Masters Program in Geospatial Technologies

Comparison of Different LiDAR Sensors for Safe Object Detection Using Deep Learning Algorithm

Muhammad Ammad

Dissertation submitted in partial fulfilment of the requirements
for the Degree of *Master of Science in Geospatial Technologies*

NOVA IMS Information Management School
Universitat Jaume I
Westfälische Wilhelms-Universität Münster

Comparison of Different LiDAR Sensors for Safe Object Detection Using Deep Learning Algorithm

Dissertation supervised by:

Prof. Dr. rer. nat. Toralf Trautmann

Automotive Mechatronics,

University of Applied Sciences,

Dresden,

Germany

Joaquín Huerta Guijarro, PhD

Associate Professor, Institute of New Imaging Technologies (INIT),

Universitat Jaume I (UJI),

Castellon de la Plana,

Spain

Dr. Marco Painho, PhD

Full Professor, NOVA Information Management School (NOVA IMS),

Universidade NOVA de Lisboa,

Lisbon,

Portugal

February 19, 2024

Acknowledgement

I would like to express my sincere gratitude to all those who have supported me in making this thesis a reality. First and foremost, I would like to thank my parents for their unconditional love and support throughout this journey.

I would like to thank Franziskus for his invaluable assistance during the data collection process and technical support throughout the project duration. His expertise and hands-on help were instrumental in developing the experimental setup and collecting high-quality datasets from different scenarios.

I am deeply grateful to my thesis supervisor Professor Toralf Trautmann for his continuous guidance and for placing his trust in my abilities by allowing me to work independently in his laboratory. His insightful feedback and encouragement helped me stay motivated during various phases of the research work.

I am thankful to Roquia Salam for providing helpful feedback during the initial stages of developing my thesis proposal. Her timely guidance helped refine my ideas and put me on the right track to frame the research objectives systematically.

I would also like to acknowledge Asim Qadeer for his assistance with scientific writing and literature review. His constructive inputs helped me present my work objectively and placed it in the right context with other related studies.

I appreciate Tom Irrasch and Dirk Engert for facilitating access to the desktop computer in the lab which was instrumental in training the complex models.

I offer my regards and blessings to all of those who supported me in any respect during the completion of the project. Finally, last but not the least, I present my sincere thanks to my family for their unparalleled love, support, motivation and prayers.

Abstract

This study investigates the use of different light detection and ranging (LiDAR) sensors for object detection tasks using deep learning algorithms in autonomous driving applications. Three LiDAR sensors - LS LiDAR, Livox, and Ouster - were tested by collecting point cloud data from various road scenes involving cars and pedestrians. The data was labelled using MATLAB's Ground Truth Labeler and used to train a Complex YOLO-V4 neural network model. The performance of the trained model was evaluated on test data from each sensor using mean Intersection over Union (IoU) scores, Average Orientation Similarity (AOS) and Average Precision (AP) metrics. Results showed that LS LiDAR achieved a mean IoU of 0.322 for cars but 0.229 for pedestrians, while Livox scored 0.397 and 0.265 respectively. Ouster had the best results with 0.471 for cars and 0.332 for pedestrians, demonstrating its strong object classification capabilities. Point clouds from Ouster also exhibited higher localization performance compared to other sensors based on IoU value graphs. Ouster's high-resolution 3D point clouds worked optimally with the YOLO-V4 model to achieve the highest accuracy for both vehicle and pedestrian detection among the three LiDARs tested. The study provides insights into selecting the appropriate LiDAR sensor for autonomous driving applications based on object detection performance.

Contents

List of Figures

List of Tables

List of Abbreviation

LiDAR - Light Detection and Ranging

ADAS - Advanced Driver Assistance Systems

GIS/RS - Geographic Information Systems/Remote Sensing

CNNs - Convolutional Neural Networks

IoU - Intersection over Union

AOS - Average Orientation Similarity

AP - Average Precision

GTL - Ground Truth Labeller

ROI - Region of Interest

GPU - Graphical Processing Unit

1 Introduction

Light Detection and Ranging (LiDAR) technology has proved to be effective in detecting and identifying objects, especially for autonomous vehicles and robotics. The use of deep learning and machine learning algorithms for detecting objects from LiDAR point clouds has gained significant attention in various fields such as autonomous driving, robotics, and geospatial analysis. Several popular algorithms have been developed and applied for this purpose. The LiDAR point cloud resolution in is different based on the specifications of each LiDAR sensor, reflecting the diverse capabilities and technologies employed by various manufacturers. The range of availability of different LiDAR sensors provide users with the flexibility to choose the sensor according to their particular requirements, whether those involve mapping, surveying, autonomous vehicles, or environmental monitoring.

The core objective of this research is to harness the rich data provided by LiDAR point clouds and apply it to train a sophisticated Complex YOLO-V4 model. The YOLO (You Only Look Once) architecture, known for its efficiency and accuracy in real-time object detection, forms the foundation of our model. However, in this study, the Complex YOLO-V4 model is tailored specifically to address the unique challenges posed by different LiDAR sensors in study.

1.1 Aims and Objectives

The main objective of this research is to compare the performance of different LiDAR sensors for object detection tasks in the context of autonomous driving applications. The research will be carried out in the following steps:

- Train the Complex YOLO-V4 neural network model on the labeled LiDAR point cloud data for three different sensors namely LS LiDAR, Livox, and Ouster.
- Evaluate the performance of the trained model on unseen test data from each sensor, using metrics like mean IoU, AOS, and AP.
- Performance comparison to determine which sensor is most suitable for object detection tasks related to autonomous driving.

This research is to determine the best LiDAR sensor for autonomous driving applications based on the comparison of object detection performance using deep learning techniques.

1.2 Methodological Approach

This research follows a multi-step methodological process to achieve its objectives. The process involves data collection, data pre-processing, data labelling, training of a complex YOLO-V4 model, model evaluation, and performance comparison.

- The first step involves collecting LiDAR point cloud data from three different sensors, namely LS LiDAR, Livox, and Ouster. Scenes captured include individual and grouped cars as well as pedestrians under different conditions.
- The second step is data pre-processing, where the raw point cloud data from different sensors is converted to a common format (.PCD) using sensor-specific conversion tools.
- The third step is data labelling, where objects in the point cloud data are labelled using the MATLAB Ground Truth Labeler tool. Two classes, namely cars and pedestrians, will be labeled. The labelled data will be exported in .mat format for training.
- The fourth step involves training a complex YOLO-V4 model on 70% of the labelled data using the parameters discussed in section 4.3. Training will be performed using a GPU for faster processing.
- The fifth step is model evaluation, where the trained model will be evaluated on the remaining 30% of unseen test data.
- The final step is performance comparison. Results will be compared between the 3 sensors to analyze which sensor performs the best. Advantages and limitations of each sensor will be discussed.

2 Literature Review

Various sources on object detection-based LiDAR algorithms are included in the literature review. The first section summaries literature about the existing techniques for object detection. The second section provides a basic overview of how the LiDAR works and how it is used for various applications. The third section briefly summarizes what does the point cloud data stores and how the recorded point clouds are saved in different formats. Finally, the last section provides an in-depth literature about the deep learning algorithms for object detection using the point clouds.

2.1 A Comprehensive Review of Techniques and Innovations for Object Detection

(Zamanakos, Tsochatzidis, Amanatiadis, & Pratikakis, 2021) provides a comprehensive review of various LiDAR-based object detection techniques. They have discussed the principles of LiDAR technology and explored different algorithms used for object detection, including point cloud-based and deep learning-based approaches. A review conducted by (Xiao et al., 2023) highlights the strengths and limitations of each technique and discusses their applicability in different scenarios. This article serves as a valuable resource for understanding the current state-of-the-art in LiDAR-based object detection. There exist challenges associated with LiDAR data processing (Alaba & Ball, 2022). They have provided an in-depth analysis of various algorithms used for object detection, such as voxel-based, point cloud segmentation, and deep learning methods. In the same article, they also addressed issues related to real-time processing and system integration. Their work offers valuable insights into the advancements and limitations of LiDAR-based object detection in the context of autonomous vehicles.

(D. Wu, Liang, & Chen, 2022) have focused on enhancing LiDAR-based object detection by integrating semantic segmentation. They have proposed a framework that combines LiDAR point cloud data with high-resolution images to achieve more accurate and detailed object detection results. The paper explores the benefits of incorporating semantic information in the LiDAR-based detection pipeline and demonstrates improved performance through experiments and evaluations. The findings of their work can contribute to the advancement of LiDAR-based object detection by leveraging the complementary nature

of LiDAR and visual data. (Olawoye & Gross, 2023) has presented an efficient approach for LiDAR-based object detection using deep learning techniques. In this article the use of convolutional neural networks (CNNs) and recurrent neural networks (RNNs) for processing LiDAR point cloud data and extracting object features have been investigated. The method proposed by (Fan, Yelamandala, Chen, & Huang, 2021) aims to improve both accuracy and efficiency by reducing the computational complexity of LiDAR-based detection. Experimental results demonstrate the effectiveness of the approach in achieving high detection accuracy while maintaining real-time performance.

2.2 Overview of LiDAR Technology and its Usage

LiDAR is a remote sensing technology that uses laser light to densely sample the surface of objects and terrain. At its core, LiDAR works on the principle of light time-of-flight measurement to determine the distance to a target. A laser beam is emitted from the sensor directed towards the object to be measured. A part of this laser beam is scattered while remaining is reflected back to the sensor. The time elapsed between the emitted laser pulse and its return is used to calculate the range or distance to the target based on the known speed of light. Technically, the laser beams can rapidly sample surfaces in 360 degrees, effectively capturing 3D point cloud representations of the surrounding area. In the realm of autonomous vehicles and GIS/RS mapping applications, various types of LiDAR sensors are employed to facilitate accurate perception and mapping of the environment. LiDAR sensors are pivotal for these applications due to their ability to provide high-resolution 3D information about the surroundings.

LiDAR technology is emerging as a key technology in the field of autonomous driving and enhanced driver assistance systems. With its ability to capture high-resolution 3D images of the surrounding environment, LiDAR has become an integral component in the development of advanced driver assistance systems (ADAS) and autonomous driving. According to (Yan, Mao, & Li, 2018), LiDAR-based object detection has become an essential component in the development of autonomous systems. The researchers discuss the significance of LiDAR in enabling autonomous vehicles to accurately detect and recognize objects in their surroundings, ranging from other vehicles and pedestrians to traffic signs and road markings. One of the most significant advantages of LiDAR-based object detection is its ability to provide accurate and reliable data in a range of weather and

lighting conditions. This is particularly important in applications such as autonomous driving, where the vehicle must be able to operate safely and effectively in a wide range of environmental conditions. In addition to its use in autonomous driving, LiDAR-based object detection has also found applications in areas such as robotic vision and object recognition. The technology's ability to provide accurate 3D data has made it an essential component in the development of advanced robotics systems.

LiDAR systems mounted on aerial platforms, has been instrumental in capturing high-resolution Geospatial data has become increasingly important for a wide range of applications, including environmental monitoring, archaeological research, urban planning, and natural resource management. One of the most powerful technologies for capturing high-density geospatial data is airborne LiDAR, which uses laser pulses to create highly detailed point clouds of the earth's surface. A recent study by (Petras, Petrasova, McCarter, Mitasova, & Meentemeyer, 2023) provides a comprehensive overview of point density variations in airborne LiDAR point clouds. The study highlights the use of airborne LiDAR technology for topographic and geospatial modelling research, demonstrating the capability of the technology to capture high-density point cloud data for precise digital mapping.

With the help of interpolation algorithms, digital terrain models (DTMs) and digital surface models (DSMs) are generated from the LiDAR point cloud data. DTMs depict bare earth topography below vegetation and structures, while DSMs represent the elevations of all surfaces including features. From monitoring the impact of climate change on landscapes to designing better transportation infrastructure, the applications of LiDAR point clouds are vast and varied. The technology has become a cornerstone for the production of foundational geospatial datasets and models, which are used to make informed decisions in various fields.

