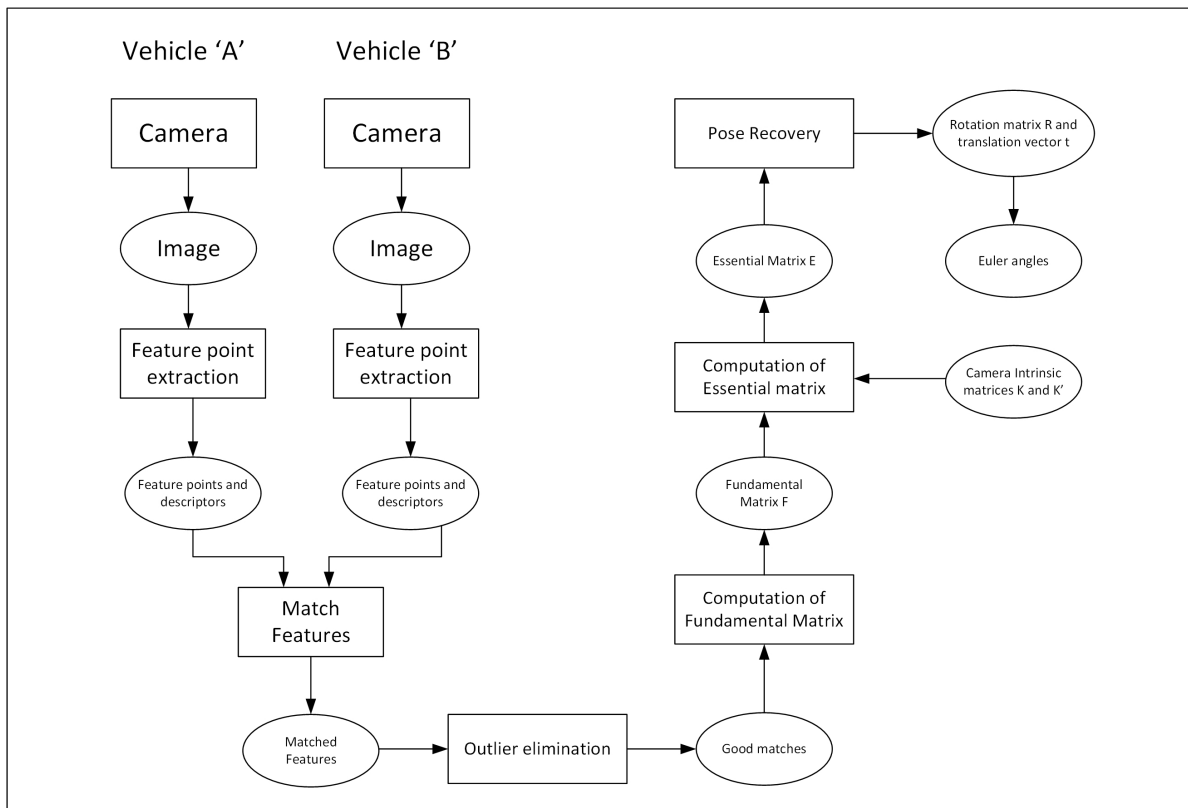


Figure 3.4: Relative camera orientation estimation

Chapter 4

Object localization on the Inverse Perspective Map

Once a pose correspondence is established between two vehicles, objects observed by one vehicle can be transmitted into the reference frame of the other vehicle. In this chapter, a technique is described to approximately localize an object in the vehicle's field of view using a monocular camera using inverse perspective mapping, coupled with object detection. Once this information is obtained, a simple coordinate system transformation would make this information available to the other vehicle, thus enabling it to make cooperative driving decisions. Also included in this chapter, is an overview of the entire cooperative perception system.

4.1 Inverse perspective mapping

The image obtained by the camera is termed as perspective view. This is a projective transformation as there is a projection or a mapping of the 3 dimensional scene onto a 2 dimensional sensor array inside the camera. Perspective transformation thus transforms a 3D object to a 2D representation. [22] and [21] contain a detailed description of projective geometry. The inverse of the perspective transformation operation helps to obtain the possible 3D positions of the objects given their 2D pixel location in the image. This is achieved by a technique known as Inverse Perspective Mapping (IPM). IPM helps obtain a view of the image as it would have been observed from an overhead camera or bird's eye view. Since

there is loss of information of one dimension in this reconstruction, an additional constraint is required for IPM, which is a plane onto which the image is to be projected.

A formulation of the IPM algorithm adopted from [11] is briefly explained as follows:

The Camera Equation is given by:

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \mathbf{K} \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \mathbf{K} \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix} \quad (4.1)$$

The Camera matrix transforms a point in the 3D world onto a 2D pixel point on the image plane. Here, \mathbf{K} is the intrinsic camera matrix.

$$\mathbf{K} = \begin{bmatrix} \alpha_x & \beta & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.2)$$

The above equation gives us perspective projection. The following expression is used to obtain the pixel coordinates from the homogeneous coordinates:

$$x = u/w \quad \text{and} \quad y = v/w \quad (4.3)$$

For simplification, the intrinsic camera matrix is multiplied with the rotation and translation vector and represented as follows:

$$\mathbf{K} \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} \quad (4.4)$$

$$\mathbf{K} \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix} = \begin{bmatrix} n_1 \\ n_2 \\ n_3 \end{bmatrix} \quad (4.5)$$

While the perspective projection equation obtains the resulting image coordinates from a

point in the real world with spatial coordinates X, Y and Z , the inverse perspective map recovers X, Y and Z from the image coordinates x and y . The additional constraint is that the image is assumed to be obtained from a single plane and this is necessary to compensate for the loss of information along one of the dimensions. This plane is defined as follows:

$$aX + bY + cZ + d = 0 \quad (4.6)$$

In this case, the plane on which the image is projected is the ground plane. The perspective projection equation may in turn be modified as

$$w \begin{bmatrix} x \\ y \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} & 0 \\ m_{21} & m_{22} & m_{23} & 0 \\ m_{31} & m_{32} & m_{33} & 0 \\ a & b & c & d \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} + \begin{bmatrix} n_1 \\ n_2 \\ n_3 \\ 0 \end{bmatrix} \quad (4.7)$$

Rearranging this equation, the inverse perspective mapping equation [11] can be arrived at, which transforms a point from the image homogeneous coordinates to the real world.

$$\begin{bmatrix} X \\ Y \\ Z \\ w \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} & -x \\ m_{21} & m_{22} & m_{23} & -y \\ m_{31} & m_{32} & m_{33} & -1 \\ a & b & c & 0 \end{bmatrix}^{-1} \begin{bmatrix} -n_1 \\ -n_2 \\ -n_3 \\ -d \end{bmatrix} \quad (4.8)$$

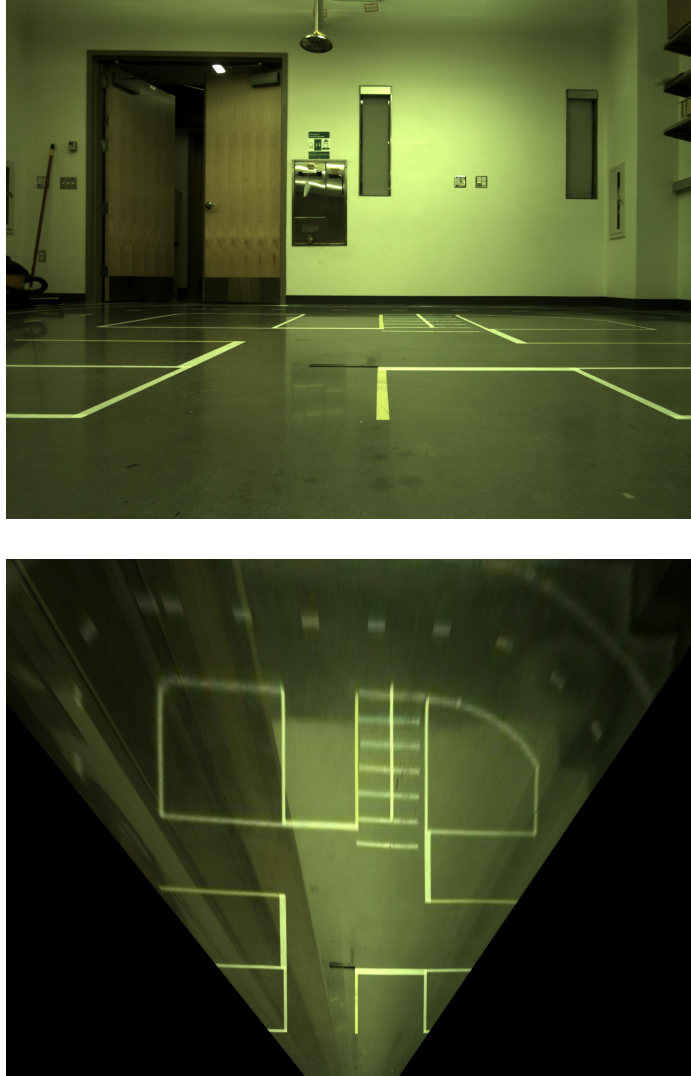


Figure 4.1: Perspective image and Inverse Perspective image

[11] provides a comprehensive overview of literature on the Inverse perspective mapping technique in Computer Vision. Early works can be found in [31] and [32], and some recent studies on Intelligent Vehicles continue to use the method [33], [34], [35]. Some of the applications for which IPM is a preprocessing component in Autonomous Vehicles include lane and road marking detection and pedestrian/obstacle detection. IPM thus occupies a significant place in vision-based perception, as each pixel represents the same distance in real-world units after the perspective effect is removed [11]. [36] uses a technique to use an EKF (Extended Kalman Filter) to track vehicles on the Inverse Perspective Map,

but it has been discussed in the case of a two-camera stereo-vision system. In the case of Autonomous Vehicles, a monocular camera is advantageous over a stereo-vision system, as real-time operation at high speeds is crucial and the processing load needs to be low. Several works in literature related to Autonomous Vehicles have cited the use and importance of IPM. In the process of Inverse perspective mapping, the entire image is assumed to have been obtained from the ground plane in the real world, and so this mapping projects this back onto the same plane in the real world. This helps in the recovery of road geometry including lane markings. [24] mentions how IPM can be used to recover knowledge about the road, and how pedestrians and vehicles can be visualized using other sensor modalities.

[37] elaborates on the interplay between scale of objects, their position in the 3D world and the camera viewpoint. It explains how object detection can be improved through contextual understanding of the scene. To state an example, if the ground plane in the image is identified, then the detection of a pedestrian in the image at a point above the ground plane is not possible as that would imply that the pedestrian were present above the ground plane in the real world. Using contextual information that pedestrians and other traffic entities rest on the ground plane, which is the road in this case, and assuming known camera position on the vehicle, the location of the objects can be estimated.

4.2 Object Localization

The Inverse perspective map projects the observed entities onto the ground and does not reverse the perspective effect altogether. Since pedestrians do not have zero height, they cannot be directly represented on the Inverse perspective map. However, the point where they rest on the ground can be transformed through this inverse map since it is a representation of a point on the ground in the real-world. This is obtained from the location of the bounding box denoting the object in image coordinates. The technique for obtaining the position of these objects starts with detecting the objects with the bounding box. The assumption made in [37] is also made in this work regarding the objects of interest such as pedestrians resting on the ground plane. A bounding box detects the position of the object on the image. Several techniques exist in literature and practice for the detection of vehicles, pedestrians and other objects [23]. While the argument of [24] is that obstacles cannot be represented on the inverse perspective map as they possess a height, and [11] also introduces a multi-modal technique to selectively mask the obstacles to obtain a clear inverse perspective

map, it is known that the bottom point of the obstacles contact the ground. As a result, if we detect the object with a bounding box, the bottom side of the box must approximately correspond to its position on the ground. Thus, the image coordinates corresponding to the mid-point of the lower side of the bounding boxes is transformed through mapping. The method is computationally light since we transform only a point and not the entire image through the transform. The box might not exactly fit the object, in which case the image coordinate might not always correspond to the actual 3D point on the ground representing the object's position in the real-world. However, this error can be eliminated through tracking the object in multiple frames [38]. Knowing the intrinsic and extrinsic camera calibration parameters, the inverse perspective map may be calibrated to scale, and thus the technique allows for estimating the object's state as position x and y in the frame of reference of the ego-vehicle.

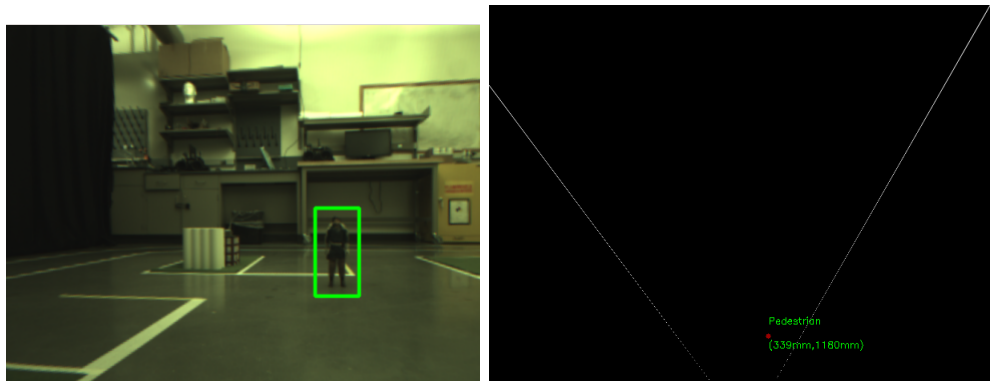


Figure 4.2: Pedestrian detection (left) and localization (right)

When a vision-based relative pose correspondence is established between two vehicles that have a common field of view, the relative orientation of the other vehicle is constantly estimated by the ego-vehicle with the help of the gyroscope readings from the two vehicles, and GPS information between the two vehicles helps obtain positional information. Thus, the ego-vehicle has an estimate of the other vehicle's relative position and orientation over time. When the other vehicle observes an entity such as a pedestrian, it transmits the approximate location of the detected entity which can be represented through a coordinate transformation in the ego-vehicle's frame of reference, thus guiding the ego-vehicle's driving decisions.

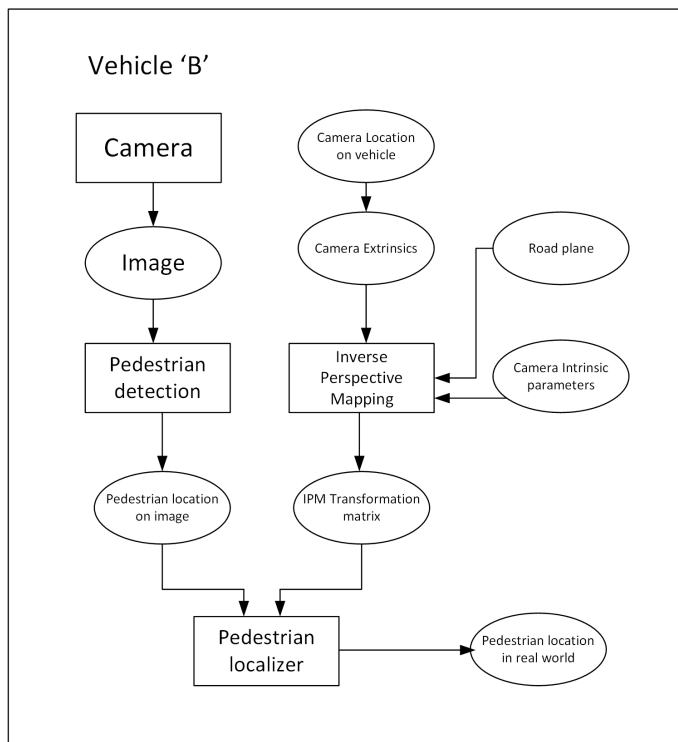


Figure 4.3: Pipeline for pedestrian localization

4.3 System Overview

The working of the entire cooperative perception system is explained as follows:

- Ego-vehicle 'A' observes the front camera image and extracts feature points and descriptors
- Ego-vehicle 'A' receives a message from another vehicle B. The other vehicle 'B' shares a common visual field of view with the ego vehicle 'A'. The message contains the camera matrix K of the vehicle, feature points and descriptors extracted from the image observed by 'B' and the data from the IMU of 'B'
- The Ego-vehicle 'A' matches its own features with the features of 'B'. From the set of matched features, the Fundamental Matrix F is extracted. Then using the camera intrinsics K and the Fundamental Matrix F , the Essential Matrix E is computed. Then

E is decomposed to the Rotation matrix R and the translation vector t upto scale. From R , the Euler angles are computed which gives the relative orientation

- Subsequently, IMU data received from vehicle 'B' is used by vehicle 'A' to update relative orientation over time. Vision-based pose fixes are not established at each time instant post the initial, as transfer of such data is communication-wise expensive
- Relative position information is assumed to be known through a combination of absolute position estimates from GPS and an absolute orientation estimate of the ego-vehicle
- Vehicle 'B' spots an entity such as a pedestrian and obtains the entity's approximate location in its field of view and transmits this location information in the message
- Ego vehicle 'A' receives the location information from vehicle 'B', transforms the location through the relative position and orientation estimate computed by vehicle 'A' to arrive at the location of the entity in the frame of reference of vehicle 'A', thereby achieving cooperative perception. Subsequently, it generates a response to the hazard and evades it.

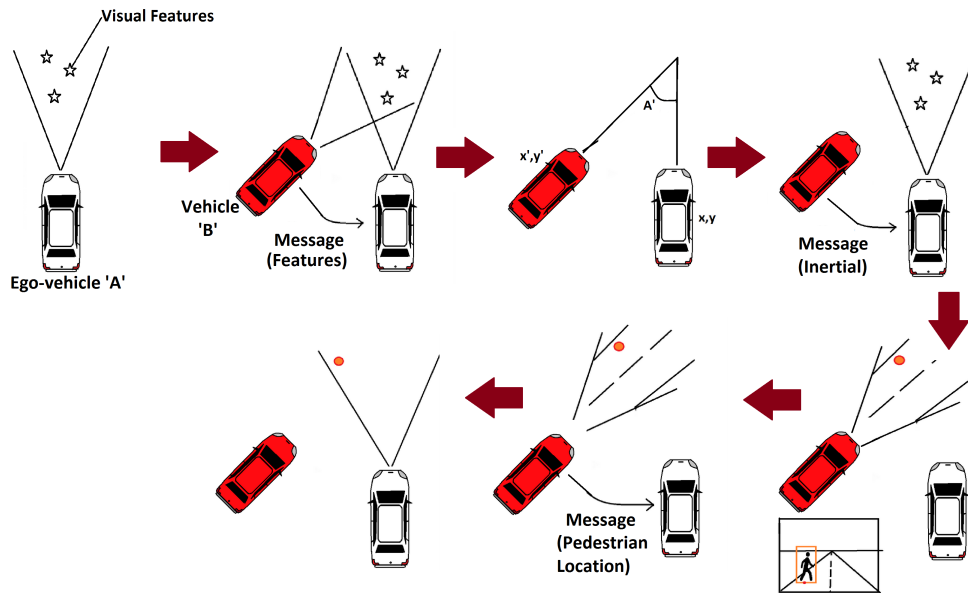


Figure 4.4: Cooperative perception - System overview

Chapter 5

Mobile Robot Testbed Development

To demonstrate the working of the proposed algorithm, an experimental evaluation was conducted on mobile robots. This chapter describes the capabilities of the testbed that was developed for the experiment - a wheeled mobile robot called Wifibot along with the programs and the architecture developed to form the Cooperative Perception System. In the next chapter, the experiments pertaining to Cooperative Perception are described, along with an illustration of results obtained.

