A Comprehensive Model for Advanced Road Scene Understanding: YOLO-CNN Fusion for Accurate Road Segmentation and Object Detection in Varied Conditions

by

WASHIFUR RAHMAN
20201124

SHAMIR ROY
20241015

AHSANUL ISLAM
22241153

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science and Engineering

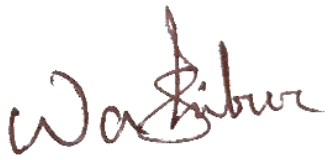
Department of Computer Science and Engineering
Brac University
January 2025

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing a degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material that has been accepted or submitted for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:



Washifur Rahman
20201124

Shamir Roy
20241015

Ahsanul Islam
22241153

Approval

The thesis titled "A Comprehensive Model for Advanced Road Scene Understanding: YOLO-CNN Fusion for Accurate Road Segmentation and Object Detection in Varied Conditions" submitted by

1. Washifur Rahman (ID: 20201124)
2. Shamir Roy (ID: 20241015)
3. Ahsanul Islam (ID: 22241153)

of Fall, 2024 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on January, 2025.

Examining Committee:

Supervisor:
(Member)

Amitabha Chakrabarty, PhD
Professor
Department of Computer Science and Engineering
BRAC University

Program Coordinator:
(Member)

Md Golam Rabiul Alam, PhD
Professor
Department of Computer Science and Engineering
BRAC University

Head of Department:
(Chair)

Sadia Hamid Kazi, PhD
Chairperson and Associate Professor
Department of Computer Science and Engineering
BRAC University

Abstract

The critical component in autonomous driving is road scene understanding that includes accurate object recognition and precise road segmentation. Most existing models do not accurately represent the roads of Bangladesh as those are are mostly trained and tested on the data from the organized roads found in developed countries. This study tackles the challenging problem of road perception in the context of autonomous driving in Bangladesh by proposing a YOLO-CNN fusion model that takes into account the country's diverse weather patterns and unstructured road layouts. To evaluate model performance, we tried out various object detection architectures, such as YOLOv8, YOLOv9, YOLOv10, and YOLOv11, and for road segmentation, we tested U-Net and ResU-Net segmentation models as well. We gathered a dataset of 2,495 images of roads from different locations and environments including highways, villages, foggy and rainy situations, as well as nighttime. The results of the experiments confirmed that the Medium model of YOLOv11 was the most accurate of all versions of YOLO at object detection with an accuracy of 79.3%. Likewise, U-Net outperformed ResU-Net in terms of accuracy and the IoU score of 80.69% for road areas, indicating a closer match to actual road areas. The best-performing models of object detection and segmentation are then combined to create a comprehensive road scene understanding system. The findings show that the combination of YOLOv11 and U-Net Fusion Model enhances object detection segmentation in road environments greatly, which makes it relevant for self-driving car applications. The system was additionally implemented as a web based prototype so that users could upload images and see the results of detection and segmentation visually. Work on these objectives will emphasize improving the economization of processing power for use with low-resource devices, including depth perception with LiDAR, and widening the dataset for better performance across different driving environments.

Keywords: YOLO, CNN, Road Segmentation, Object Detection, Autonomous Vehicles

Acknowledgement

First of all, we would want to thank Almighty Allah for supporting us both mentally and physically and to finish our thesis titled "A Comprehensive Model for Advanced Road Scene Understanding: YOLO-CNN Fusion for Accurate Road Segmentation and Object Detection in Varied Conditions" within the deadline. Also for giving us the willpower, motivation and information that was necessary to complete our thesis.

We are very much thankful to our supervisor, Dr. Amitabha Chakrabarty, Professor of Computer Science and Engineering. His indispensable advice and supervision through-out the whole thesis -writing process was invaluable. We want to express our gratitude toward him for his unwavering support that allowed us to process our ideas and work more efficiently.

Lastly, this research would not have been possible without the effort of individual people who put their efforts in an organized way. We would like to thank our beloved parents for supporting us since without their support, we might not have been able to complete our thesis work.

Table of Contents

Declaration i

Approval ii

Abstract iii

Acknowledgement iv

Table of Contents v

List of Figures vii

List of Tables viii

Nomenclatures ix

1 Introduction 1
- 1.1 Motivation . . . 2
- 1.2 Research Problem Statement . . . 2
- 1.3 Research Objectives . . . 3

2 Literature Review 4
- 2.1 Artificial Intelligence . . . 4
- 2.2 Deep Learning Techniques . . . 5
- 2.3 Object Detection Models . . . 5
- 2.4 Computer Vision Regarding Weather . . . 5
- 2.5 Semantic Segmentation for Road Structures . . . 6
- 2.6 Related Works . . . 7
- 2.7 Research Gap . . . 9

3 Dataset 10
- 3.1 Description . . . 10
- 3.2 Data Splitting . . . 10
- 3.3 Data Samples . . . 10
- 3.4 Data Pre-processing . . . 11
- 3.5 Data Annotation . . . 12

4 Methodology 13
- 4.1 Proposed Methodology for Object detection . . . 13
  - 4.1.1 Yolov8 . . . 13

4.1.2 Yolov9 . . . 14
4.1.3 Yolov10 . . . 14
4.1.4 Yolov11 . . . 15
4.2 Proposed Methodology for Road segmentation . . . 16
4.2.1 U-Net . . . 16
4.2.2 ResU-Net . . . 16
4.3 Working Plan . . . 18

5 Implementation **20**
5.1 Object Detection . . . 20
5.1.1 Training Setup and Hyperparameters . . . 20
5.1.2 Optimizer . . . 21
5.2 Road Segmentation . . . 22

6 Result Analysis **23**
6.1 Object Detection . . . 23
6.1.1 YOLOv8 Summary . . . 24
6.1.2 YOLOv9 Summary . . . 26
6.1.3 YOLOv10 Summary . . . 28
6.1.4 YOLO v11 Summary . . . 30
6.2 Road segmentation . . . 32
6.2.1 U-Net: . . . 32
6.2.2 ResU-net . . . 34
6.3 Analysis and Discussion . . . 35
6.3.1 Object Detection . . . 35
6.3.2 Road Segmentation . . . 36
6.4 Merged Overview . . . 37

7 Website Deployment **38**
7.1 Limitations . . . 39

8 Conclusion **40**
8.1 Conclusion . . . 40
8.2 Future Work . . . 40

Bibliography **43**

List of Figures

3.1 Data Partition (bar chart) . . . 11
3.2 Sample images . . . 11
3.3 Annotated image . . . 12
3.4 Predicted mask . . . 12

4.1 Architecture Visualization of YOLOv8 Model [27] . . . 13
4.2 Architecture Visualization of YOLOv9 Model[24] . . . 14
4.3 Architecture Visualization of YOLOv10 Model[23] . . . 15
4.4 Architecture Visualization of YOLOv11 Model[25] . . . 15
4.5 U-Net Architecture[1] . . . 16
4.6 ResU-Net Architecture[26] . . . 17
4.7 Workflow diagram . . . 19

6.1 YOLOv8m Training and Validation Accuracy and Loss Plot . . . 24
6.2 Confusion Matrix YOLOv8 . . . 25
6.3 YOLOv9m Training and Validation Accuracy and Loss Plot . . . 26
6.4 Confusion Matrix YOLOv9 . . . 27
6.5 YOLOv10m Training and Validation Accuracy and Loss Plot . . . 28
6.6 Confusion Matrix YOLOv10 . . . 29
6.7 YOLOv11m Training and Validation Accuracy and Loss Plot . . . 30
6.8 Confusion Matrix YOLOv11 . . . 31
6.9 U-Net Evaluation Metrics . . . 32
6.10 Original vs Predicted mask example (U-Net) . . . 33
6.11 ResU-Net Evaluation Metrics . . . 34
6.12 Original vs Predicted mask example (ResU-Net) . . . 35
6.13 Recall-Confidence curve . . . 36
6.14 U-Net vs ResU-Net evaluation metrics . . . 36
6.15 Merged examples . . . 37

7.1 Web landing interface . . . 38
7.2 Web result interface . . . 39

List of Tables

3.1 Data splitting 10
3.2 Preprocessing pipeline 11
3.3 Class Labels and Their Grayscale Mappings 12

5.1 Default Hyperparameters 20
5.2 Training Configuration 22

6.1 Model Performance 23
6.2 Classification Report 1 24
6.3 Classification Report (continued) 24
6.4 Classification Report 1 26
6.5 Classification Report (continued) 26
6.6 Classification Report 1 28
6.7 Classification Report (continued) 28
6.8 Classification Report 1 30
6.9 Classification Report (continued) 30

Nomenclature

Several symbols and abbreviations that will be used later in the document's body are described in the following list.

AI Artificial Intelligence

ML Machine Learning

U-Net U-shaped encoder-decoder network architecture

Res U-Net Residual U-Net

YOLO You only look once

Chapter 1

Introduction

Since 1956, Artificial Intelligence has been playing a vital role in making our life more easier. AI is being developed day by day with all new features to different algorithms. This has become more popular in many countries. It is because it lessens the workload of humans, in a more efficient way which is way faster. Moreover, it ensures safety in most cases because humans can make errors. But machines with proper models and algorithms, have less chance to make an error compared to humans.