Looking forward, LiDAR technology is expected to become even more prevalent and integrated across multiple industries as data capturing capabilities improve and costs decline. With ongoing advancements in LiDAR technology, we can expect to see new and expanded applications emerge, supporting a diverse range of industries from transportation and infrastructure to agriculture, mining, and more. As LiDAR becomes more accessible and affordable, it has the potential to revolutionize the way we approach various tasks

and challenges, enabling us to work more efficiently and effectively than ever before.

2.3 LiDAR Point Cloud

2.3.1 Generation of LiDAR Point Cloud

LiDAR is a remote sensing technology that measures the distance between a sensor and an object by emitting laser pulses and measuring the time taken for the light to return. The data obtained from a LiDAR sensor is highly accurate, dense, and geo-referenced, making it suitable for a wide range of applications such as autonomous driving, forestry, urban planning, and infrastructure maintenance. A LiDAR point cloud is a collection of 3D points in space obtained by emitting laser pulses and measuring the time taken for the light to return, thus creating a 3D representation of the scanned area (Yastikli & Cetin, 2016).

Various methods are used to generate LiDAR point clouds, such as UAV LiDAR scanning systems (F. Wang, Zhang, Wang, & Zhang, 2023), single-line LiDAR top view scanning (Sun, Chai, Sun, & Peng, 2023), and integration of LiDAR with mobile laser scanning and unmanned aerial vehicle systems (Room & Anuar, 2022). The point cloud data obtained from LiDAR scanning can be used for various applications such as 3D city visualization, mapping urban tree canopy cover, and automatic 3D building model generation. The generation of LiDAR point clouds involves several processes, including point cloud classification, registration, and feature extraction. LiDAR point cloud classification methods are based on point sets and contexts, and they require the extraction and construction of single point or point set features (Gao & Li, 2020).

To align point clouds from different scans, algorithms such as iterative closest point (ICP) and sample consensus initial alignment (SAC-IA) are used for LiDAR point cloud registration (Xu, Pang, Bai, Wang, & Lu, 2021). Feature extraction from LiDAR point clouds involves methods such as multi-feature combination for point cloud intensity feature image and UAV optical image matching. Moreover, LiDAR point clouds provide a powerful tool for capturing and representing the real world environment in high resolution. The various methods used for LiDAR point cloud generation, registration, feature extraction, and management offer a range of possibilities for using this technology in various applications.

2.3.2 The Information stored in LiDAR Points

LiDAR technology has become increasingly popular in recent years due to its ability to generate high-resolution 3D point clouds of the real world environment. Each point within a LiDAR point cloud contains spatial information as well as other attributes that are recorded during the LiDAR scanning process. In terms of spatial information, each point contains its three-dimensional coordinates in Euclidean space, which are commonly denoted as the x, y, and z coordinate values. The x and y coordinates represent the horizontal positioning of the point, while the z coordinate corresponds to the vertical or depth positioning of the point relative to the LiDAR sensor. Together, these 3D coordinate values precisely locate where the reflecting surface was detected in relation to the LiDAR sensor's coordinate system, allowing each point to be accurately mapped within the overall 3D point cloud representation.

In addition to spatial coordinates, LiDAR points may also contain intensity data, which refers to the strength or magnitude of the reflected laser pulse returned to the sensor from that measurement location. Intensity values provide extra information about surface properties like reflectivity. Some advanced LiDAR systems may also record additional sensor-specific attributes like scattering angle, signal-to-noise ratio, or surface normal vectors. But fundamentally, each point captures where it resides spatially through X, Y, Z values as well as surface characteristics through attributes like intensity or color data.

The information contained in a LiDAR point cloud is crucial for various applications such as autonomous driving, forestry, urban planning, and infrastructure maintenance. For example, LiDAR point clouds can provide detailed information about the scanned area, including terrain, vegetation, buildings, and other objects. LiDAR point clouds contain essential information about the 3D structure and characteristics of the scanned area, including 3D coordinates, laser intensity, and detailed representations of terrain, vegetation, buildings, and other objects.

2.4 Deep Learning for Object Detection using Point Cloud

Deep learning is a sub-field of machine learning that enables computers to learn from large datasets and perform tasks that were previously accomplished only by humans. It utilizes artificial neural networks to recognize patterns and make decisions. In computer vision,

deep learning networks can be trained on large amounts of labeled image or point cloud data to automatically learn visual representations and useful image features at multiple scales of abstraction. This has led to state-of-the-art performance in many computer vision tasks like object detection, segmentation, and classification.

One of the most influential types of deep neural networks in computer vision is Convolutional Neural Networks (CNNs). They have demonstrated exceptional performance in tasks such as image recognition, object detection, and image segmentation. CNNs can learn hierarchical representations of visual data, which enable them to identify complex patterns and features within images (Yamashita, Nishio, Do, & Togashi, 2018).

2.4.1 Deep Learning and Neural Networks

In recent years, deep learning techniques have been extensively applied to object detection from LiDAR point clouds. Researchers have leveraged deep learning models, particularly neural networks, to address the challenges of 3D object detection, segmentation, and classification in the context of LiDAR point cloud data.

For instance, (Lang et al., 2019) proposed the PointPillars approach, which utilizes fast encoders and end-to-end learning for accurate and efficient object detection from LiDAR point clouds. Similarly, (Zhou & Tuzel, 2017) introduced the VoxelNet method, which utilizes voxelization to extract features from point clouds and achieve accurate 3D object detection. Moreover, (Kang et al., 2023) focused on attention mechanism-based vehicle detection from point clouds, highlighting the importance of deep learning techniques in accurately detecting and localizing objects in 3D space.

Furthermore, (B. Wu, Wan, Yue, & Keutzer, 2018) addressed real-time road-object segmentation from 3D LiDAR point clouds using convolutional neural networks with recurrent conditional random fields, showcasing the application of deep learning for semantic segmentation tasks in LiDAR data. These references demonstrate the wide-ranging application of deep learning techniques, particularly neural networks, for object detection from LiDAR point clouds. These approaches have significantly advanced the capabilities of 3D object detection, segmentation, and classification, contributing to the development of autonomous driving systems and other related fields.

2.4.2 Complex YOLO-V4

You Only Look Once (YOLO) was introduced by (Redmon, Divvala, Girshick, & Farhadi, 2016) and it stands out for its remarkable simplicity. It employs a single convolutional network that concurrently predicts multiple bounding boxes along with the class probabilities for each box. Unlike traditional methods, YOLO is trained on entire images and directly focuses on optimizing the overall performance of detection. This cohesive approach offers numerous advantages compared to conventional object detection techniques. Subsequent advancements in the YOLO series have led to the introduction of more refined versions, such as YOLOv2, YOLOv3, and YOLOv4. These iterations represent significant enhancements in terms of accuracy, speed, and functionality, marking notable progress in the field of object detection.

In this research, Complex YOLO-V4, with a training approach as outlined by MathWorks is applied. Specifically tailored for LiDAR data, this method is integral in autonomous vehicles and robotics for creating 3D environmental maps. The Complex-YOLO algorithm, a significant enhancement of the original YOLO algorithm, effectively utilizes LiDAR data and bird's-eye-view RGB maps for precise localization of 3D multi-class bounding boxes.

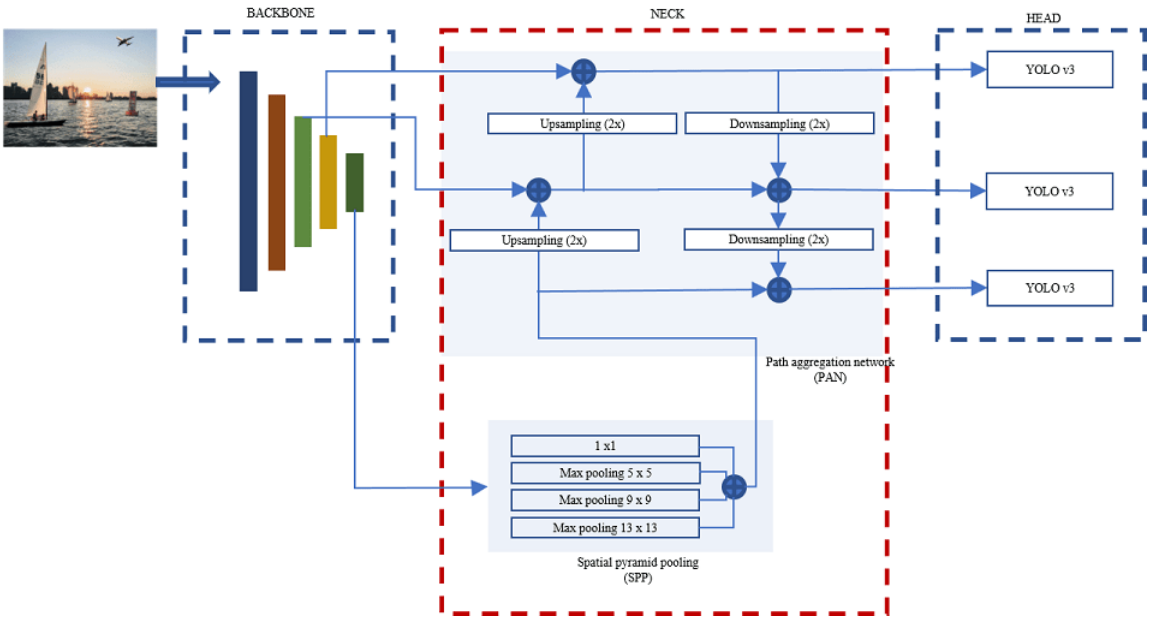


Figure 1: Complex YOLO-V4 Architecture

The YOLO-V4 architecture is a powerful tool for real-time object detection, consisting of three main components: the backbone, neck, and head. These components work together

to efficiently and accurately detect objects from input images or point clouds.

The backbone, which is typically based on CSPDarknet53, is responsible for extracting high-level features from the input data. It processes the input and captures the spatial and semantic information needed for object detection tasks. As (Bochkovskiy, Wang, & Liao, 2020) notes, the backbone is critical for enabling the network to understand the content of the input image or point cloud.

The neck component connects the backbone to the head and further refines the features extracted by the backbone. It includes modules such as Spatial Pyramid Pooling (SPP) and Path Aggregation Network (PANet), which enhance the representation of features and improve the network's ability to detect objects at different scales and aspect ratios. This component plays a crucial role in enhancing the accuracy of the network.

The head component is responsible for generating the final predictions, including bounding boxes, objectness scores, and class predictions for the detected objects. It consists of multiple convolutional layers that process the refined features from the neck component and produce the final output. As (Simon, Milz, Amende, & Gross, 2018) note, the head component is critical for accurately localizing and classifying objects in the input data.

In short, the YOLO-V4 architecture's backbone, neck, and head components work together to enable efficient and accurate real-time object detection. The backbone extracts high-level features from the input data, the neck refines these features, and the head produces the final predictions, allowing the network to detect and classify objects in real time.

2.5 Choosing Complex YOLO-V4 for Object Detection

In recent years, Convolutional Neural Networks (CNNs) have become a go-to method for object detection tasks. Among the many CNN architectures available, YOLO-V4 has emerged as a popular choice, especially for real-time object detection tasks (R. Wang et al., 2021).

One of the main reasons for its popularity is its speed and accuracy. YOLO-V4's approach to object detection sets it apart from other methods like Faster R-CNN and SSD. While these methods rely on region proposals to identify objects, YOLO-V4 frames detection

as a regression problem and directly predicts bounding boxes in a single evaluation. This leads to faster predictions and better real-time performance (Chen, Deng, Wang, Huang, & Liu, 2023).

Moreover, YOLO-V4 strikes a balance between complexity and capabilities. Although newer versions like YOLOv5 exist, YOLO-V4 has several improvements over previous versions like YOLOv3. These include self-adapting feature maps, cross-dimensional awareness and generalized anchor boxes. These features make it well-suited for complex LiDAR point clouds in real-time applications.

Apart from its technical capabilities, YOLO-V4 was also chosen due to the availability of pre-trained models from sources like MathWorks. This made model development easier and faster, thus allowing the research team to focus more on the comparative study between LiDAR sensors for autonomous driving needs.

Overall, these factors led to the selection of Complex YOLO-V4 as the optimal CNN architecture within the scope of this comparative study. However, future works could experiment with newer network designs to further improve the accuracy and performance of object detection tasks.