5.1 Wifibot

The Wifibot is programmed to mimic an autonomous vehicle. A few of the features of an autonomous vehicle were developed as individual programs and deployed on the Wifibot. The platforms used for the development of robot software include C/C++ and Python. A Middle-ware called Robot Operating System (ROS) was used as a platform to develop the individual modules. The reader is encouraged to refer [39] and [40] for a detailed introduction to ROS. OpenCV libraries were used to accomplish computer vision-based tasks.

The Wifibot is a four-wheeled differential drive mobile robot that consisted of motors driving individual wheels, tethered to a motor controller board. This motor controller was in turn connected to a PC. The processor communicated with the motor controller board through RS232 serial communication. The motor controller sent signals to 4 brushless motors fitted with hall encoders that measured the speed of motor rotations. Further, Infrared

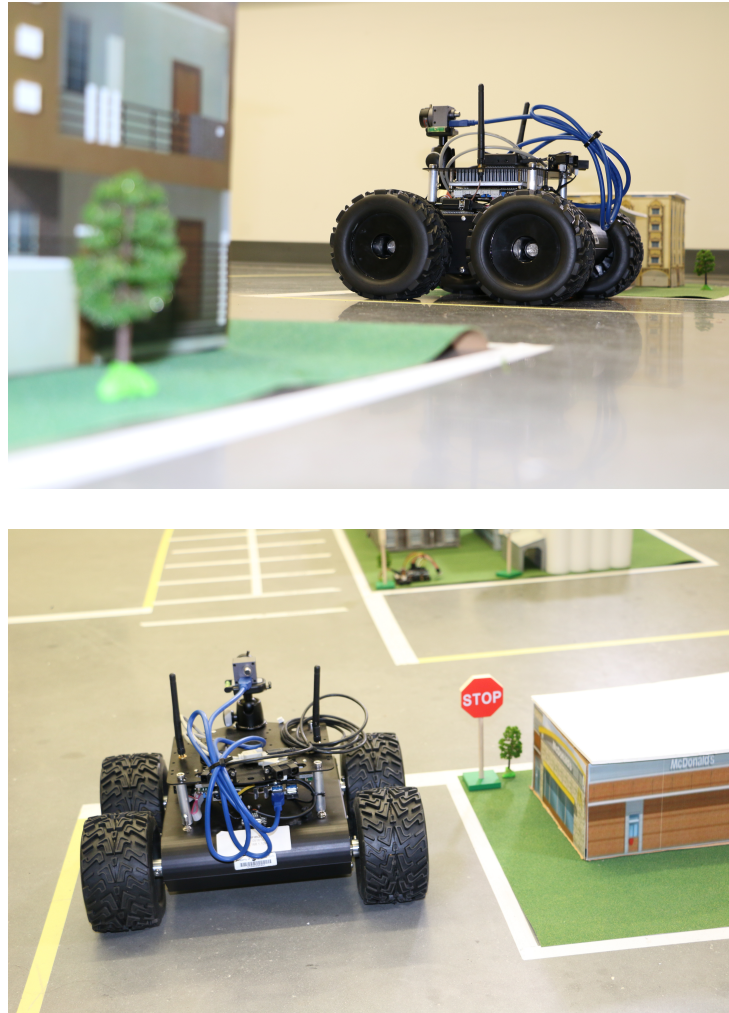


Figure 5.1: Wifibot Robot

sensors were attached to the same controller board shared by the motor system. The Wifibot also contained a GPS unit, an IMU and a Camera. GPS data was not available indoors, because of which an alternative indoor positioning unit was set up in the lab space. The camera on the Wifibot provided 12-bit images. This image was converted to an 8-bit image before processing. The Inertial Measurement Unit (IMU) on the Wifibot consisted of an accelerometer, gyroscope and magnetometer and was capable of directly providing the Euler angles as output. The camera, IMU and GPS were directly connected to the processor through USB. The following tables list the specifications of the sensors of the Wifibot

Table 5.1: Camera Specifications

Property	Value
Camera Model	Leopard Imaging LI-USB30-P031XB
Dynamic Range	12-bit
Sensor	Aptina MT9P031 5 Megapixel sensor
Format	1/2.5"
Resolution	2592x1944
Pixel format	YUV
Sensitivity	1.4V/lux-sec
Lens Model	Ricom TS0416F.IR (MP)
Focal Length	4mm
Aperture	1:1.6
Field of view	D112° x H89.6° x V67.2°

Table 5.2: IMU Specifications

Property	Value
Model	YEI USB 3-Space Sensor
Communication Interfaces	USB 2.0, RS232 Asynchronous Serial
Serial baud rates	1200 921600 selectable, 115200 (default)
Orientation range	360° about all axes
Orientation resolution	<0.08°

Table 5.3: Motor Specifications

Property	Value
Motor Type	Brushless Motor
Phase	2
Voltage	12
Rated Power	16 W
Rated Speed	3000 RPM
Rated Torque	0.051N.m
Ratio	28.5:1
Rated Speed from gearbox	105 RPM
Rated Torque from gearbox	1.1 N.m

Table 5.4: Infrared Sensors Specifications

Property	Value
Model	SHARP GP2Y0A02YK
Detection Range	20-150cm
Operating Supply Voltage	4.5 - 5.5 V

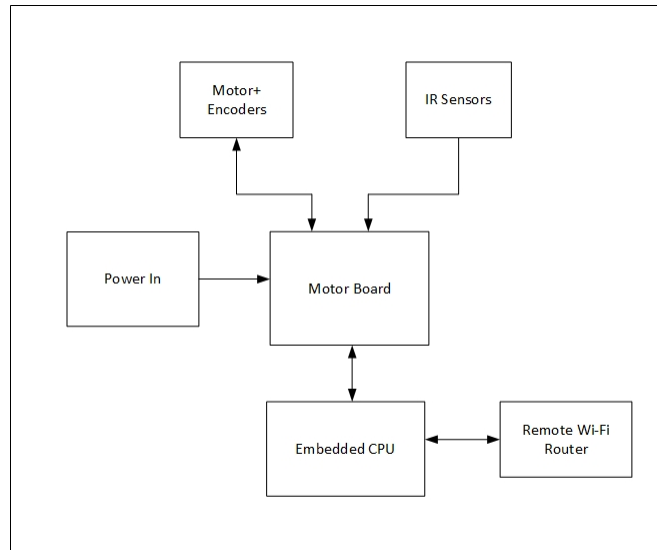


Figure 5.2: Wifibot Hardware Architecture [Source:Nexter Robotics]

5.1.1 ROS Architecture

The software architecture of the robot defined independent modules with specific functionalities and the data transfer between them. Having a well planned architecture ensured that there was no conflict of decision between two independent processes and that the input of one process was suitably used by its successor. The modularity of the model also enabled modifying a part of the system without affecting the rest of the system. ROS allowed the development of such an architecture. Individual programs called nodes which are C/C++ executable files or python scripts, communicated with other nodes using messages and topics. These messages were published onto topics, from which any number of nodes could subscribe and listen to those messages. Nodes could be independently started and shut down, enabling

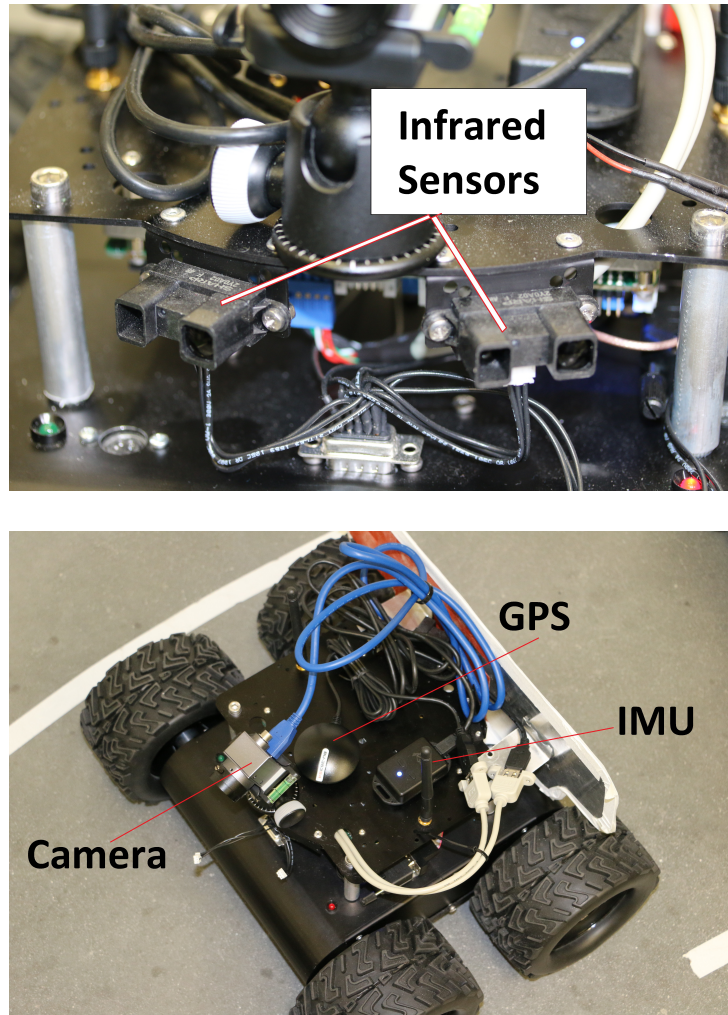


Figure 5.3: Sensors on the Wifibot

the study of the entire system in the availability and unavailability of certain functionalities. The flow of data could also be monitored through process monitoring tools exclusive to ROS. ROS nodes running on multiple machines were controlled by the ROS Master running on one of the machines. In this case, the ROS master running on the ego-robot communicated to the nodes running on both robots as well as the ground station through a dedicated Wi-Fi network. Standard ROS packages were installed and utilized.

5.1.2 Indoor Positioning System (IPS)

Since the Global Positioning System (GPS) only functioned outdoors, an indoor positioning system using an overhead camera was installed within the lab space. This constituted a USB camera with a fisheye lens. The fisheye lens ensured a wide field of view. To locate the position of the robot, an LED was installed on top of the robot and this was detected by the overhead camera after performing image thresholding and blob detection operations on the obtained image. The following are additions made to this existing system as part of this research work.

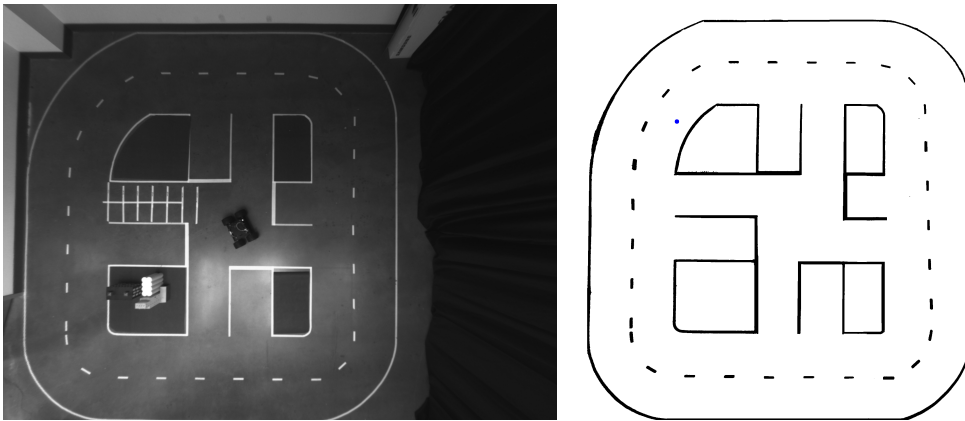


Figure 5.4: Overhead view of the Indoor track (left) and Detected Wifibot Location superimposed on a binary image of the track (right)

A radial correction factor was applied to the resultant position to get the exact position of the robot because the LED was installed at a certain height from the ground, but undistortion of the fisheye image projected the position of the LED on the ground. Using the principle of similar triangles, accounting for the height of the installed camera and the LED light, a correction factor was applied depending on the radial distance of the robot from the center of the image. The images from the overhead camera were also used for the ground truth evaluation of the proposed cooperative perception algorithm.

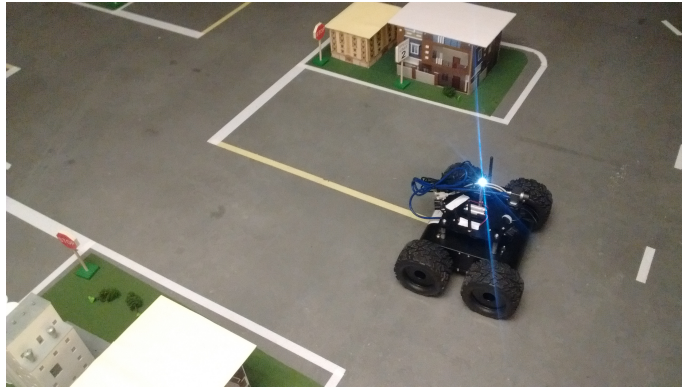


Figure 5.5: Indoor track and LED on the Wifibot

5.1.3 Tele-operation and low-level control

Tele-operation is the process of remotely controlling electronic machines. In this case, signals were sent to the robot to move from one location to another. The output of tele-operation was a velocity command. This velocity command was translated to RS232 serial commands and sent to the motor by the low-level controller. The low-level controller could also be commanded by preset velocity commands. Manual tele-operation was achieved in this case using a Joystick and a Steering Console. Commands from such devices included throttle control (forward and reverse) or speed control, braking and steering. The robot moved and turned using a differential drive mechanism with both wheels spinning in the same direction but at different speeds.



Figure 5.6: Steering and Joystick - Teleoperation

5.1.4 Stop Sign Detection

A node was developed to perform stop sign detection from visual input. If a stop sign was detected by the front camera, then a command was issued to the low level controller to immediately stop. The technique used to detect the stop sign was the benchmarked algorithm for object detection based on a cascaded classifier and haar-like features. The algorithm used for this multi-stage cascaded object detection is described in detail in [41].

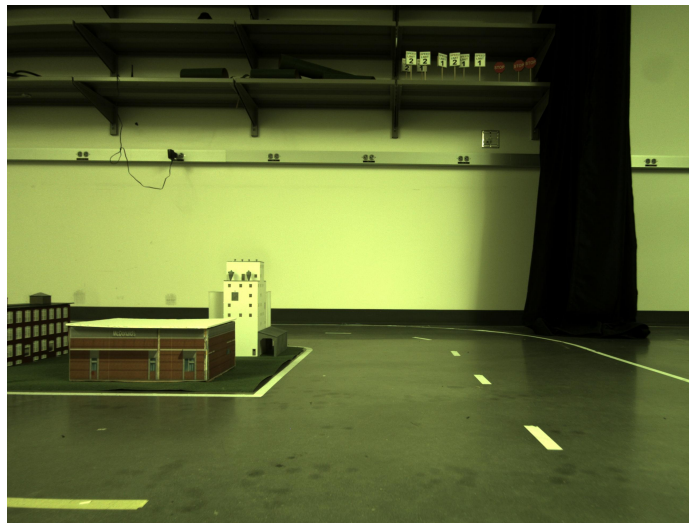


Figure 5.7: Front camera image observed by the wifibot

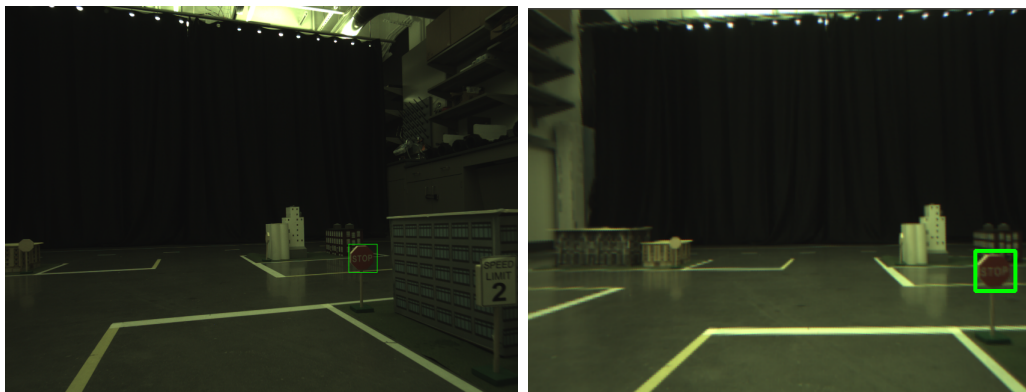


Figure 5.8: Stop sign recognition

5.1.5 Simple Emergency braking

A simple emergency braking feature was added to the Wifibot through Infrared sensors. The IR sensors were capable of measuring a single value that was proportional to the distance of the object in front of them. Thus, these values were calibrated to real-world distance units and if the Wifibot came within a certain distance to any obstacle, then a stop command would be issued to the low-level controller to immediately stop the Wifibot.

5.1.6 Pedestrian Detection

Pedestrian detection was achieved using the HOG (Histogram of Oriented Gradients) based pedestrian detection algorithm. The HOG is a feature vector which utilizes the shape of humans to classify sample images as containing humans and non-humans. A sliding window was moved across the image and if a human was detected in that window, then the same would be highlighted with a bounding box. The popular HOG-based pedestrian detection algorithm was proposed by [42].