In autonomous driving, AI makes a great impact. With the advancement of integrated AI, cars can detect roads and objects, and evaluate environments. As it can mimic human intelligence, we can train our AI algorithm to track which is a better route to take to reach our destination. In many countries, there are different kinds of research regarding AI in autonomous vehicles and self-driving systems. Two important aspects of this research are road segmentation and object detection. Accidents can happen because of errors in human judgment. According to [14], common errors that result in road accidents are drunk driving, drowsy driving, recognition errors, decision-making errors, speeding, not checking for blind spots, etc. Even though there is a very low chance that a machine might malfunction and accidents might occur, there is a higher chance that AI algorithms cannot make these errors because before implementing models in self-driving cars these models are tested thoroughly. This is why we have proposed our model.

There are several techniques developed that have been used in computer vision in aspects of road object detection and segmentation. However, in recent years we can see the advancement of the techniques and algorithms regarding object detection and segmentation. These image processing and deep learning algorithms are thoroughly developed. Image classification algorithm YOLO (You Only Look Once) is a popular object classification technique widely used. There are many versions of YOLO. We have also merged YOLO with a deep learning approach using a popular neural network which is a Convolution neural network (CNNs).

1.1 Motivation

In Bangladesh, road accidents occur day to day at an alarming rate. The article [10] highlights the reasons for road accidents which are: reckless driving, over-speeding, overtaking, breaking traffic laws, etc. To ensure the safety of human lives in these difficult times, it is better to make advancements in autonomous driving systems. This is why we have collected data from Bangladesh to work with our research model. The road scenarios are quite complicated in Bangladesh as there are too many objects here and there. Additionally, village roads have not been studied before. So it is necessary to include those data in our model to further research on this environment. Lastly, different weather conditions may affect these results. Therefore, it worked as a motivation for us to see how our model would work in various weather and if we can implement the model efficiently.

1.2 Research Problem Statement

Day by day, the use of automation in cars is increasing at a higher rate as the modern world moves forward with daily discoveries. According to [17], some aspects of automation were integrated into the automobile industry which was approximately around 31 million cars and it is expected that this number will rapidly surpass 54 million this year of 2024.

It is important that to keep up with the modern advancing world, we focus on our country's situation for our research problem. There are various kinds of issues in aspects of Bangladesh to complete our research and development because the roads are not the same as in other foreign countries. There are too many objects for the autonomous model to classify and many kinds of vehicles.

Furthermore, the weather is another important thing to mention for our research problem. According to [13], rainfall, snowfall, sunbeams, and fog can have a great negative impact on recognition, classification, and segmentation. To clarify this, in Bangladesh, we need to focus more on the rainfall and fog problems along with sunny days as these are the problems mostly seen here, and snowfall does not occur in Bangladesh.

In addition to this, nighttime is another research problem as it might be hard to detect objects and segmentation of roads at night. Poor lighting conditions may degrade the visibility of images in a significant way and this may affect the performance of computer vision system models [5]. With this in mind, it is important to enhance these images in a preprocessing method so that we can feed them into our model. Rural areas have different roads and pathways compared to urban areas. According to [2], rural or village areas consist of narrow roads, buildings, and trees beside roads and shelters the road segments, larger curvature compared to urban areas are seen in rural areas. Therefore, it is a difficult task to extract roads and objects in rural areas.

With all of this, a question might arise whether the YOLO-CNN technique can be used for road segmentation and object detection in different conditions. The

answer to this is yes and there is a lot of research that has been conducted in this field. However, with the set of data from Bangladesh, there is not much research in this field. So we have collected a dataset from Bangladesh to see how well our model works.

Therefore, the question that this research is trying to answer is: ***How effective our YOLO-CNN model is to achieve accurate road segmentation in differ ent environmental conditions?***

This research will answer the above question by exploring the combination of YOLO and CNN algorithms.

1.3 Research Objectives

Our research aims to develop a comprehensive model for accurate road segmentation and object detection using YOLO and CNN in the field of Artificial intelligence and Deep learning that can be used in the study of autonomous vehicles. Regardless of other things for autonomous driving our research is one crucial part that can be used later on. The main objectives of this research are:

1. To make the complex Bangladesh roads safer with the power of AI and ML.
2. To automate vehicles on roads that reduce human intervention and make journey better.
3. To develop a robust model with YOLO and CNN that can automate cars and reduce road accidents.
4. To automate cars in complex, adverse and unpredictable road conditions of Bangladesh.
5. To offer recommendations on the model in improving it further.

Chapter 2

Literature Review

Artificial Intelligence is gaining popularity day by day to be used in many sectors as it ensures the performance of tasks that require human intelligence. It uses a wide range of techniques, and algorithms, and also learns itself. According to [20], the automotive industry is projected to experience a substantial compound annual growth rate of approximately 40%, reaching a valuation of $15.9 billion by 2027, and this growth will be driven by the increasing rise in demand for the adoption of intelligent technologies, including image recognition. Speaking of image recognition, we dive into realms of AI, one of which is deep learning which is widely used. Now the question arises about why AI is rapidly used in autonomous driving. The answer is very simple; it is more efficient and easy. As mentioned earlier, many road accidents happen because of the driver's recklessness, carelessness, and sometimes because of human error. However, autonomous driving is safe in this case with proper application and efficient models. Therefore, we have reviewed some literature that can help us make an efficient model for our research. These works of literature include AI, Deep learning techniques, Object detection models, Rural Scene understanding, Computer vision models regarding weather, semantic segmentation for road structures, etc.

2.1 Artificial Intelligence

According to [22], AI mimics human intelligence through problem-solving and decision-making with the combination of data, computers, and technology. With methods like machine learning and deep learning, AI developments have grown in the field of autonomous vehicles. It is further explained that the Advanced Driver Assistance Systems (ADAS) technology uses various features like emergency braking, segmentation, departure alerts, detection, etc. with its in-built sensors, cameras, and machine learning algorithms. The perception system that is integrated into autonomous vehicles uses sensors that can detect environmental elements including humans, traffic signs, etc. They also use GPS (Global Positioning System) and GNSS (Global Navigation Satellite System) to know their own position and destination. However, it was indicated in this research that 50% of customers wouldn't feel safe using autonomous-driving cars because of safety issues. In this case, the advancement of a fully autonomous vehicle that has the capability of understanding the environment, objects, roads, traffic signs, and road markings can revolutionize transportation while ensuring road safety for customers.

2.2 Deep Learning Techniques

The authors of [3] address the many ways in which deep learning methods for autonomous driving might improve the efficiency and security of these cars in their thorough analysis of the field. The study highlights the importance of Convolutional Neural Networks (CNNs) in image recognition tasks necessary for interpreting road scenes and obstacle detection. Moreover, through a system of rewards and penalties, it emphasizes how reinforcement learning is applied in decision-making processes to optimize behaviors like route planning and obstacle navigation. Even with these developments, the research notes the difficulties that still need to be overcome, including the need for large and varied training datasets, the complexity of computational models, and maintaining consistent performance under a range of driving scenarios. It also focuses on the use of Bounding-Box-Like Object Detectors, which are essential for autonomous navigation since they help locate and identify things in a scene. A crucial component of autonomous vehicle sensory processing, this technology uses algorithms to create bounding boxes around items it detects. This allows for accurate object localization and categorization in real time.

2.3 Object Detection Models

The study [16] aims to comprehensively evaluate recent developments and techniques in real-time object detection, with specific emphasis given to the YOLO methodology. It looks at how YOLO evolved, highlighting the many iterations and highlighting how effective it has become in terms of processing speed and accuracy of detection. The study goes on to highlight the distinctive qualities of YOLO, such as its single detection method, by contrasting it with other conventional approaches to object identification. Additionally, it emphasizes solutions for object detection issues, primarily about YOLO.

This study [12] critically analyzes the YOLOv4 algorithm highlighting its application towards real-time object detection including integration with a webcam within Google Colab. It is an overview of the development of object detection systems, with a special focus on speed and accuracy improvements that came with YOLOv4. It is discussed how with YOLOv4, almost overnight the approach revolutionized many fields, including robotics, human-computer interaction, and surveillance among others. This underscores the efficiency and the user-friendliness of the algorithm, which explains its indispensability as a tool to bridge the gap in democratizing state-of-the-art computer vision technology. The paper also discusses the wider implications of these developments in practical applications showing how YOLOv4 might potentially revolutionize real-time analysis and decision-making in varied fields and sectors.

2.4 Computer Vision Regarding Weather

The paper [18] describes an upgraded model for recognizing objects in foggy weather situations. The model is based on YOLOv5s and plays a crucial role in autonomous

driving. The authors of the paper have made some key modifications to YOLOv5s by incorporating SwinFocus, a target detection layer, a decoupled head, and the Soft-NMS approach. These adjustments have been made to improve the detection performance of the model in conditions with decreased visibility and blurry objects. The results show how customized deep learning models can adapt to specific weather conditions, leading to significant improvements in computer vision applications in various scenarios.

