NANYANG TECHNOLOGICAL UNIVERSITY
SINGAPORE

Simulation Based Perception Testing for Autonomous Vehicles

Yang Xuehuan

School of Computer Science and Engineering

2022

Simulation Based Perception Testing for Autonomous Vehicles

Yang Xuehuan

School of Computer Science and Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirements for the degree of
Master of Engineering

2022

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research, is free of plagiarised materials, and has not been submitted for a higher degree to any other University or Institution.

04.Jan.2022
....

Date

....

Yang Xuehuan

Supervisor Declaration Statement

I have reviewed the content and presentation style of this thesis and declare it is free of plagiarism and has sufficient grammatical clarity to be examined. To the best of my knowledge, the research and writing are those of the candidate except as acknowledged in the Author Attribution Statement. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

04.Jan.2022

. . . .

Date

Prof. Liu Yang

Authorship Attribution Statement

This thesis **does not** contain any materials from papers published in peer-viewed journals or from papers accepted at conferences in which I am listed as an author.

04.Jan.2022

. . . .

Date

. . . .

Yang Xuehuan

Acknowledgements

I would like to thank my supervisor, Prof LIU Yang from Nanyang Technological University for providing valuable guidance throughout this research. In addition, I would like to thank Professor ZHANG Tianwei from Nanyang Technological University and Dr. ZHOU Yuan and Dr. XIE Xiaofei from Nanyang Technological University for their useful discussions and methodology for the research. I also would like to thank Dr. TANG Yun, and Dr. HAN xingshuo from Nanyang Technological University for their guidance and knowledge transfer during the initial idea development stage and for guiding me for experiments. This work was supported by Singapore National Research Funding. Moreover, I am grateful to my family for caring and continuing assistance to complete my study journey.

Abstract

Perception Testing technologies are widely applied in various scenarios, like industrial and academic research applications for autonomous driving systems. Accurate and robust autonomous driving simulation perception is pivotal for safety-guidance autonomous vehicles (AV). The autonomous vehicle system is facing the main challenges of a complex real-world environment with multi sensors' performance and their neighborhood view with an uncertain environment. The perception module exploits deep learning models to detect surrounding obstacles, including their types, positions, and velocities. However, the issue of LiDAR performance, which is the prediction of multi obstacle properties based on their deep learning model, remains unresolved. In addition, autonomous Vehicle systems rely on deep learning models to collect and modify raw point cloud data. It's crucial to check autonomous vehicles' robustness in several scenarios to generate automotive vehicles more safety and reliable. At this stage, it is easier to perform sensor integration to simulate sensors' performance for Autonomous Driving at various levels of verification. This thesis utilizes a realistic LiDAR point cloud and compares the difference between real-world and simulation environments to test perception modules for autonomous driving platform systems. This thesis proposes a simulation-based testing platform for autonomous vehicles to discover the potential shortage of perception by analysing huge scenarios and testing the suitable sensor performance in multi particular levels of the automated driving platforms. Additionally, this thesis introduces a simulation-based testing method, which involves multi-sensor configurations and includes virtual environment testing with various scenarios. To deal with real-time traffic scenarios, an effective way is utilized to search for the closest scenarios. Also, the proposed method has been tested in popular autonomous vehicle platforms and simulators . To utilize the existing methodology, we perform industry platforms on Baidu Apollo and guide it to assess the quality and enhance the perception system based on a simulation-based testing platform.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The autonomous vehicle perception module is widely used by the industrial and government due to its high accuracy. The safety-oriented autonomous driving platform requires a more accurate and robust autonomous driving environment for the perception module [2]. At this moment, the perception performance affects with simulation environment for real-world raw data. In many cases, an open-source autonomous driving system such as Autoware and Apollo platform is used to identity perception and derive estimates of the accurate obstacles' type, a position as well as velocity. Out of these tasks, obstacle classification is considered one of the challenges, given the possible velocity through deep learning modules. In addition, the autonomous driving system platform (ADS) involves valuable machine learning-powered systems for the automotive platform. IT relies on various hardware to acquire necessary road conditions data and acquires deep learning modules to make real-time decisions during experiments [3]. To understand and analyze the perception module with performance and explore the relationship between the real world and scenario cased perception module [2]. The public transportation system is exposed to the autonomous vehicle, whose information is acquired and applied in majority by industrial and academic research. Hence, the robust and accurate perception modules is a key component and pay more and more attention, in addition, some complex autonomous driving system through simulation testing and collecting data to understand road scenarios. Moreover, digital twin technology transfers the real-world environment into the digital world to implement, evaluate and study the autonomous vehicle perception's approach and comprehensive study.

1.1 Autonomous Driving Level 4

For the experiment, we import Baidu Apollo 6.0 open-source code with an autonomous vehicle level 4 1.1 system to perform object detection tasks. In Autonomous Driving level 4, the vehicle can operate in self-drying mode within a limited distance. And they do not require humans all the time, however, a human still has the option to manually overtake the vehicles. The main difference between ADAS (Advanced Driving Support System) and Autonomous Driving levels is the location of safety, security, and responsibility during situations. The Apollo perception module outputs include 1) Traffic light status. The localization of the traffic light is first detected by the detection network, afterward recognized by the classification model name RTNet. 2) the 3D obstacle tracking information, which includes the obstacle heading, velocity, and classification information. In addition, ADAS is part of the big idea of the Autonomous Driving framework. Baidu's Apollo autonomous driving cars have been tested for over 7.5 million miles by June 2021 in the real-world road and highway.

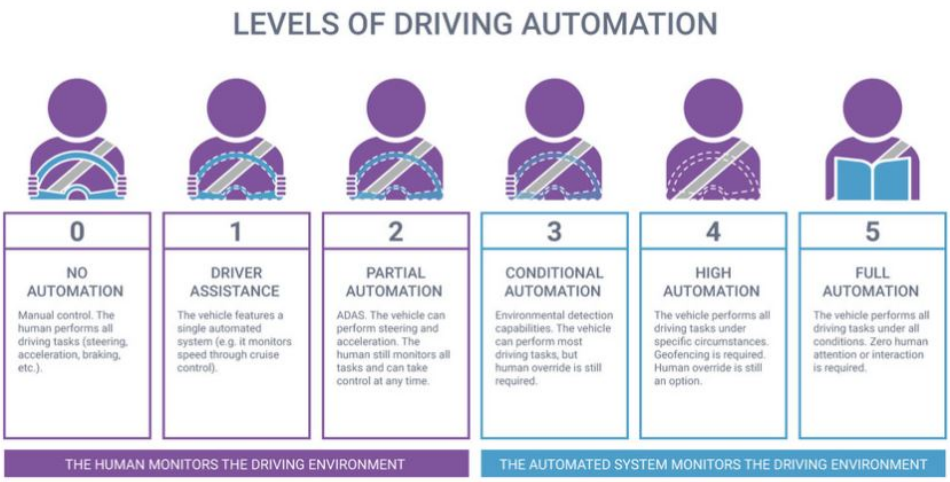


FIGURE 1.1: Apollo Autonomous Driving Level 4

1.2 Confidence issue statement

Although deep learning systems have been highly developed, it shows that deep learning systems for obstacle prediction regarding their type, position, and velocity use a decimal number between 0 and 1, which can be interpreted as a percentage of confidence for the obstacles detecting systems. When we integrate the deep learning model to build a prediction that contributes to a decision. To better understand the predictions module, it dives into three metrics used for object detection issues: accuracy, recall, and precision [4]. We apply a modern deep learning model to estimate the performance of the perception module, test and update different confidence values with various scenarios. It meets the efficient data generation for urban driving scenes to augmented reality for autonomous vehicle driving platforms [5]. We collect the Baidu Apollo sensor data, implement confidence value from Apollo deep learning module, limit the usage of perception module for object detection tasks in their autonomous platform. It's likely to have similar concerns for autonomous vehicles to detect multi obstacles in the application of decision and prediction research.

1.3 Related work in scenario-based testing

In the field of scenario-based test and analysis, it involves the evaluate method and update the sensor perception modules for autonomous vehicle [2] and recent research work has studied scenario-based test architecture during enhancement performance, which is commonly used to detect the type, position, and velocity of the obstacles. With the development of automotive intelligent vehicle task, which is commonly measured by the absolute error of predicting obstacles in the deep learning system. In the end, the scenario-based perception testing method has coming important research topics for evaluating the autonomous vehicle's performance and testing features. Some research papers involve testing scenario methods to understand the real environment [6]. Some research paper involves route coverage testing for an autonomous vehicle to find the corn cases [7] and applies collision avoidance testing methods to test the autonomous vehicle driving system [8]. However, some research direction involves the autonomous driving perception method to verify the safety, robustness, and complexity of various sensors.

1.4 Overview of the method

This work is structured in two parts with presented areas. First of all, we conduct a comprehensive study to integrate the most recent works in academia [9] as well as evaluate the main industry services provides by the Apollo platform. We further integrate a software study as a comparison, which integrates many scenarios. And we collect point clouds raw data from LiDAR devices and acquire image raw data from camera modules within various scenarios to predict obstacles properties. We investigate to simulate the environment and formulate the following research questions.

- *RQ1: Compared GA and random testing, which method can discover perception issues more efficiently?* assesses how well-related work performs in various scenarios for autonomous vehicle testing platforms.
- *RQ2: What kinds of issues have been discovered in Apollo's perception module and how can they cause collisions?* motives the analysis of commonly used in obstacles avoidance and perception scenario-based testing during real-time autonomous running system on multi roads.
- *RQ3: What are the main differences of perception errors between normal and collision executions?* builds on different findings of perception types, position, and velocity to enhance study to academic and factory areas.
- *RQ4: Can the perception module recognize various objects with multi shapes on roads?* enlarges research studies for autonomous vehicles platform in the roads with various unexpected shapes and tests existing in the majority in collecting and acquiring test sets.
- *RQ5: What benefits or experience can AV developers gain from the study?* motives Apollo's development teams to learn from corn cases and reflect similar network architecture with respect to main defects of scenarios perception.

