National College of Ireland

Lidar – Infused YOLO: A Lidar infused computer vision model to improve the object detection for autonomous vehicle

MSc Research Project
Artificial Intelligence

Pragat Pravin Pagariya
Student ID: x23141221

School of Computing
National College of Ireland

Supervisor: Dr. Muslim Jameel Syed

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Pragat Pravin Pagariya
Student ID:	x23141221
Programme:	Masters in Artificial Intelligence
Year:	2024
Module:	MSc Research Project
Supervisor:	Dr. Muslim Jameel Syed
Submission Due Date:	12/08/2024
Project Title:	Lidar – Infused YOLO: A Lidar infused computer vision model to improve the object detection for autonomous vehicle
Word Count:	7367
Page Count:	24

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	
Date:	13th September 2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Lidar – Infused YOLO: A Lidar infused computer vision model to improve the object detection for autonomous vehicle

Pragat Pravin Pagariya
x23141221

Abstract

This research aims at evaluating YOLOv8, YOLOv10, and Lidar-based models in enhancing the precision of 3D objects' detection in self-driving cars. In automobile detection, YOLOv8 has impressive and fast results; however, it struggles with pedestrian, van, and other objects' detection, leading to higher false-positive and false-negative rates. YOLOv10 improves the detection accuracy of automobiles and people; however, it is not good at detecting specific objects, including trams and people sitting down. Average accuracies of Frustum PointNets (FPN) are moderate, and better performance is obtained due to the inclusion of fully connected layers, and thus they vary with the level of difficulty. The PointPillers model based on Lidar technology provides high classification accuracy, which makes it 87 percent. 15% and a Loss of 1.72. This effectively separates walkers, bikes, and autos by using 3D bounding boxes in Lidar point cloud. The combined YOLO model using Lidar data with the features of object recognition of YOLO reaches 98% accuracy on the KITTI validation dataset. It is possible to identify item placement and check the correctness of the model with the help of Bird's Eye View (BEV) photos, which contain information about the possible overlaps and misidentification. The combination of Lidar to YOLO is a new innovation that can be used in real-time 3D object detection for the purpose of self-driving cars. As it stands this technology has the capability for future enhancement and can be applied to many driving scenarios.

Keywords: YOLO, LIDAR Infused, Fusion, PointFillers, Bird Eye View, Autonomous Vehicle

1 Introduction

The future development in LIDAR technology and the existence of new sophisticated machine learning algorithms has improved the identification of items on the roadside greatly. Modern LIDAR systems have higher resolution and range, enabling it identify objects which are small as well as those which are far away. At the same time, deep learning and neural networks have improved the way in which objects could be recognized and classified with higher precision and shorter time required. Some of the methods like PointNetVinodkumar et al. (2023)and VoxelNet, have brought a drastic change in the processing of LIDAR data:

As can be seen above the autonomous vehicles have the sensors attached in the cars that help to get the obstacles detected and that tries to take an independent decision.

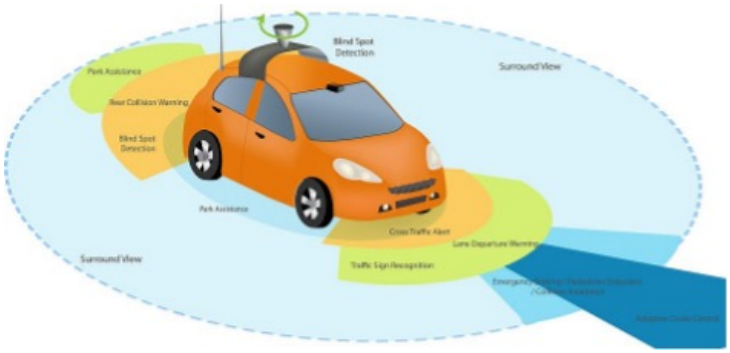

Figure 1: An example of how the autonomous vehicle work using different sensors (Source: Dubizzle)

The Artificial Intelligence (AI) base system assists in making the decisions and creating the driverless systems. These procedures work on the 3D point cloud data straight and they do not require the conversion of the data to images or grids which sometimes loses some information. The use of LIDAR for the detection of objects on the roadside in AVs Li et al. (2023) Haghighi et al. (2024) not only increases safety through the likelihood of collision detection and better positioning but also helps in the advancement of ITS. The key purpose of such systems is to establish interaction between vehicles and the surrounding infrastructure to improve traffic conditions and to reduce pollution.

1.1 Motivation

The need to improve on the functionality and safety of the autonomous vehicles (AVs) is the main reason why this research had to be conducted. Two of the most critical features of AVs and safety on the roads are object recognition on the roadside for the avoidance of accidents and smooth driving. LIDAR technology is quite accurate in terms of providing spatial measurements and proves to be quite effective in different environmental conditions. Thus, there are some problems that still remain, for example, in the recognition and analysis of complex situations. This research paper will be a pioneer in the field of autonomous driving to expand the knowledge on the subject matter. Moreover, it aims at decreasing the occurrence of traffic accidents, enhancing the traffic management, and popularizing ITS technologies. Thus, the goal of this attempt is to fulfill the objective of increasing the safety and dependability of self-driving cars and, therefore, the rate at which they are adopted and integrated into society.

1.2 Research Aim

The use of LIDAR with other sensors like cameras and radar through a technique known as sensor fusion has boosted the reliability of the object recognition systems. This is together with other developments in technology that have been witnessed. Sensor fusion takes advantage of each of the sensor types where the other has a weakness in order to build a better overall picture of the driving scene. LIDAR, for example, can give accurate distance measurements while radar is highly effective in severe weather situations. However, what really camera can do is to provide detailed information about colour and texture of a surface.

1.3 Research Problem

At present, different kinds of YOLO models such as YOLOv8, YOLOv9, and YOLOv10 are available; each model has its advantages in object detection. However, there are some cases and object classes for which none of these models can be applied. In this study, the emphasis is made on the evaluation of these YOLO models on KITTI. The primary objective of the work discussed in the paper is to reveal the effectiveness of the proposed models by the criteria of the ability to distinguish cars, pedestrians, and others. It also looks at how the LIDAR and YOLO datasets can be combined to improve real-time processing and object detection. The goal is to enhance the current detection algorithms through the fusion of multi-modal data and also by using better models thus increasing the level of performance for application such as autonomous driving and real-time traffic control.

1.4 Research Question

Research Question 1: How do different YOLO models compare in terms of performance of objects detection in the images or videos for an autonomous vehicle response system?
Research Question 2: How can a fusion of LIDAR and computer vision (CV) based deep learning models (YOLO) be optimized to improve real-time 3D object detection in autonomous driving scenarios?

1.5 Thesis Structure

In the upcoming chapters we will be discussing different research for determine the enhancements done towards the LIDAR and also computer vision. In the chapter on methodology, we will discuss in details about the different techniques. In the chapter on implementation, we will about the proposed idea of LidarYOLO which will be an infused lidar based deep learning model for enhanced autonomous vehicle working. In the results and analysis, we will compare different methods and discuss the reason behind the results obtained.

2 Literature Review

2.1 Range and depth perception

The use of BEV or RV models is applied for the task of 3D detection with the help of LiDAR. The techniques of BEV are precise, but at the same time costly since they require voxelization and 3D convolutions. Despite the fact that RV approaches are more cost effective, they lack accuracy. In relation to the three-dimensional detection, there is a RangePerception Wilson et al. (2024) that is based on RVs and it aims to be as efficient as the battery electric vehicles (BEVs) while not lagging behind in performance. The research reveals two difficulties with RV techniques: First of all, the difference of the domain between 3D global coordinates and 2D range picture coordinates and second, the vision corruption in the edge of the range images. The BEV-based method, CenterPoint, takes longer in inference speed and average precision (AP) compared to other methods. A generative model for range images from LiDAR is provided with the focus on the