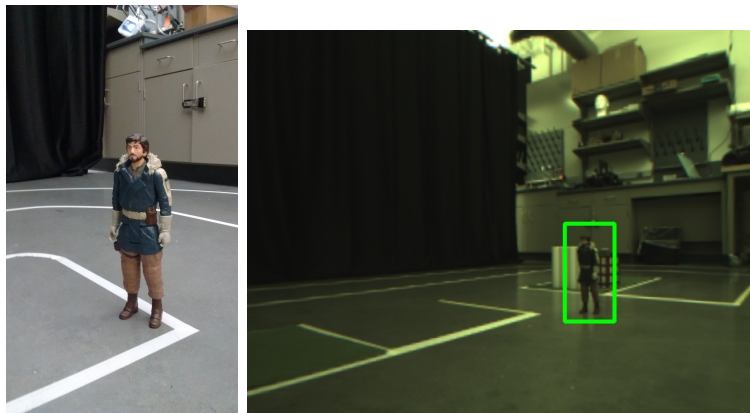


Figure 5.9: Pedestrian toy (left) and Pedestrian detection result(right)

5.1.7 Lane Detection

A survey of lane detection techniques is presented in [43]. Lanes are usually detected through visual sensors. Most methods perform edge detection as the first step. In this process, the edges in the image are extracted. An edge is any boundary where a sudden

change in intensity of light is detected. For a more robust technique that can even detect curves, another technique is used. The edge image is transformed through Inverse Perspective Mapping, a homography technique that renders a bird's eye view of the scene. From this image, the pixels on the left and right half of the image are collected and a second-order polynomial is fitted onto them. These capture the shape of the lane curve. From the two lane curves, the lane mid-points are identified.

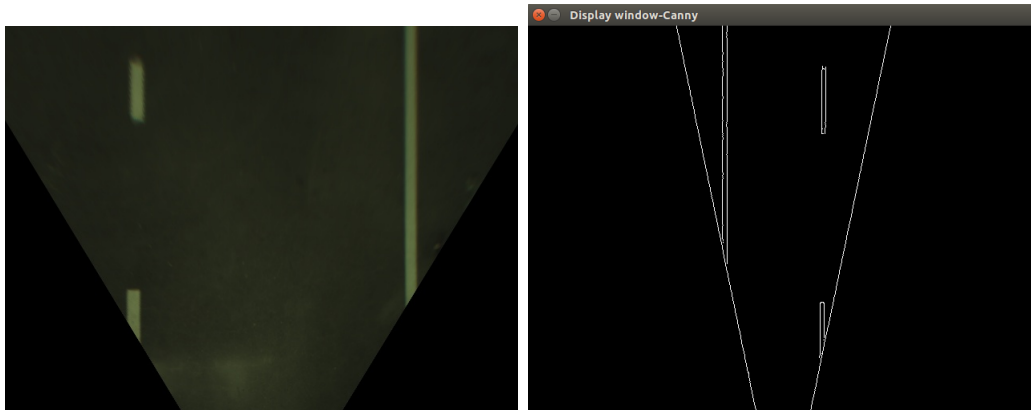


Figure 5.10: Inverse perspective map and edge image of the Inverse perspective map

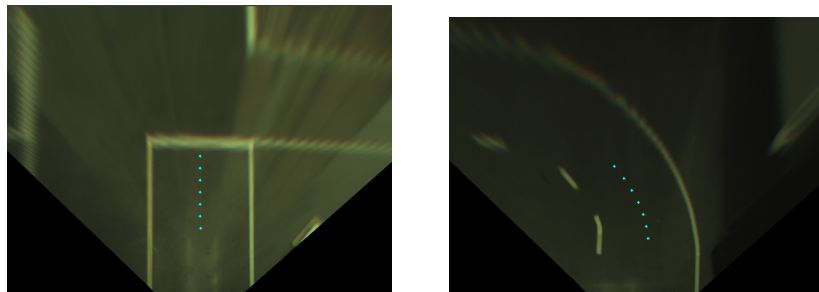


Figure 5.11: Lane detection

5.2 Cooperative Perception Algorithm

The entire cooperative perception program was divided into several programs, each running on a separate machine. The entire testbed consisted of two Wifibots and a remote ground station. Communication was enabled through a private Wi-Fi network and SSH (Secure Shell). The ROS master was initialized on the Ego-vehicle and the other two machines

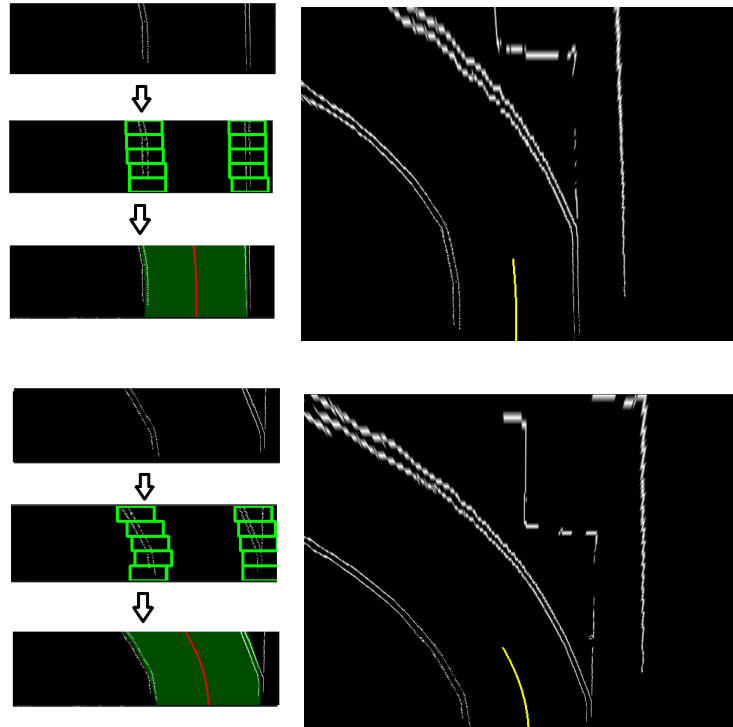


Figure 5.12: Lane detection based on polynomial fitting on the IPM edge image

were provided the URI of this ROS master. This ensured that the entire system functioned as one unit online.

The working of the system is described through each individual type of node as follows:

- **Camera node** The Camera node obtains 12-bit images from the USB camera, converts it into 8-bit images suitable for OpenCV. Then the image is debayered and then rectified using the pre-computed camera calibration matrix. Then the rectified image is published
- **IPS node** The Indoor Positioning System (IPS) node obtains 12-bit images from the overhead camera, converts it into 8-bit images suitable for OpenCV. Then the image is debayered and then rectified using the pre-computed fisheye camera calibration matrix, as the camera used a fisheye wide-angle lens. Through an intensity thresholding operation, bright locations in the image are alone retained. Through the blob detection process, the location of the LEDs are obtained and published

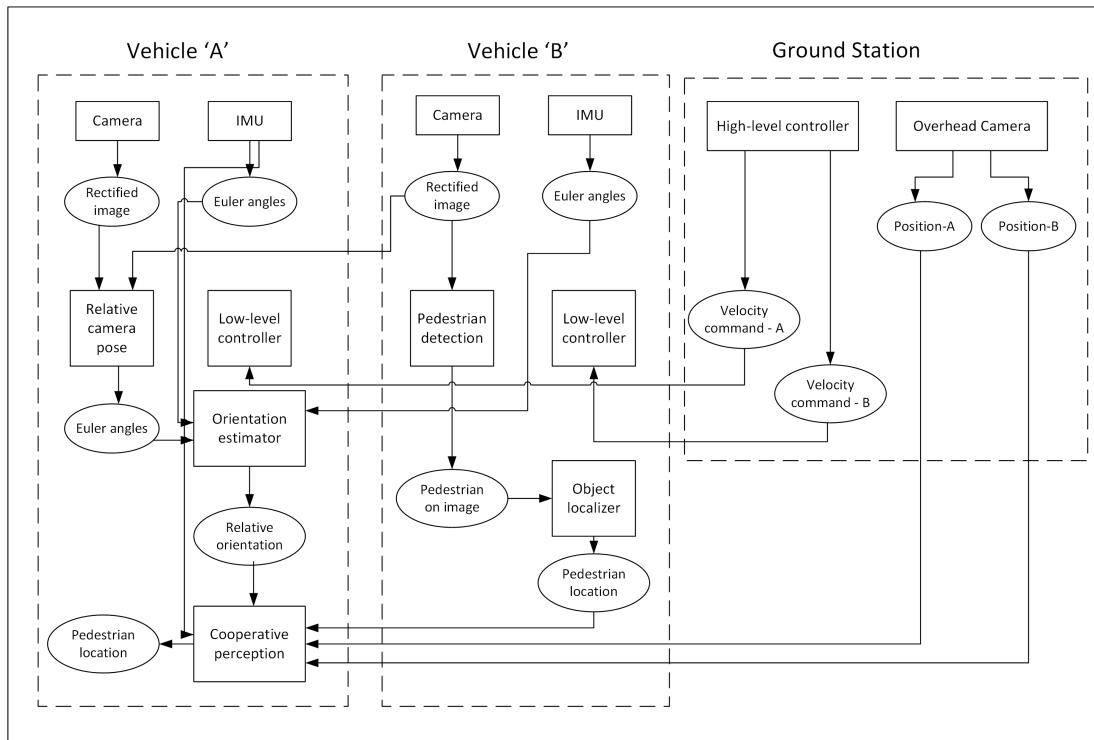


Figure 5.13: Implementation of Cooperative Perception system

- **Drive-robots node** This node issues a preset linear and angular velocity command to each robot separately for a fixed amount of time
- **Camera-relative-pose node** This node obtains rectified images from camera nodes on both the robots, extracts feature points and descriptors from each image and then runs a feature matching algorithm to compute a set of matched features. Then through distance thresholding operations, a set of good matches are found. These good matching points are used to estimate the fundamental matrix, and in turn estimate the essential matrix. A singular value decomposition of the essential matrix then yields a rotation matrix and a translation vector upto scale, the former of which can be decomposed into Euler angles. The Euler angle on the xy plane is then published as the relative camera pose
- **IMU node** The IMU node directly interacts with the IMU through serial communication and obtains the three Euler angles. It is to be noted that the IMU should be tared on the xy plane since the motion is constrained to the xy ground plane. Taring

is the process of zeroing out biases and aligning the coordinate system.

- **Low-level controller** The low-level controller obtains the command velocity from the Drive-robots node and converts into appropriate serial commands. These serial commands are then sent to the motors directly to run the robot
- **Orientation estimator node** The orientation estimator node provides a visual representation of the net resultant relative orientation of the two robots at any given instant of time. The ego-robot is set to a constant 0 degrees and the other robot's orientation relative to the ego-robot is shown. As soon as the relative orientation is obtained from the camera, the IMU biases are computed from the IMU readings and camera relative orientation and canceled out for each subsequent reading, and finally the relative orientation data is published

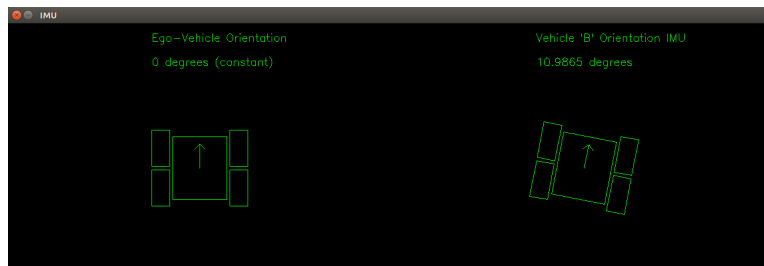


Figure 5.14: Relative Orientation - Initialized by Camera Relative pose and updated by IMU readings

- **Pedestrian detection node** This node reads the rectified camera image and detects the location of pedestrians on the image as pixels x,y. It publishes an image to visualize the bounding box as well as the bounding boxes themselves as OpenCV rectangle objects
- **Pedestrian localizer node** This node reads the bounding box data from the pedestrian detection node and transforms the bottom center point of the bounding boxes through a pre-computed 3x3 inverse perspective mapping transformation. Thus it publishes the approximate location of the pedestrian in the frame of reference of the robot in real-world units. Before publishing, it incorporates a check to see if the transformed point is indeed a point within its field of view and a point on the ground. This helps eliminate any false positive detections from being published

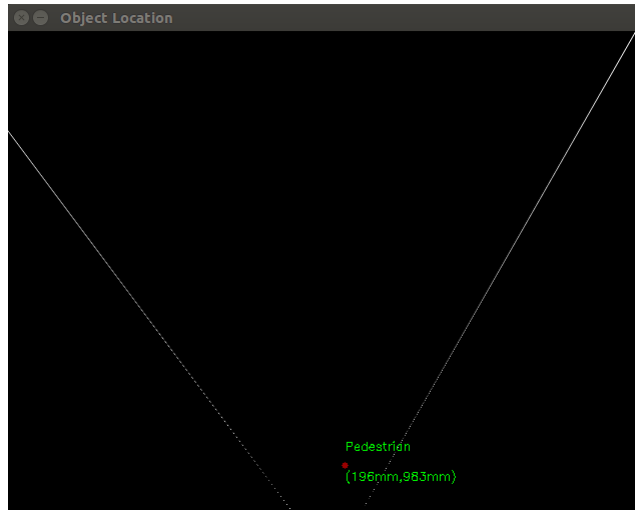


Figure 5.15: Pedestrian localization

- Cooperative perception node** The cooperative perception node is the final destination of all data. It reads from the IPS nodes, the orientation estimator node and the object localizer node. Using the former two, it obtains the necessary coordinate transformation between the two robots. Using this computed transformation, the position of the pedestrian is now obtained in the field of view of the ego-robot which did not perceive it directly, thus enabling cooperative perception

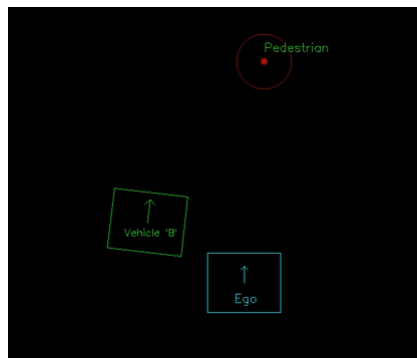


Figure 5.16: Result of Cooperative perception

Chapter 6

Experimental Evaluation of Cooperative Perception

The previous chapter explained the details of the testbed robot and its ADAS features, along with the details of the architecture of the proposed cooperative perception system on the Wifibot, along with a description of each of its individual programs. This chapter contains an account of the experiments conducted to validate the proposed technique of cooperative perception. Real-life traffic scenarios are emulated to show the working of this algorithm and the results are analyzed and presented in this chapter. Ground truth verification is performed using images obtained from an overhead camera.

6.1 Evaluation of approximate object localization using a monocular camera

Although, it is not possible to accurately recover depth/scale information from a monocular camera due to the loss in information of one dimension, the proposed object localization algorithm helps obtain an approximate location of the entity. In this case, a pedestrian toy was used to emulate a human being on the road. By using the pedestrian detection node, the human being was detected on the image plane. Knowing that the pedestrian would be situated on the ground plane, and knowing the height of the camera from the ground plane and its orientation, the inverse perspective mapping technique was used to recover the human's

approximate location. The following table shows six sample estimates from an experiment in which the camera was operated at 480x640 pixel resolution. The x and y locations are represented in the frame of reference of the ego-vehicle:

Table 6.1: Approximate Object Localization Results (distance units are in centimeters)

S.No.	Actual 'x'	Estimated 'x'	Actual 'y'	Estimated 'y'
1	40	47	100	95.2
2	0	3.2	100	94.1
3	-40	-39.3	100	91
4	40	45.9	140	123.2
5	0	4.3	140	118
6	-40	-36.1	140	120.1

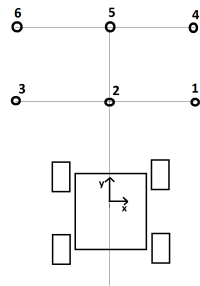


Figure 6.1: Evaluation of the approximate localization system (image not to scale)

The result of pedestrian detection is the location of the pedestrian on the image plane, denoted by the bounding box. The location of the bottom center point of the bounding box's position in real-world coordinates on the ground plane is recovered and represented to denote his location in the real world. Since the bounding box might not tightly fit the detected entity, a few pixels gap between the foot of the pedestrian and the lower side of the bounding box might result in an error in this depth estimate. It can be observed that the error in the longitudinal direction is larger than in the lateral direction. This is because the object detection technique is still laterally centered around the pedestrian. It can also be observed that this error is larger at long distances, as a few pixels at a greater depth on the perspective image represents a larger distance in real-world units due to the perspective effect. Operation at a higher pixel resolution and an accurate object detection algorithm that

gives the accurate location of the object on the image plane could help reduce this error, thereby making this technique more robust. This system helps provide an approximate object location, which is used to trigger an emergency stop signal to the vehicle or to move around the obstacle, if the detected entity is near the future trajectory of the vehicle. Stereo-cameras or range-based sensors should be used if a more accurate location of the object is desired.

6.2 Evaluation of Relative pose estimation

According to the standard pipeline from epipolar geometry, the relative orientation of the cameras on the two vehicles can be recovered by matching the same points in both images. In this case, matching points are represented by their pixel locations on the image. Matching points are detected through feature descriptor vectors which are determined based on the pixel intensities of the neighborhood pixels around a pixel of interest. Once this orientation fix was established using vision, the IMU was used to update this value over subsequent time instances. The Magnetometer on the IMU was not used owing to magnetic interferences while placed on the robot. The accelerometers and gyroscopes were used, and the internal circuitry directly provided the euler angle estimate. The IMU was tared (zero-biased) on startup, but owing to the lack of a compass, a drift was observed in the values that varied from as low as 0.05 deg/sec to 0.1 deg/sec as recorded in multiple experiments. Since the experimental trials were limited to a maximum time-span of 20 to 30 seconds, this drift was assumed admissible.

The following experiments were conducted to test the vision-based relative orientation estimation system with two stationary robots observing a static environment: Three different camera exposure values were set and under a constant background, relative orientation estimates were obtained for three different orientation angles in a 30-second time window. Note that exposure settings 1 to 3 are in the order of low to high. Also, good feature matches here refer to the inlier keypoints after the estimation of the Fundamental matrix. Outliers were occasionally observed due to bad feature matching. The results are tabulated and plotted below:

Table 6.2: Relative orientation under different camera exposure settings (units are in degrees for fields 2 through 4)

Exposure setting	1	1	1	2	2	2	3	3	3
Ground truth	0	15	30	0	15	30	0	15	30
Average estimate	1.3	11	25	2.4	11.7	25.3	1.6	11.8	25.4
Standard deviation	2.4	2.7	2.2	1.8	1.9	2.3	1.8	2.1	3.9
Average features points	196	224	223	325	344	366	400	404	446
Average good matches	86	67	49	131	114	73	146	118	77

Two different backgrounds were used to match features under a constant camera exposure setting, and relative orientation estimates were obtained for three different orientation angles in a 30-second time window. Background 1 was the same as the indoor labspace. The results are tabulated below:

Table 6.3: Relative orientation under different backgrounds (units are in degrees for fields 2 through 4)

Scene	1	1	1	2	2	2
Ground truth	0	15	30	0	15	30
Average estimate	1.3	11	25	0.5	15.5	29
Standard deviation	2.4	2.75	2.2	1.3	2.1	3.4
Average features points	196	225	223	248	261	263
Average good matches	86	67	49	100	62	35

From these two experiments, it was generally observed that a lower exposure setting would not enable extraction of a large number of keypoints in the case of the current experiment under the given setting, and would not provide a large number of feature matches. A large number of feature matches is preferable to obtain a more accurate orientation estimate. However, at high exposure, camera frame rate becomes low which was found to lead to a failure in pedestrian detection and feature-point extraction while in motion. Thus, this trade-off helped select an optimum value for camera exposure under given illumination conditions. A higher pixel resolution would also help better feature extraction and matching. However, it was not chosen as it would significantly affect computational time and communication latency. Outliers were observed due to bad feature matches, but such outliers could

be detected and eliminated by monitoring the resulting rotation matrix and euler angles of pitch and roll, the latter two of which would be unusual, not permitted by the dynamics of the vehicle.