3 Data Collection

3.1 Introduction to LiDAR Dataset Collection

LiDAR data collection is a crucial process in the development of advanced autonomous driving systems and other applications that require high-precision 3D mapping. LiDAR sensors utilize laser pulses to measure distances and generate detailed, three-dimensional information about the shape and characteristics of surfaces. The point cloud data needs to be collected for variety of distinct scenes, so the data collection process needs to be diverse.

The LiDAR dataset collection process examined in this study was conducted in the test area of the Hochschule für Technik und Wirtschaft (HTW) Dresden. The confined relatively flat space made it well-suited for systematically testing the capabilities of different LiDAR systems under development. The enclosed campus setting also permitted safe repeat data collections under different conditions to compare results.

In this study, LiDAR scans were collected incorporating 4 different classes of vehicles: an SUV, sedan, hatchback, and micro car. Each vehicle type provides a distinct geometry and reflective properties to test the LiDAR system's object recognition capabilities. Additionally, point clouds were captured including 3 pedestrians walking alone and crossing themselves. The main objective is to compare the object detection accuracies by training a deep learning model on the stream of the recorded LiDAR measurements.

The dataset was collected using three different LiDAR sensors – Ouster, Livox, and LS LiDAR. This approach allowed for a comparative analysis of the strengths and weaknesses of each sensor, which can help fine-tune the sensor selection process in the future. The variety of scan targets adequately served the research goals and provided insight into the performance of the different LiDAR sensors. The LiDAR data collection process is a crucial process for the success of the model. The data that will be collected needs to be diverse to achieve the diversity of six distinct scenes. The process of collecting the data was carefully planned and executed to ensure that the results were accurate and reliable. The data collected will be used to train deep learning models to improve the accuracy of object detection in LiDAR systems.

3.2 Setup & Equipment Arrangements

In order to obtain accurate and reliable data, it is essential to set up the equipment in an optimal way. For this project, a movable bench was used to set up the data collection equipment to mount the sensors and other necessary components. These sensors were utilized to capture the required data and measurements, and a small camera was also installed to record the measurement as a video. The setup was designed meticulously to ensure that all the sensors were mounted perfectly, and the data was captured with utmost precision. The movable bench was chosen to allow the sensors to be placed at an optimal height and angle, which helped capture the data more efficiently. The setup is shown in Figure 2, which highlights the placement of sensors and other necessary components.

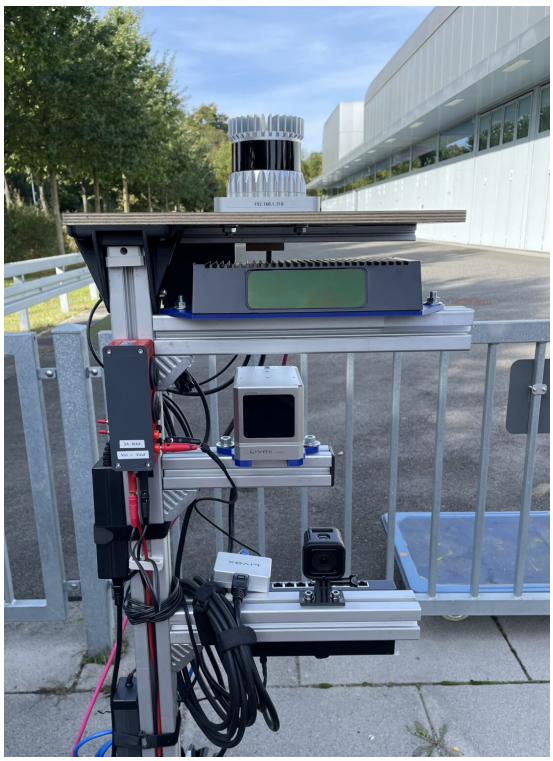


Figure 2: Sensor Equipment Bench for Data Collection

To enable synchronized recording of the three LiDAR streams, separate laptop computers were connected to each individual sensor for independent data logging. This approach ensured that the data captured by each sensor was perfectly timestamped and synchronized, which helped in analyzing and interpreting the data accurately.

3.3 Data Collection Scenarios

To test the effectiveness of LiDAR-based object detection algorithms, researchers often create a range of scenarios to simulate different real-world conditions. In this case, six different scenarios were developed to capture data from LiDAR sensors.

- **Cars Single Lane** - The first scenario involved capturing data from vehicles moving inside a single lane, which is a typical traffic flow in controlled driving conditions.
- **Cars Lane Change** - The second scenario introduced greater complexity by including lane change maneuvers performed by vehicles. Each car changed lanes after a specific distance to test the algorithm's ability to detect objects at tilted angles.
- **Cars Cut Out** - The third scenario simulated a scenario in which one car overtakes another, creating a shadow on the LiDAR point cloud. This test was designed to evaluate the algorithm's ability to overcome such obstacles and accurately detect objects in such situations.
- **Pedestrian Single Walk** - In this scenario, three pedestrians were asked to move individually in a single lane from the starting point of the LiDAR source until a certain distance and then return to the original point.
- **Single Pedestrian Walk Lane Change** - The fifth scenario combined pedestrian movement with lane changes, with individuals changing lanes after a specific distance and then returning to the starting point.
- **Multi Pedestrian Line Change** - The last scenario involved three pedestrians crossing each other in multiple lines to test the impact of shadows on the object detection model in different scenarios throughout the recording.

Each scenario was designed to provide a unique challenge to the object detection algorithm, allowing to evaluate its effectiveness in a range of real-world conditions. The routes followed by the subjects in the first scenario is marked in blue as shown in Figure 3, while the LiDAR setup was positioned on the bottom right. Similar figures for different scenarios are shown in the figures provided in Appendix A. The start time was also recorded for each measurement to cross-check with the collected LiDAR dataset.

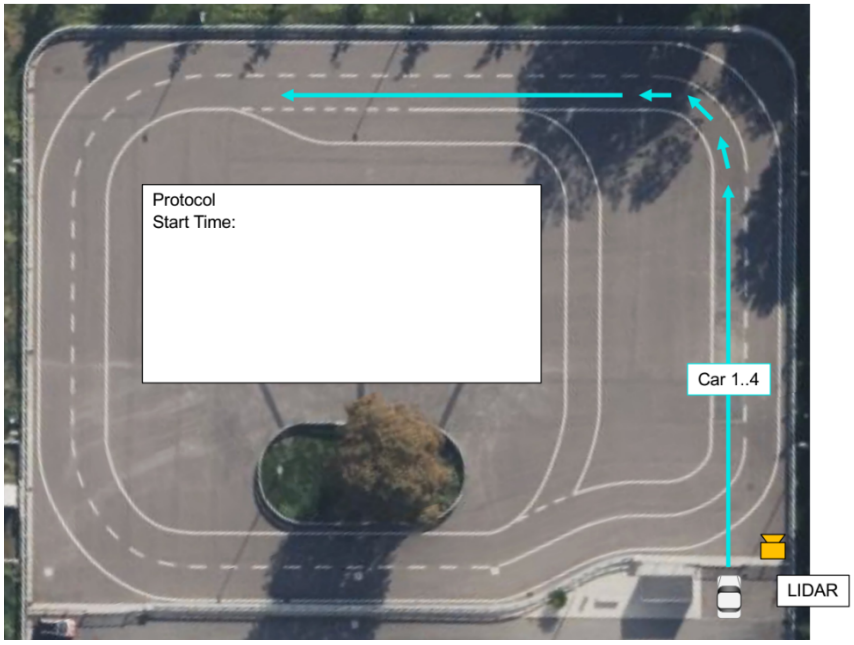


Figure 3: Data Collection Route for Scenario 1 - Cars Single Lane

Overall, these scenarios provided a valuable insight into the effectiveness of LiDAR-based object detection algorithms in various real-world conditions. By testing the algorithm's ability to detect objects at different angles, overcome shadows, and accurately detect pedestrians, we can develop more accurate and reliable object detection systems that can be used in a range of applications, from self-driving cars to robotics and industrial automation.

4 Methodology

This section discusses the methodology followed for this thesis. Figure 4 shows the flowchart methodology diagram which provides an understanding of the entire methodology workflow in detail. It provides an overview of the key steps involved in data collection, pre-processing, modeling, evaluation, and results analysis at a high abstraction level.

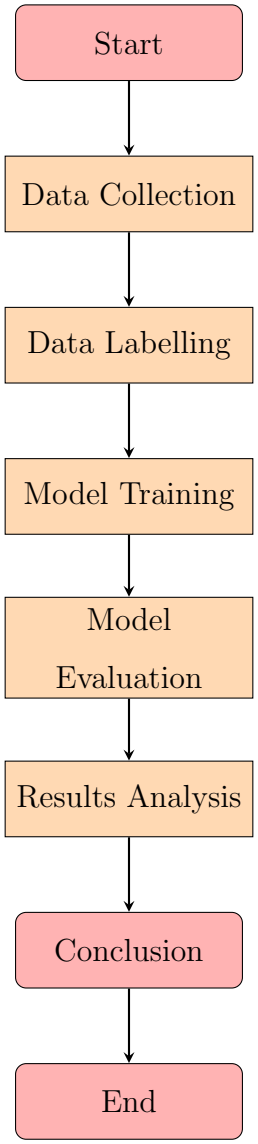


Figure 4: Simplified Object Detection Workflow Methodology

4.1 Preparing Data for Labelling

As we move towards developing a deep learning model for our test area, we have collected data for different scenarios using various LiDAR sensors. However, before we can proceed with training the model, it is imperative that we prepare the data for labelling. This step

is crucial as the output of streams of point cloud data from each LiDAR sensor is unique and different. Therefore, we need to standardize the data into a single format for training the deep learning model.

The output formats for the Ouster, LS LiDAR, and Livox sensors are ".PCAP" + ".JSON", ".PCAP", and ".LVX", respectively, as shown in Table 1. Each of these formats has its own set of characteristics and features. For instance, the Ouster format is known for its high resolution and accuracy in generating 3D images. On the other hand, the LS LiDAR format is popular for its fast data transfer rate and ease of use.

Device/Sensor	**Output Format**	**Proprietary Format Converter**	**Software Used for Conversion**
OUSTER OS1-64 REV2	*.pcap + *.json	*.csv	MATLAB
LIVOX HORIZON	*.lvx	*.las, *.csv	MATLAB
LSLIDAR VI-S128	*.pcap	*.pcd	Wireshark and LSLIDARLS View (Proprietary)

Table 1: Sensor Data Format Information

The process of preparing the data for labelling involves several steps, including data cleaning, normalization, and transformation. First, we need to convert the data from each of these formats into a unified format that can be easily labelled. This is followed by data cleaning, where we remove any irrelevant or erroneous data points that may affect the accuracy of the model. Finally, we transform the data to bring it to a specific format that can be used for training the model.

In the process of labeling point clouds for various applications, the .PCD format plays a crucial role. However, it is not always possible to generate point clouds in this format from all sensors. For example, the Ouster and Livox sensors produce proprietary formats that are not compatible with the requirements of the MATLAB Ground Truth Labeller. Thus,

we need to convert the output from these sensors to the .PCD format using MATLAB scripts.

On the other hand, the LS LiDAR sensor provides proprietary software that converts PCAP files to PCD format. However, there are limitations when dealing with large PCAP files in the LS_LIDAR VIEW application, which is the primary software provided by the organization. To tackle this issue, we use Wireshark (*Wireshark*, n.d.), a network protocol analysis tool that enables us to segment larger PCAP files into smaller ones that can be more efficiently processed by the LS_LIDAR VIEW application.

By performing these conversions, we can ensure that all the point cloud data from different sensors is in a consistent format that can be used in the MATLAB Ground Truth Labeller. Consistency is crucial for accurate labeling, which is essential in many applications such as autonomous driving and robotics. This process of converting multiple formats to a single format helps us to maintain the quality of the data and achieve better results.

During the evaluation process, the point cloud data collected from each sensor was analyzed separately to ensure accurate results. For each sensor, the data from individual scenarios, such as Cars Single Lane, Cars Lane Change, etc., was divided into training and test sets. The ratio of 70:30 was maintained to ensure that the data from a particular scenario was either used entirely for training or testing and not split between the two sets.