data-level domain transferNakashima et al. (2023). The model uses a differentiable ray-drop effect and GANs that are based on the implicit picture representation to solve such problems as variety and fidelity, which are characteristic of generative models based on images and points. Preprocessing LiDAR data for its usage in Sim2Real semantic segmentation is advised to be done on the restore and upsample processes. It is more accurate than ray-drop effects When tests were carried out on the KITTI Raw dataset, the suggested model was found to outperform the other generative models including r-GAN, l-WGAN, vanilla GAN, and DUSty. As illustrated by the discussed point-based and image-based models, they were surpassed by the suggested model. The evaluation process of the model entails the application of SW distance, JSD, COV, and MMD. While the earlier methods of range view representations are used, recent ones prefer point- or voxel-based techniques. There are more recent techniques that are preferred in LiDAR segmentation, and autonomous driving depends on it. This research presents a novel method called RangeFormerKong et al. (2023). Its primary goal is to address the three main problems with range view models: These are shape distortion, semantic incoherence and many to one mapping. And with the strategic network design, data augmentation, and post-processing methods developed by the researchers. The findings indicate that RangeFormer outperforms some of the most advanced range view methods in terms of mIoU scores. On the LiDAR semantic and panoptic segmentation benchmarks, it outperforms voxel, point, as well as fusion-based approaches. Challenges of 3D LiDAR semantic segmentation are: Future perception-aware multi-sensor fusion (PMF), is a new approach to collaborative fusionZhuang et al. (2021). Self-driving cars and robotics are some of the target uses of this approach because they require precise calculations. The use of LiDAR point clouds in addition to the RGB pictures results in an improvement in scene understanding. Thus, due to this disparity, PMF uses a systematic approach to solve the problem of integration of these two modes. Projection of the point cloud onto the camera coordinates enables joining RGB photos with the scene's spatial depth in PMF. This setup uses a two-stream network known as TSNet. It also becomes clear that the PMF method is rather effective on different datasets, which indicates that it can easily solve rather complex real-life problems. Better results demonstrate this. From the nuScenes dataset, PMF attains a 0. An 8% increase in the mean Intersection over Union (mIoU) is achieved, thus outperforming the state of the art.

2.2 Object detection using lidar

The integration of advanced sensors like radar, high-resolution cameras, and LiDAR has been a key component of these upgrades, as it ensures precise environmental sensing. Since LiDAR can detect objects up to 120 metres away and gives a view of the horizontal field that includes 360 degrees, it is hard to overstate its importance in traffic accident prevention and route planning. To enhance autonomous vehicles' object identification skills, this research made use of the KITTI and PASCAL VOC databasesFan et al. (2021). Using the KITTI dataset, LiDAR segmentation (LS) was able to detect ROI in images. But the PASCAL VOC dataset was used to train the object detecting YOLOv4 neural network. In order to prepare the LiDAR point clouds for the YOLOv4 network, the system preprocesses them, segments the objects, and then performs 3D to 2D picture matching. The LS-R-YOLOv4 outperforms the standard YOLO in terms of accuracy (97.7%), recall (92.3%), and F-1 measure (95.2%).

The paperHekimoglu et al. (2024)introduces MonoLiG, a new framework for mon-

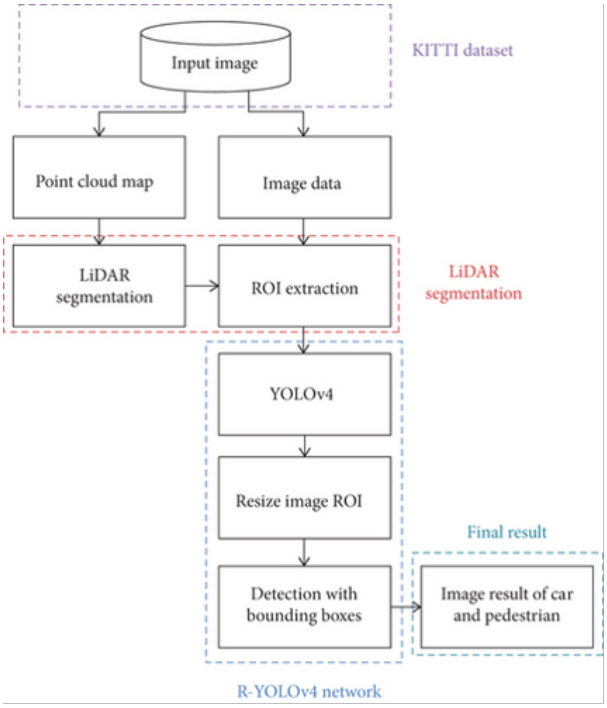

Figure 2: KITTI dataset based on LIDAR segmentation to be used in YOLOv4 model for car and pedestrian prediction [8]

ocular 3D object identification with LiDAR guidance. Applying a semi-supervised active learning technique enhances the system's performance. In order to create a model, MonoLiG takes use of all of the available data modalities. Data selection and model training are both aided by LiDAR. This is achieved at the inference step without any further overhead being added. Information is retrieved and processed into pseudo-labels from unlabeled data using a cross-modal architecture that involves a LiDAR instructor and a monocular student during training. This training method delivers the best possible performance on the KITTI 3D and bird's-eye-view (BEV) monocular object identification benchmarks, with an improvement of 2.02 points in BEV Average Precision (AP). The outcomes of the evaluation as a whole are shown in these frames. The DD3D student model and the PV-RCNN instructor model were both utilised for the active learning investigations that were conducted on an NVIDIA Tesla V100 GPU. The proposed MonoLiG paradigm outperforms the active learning and semi-supervised learning baselines by a substantial margin. Modifying it for use with any monocular detector is a breeze. In order to produce more accurate pseudo-labels, future research will combine additional modalities including radar and temporal monitoring.

2.3 LIDAR in Autonomous vehicles and roadside object detection

The creation of a sensor fusion system allowed for the integration of thermal infrared cameras with LiDAR sensorsChoi and Kim (2023). Regardless of the time of day, this technology can detect and identify objects in low-light conditions with great precision. The system uses a three-dimensional calibration target to externally calibrate the LiDAR sensor and the thermal infrared camera. This procedure guarantees that their coordinate

systems will be aligned into a single shared frame. The calibration process pays close attention to the thermal image's brightness distribution. Using a histogram, the 3D calibration objective may be determined. To convert the coordinates from the three-dimensional LiDAR world to the two-dimensional image from the thermal camera, the calibration process makes use of the translation matrix (t) and the rotation matrix (R). To achieve object detection, the YOLOv4 Convolutional Neural Network (CNN) model is employed. Its quickly detecting stuff is well-known, and it can also localise and classify things at the same time. By integrating the data from the thermal camera and the LiDAR sensor, we can leverage the unique characteristics of each device to enhance the detection accuracy. Using the three-dimensional point cloud data from the LiDAR sensor, we can verify and enhance the items seen in the thermal image. The results show that the suggested strategy improves memory and accuracy in object detection when tested in both daytime and nighttime settings. Therefore, this proves that the method works reliably in different kinds of visibility.

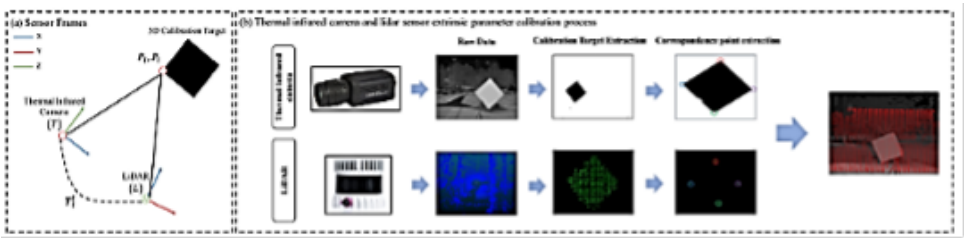

Figure 3: Sensor frames, thermal camera with LiDAR calibration for segmentation [3]

Use of LiDAR, a source of 3D spatial information Viswanath et al. (2023), allows autonomous vehicles to map and plan routes, detect obstacles, and more, particularly when travelling off-road. Areas and objects in point clouds can be identified using LiDAR semantic segmentation that is based on machine learning. With all the many textures, colours, and unclear boundaries in off-road environments, it might be challenging to carve out geometric shapes. In order to anticipate for LiDAR point clouds, intensity values are adjusted using an FCN equation. Neighbourhood prediction policies tune class intensity levels close to class mode values. Predictions of puddles, grass, trees, bushes, and people in RELLIS-3D sequences 0001 and 0002 average 47% mIoU. Research shows that the "grass" and "puddle" classes perform better than the RELLIS-3D benchmarks. A comprehensive system has been developed to digitally model roadways using LiDAR data and accurately extract geometry informationWang et al. (2023). A new semantic segmentation network is used to initiate the procedure. This network allows for the precise classification of road surfaces and infrastructure within large-scale point clouds. Next, geometric features such as road limits and centerlines can be extracted using key techniques like alpha-shape and Voronoi diagrams. Accurate information on road geometry can be obtained by utilising these features, along with a coordinate transformation matrix and the method of least squares. By utilising Dynamo and Revit, the framework seamlessly combines these components to create a digital modelling process aimed at constructing intricate three-dimensional models of road scenarios and infrastructures.By utilising the Toronto-3D and Semantic3D datasets, the techniques have been proven to be effective, yielding remarkable results. The OA ratings for both datasets are 95.3% and 95.0% respectively, while the IoU values for road surfaces are 95.7% and 97.9%.