6.3 Evaluation of potential traffic scenarios

These experiments were performed with moving Wifibots in the indoor lab environment. The details are presented as follows:

- Each robot had a LED on top. Two LEDs were placed on the ego-robot and one LED was placed on the robot ‘B’. This resulted in two blobs of different sizes when image thresholding was done on the overhead camera image to obtain position estimates of the Wifibot.
- Overhead lighting was adjusted such that the lighting above the robots was minimum to ensure robust thresholding and detection of LEDs whereas the lighting was maximum over the features that were in the field of view of the camera.
- The exposure of the front facing cameras on the robots was set to a high level to comply with the low lighting, which resulted in a low frame rate. Overhead camera images were used to find the actual location of the robots and the pedestrian and served as the ground truth. Overhead camera-based position information and ground truth information was available at an approximate rate of 1 Hz.
- Pedestrian detection was enabled on only one of the robots, robot ‘B’ and the Ego-robot had no means of detecting the pedestrian using its on-board sensors and relied on robot ‘B’ for this information.
- Relative orientation was instantly estimated by the vision-based system upon startup and this was used to note down the IMU biases on both vehicles which were then canceled in every subsequent reading to obtain the actual relative orientation, except in the case of an intersection scenario, where this initialization happened after the robots reached the intersection.
- The pedestrian toy used in this experiment occupied an approximate area of 10cm x 5cm on the ground. In each case, robot ‘B’ detected the pedestrian, computed an

approximate location and communicated this information to the ego-robot which then made decisions to evade the potential threat.

- An additional level of intelligence built onto the system ensured that the transformed point representing the pedestrian's location by robot 'B' was indeed a point on the ground and within its field of view, to help eliminate false positive detections from being transmitted
- The ego-vehicle, upon the detection of the pedestrian in its future trajectory area, turned red in color on the cooperative perception frame window to indicate a hazard warning. Along with the same, a HAZARD warning signal flashed and a collision signal was immediately sent to the low level controller to intervene
- Time synchronization was achieved among the three computer systems i.e two robots and one ground station so that every data was timestamped with the same information allowing the analysis of simultaneous events across individual programs.
- Data was collected in the form of rosbags which were later analyzed and plotted using MATLAB and its Robotics System Toolbox. The red dashed vertical line on the plots indicate the time instant when the collision warning was issued by the Cooperative perception system.

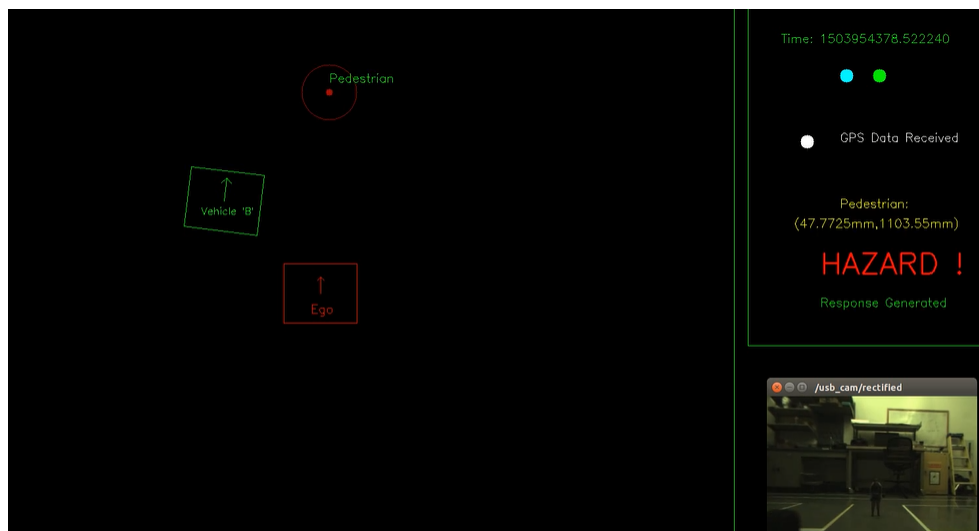


Figure 6.2: Cooperative Perception Interface - Ego Vehicle

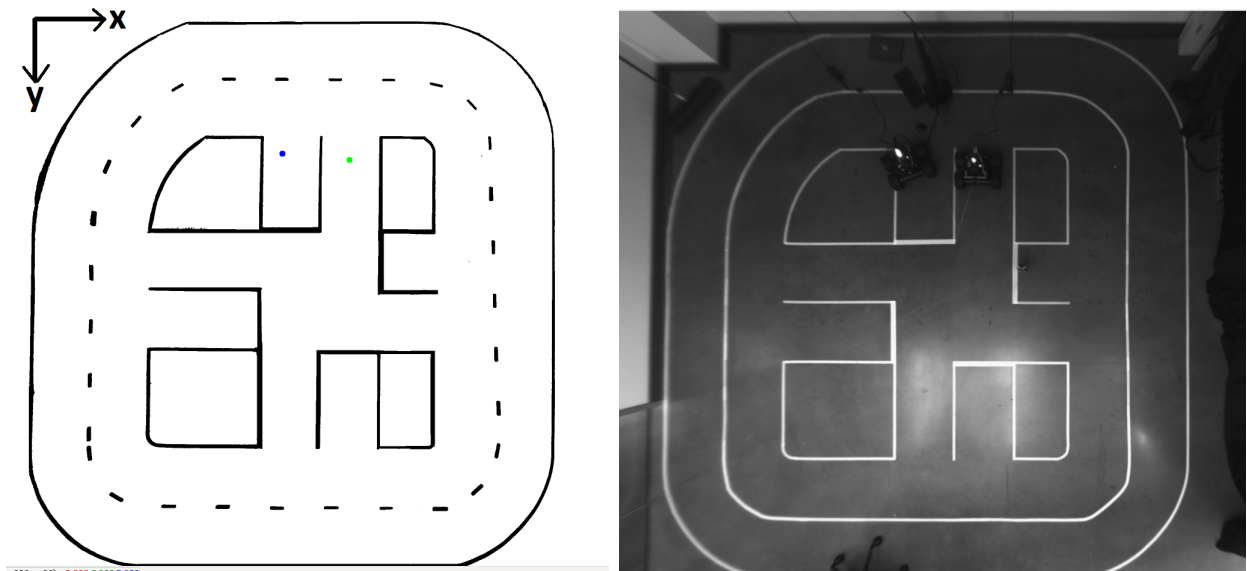


Figure 6.3: Indoor camera-based localization(left) and Ground truth images(right)

6.3.1 Scenario-1: Vehicles traveling along the same direction on adjacent lanes

Consider a scenario as shown in Fig.6.4 in which there are two vehicles A and B on adjacent lanes and vehicle B spots a pedestrian in front of itself. This pedestrian is not in the field of view of vehicle A at the moment or this pedestrian has not been detected by vehicle A. However, the future trajectory of the pedestrian might cross in front of vehicle A. Thus, there is a necessity for vehicle A to be aware of this pedestrian.

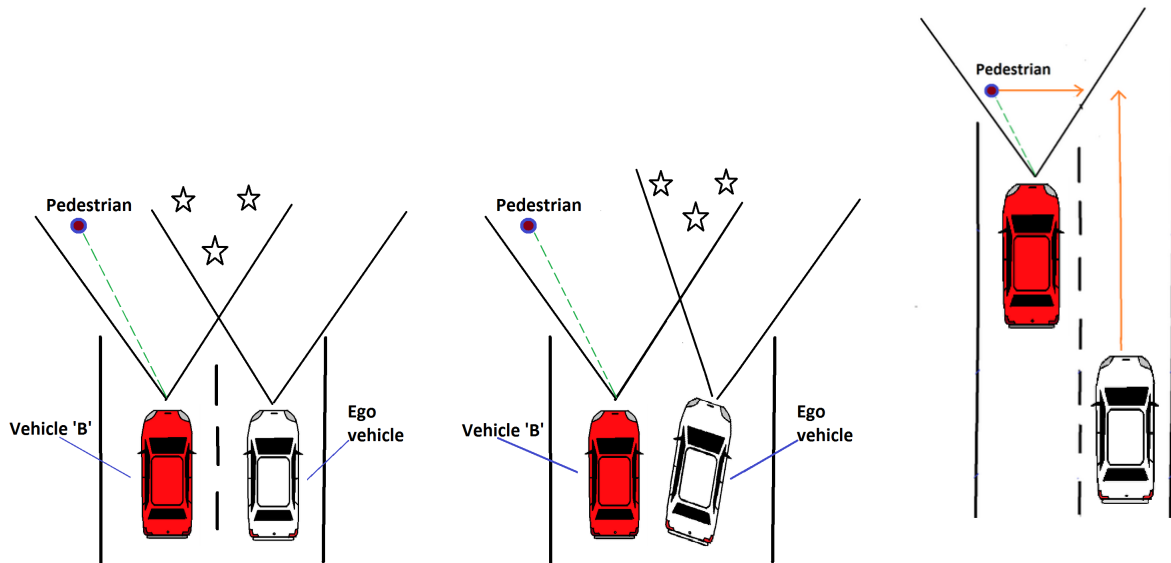


Figure 6.4: Cooperative perception across vehicles on adjacent lanes (left), on a wide road without lanes (center) and pedestrian crossing in front of stationary vehicle 'B' and suddenly emerging onto the future trajectory of the ego-vehicle on the adjacent lane (below)

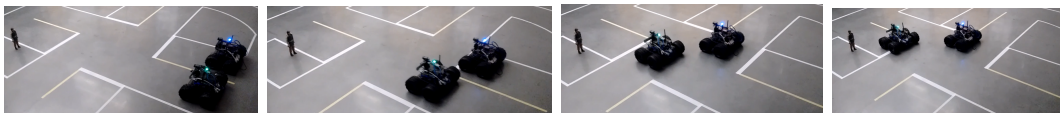


Figure 6.5: Experimental Evaluation of Scenario-1

To demonstrate this scenario, the following experiment was performed: Two robots were made to move adjacently in the same direction. The pedestrian visible to robot 'B' was not in the field of view of the ego-robot. When the pedestrian was spotted in front by robot 'B', it communicated this message to the ego-robot which generated a hazard message and directly sent it to its low-level controller. Fig. 6.5 shows the response with a fully-functional cooperative perception system. It can be observed from the charts shown in Fig. 6.22 and Fig. 6.23 that, although the linear velocity command was non-zero from the high-level controller, the wheel speeds reduce and eventually reach zero, making the robot stop when the pedestrian was spotted in front through cooperative perception.

6.3.2 Scenario-2: Vehicle merging onto traffic

Consider a merging scenario as shown in Fig. 6.6 where a vehicle A approaches a road and is about to merge onto it. A pedestrian is seen crossing the road by a vehicle B traveling straight on the road, but this pedestrian will not be spotted by vehicle A until it completely merges onto the road. In this case, vehicle B traveling straight provides this information to the vehicle A that is merging. If vehicle A was made aware of a pedestrian on its future trajectory early, then it would have time to react by slowing down and stopping.

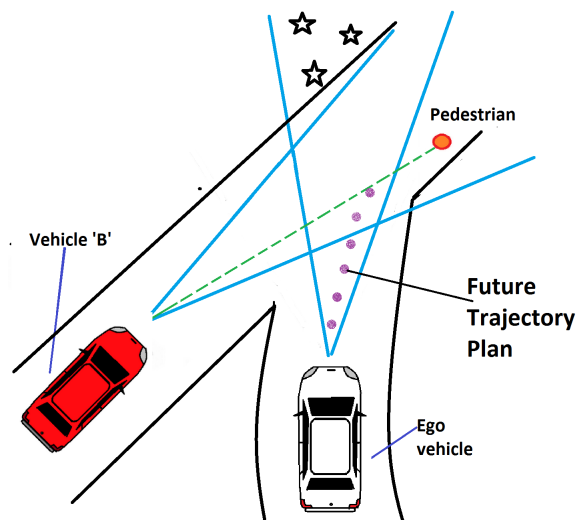


Figure 6.6: Cooperative perception across vehicles that approach each other in a traffic merging scenario

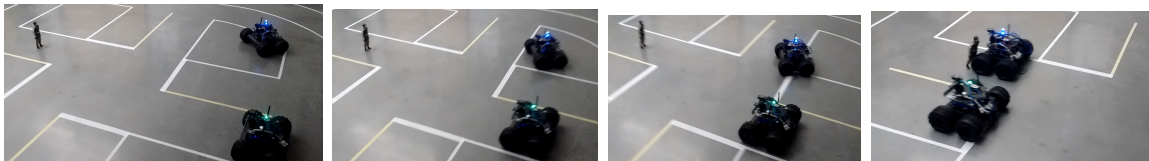


Figure 6.7: Scenario-2 in the absence of Cooperative Perception

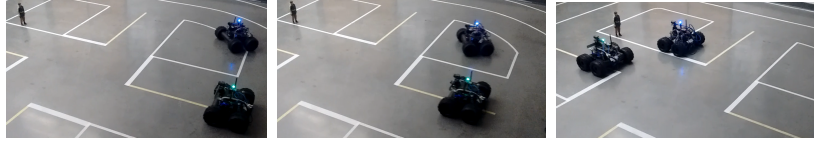


Figure 6.8: Experimental Evaluation of Scenario-2

To demonstrate this scenario, the following experiment was performed: The ego-robot was placed at an angle to robot 'B' and was made to merge onto a lane adjacent to robot 'B'. The pedestrian was placed in the future trajectory of the ego-robot. It was spotted by robot 'B', localized and communicated to the ego-robot, which generated a hazard message and an emergency response and directly sent it to the low-level controller of the ego-robot. The remedial action was to slow down and stop. Fig. 6.8 shows the response with a fully-functional cooperative perception system and Fig. 6.7 shows the same without the cooperative perception system to demonstrate the threat otherwise. The data obtained is presented in Fig. 6.24 and Fig. 6.25. It can be observed from the charts that, although the linear velocity command was non-zero from the high-level controller, the wheel speeds reduce and eventually reach zero, making the robot stop when the pedestrian was spotted in front through cooperative perception. The RMS (Root Mean Squared) Error in the ego-vehicle's pedestrian location estimate in this case, was computed to be 13.92 cm in the lateral direction and 13.3 cm in the longitudinal direction.

6.3.3 Scenario-3: Sensor/Detection Failure: Vehicles traveling along the same direction on adjacent lanes

Consider a scenario as shown in Fig.6.9 in which there are two vehicles A and B on adjacent lanes and vehicle B spots a pedestrian in front of vehicle A. This pedestrian is in the field of view of vehicle A but has not been detected by vehicle A owing to a sensor/detection failure. The lack of availability of favorable visual features could result in such a scenario. Thus, there is a crucial compulsion for vehicle A to be aware of this pedestrian.

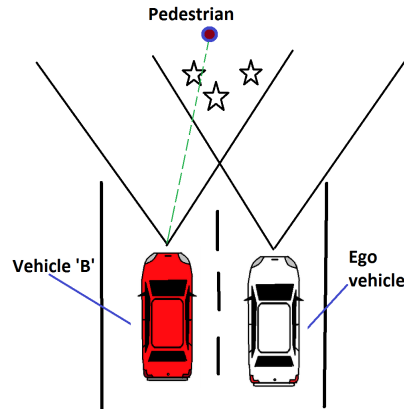


Figure 6.9: Cooperative perception across vehicles on adjacent lanes (Sensor/Detection Failure)

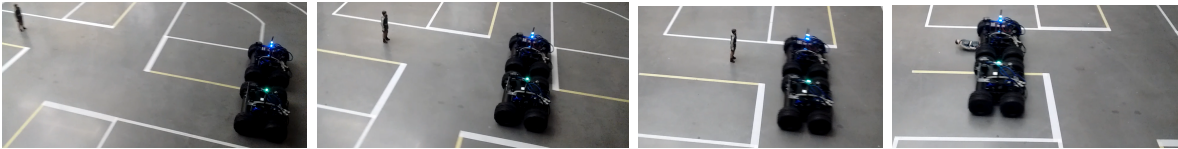


Figure 6.10: Scenario-3 in the absence of Cooperative Perception : Ego-vehicle's sensor fails



Figure 6.11: Experimental Evaluation of Scenario-3: Emergency braking response

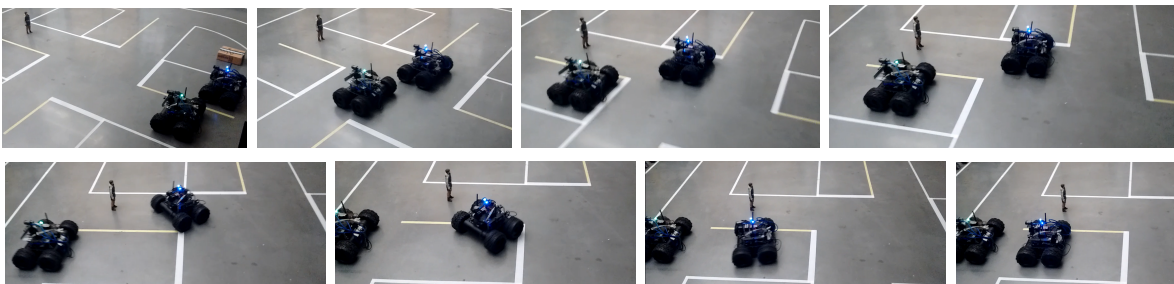


Figure 6.12: Experimental Evaluation of Scenario-3: Emergency lane change response

To demonstrate this scenario, the following experiment was performed: Two robots were made to move adjacently in the same direction. The pedestrian visible to robot 'B' was also in the field of view of the ego-robot and directly in front of it on its future trajectory. However, the ego-robot was not empowered with the capability to detect the pedestrian on its own, to emulate detection failure. When the pedestrian was spotted in front, a hazard message and an emergency response were generated and directly sent to the low-level controller. In another trial, an emergency lane-change maneuver was generated in which the ego-robot swerved to the adjacent lane to avoid the pedestrian. Fig. 6.11 and Fig. 6.12 show the response with a fully-functional cooperative perception system and Fig. 6.10 illustrates the case without the cooperative perception system to demonstrate the threat otherwise. It can be observed from the charts shown in Fig. 6.26 and Fig. 6.27 that, although the linear velocity command was non-zero from the high-level controller, the wheel speeds reduce and eventually reach zero, making the robot stop when the pedestrian was spotted in front through cooperative perception. In the second case shown in Fig. 6.28 and Fig. 6.29, although the angular velocity command from the high-level controller was zero, the robot turned with an angular velocity to move onto the adjacent lane, visible from the wheel speed data. The RMS Error in the ego-vehicle's pedestrian location estimate in the case of the emergency stop response, was computed to be 7.49 cm in the lateral direction and 16.63 cm in the longitudinal direction. Similarly, the RMS Error in the ego-vehicle's pedestrian location estimate in the case of the lane change response, was computed to be 9.19 cm in the lateral direction and 19.24 cm in the longitudinal direction.