2.5 Semantic Segmentation for Road Structures

Using a modified Segnet model with a pre-trained auto-driving model, as suggested by Badrinarayan et al., the research [6] focuses on implementing semantic segmentation for autonomous ground vehicles. Eleven classes were included in the SegNet model for things including bikes, vehicles, and roads, which was based on the VGG16 architecture. According to the experimental results, the performance was encouraging, with an average intersection over union (IoU) of 66.1 and individual class IoU ranging from 11.5 to 97.1. In autonomous driving scenarios, the proposed approach improves object detection and classification by efficiently classifying road regions. The research emphasized the promise of deep learning techniques, particularly Seg net architecture, for precise semantic segmentation in real-time autonomous ground vehicle applications.

The paper [7], explores different methodologies of multiple object tracking (MOT) to draw a survey about deep learning. Herein, the main challenges in MOT such as occlusion and ID switching are outlined then several deep learning algorithms primarily Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) networks, and Convolutional Neural Networks (CNNs) are introduced. The article comes short of being exhaustive and provides a complete review of the benchmark datasets, as well as the industry-standard MOT evaluation approaches by classifying former MOT algorithms while detailing the fundamental methodologies. It also discusses the diversity of MOT challenges and trends, outlines prospective research directions as well as showcases the most recent developments and trends in MOT.

The research [9] describes a deep learning technique termed background blurring, or rather, "bokeh," which improves semantic segmentation. With this method, the impact of unimportant background features in photos is supposed to be lessened, enabling neural networks to concentrate more on their primary subject. According to the paper, integrating the bokeh module on top of fully convolutional network-based architectures already in place is viable, and benchmark datasets like PASCAL VOC 2012 and CamVid can be used to significantly improve segmentation accuracy. It could provide a more advanced approach to semantic segmentation tasks in a variety of applications since it presents a new paradigm shift in how dataset properties and foreground-background differentiation are handled.

The study [19] focuses on exploring medical image segregation experiments. The SAM and HQ-SAM models are integrated to achieve exact segregation. To identify the boundary boxes, the YOLOv8 model is used heuristically. Various medical imaging datasets are used in the study, including ultrasound images, CT images, and

X-ray pictures. Evaluation measures such as recall, F1 score, Dice Score, and precision are applied to measure the segmentation accuracy in the research. The results show that SAM always outperformed YOLOv8 and HQ-SAM across all models when it comes to segmentation accuracy. Although the benefits provided by HQ-SAM do not justify its processing cost, the YOLOv8+SAM model is still very promising for medical applications, as it improves the segmentation performance.

In this paper titled [8], the FasterSeg model for autonomous driving has been evaluated and verified in terms of the levels of accuracy and real-time capacities in the Carolo-Cup environment with the employment of the NVIDIA Jetson AGX Xavier hardware. Neural network FasterSeg of semantic segmentation model was able to show more than 64 percent prediction accuracy and frame rate reaching 247.11 FPS. Synthetic and real image training allowed this model to present itself as an example of low latency/high-accuracy balance. The study also compared the first person and bird's eye views, highlighting other lessons of versatility in FasterSeg as there was an observation of similar mean Intersection over Union (mIoU) value for both. Faster, the research found out that utilizing 16-bit Floating Point arithmetic doubled the frame rate with negligible accuracy loss when compared to 32-bit. The results verify the capacity of FasterSeg for real-time semantic segmentation in autonomous driving.

From the above discussion, the approach in road object detection has similar approaches using different versions of YOLO models. Similar to Road segmentation various techniques were introduced for Convolution neural networks. However, we have to consider unique challenges such as environmental changes, nighttime images, rainy days, foggy days, and adverse conditions. This way we merge the best method for our model to work with our dataset.

2.6 Related Works

According to the study [21], the research carefully evaluates the efficacy and accuracy of numerous YOLO models, including YOLOv5s, YOLOv5m, YOLOv5l, YOLOv5x, and YOLOv7, in the context of diagnosing road defects using drone-based photos. The paper compares the capabilities of different models and provides information on how practical they are for use in surveillance and road maintenance applications. It draws attention to the ways that advancements in YOLO designs enhance road safety, highlighting the accuracy as well as dependability of these models in identifying and classifying potential risks in a range of environments, including potholes and fissures. During the trial, it was observed that YOLOv7 had the highest success rate of 68.3%. Other models such as YOLOv5l, YOLOv5m, YOLOv5x, and YOLOv5s, were also performing well, but not as successful as YOLOv7. YOLOv5l achieved a score of 66.8%, while YOLOv5m was able to attain 66.3%. Similarly, YOLOv5x scored 66%, and YOLOv5s obtained a score of 63%.

The study [11] provides a thorough examination of object extraction and detection, focusing primarily on the combination of YOLOv7 and U2-Net technologies. It provides an overall view of the improvements in the YOLO algorithm, highlighting its ingenious architectural design and its prominence in item recognition efficiency com-

pared to other methods. The paper further discusses developments like PP-YOLO and how they enhance object categorization and location accuracy. Additionally, the study delves into the challenges of working with noisy photos and complex image segmentation techniques. This research makes a significant contribution to the field of computer vision by presenting some of the most recent advances in methods for object identification and extraction.

The work [4] highlights the difficulties of rural road extraction from remote sensing photos and makes a significant addition to our understanding of rural settings. Compared to typical pixel-based analysis, the OBIA technique considers spatial, spectral, and contextual information of objects. This makes it an innovative approach in geographic information systems (GIS) and remote sensing. Method used in this work. In rural areas with complex route designs that often blend in with the surrounding landscape, this approach works very well. OBIA's ability to precisely detect and extract rural roads from high-resolution satellite images is demonstrated by a case study in Assam, India. The study combines object-based segmentation and classification techniques to address common problems in rural road mapping, including uneven alignment, varying road widths, and the presence of natural features such as vegetation and water bodies. The research results reveal that rural road drainage can be done with a high degree of accuracy, demonstrating how OBIA can improve the mapping and analysis of rural transportation networks. Besides making significant contributions to the field of remote sensing and GIS, this research has important implications for rural planning and development. It provides a powerful tool to better understand and control rural infrastructures.

The study [15] provides an explanation of image segmentation in computer vision, outlining its significance as well as the techniques that are employed in applications like augmented reality, driverless cars, and medical imaging. Additionally, it describes instance segmentation and semantic segmentation and shows how they vary from traditional picture segmentation. It describes cutting-edge methods like U-Net, DeepLab, and MasK R-CNN and examines how to balance the needs of accuracy and computation. The statement highlights the increasing necessity of industry segmentation, acknowledging the potential and constraints of deep learning technologies.

2.7 Research Gap

While significant research has been conducted on road segmentation and object recognition in urban scenarios; there has not been much study focusing on village roads in Bangladesh. Most existing models are mostly trained and tested on the data from the organized roads found in developed countries, which do not accurately represent the Bangladeshi village roads which are narrow and unstructured. Moreover, the rural roads of Bangladesh lack proper lane markings, mixed traffic, and frequent obstructions such as parked vehicles, pedestrians, and livestock. The existing datasets do not incorporate such complexities, which makes the models particularly ill-suited for generalization in rural environments. The absence of a publicly available dataset that captures the essence of Bangladeshi roads, specifically the village roads, is another prominent gap in the existing research.

Chapter 3

Dataset

3.1 Description

For this study we have collected 2495 images of road scenes from different parts of Bangladesh. All the data went through a cleaning process and any duplicate images and poor quality images are removed to maintain a higher standard of data. We focused on grouping our collection of data into different scenarios and weather conditions such as foggy weather, rainy weather, village roads, flyover scenarios, highway scenarios, night scenarios, and mixed scenarios. Then we fed the model with an efficient amount of images from each of these categories.

3.2 Data Splitting

The data contained 2495 images which were split into three categories:

Data Split	**Training**	**Validation**	**Testing**
Percentage	80%	10%	10%
Image numbers	1996	250	249

Table 3.1: Data splitting

We followed the Table 3.1 structure to sort out the split categories. This split enables us to make sure the model has enough data for training while sustaining different sets for evaluation and validation to prevent overfitting.

3.3 Data Samples

For the Object Detection model, from these data images, we have narrowed down to 13 different classes that are common to appear on roads in Bangladesh. These classes are 'Person', 'Rickshaw', 'Rickshaw Van', 'Auto Rickshaw', 'Truck', 'Pickup Truck', 'Private Car', 'Motorcycle', 'Bicycle', 'Bus', 'Micro Bus', 'Covered Van', and 'Human Hauler'. Additionally, we partitioned our primary data where different environments and weather conditions were acknowledged. In figure 3.1, we can visualize the partitioned data. Furthermore, we have shown some sample images from

our primary data collection in figure 3.2.