2.4 Summary of the Papers

Summary of different research papers followed for this research

Name	Author	Dataset	Model	Result
RangePerception: Taming LiDAR Range View for Efficient and Accurate 3D Object Detection	Bai, Y., Fei, B., Liu, Y., Ma, T., Hou, Y., Shi, B., and Li, Y.	Waymo Open Dataset (WOD)	RangePerception (incorporating RAK and VRM)	Superior AP, fastest inference speed
Generative Range Imaging for Learning Scene Priors of 3D LiDAR Data	Nakashima K., Iwashita, Y. and Kurazume, R.	KITTI Raw Dataset	Implicit Representation-based GAN with Differentiable Ray-Drop	Superior fidelity, diversity in range images, improved Sim2Real segmentation
Perception-Aware Multi-Sensor Fusion for 3D LiDAR Semantic Segmentation	Zhuang, Z., Li, R., Jia, K., Wang, Q., Li, Y. and Tan, M.	SemanticKITTI, nuScenes	ResNet-34, SalsaNext	Outperforms state-of-the-art by 0.8% mIoU on nuScenes
Real-Time Object Detection for LiDAR Based on LS-R-YOLOv4 Neural Network	Fan, Y.C., Yelamandala C.M., Chen, T.W. and Huang, C.J.	KITTI, PASCAL VOC	LS-R-YOLOv4	High-accuracy real-time detection
Monocular 3D Object Detection with LiDAR Guided Semi Supervised Active Learning	Hekimoglu, A., Schmidt, M. and Marcos-Ramiro, A.	KITTI, Waymo	DD3D (student), PV-RCNN (teacher)	Improved BEV AP by 2.02 points, 17% labeling cost savings
iDet3D: Towards Efficient Interactive Object Detection for LiDAR Point Clouds	Choi, D., Cho, W., Kim, K. and Choo, J.	KITTI, nuScenes	iDet3D , IA-SSD (with iDet3D)	Superior detection accuracy , Improved mAP with user clicks
Rethinking Range View Representation for LiDAR Segmentation	Kong, L., Liu, Y., Chen, R., Ma, Y., Zhu, X., Li, Y., Hou, Y., Qiao, Y. and Liu, Z.	SemanticKITTI, nuScenes, ScribbleKITTI	RangeFormer (self-attention based)	Superior to SoTA in semantic and panoptic segmentation (mIoU improvements up to 9.8%)
A sensor fusion system with thermal infrared camera and LiDAR for autonomous vehicles and deep learning based object detection	Choi, J.D. and Kim, M.Y.	Real data were acquired	YOLOV4	Improved accuracy, day and night reliable
Off-Road LiDAR Intensity Based Semantic Segmentation	Viswanath, K., Jiang, P., PB, S. and Saripalli, S.	RELLIS-3D	-FCN	47% mIoU
Framework for Geometric Information Extraction and Digital Modeling from LiDAR Data of Road Scenarios	Wang, Y., Wang, W., Liu, J., Chen, T., Wang, S., Yu, B. and Qin, X.	Toronto-3D	Improved SCF-Net.	OA 95.3%, IoU 97.9%.

Table 1: Summary Research paper.

2.5 Research Niche

The research seeks to be at the forefront of combining YOLOv8 with sophisticated LIDAR models, such Pointfillers etc., in order to create a groundbreaking YOLO model enriched with LIDAR technology. This technique is anticipated to greatly improve the accuracy of object identification and the ability to act in real-time in autonomous driving situations.

This project aims to investigate the integration of advanced deep learning techniques with high-resolution 3D spatial data from LIDAR in order to enhance navigation and safety in autonomous cars. Unlike earlier studies that mostly concentrated on YOLOv4, this research will explore the potential synergy between both approaches, providing a state-of-the-art solution.

3 Methodology

From the above research summary table (table 1) we find that the maximum times that the dataset that has been used is KITTI dataset and hence we will be using the same for the analysis of our case s well. This will give us an apple-to-apple comparison of how our proposed model is performing.

3.1 KITTI Dataset

The KITTI dataset Geiger et al. (2013) offers a comprehensive assortment of data that allows for the development and assessment of detection algorithms under realistic conditions. This dataset is crucial for the progress of 3D object detection technologies. The collection include both left and right color images, which are necessary for visual object detection. These photographs are part of the resources it includes. These images enable three-dimensional vision, enhancing depth perception and improving the accuracy of item localization within the scene. The KITTI dataset include Velodyne point cloud

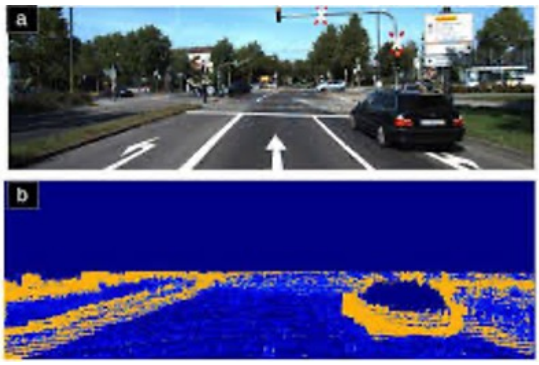

Figure 4: A sample of KITTIE dataset with point cloud

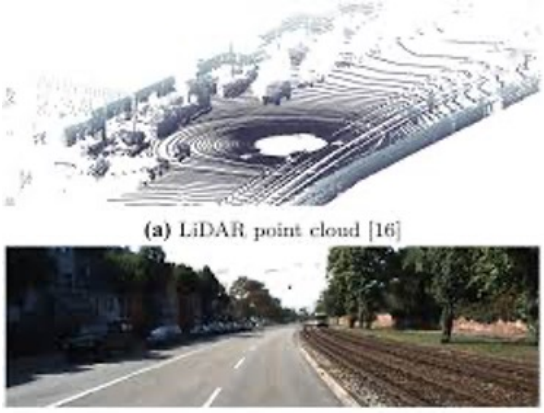

Figure 5: A sample of KITTIE dataset with point cloud

data, which captures intricate three-dimensional spatial data of the surroundings, making it a significant component of the dataset. This data provides a comprehensive view of the scene that enhances the visual information gained from color images. It is essential for accurately detecting three-dimensional objects and determining their spatial location. The dataset also provides camera calibration matrices to ensure accurate integration of two-dimensional pictures with three-dimensional point clouds. These matrices are essential for mapping the visual data captured by cameras to the spatial coordinates retrieved from the LiDAR sensor.

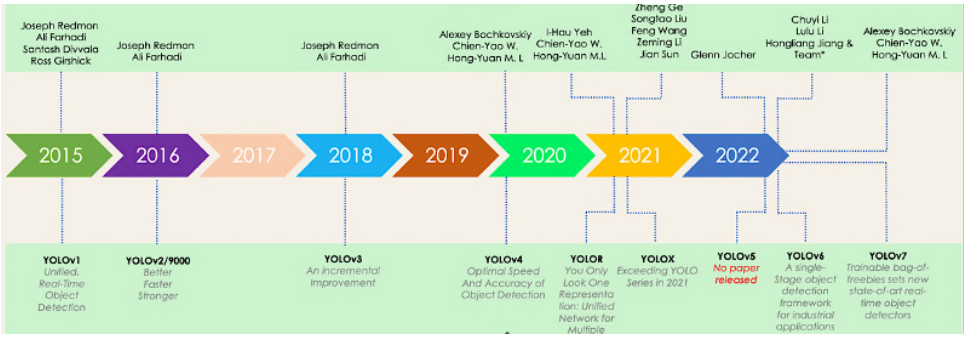

Figure 6: Advancements towards the publication about the architecture of YOLO model Jiang et al. (2022)

3.2 Computer Vision based YOLO Models

YOLO Jiang et al. (2022), an acronym for "You Only Look Once," comprises a set of convolutional neural network (CNN) models designed specifically for real-time visual object recognition. YOLO models tackle object recognition by treating it as a regression problem, directly converting image pixels into bounding box coordinates and class probabilities. This sets them apart from conventional methods that repeatedly apply the model to different parts of the image. YOLO models are highly effective for real-time applications due to their unique technique, allowing them to achieve fast detection speeds without sacrificing accuracy.