6.3.4 Scenario-4: Vehicles traveling on the same lane with one vehicle behind the other vehicle

Consider a scenario as shown in Fig. 6.13 where two vehicles A and B are moving one behind the other, in the same lane. The vehicle B in front spots a pedestrian ahead on the adjacent lane. However, this pedestrian is not in the field of view of the vehicle A in rear, as vehicle B in front has obstructed this view. Vehicle A is planning to make a passing/overtaking maneuver and is unaware of the pedestrian. The knowledge about the pedestrian becomes crucial for vehicle A, as the ego-vehicle would have very less time to respond to the pedestrian after the lane change. A possible remedy would be to avoid the lane change action at this time.

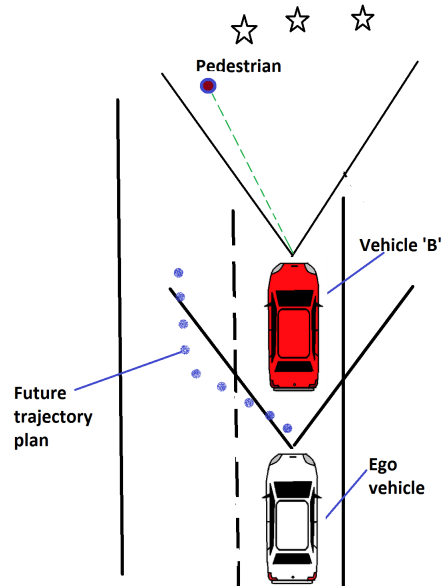


Figure 6.13: Cooperative perception across vehicles on the same lane



Figure 6.14: Scenario-4 in the absence of Cooperative Perception



Figure 6.15: Experimental Evaluation of Scenario-4

To demonstrate this scenario, the following experiment was performed: Two robots were made to move in the same direction with the ego-robot in the rear and robot 'B' in front. The pedestrian was spotted in front by robot 'B', localized and communicated to the ego-robot, a hazard message and an emergency response were generated and directly sent to its low-level controller. The remedial action was to continue moving on course and avoid the

lane change action. Fig. 6.15 shows the robot response with a fully-functional cooperative perception system and Fig. 6.14 shows the response without the cooperative perception system to demonstrate the threat otherwise. The data obtained is presented in Fig. 6.30 and Fig. 6.31. It can be observed from the charts that, although the angular velocity command was non-zero from the high-level controller, the wheel speeds do not change and the ego-robot continues to keep its course when the pedestrian was spotted and a hazard warning was generated.

6.3.5 Scenario-5: Vehicles on adjacent lanes with a non-cooperative vehicle in front

Consider a scenario as shown in Fig. 6.16 where there are two vehicles A and B traveling on adjacent lanes, and the field of view of one vehicle is blocked by another vehicle C in front that is not cooperative and hence is not capable of communicating information to vehicle A behind. If vehicle C in front were to spot a pedestrian in front and reduce speed or brake suddenly, vehicle A on the rear might not have sufficient time to react. The knowledge about this pedestrian could be made available to vehicle A in rear, by vehicle B on the adjacent lane. The remedial action would be to brake in advance to avoid bumping onto the vehicle C in front that braked suddenly.

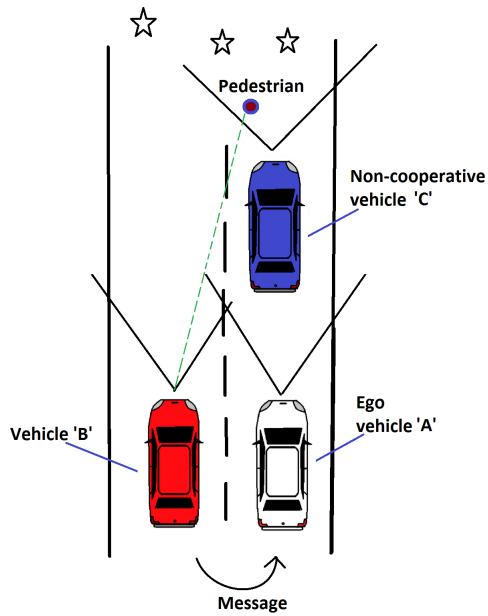


Figure 6.16: Cooperative perception across vehicles with a non-cooperative vehicle in front

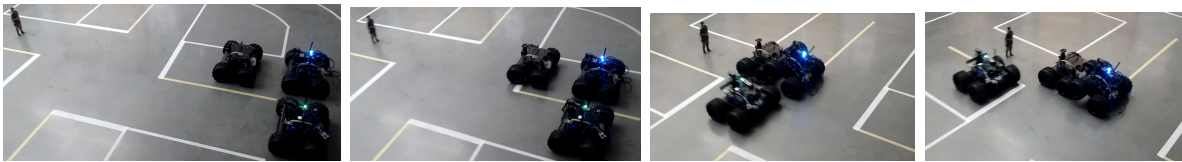


Figure 6.17: Scenario-5 in the absence of Cooperative Perception



Figure 6.18: Experimental Evaluation of Scenario-5

To demonstrate this scenario, the following experiment was performed: The ego-robot and robot-B were placed on adjacent lanes and made to travel in the same direction. Another robot without communication was made to travel in front of the ego-robot. A pedestrian

was placed in the lane of the ego-robot and the robot in front was preprogrammed to stop at a point before hitting the pedestrian. The pedestrian was spotted by robot 'B', localized and communicated to the ego-robot, which generated a hazard message and an emergency response and directly sent it to its low-level controller. The remedial action was to slow down and stop, sensing that the vehicle in front would stop for the pedestrian. Fig. 6.18 shows the response with a fully-functional cooperative perception system and Fig. 6.17 shows the same without the cooperative perception system to demonstrate the threat otherwise. The data obtained is presented in Fig. 6.32 and Fig. 6.33. Although the linear velocity command reached zero only later, a collision message was generated and the wheel speeds reduce and eventually reach zero, making the robot stop.

6.3.6 Scenario-6: Intersection of Roads

Vehicles at an intersection at right angles to each other do not have a visual field of view overlap, where features can be matched to obtain a pose fix, as the vehicles might be observing different sides of the same object. However, accurate positioning on real-world maps can be used by a higher-level (map-level) algorithm that initializes a relative orientation of 90 degrees to vehicles at an intersection. This can be particularly useful in a scenario where two vehicles 'A' and 'B' approach each other at an intersection. Vehicle 'B' spots a pedestrian who is on the future trajectory of vehicle 'A' that plans to turn left. This pedestrian will not be spotted by vehicle 'A' directly until it turns but that would be too late. Vehicle 'B' spots the pedestrian and warns vehicle 'A' which has started turning left but slows down to avoid the hazard.

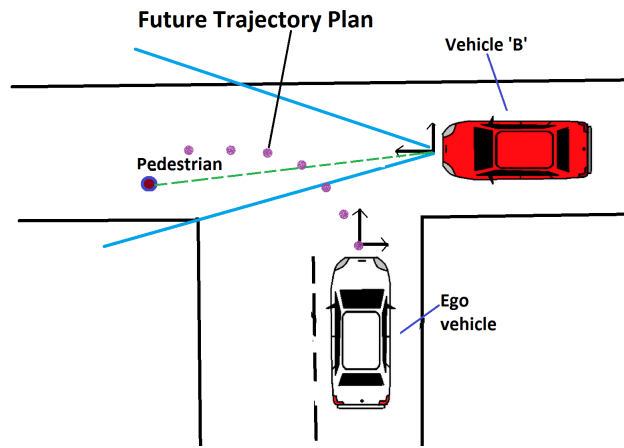


Figure 6.19: Cooperative perception across vehicles at an intersection

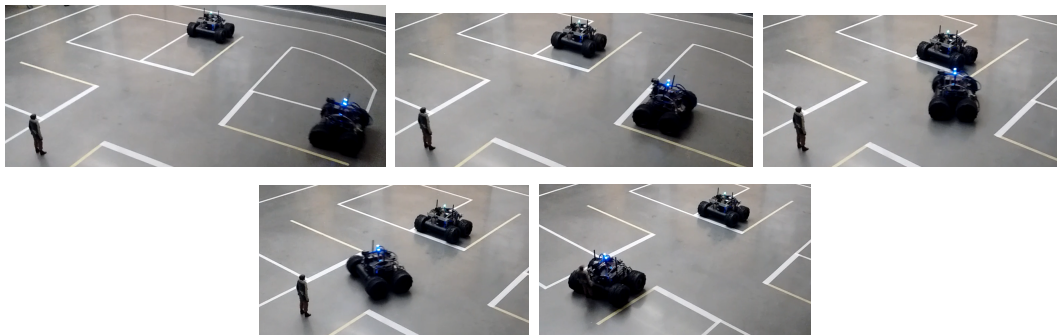


Figure 6.20: Scenario-6 in the absence of Cooperative Perception

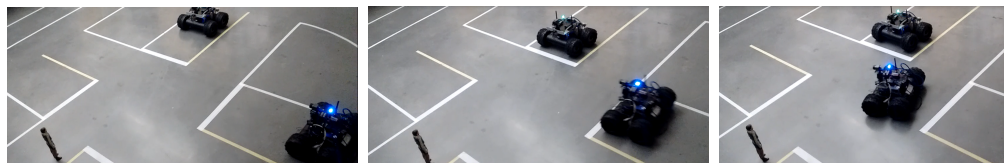


Figure 6.21: Experimental Evaluation of Scenario-6

To demonstrate this scenario, the following experiment was performed: The ego-robot and robot-B were made to approach an intersection. The ego-robot was instructed to turn left at the intersection. A pedestrian was placed at the beginning of lane onto which the ego-robot turned. The pedestrian was spotted by robot 'B', localized and communicated to

the ego-robot that has already started turning left, which generated a hazard message and an emergency response and directly sent it to its low-level controller. The remedial action was to slow down and stop, however in the absence of any oncoming traffic. Fig. 6.21 shows the robot response with a fully-functional cooperative perception system and Fig. 6.20 shows the same without the cooperative perception system to demonstrate the threat otherwise. The data obtained is presented in Fig. 6.34 and Fig. 6.35. It can be observed from the charts that, although the linear velocity command was non-zero from the high-level controller, the wheel speeds reduce and eventually reach zero, making the robot stop.