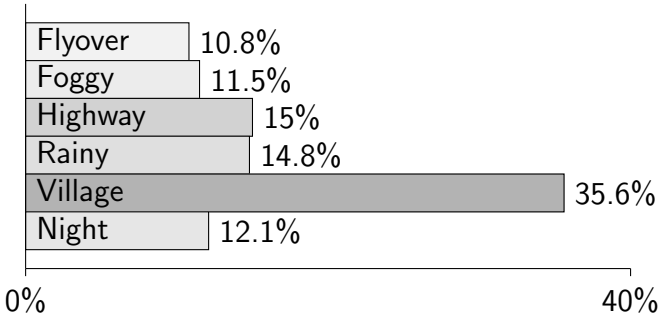


Figure 3.1: Data Partition (bar chart)

Figure 3.2: Sample images

3.4 Data Pre-processing

Table 3.2 outlines the preprocessing pipeline, which involves resizing images to 640×640 and a number of data augmentation approaches were used in the pre-processing stage in order to enhance the generalization capability of the models. The transformations performed include rotation, brightness changes, flipping, and modification of images which were previously blurred so that clarity and feature recognition is prominent. These augmentations changed the dataset in such a manner that it enabled the model to learn variations in road types, lighting conditions, and motion blurs. These approaches reduced overfitting and improved segmentation performance by exposing the model to different scenarios. Augmentation is done using the Albumentations library.

Data Augmentation	**Parameters**
Resize	Size: (640, 640)
Random Horizontal Flip	-
Random Vertical Flip	-
Random Rotation	Degree: 40
Brightness	0.5

Table 3.2: Preprocessing pipeline

3.5 Data Annotation

For Object detection, as illustrated in Figure 3.3, we have used "LABELIMG" to annotate images to ensure accurate bounding boxes around the objects that can be classified in road scenarios.

Figure 3.3: Annotated image

For the Road Segmentation model, each image was separately annotated in order to create segmentation masks for the road which were then classified into three categories: Non-drivable Area , My Way, and Other Way. The segmentation masks, initially labeled using grayscale values and then were mapped to integer class labels to make them suitable for model training. The predicted mask is shown in Figure 3.4, which corresponds to the class labels as well as the grayscale mappings presented in Table 3.3.

Class	**Grayscale Value**	**Mapped Label**
Non-Drivable Area	0	0
My Way	128	1
Other Way	255	2

Table 3.3: Class Labels and Their Grayscale Mappings

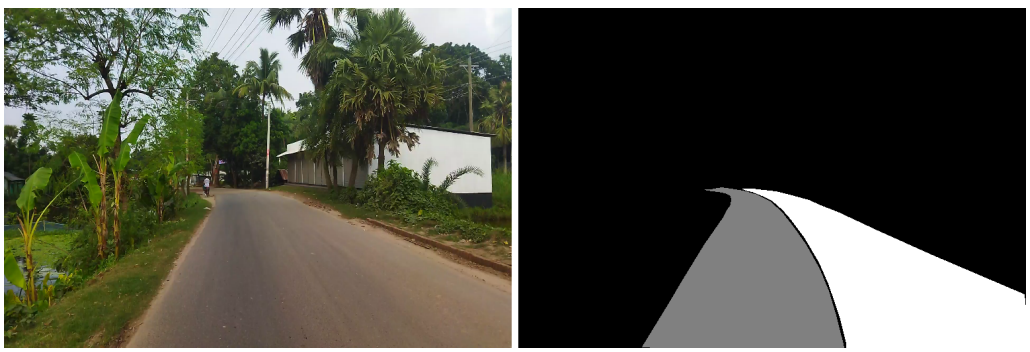


Figure 3.4: Predicted mask

Chapter 4

Methodology

4.1 Proposed Methodology for Object detection

4.1.1 Yolov8

Yolov8 is a state of the art object detection model which is designed for applications in real time. It is based on CNN architecture that is built upon convolution layers, batch normalization and other activation functions like SiLU that can enhance its learning capabilities and feature extractions. The network structure comprises a head for object detection, a backbone for feature extraction and a neck for multi-scale feature fusion. With better model scaling and anchor-free detection its structure benefits from fast inference speed making it a suitable model to detect real time objects. However, YOLOv8 requires notable computational resources and it is difficult to tune its hyperparameter. In Figure 4.1, we can visualize the architecture for YOLOv8.

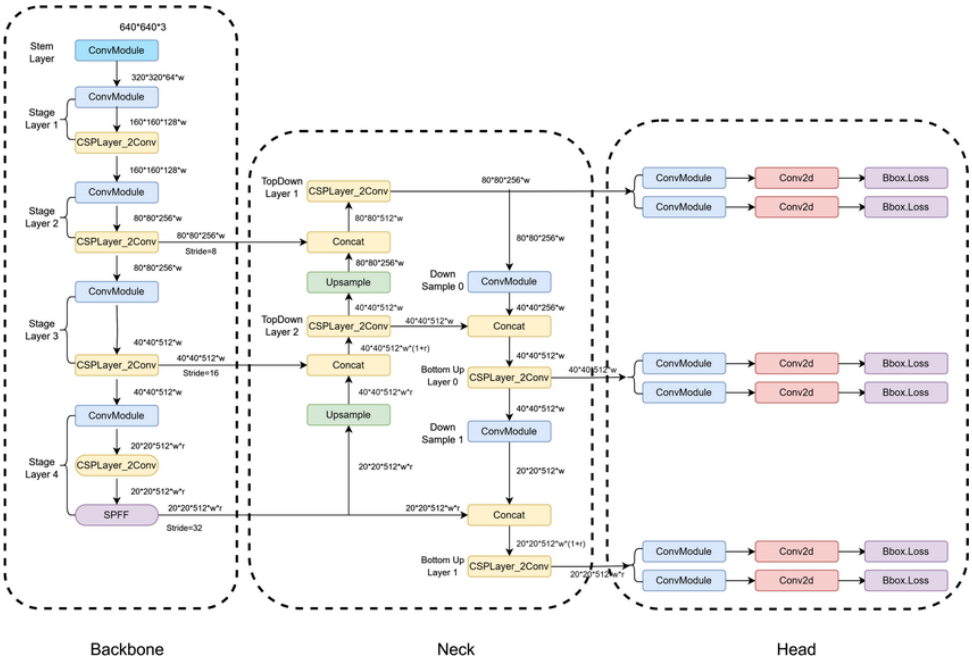


Figure 4.1: Architecture Visualization of YOLOv8 Model [27]

4.1.2 Yolov9

By utilizing and introducing novel enhancements in attention mechanisms and model scaling, YOLOv9 makes complex detection tasks more adaptable than its previous versions. Similar to other YOLO versions it contains three main components: Backbone, Neck and Head. In addition to this, its Neck fuses multi-scale features using an improved Feature Pyramid Network (FPN) and Path Aggregation Network (PAN). The architecture of YOLOv9 is visualized in Figure 4.2.

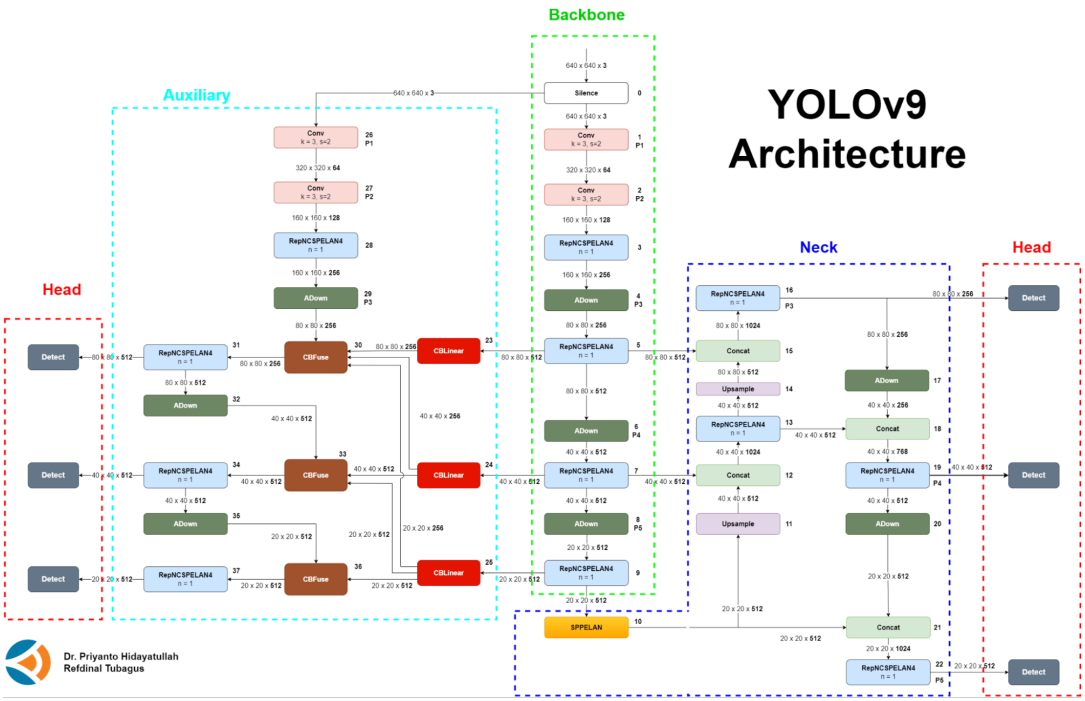


Figure 4.2: Architecture Visualization of YOLOv9 Model[24]

4.1.3 Yolov10

YOLOv10 comprises various key innovations such as improved dynamic sparse computation, model scaling strategies and hybrid and improved attention mechanisms. Its Backbone is improved for better representation learning with advanced channel attention mechanisms. Also, localization and classification accuracy are improved through an advanced detection Head. Figure 4.3 provides a detailed visualization of the YOLOv10 architecture.