Benefits from the latest research advancements in the deep learning network, our work uses a detection algorithm to acquire confidence value from the Apollo system and change scenarios based on detecting environment for checking model feedback. Our results acquire various corn cases towards sensors detection for perception modules in the industries area, with Apollo sensors for applying sensor fusion techniques to integrate perception modules and to improve safety and accuracy. We find that Apollo open-source platform involves the high definition map for their complicated functions and it integrates detecting dynamic traffic lights, predicting static obstacles, knowing estimation position in global GPS with localization based on real-time LiDAR sensors to create a 3D real-time map. In addition, Apollo developers build different machine learning modules using various deep learning models and maintain the source codes from various corn cases. Our studies may assist developers to reduce maintenance by providing a better abstract of different machine model networks. These findings give us strong motivation to develop a new approach to detect unfair behavior and create high-standard deep learning modules for detecting LiDAR real-time 3D obstacles with some unexpected weather conditions. First, we present an algorithm to detect autonomous vehicle perception modules and detect their obstacles' type with real-time confidence value from a complex environment. Second, we optimize the random and genetic algorithm to compare various scenarios through the simulated environments for dynamic vehicles with road junctions.

Further studies might support perception modules evaluation and comparison management. As we found in Apollo, developers might integrate corn cases in their autonomous vehicles with the driving environment. Moreover, future research studies might involve assistance for developers to select machine learning modules and assists developers to maintain variants of different models.

In this work, this thesis involves the following parts:

- *We propose a simulation-based testing framework for the Apollo driving platform, which involves confidence based on their perception modules.*
- *We find that improve perception performance such reduce the horizontal velocity effects, reduce horizontal type error, and shows fairness score compared to their related work*
- *We evaluate simulation performance and acquire confidence value within the simulation and identify differences in predicting among static vehicles, dynamic vehicles, and pedestrians.*

Paper Structure: Chapter 2 involves a background on autonomous vehicles comparing this work with related work. Chapter 3 provides a comprehensive study for a methodology for enhancement. Chapter 4 studies and evaluates the perception model in the Apollo platform. Chapter 5 describes results and discusses overall research guidance. Chapter 6 concludes this work.

Chapter 2

Background

2.1 Obstacles Avoidance Algorithm

Autonomous Vehicles (AV) are popular in both research platforms and industrial areas. To ensure a robust and accurate autonomous vehicle driving system, perception is one of the key important components to detect surrounding obstacles in road scenarios. Leveraging the use of sensors such as LiDAR, Camera, and Radar, surrounding obstacles in road scenarios can be detected [10]. To achieve the obstacle detection algorithm, many researchers have studied the security of autonomous vehicles to attack multi-sensors in various fields [11], or shooting lasers to LiDAR sensor[12]. It is crucial to have a robust obstacle detecting algorithm to achieve higher accuracy from various perception sources. Overall, various progress has been made when integrated features can be given in the deep learning modules, which are not detected for various obstacle detecting algorithms. Therefore, the reliability of the previous method is questioned and called a novel approach based on the high performance of the LiDAR sensor system.

2.2 Sensor Simulation Analysis

Generally, the most robust and accurate are crucial for autonomous driving systems, especially deep learning detection perception modules for the virtual simulation environment. In another study, LiDARsim sensor [4] studies the better understanding related to the realistic sensor simulation in closed-loop settings under safety-critical sensors with autonomous vehicle platforms. Similar to initial approaches, Gu [13] introduces a deep learning strategy to acquire 3D shapes with given observation with various distances. Simon [14] proposes a traffic simulation environment, a multi-agent behavior perception module to leverage real-world data for learning directly from human explanation. Kelvin [15] presents a comprehensive study to simulate the perception's performance to test the safety and robustness in the simulation environment. Gusmo [16] introduced a method to calibrate the LiDAR sensor based on ray casting perception algorithm and compare with the physical measurements[17]. Some works present a novel simulation-based within the perception layer to obtain the performance of sensors [18]. Yuan [19] introduces an innovative perception method with deep learning technologies. In many cases, sensors consist of thousands of raw data from sensors to check their diversity manually. Therefore, it calls for a systematic approach to analyze the overall confidence value, position, and velocity of multi obstacles from point cloud raw data.

2.3 Point Cloud Integration

Many research works had been generated with various tools for the point cloud generation. It consists of transforming the 3D dataset into a point cloud shape with noisy samples through Generative Adversarial Attack technologies. Mazen [20] introduces adversarial attack for their LiDAR sensor to improve realizable scenarios to test respective outcome[21]. Lin [22] introduces a localization method to acquire point cloud data and build up the maps with dense visual point data. The development of research studies involves object detection methods to test autonomous vehicle perception [23]. This work proposes an object detection method to compare various methods to examine the sensor performance for autonomous vehicle environment and some studies have been introduced methods to compress point cloud data to reduce the object detection algorithm dataset. Tu [24] proposes a novel method to format point clouds for reducing the temporal point cloud data redundancy. The vulnerability of the deep learning module is too adversarial attack algorithm has been heavily investigated in the Point Cloud raw data.

2.4 Data Fusion

An important application of simulation is the validation for multi-sensor fusion functionality, extended Kalman Filter is used in this field[25] and this work introduces an extended Kalman Filter Algorithm to improve the harmonic distortion scenarios. Farag [26] introduces a novel method to fuse LiDAR sensor and radar data for autonomous vehicle simulation environment to improve tracking performance compare with various sensors. Yang [27] proposes a method to fuse the multi-sensor for autonomous vehicles platform to acquire the data fusion algorithm. Zhan [28] proposes a driving system based on data fusion to collect surround-view algorithms to perform the tracking robot movements. Han [29] introduces a methodology to detect lane following scenarios to test autonomous vehicle systems for anomaly detection.

In addition, the monitor system identifies the real-time robot movements, recognizes the types with the surrounding environment. Simulators are applied to optimize the sensor integration with a baseline for the perception detection module. Some testing scenarios experiments are conducted on open source platform and outcome involves the filtering method presented a performance in object detection and obstacle prediction modules[30].

In conclusion, this experiment integrates PTS (Perception Testing System) platform in various obstacles avoidance algorithm and perception testing domains which enable modern deep learning modules to learn from prediction data and various scenes to validate prediction such scenes similar among protected features. Furthermore, We explore a scenario-based testing simulation to acquire the intensity of collecting LiDAR point cloud given specific sensors and 3D dataset.

Chapter 3

Methodology

3.1 Overview

Our objective is to develop a useful tool to simulate complex scenes with obstacles and analyze perception performance in a realistic environment. We can introduce a comprehensive study to evaluate how the accuracy and robustness of obstacle avoidance algorithms, which are used in autonomous vehicle driving platforms, on different types of obstacles in various scenarios. Toward this goal, we separate this work into two parts. First of all, we need to generate multiple scenes of both static and dynamic obstacles to generate a simulation environment. Based on experiment findings, the next part addresses identified issues with weakness and proposes different enhancement and validation techniques to analyze perception module performance with multiply backgrounds and dynamic object detection modules.

This proposed workflow involves the perception module process flow from data collection from simulation ground truth values that are from real perception module from Apollo platform), data processing(Apollo platform), and data analysis to perform the perception module. The module collects ground truth data from an environment simulation that is LGSVL open source code based on obstacles environments. The virtual scenarios

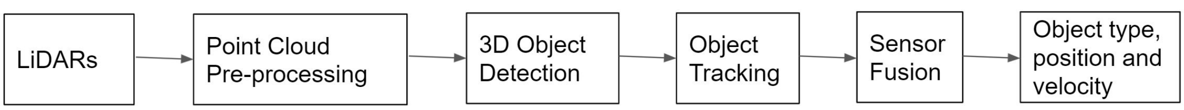


FIGURE 3.1: Workflow of Perception Module

consist of a traffic light with complex scenarios. We integrate the source code to build up a simulate tool for the environment application and it integrates with the data fusion algorithm. One of the main findings in related work is that various obstacles are can't detect in the perception module and they treat stone-like static vehicles which are used to train and evaluate deep learning module performance. After that, we import a data fusion module to enhance the approach towards simulated sensor data and it consists of a fusion data module, which depends on the Kalman Filter Algorithm and feeds into the deep learning module. Hence, the improvement purposes an algorithm approach to simulate and fuse raw sensor data from the initial ground truth environment for the data fusion algorithm. Then we evaluate the PTS module to validate the perception system on target data similarly among point cloud features.

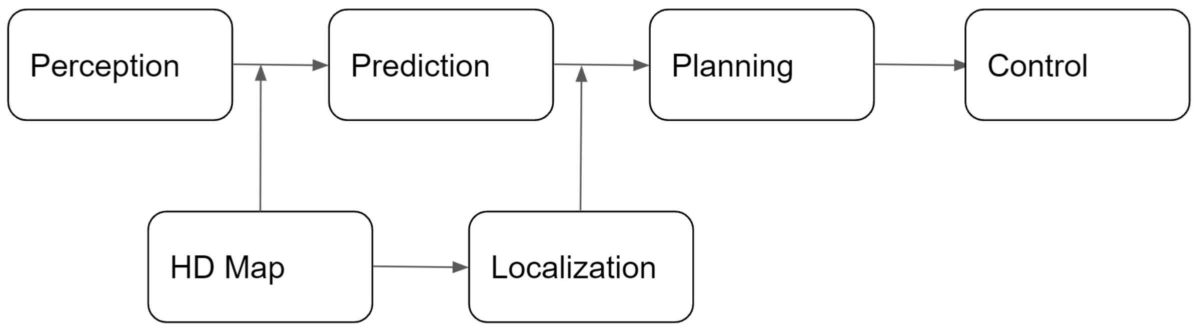


FIGURE 3.2: Apollo Software Structure

Although LiDAR acquires high-quality point clouds with raw data, some obstacles may have less representative point clouds due to occlusion or distance issues. Therefore, we import an attention layer for the encoder module to improve the specific features and We identify that even with best efforts, a perfect point cloud data among the certain distance with limited features and perception component cannot be given the available which is a common issue in the real simulation world. Then, we propose the data evaluation part as the next enhancement to detect obstacles in the Apollo platform with GA and Random algorithms. Furthermore, we study the relationship with multi parts of filtering specifications in the evaluation element.

3.2 LiDAR-based Object Detection

The perception module integrates the capability to recognize obstacles and obtain an object detection algorithm. This sub-module involves the functions to detect obstacle motion and position information (heading and velocity). Tesla imports the Camera-based vision object detection for autonomous driving with 2D images. However, the LiDAR-based object detection to detect 3D obstacles with depth information. Based on the images, we might not collect accurate depth information for the obstacles. The perception module outputs are the 3D obstacle track with the heading and velocity (cyber_channel/apollo/perception/obstacles) and the output of traffic light detection and recognition (cyber_channel/apollo/perception/traffic_light).