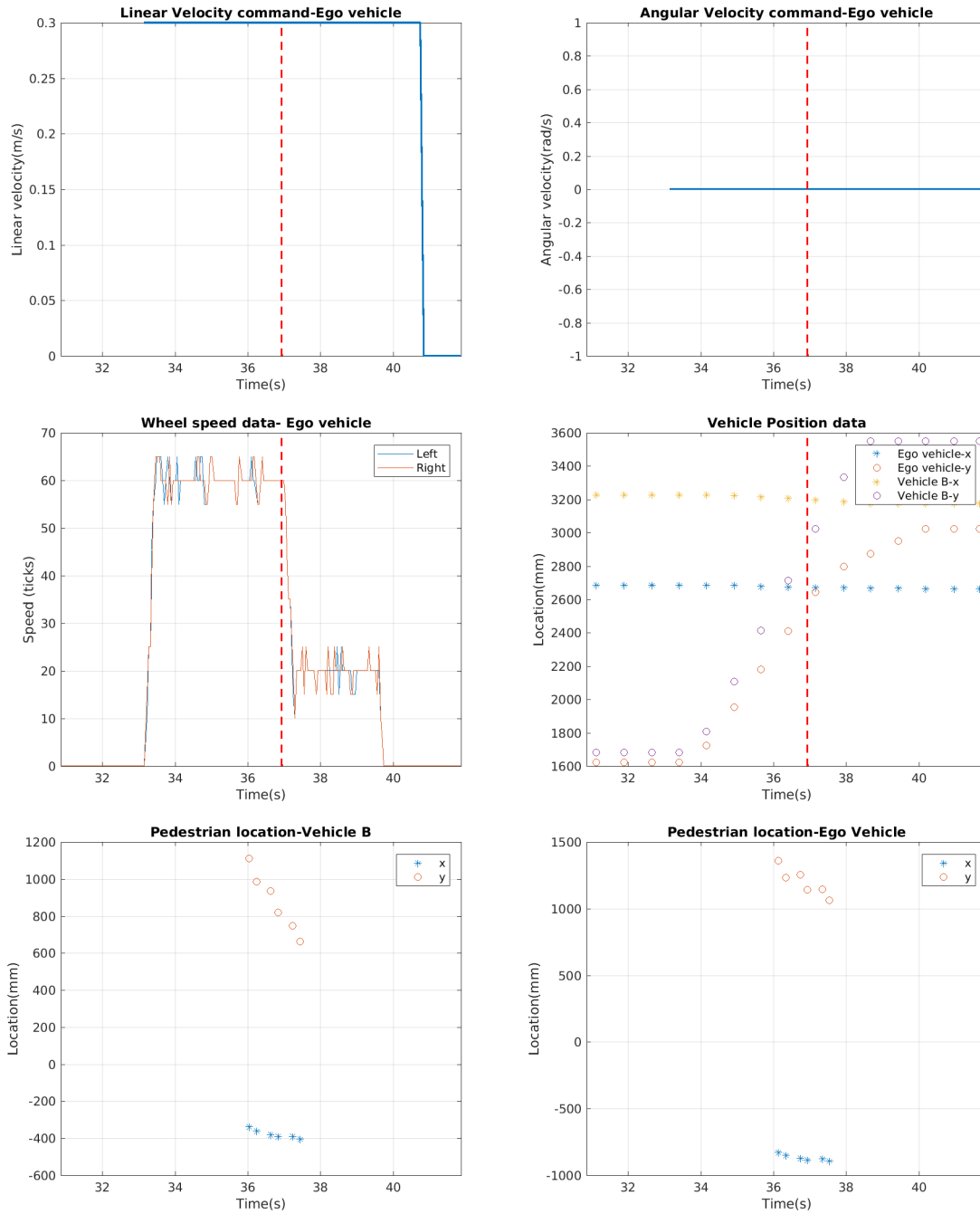


Figure 6.22: Experimental Results of Scenario-1

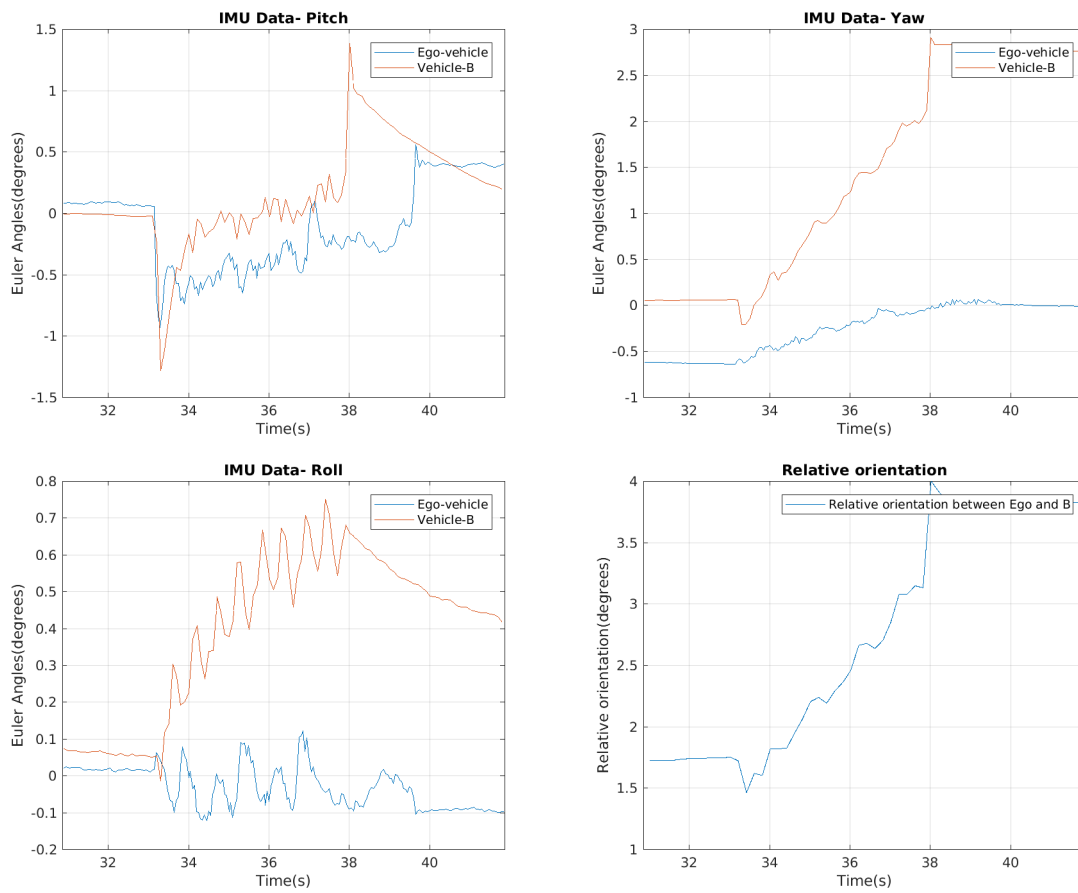


Figure 6.23: Experimental Results of Scenario-1

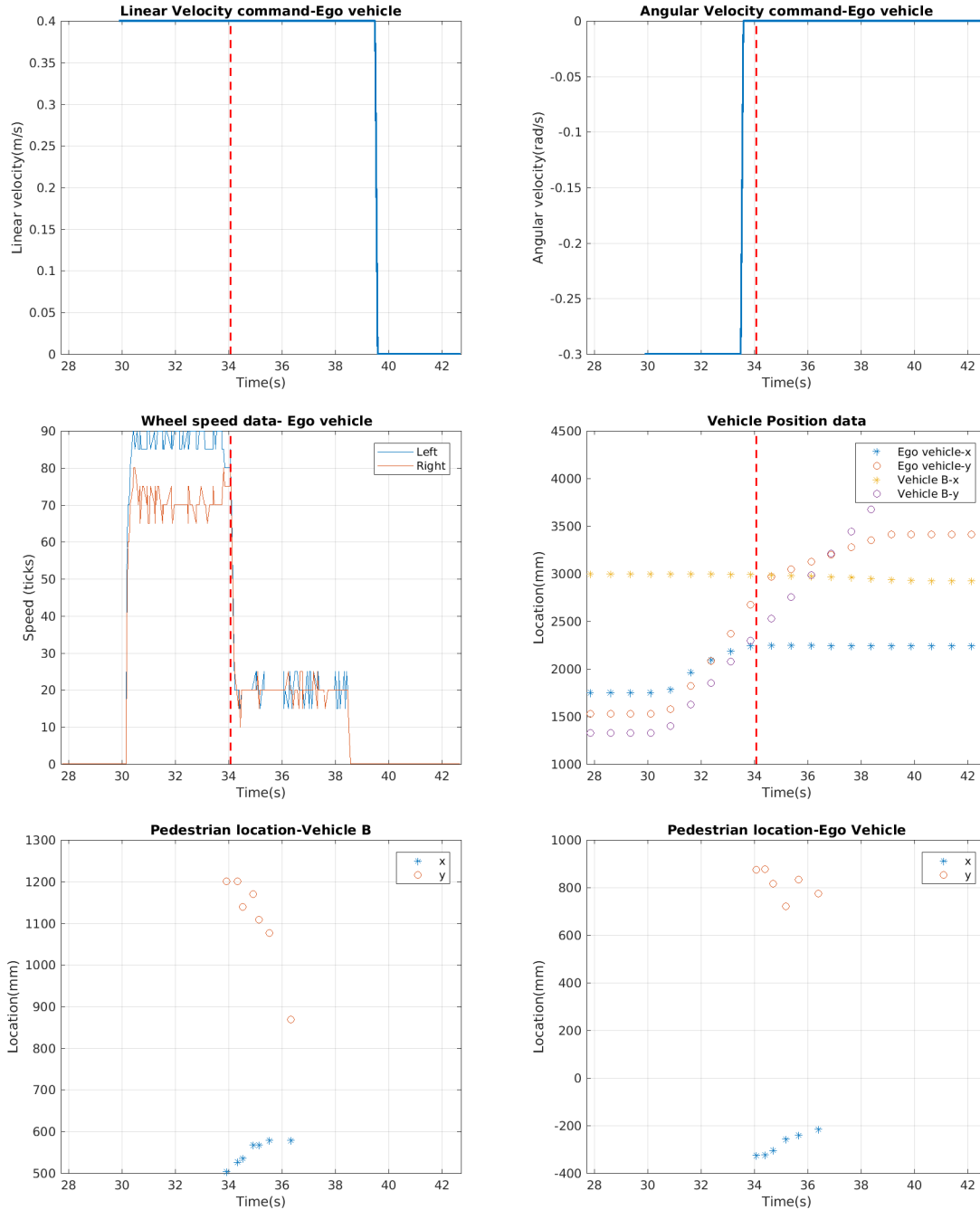


Figure 6.24: Experimental Results of Scenario-2

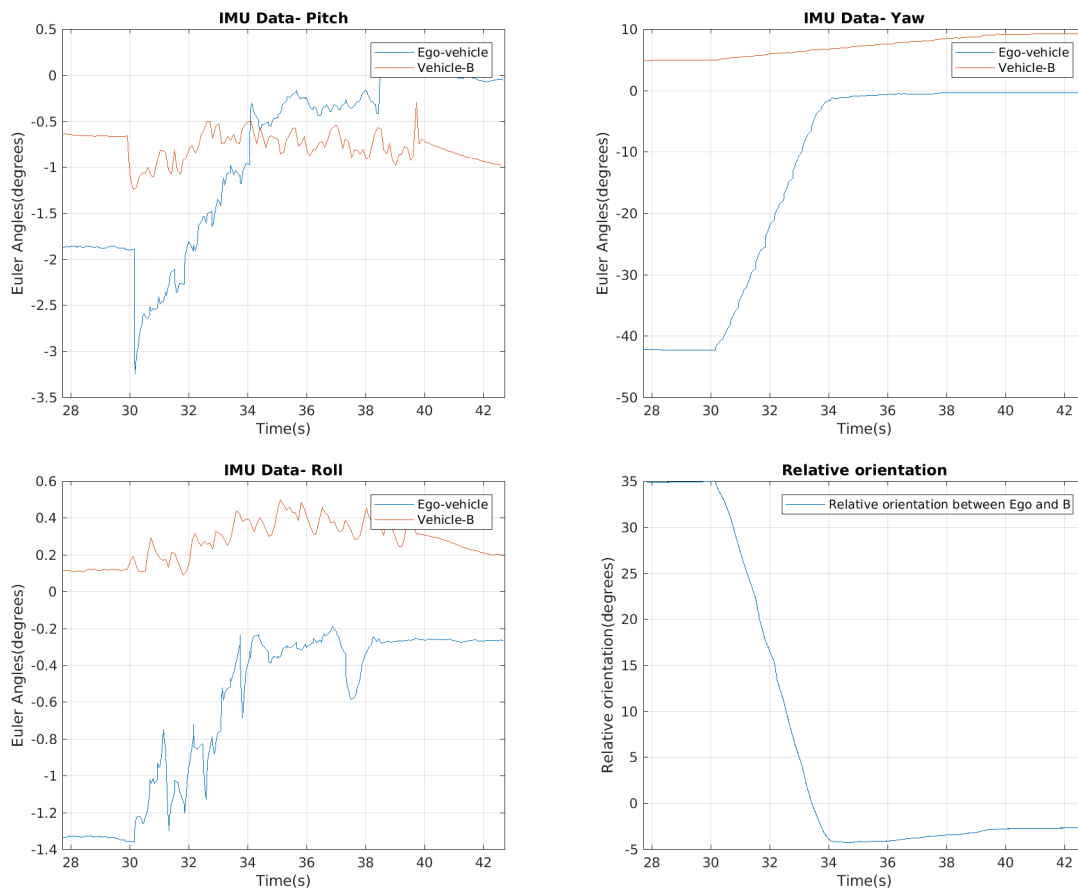


Figure 6.25: Experimental Results of Scenario-2

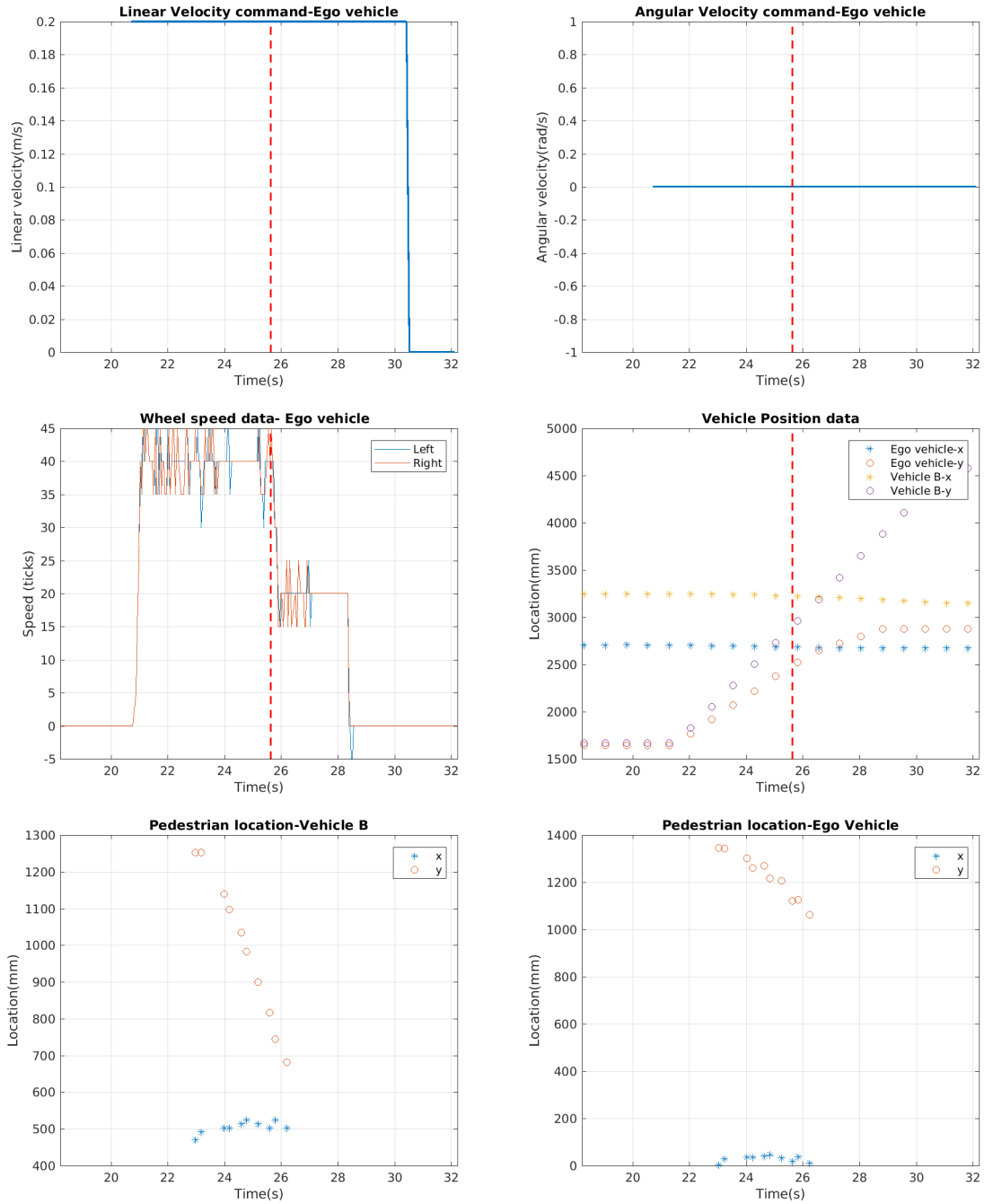


Figure 6.26: Experimental Results of Scenario-3 (emergency stop case)

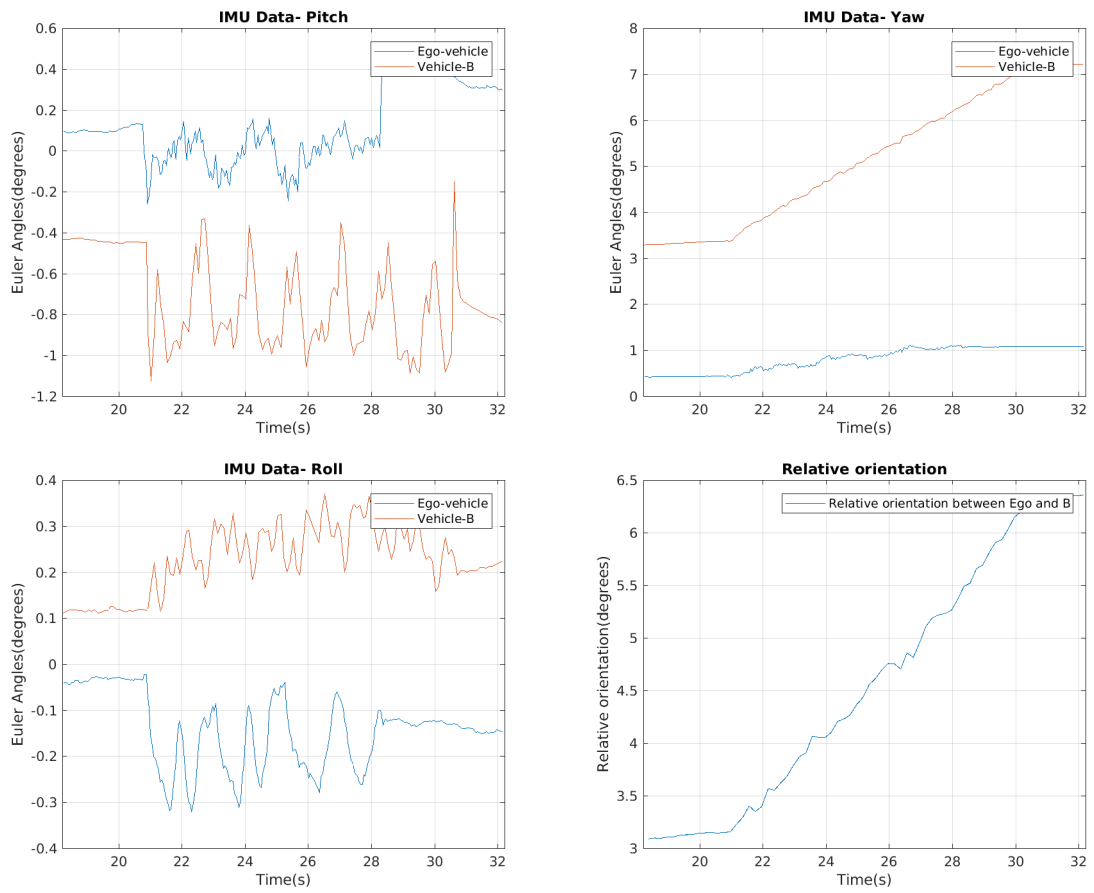


Figure 6.27: Experimental Results of Scenario-3 (emergency stop case)

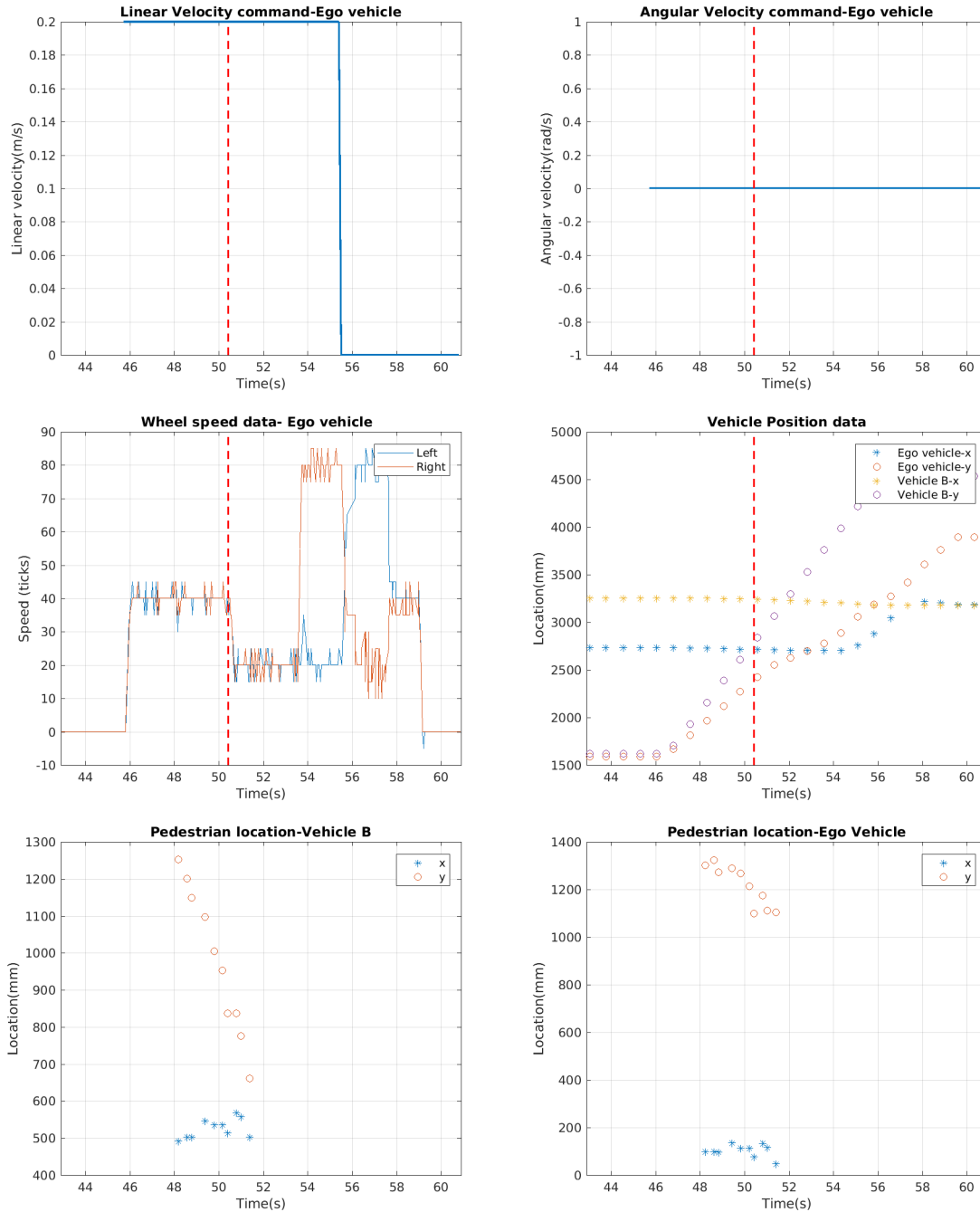


Figure 6.28: Experimental Results of Scenario-3 (lane change case)