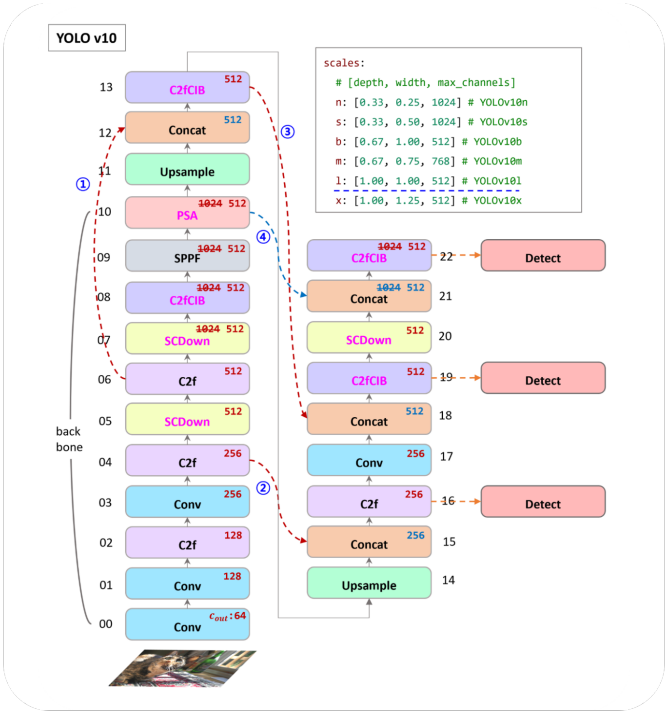


Figure 4.3: Architecture Visualization of YOLOv10 Model[23]

4.1.4 Yolov11

YOLOv11 is the latest state of the art real time object detection model. Along with other attributes similar to other YOLO versions, YOLOv11 uses an exceptional hybrid Backbone that incorporates lightweight transformer modules. By using a redefined anchor-free detection it reduces computational overhead and improves accuracy of localization. Furthermore, it allocates processing power to selective high-informative regions of the image for boosting efficiency. In Figure 4.4, we can visualize the architecture for YOLOv11.

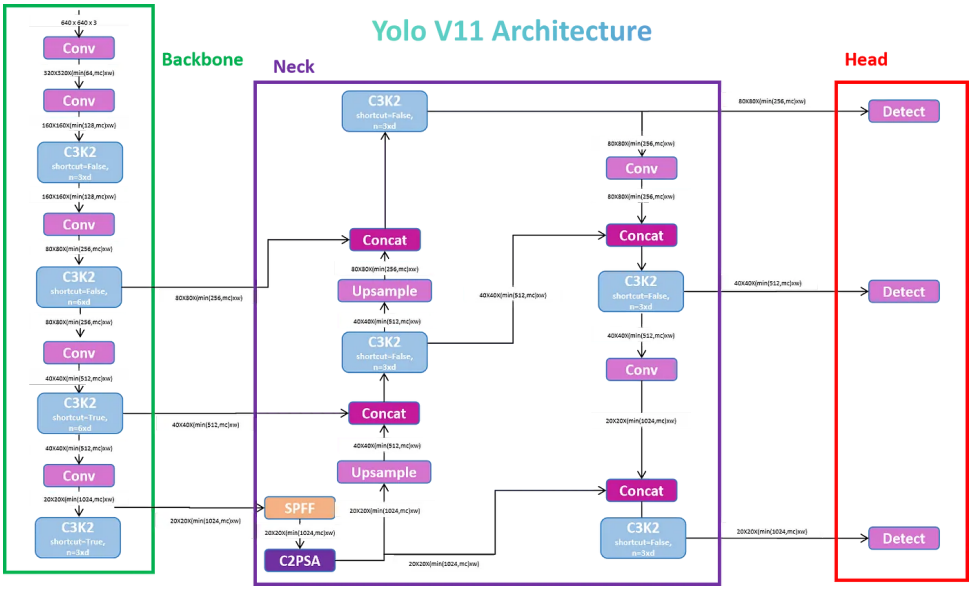


Figure 4.4: Architecture Visualization of YOLOv11 Model[25]

4.2 Proposed Methodology for Road segmentation

4.2.1 U-Net

U-Net is a type of convolutional neural network (CNN) that was developed for image segmentation purposes. It follows an encoder-decoder architecture with skip connections to facilitate better retention of spatial information. The encoder feature is obtained from hierarchical extraction performed through convolutional and max pooling layers, while the decoder reconstructs details of the space with upsampling followed by concatenation of the relevant features extracted from the encoder. Then, a multiclass segmented output is generated through softmax activated 1x1 convolution in the final output layer. Its use is widespread due to its capability of retaining details and small object structures. The U-Net architecture used for road segmentation is shown in Figure 4.5.

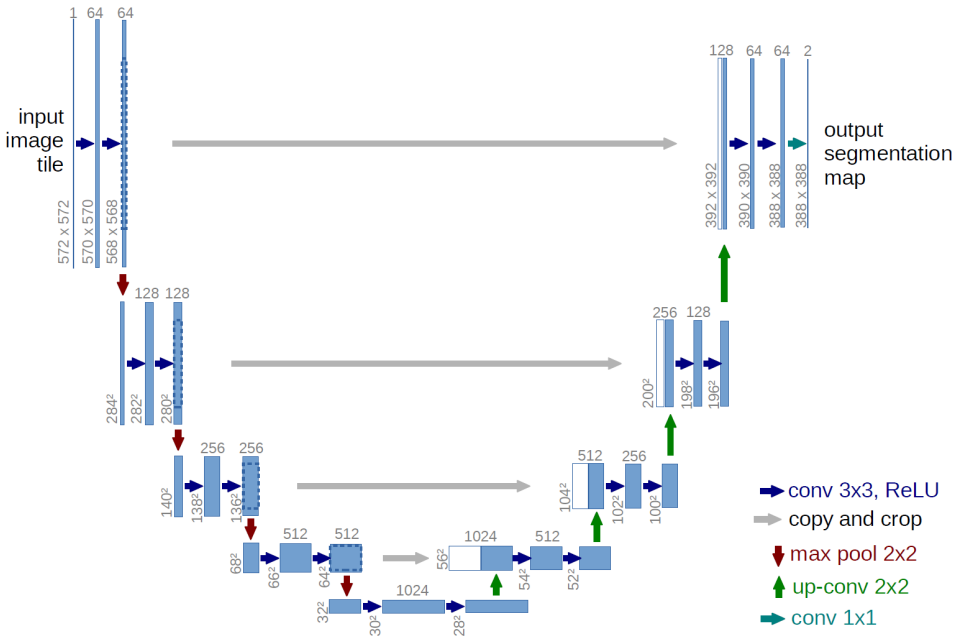


Figure 4.5: U-Net Architecture[1]

4.2.2 ResU-Net

ResU-Net adds residual blocks to the basic U-Net structure which improves gradient flow. By doing so, deeper feature extraction becomes possible. In place of the regular convolutional layers, residual blocks are utilized which help in overcoming the vanishing gradient issues. The encoder uses the U-Net configuration, but it allows for great propagation of features. The decoder also uses skip connections to up-sample and reconstruct segmentation masks. The primary goal of ResU-Net is to improve efficiency in segmentation tasks in complex environments through residual learning. The ResU-Net architecture used for road segmentation is shown in Figure 4.6.