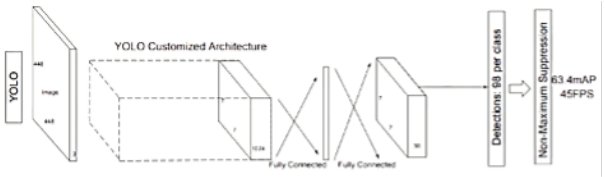

Figure 7: Structure of YOLO model Lan et al. (2018)

With each new iteration of the YOLO family, there are notable enhancements interms of performance, speed, and accuracy. From its humble beginnings, the YOLO family has undergone remarkable evolution. The latest versions, from YOLOv8 to YOLOv10, are at the forefront of this evolution, enhancing their detection skills. These versions include new approaches and improvements to the architecture.

3.3 YOLOv8

YOLOv8 Sohan et al. (2024) is an improvement over its predecessors with several architectural enhancements intended to improve the effectiveness and efficiency of object detection. This version of YOLOv8 also includes improved backbone networks as one of the major improvements over the previous version. These networks supplement a significant contribution by improving the procedure of feature abstraction from the input images leading to improved and complex feature maps. Thus, when YOLOv8 is integrated with feature pyramid networks (FPN), its object detection at multiple scales is

enhanced. Thus, using this approach, it is possible to effectively and without significant errors recognize objects of various sizes and forms. YOLOv8 incorporates different activation functions and normalization that helps in improving the model to perform well under different conditions of detection. Thus, the YOLOv8 has a high speed while

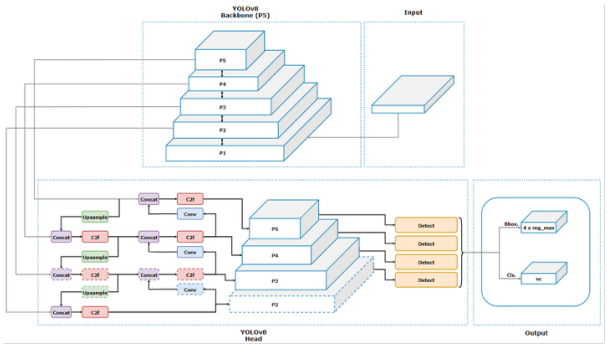

Figure 8: YOLOv8 architecture (Source: TowardsDatascience)

maintaining a good level of detail due to the presence of enhanced features. In addition, it allows distinguishing objects at various scales and incorporates the anchor boxes as the templates for accurate positioning and measurement of objects. It is even better equipped to tell between objects that look alike since it can also categorize objects and give the confidence level of the detected objects. As compared to its counterparts it has better feature representation, multi-scale detection, and overall it has better balance of speed and accuracy. It does this by using a low complexity and high speed back bone network with grouped convolutions and pointwise optimization.

3.4 YOLOv10

The last and the most developed type of the YOLO series is known as YOLOv10 Wang et al. (2024). With the help of the modern approaches in deep learning, it is possible to achieve a high level of object detection, or if you like, expand the possibilities of this technique to the limit. Self-supervised learning methods are another aspect that has been integrated into YOLOv10, and this is a major characteristic of this version. By using these tactics, the model is able to get information from large volumes of unlabelled data and therefore enhances its generality and det affliction strength in many detection strategies. In addition, YOLOv10 incorporates complex data augmentation procedures, making it possible to improve the model's performance regarding the variations in the objects' layout, illumination, and overlapping. With the help of these strategies, the model shall be able to maintain its robustness and reliability regardless of the various detection scenarios. As a key feature, the YOLOv10 model uses a new dual-head architecture that addresses the one-to-many and one to one assignments in an optimal manner. This architectural design helps in reducing the latencies and enhancing the rate of the prediction and post-processing procedures related to Non-Maximum Suppression (NMS). This architecture has a highly preferable performance in the real-time applications like self-driving cars where the real-time object detection is mandatory. YOLOv10 is less complex than the other variants of YOLO in terms of detection pipeline and implementation. This is made possible by the removal of the need for NMS post-processing.

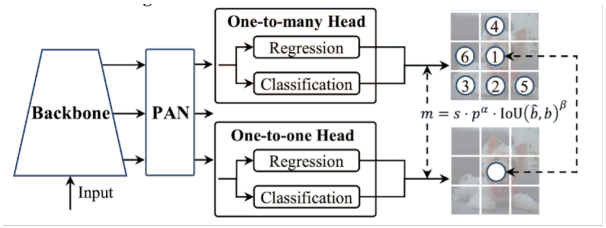

Figure 9: YOLOv10 network diagram Wang et al. (2024)

4 Design Specification

4.1 Lidar Feature Clouds : Frustrum PointNets

The Frustum PointNets Qi et al. (2018) use a complex method for the detection of 3D objects by combining the 2D object detection and 3D point cloud data. This algorithm works in a two-step process to improve the detection accuracy and speed of the algorithm. This use a 2D object identification network like YOLO or Faster R-CNN at the start of the process to detect objects in 2D images. After that, these detections are converted into 3D frustums that are triangular prismatic areas that stretch from the image plane into the three-dimensional space. This translation is realized by using the parameters of

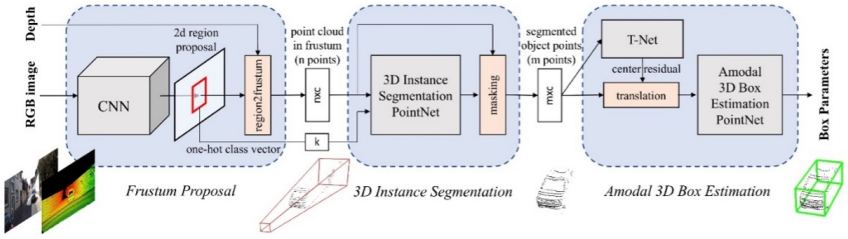

Figure 10: Pipeline for Frustrum PointNets (Source: https://stanford.edu/ rqi/frustum-pointnets/)

the camera calibration and the depth data obtained by LiDAR. Therefore, it enables direct mapping of two-dimensional detections to the three-dimensional environment. These frustums can be used as the region of interest (ROI) for the succeeding three-dimensional processing. This helps in directing the computational resources to the likely areas of finding the objects. The second step makes use of a unique neural network called PointNet that is used for processing unordered three-dimensional point clouds. PointNet produces a complete description of the point cloud contained in each frustum by taking each point and fusing its features. Thus, this skill allows for the determination of geometric properties and the generation of recommendations for three-dimensional objects. Subsequently, from the point cloud data analysis, the network generates accurate bounding boxes for three-dimensional objects along with their class labels.

The final step in the process is combining the updated 3D recommendations with the initial 2D detections. By integrating the 3D bounding boxes with the 2D object detections, we achieve accurate localization and categorization of objects in three dimensions. Frustum PointNets enhance object detection by effectively combining two-dimensional and three-dimensional data. Due to its capability to optimize computing efficiency and

improve detection accuracy, this approach is especially advantageous for applications that need accurate localization of 3D objects, such as advanced robotics and autonomous driving.