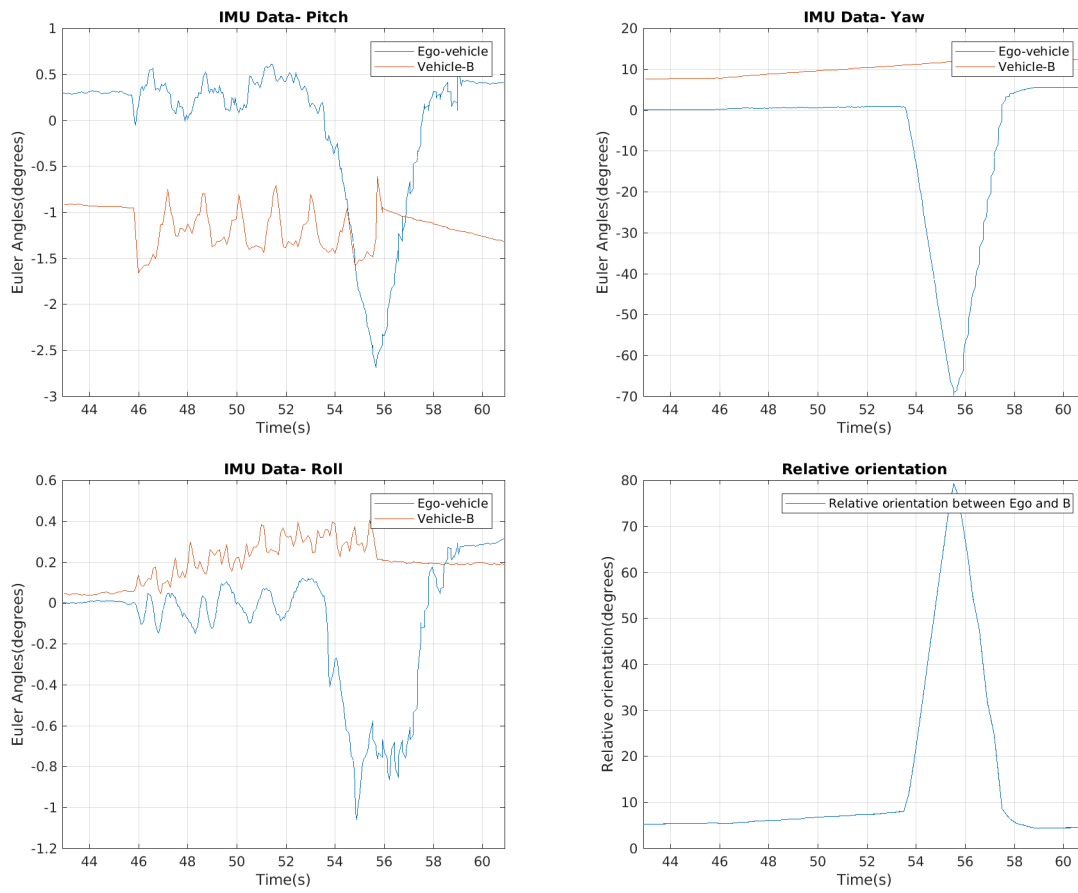


Figure 6.29: Experimental Results of Scenario-3 (lane change case)