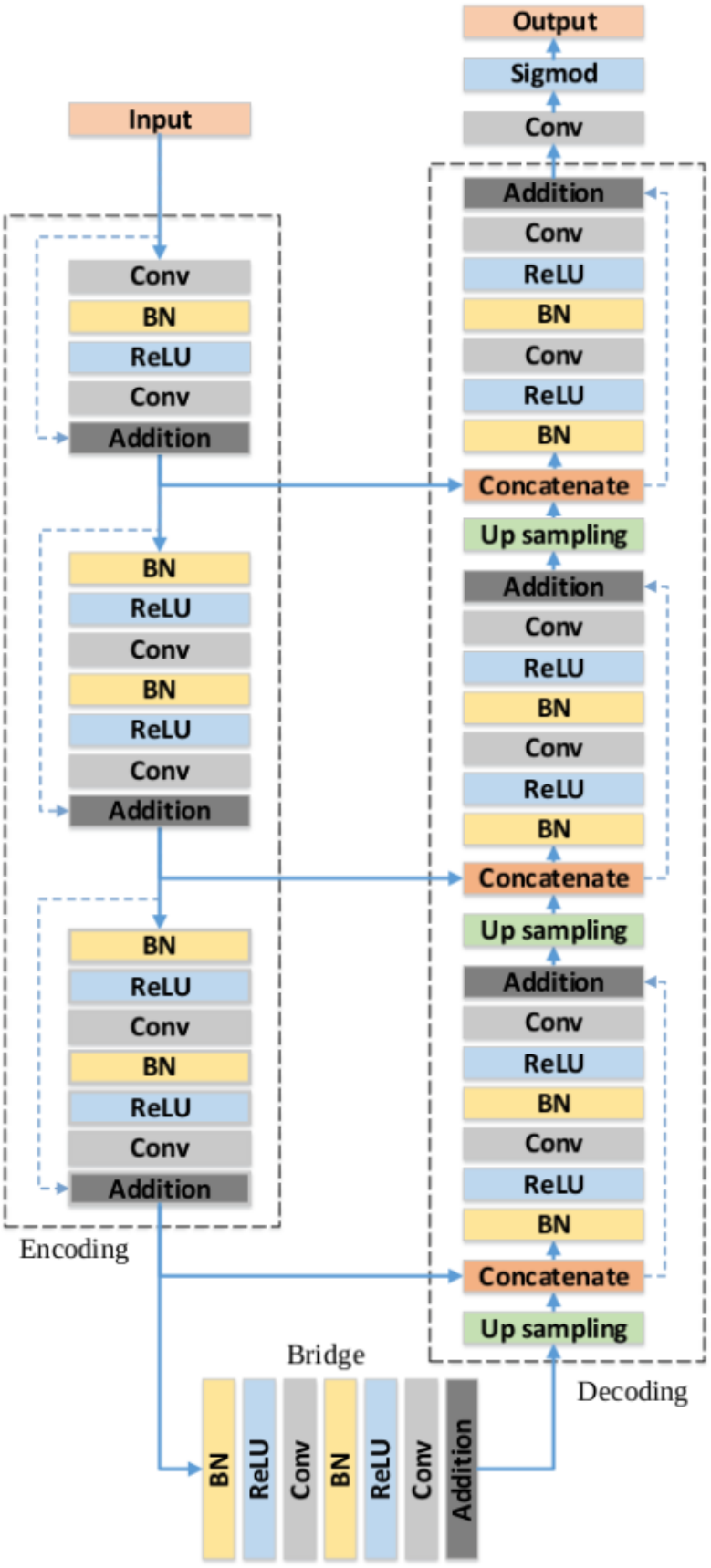


Figure 4.6: ResU-Net Architecture[26]

4.3 Working Plan

The whole process follows a structured workflow consisting of data collection, preprocessing, model training, evaluation, and result analysis. The process begins with collecting diverse road images from various environments to ensure adaptability. The collected images undergo preprocessing, which includes resizing, normalization, class mapping, and data augmentation to enhance model generalization. Then the process is followed by class identification and annotation.

Once the data is prepared, it is split into training, validation, and test sets. The selected models, U-Net and ResU-Net for segmentation and various YOLO versions for object detection, are then trained on the processed dataset, optimizing their performance using suitable hyperparameters.

After training, the models are evaluated using accuracy, IoU, F1-score, recall, and precision for road segmentation, while object detection results are assessed using mAP (mean Average Precision). The trained models are then used to predict segmentation masks for test images, which are compared against ground truth masks to assess performance, while YOLO models predict bounding boxes for road objects, which are evaluated based on precision and recall metrics.

The final stage involves visualizing and analyzing the results, including predicted masks, evaluation metrics, confusion matrices, and inference time analysis. For object detection, mAP scores, bounding box accuracy, and class-wise performance metrics are analyzed. The insights gained from these evaluations help determine the best-performing model for both road segmentation and object detection. Lastly, a visual combination of the best model from Object Detection and Road Segmentation will be integrated. In Figure 4.7, the whole working process has been depicted.

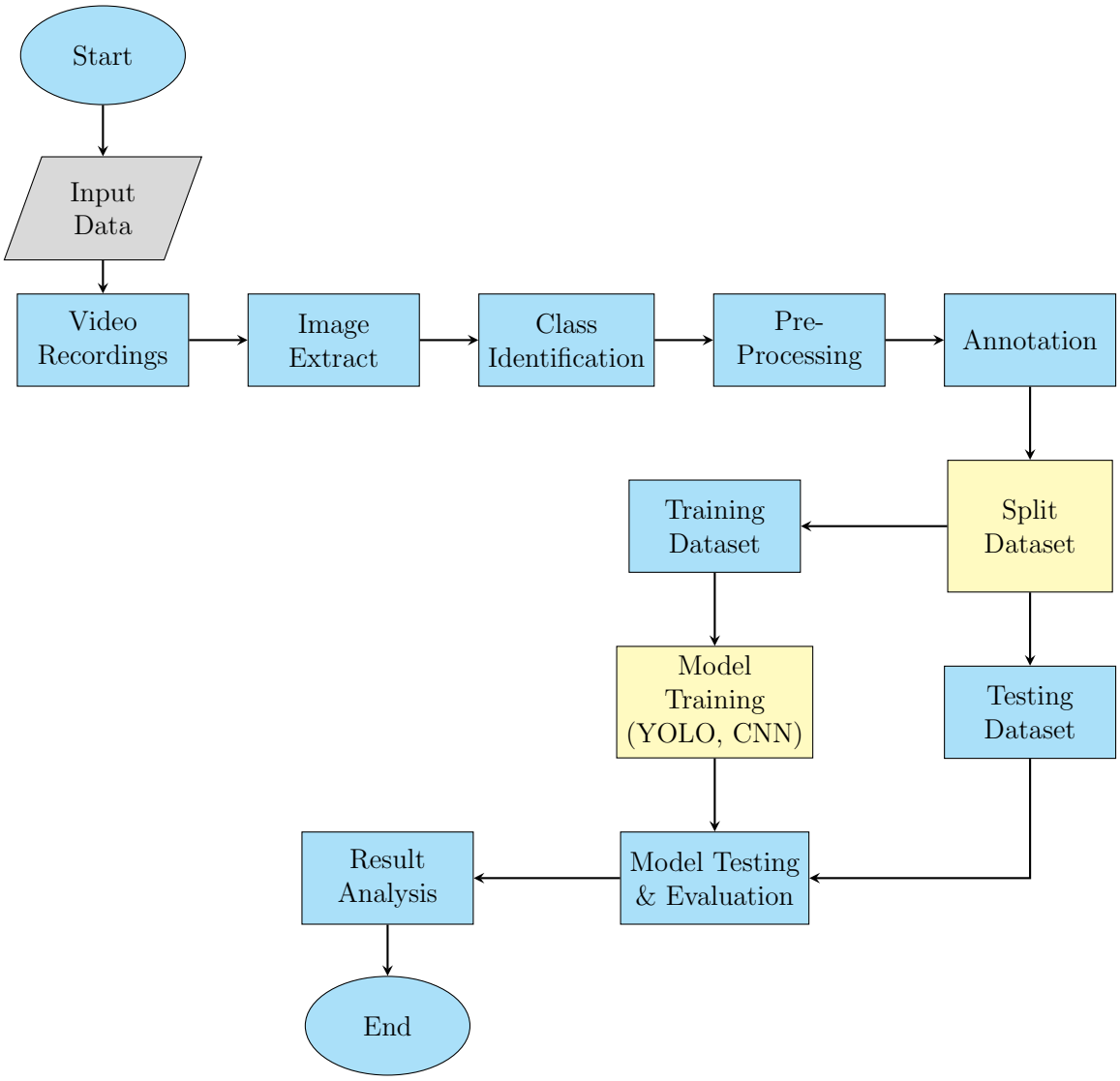


Figure 4.7: Workflow diagram

Chapter 5

Implementation

5.1 Object Detection

This chapter illustrates the implementation process of object detection models used in the study which are: YOLOv8, YOLOv9, YOLOv10, YOLOv11. These were implemented using three different model sizes: Nano, Small and Medium. The dataset contained 2495 numbers of images which were preprocessed and then split into training, testing and validation sets to feed into models. AdamW optimizer was used which was best for our dataset to optimize training performance. Lastly, The results were analyzed and compared using Average precision scores (mAP).

5.1.1 Training Setup and Hyperparameters

Each of these models was trained using the same default hyperparameters so that there is a consistency while comparison. The default hyperparameters used for training are summarized in Table 5.1.

Hyperparameters	**Values**
Image Size	640
Workers	16
Batch	32
Device	0 (NVIDIA RTX 4080)
Epochs	300
Patience	50

Table 5.1: Default Hyperparameters

5.1.2 Optimizer

To optimize the model performances we have used AdamW because according to our data size and number of classifications in annotations this optimizer works best.

- **Regularization:** AdamW provides a better regularization that comprises weight decay correctly which prevents overfitting.
- **Convergence:** AdamW provides an improved convergence by maintaining stability while it accelerates training.

AdamW helped us to achieve higher mAP scores compared to other optimizers such as ADAM or SGD.

5.2 Road Segmentation

For road segmentation, we have used U-Net and ResU-Net for the purpose of generating segmentation masks for road images with the differentiation of three classes: My Way (valid side of the road), Other Way (opposite side of the road), and Non-Drivable Areas. Both models were developed using TensorFlow and Keras with encoder-decoder configurations that used various connectivity types to enhance learning.