This approach ensured that the data was analyzed separately for each sensor and scenario, providing fine-grained results on the performance of each sensor for object detection under different conditions. By taking this approach, we were able to obtain detailed insights into the performance of each sensor, which allowed us to identify areas for improvement. The results obtained from this analysis were invaluable in helping us optimize the performance of each sensor for object detection under different conditions. By analyzing the data separately, we were able to identify patterns and trends that would not have been visible if the data had been combined.

4.2 Data Labelling using MATLAB Ground Truth Labeler

4.2.1 Introduction to Ground Truth Labeler

Data is crucial and is used to train machine learning algorithms, create computer vision models, or perform other data-driven tasks. However, before this data can be used, it needs to be labeled accurately and efficiently. This is where the Ground Truth Labeler app comes in. Once the data is prepared, the PCDs (Point Cloud Data) for each sensor can be loaded one by one into the app. The Ground Truth Labeler app is a powerful tool that allows you to label ground truth data in a video, image sequence, or LiDAR point cloud (*MATLAB Ground Truth Labeler*, n.d.). The app is designed to be user-friendly, with a simple interface that is easy to navigate. When you open the app, you can choose the type of data you want to label and the specific sensor you want to work with.

One of the key features of the app is its ability to draw bounding boxes around objects of interest. This is particularly useful when labeling data from LiDAR point clouds, where objects may not be immediately visible. You can also use the app to track objects as they move through the scene, which is useful when labeling data from streams of point clouds.

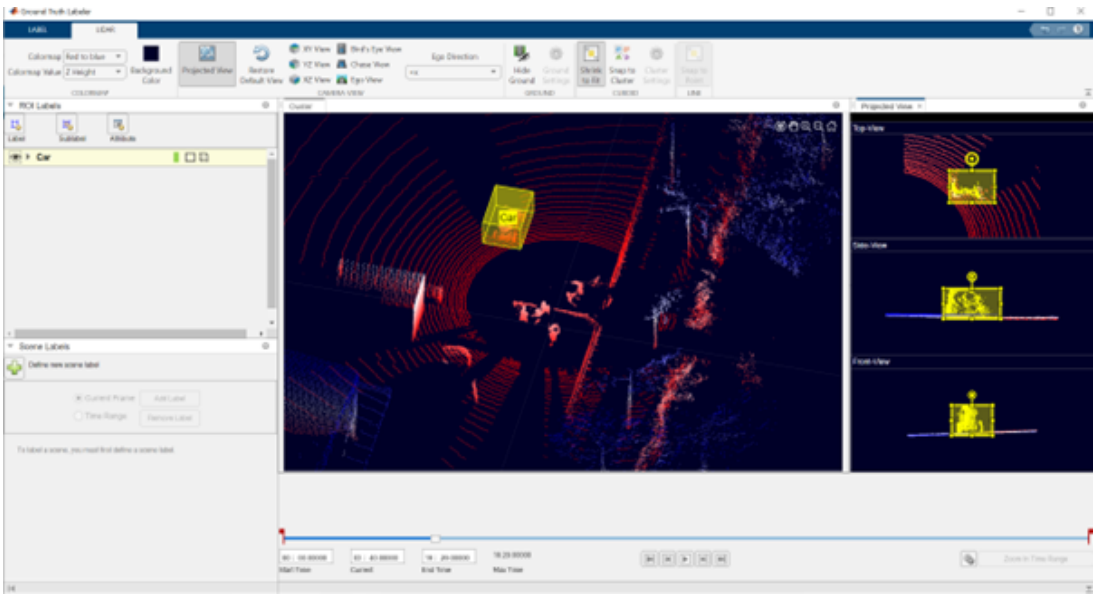


Figure 5: MATLAB Ground Truth Labeler App with Car label Bounding Box

The app also allows you to annotate your data with different types of labels, such as classification labels, attribute labels, and instance labels. This means you can label your data in a way that is most useful to your specific task. For example, if you are train-

ing a machine learning algorithm to recognize different types of vehicles, you can use classification labels to differentiate between cars, trucks, and buses as shown in Figure 5. Its user-friendly interface, powerful features, and reliability make it one of the best applications in the industry. By using this app, you can ensure that your labeled data is accurate, efficient, and of the highest quality.

4.2.2 Data Labelling Process

Once the PCD data is imported, the Ground Truth Labeler displays the stream of point clouds as a recording, which could be played frame by frame using the play/pause option on the bottom of the screen. The application also included a "Define New ROI Label" tool, which allowed the user to create a new bounding box label with a specific label name as shown in Figure 6. In this project, there were two classes of objects to be labelled: "Car" and "Pedestrian". Using the GTL, the complete stream of point cloud was labelled with these two classes. This process involved creating a bounding box around each instance of a car or pedestrian within each frame of the point cloud recording.

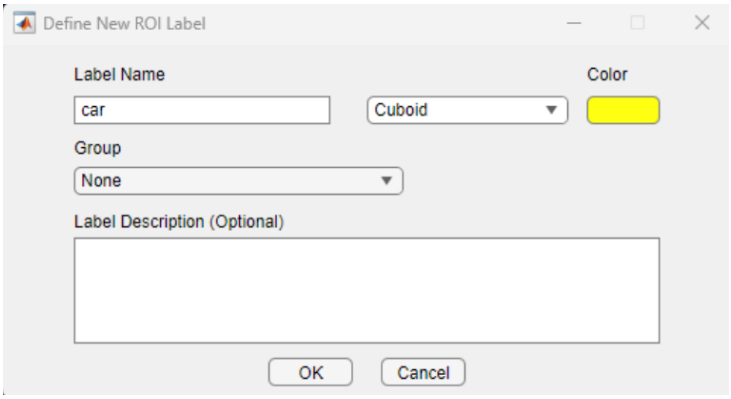


Figure 6: Define New ROI Label tool in MATLAB GTL

After the labelling process was complete for all of the captured frames for all sensors, the next step involved exporting the ground truth data as a .mat file. This file format is commonly used in machine learning applications to store labelled data, making it ready to be used for training the neural network. Figure 7 shows the block diagram of the data labelling process.

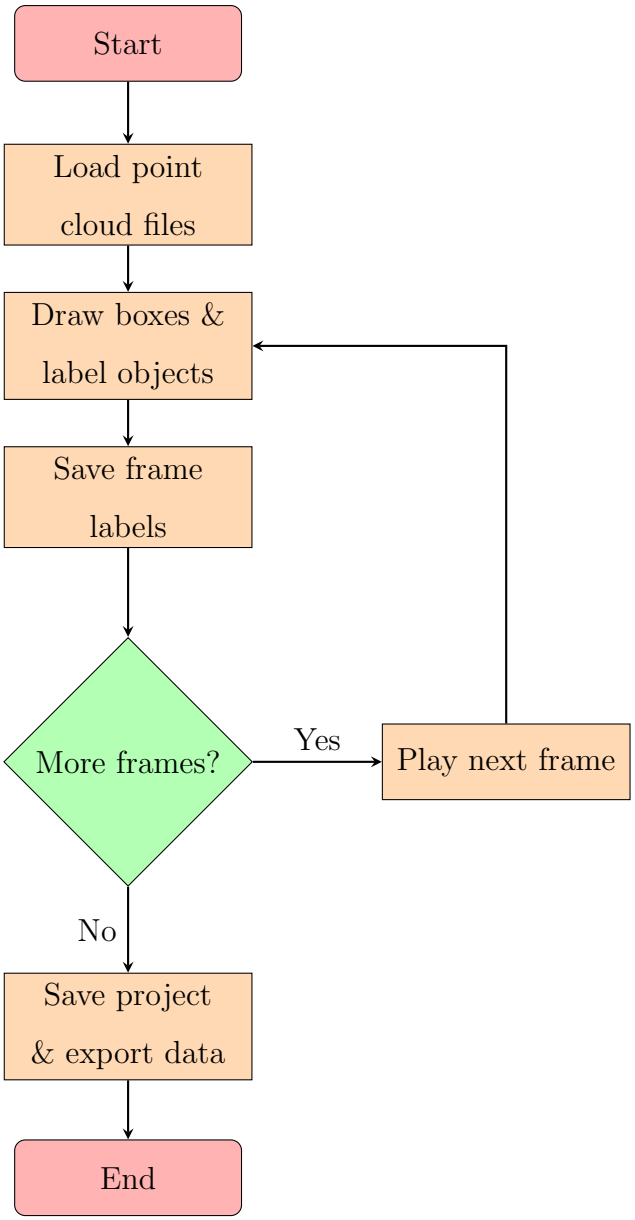


Figure 7: Data Labelling Process Flowchart

4.3 Training Complex YOLO - V4

The process of training a neural network is a crucial part of any project that involves working with a large amount of data. It is essential to choose the right methodology for training the neural network, especially when working with 3D data. In our case, since the data is in 3D format, it becomes even more important to choose the right methodology for training the neural network. There are two options that we can consider while training the neural network - creating a model from scratch or using a pretrained model.

Creating a model from scratch can offer customizability and flexibility, but it would require

extensive time to achieve desirable accuracy. Therefore, after careful consideration and research, we have decided to utilize a already available model for our project to compare the three different LiDAR sensors. We have chosen Complex YOLO-V4 model, which is a widely used model for object detection tasks. This model has proven its proficiency in processing LiDAR data, which is the type of data we are dealing with in our project. This methodology is based on the Complex YOLO-V4 model detailed in the MathWorks guide(*Complex YOLO-V4*, n.d.).

Since the working of Complex YOLO – V4 is already discussed in the Literature Review section, we will focus on the training parameters that we have used while training the model. These parameters include Max Epochs, Mini Batch Size, Learning Rate, Warmup Period, L2 Regularization, and Penalty Threshold. The parameter values are shown in the table below.

Parameter	**Value**
Max Epochs	50
Mini Batch Size	16
Learning Rate	0.001
Warmup Period	500
L2 Regularization	0.001
Penalty Threshold	0.5

Table 2: Training Parameters used to train the Complex YOLO-V4

The number of epochs is a hyperparameter that defines the number of times the learning algorithm will work through the entire dataset. We have set the maximum number of epochs to 50, which is a reasonable number for our dataset. The mini-batch size is another hyperparameter that defines the number of samples to work through before updating the model. We have set the mini-batch size to 16. The learning rate is another critical hyperparameter that determines how quickly the model learns. We have set the learning rate to 0.001, which is a reasonable starting point for our model. Moreover, the warmup period is the number of iterations to use when gradually increasing the learning rate whose value is set to 500. L2 regularization is a technique used to prevent overfitting in the model. We have set the L2 regularization parameter to 0.001, which worked out the

best for our dataset. The penalty threshold hyperparameter is used to control the false positives and false negatives in the predictions.

The model has been trained using a GPU with Parallel Computing Toolbox, which allows faster processing. The GPU we have used for this project is GeForce GTX 1180. By using a GPU, we were able to train the model much faster than using a CPU, which is essential when working with large datasets.

4.4 Evaluating Model's Accuracy

Evaluating the accuracy of a model is a crucial step in any machine learning project. It involves measuring the performance of the trained model by comparing its predictions with the actual test data. In our methodology, we evaluate the accuracy of the model in two ways, one for 2D bounding boxes and the other for 3D bounding boxes. To ensure that our model is robust and generalizes well to unseen data, we split our data into training and test sets in a 70:30 ratio. This means that 70% of the data is used for training the model, while the remaining 30% is used to evaluate the model's performance.

The 2D bounding boxes are evaluated using the Average Orientation Similarity (AOS) and the Average Precision (AP). AOS is basically a metric that measures the detector performance based on the rotated rectangles that are detected. A higher AOS value signifies a more accurate alignment of predicted orientations with the actual orientations. On the other hand, AP focuses on the precision-recall trade-off. AP measures how well a model balances the precision of its predictions at various recall levels, offering insights into the overall performance in capturing relevant objects.

The 3D bounding boxes are evaluated using Intersection over Union (IoU) metric. This parameter is also a metric which is a measure that shows how well the prediction bounding box aligns with the ground truth box. The 3D bounding boxes are a bit complex than 2D boxes as it involves height as well, which might lead to getting a lower accuracy than the 2D boxes metrics. The IoU metric is calculated for each frame so that a relationship could be find out with the distance of the object from the LiDAR sensors which is later plotted as line graphs.