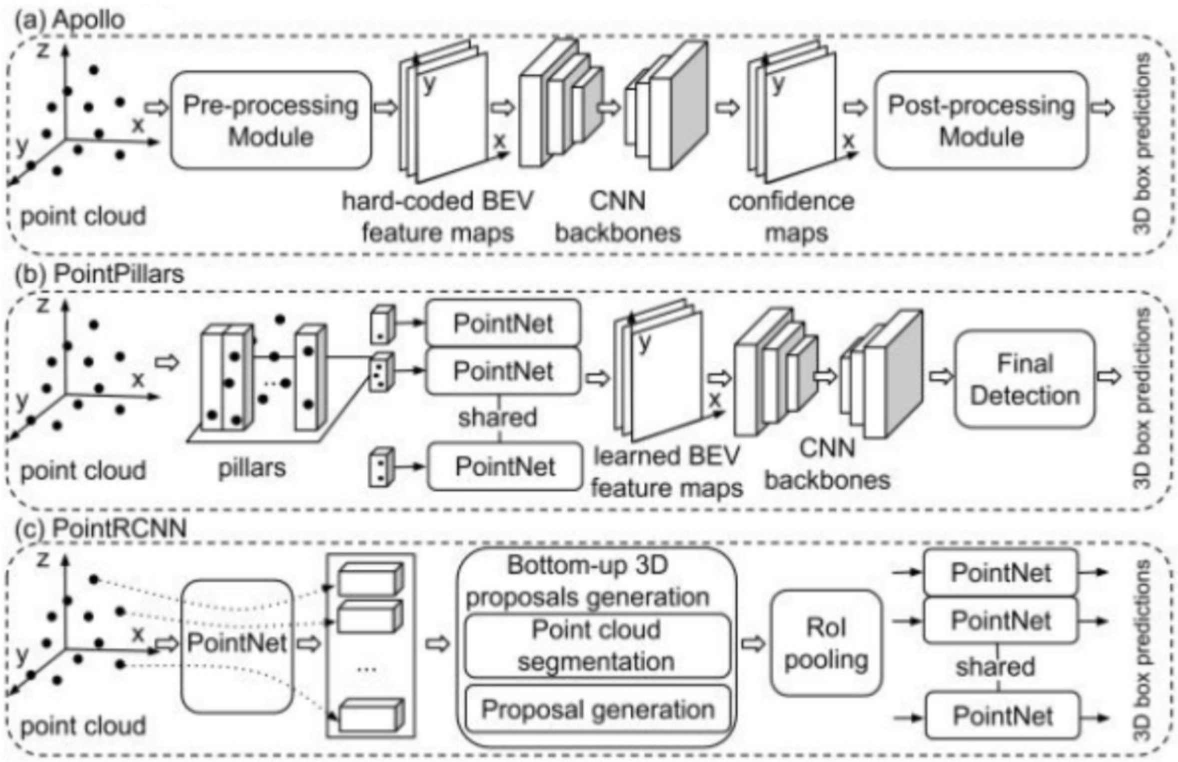


Figure 3.3: Apollo Software Structure

LiDAR-based involves three popular methods to detect obstacle modules.

3.2.1 Bird's-eye view-based

First of all, it is Bird's-eye view (BEV) to obtain a 3D point cloud for LiDAR-based object detection. Here are many existing works that convert LiDAR points cloud with 3D object detection in the AD system [31] [32] [33] [34].

And it conducts the projection and transformation into the top-down view and utilizes the convolution neural network (CNN) to perform the final detection.

3.2.2 Voxel-based

Voxel Net is the important model that slices Voxel Net to shift point clouds position and intensity into voxels and a PointNet analyzes specific features to the detection layer [35] [36] [37] [38]. Many researchers work together to build up this voxel-based object detection. The experiments import the architecture of PointPillars to transform the point cloud into pillars for the detection module.

3.2.3 Point-wise

Instead of transforming point clouds to various voxels for extracting features. Many researchers study [39] [40] [33] various methods to operate on points for object detection algorithm. Recently studies propose to convert the high-quality into 3D space and regress the bounding box for classifying the detection objects.

3.3 Study Design

This section shows the general experiment of the comprehensive study. We evaluate the performance of the perception and evaluation algorithm and identify its fitness and confidence towards obstacles' type to acquire the performance under the testing scenarios. This experiment consists of relevant research questions to understand the virtual environment for the autonomous vehicle perception module. The comprehensive study follows five research questions:

RQ1: Compared random and GA search algorithm, which method can discover perception issues more efficiently? The work look through the various methods to detect the individual perception type and confidence value to compute the gradient from the deep learning module. Therefore, we identify certain simulation environments with various testing scenarios. After that, we understand the relationship between the comprehensive study among the Apollo perception modules and the simulation baseline.

RQ2: What kinds of issues have been discovered in Apollo's perception module and how can they cause collisions? We take a look at the approaches for the perception prediction module to compute fitness value and understand the region of interest (ROI) value to retrieve from specific maps. In addition, it requires a Look-Up table to check the quantization of the region around the vehicle to detect the collision types. We utilize the collision datatype to build up the perception system to understand raw data with specific topics to check collision reasons.

RQ3: What are the differences of the perception errors between normal and collision executions? To sufficiently and profoundly demonstrate the framework's effectiveness and import search algorithms on the various scenarios. We aim to analyze the normal and collision executions by extending the study to the platform, including Apollo, as well as the simulation, which is considered as baseline. Furthermore, the controller can efficiently compare the simulation base and perception module. To understand the testing performance, the experiments are repeated 20 times to reduce the experiment error.

RQ4: Can the perception module recognize various objects with different shapes on roads? To answer this research topic, the experiment acquires a comprehensive understanding of perception testing system model performance and the individual performance regarding the type, position, and velocity. Given various shapes, apollo builds a birds-eye view that is a 2D grid in the local coordinate system. Each point within a predefined range concerning the origin is quantified into one cell of the 2D grid based on its coordinates. Thereby, we could study the various shapes to formulate the conclusive perspective on the evaluation module with the current Perception Testing System(PTS) platform.

RQ5: What benefits or experience can AV developers gain from the study? the test sampling matches to obtain value concerning the perception module performance and update the CNN-based obstacle prediction with deep learning modules. Furthermore, it generates efficiently detecting results for developers to understand the scenarios more reliable.

3.4 Enhancement Approach

Regarding obstacle avoidance algorithms in Chapter 2 As described in Chapter, the importance of related work is to improve the performance of detecting obstacles in an uncertain environment. To generate multi obstacle objects test cases, the issue of obstacles prediction is mostly neglected. Based on the findings of the comprehensive study, we propose algorithmic approaches to (1) improve type confidence, regarding perception detecting module (2) optimize the efforts with data simulation. For both enhancements, we further introduce confidence-awareness using detection as validate skill and data collection technique. Therefore, we acquire the confidence value and select enhancement from a data-driven perspective, and evaluate testing techniques with the proposed benchmark.

3.4.1 Confidence Feedback

To systematically generate the guidance inputs, prior works generally acquire specific approaches, which consist of high effectiveness. Since the confidence generation involves many integration iterations, it is easier to test autonomous vehicles on road to acquire LiDAR point cloud raw data and update the situation. Thus, we need to digital synthesize the impacts of the class probability for obstacles. For example, for overtaking roads, some point clouds are negligible for the more styles of shapes and sizes. We hypothesis that too similar and difficult data testing in respect to training sets. We could utilize the ground detector to acquire the ground surface detector' type. Furthermore, we integrate object height with point cloud are filtered in the post-processing step.

The main purpose of the PTS requirement is to collect confidence values and utilize searching space in a simulation environment. In this thesis, the PTS searching algorithm integrates space by acquiring relevant domain knowledge to select the representation of vehicle attributes for the target obstacles. Therefore, the PTS algorithm integrates the traditional testing method to understand deep learning algorithms with a better description and subscribe confidence values to eliminate the need for features for accurate and robust properties. We propose to illustrate the obstacles issue by building up with detecting algorithm and find the gap with extensive comparison providing implementing details.

3.4.2 Simulation-base

The ground truth for this experiment is the output topic in the simulation and We acquire the simulation ground truth data and compare the error between ground truth and autonomous driving platform with the Apollo perception module. The simulation environment covers the entire scope of challenges including pedestrian detection, lane changing, lane detection, road sign recognition, traffic light, and various obstacles. Hou introduced a simulation-based study and it integrates the perception technologies [41] and it integrates with flexible virtual environment with driving scenarios. In addition, simulation development is a crucial area for research community and it provides baseline for perception module to test autonomous vehicle platform. Apollo open source platform integrating with new machine learning models to improve the effectiveness and accuracy for the perception module[42]. In addition, it provides the open-source code to acquire continuous research and experiments towards an autonomous vehicle for the failure of components and behavior in perception algorithms.

3.5 Testing Method

Accurate and robustness of sensors requires to be tested in the autonomous vehicle simulation environment in various conditions with different scenarios. It also involves a sensor fusion algorithm under the simulation-based environment for autonomous vehicle perception modules [42].

3.5.1 Random Algorithm

In the random algorithm, this thesis focuses on testing autonomous vehicle scenarios with different methods and it involves a random algorithm to acquire case cases and build the autonomous vehicle with road coverage specifications. What's more, the module integrates with perception issues in various scenarios. Autonomous vehicles are in a research and development state, and these system targets to import of the current technology in the future. To test and understand the boundary case scenarios, automated test generation approach and unitize the random algorithm to generate the testing cases [43]. Then, it takes a much longer time to update test cases and generate scenarios to update the testing cases. Traditional driving simulator experiments test components with the autonomous vehicle system.

3.5.2 Genetic Algorithm

In this part, many existing research scenarios are discussed for autonomous vehicle perception. Some testing methods import a Genetic Algorithm to generate testing cases in a simulation environment. GA acquires search algorithms and it consists of the evaluation criterion for various scenarios.

We hypothesize the confidence value with high scores enhances the performance in classification modules and may be considered unreal towards real-world scenarios. To test our hypothesis, we import a benchmark dataset which is from LGSVL with a trained model. In the end, the module consists of obstacles to validate the model's ability to generate unrealistic data from various scenarios and settings [44].

The experimental data are recorded from three scenarios including intersection, interchange, and lane change. We compare the errors between the ground truth obtained from LGSVL and obstacle perception outputs from Apollo, to analyze the impact of different parameters. There are three steps to generate GA test cases. 1) We selected the initial location, target lanes, and obstacle positions of the ego vehicle in the HD map. 2) We mutated the ego vehicle and target vehicle's relative location and velocities to collect confidence feedback for the deep learning module. 3) We recorded the respective velocity, position, and type for next-generation until collision happens.