The model U-Net contains an encoder and decoder in which the encoder utilizes convolution and max-pooling layers to capture details at various levels while the decoder uses upsampling followed by skip connections to restore the spatial context. The last layer comprises a softmax activated convolutional layer of dimensions 1×1 which permits the classification of pixels. Whereas, ResU-Net augments U-Net by adding residual blocks that improve gradient flow and solve the problem of vanishing gradient. Instead of performing convolutions directly, they enable shortcut mappings together known as residual connections which make it possible to extract deeper features more efficiently.

As shown in Table 5.2, the models were trained with a learning rate of 1×10^{-4} using the Adam optimizer and a categorical cross-entropy loss function because of the multi-class aspect of the task. The models were trained over 100 epochs with a batch size of 8. To prevent overfitting early stopping was implemented based on validation loss and model checkpoint was used to save the best model.

Hyperparameter	**Value**
Learning Rate	1×10^{-4}
Batch Size	8
Number of Epochs	100
Optimizer	Adam
Loss Function	Categorical Cross-Entropy
Activation Function	Softmax
Training Strategy	Early Stopping, Model Checkpointing
Number of Classes	3 (My Way, Other Way, Non-Drivable Area)

Table 5.2: Training Configuration

Chapter 6

Result Analysis

6.1 Object Detection

In the beginning, we evaluated the models based on their mAP (Average Precision). As shown in Table 6.1, these scores help us to understand the model's accuracy.

Model	**Model Size**	**Accuracy**
YOLOv8	Nano	0.753
	Small	0.773
	Medium	0.785
YOLOv9	Nano/Tiny	0.76
	Small	0.763
	Medium	0.782
YOLOv10	Nano	0.746
	Small	0.766
	Medium	0.789
YOLOv11	Nano	0.761
	Small	0.780
	Medium	0.793

Table 6.1: Model Performance

From the Table 6.1 Medium Sized Models gave us the best accuracy overall. The confusion matrices and training/validation loss from all model sizes (Nano, Small, Medium) of different YOLO versions are almost the same. Therefore we will discuss only the medium sized model confusion matrix in the summary section.

6.1.1 YOLOv8 Summary

In the Figure 6.1, the training and validation accuracy/loss plot has been illustrated. Both of the Tables 6.2 and 6.3 , represent the classification report of this model.

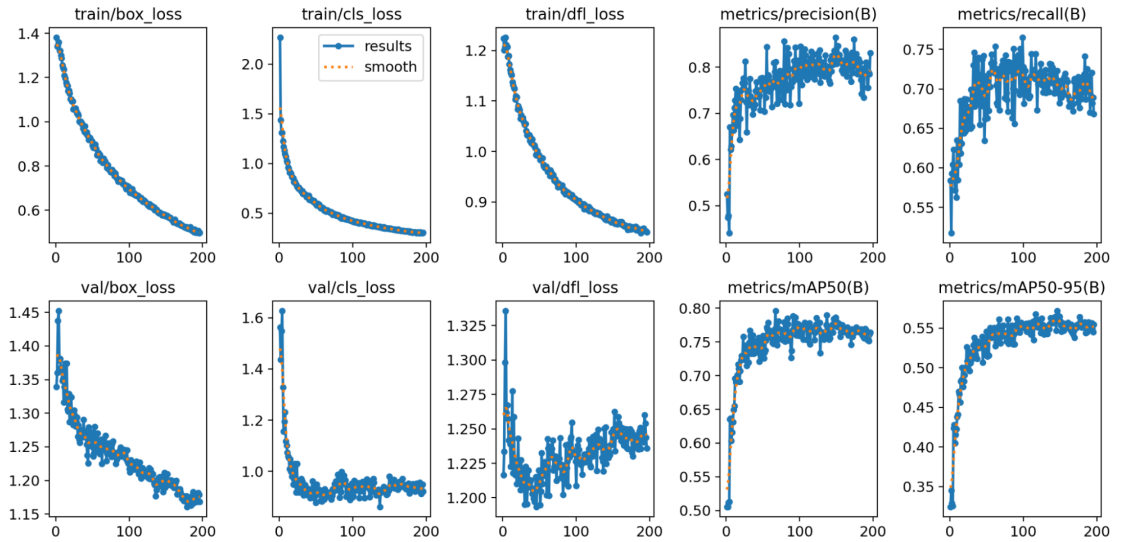


Figure 6.1: YOLOv8m Training and Validation Accuracy and Loss Plot

Class	**mAP50**
All	0.785
Person	0.3330
Rickshaw	0.586
Rickshaw Van	0.485
Auto Rickshaw	0.725
Truck	0.737

Table 6.2: Classification Report 1

Class	**mAP50**
Pickup Truck	0.629
Private Car	0.781
Motorcycle	0.379
Bicycle	0.206
Bus	0.559
Micro Bus	0.754
Covered Van	0.693

Table 6.3: Classification Report (continued)

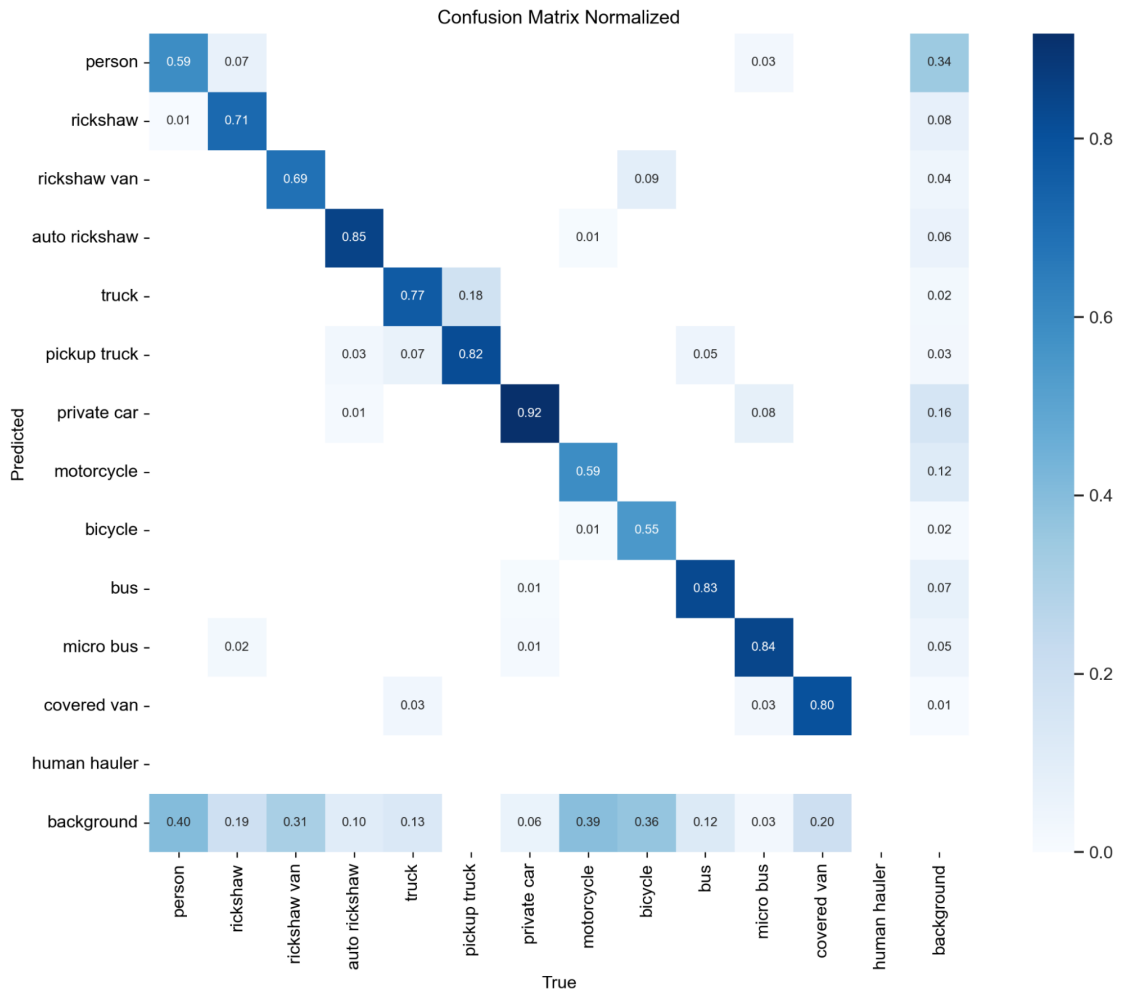


Figure 6.2: Confusion Matrix YOLOv8

Some key significance from the Confusion Matrix are shown in Figure 6.2:

- There is a High TP (True Positives) for Auto Rickshaws (0.85), Private Cars (0.92), Buses (0.83) and Micro buses (0.84). So these classes are well-learned by the model
- Trucks and Pickup Trucks had some noticeable confusion. It is because they have similar features. Trucks were misclassified (0.18) as Pickup Trucks.
- Person class was misclassified (0.34) as background. It is because some long distance objects like persons were not annotated because they were not visible. So the model can infer the persons which are in its radius.

6.1.2 YOLOv9 Summary

In the Figure 6.3, the training and validation accuracy/loss plot has been illustrated. Both of the Tables 6.4 and 6.5 , represent the classification report of this model.