To obtain the model's accuracy, we evaluated the test data using mean Intersection over Union (IoU), Average Orientation Similarity (AOS), and Average Precision (AP) scores.

For each predicted 3D bounding box, the IoU with the corresponding ground truth box was calculated using the formula:

$$IoU = \frac{\text{Area of Overlap}}{\text{Area of Union}} \tag{1}$$

The area of overlap is the intersection area between the predicted and ground truth box projected on the x-y plane. The area of union is the total area covered by the two boxes. This process was repeated for all predictions to obtain individual IoU scores. When assessing 3D data models, especially those predicting point cloud outputs, it is crucial to adapt traditional 2D intersection over union (IoU) metrics to a 3D setting. This adaptation is necessary to account for the complexities of representing volumetric data in three-dimensional space. Our approach relies on the simplifying assumption of axis-aligned bounding boxes, which streamlines the computation of intersection volumes. Effectively evaluating how well 3D models generate or classify point cloud data requires metrics like IoU that have been generalized to three dimensions, since they directly measure overlap between predicted and ground truth spatial representations. The use of axis-aligned boxes in our method facilitates calculating such 3D IoU scores.

The intersection volume, $\mathrm{V}_{\text{intersection}}$, is the product of overlaps along the x, y, and z axes:

$$V_{\text{intersection}} = x_{\text{overlap}} \times y_{\text{overlap}} \times z_{\text{overlap}} \tag{2}$$

The union volume, $\mathrm{V}_{\text{union}}$, combines the individual volumes of the bounding boxes, adjusted for the intersection volume:

$$V_{\text{union}} = V_{\text{A}} + V_{\text{B}} - V_{\text{intersection}} \tag{3}$$

Consequently, the IoU in 3D is defined as the ratio of the intersection volume to the union volume:

$$IoU = \frac{V_{\text{intersection}}}{V_{\text{union}}} \tag{4}$$

For AOS, we measure the angle (α i) between the predicted and actual orientation vectors for every detection. We then calculated AOS as the average of cosine(α i) values over all detections, providing a measure of orientation prediction accuracy. The formula for AOS is given below:

$$AOS = \frac{1}{N} \sum \cos(\alpha_i) \tag{5}$$

5 Results & Discussion

This section discusses the results of the trained model on three different sensors. The results achieved by the trained model on the three different sensors provide a comprehensive evaluation of the performance of our model. The qualitative perspective offers an in-depth understanding of how our trained model predicts the bounding boxes of cars and pedestrians in an unseen dataset. The analysis takes into consideration parameters such as the accuracy of the predictions, the precision of the bounding boxes, and the consistency of the predictions.

The qualitative perspective helps us understand how well the model is able to identify and predict the objects in the real-world scenario. It also helps us in identifying if there are any biases or limitations of the model. On the other hand, the quantitative results provide various metrics that can be used to evaluate the performance of our model and are essential to measure the effectiveness of the model in detecting objects accurately.

Moreover, the quantitative results also feature IoU (Intersection over Union) graphs. These graphs help in determining how the predicted bounding boxes change concerning the distance between the object to be detected and the sensor. The results for each sensor individually are presented individually to get a better understanding of their performance. The individual analysis helps us understand how the model performs on each sensor and how it can be improved. Additionally, a comparative analysis has been conducted to identify any differences in the performance of the three sensors. The comparative analysis helps us understand if there are any differences in the performance of the sensors and how they can be improved.

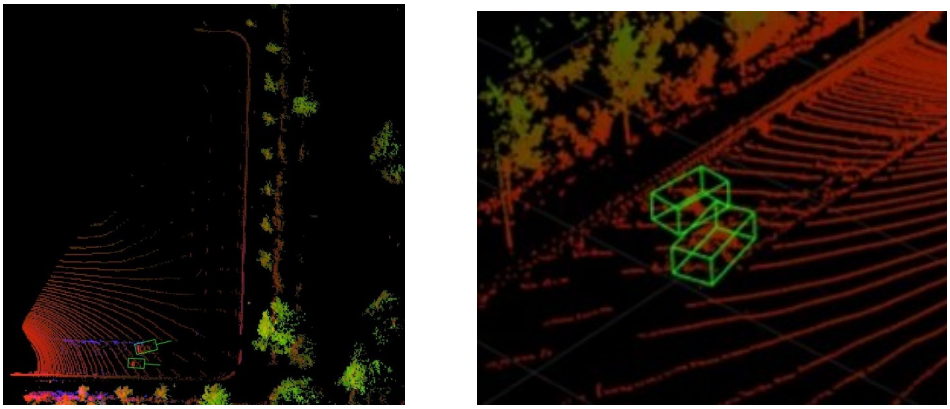
5.1 LS LiDAR

The evaluation was done based on mean IoU values, which measure the similarity between two sets of data, in this case, the ground truth data and the LS LiDAR system's detected objects. Table 3 shows the mean IoU values for all the scenes individually that the model detects object as 3D bounding boxes.

Scenario	Mean IoU Value
Cars Single Lane	0.321
Cars Lane Change	0.295
Cars Cut Out	0.351
Pedestrian Single Walk	0.237
Single Pedestrian Walk Lane Change	0.245
Multi Pedestrian Line Change	0.205

Table 3: Mean IoU Values for all Scenarios LS LiDAR

The results shows that the LS LiDAR system performed exceptionally well in detecting cars in the Cars Cut Out scene, with a mean IoU value of 0.351. This was because cars have well-defined shapes that are easier to segment, even when partially occluded. However, detecting pedestrians accurately proved to be more challenging. The Single Pedestrian Walk Lane Change and Multi Pedestrian Line Change scenes had lower mean IoU values of 0.245 and 0.205, respectively. This was due to the variability in the shapes and sizes of pedestrians, coupled with the lack of well-defined edges in point clouds.

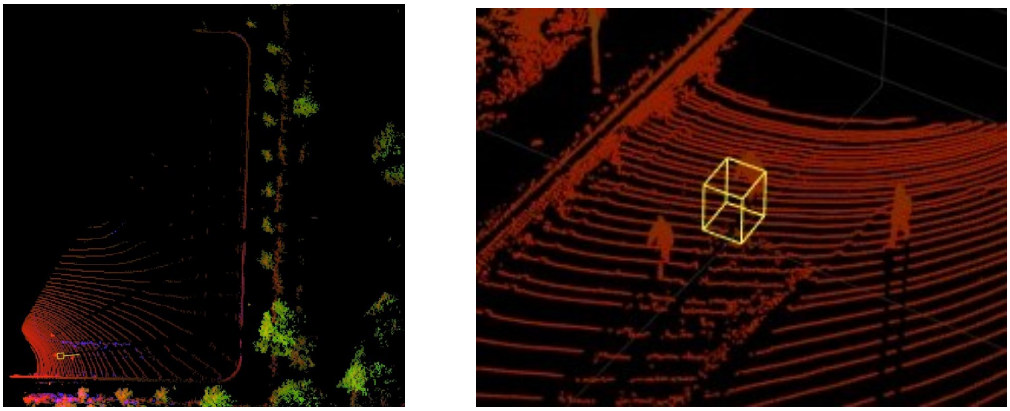


(a) 2D Bounding Box Detection (b) 3D Bounding Box Detection

Figure 8: Car detection using trained model on LS LiDAR point cloud

Figure 8(a) shows the output of the object detection algorithm on a bird's eye view image of the point cloud. It can be seen that the model correctly identifies two cars side by side. Figure 8(b) shows the 3D output for the same instant in the stream of point cloud. The cars are detected as 3D bounding boxes with height, width, and length according to the

identified object.



(a) 2D Bounding Box Detection (b) 3D Bounding Box Detection

Figure 9: Pedestrian detection using trained model on LS LiDAR point cloud

Figure 9(a) shows the predicted output of pedestrian on 2D planar view of the point cloud. It can be seen that the model does not identify the pedestrian correctly since it can be seen in Figure 9(b) that actually there are three pedestrians in a single frame but the model only predicts one; and that too not with good accuracy. With the object detection results, it can be seen that the model can predict cars with better accuracy than the pedestrians.

Table 4 shows the mean IoU values for 3D bounding boxes of the predicted class labels i.e. Car and Pedestrian. Moreover, the accuracy metrics AOS and AP are calculated based on the 2D bounding boxes.

Class Label	**Mean IoU Value**	**AOS**	**AP**
Car	0.322	0.7895	0.7903
Pedestrian	0.229	0.2315	0.2315

Table 4: Accuracy metrics for Car and Pedestrian class labels for LS LiDAR

Further analysis of the mean IoU values by class showed that cars had a higher mean IoU of 0.322 than pedestrians at 0.229. This followed the same trend as above since LS LiDAR point clouds were likely better able to capture the distinct surface geometries of cars compared to humans as also verified by the visual results.

It can also be seen that the accuracy metrics for 2D detection is comparatively higher than 3D bounding boxes detection. Since the 3D detection also involves the rotation, and height parameter of the bounding box, therefore it is more accurate accuracy metric for real world scenarios. Moreover, the 2D detection from bird's eye view perspective provides an excellent accuracy result for the Car, but detecting pedestrians remains a challenge. Further evaluation and testing on different datasets and metrics could provide valuable insights to improve its accuracy and robustness for autonomous driving needs.

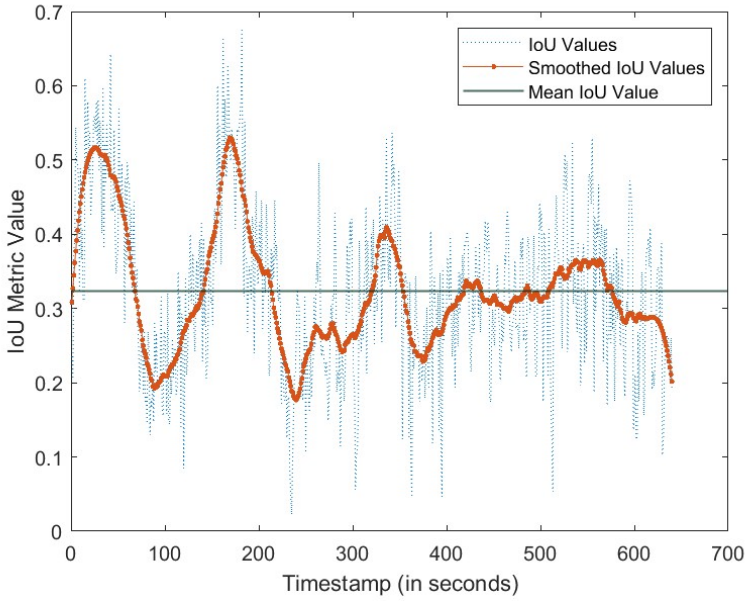


Figure 10: IoU value graph for LS LiDAR for Cars Single Lane scenario

Figure 10 shows another evaluation metric that can be used to identify how good the object detection model works. The graph has the Timestamps in seconds on the x-axis and the y-axis shows the IoU values for the detected objects at each frame. It can be seen that there are peaks and dips in the IoU graph. The peaks indicate the object location when they are close to the sensor and it can be seen that as the distance increases the IoU value decreases. The IoU value starts to rise again when the second car passes from in front of the sensor again. The blue lines at shows the actual IoU values whereas the red line shows the IoU values with Savitzky-Golay (Schafer, 2011) filter that averages the values at specific intervals.

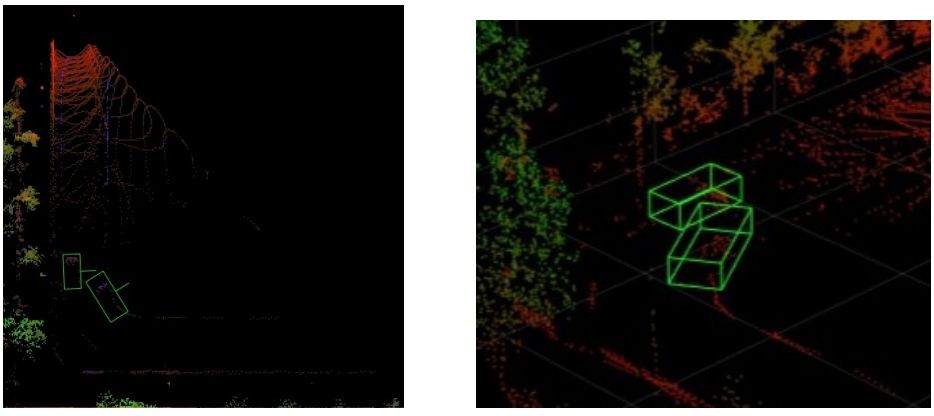
5.2 Livox

Table 5 outlines the mean IoU scores achieved by Livox in various scenes, along with its potential as a sensor technology for autonomous driving. Livox has emerged as a promising sensor technology that offers real-time, high-precision LiDAR data for perception tasks, such as object detection, tracking, and classification in autonomous driving systems.