Chapter 4

Evaluation

We separate the evaluation module into two parts based on the proposed methodology. In the first section, the experiment collects ground truth data from the simulation environment based on objects details. Then we conduct a comprehensive study of the existing state-of-art to analyze module performance. In the end, we conclude various relative experiments in a simulator environment.

4.1 Experiment Establish

Simulation Environment This work presents simulation-based perception testing for autonomous vehicles to evaluate perception object detection systems [1]. This work introduces a comprehensive study to evaluate closed-loop simulation testing for autonomous driving models that include deep learning detection modules learning [1] .

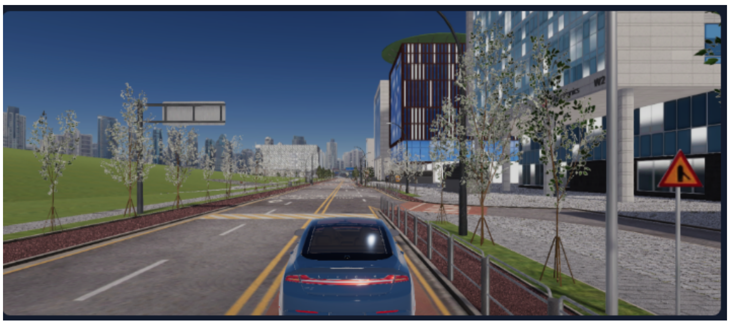


FIGURE 4.1: LGSVL Road Scenarios [1]

In addition, the perception system imports object detection algorithms to simulate the testing scenarios. In the meantime, it involves an evaluation suite to describe scenarios in a virtual environment. The environment involves the ground truth data for testing scenarios to generate the evaluation scenarios and it combines the analysis of the PTS module in a setting with an evaluation environment. LGSVL simulator involves the static road connection, intersection, lights, signal, cones, and other parts to improve the perception module around the roads for a simulation driving environment.

Sensor Models LGSVL is the simulator to perform autonomous vehicles in a virtual environment. It is a testing platform with open source code and generates the various scenarios of the real-world environment and validates with various scales in a simulation environment, for example, Robot Operating System (ROS), autonomous vehicle Apollo, and Autoware Platform. The following table shows the scenarios for various sensors4.1.

Sensors	LiDAR	Camera	Radar
Range	Medium	Short	Long
Object detection	Best	Good	Good
Accuracy	High	Medium	Low
Night Vision	Good	Poor	Good
Rain, snow, Dust	Good	Poor	Good
Cost	High	Medium	Medium
Road Signs	Poor	Best	Poor
Fog	Poor	Poor	Good

TABLE 4.1: Autonomous driving sensor performance analysis

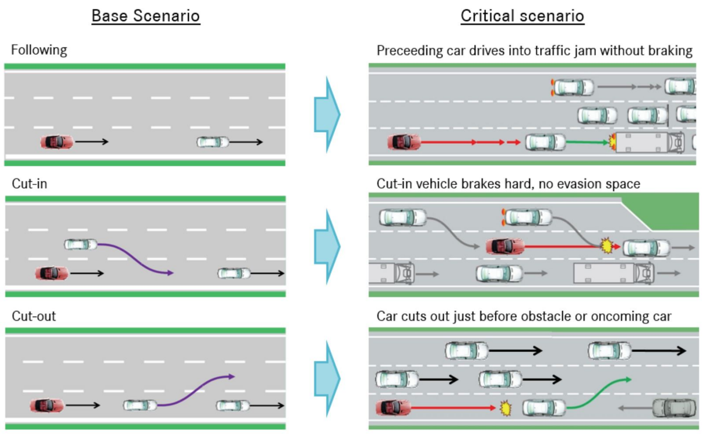


FIGURE 4.2: Basic Scenario and Critical scenario

The module involves the simulated sensor data in the fusion part, and it requires the Kalman Filter to feed into the environment evaluation module. To further accelerate the discovery of point cloud data, simulated data are integrated with initial raw data in the evaluation elements. The following figure shows the autonomous vehicle overview. The module includes LiDAR, RADAR, Camera, and IMU sensor[3]. For autonomous vehicles, it consists of different sensors to perform multiple tasks, for example, autonomous vehicle parking, lane changing, and overtaking scenarios.

The high-performance simulator involves the environment, sensor simulation for autonomous vehicles. To enable vehicle manufactures and future mobility players to accelerate development using simulation, a digital twin method to convert the 3D environment with point cloud and image to real-world scenarios. The project runs a simulation that is accurate reflections of the autonomous driving system in real-world scenario Figure 4.1. In the virtual testing platform, PTS imports virtual environment components includes LiDAR and Radar sensors to fulfill perception functions and understand the potentially outdated information about road signs and infrastructure. In addition, the component involves the software interface to process the signal interfaces to perform safety purposes in Figure 4.2 [42].

It generates systematic testing data for training in the virtual night environment to investigate vehicles' performance in the special traffic situation to extract the typical sensor to recognize object detection algorithm through the deep learning module Figure 4.3.

Perception The perception module involves multiple sensors for the autonomous vehicle platform to detect obstacles. And it detects various objects in different scenarios to achieve the obstacle avoidance algorithm with the position information, type, position, and velocity. Besides, the module provides an online service for training PointPillars models using collection real-time data. To improve the performance of the perception detection module, the dataset is collected in the simulation environment. Collecting raw data from LiDAR sensor in different scenarios covers the autonomous driving environment as much as possible in the city environment and the scenarios imports various types of obstacles such as dynamic vehicle and pedestrians. To demonstrate results and analysis for evaluation methods, the graph shows configurations with various different setups in the simulation environment. Various city environment includes covering driving on a rural road with random scenarios.

The perception module involves the following topics:

16 channel LiDAR sensor data (cyber_channel /apollo/sensor/lidar_front, lidar_rear_ lidar_rear_right) Radar data (cyber_channel /apollo/sensor/radar_front, radar_rear) velocity and angular velocity of ego vehicle (cyber_channel /apollo/localization/pose)

The following graphFigure 4.3 shows the perception segmented object topic with position and type for predicting obstacle algorithm.

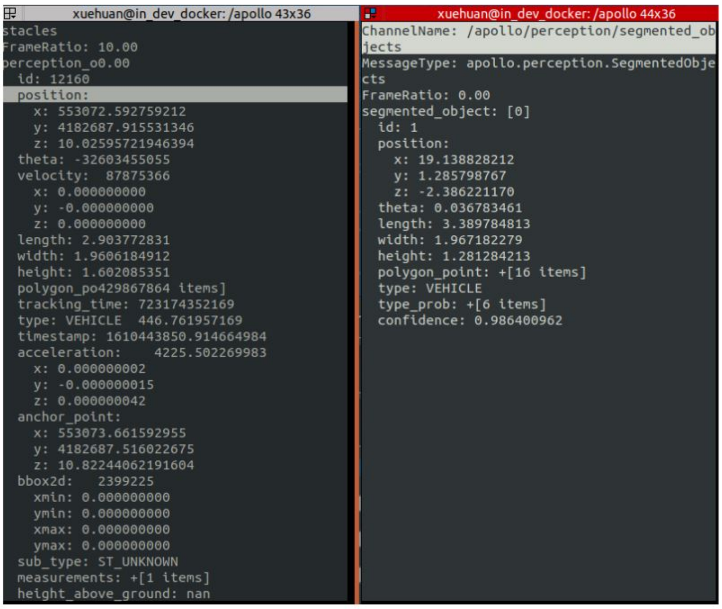


FIGURE 4.3: Cyber Channel Topic

The perception sensor module output includes the following topics and it involves the types, position, and velocity topics in the (cyber_channel/apollo/perception/obstacles) and the output of traffic light detection and recognition (cyber_channel/perception/traffic_light) with a final object with type, distance & velocity.

Practical Setup The experiment is conducted with Ubuntu 18.04 Linux operating system with Intel CPU ES-2600 v4@2.00GHZ x 28 with 256GB equipped with GeForce RTX 2080 Ti GPU to perform various scenario tasks. Baidu Apollo provides the autonomous driving system and driving monitor system and visualization environment called **Dreamview**. We import the Apollo module on the Ubuntu 18.04 operating system to view the perception module output.

The following 4.4shows the various sensors in the LGSVL simulator. The left bottom shows the image segmentation result ground truth for the simulator and the left middle part shows the LiDAR point. In the left top corner, it includes the RADAR sensor around 120 degrees for field of view.

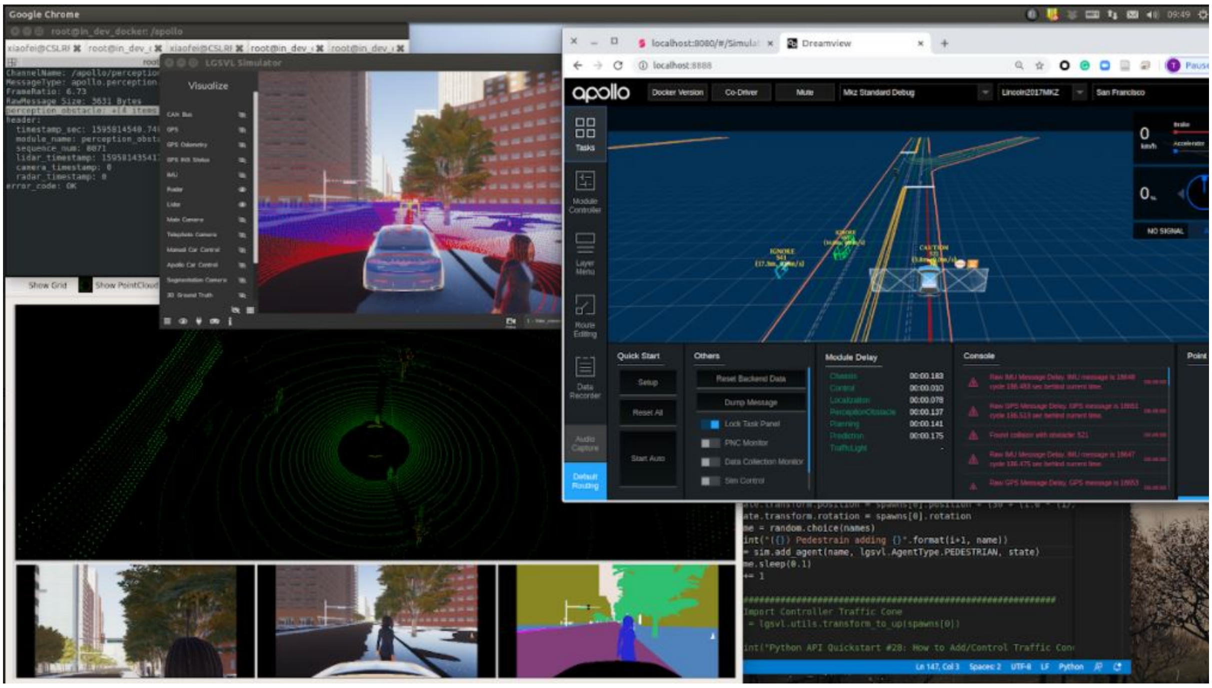


FIGURE 4.4: Sensors in the LGSVL simulator

4.2 Evaluation

4.2.1 Confidence Feedback

For confidence feedback evaluation, this work imports the existing principles towards obstacle avoidance scenarios. The metric is established in assessing to separate the traffic cones, autonomous vehicles, and pedestrians. In detail, the confidence value provides a deep learning model to compare how different protected features are predicted to protect the autonomous vehicle for safety-critical applications.