4.2 Lidar Feature Clouds :PointPillars

With the use of LiDAR point cloud data, PointPillars Lang et al. (2019) is an advanced algorithm that greatly improves the efficiency and accuracy of processing three-dimensional object detection. This program utilizes a unique approach to effectively handle the challenges that come with managing extensive 3D point clouds. Specifically, the main purpose

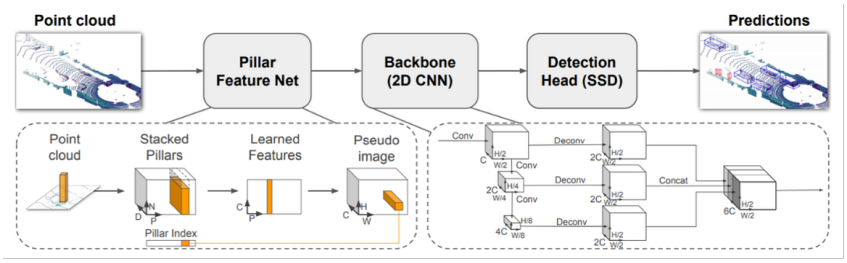

Figure 11: Pointpillars Lang et al. (2019)

of PointPillars is to take the three-dimensional point cloud and convert it into something that can be processed more easily. The first step includes partitioning the point cloud into columns, or as practitioners call it, "pillars," in which each of the vertical columns is associated with a specific part of the 3D scene. The voxelization technique adopted here makes the points in the cloud to be more organized and efficient in processing hence the improvement. Each of the pillars consist of a set of 3D points transformed into 2D points as they create an image like representation of the actual point cloud. After this, a convolutional neural network (CNN) is applied on the pseudo-image in two dimensional space. Feature extractors are used by PointPillars. These feature extractors employ two dimensional convolutions to analyze the pseudo-image and to extract the features. This technique is effective in capturing the spatial dependencies or relationships within each pillar and the properties of the items. It then transforms this information to a feature map that is useful for object detection. Finally, in the last step, the characteristics that are extracted undergo a detection head. This component can be considered as being in charge of the prediction of the classes of the objects and the bounding boxes of these objects as well. In addition, the detection head should be mainly responsible for features interpretation in order to correctly identify and locate the objects within the three-dimensional space. PointPillars has the ability to convert complex point clouds into a form that can be easily handled using two dimensional models. This leads to improvement in the effectiveness of detecting 3D objects with a significant boost in efficiency. Besides, this approach makes the calculations easier, which, in turn, improves the algorithm to recognize the objects and their categories. When it comes to applications that need to be processed and have high accuracy in as short as possible time, for example, autonomous driving or robot vision systems, then this method works as a charm.

4.3 Mean Average Precision (mAP)

It is imperative to point out that Mean Average Precision (mAP) Tychsen-Smith and Petersson (2018) is a critical measure used to evaluate the detection algorithms, such as YOLO models and PointPillars. This is especially a key consideration in the field of 3D object recognition for self-driving car and similar application domains that demands high accuracy. mAP, or mean Average Precision, gives a comprehensive measure of a model's capabilities for object detection and localization by taking into account the accuracy of object categorization as well as the precision of the bounding box predictions.

The Mean Average Precision (mAP) is given by:

$$\text{mAP} = \frac{1}{|\text{classes}|} \sum_{c \in \text{classes}} \frac{|TP_c|}{|FP_c| + |TP_c|}$$

While evaluating and comparing the efficiency of various algorithms for 3D object recognition for our project such as YOLO models and PointPillars, the mean average precision (mAP) is a very useful measure. Both YOLO models, due to their speed and

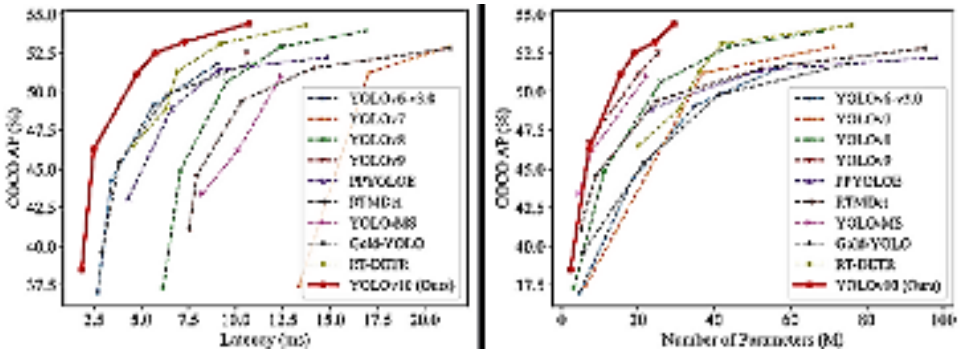

Figure 12: Comparison of different YOLO models Alif and Hussain (2024)

efficiency, and PointPillars, which has a unique approach to the processing of LiDAR point clouds, can be compared using mAP to calculate the performance of each of the models. The reader can then determine the object categorization and localization models' precision and accuracy by evaluating their mAP scores. These may help in the process of optimizing and improving the system since it is identified and well understood. This statistic ensures that the chosen model does not only perform well in specific circumstances, but also consistently perform well in many circumstances. This statistic is useful for creating dependable and stable object detection in self-driving car technologies, which is essential for their operation.

4.4 3D Bounding Box IoU

The Intersection over Union (IoU) Tychsen-Smith and Petersson (2018) is also important when calculating the precision of the bounded boxes in relation to the ground truth boxes for the detection of three dimensional objects. The 3D Bounding Box IoU evaluation gives a very accurate quantification of the efficiency of the detection algorithm in localizing an object in three-dimensional space. It achieves this using a quantitative measure of the intersection over union of predicted and ground truth 3D boxes. Intersection over Union (IoU) can be calculated by making a product of the volume of the intersection of the

bounding boxes predicted and the ground truth bounding boxes and dividing this by the volume of the union of the two bounding boxes. A perfect match is indicated by a score of 1 and this totally covers this type of ratio, which is on a scale of 0 to 1. When

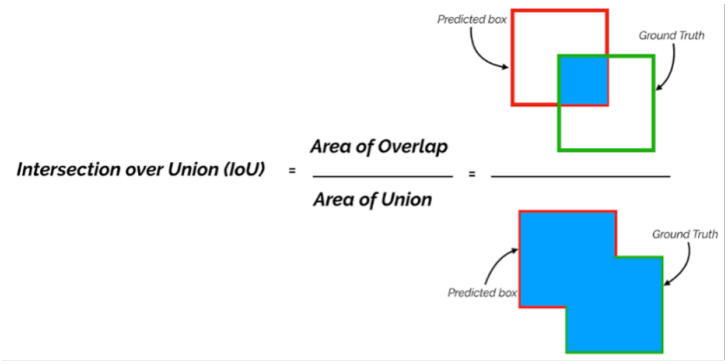

Figure 13: IOU – Intersection Over Union explanation (Source: Medium)

reviewing your models' ability to incorporate spatial factors into the object detection process, 3D Bounding Box IoU should be included as a criterion in your project. In general, this issue is critical when working with complex three-dimensional terrains and when combining information from various sources.

5 Implementation

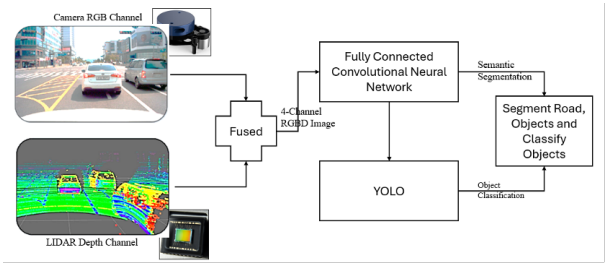

Figure 14: Proposed Lidar Infused Yolo

Step 1: Image Acquisition – From the detailed literature review we found that KITTI dataset can be used for the analysis. In the real time the data can be an infused camera which will monitor and pick the images. Following tasks is used then

a. LIDAR infusion

b. Computer Vision

Step 2: Fusion – Since two different dimensions are used, fused depths and features from the above step is taken into the considerations and feed into the deep Learning models

a. Fustrum :

Step1: 2D Object Detection – Here it uses a pre-trained 2D object detector (e.g., EfficientNet) in which it generate 2D bounding boxes on different RGB images. These kinds of 2D bounding boxes act as proposals for the subsequent 3D processing.

Step 2: Frustum Generation – The algorithm for the Fustrun then Convert the 2D bounding boxes into 3D frustums, this then define sthe search space in the point cloud. This conversion aligns the 2D proposals with the corresponding 3D points from the LIDAR data.

Step 3: PointNet Segmentation – It then uses PointNet architecture to segment the points in each frustum. This step is a binary classification of points being either object or background. Bounding Box Regression. Internally it uses pointnet++ on the segmented points to get final 3D bounding boxes. This will adjust the object's position , orientation and the size of 3D space.

b. Point Pillers:

Step 1: Pillar Generation – Another method used for comparison is to convert the 3D point clouds into a set of pseudo-images, called pillars. This involves dividing the point cloud into vertical columns (pillars) and encoding the features of points within each pillar.