Scenario	**Mean IoU Value**
Cars Single Lane	0.407
Cars Lane Change	0.395
Cars Cut Out	0.391
Pedestrian Single Walk	0.218
Single Pedestrian Walk Lane Change	0.298
Multi Pedestrian Line Change	0.279

Table 5: Mean IoU Values for all Scenarios Livox

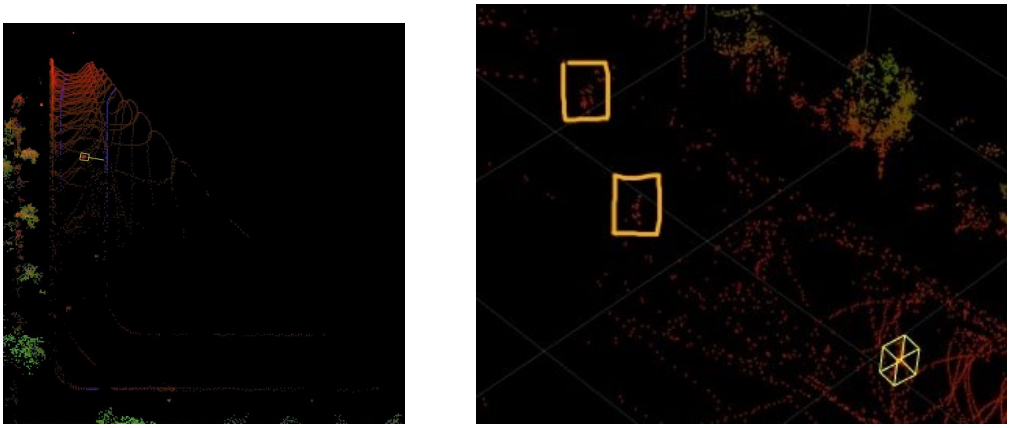
The study results indicate that Livox can accurately localize individual vehicles with a high mean IoU score of 0.407 in scenes with a single lane. However, it faces challenges in dynamic object motion scenes, such as lane changes (mean IoU of 0.395) and cutouts (mean IoU of 0.391), which require further improvement. Figure 10 shows the point cloud object detection on Livox sensor. It is worth noting that the density of the point clouds for Livox is very low compared to LS LiDAR in Figure 7. The reason lies in the working principle of both the LiDAR sensors.



(a) 2D Bounding Box Detection (b) 3D Bounding Box Detection

Figure 11: Car detection using trained model on Livox LiDAR point cloud

Figure 11(a) shows the 2D object detection bounding boxes for the cars that are at the round corner of the test area. It can be seen that the model detects cars with very good accuracy even though that they are very distant from the LiDAR source placed at the top right corner of the test area. Figure 11(b) shows the 3D scene of the same frame and the cars are detected with very good accuracy.



(a) 2D Bounding Box Detection (b) 3D Bounding Box Detection

Figure 12: Pedestrian detection using trained model on Livox LiDAR point cloud

Figure 12 shows the pedestrian detection on Livox point cloud. The pedestrians are not detected properly as can be seen in the figure. Figure 12(b) shows two yellow boxes which are manually drawn to emphasize that there are pedestrians in the frame but they are not detected and only one pedestrian is detected which is standing close to the LiDAR source. This analysis shows that the as the distance of the object from the LiDAR sensor increases, the detection capability of the model decreases. This could be one the reasons of getting low values for the average IoU as well as the AP and AOS.

Class Label	**Mean IoU Value**	**AOS**	**AP**
Car	0.397	0.4957	0.4961
Pedestrian	0.265	0.2499	0.2499

Table 6: Accuracy metrics for Car and Pedestrian class labels for Livox

Further analysis of the results shows that Livox is well-suited for vehicle perception, with a mean IoU score of 0.397 for cars as shown in Table 6. However, it still struggles with

human features, as pedestrians only managed to achieve a mean IoU of 0.265, The AOS and AP metrics of Livox for cars (0.4957, 0.4961) and pedestrians (0.2499, 0.2499), respectively, match the IoU trends, establishing decent but room for improved classification.

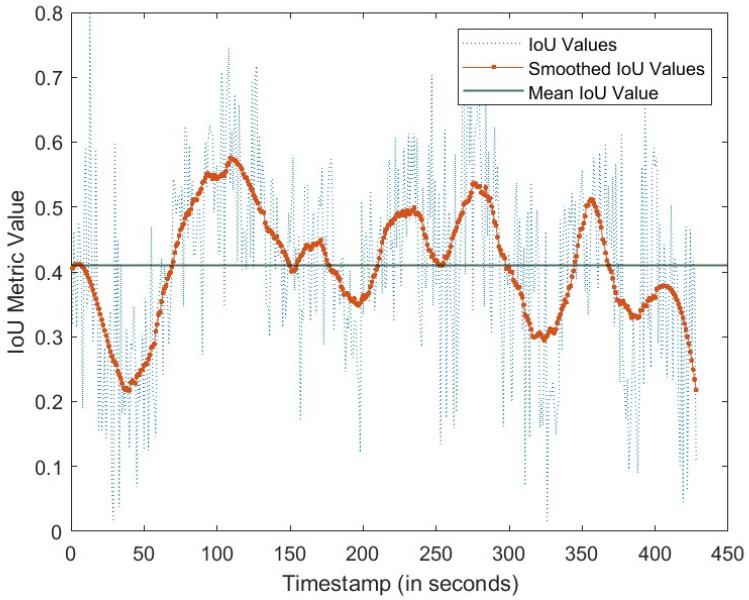


Figure 13: IoU value graph for Livox sensor for Cars Single Lane scenario

Figure 13 shows the IoU graph similar to the LS LiDAR. Here it can be seen that the peaks and dips are not that prominent which shows that the object detection at a greater distance sometimes detects the objects with good accuracy as the IoU values are higher. It should be noted that in some frames there are no objects detected even though there are objects present on the frame, so the IoU values for those frames are not included in this graph.

The study suggests that Livox's laser range may lack fine details important for boundary demarcation as also seen in qualitative results. This could be improved by incorporating visual semantics leveraging color and texture in complex scenes. Moreover, the accuracy could be improved if more training data is incorporated into the model especially for the pedestrians.

5.3 Ouster

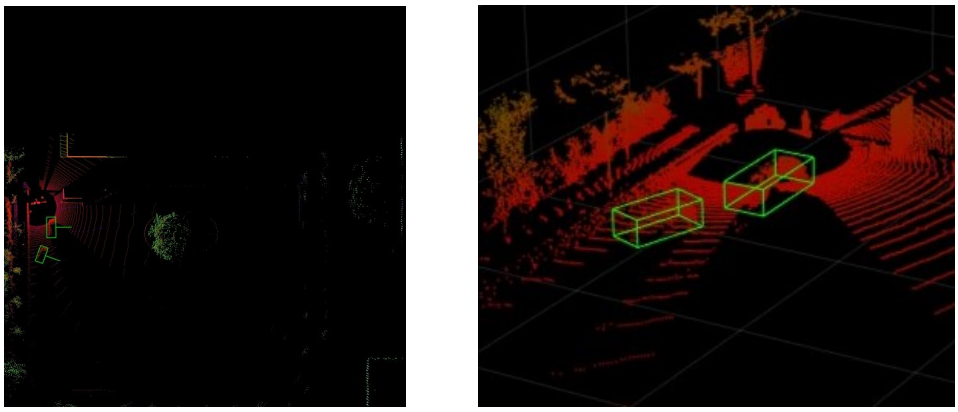
Ouster is a high-resolution 3D LiDAR sensor, and similar evaluation is also done using the same metrics as for LS LiDAR and Livox. Table 7 shows the mean IoU values for all

the scenes individually that the model detects object as 3D bounding boxes.

Scenario	**Mean IoU Value**
Cars Single Lane	0.508
Cars Lane Change	0.471
Cars Cut Out	0.433
Pedestrian Single Walk	0.316
Single Pedestrian Walk Lane Change	0.334
Multi Pedestrian Line Change	0.348

Table 7: Mean IoU Values for all Scenarios Ouster

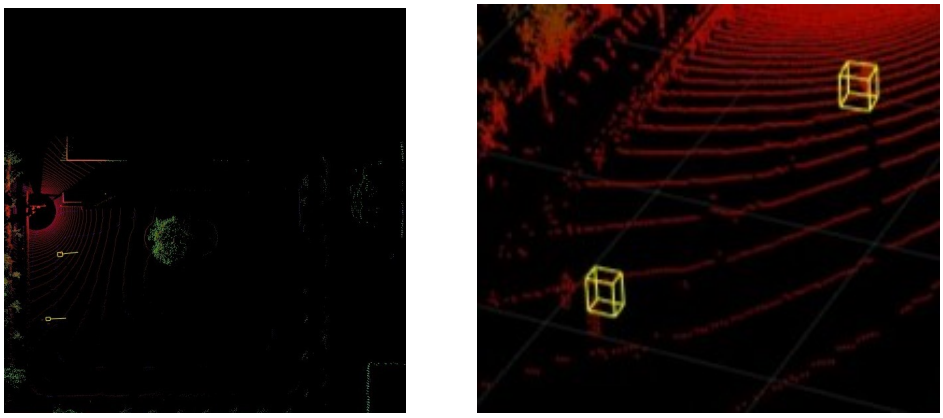
The study findings demonstrated that the Ouster had the highest mean IoU of 0.508 for identifying single cars in isolation, indicating that it has outstanding localization performance. Additionally, the model showcased its ability to track objects even when they were partially obscured in complex lane change and cutout scenes, with mean IoUs of 0.471 and 0.433, respectively



(a) 2D Bounding Box Detection (b) 3D Bounding Box Detection

Figure 14: Car detection using trained model on Ouster point cloud

Figure 14(a) shows the output of the object detection on Ouster point cloud. It can be seen that the model correctly identifies two cars side by side in this case as well. The 3D detection for the same instant is also shown in Figure 14(b).



(a) 2D Bounding Box Detection (b) 3D Bounding Box Detection

Figure 15: Pedestrian detection using trained model on Ouster point cloud

Figure 15(a) shows the pedestrians which are detected in the Ouster point cloud. It is notable that two pedestrians are detected in this point cloud. One pedestrian is closer to the sensor and second is at a far distance, and this shows that Ouster performs better in predicting pedestrians due to better localization of the point cloud in the vicinity of the sensor.

When analyzed by class as shown in Table 8, the Ouster achieved a mean IoU of 0.471 for cars, indicating that it is well-suited for detecting vehicles due to the dense point clouds that capture well-defined edges. Moreover, the study discovered that the Ouster detected pedestrians with a mean IoU of 0.332, which is higher than other sensors, suggesting that it captures nuanced details, such as human poses, even in crowded situations.

Class Label	**Mean IoU Value**	**AOS**	**AP**
Car	0.471	0.7210	0.7215
Pedestrian	0.332	0.4062	0.4063

Table 8: Accuracy metrics for Car and Pedestrian class labels for Ouster

The AOS and AP metrics for cars were 0.7210 and 0.7215, respectively, while for pedestrians, they were 0.4062 and 0.4063. The combination of these metrics with the mean IoU scores demonstrates the Ouster's strong overall classification capability.