4.2.2 Correlation Analysis

Pearson's evaluation method uses to calculate the data correlation and analyze the statistical for point cloud data. The application module checks the scenario with correlation possibilities for scenarios [45].

$$r_1 = \frac{\sum_i (a_i - a_n)(b_i - b_n)}{\sqrt{\sum_i (a_i - a_n)^2}\sqrt{\sum_i (b_i - b_n)^2}} \tag{4.1}$$

where a_i is the ith intensity of point cloud data in the testing environment, b_i is the intensity of the ith of point cloud data in the testing environment, a_n is the mean intensity of the testing scenario one and b_n is another mean intensity of testing scenario.

4.3 Comprehensive Study

In this section, we study the Apollo perception module and summarize relative analysis for testing the platform with the LGSVL simulator.

To answer RQ1, we run either method 20 times on all scenarios. Each time, we compute the average number of test cases generated and execution time to trigger a collision. To answer RQ2, we give a detailed analysis of the discovered collisions and discovered three kinds of issues on Apollo's perception module. To answer RQ3, we divide the perception error into 7 categories: shape error, ego vehicle distance error, ego vehicle lateral distance error, ego vehicle longitudinal distance error, ego vehicle velocity error, ego vehicle lateral velocity error, and ego vehicle longitudinal velocity error, then do comprehensive statistical analysis and regression analysis. To answer RQ4, we set the various height scale of pedestrians, various height scales of traffic cones, and import sphere shape objects in the LGSVL simulator to detect the perception performance. To answer RQ5, we focus on different sizes of traffic cones and mutations on a sphere and we summarize our findings and discuss the knowledge that can benefit AV developers during the whole software development life cycle.

RQ1: Compared random and GA search algorithm, which method can discover perception issues more efficiently? We record experiment results by conducting testing cases for Baidu Apollo perception modules with an open-source LGSVL simulator. The table 4.2shows that GA and Random algorithm finds the collision average time. GA and Random algorithms both can find perception limitations for vehicle collision scenarios. However, the result shows that GA is faster to identify the perception errors in the table 4.2. In addition, we finalist the simulator environment and understand the gap between the simulator and the real world.

Algorithm	intersection	lanechange	overtake
GA	3.6	1.4	2.7
random	5.2	2.4	4.1

TABLE 4.2: Testing Cases Execution Timing Taken(hours)

The table 4.3shows the respective results for Levevne test and T-test between distance and its distance error for perception with ground truth. The experiment consists of Levene statistics and Levene P-value for testing results.

intersection	GA	Random
levene_statistic	41.6371	801.9304
levene_pvalue	4.9143e-10	2.0478
T_statistic	155.3099	96.5506
T_pavalue	1.0443e-18	2.945e-10

TABLE 4.3: GA and Random Leneve Test and T-test Result with distance error

Answer to Q1: We import the analysis of the scenarios that contain data with largely different obstacles among type, position, and velocity within the detection module. Hence, various scenarios are tested with significant difficulties when evaluating for application when comparing results with other modules. In addition, the GA algorithm is faster to detect obstacle issues than the random algorithm for various scenarios.

RQ2: What kind of issues have been discovered in Apollo's perception module and how can they cause collisions? Given that RQ1, the raw data are recorded in the simulation platform and testing the perception module with deep learning system in the Apollo platform. The first step is to analyze data from the recording files and PTS analyses time stamps, raw data with point cloud output.

It shows the distance means and distance mean error for the intersection execution to understand the distance parameter relationship during the perception module execution for segmenting objects with predicting obstacles distance.

The table 4.4shows the execution errors for ten cases with various results about experiments. The seven errors as shown in the figure for the autonomous vehicle testing platform. Similar patterns are observed for various scenarios. Here is the list of intersection scenarios with two algorithms to test module performance. From the table, we find that lateral distance error and longitudinal distance error have big possibilities to make the collision of the autonomous vehicle. From the table4.4, the experiment shows that distance error and lateral distance error are 50% for collision scenarios.

intersection	GA	Random
shape error	2	3
dist error	3	2
lateral dist error	4	4
longitudinal dist error	5	4
velocity error	3	2
lateral velocity error	2	3
longitudinal velocity error	1	2

Table 4.4: Failed execution for intersection scenario

GA has the functionality to analyze the continuous and discrete functions and it integrated a method to improve the time iteration. Furthermore, the GA algorithm is quite useful to produce better results when the search space is very large. However, it is important to implement the algorithm correctly, it might fail to reach the optimal solution, because it may not converge.

Answer to Q2: The system error has many types of data to acquire the intersection scenario. It appears that Random Lateral distance error the testing system. Hence, reported performance is mostly included she distance error and lateral distance error for significant difficulties when evaluating the testing scenarios.

RQ3: What are the differences of the perception errors between normal and collision executions RQ2 shows that various perception errors in the PTS experiment to acquire the experiment data and experiment the ability to compare the related work with the reported results with intersection scenarios. In addition, we compare the collision respective with normal scenarios and find the gap for this experiment.

The table shows that the distance means and longitudinal distance mean to have a big gap in a collision and normal scenarios. In addition, lateral velocity std and lateral velocity mean almost has the same value for this experiment. In this intersection scenario, the longitudinal distance has important to determine the collision scenarios.

The table 4.5shows the following approach with various mean and standardization values for GA and Random algorithm. In addition, the experiment to analyze the Levene test and t-Test methods in selected intersection scenarios to analyze the environment in the simulation platform. However, the GA algorithm has a slightly small value than the random algorithm to perform the Levene Test outcome.

Algorithm	mean	std	Levene Test	t-Test
GA	1.026	0.096	p_value: 0.20	p_value: 2.563e-05
Random	1.764	3.542	p_value: 0.48	p_value: 3.578e-02

TABLE 4.5: Significance Testing between GA and Random algorithm in intersection Scenarios

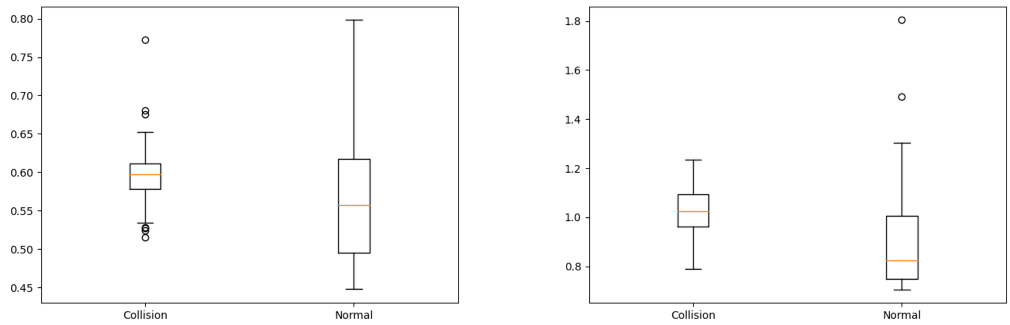
The table 4.6shows the relations with ego vehicles and various parameters to understand the environment. The result shows that distance and distance in latitudinal have a greater relationship based on this experiment than velocity parameter.

ego vehicle	regression model
ego vs shape	y = 0.0075x + 0.5232
ego vs D	y = -0.0041x + 1.0297
ego vs Dx	y = 0.0022x - 0.3479
ego vs Dy	y = 0.0016x + 0.8101
ego vs V	y = -0.0097x + 0.2961
ego vs Vx	y = 0.0004x + 0.0189
ego vs Vy	y = 0.0042x - 0.0758
D vs Dx	y = -0.1648x - 0.1581
D vs Dy	y = 1.2424x - 0.4502
V vs Vx	y = 0.089x + 0.0037
V vs Vy	y = -0.2896x + 0.0229

TABLE 4.6: Regression Model for ego vehicle

The Figure 4.5 shows collision status to detect probability with plotting in python platform to acquire data from visualization part and the results show the deep learning module outcome for distance with perception module. In Figure 4.5 shows the shape overlap changes less than distance parameter with collision and normal situation 4.7 and 4.6. We adjust the pedestrian scales to perform object detection tasks with various environments.

Having identified various shapes in performance when evaluating with the testing dataset, this experiment acquires the deep learning module in Point Pillar module. The result is shown in the table 4.8 to perform the correlation with various errors. The distance in



(a) Shape Overlap (b) Distance

FIGURE 4.5: Boxplot for shape and distance estimation

intersection	Collision	Normal
shape mean	0.7935	0.5936
shape std	0.3089	0.1011
distance mean	1.0263	0.8916
distance std	0.1961	0.1669
lateral distance mean	-0.3272	-0.3500
lateral distance std	0.1701	0.0849
longitudinal distance mean	0.8249	0.646
longitudinal distance std	0.1244	0.1988
velocity mean	0.2554	0.2129
velocity std	0.1896	0.112
lateral velocity mean	0.1222	0.0029
lateral velocity std	0.1235	0.0244
longitudinal velocity mean	-0.1366	-0.0507
longitudinal velocity std	0.2316	0.0573

TABLE 4.7: Difference Mean and standard derivation value in normal and collision executions

ego	shape	D	Dx	Dy	V	Vx	Vy
shape	1	0.483	0.140	0.441	0.106	0.324	0.063
D	0.483	1	-0.224	0.959	0.307	0.198	-0.336
Dx	0.140	-0.22	1	-0.361	0.263	0.330	0.226
Dy	0.441	0.959	-0.361	1	0.159	0.143	-0.389
V	0.107	0.308	0.263	0.159	1	0.240	-0.421
Vx	0.324	0.197	0.330	0.143	0.239	1	0.006
Vy	0.061	-0.33	0.223	-.0389	-0.421	0.006	1

TABLE 4.8: Describe correlation coefficients for seven types of error

longitudinal direction and distance has a large value in this experiment and it might cause too closer for the vehicle detecting surroundings obstacles.