Step 2: Feature Extraction – It then extracts features from the pillars using one of a convolutional neural network. Each pillar's features are encoded into a fixed-size vector.

Step 3: 2D Convolution – It then applies a standard 2D convolutions to the pseudo-images generated from the pillars. This step processes the encoded pillar features to extract higher-level spatial features.

Step 4: Detection Head – This also uses a detection head to generate 3D bounding boxes from the feature maps. The detection head predicts the object classes and refines the bounding box coordinates.

c. Infusion :

Step 1: Data Fusion : Combining Lidar and RGB Data – We plan to integrate Lidar point cloud data with RGB images to create a comprehensive input representation. This kind of fusion helps in utilizing the strengths of both data types for improved detection accuracy.

Step2: Alignment and Preprocessing – Another important step is to align the Lidar data with the corresponding RGB images to ensure synchronized input for the model. **Step 3:** YOLO Backbone : Modified Architecture – We will then propose to utilize a modified YOLOv8 architecture to process the fused Lidar and image data. The backbone is adapted to handle multi-modal inputs effectively.

Step 4: Feature Extraction – It will extract features from the combined data using the YOLO backbone, which processes both the depth information from Lidar and the visual features from the images.

Step 5: 3D Detection – 3D Bounding Box Prediction: We have to add a detection head that predicts 3D bounding boxes using the extracted features. This head refines object positions and classifies them in the 3D space.

Step 6: Multi-Modal Fusion – The detection head integrates information from both Lidar and image features to improve the accuracy of object localization and classification.

Step 3: Fitting the Deep Learning module to understand the different objects and segment

Step 4: Deep Learning Module – The features from both the module is extracted and trained n a feed forward networks and the insights are drab into the feature maps

5.1 Computer Vison based YOLO models

The basic idea of implementing YOLO is to subsample the input image into a set of cells and for every cell determine the probability of the object being in that cell and the

parameters of the cell's bounding box. The YOLO process may be dissected into many sequential steps:The YOLO process may be dissected into many sequential steps:

Step 1: Pre-processing of the input image where the picture goes through a convolutional neural network (CNN) to get the features of the picture.

Step 2: The characteristics are then passed to a sequence of fully connected layers that predict the probabilities of the different classes as well as the coordinates of the bounding box.

Step 3: The picture is divided into cells and each of them receives the task to predict several groups of bounding boxes and probabilities for classes.

Step 4: Detection network generates the set of the bounding boxes and the probabilities of classes for each cell.

Step 5: The localization is done by using following formulation Further, the bounding boxes are post processed using technique called as Non-Maximum Suppression to remove the overlapping boxes and select the box with highest probability.

The final output entails ordinary bounding boxes and class probabilities for each object there is on the picture.

5.2 Lidar Depths Algorithm 1: Fustrum PointNets

Frustum PointNets is a 3D extension to an object detection architecture called PointNet that efficiently works with LIDAR information. It works by dividing point clouds in the 3D space once it translates them into 2D frustums based on the 2D bounding box. It affords great accuracy in the localization and identification of objects within three-dimensional space, which is very imperative for the path finding of the freelance autos.

Step 1: Data Gathering and Preparation – KITTI is chosen as the input data source, which contains RGB images, point clouds collected by the LIDAR and ground truth labels for 3D object detection. First, it is necessary to prepare the LIDAR data to match the RGB images and then divide them to get training, validation, and testing datasets.

Step 2: Model Training – The trained Frustum PointNets structure with the use of the above discovered 2D bounding boxes as frustum proposals. A binary cross entropy loss should be applied to the segmentation since it results in amazing predictions when applied to the segmentation while using smooth L1 losses for the bounding box regressions.

5.3 Lidar Depths Algorithm 2: PointPillars

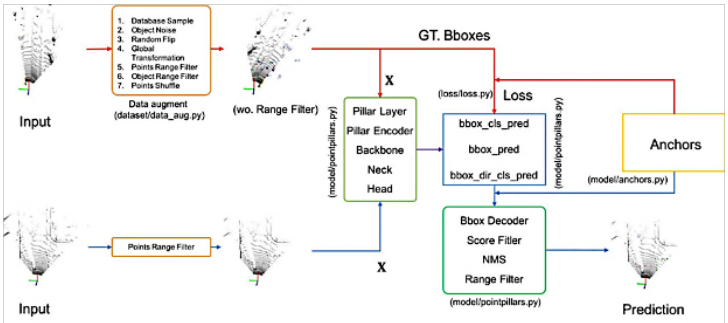

Figure 15: Pointpillers architecture [20]

The best 3D object detection on KITTI PointPillars in the house (Google AI Blog) D3Feat preprocesses raw point clouds into pseudo-image representation, which enables the use of 2D convolutional neural networks (CNNs) for 3D detection. While this is being an efficient and effective approach, especially for real-time applications like autonomous driving.

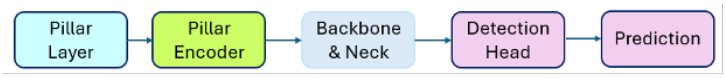

Figure 16: Pillar Network

Pillar Layer: Convert raw point clouds into pillars using the PillarLayer class, which voxelizes the point cloud data.

Pillar Encoder: Encode the pillars using a neural network to extract meaningful features. The features of each pillar are encoded into a fixed-size representation.

Backbone and Neck: Use convolutional layers to process the pseudo-images and extract higher-level features. The backbone processes the encoded pillar features through multiple layers of convolutions.

Detection Head: Predict 3D bounding boxes from the feature maps. The detection head is responsible for generating the final 3D bounding box predictions and classifying the objects.

5.4 Proposed Lidar-Infused Yolo model

The proposed YOLO- LIDAR hybrid comprises the incorporation of Lidar data into the YOLO object detection model to boost detection and localization. This is a combination of Lidar that provides detailed depth information and YOLO, which is a powerful object detection model.

Stage 1: Lidar to Camera Transformation – To convert a point from the Lidar to the camera image space, four transformations are performed:Stage 1: Lidar to Camera Transformation Summary – To convert a point from the Lidar to the camera image space, four transformations are performed:

Sub Step 1: Lidar to Camera 0: The Lidar n-point is first passed through a rigid transformation which is a rotation followed by translation to get the Lidar point in camera 0 frame of reference.

Sub Step 2: Camera 0 to Camera 2: As in the previous step, one more rigid transformation is made in order to translate the point from Camera 0 to Camera 2 (or to any other chosen camera).

Sub Step 3: Rectifying Transformation: Perform a rectification on the obtained stereo images, in other words warp them in such a way that they are rectified.

Sub Step 4: Camera Projection Transformation: After that, project the 3D point into 2D image space in order to obtain the (u, v) values. These can be chained into a single transformation matrix T_cam2velo to move Lidar points from the point cloud coordinate space to camera space using homogeneous transformation.

Sub Step 5: Transformation Matrices: Here we require Lidar to Camera Reference, then Rigid Body Transform from Camera 0 to Camera 2, then Camera 2 to the Rectified Camera 2, then from Rectified Camera 2 to 2D Camera 2 (u, v, z) coordinate system.

Stage 2: The final position of the camera coordinates in terms of (u, v, z) is given in terms of rectification and projection transforms along with division by depth (z).

Stage 3: Data Preparation: It involves;

Sub Step 1: Dataset: This work will use the KITTI dataset that is composed of RGB images, point clouds from Lidars, and ground truth annotations for the 3D detection of objects.

Sub Step 2: Data Preprocessing: L general Inputs vs Multi-Modal Inputs: Segment the Lidar data same as the RGB images to have the desired synchronization of multi-modal inputs. Organization: Split the data into training and validation, as well as test sets that can be employed to work on the model training and, accordingly, evaluate the model's performance.

Stage 4: Evaluation Metrics – Mean Average Precision (mAP): Calculates the detection accuracy in terms of the mean of the precision scores for various values of recall. 3D Bounding Box IoU: Checks the intersection over union (IoU) of the predicted 3D bounding boxes against the ground truth ones.

Stage 5: Detection Categories are overall, pedestrian, cyclers, and cars categories are chosen as the performance metric.