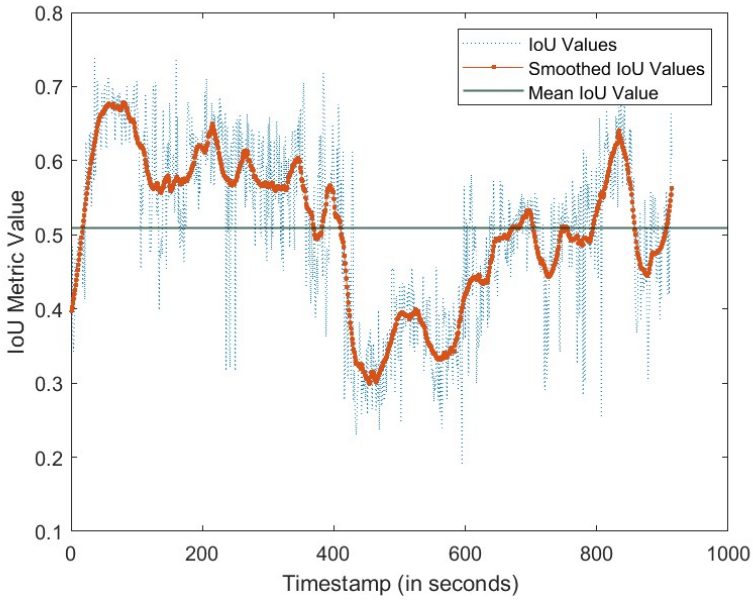


Figure 16: IoU value graph for Ouster sensor for Cars Single Lane scenario

Figure 16 shows the IoU graph for the objects detected in the first scenario - Cars Single Lane. The graph shows that the IoU values don't decrease much and it is quite different than the previous two IoU graphs. This shows that with an increase in distance from the Ouster sensor, the object detection model performs better than the other two sensors. Moreover, the mean IoU value is the highest for this sensor for the same scenario.

In summary, the Ouster's high-resolution 3D point clouds provide perceptual features that work optimally with the YOLOv4 model to achieve excellent single-object and multi-object detection performance in various autonomous driving scenarios. The results emphasize the potential of high-resolution 3D LiDAR sensors to enhance the safety and reliability of self-driving cars.

5.4 Comparative Analysis & Discussion

This section presents a comparative analysis and discussion of the results obtained from the evaluation of three LiDAR sensors, namely Ouster, LS LiDAR, and Livox, in terms of their ability to detect cars and pedestrians accurately. The mean Intersection over Union (IoU), Average Object Scale (AOS), and Average Precision (AP) scores were used as evaluation metrics. The quality of the point clouds of different sensors has a significant impact on the accuracy of object detection algorithms. Therefore, it is essential to com-

pare the performance of different LiDAR sensors to identify the best one for autonomous driving applications.

According to the results, Ouster outperformed the other two sensors in both car and pedestrian detection with the highest mean IoU and accuracy scores. Specifically, Ouster achieved a mean IoU of 0.471 for cars and 0.332 for pedestrians. In contrast, LS LiDAR and Livox showed lower mean IoU scores, indicating their difficulty in accurately capturing object boundaries and poses, especially for pedestrians. Table 9 provides the summary of accuracy scores for each sensors for both the classes.

	Mean IoU	AOS	AP
Car			
LS LiDAR	0.322	0.7895	0.7903
Livox	0.397	0.4957	0.4961
Ouster	0.471	0.7210	0.7215
Pedestrian			
LS LiDAR	0.229	0.2315	0.2315
Livox	0.265	0.2499	0.2499
Ouster	0.332	0.4062	0.4063

Table 9: Accuracy scores of each sensor for Car and Pedestrian

The superior object detection performance of Ouster can be attributed to its higher point cloud resolution of 128x128 pixels, which allowed it to extract richer perceptual features for object classification. On the other hand, LS LiDAR and Livox had lower resolutions, which negatively impacted their performance in detecting pedestrians.

In complex scenes involving dynamic objects such as lane changes and cutouts, the Ouster sensor was able to persist distinct points over partially occluded objects, maintaining accurate representations of transforming shapes. This gave its predictions a significant advantage over LS LiDAR and Livox. Ouster's superiority was further highlighted by its ability to discern even articulated human poses, achieving by far the highest mean IoU for pedestrians.

An in-depth analysis of the performance by class revealed that while LS LiDAR scored

respectably by distinguishing overall body configurations of pedestrians, Livox struggled with differentiating similar humans. Its low-density point clouds appear to conflate neighboring pedestrians. On the other hand, the Ouster sensor was uniquely capable of discerning even articulated human poses, achieving by far the highest mean IoU for pedestrians.

Despite the strong performance of Ouster, all three sensors faced challenges in detecting pedestrians accurately. The study suggests that more training data that focuses on human poses could help improve their performance. Additionally, semantic information from camera data could aid in boundary demarcation for Livox and LS LiDAR.

6 Conclusion

This study aimed to compare the performance of three LiDAR sensors, namely LS LiDAR, Livox, and Ouster, for real-time object detection using the Complex YOLO-V4 model based on deep learning. The goal was to determine the most suitable sensor for autonomous driving applications based on object detection performance. The results showed that all three sensors were capable of detecting cars and pedestrians in real-time, with Ouster demonstrating the best overall performance. For the car class, Ouster achieved the highest mean IoU of 0.471 and accuracy metrics of 0.7210 for AOS and 0.7215 for AP. While Livox and LS LiDAR also performed well for cars, Ouster exhibited improved localization capabilities. However, detecting pedestrians proved more challenging, as all three sensors had lower mean IoU scores ranging from 0.229 to 0.332. This suggests that detecting subtleties in human shapes and poses is more difficult using LiDAR point clouds alone. Nevertheless, Ouster detected pedestrians with relatively better accuracy than the other sensors. The study found that Ouster's higher spatial resolution helped extract richer object features, leading to better single-object and multi-object identification results. This highlights the importance of high point density in autonomous driving perception tasks. This comparison provides useful guidance for sensor selection in autonomous driving applications based on deep learning-based object detection performance metrics. Future work can focus on techniques leveraging multi-sensor fusion to further improve robustness.

7 Limitations & Future Scope

The research mentioned above highlights some key limitations and future scope for improving and expanding the study. One of the ways to improve the study is by evaluating more sensors, diverse datasets, and various environmental conditions, which could provide a broader understanding of the LiDAR sensor comparison. This approach would help in identifying the strengths and weaknesses of different sensors in different scenarios and selecting the most appropriate one for particular situations.

Another way to enhance the boundary definitions and improve detection capabilities is by incorporating camera data and using multi-modal sensor fusion techniques. This approach would enable the fusion of data from different sensors, which would provide a more comprehensive and accurate representation of the environment and objects in it.

In addition to the above, investigating lightweight deep learning architectures tailored for embedded applications could lead to the development of efficient and optimized models. This approach would help in reducing computational overhead and power consumption, making it suitable for real-time applications.

Furthermore, quantifying computational overhead and power consumption of different sensors and algorithms can help in selecting the most suitable option. By doing so, the researchers can identify the most efficient and cost-effective sensor or algorithm for their specific needs.

It is also suggested to deploy the best-found solutions on test vehicles and benchmark real-world detection reliability. This approach would help in validating the results obtained in the laboratory and ensuring that the solutions are reliable in real-world scenarios.

The study can be expanded to cover more complex traffic situations involving various vehicle maneuvers. This approach would provide a more realistic representation of the environment and help in identifying the limitations of the current solution.

Finally, engaging industry experts and receiving feedback on research directions and prototype capabilities can aid in further improving the study. Industry experts can provide valuable insights and feedback on the research direction and prototype capabilities, which can help in identifying the strengths and weaknesses of the study and further improving it.

References

Alaba, S. Y., & Ball, J. E. (2022). A survey on deep-learning-based lidar 3d object detection for autonomous driving. *Sensors*, *22*(24). Retrieved from https://www.mdpi.com/1424-8220/22/24/9577 doi: 10.3390/s22249577

Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). *Yolov4: Optimal speed and accuracy of object detection.* arXiv. Retrieved from https://arxiv.org/abs/2004.10934 doi: 10.48550/ARXIV.2004.10934

Chen, J., Deng, S., Wang, P., Huang, X., & Liu, Y. (2023, January). Lightweight helmet detection algorithm using an improved yolov4. *Sensors*, *23*(3), 1256. Retrieved from http://dx.doi.org/10.3390/s23031256 doi: 10.3390/s23031256

Complex yolo-v4. (n.d.). https://de.mathworks.com/help/lidar/ug/lidar-object-detection-using-complex-yolov4.html. (Accessed: 25-01-01)

Fan, Y.-C., Yelamandala, C. M., Chen, T.-W., & Huang, C.-J. (2021, May). Real-time object detection for lidar based on ls-r-yolov4 neural network. *Journal of Sensors*, *2021*, 1–11. Retrieved from http://dx.doi.org/10.1155/2021/5576262 doi: 10.1155/2021/5576262

Gao, Y., & Li, M. C. (2020, February). Airborne lidar point cloud classification based on multilevel point cluster features. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *XLII-3/W10*, 1231–1237. Retrieved from http://dx.doi.org/10.5194/isprs-archives-xlii-3-w10-1231-2020 doi: 10.5194/isprs-archives-xlii-3-w10-1231-2020

Kang, Z., Wang, S., Cui, Y., Song, Z., Kang, P., Long, Y., & Wang, Y. (2023, February). Attention mechanism based for vehicle detection from point cloud. In T. Zhang & T. Yang (Eds.), *Third international conference on computer vision and data mining (iccvdm 2022).* SPIE. Retrieved from http://dx.doi.org/10.1117/12.2660035 doi: 10.1117/12.2660035

Lang, A. H., Vora, S., Caesar, H., Zhou, L., Yang, J., & Beijbom, O. (2019). *Pointpillars: Fast encoders for object detection from point clouds.*

Matlab ground truth labeler. (n.d.). https://de.mathworks.com/help/driving/ug/get-started-with-the-ground-truth-labeler.html. (Accessed: 25-01-01)

Olawoye, U., & Gross, J. N. (2023). Uav position estimation using a lidar-based 3d object detection method. In *2023 ieee/ion position, location and navigation symposium*

(plans) (p. 46-51). doi: 10.1109/PLANS53410.2023.10139979

Petras, V., Petrasova, A., McCarter, J. B., Mitasova, H., & Meentemeyer, R. K. (2023, February). Point density variations in airborne lidar point clouds. *Sensors*, *23*(3), 1593. Retrieved from http://dx.doi.org/10.3390/s23031593 doi: 10.3390/s23031593

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). *You only look once: Unified, real-time object detection.*

Room, M. H. M., & Anuar, A. (2022, July). Integration of lidar system, mobile laser scanning (mls) and unmanned aerial vehicle system for generation of 3d building model application: A review. *IOP Conference Series: Earth and Environmental Science*, *1064*(1), 012042. Retrieved from http://dx.doi.org/10.1088/1755-1315/1064/1/012042 doi: 10.1088/1755-1315/1064/1/012042

Schafer, R. W. (2011). What is a savitzky-golay filter?. Retrieved from https://api.semanticscholar.org/CorpusID:15584363

Simon, M., Milz, S., Amende, K., & Gross, H.-M. (2018). *Complex-yolo: Real-time 3d object detection on point clouds.*

Sun, J., Chai, Q., Sun, H., & Peng, Y. (2023, March). Research and practice on the mining vehicle loading evaluation by single-line lidar top view scanning. *Journal of Physics: Conference Series*, *2470*(1), 012016. Retrieved from http://dx.doi.org/10.1088/1742-6596/2470/1/012016 doi: 10.1088/1742-6596/2470/1/012016

Wang, F., Zhang, X., Wang, W., & Zhang, K. (2023, March). Figure volume calculation based on lidar scanning technology. *Journal of Physics: Conference Series*, *2449*(1), 012044. Retrieved from http://dx.doi.org/10.1088/1742-6596/2449/1/012044 doi: 10.1088/1742-6596/2449/1/012044

Wang, R., Wang, Z., Xu, Z., Wang, C., Li, Q., Zhang, Y., & Li, H. (2021, December). A real-time object detector for autonomous vehicles based on yolov4. *Computational Intelligence and Neuroscience*, *2021*, 1–11. Retrieved from http://dx.doi.org/10.1155/2021/9218137 doi: 10.1155/2021/9218137

Wireshark. (n.d.). https://www.wireshark.org/. (Accessed: 25-01-01)