Answer to Q3: Tt has the perception error between normal and collision executions. 1) We validate seven experimental parameters have an impact on perception error as shown in the 4.7. Seven parameter includes shape, distance, lateral distance, longitudinal distance, velocity, lateral velocity, and longitudinal velocity. And each parameter has two indicators mean and standard derivation to describe. 2) The result shows that shape means to value and shape standard volatility are more in the collision scenario than the normal scenario. 3) The lateral velocity mean varies around 10 times in an intersection scenario during a collision than a normal scenario. 4) When the collision happened, the longitudinal velocity error great fluctuation than normal scenarios.

RQ4: Can the perception module recognize various objects with different shapes on roads? RQ3 shows the various collision and normal execution with various parameters to describe the scenarios.

The perception module recognizes the various objects, like traffic cones, different heights for pedestrians, and stones. To perform the accuracy and robustness of the perception module, the experiment builds up to acquire various shapes in simulation roads.

FIGURE 4.6: Pedestrians in the LGSVL simulator with a 0.5 scale

FIGURE 4.7: Pedestrians in the LGSVL simulator with a 0.3 scale

The experiment imports various sizes of traffic cones and multi mutations on different shapes and the experiment shows the following result with range and perception type confidence value. In the LGSVL simulation, we adjust the pedestrian scale table 4.9to test the perception machine learning system. To test the confidence level for the perception output, the module simulates the pedestrian with various ranges and perception types.

LGSVL Scale	Detection Range	Perception
0.3	[2, 200]	No
0.4	[2, 10]	Yes
0.6	[2, 36]	Yes
0.7	[2, 58]	Yes
1.0	[2, 128]	Yes

TABLE 4.9: Pedestrian with different range

The following graph shows the LiDAR detection result with various heights for the object detection modules. The figure 4.8 shows the system can detect the obstacles. However, we shift the obstacles upwards around 2 meters and they disappear for the perception detection system in figure 4.9.

The traffic cones with various shape results are shown in the table 4.10 and it shows the perception system could detect the obstacle as pedestrians with 0.7 scale of the original height.

In addition, the experiment builds autonomous vehicles with traffic cones. Here is the point cloud detected in the point cloud bin file as shown in the table. It includes the

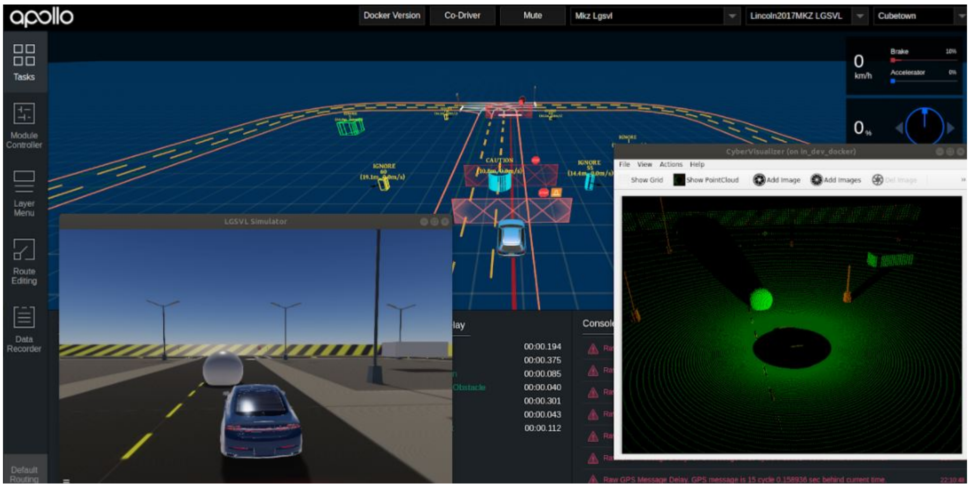


FIGURE 4.8: Object in the LGSVL simulator with an unknown type

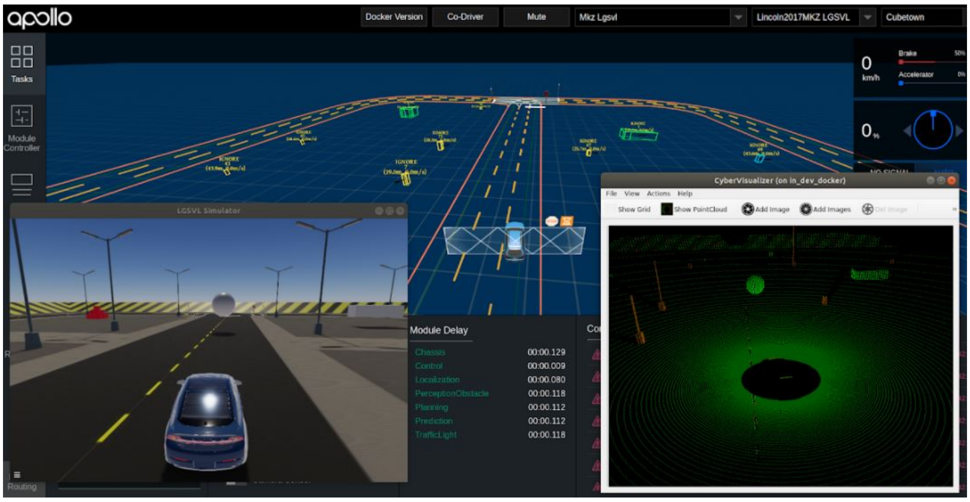


FIGURE 4.9: Object in the LGSVL simulator with higher height

Scale	Range	Perception	Type	Height
0.3	[5, 10]	No	No	No
0.4	[5, 10]	Yes	Unknown	0.44m
0.6	[5, 10]	Yes	Unknown	0.75m
0.7	[5, 10]	Yes	Pedestrian	0.88m
1.0	[5, 10]	Yes	Pedestrian	1.58m

TABLE 4.10: Traffic Cone Performance Results for perception module

point cloud in the robot operating system visualization platform. The result shows the various traffic cone height and detection range distances are important parameters for the perception detection system.

Traffic Cone	Detection Range	Perception	LiDAR number of Points
0.2	[0, 3]	No	0
0.6	[3, 5]	Yes	around 100
0.8	[5, 7]	No	around 150
0.6	[7, 10]	Yes	around 90
0	[10, infinity]	No	0

TABLE 4.11: LiDAR Detection results with traffic cone various scale and range

Answer to RQ4: Yes, we provide three experiments to perform a Perception module that can recognize various objects with different shapes. 1) We place various scales of pedestrians in the LGSVLas shown in Figures 4.7 and 4.6. 2) We adjust various scales of traffic cones as shown in Figures 4.11 and 4.10. We detect traffic cones as unknown type with limited scale in the 4.10 3) We import shape objects in the LGSVL simulator in Figure 4.4. We manually added a sphere object and it is detected with an unknown type by the LiDAR-based object detection module.

RQ5: What benefits or experience can AV developers gain from the study? RQ4 shows the various shapes and pedestrians in the simulation environment to perform the perception module robustness and accuracy for the autonomous vehicle platforms. There are multiply scenarios in the Apollo platform and we import several scenarios for testing autonomous vehicles. Based on the experiment, this work finds that lane change and overtake scenarios are easier to find collision-based with Apollo testing platform compared with lane follow scenario.

Figure 4.12 presents the various vehicles and testing perception modules with various scenarios. It has the distance limitation for testing performance of sensors and to determine the perception module robustness and accuracy, the experiment builds limitations for a testing system with distance to test over leaping effectiveness. The experiment involves a truck, SUV vehicles to test performance with a certain range and it affects distance and overlapping. The outcome results show that a certain range will detect two vehicles as one vehicle. The following graph shows the traffic cone detection results in the simulator environment 4.10 and 4.11.