6 Evaluation

The purpose of this section is to provide a comprehensive analysis of the results and main findings of the study as well as the implications of these finding both from academic and practitioner perspective are presented. Only the most relevant results that support your research question and objectives shall be presented. Provide an in-depth and rigorous analysis of the results. Statistical tools should be used to critically evaluate and assess the experimental research outputs and levels of significance.

Use visual aids such as graphs, charts, plots and so on to show the results.

6.1 Experimental Scenario – YOLO models

YOLOv8: YOLOv8 produce a good detection of cars as depicted by the high true positive rate achieved. However, it has problems with exact detection of pedestrians, vans and the other classes, thus, indicating a greater amount of false positives and false negatives for these classes. The detection of miscellaneous objects and trams is generally low and this belongs to the weakness of this model.

Bounding Boxes and Confidence Scores: In regard to car detection, YOLOv8 is quite successful and prevalent in numerous situations, spiriting boxes around several cars. Each detection is followed by the confidence score which shows the opinion of the model on the detected item. This is true to the fact that some detections give results that are nearly a value of 1. The maxProb attribute of the output is 0, which means a very high probability that the detected object is a car. Lower scores in other cases are indicative of lower confidence levels in the model's classifications.

YOLOv10: YOLOv10 reveal further enhancement in car, true positives in addition to fewer false positives compared to YOLOv8. It demonstrates better results in detecting pedestrians but still has a problem with false positives and false negatives. Nevertheless, YOLOv10 reveals a higher mAP on vans, trucks, and cyclists compared to YOLOv30 and simultaneously, it demonstrates a worse performance when detecting miscellaneous objects, trams, and persons sitting.

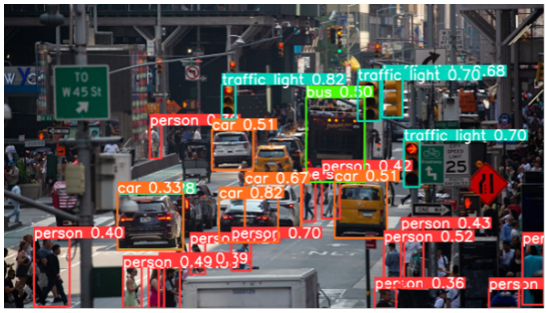

Figure 17: Yolov8 detection

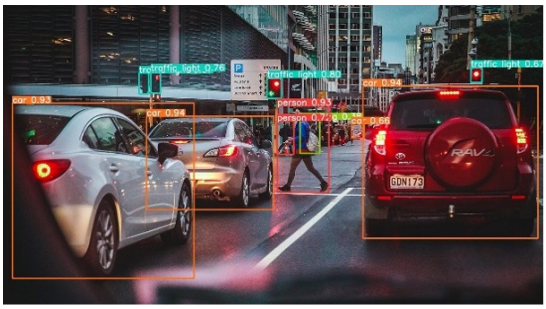

Figure 18: Yolov8 detection

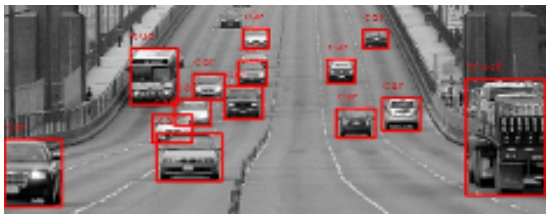

Figure 19: Yolov10 detection

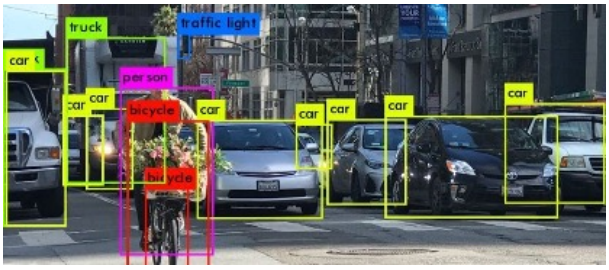

Figure 20: Yolov10 detection

6.2 Experimental Scenario – Fustrum PointNets

Baseline Model (FPN): The original Frustum PointNets model achieved an average precision of 36.34% on the Easy dataset, 31.54% on the Moderate dataset, and 29.6% on the Hard dataset.

Improved Model with Fully Connected Layers (FPN+FC): The addition of fully connected layers resulted in an average precision of 32.12% on the Easy dataset, 29.02% on the Moderate dataset, and 24.2% on the Hard dataset.

Classification accuracy: 87.15
Loss : 1.72

Average Precision Percentage

Method	FPN*	FPN+FC*
Easy	36.34	**32.12**
Mod.	31.54	**29.02**
Hard	29.6	**24.2**

Figure 21: Fustrum performance

Visualization: This image helps the initial draft of the drawing interface for assessing the multitask learning and the multiformity data fusion. The visualization itself presents ground truth annotations and model's predictions in parallel which helps identifying differences and comparing the level of accuracy. The feature for supporting multiple backends, e.g., TensorBoard, guarantees the control over the training status like loss rates or learning rate.

6.3 Experimental Scenario – PointPillers

Mean Average Precision (mAP), metric quantifies the accuracy of detecting the object in a scene by averaging different precision levels over a range in recall. Calculates the average IoU value with the help of the 3D bounding boxes of prediction and the actual values. The model is evaluated according to the following categories in the case of detection: in general, walkway, bike, and automobile. Accuracy and Loss is determined to be equal to 87 percent in classification. 15% with loss 1.72

Metric	Overall	Pedestrian	Cyclist	Car
3D-BBox	73.3259	51.4642	81.8677	86.6456

Figure 22: Mean Average Precision(mAP) on KITTI validation Set

Detection Visualization: the LIDAR point cloud visualization shows detected objects with bounding boxes. Different colors represent different classes.

- o Pedestrian: Red,
- o Cyclist: Green,
- o Car: Blue.

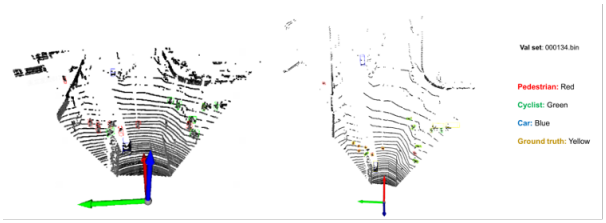

Figure 23: : Matching the pointcloud with the CV based detection

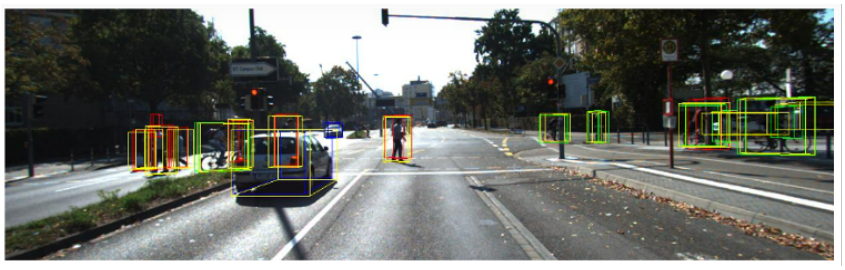

Figure 24: : Matching the pointcloud with the CV based detection

6.4 Lidar Infused YOLO

This is a sample of experiment result where car, pedestrians are detected by 3d bounding box with overall 98% accuracy.

This is a BEV image, which displays an overhead or a top view of structures making it possible to note where everything is in relation to the other. This perspective is useful for

Metric	Overall	Pedestrian	Cyclist	Car
3D-BBox @0.70, 0.70, 0.70				
bbox AP	98.1839	89.7606	88.7837	97.70
bev AP	89.6905	87.4570	85.4865	88.73
3d AP	87.4561	76.7569	74.1302	87.34
aos AP	97.70	88.73	87.34	
3D-BBox @0.70, 0.50, 0.50				
bbox AP	98.1839	89.7606	88.7837	97.70
bev AP	98.4400	90.1218	89.6270	88.73
3d AP	98.3329	90.0209	89.4035	87.34
aos AP	97.70	88.73	87.34	

Figure 25: Result on KITTI validation dataset

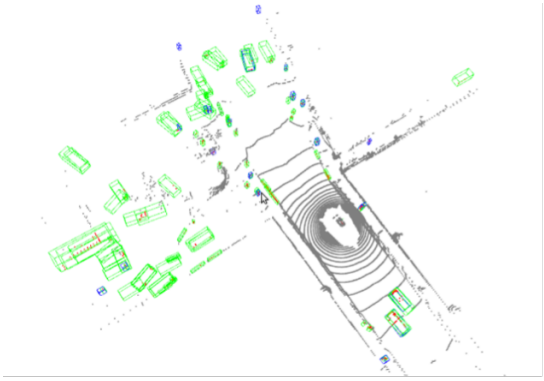

Figure 26: Bird eye view (BEV) representation of results on another example.