Wu, B., Wan, A., Yue, X., & Keutzer, K. (2018, May). Squeezeseg: Convolutional neural nets with recurrent crf for real-time road-object segmentation from 3d lidar point cloud. In *2018 ieee international conference on robotics and automation (icra).*

IEEE. Retrieved from http://dx.doi.org/10.1109/icra.2018.8462926 doi: 10.1109/icra.2018.8462926

Wu, D., Liang, Z., & Chen, G. (2022). Deep learning for lidar-only and lidar-fusion 3d perception: a survey. *Intelligence amp; Robotics*, *2*(2), 105–129. Retrieved from http://dx.doi.org/10.20517/ir.2021.20 doi: 10.20517/ir.2021.20

Xiao, Q., Pan, X., Lu, Y., Zhang, M., Dai, J., & Yang, M. (2023). *Exorcising "wraith": Protecting lidar-based object detector in automated driving system from appearing attacks.*

Xu, G., Pang, Y., Bai, Z., Wang, Y., & Lu, Z. (2021, April). A fast point clouds registration algorithm for laser scanners. *Applied Sciences*, *11*(8), 3426. Retrieved from http://dx.doi.org/10.3390/app11083426 doi: 10.3390/app11083426

Yamashita, R., Nishio, M., Do, R. K. G., & Togashi, K. (2018, June). Convolutional neural networks: an overview and application in radiology. *Insights into Imaging*, *9*(4), 611–629. Retrieved from http://dx.doi.org/10.1007/s13244-018-0639-9 doi: 10.1007/s13244-018-0639-9

Yan, Y., Mao, Y., & Li, B. (2018, October). Second: Sparsely embedded convolutional detection. *Sensors*, *18*(10), 3337. Retrieved from http://dx.doi.org/10.3390/s18103337 doi: 10.3390/s18103337

Yastikli, N., & Cetin, Z. (2016, June). Classification of lidar data with point based classification methods. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *XLI-B3*, 441–445. Retrieved from http://dx.doi.org/10.5194/isprs-archives-xli-b3-441-2016 doi: 10.5194/isprs-archives-xli-b3-441-2016

Zamanakos, G., Tsochatzidis, L., Amanatiadis, A., & Pratikakis, I. (2021). A comprehensive survey of lidar-based 3d object detection methods with deep learning for autonomous driving. *Computers Graphics*, *99*, 153-181. Retrieved from https://www.sciencedirect.com/science/article/pii/S0097849321001321 doi: https://doi.org/10.1016/j.cag.2021.07.003

Zhou, Y., & Tuzel, O. (2017). *Voxelnet: End-to-end learning for point cloud based 3d object detection.*

A Appendix A

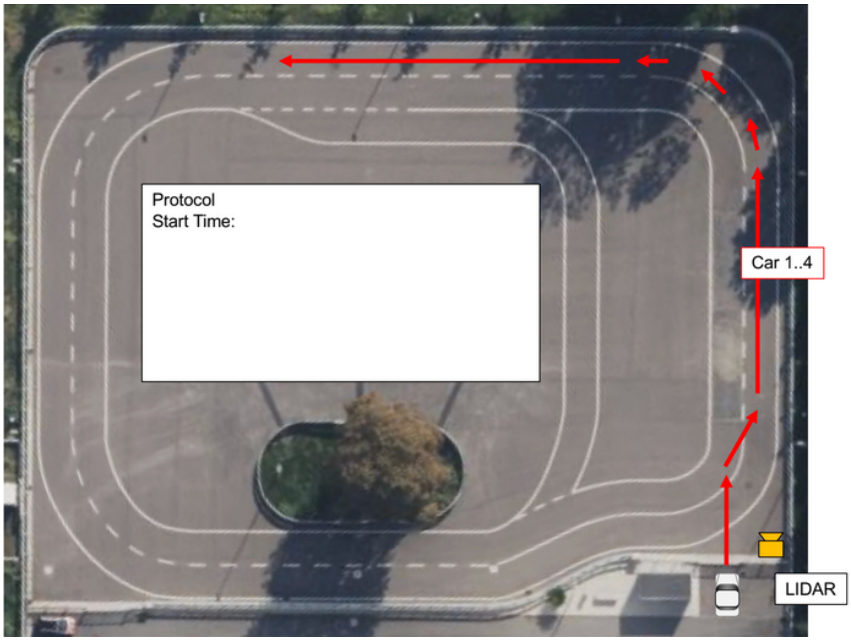


Figure A.1: Data Collection Route for Scenario 2 - Cars Lane Change

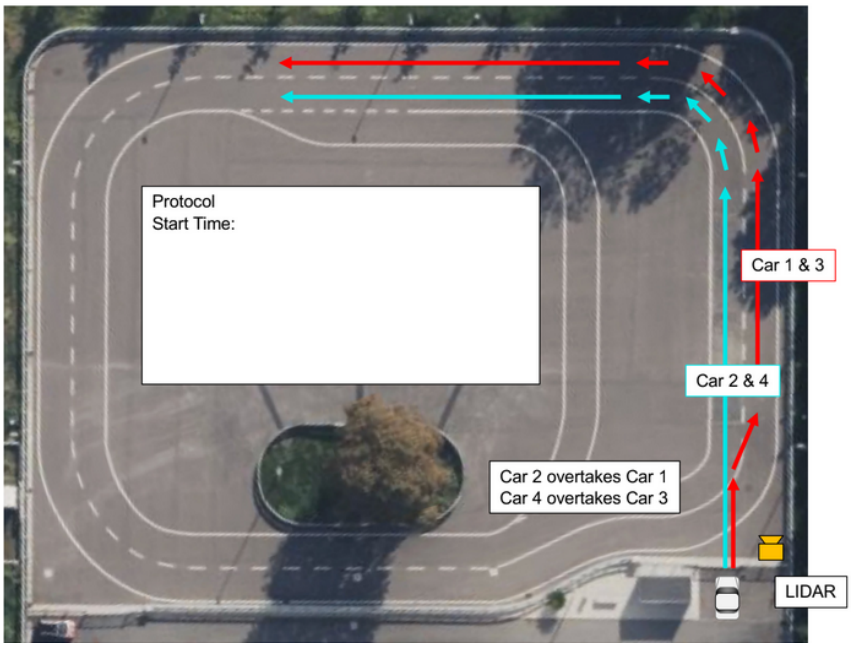


Figure A.2: Data Collection Route for Scenario 3 - Cars Cut Out

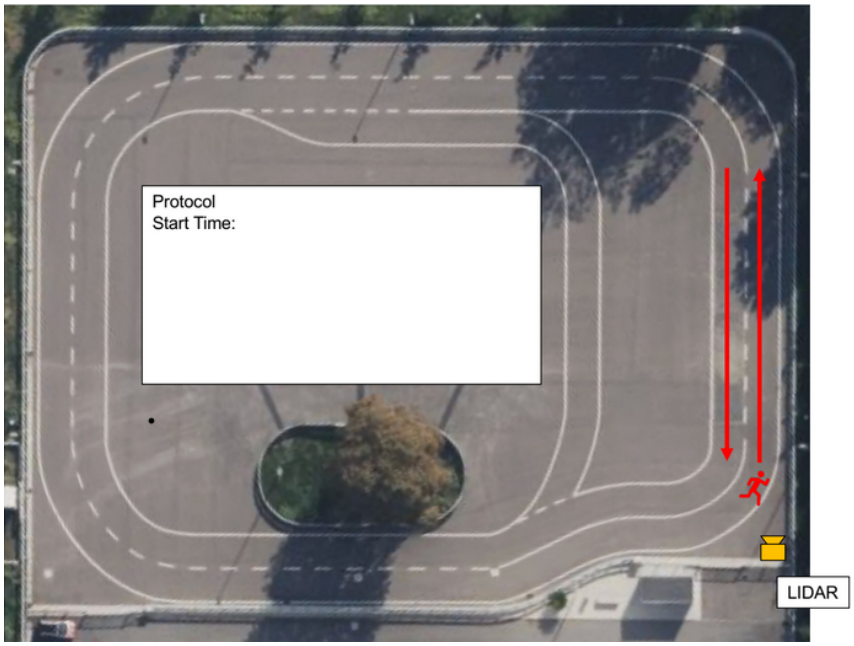


Figure A.3: Data Collection Route for Scenario 4 - Pedestrian Single Walk

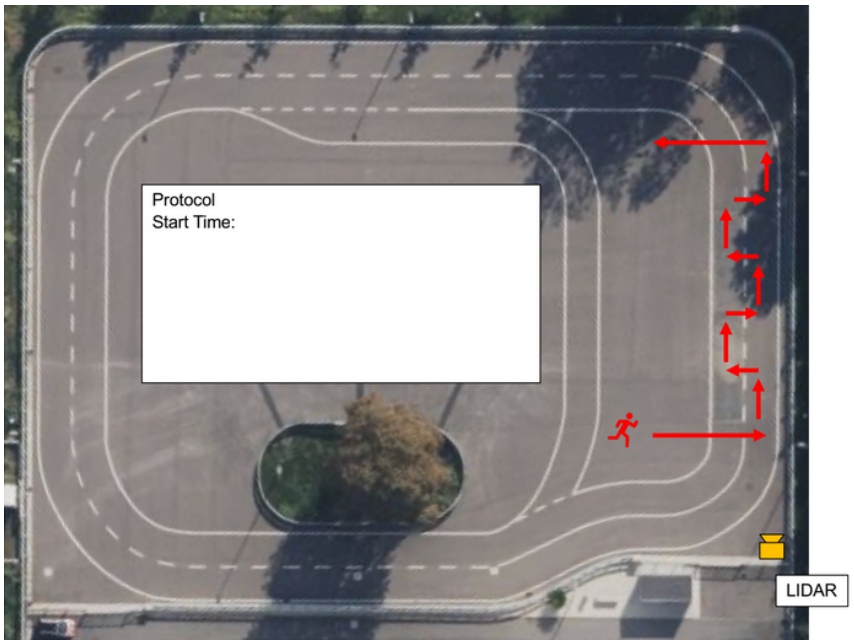


Figure A.4: Data Collection Route for Scenario 5 - Single Pedestrian Walk Lane Change

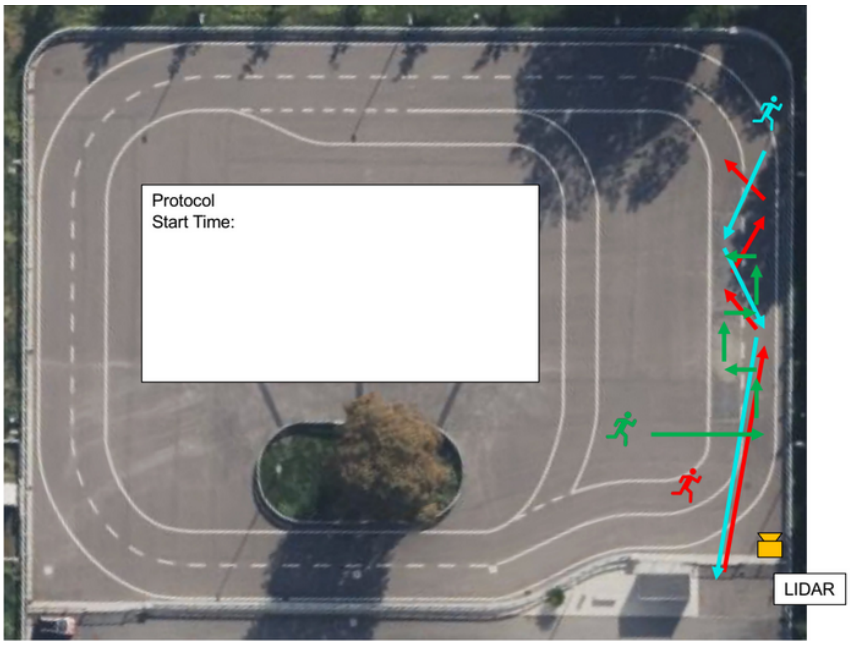


Figure A.5: Data Collection Route for Scenario 6 - Multi Pedestrian Line Change

Masters Program in **Geospatial Technologies**

Supported by:
Education and Culture
ERASMUS MUNDUS