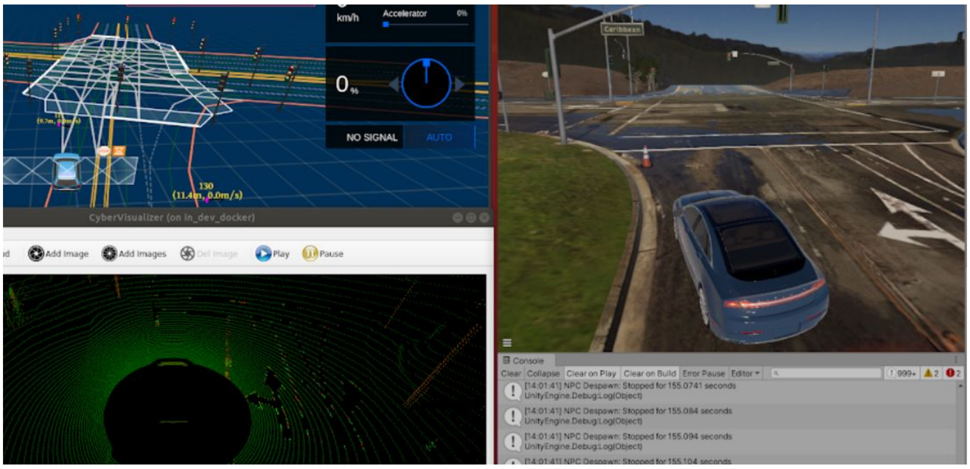


FIGURE 4.10: Traffic Cone in LGSVL simulator

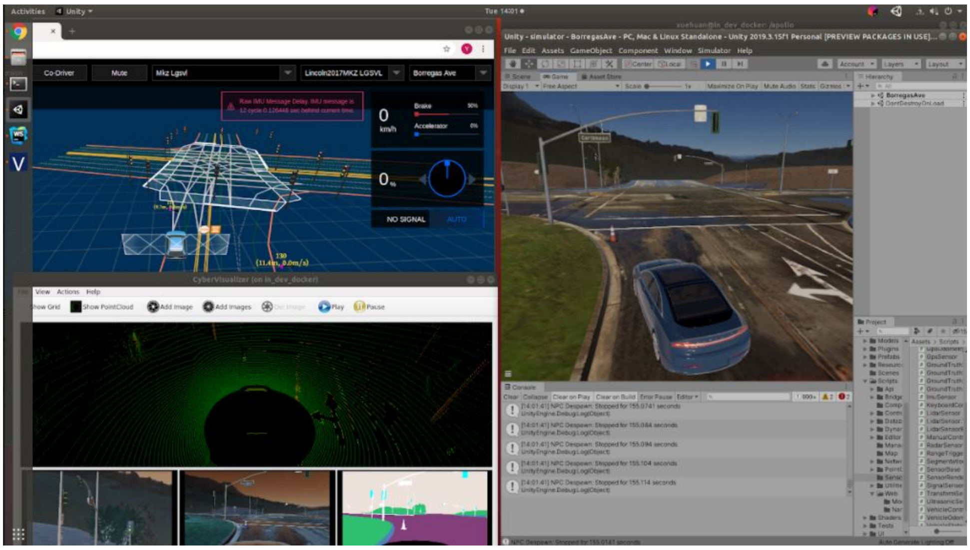
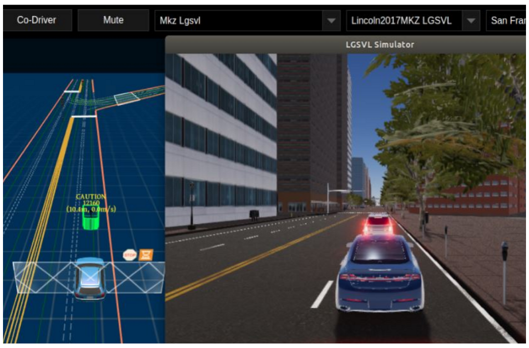
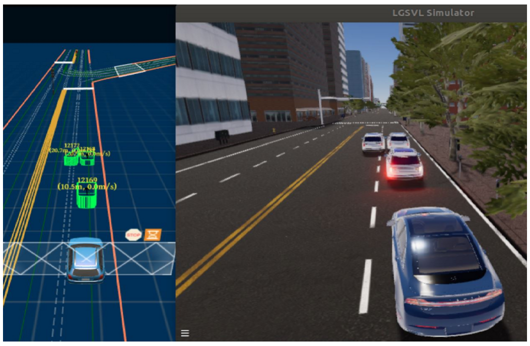


FIGURE 4.11: Traffic Cone in LGSVL simulator

Answer RQ5: This RQ5 benefits the developer to propose the autonomous vehicle perception algorithm to acquire the confidence value. To test the real scenarios in the real world, the experiment builds up various scenarios to find the corn cases. It assists developers to understand the testing platform with a certain range in the autonomous vehicle platform.



(a) Two



(b) Three

(c) Truck

FIGURE 4.12: Apollo Testing scenario for various vehicles

Chapter 5

Discussion and Research Guidance

- **Evaluation** The comprehensive study indicates that commonly required perception and decision modules towards obstacles' type, position, and velocity for the autonomous driving system. The reported results of related work when the dataset tested in multi roads with diverse scenarios. When enlarging the study to compare baseline with ground truth data from LGSVL simulation and we conclude a root cause that different shape sizes of different obstacles with various performance with low-diversity training sets, which makes it difficult to recognize obstacles in several scenarios and the most related work are less performance as reported in the analysis setting when facing unexplored and diverse data input.

  Research guidance: This task integrates an important step towards a diverse and simulation dataset with a baseline among protected features for comparing related work. We identify a large gap between simulation and real-world environment. For example, the simulator raw sensor data are filtered in various sensor fusion modules, which involves in extend the Kalman Filter algorithm and feeds into the prediction modules for a complex simulation environment. Thus, a possible direction is to identify gaps between simulate and real-world environment when evaluating the perception module whether it relies on the possibilities of perception component with unrealistic effect. Another possible direction is to establish a common obstacle dataset benchmark for evaluating and comparing detection modules and algorithms. In addition, it is possible to integrate multi-sensor data to perform perception search algorithms, which greatly reduces confidence value and reduces physical testing costs in a simulation environment.

- **Confidence Value** In the experiment, our result indicates simulation perception systems have a significant meaning towards the performance of detecting numerous obstacles within the open-source Apollo autonomous vehicle platform. Moreover, the deep learning model predicts uncertain confidence value from input data point cloud data and learns from the preset environment data, then it generates a relative confidence value to predict obstacle types, position, and velocity with a converter to understand the simulation environment. In some cases, we observe various obstacles with dynamic velocity varies out offset value in changing lane scenarios.

  Research Guidance: various standards might be developed to prove that obstacle avoidance systems meet specification requirements and classify various scenarios to follow velocity guidance research direction. To further integrate fusion module towards simulation research guidance, feedback and acquire the obstacle outlines might be easier to access, which are developed on obstacles simulation environment. In the autonomous vehicle domain, we import this work to utilize perception limits and acquire useful raw data to acquire detection modules usefully. Using Perception Testing System tools to generate the most effective detection can be identified to improve obstacle avoidance algorithm. Finally, we make automotive vehicles to meet performance with various datasets and it integrates a slight decrease in performance of various autonomous vehicle detection algorithms. In particular, the perception module reflects the distribution of the perception dataset with respectively evaluating and testing modules. Thereby, the relevance of confidence value should be high in the deep learning training and testing module and it is sufficiently addressed in academic and industry platforms.

- **Effectiveness of enhancement** the experiment explores the training data within the detection module. Furthermore, the effectiveness of distribution-awareness using PTS techniques is integrated with the autonomous vehicle platform. Thereby, PTS servers as a control and validation measure for data diversity and deep learning perception module. In addition, this work analyzes and identifies the distribution with various scenarios to improve scenarios generation and increase the safety to generalize with unrealistic data. A possible direction to link with an effective platform in a sensor point cloud environment[46], presents an efficient learnable clustering module, for varying instance sizes and a dynamic environment that integrates object detection algorithms for various scenarios.

  Research guidance: A possible direction is to import the PTS score with confidence value and reconstruct the filter data as a guiding mechanism, which controls the diversity of the perception module and integrates a direct deep learning module for feedback and approach module. Another possible direction is that demonstrates multiple methodologies contain complementary information and improve the safety and robustness of the perception module. The experiment involves a generated platform around vehicle interconnection networks, which secures the object detection for robustness to hand complex applications and safety-critical platforms[47].

- **Industry guidance:** While autonomous vehicle systems may rely on one module to another, the testing platform is a complex system that involves many sub-components. The significant sensor parameter with various locations, points of view, might affect the autonomous vehicle detection results. The Apollo system consists of hardware and software components to discuss the perspective blocks required within the system from collecting data and exporting various data fusion results. Developers might focus on the complex scenarios to detect obstacles with complex maps, for example, when changing lanes and overtaking scenarios, it detects the stone as a vehicle and does not affect the perception system platform. However, it misses detecting the traffic cones with various shapes to make collision with cones. The intersection and interchange scenario is relevant to reversal the performance difference in various setups in the platform. Another it exposed by learning the position error with various LiDAR hardware in the sensor simulation environment and the simulation show that the perception module miss detecting the pedestrians and autonomous vehicle still moving forward towards the target.

Chapter 6

Conclusion

In this work, we analyze the performance of the introduced method and discuss the safety and robustness of integrating modules and experiments relying more on perception 3D obstacle detection tasks. We proposed a simulated-based testing platform for evaluating various scenarios with autonomous vehicles for various sensors integrating, configuring, and analyzing for the autonomous driving platform system. It is urgent marketing demand for simulation progress that involves reliable sensor data in a virtual testing environment with random and genetic algorithm methods. The evaluation suite is a key component that assists autonomous vehicle systems during vehicle development and assists the performance overview with various scenarios. We identify the lower bound of the performance of the simulation environment and build the autonomous driving system more reliable and secure.

We conduct a simulation-based testing on the state-of-the-art Apollo autonomous vehicle platform and we focus on where a physics-based simulation method that extends the evaluation methodology by considering physical effects. We import the modules for the autonomous driving system in various modules with high noise and variability and our results show that the existing Apollo system varies in performance with the perception error on the obstacle's type, position, and velocity error. What is more, we are able to enhance the simulation-based environment for predicting the velocity with absolute error. This work points out the important step to discover the autonomous vehicle system for the simulation-based platform, which enables the confidence of the obstacles for deep learning training modules.

In the future, we will focus on where a physics-based simulation method that extends our evaluation methodology by considering physical environment effects and it protects various modules for the autonomous driving system in multiple scenarios with high noise and variability to collaborate the obstacles of the type, position, and velocity properties. Simulation-based perception testing integrates an important role in developing accurate and robust software to generate huge testing cases to acquire perception module performance.

Bibliography

[1] Andrea Piazzoni, Jim Cherian, Martin Slavik, and Justin Dauwels. Modeling sensing and perception errors towards robust decision making in autonomous vehicles. *arXiv preprint arXiv:2001.11695*, 2020. viii, 17

[2] Liheng Xu, Chi Zhang, Yuehu Liu, Le Wang, and Li Li. Worst perception scenario search for autonomous driving. In *2020 IEEE Intelligent Vehicles Symposium (IV)*, pages 1702–1707. IEEE, 2020. 1, 3

[3] Zi Peng, Jinqiu Yang, Tse-Hsun Chen, and Lei Ma. A first look at the integration of machine learning models in complex autonomous driving systems: a case study on apollo. In *Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, pages 1240–1250, 2020. 1, 19

[4] Sivabalan Manivasagam, Shenlong Wang, Kelvin Wong, Wenyuan Zeng, Mikita Sazanovich, Shuhan Tan, Bin Yang, Wei-Chiu Ma, and Raquel Urtasun. Lidarsim: Realistic lidar simulation by leveraging the real world. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11167–11176, 2020. 3, 6

[5] Hassan Abu Alhaija, Siva Karthik Mustikovela, Lars Mescheder, Andreas Geiger, and Carsten Rother. Augmented reality meets computer vision: Efficient data generation for urban driving scenes. *International Journal of Computer Vision*, 126(9):961–972, 2018. 3

[6] Ziyuan Zhong, Yun Tang, Yuan Zhou, Vania de Oliveira Neves, Yang Liu, and Baishakhi Ray. A survey on scenario-based testing for automated driving systems in high-fidelity simulation, 2021. 3

[7] Yun Tang, Yuan Zhou, Fenghua Wu, Yang Liu, Jun Sun, Wuling Huang, and Gang Wang. Route coverage testing for autonomous vehicles via map modeling. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11450–11456, 2021. doi: 10.1109/ICRA48506.2021.9560890. 3

[8] Yun Tang, Yuan Zhou, Yang Liu, Jun Sun, and Gang Wang. Collision avoidance testing for autonomous driving systems on complete maps. In *2021 IEEE Intelligent Vehicles Symposium (IV)*, pages 179–185, 2021. doi: 10.1109/IV48863.2021.9575536. 3