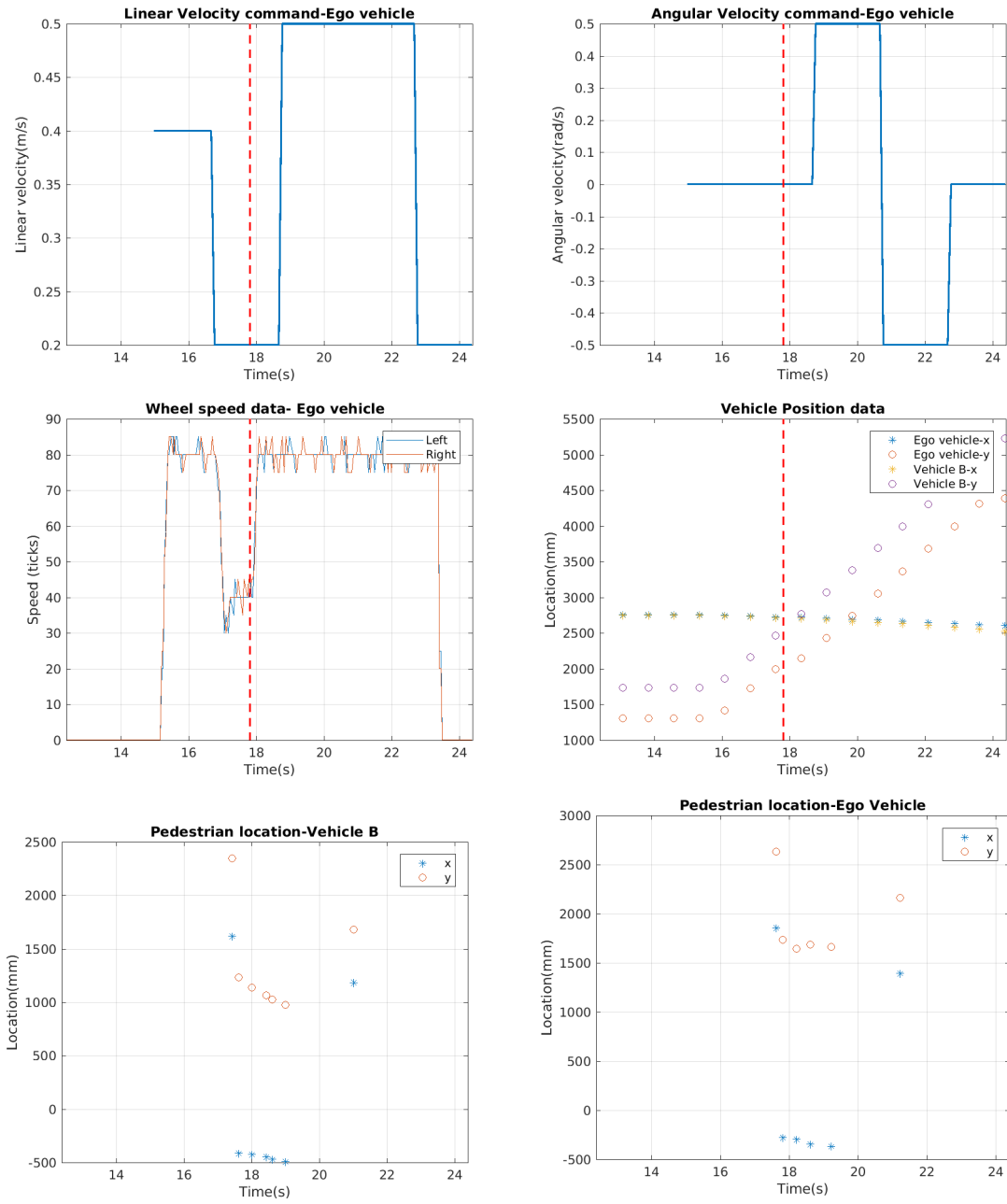


Figure 6.30: Experimental Results of Scenario-4

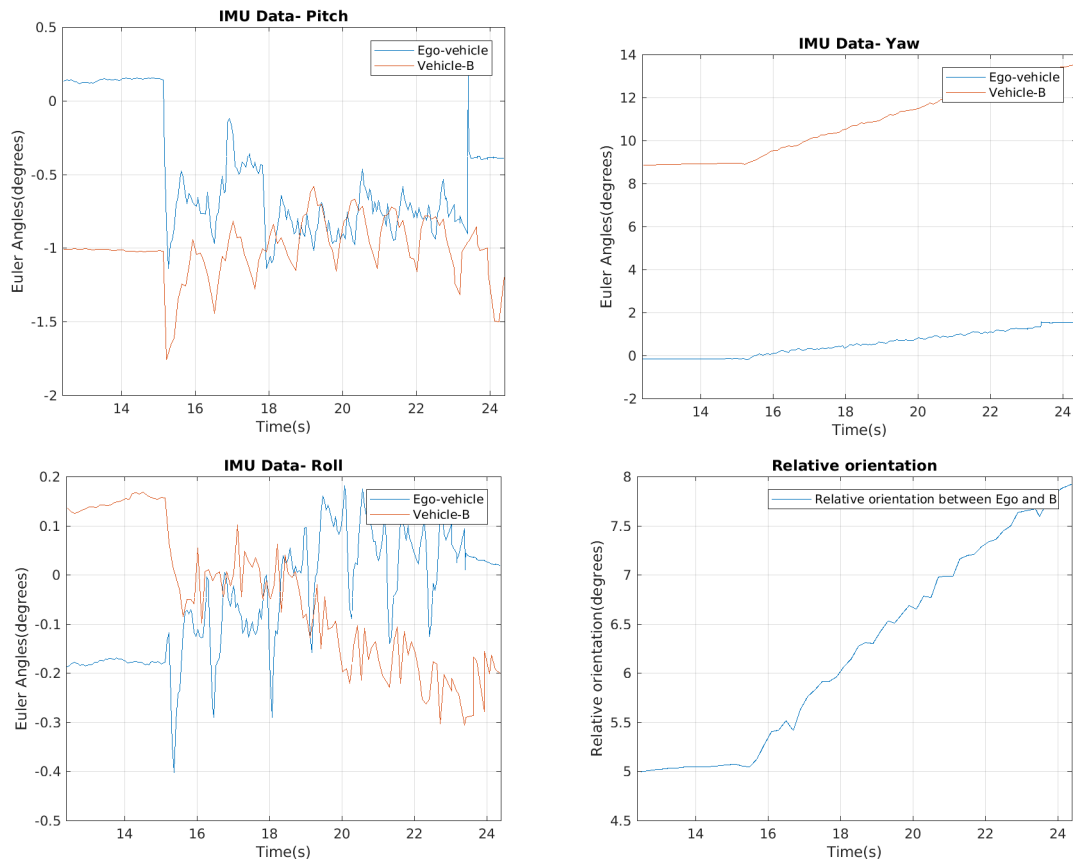


Figure 6.31: Experimental Results of Scenario-4

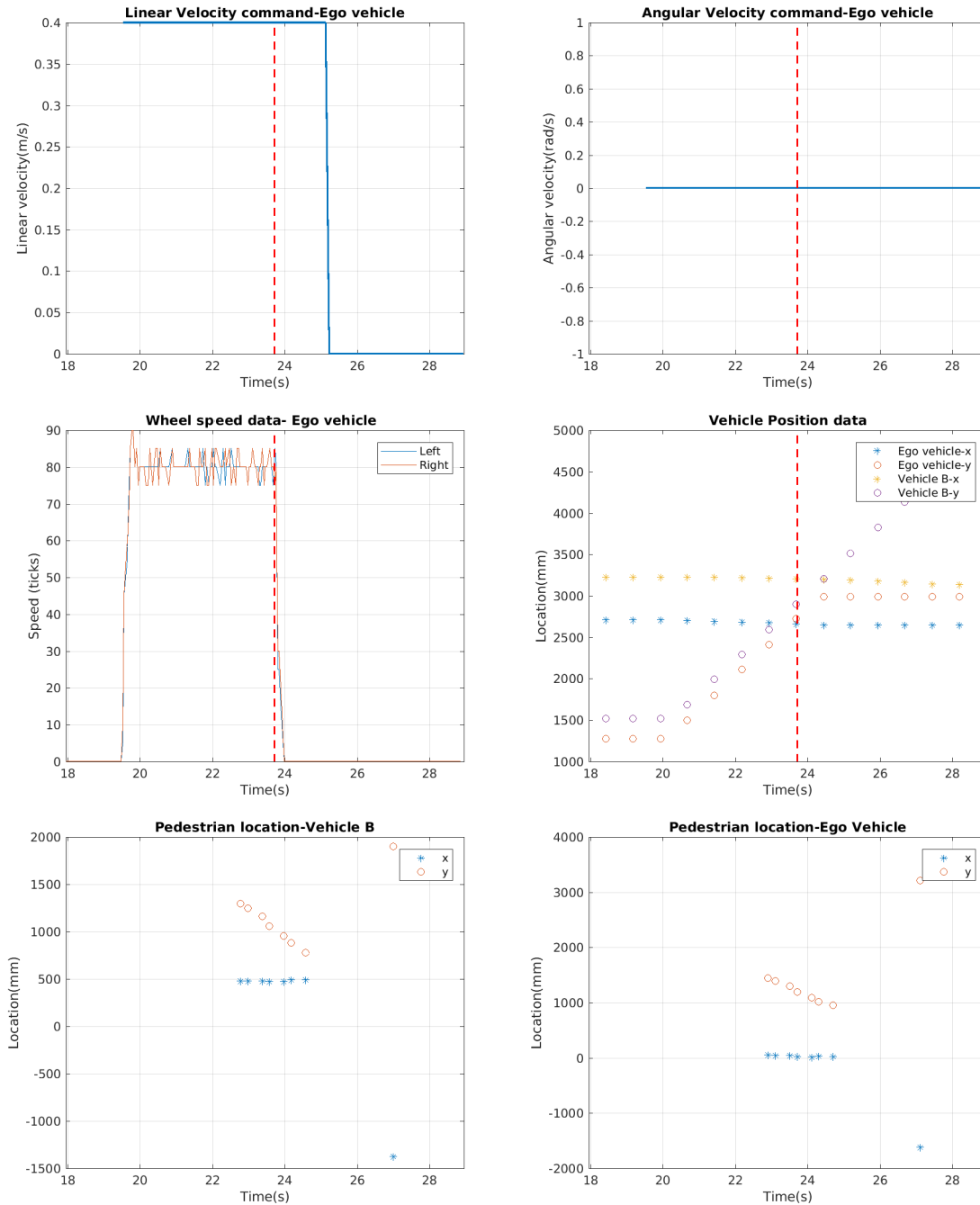


Figure 6.32: Experimental Results of Scenario-5

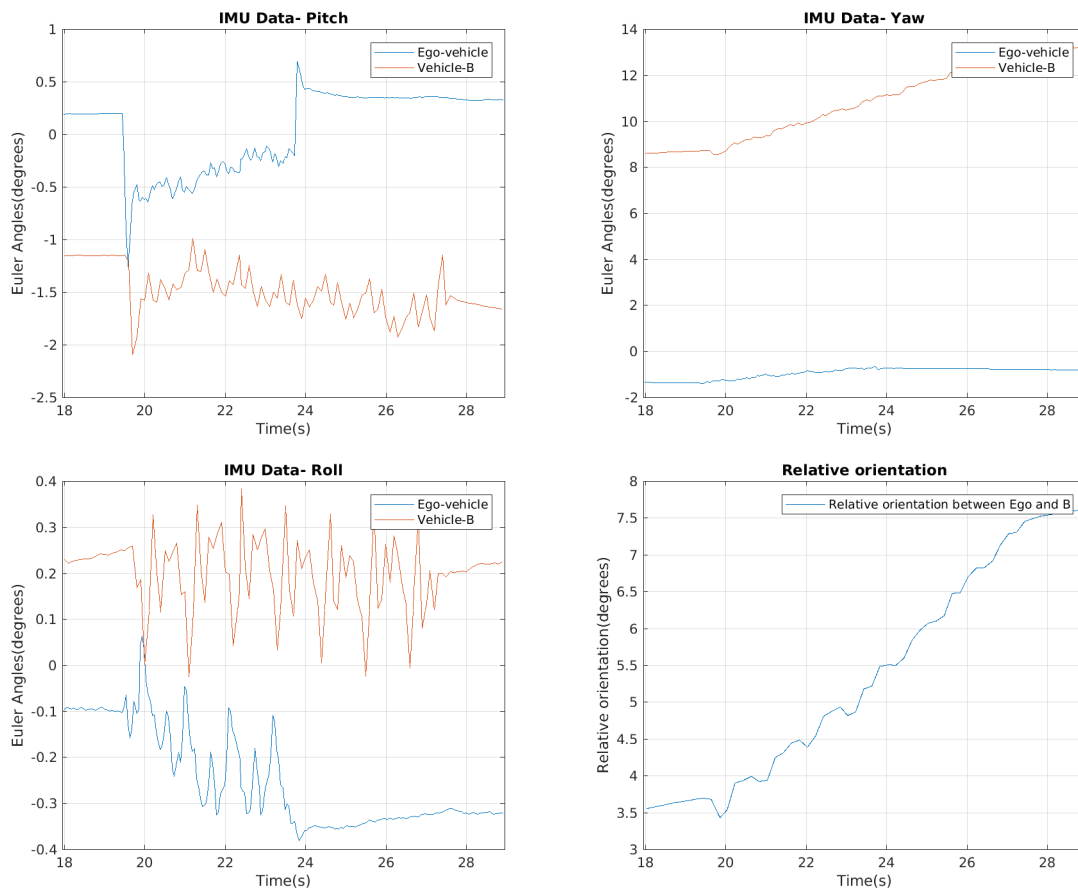


Figure 6.33: Experimental Results of Scenario-5

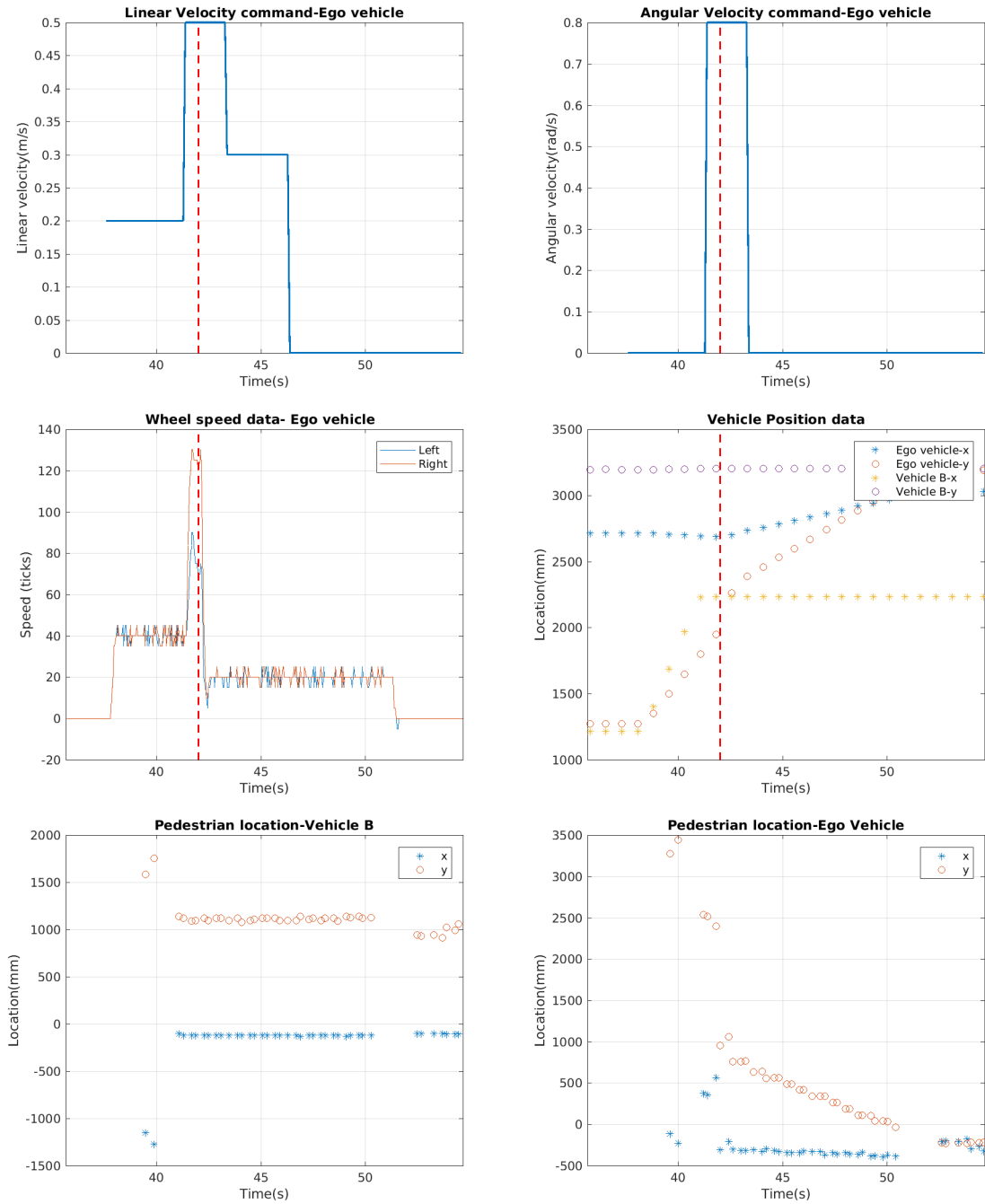


Figure 6.34: Experimental Results of Scenario-6

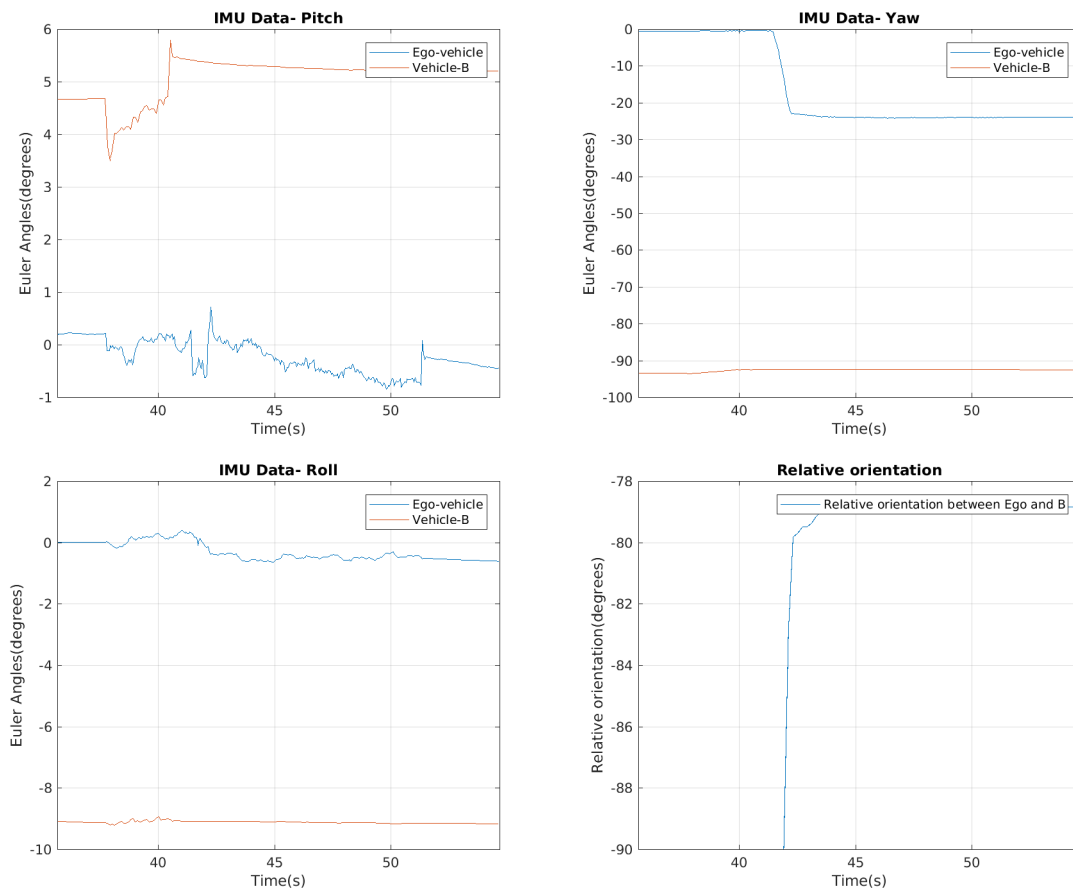


Figure 6.35: Experimental Results of Scenario-6

Chapter 7

Conclusion

The previous chapters described recent literature on mobile robotics, autonomous vehicles and specifically on cooperative perception, following which a specific method for cooperative perception was proposed in this work. The corresponding parts of relative localization among vehicles and approximate object localization using a monocular camera were explained in detail. The individual modules developed to form the testbed robot architecture were illustrated and the specific cooperative perception software pipeline was described. Potential traffic scenarios that could benefit from this system were demonstrated and results were presented. This chapter concludes the work, analyzing the results as well as outlining future directions for the research.

7.1 Conclusions

Autonomous Vehicles are limited by their on-board sensors and their field of view. Obstacles in blind spots could pose potential threats. A part of the environment could be occluded by other vehicles. Additionally, on-board sensors are prone to failure. In such cases, turning each vehicle on the road into a sensor capable of relaying information could extend the line of sight and thus result in better motion planning and improve traffic safety. In this work, one such cooperative perception algorithm was proposed that could be applicable to certain traffic scenarios. The problem of cooperative perception was broken down into the relative pose estimation problem followed by map merging [24]. In the absence of range-based sensors, an approximate object location estimation technique using a monocular camera was

proposed to be used. The algorithm was programmed onto two mobile robots and a ground station, and experiments were performed to demonstrate the working of the system and the results were tabulated and shown in the previous chapter. The conclusions and inferences are summarized as follows, for each individual subsystem analyzing their advantages and limitations:

7.1.1 Relative pose estimation

- The problem of cooperative localization has been solved using visual and inertial sensors. Contemporary literature provides techniques to obtain relative poses from range-based sensors such as LIDAR through scan matching techniques. Vision-based sensors are proposed as a better choice owing to their cheaper cost, and a standard technique to obtain relative camera orientation from images has been used
- Visual feature points are matched to obtain the relative camera orientation. Incorrect feature matches was found to result in incorrect estimates. However, by observing the overall resulting rotation matrix and euler angles, knowing that certain vehicle relative pose combinations are not permitted by vehicle dynamics, such outliers can be eliminated
- Image feature points and descriptors such as SURF are invariant to scale and rotation but only partially invariant to illumination [44]. Although two vehicles observing the same scene obtain images with the same illumination, camera exposure also plays a role in robust feature extraction and matching
- Repeated features such as building windows and roadside trees can hinder a good estimate of relative pose since features from different objects might be mismatched
- In most cases, a form of visual field-of-view overlap is possible among vehicles that are present in adjacent lanes or the same lane, traveling in the same direction. Vehicles traveling towards each other do not share a common field of view and are outside the scope of this work
- IMU serves as the dead-reckoning counterpart to estimating the orientation once vision provides a starting reference frame. In the absence of the magnetometer, the IMU was found to drift by a minor value but this is admissible over a short distance

- Communication latency proves to be a challenge. To achieve cooperative localization, images or image features need to be transmitted which required advanced communication capabilities. [7] and [24] illustrate the feasibility of different communication mediums, even in cases where compressed images need to be transferred. Scaling such a system to tens of vehicles in an area increases the complexity of the problem

7.1.2 Object detection and localization

- Locating objects in a traffic scenario is an essential activity for an autonomous vehicle. The vehicle alters its trajectory based on the position of these obstacles. For locating these obstacles accurately, range-based sensors or stereo vision may be used. In the absence of such sensors, a simple technique is proposed to localize pedestrians using a monocular camera
- The method is computationally light since only a single point is transformed through the inverse perspective transformation
- The estimate obtained from this technique has an error in the longitudinal direction due to the pixel distance between the foot of the pedestrian resting on the ground and the bounding box boundary. This error is larger at greater distances owing to the pixel to real-world units ratio being larger at a greater distance on the perspective image. The lateral error is relatively less, as the detection bounding box is usually centered around the pedestrian
- A higher pixel resolution and an accurate pedestrian detection technique were found to be crucial for this particular technique to be robust. An error at large distance is assumed admissible from a system perspective as safety is concerned with the immediate vicinity of the vehicle
- Through the inverse perspective mapping, a point on the perspective image is transformed onto a point on the ground plane. An assumption that is made that the ground plane is flat. Thus, the estimate would be error-prone in the absence of a flat ground plane
- The viewpoint of the front-facing camera with respect to the ground plane can be measured, pre-calibrated and used as a standard value. Owing to the pitching motion of the vehicle as well as uneven loading of the vehicle, this angle is subject to change.

However, as long as the change in angle is small, and the initial angle of tilt itself with respect to the ground plane is small, the estimates are not affected much, as pointed out by [37]

- Communication of the observed entity is not challenging as only the point location of the entity is transmitted.

7.2 Future Research

The proposed system demonstrates two individual techniques, which have been combined to form the overall cooperative perception system. This section lists further work that could be carried out to improve and refine each of the two subsystems and the overall system as a whole, as follows:

7.2.1 Relative pose estimation

- Accurate position was assumed to be known in this work. The error under conditions where approximate position information of the vehicles is available could be studied
- This technique could be experimented with other feature points such as ORB, FAST, BRIEF and MSER to test if better feature matches can be achieved. Various algorithms such as the 5-Point algorithm and the normalized 8-point algorithm could also be used to obtain the Fundamental matrix
- The proposed system could be tested with a system that permits computation and communication of high-resolution images to obtain better results
- State estimation of relative orientation could be improved by using a Kalman Filter or a Particle Filter

7.2.2 Object localization and map merging

- The Inverse Perspective Mapping transformation assumes the presence of a flat road plane. The system could be improvised to detect a non-flat road grade and adjust the transformation accordingly

- The effect of the pitching motion of the vehicle and vehicle loading on the accuracy of a pre-calibrated inverse perspective transformation could be studied
- Object localization accuracy could be improved by using range-based sensors or stereo-vision cameras

7.2.3 Cooperative Perception System

- The algorithm could be tested on actual vehicles fitted with sensors
- The effect of scaling the system across more than two vehicles could be studied
- Further traffic scenarios such as a sharp turn blocked by vision and downhill roads are additional scenarios in which the proposed system could be tested
- The proposed system has been tested emulating Vehicle-to-Vehicle communication. This system could be extended to Vehicle-to-Infrastructure scenarios as well
- The proposed algorithm only applies to situations where the vehicles share a common field of view. Cooperative perception could be extended to a more generalized framework that applies to all vehicles within a vicinity irrespective of a field-of-view overlap

Bibliography

- [1] U. Ozguner, T. Acarman, and K. Redmill, *Autonomous ground vehicles*. Artech House, 2011.
- [2] S. D. Pendleton, H. Andersen, X. Du, X. Shen, M. Meghjani, Y. H. Eng, D. Rus, and M. H. Ang, “Perception, planning, control, and coordination for autonomous vehicles,” *Machines*, vol. 5, no. 1, p. 6, 2017.
- [3] S. A. Bagloee, M. Tavana, M. Asadi, and T. Oliver, “Autonomous vehicles: challenges, opportunities, and future implications for transportation policies,” *Journal of Modern Transportation*, vol. 24, no. 4, pp. 284–303, 2016.
- [4] A. Eskandarian, *Handbook of intelligent vehicles*. Springer, 2012.
- [5] N. H. T. S. Administration *et al.*, “2015 motor vehicle crashes: overview,” *Traffic safety facts research note*, vol. 2016, pp. 1–9, 2016.
- [6] M. Kyriakidis, R. Happee, and J. C. de Winter, “Public opinion on automated driving: Results of an international questionnaire among 5000 respondents,” *Transportation research part F: traffic psychology and behaviour*, vol. 32, pp. 127–140, 2015.
- [7] S.-W. Kim, W. Liu, M. H. Ang, E. Frazzoli, and D. Rus, “The impact of cooperative perception on decision making and planning of autonomous vehicles,” *IEEE Intelligent Transportation Systems Magazine*, vol. 7, no. 3, pp. 39–50, 2015.
- [8] F. Castanedo, “A review of data fusion techniques,” *The Scientific World Journal*, vol. 2013, 2013.
- [9] M. Brückner, F. Bajramovic, and J. Denzler, “Experimental evaluation of relative pose estimation algorithms,” in *VISAPP (2)*, pp. 431–438, 2008.

- [10] V. Rodehorst, M. Heinrichs, and O. Hellwich, "Evaluation of relative pose estimation methods for multi-camera setups," *International Archives of Photogrammetry and Remote Sensing (ISPRS08)*, pp. 135–140, 2008.
- [11] M. Oliveira, V. Santos, and A. D. Sappa, "Multimodal inverse perspective mapping," *Information Fusion*, vol. 24, pp. 108–121, 2015.
- [12] F. de Ponte Müller, "Survey on ranging sensors and cooperative techniques for relative positioning of vehicles," *Sensors*, vol. 17, no. 2, p. 271, 2017.
- [13] S.-W. Kim and W. Liu, "Cooperative autonomous driving: A mirror neuron inspired intention awareness and cooperative perception approach," *IEEE Intelligent Transportation Systems Magazine*, vol. 8, no. 3, pp. 23–32, 2016.
- [14] G. Dudek and M. Jenkin, *Computational principles of mobile robotics*. Cambridge university press, 2010.
- [15] R. Siegwart, I. R. Nourbakhsh, and D. Scaramuzza, *Introduction to autonomous mobile robots*. MIT press, 2011.
- [16] A. Kelly, *Mobile Robotics: Mathematics, Models, and Methods*. Cambridge University Press, 2013.
- [17] H. Cheng, *Autonomous Intelligent Vehicles: Theory, Algorithms, and Implementation*. Springer Science & Business Media, 2011.
- [18] M. Buehler, K. Iagnemma, and S. Singh, *The DARPA urban challenge: autonomous vehicles in city traffic*, vol. 56. springer, 2009.
- [19] C.-Y. Fu, C.-K. Wang, and E. Park, "A survey of computer vision research for automotive systems," 2015.
- [20] K. Kovacic, E. Ivanjko, and H. Gold, "Computer vision systems in road vehicles: a review," in *the Proceedings of the Croatian Computer Vision Workshop*, 2013.
- [21] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.
- [22] R. Szeliski, *Computer vision: algorithms and applications*. Springer Science & Business Media, 2010.

- [23] J. Janai, F. Güney, A. Behl, and A. Geiger, “Computer vision for autonomous vehicles: Problems, datasets and state-of-the-art,” *arXiv preprint arXiv:1704.05519*, 2017.
- [24] S.-W. Kim, B. Qin, Z. J. Chong, X. Shen, W. Liu, M. H. Ang, E. Frazzoli, and D. Rus, “Multivehicle cooperative driving using cooperative perception: design and experimental validation,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 2, pp. 663–680, 2015.
- [25] L. Merino, F. Caballero, J. Ferruz, J. Wiklund, P.-E. Forssén, and A. Ollero, “Multi-uav cooperative perception techniques,” in *Multiple Heterogeneous Unmanned Aerial Vehicles*, pp. 67–110, Springer, 2007.
- [26] M. W. Achtelik, S. Weiss, M. Chli, F. Dellaert, and R. Siegwart, “Collaborative stereo,” in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2242–2248, IEEE, 2011.
- [27] S.-W. Kim, Z. J. Chong, B. Qin, X. Shen, Z. Cheng, W. Liu, and M. H. Ang, “Cooperative perception for autonomous vehicle control on the road: Motivation and experimental results,” in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5059–5066, IEEE, 2013.
- [28] S.-W. Kim, W. Liu, M. H. Ang, S.-W. Seo, and D. Rus, “Cooperative autonomous driving using cooperative perception and mirror neuron inspired intention awareness,” in *2014 International Conference on Connected Vehicles and Expo (ICCVE)*, pp. 369–376, IEEE, 2014.
- [29] S.-W. Kim, T. Bandyopadhyay, B. Qin, Z. J. Chong, W. Liu, X. Shen, S. Pendleton, J. G. M. Fu, M. H. Ang Jr, E. Frazzoli, *et al.*, “Vehicle autonomy using cooperative perception for mobility-on-demand systems,” in *Motion and Operation Planning of Robotic Systems*, pp. 331–360, Springer, 2015.
- [30] W. Liu, S.-W. Kim, Z. J. Chong, X. Shen, and M. H. Ang, “Motion planning using cooperative perception on urban road,” in *2013 6th IEEE Conference on Robotics, Automation and Mechatronics (RAM)*, pp. 130–137, IEEE, 2013.
- [31] M. Bertozz, A. Broggi, and A. Fascioli, “Stereo inverse perspective mapping: theory and applications,” *Image and vision computing*, vol. 16, no. 8, pp. 585–590, 1998.

- [32] H. A. Mallot, H. H. Bühlhoff, J. Little, and S. Bohrer, “Inverse perspective mapping simplifies optical flow computation and obstacle detection,” *Biological cybernetics*, vol. 64, no. 3, pp. 177–185, 1991.
- [33] G. Ma, S.-B. Park, S. Muller-Schneiders, A. Ioffe, and A. Kummert, “Vision-based pedestrian detection-reliable pedestrian candidate detection by combining ipm and a 1d profile,” in *Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE*, pp. 137–142, IEEE, 2007.
- [34] S. Tan, J. Dale, A. Anderson, and A. Johnston, “Inverse perspective mapping and optic flow: A calibration method and a quantitative analysis,” *Image and Vision Computing*, vol. 24, no. 2, pp. 153–165, 2006.
- [35] J. C. McCall and M. M. Trivedi, “Performance evaluation of a vision based lane tracker designed for driver assistance systems,” in *Intelligent Vehicles Symposium, 2005. Proceedings. IEEE*, pp. 153–158, IEEE, 2005.
- [36] Y.-C. Lim, M. Lee, C.-H. Lee, S. Kwon, and J.-h. Lee, “Improvement of stereo vision-based position and velocity estimation and tracking using a stripe-based disparity estimation and inverse perspective map-based extended kalman filter,” *Optics and Lasers in Engineering*, vol. 48, no. 9, pp. 859–868, 2010.
- [37] D. Hoiem, A. A. Efros, and M. Hebert, “Putting objects in perspective,” *International Journal of Computer Vision*, vol. 80, no. 1, pp. 3–15, 2008.
- [38] A. Møgelmoose, M. M. Trivedi, and T. B. Moeslund, “Trajectory analysis and prediction for improved pedestrian safety: Integrated framework and evaluations,” in *Intelligent Vehicles Symposium (IV), 2015 IEEE*, pp. 330–335, IEEE, 2015.
- [39] A. Koubâa, *Robot Operating System (ROS): The Complete Reference*, vol. 1. Springer, 2016.
- [40] A. Martinez and E. Fernández, *Learning ROS for robotics programming*. Packt Publishing Ltd, 2013.
- [41] P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features,” in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 1, pp. I–I, IEEE, 2001.

- [42] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1, pp. 886–893, IEEE, 2005.
- [43] A. Bar Hillel, R. Lerner, D. Levi, and G. Raz, “Recent progress in road and lane detection: a survey,” *Machine vision and applications*, pp. 1–19, 2014.
- [44] M. R. Petry, A. P. Moreira, and L. P. Reisinst, “Increasing illumination invariance of surf feature detector through color constancy,” in *Portuguese Conference on Artificial Intelligence*, pp. 259–270, Springer, 2013.