Figure 27: Bird eye view (BEV) representation of results on another example.

determining the placement of certain items such as cars, humans, or bicycles in relation to the surrounding world. Visualization of detection results in the BEV image: it makes it possible to do direct checks on how well and how frequently the predictions model is operational. Since the objects are clearly divided in the top-down view, if there is any conflict regarding how they may superimpose on each other or where there could be confusion with detection, then it is easily seen, making it easy to fine tune my model to perfection.

6.5 Discussion

YOLO with Lidar yields a very high performance in 3D object detection for scenarios with multi-sensor data. This is an efficient and accurate approach of incorporating Lidar into YOLO's object detection effectiveness. As a result, the future studies may entail fine-tuning and validation with different data structure to provide enhanced generality of the model. Thus, in the BEV image, it is possible to judge the accuracy and stability of the model's predictions. The top-down view enables the objects to be well-separated which aids in the determination of areas which have been confused in one model thus warranting

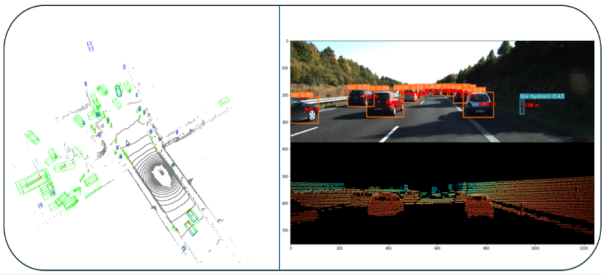

Figure 28: Object detection in BEV representation

some adjustments. As per the above result, it can be concluded that the integrating of Lidar data with YOLO as a novel technique of real-time 3D object detection for self-driving cars can be a fruitful path.

7 Conclusion and Future Work

As far as the results related to the computer vision solutions are concerned, it can be pointed out that YOLOv8 is well-performing in detecting autos, as evident through the disparity of true positive detection's obtained overall. Nevertheless, it still does not work at times distinguishing trams and people who are sat in a seated position. So, in general, FPNs turned out to be sufficiently accurate on average, but the presence of completely connected layers yielded a diversity of results. This deteriorated the already unenviable loss of 1. 72, this in regards to a classification accuracy of 87. Thus, PointPillers achieved 15% as desired, which allowed them to meet their goal. By employing three-dimensional boundary boxes encoded in red, green and blue it was able to distinguish human beings; bicycles and automobiles. With Lidar data incorporated into the YOLO model, the techniques were enhanced and successively the YOLO model being validated on the KITTI validation dataset with an 98% accuracy it was able to out-compete other techniques. The photographic method using Bird's Eye View images proved efficient in auditing the placement of the items and the reliability of sensing. Described integrated model shows high effectiveness of the real-time three-dimensional object detection in autonomous driving and there is scope for improvement to work on the variety of more datasets in the future.

For the future improvement of the work, the following points can be considered to strengthen the performance of the integrated YOLO model together with the LIDAR data for the 3D obstacle detection in self-driving cars. Some of the solutions mentioned are the use of sophisticated data augmentation methods, and domain adaptation to enhance the model's ability to generalize across domains, the use of multi-sensor fusion, and attention mechanisms, to optimize models, and availing better Bird's Eye View projections. The respective components of real-time processing can be supported by using hardware acceleration and model pruning; further, continuous learning or object tracking can improve the consistency of the objects' detection. Thus, the methods such as adversarial testing and safety-critical evaluations can be used for improvement of robustness and safety, while cooperation with large-scale datasets can increase its generalization. To accelerate the progress in the presented area, it is stated that there is more to discover in new architectures like hybrid models and NAS.

References

Alif, M. A. R. and Hussain, M. (2024). Yolov1 to yolov10: A comprehensive review of yolo variants and their application in the agricultural domain, *arXiv preprint arXiv:2406.10139* .

Choi, J. D. and Kim, M. Y. (2023). A sensor fusion system with thermal infrared camera and lidar for autonomous vehicles and deep learning based object detection, *ICT Express* **9**(2): 222–227.

Fan, Y.-C., Yelamandala, C. M., Chen, T.-W. and Huang, C.-J. (2021). Real-time object detection for lidar based on ls-r-yolov4 neural network, *Journal of Sensors* **2021**(1): 5576262.

Geiger, A., Lenz, P., Stiller, C. and Urtasun, R. (2013). Vision meets robotics: The kitti dataset, *The International Journal of Robotics Research* **32**(11): 1231–1237.

Haghighi, H., Wang, X., Jing, H. and Dianati, M. (2024). Review of the learning-based camera and lidar simulation methods for autonomous driving systems, *arXiv preprint arXiv:2402.10079* .

Hekimoglu, A., Schmidt, M. and Marcos-Ramiro, A. (2024). Monocular 3d object detection with lidar guided semi supervised active learning, *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2346–2355.

Jiang, P., Ergu, D., Liu, F., Cai, Y. and Ma, B. (2022). A review of yolo algorithm developments, *Procedia computer science* **199**: 1066–1073.

Kong, L., Liu, Y., Chen, R., Ma, Y., Zhu, X., Li, Y., Hou, Y., Qiao, Y. and Liu, Z. (2023). Rethinking range view representation for lidar segmentation, *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 228–240.

Lan, W., Dang, J., Wang, Y. and Wang, S. (2018). Pedestrian detection based on yolo network model, *2018 IEEE international conference on mechatronics and automation (ICMA)*, IEEE, pp. 1547–1551.

Lang, A. H., Vora, S., Caesar, H., Zhou, L., Yang, J. and Beijbom, O. (2019). Pointpillars: Fast encoders for object detection from point clouds, *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12697–12705.

Li, S., Geng, K., Yin, G., Wang, Z. and Qian, M. (2023). Mvmm: Multiview multimodal 3-d object detection for autonomous driving, *IEEE Transactions on Industrial Informatics* **20**(1): 845–853.

Nakashima, K., Iwashita, Y. and Kurazume, R. (2023). Generative range imaging for learning scene priors of 3d lidar data, *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1256–1266.

Qi, C. R., Liu, W., Wu, C., Su, H. and Guibas, L. J. (2018). Frustum pointnets for 3d object detection from rgb-d data, *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 918–927.

Sohan, M., Sai Ram, T., Reddy, R. and Venkata, C. (2024). A review on yolov8 and its advancements, *International Conference on Data Intelligence and Cognitive Informatics*, Springer, pp. 529–545.

Tychsen-Smith, L. and Petersson, L. (2018). Improving object localization with fitness nms and bounded iou loss, *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6877–6885.

Vinodkumar, P. K., Karabulut, D., Avots, E., Ozcinar, C. and Anbarjafari, G. (2023). A survey on deep learning based segmentation, detection and classification for 3d point clouds, *Entropy* **25**(4): 635.

Viswanath, K., Jiang, P., Sujit, P. and Saripalli, S. (2023). Off-road lidar intensity based semantic segmentation, *International Symposium on Experimental Robotics*, Springer, pp. 608–617.

Wang, A., Chen, H., Liu, L., Chen, K., Lin, Z., Han, J. and Ding, G. (2024). Yolov10: Real-time end-to-end object detection, *arXiv preprint arXiv:2405.14458* .

Wang, Y., Wang, W., Liu, J., Chen, T., Wang, S., Yu, B. and Qin, X. (2023). Framework for geometric information extraction and digital modeling from lidar data of road scenarios, *Remote Sensing* **15**(3): 576.

Wilson, B., Mitchell, N. A., Pontes, J. K. and Hays, J. (2024). What matters in range view 3d object detection, *arXiv preprint arXiv:2407.16789* .

Zhuang, Z., Li, R., Jia, K., Wang, Q., Li, Y. and Tan, M. (2021). Perception-aware multi-sensor fusion for 3d lidar semantic segmentation, *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16280–16290.