[9] Maike Hartstern, Viktor Rack, Mohsen Kaboli, and Wilhelm Stork. Simulation-based evaluation of automotive sensor setups for environmental perception in early development stages. In *2020 IEEE Intelligent Vehicles Symposium (IV)*, pages 858–864. IEEE, 2020. 3

[10] Yulong Cao, Ningfei Wang, Chaowei Xiao, Dawei Yang, Jin Fang, Ruigang Yang, Qi Alfred Chen, Mingyan Liu, and Bo Li. Invisible for both camera and lidar: Security of multi-sensor fusion based perception in autonomous driving under physical-world attacks. In *2021 IEEE Symposium on Security and Privacy (SP)*, pages 176–194. IEEE, 2021. 6

[11] Yujie Li, Xing Xu, Jinhui Xiao, Siyuan Li, and Heng Tao Shen. Adaptive square attack: Fooling autonomous cars with adversarial traffic signs. *IEEE Internet of Things Journal*, 8(8):6337–6347, 2020. 6

[12] Yulong Cao, Chaowei Xiao, Benjamin Cyr, Yimeng Zhou, Won Park, Sara Rampazzi, Qi Alfred Chen, Kevin Fu, and Z Morley Mao. Adversarial sensor attack on lidar-based perception in autonomous driving. In *Proceedings of the 2019 ACM SIGSAC conference on computer and communications security*, pages 2267–2281, 2019. 6

[13] Jiayuan Gu, Wei-Chiu Ma, Sivabalan Manivasagam, Wenyuan Zeng, Zihao Wang, Yuwen Xiong, Hao Su, and Raquel Urtasun. Weakly-supervised 3d shape completion in the wild. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*, pages 283–299. Springer, 2020. 7

[14] Simon Suo, Sebastian Regalado, Sergio Casas, and Raquel Urtasun. Trafficsim: Learning to simulate realistic multi-agent behaviors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10400–10409, 2021. 7

[15] Kelvin Wong, Qiang Zhang, Ming Liang, Bin Yang, Renjie Liao, Abbas Sadat, and Raquel Urtasun. Testing the safety of self-driving vehicles by simulating perception and prediction. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVI 16*, pages 312–329. Springer, 2020. 7

[16] Guilherme Ferreira Gusmão, Carlos Roberto Hall Barbosa, and Alberto Barbosa Raposo. Development and validation of lidar sensor simulators based on parallel raycasting. *Sensors*, 20(24):7186, 2020. 7

[17] Raghu Changalvala and Hafiz Malik. Lidar data integrity verification for autonomous vehicle. *IEEE Access*, 7:138018–138031, 2019. 7

[18] John M Scanlon, Kristofer D Kusano, Tom Daniel, Christopher Alderson, Alexander Ogle, and Trent Victor. Waymo simulated driving behavior in reconstructed fatal crashes within an autonomous vehicle operating domain, 2021. 7

[19] Chaochun Yuan, Tong Wang, Jie Shen, Youguo He, and Shuofeng Weng. A new active safety distance model of autonomous vehicle based on sensor occluded scenes. *International Journal of Modelling and Simulation*, 41(3):195–208, 2021. 7

[20] Mazen Abdelfattah, Kaiwen Yuan, Z Jane Wang, and Rabab Ward. Adversarial attacks on camera-lidar models for 3d car detection. *arXiv preprint arXiv:2103.09448*, 2021. 7

[21] Jiachen Sun, Yulong Cao, Qi Alfred Chen, and Z Morley Mao. Towards robust lidar-based perception in autonomous driving: General black-box adversarial sensor attack and countermeasures. In *29th {USENIX} Security Symposium ({USENIX} Security 20)*, pages 877–894, 2020. 7

[22] Xiaohu Lin, Fuhong Wang, Bisheng Yang, and Wanwei Zhang. Autonomous vehicle localization with prior visual point cloud map constraints in gnss-challenged environments. *Remote Sensing*, 13(3):506, 2021. 7

[23] Duarte Fernandes, António Silva, Rafael Névoa, Cláudia Simões, Dibet Gonzalez, Miguel Guevara, Paulo Novais, João Monteiro, and Pedro Melo-Pinto. Point-cloud based 3d object detection and classification methods for self-driving applications: A survey and taxonomy. *Information Fusion*, 68:161–191, 2021. 7

[24] Chenxi Tu. Point cloud compression for 3d lidar sensor. In *Frontiers of Digital Transformation*, pages 119–134. Springer, 2021. 7

[25] Jianxiang Zhang, Wenzhu Huang, Wentao Zhang, and Fang Li. Improved dfb-fl sensor interrogation with low harmonic distortion based on extended kalman filter. *Journal of Lightwave Technology*, 2021. 8

[26] Wael Farag. Kalman-filter-based sensor fusion applied to road-objects detection and tracking for autonomous vehicles. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 235(7):1125–1138, 2021. 8

[27] Jiachen Yang, Shan Liu, Hansong Su, and Ying Tian. Driving assistance system based on data fusion of multisource sensors for autonomous unmanned ground vehicles. *Computer Networks*, 192:108053, 2021. 8

[28] W Zhao, T Li, B Qi, Q Nie, and T Runge. Terrain analytics for precision agriculture with automated vehicle sensors and data fusion. sustainability 2021, 13, 2905, 2021. 8

[29] Xingshuo Han, Kangjie Chen, Yuan Zhou, Meikang Qiu, Chun Fan, Yang Liu, and Tianwei Zhang. A unified anomaly detection methodology for lane-following of autonomous driving systems. In *2021 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom)*, pages 836–844. IEEE, 2021. 8

[30] Hongjian Wei, Yingping Huang, Fuzhi Hu, Baigan Zhao, Zhiyang Guo, and Rui Zhang. Motion estimation using region-level segmentation and extended kalman filter for autonomous driving. *Remote Sensing*, 13(9):1828, 2021. 8

[31] Ming Liang, Bin Yang, Shenlong Wang, and Raquel Urtasun. Deep continuous fusion for multi-sensor 3d object detection. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 641–656, 2018. 11

[32] Gregory P. Meyer, Ankit Laddha, Eric Kee, Carlos Vallespi-Gonzalez, and Carl K. Wellington. Lasernet: An efficient probabilistic 3d object detector for autonomous driving. URL `http://arxiv.org/abs/1903.08701v1`. 11

[33] Bin Yang, Wenjie Luo, and Raquel Urtasun. Pixor: Real-time 3d object detection from point clouds. URL `http://arxiv.org/abs/1902.06326v3`. 11, 12

[34] Ming Liang, Bin Yang, Shenlong Wang, and Raquel Urtasun. Deep continuous fusion for multi-sensor 3d object detection. URL `http://arxiv.org/abs/2012.10992v1`. 11

[35] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. URL `http://arxiv.org/abs/1711.06396v1`. 12

[36] Alex H. Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. URL `http://arxiv.org/abs/1812.05784v2`. 12

[37] Bei Wang, Jianping An, and Jiayan Cao. Voxel-fpn: multi-scale voxel feature aggregation in 3d object detection from point clouds. URL `http://arxiv.org/abs/1907.05286v2`. 12

[38] Johannes Lehner, Andreas Mitterecker, Thomas Adler, Markus Hofmarcher, Bernhard Nessler, and Sepp Hochreiter. Patch refinement – localized 3d object detection. URL `http://arxiv.org/abs/1910.04093v1`. 12

[39] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3d object proposal generation and detection from point cloud. . URL `http://arxiv.org/abs/1812.04244v2`. 12

[40] Shaoshuai Shi, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. From points to parts: 3d object detection from point cloud with part-aware and part-aggregation network. . URL `http://arxiv.org/abs/1907.03670v3`. 12

[41] Yunfei Hou, Yunjie Zhao, Aditya Wagh, Longfei Zhang, Chunming Qiao, Kevin F Hulme, Changxu Wu, Adel W Sadek, and Xuejie Liu. Simulation-based testing and evaluation tools for transportation cyber–physical systems. *IEEE Transactions on Vehicular Technology*, 65(3):1098–1108, 2015. 15

[42] Hans-Peter Schöner. Simulation in development and testing of autonomous vehicles. In *18th Stuttgart international symposium, Stuttgart*, pages 03–14, 2018. 15, 19

[43] Cumhur Erkan Tuncali and Georgios Fainekos. Rapidly-exploring random trees-based test generation for autonomous vehicles. *arXiv preprint arXiv:1903.10629*, 2019. 15

[44] Ahmed Mateen, Marriam Nazir, and Salman Afsar Awan. Optimization of test case generation using genetic algorithm (ga). URL `http://arxiv.org/abs/1612.08813v1`. 16

[45] A Miranda Neto, A Correa Victorino, Isabelle Fantoni, Douglas Eduardo Zampieri, Janito Vaqueiro Ferreira, and Danilo Alves Lima. Image processing using pearson's correlation coefficient: Applications on autonomous robotics. In *2013 13th International Conference on Autonomous Robot Systems*, pages 1–6. IEEE, 2013. 21

[46] Fangzhou Hong, Hui Zhou, Xinge Zhu, Hongsheng Li, and Ziwei Liu. Lidar-based panoptic segmentation via dynamic shifting network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13090–13099, 2021. 34

[47] Giancarlo Di Biase, Hermann Blum, Roland Siegwart, and Cesar Cadena. Pixel-wise anomaly detection in complex driving scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16918–16927, 2021. 35

Appendix

Table 1: Difference Mean value in collision and normal executions

intersection	Collision	Normal
shape mean	0.5935	0.5846
shape std	0.0389	0.1011
distance mean	1.0263	0.8916
distance std	0.0961	0.1669
lateral distance mean	-0.3272	-0.3500
lateral distance std	0.0701	0.0849
longitudinal distance mean	0.8249	0.646
longitudinal distance std	0.1244	0.1988
velocity mean	0.2054	0.2129
velocity std	0.0896	0.112
lateral velocity mean	0.0222	0.0029
lateral velocity std	0.0335	0.0244
longitudinal velocity mean	-0.0366	-0.0507
longitudinal velocity std	0.0616	0.0573

Table 2: LiDAR Detects Traffic Cone Range

Traffic Cone	Detection Range	Perception	LiDAR number of Points
0.2	[0, 3]	No	0
0.6	[3, 5]	Yes	around 100
0.8	[5, 7]	No	around 150
0.6	[7, 10]	Yes	around 90
0	[10, infinity]	No	0