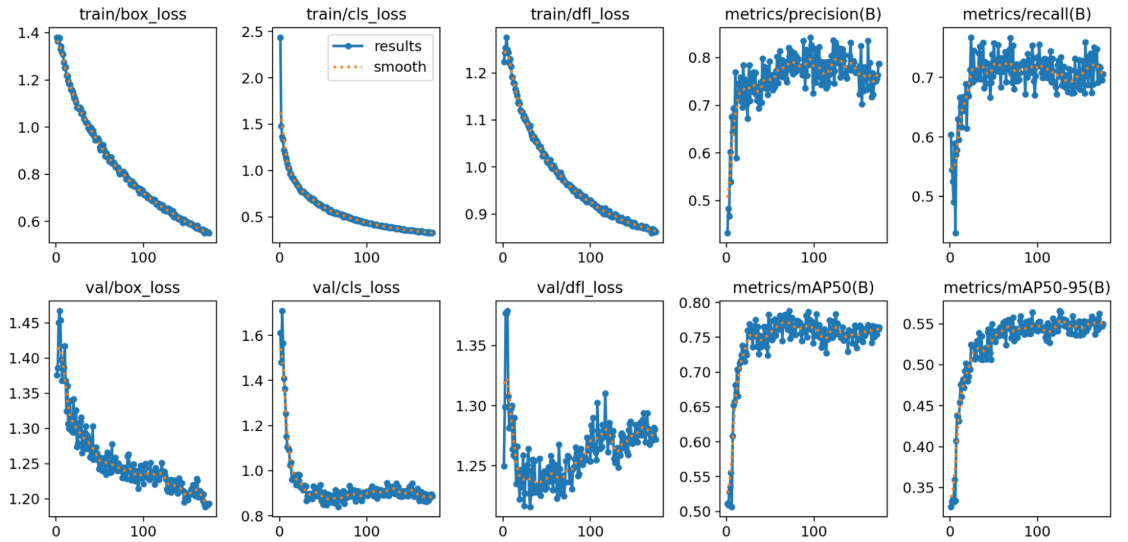


Figure 6.3: YOLOv9m Training and Validation Accuracy and Loss Plot

Class	**mAP50**
All	0.782
Person	0.324
Rickshaw	0.869
Rickshaw Van	0.690
Auto Rickshaw	0.894
Truck	0.855

Table 6.4: Classification Report 1

Class	**mAP50**
Pickup Truck	0.860
Private Car	0.960
Motorcycle	0.658
Bicycle	0.477
Bus	0.771
Micro Bus	0.930
Covered Van	0.813

Table 6.5: Classification Report (continued)

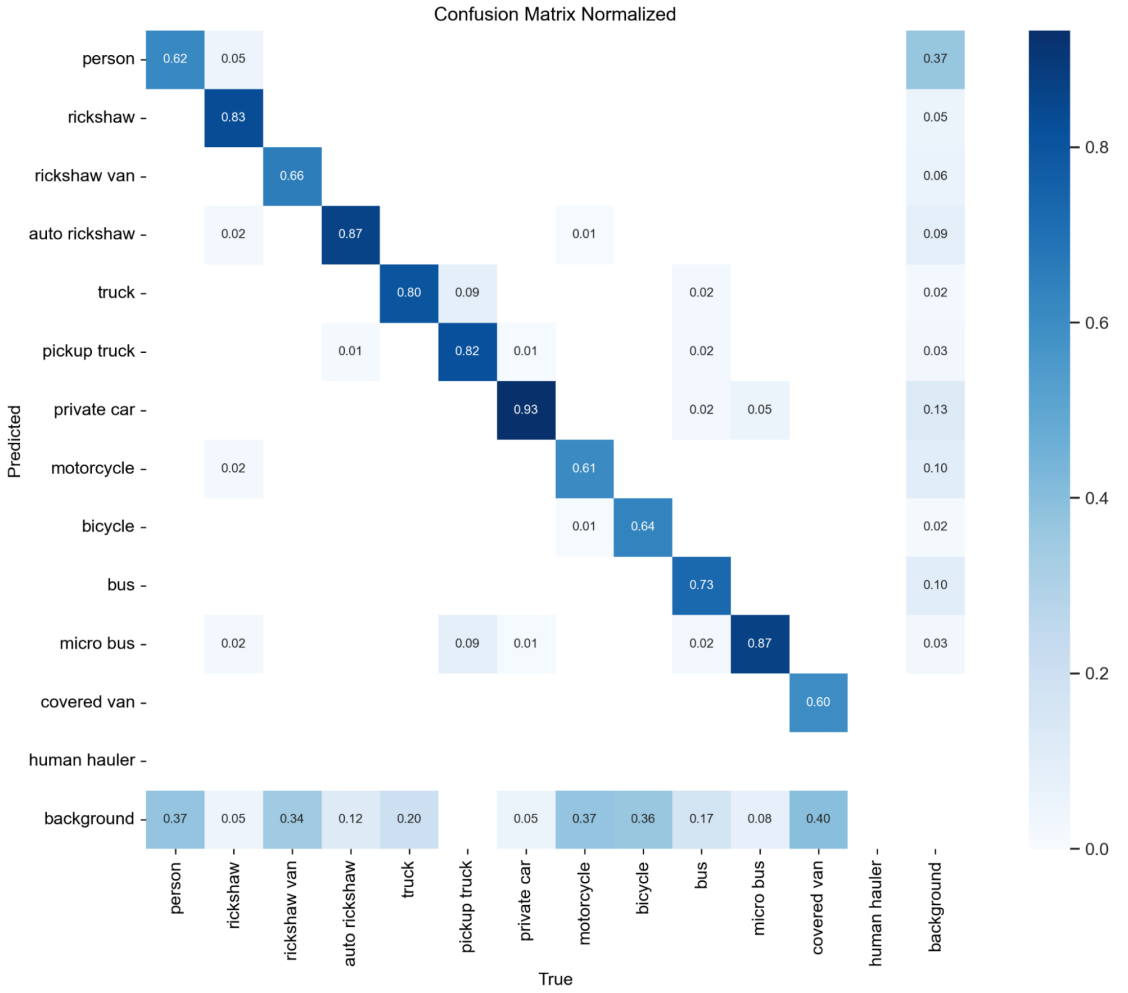


Figure 6.4: Confusion Matrix YOLOv9

Some key significance from the Confusion Matrix are shown in Figure 6.4:

- There is a High TP (True Positives) for Rickshaws (0.83), Auto Rickshaws (0.87), Private Cars (0.93) and Micro buses (0.87). So these classes are well-learned by the model
- Trucks and Pickup Trucks confusion percentage is lower than YOLOv8. Trucks were misclassified (0.09) as Pickup Trucks.
- Person class was misclassified (0.37) as background.

6.1.3 YOLOv10 Summary

In the Figure 6.5, the training and validation accuracy/loss plot has been illustrated. Both of the Tables 6.6 and 6.7, represent the classification report of this model.

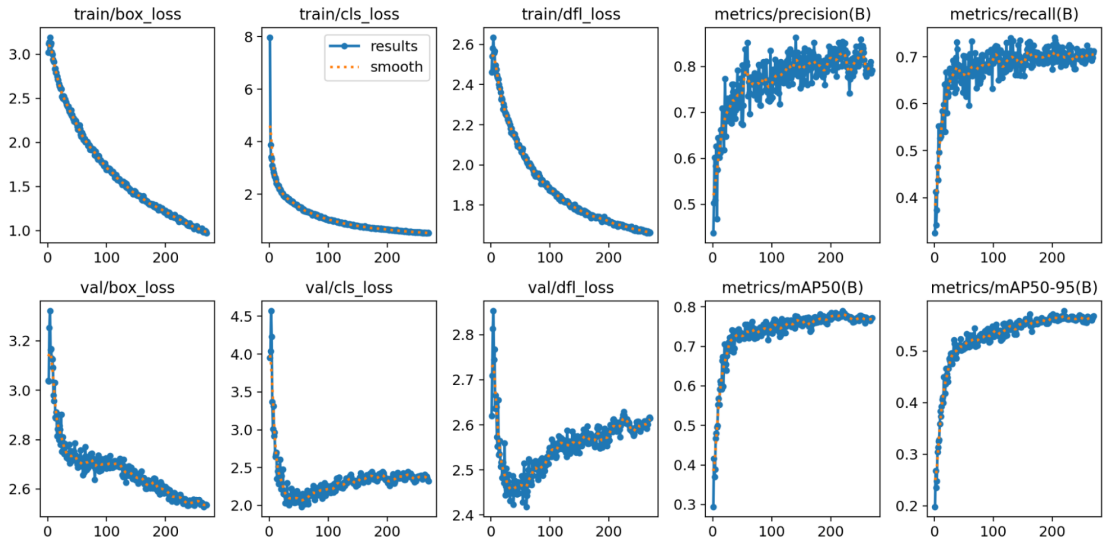


Figure 6.5: YOLOv10m Training and Validation Accuracy and Loss Plot

Class	**mAP50**
All	0.789
Person	0.640
Rickshaw	0.822
Rickshaw Van	0.684
Auto Rickshaw	0.905
Truck	0.880

Table 6.6: Classification Report 1

Class	**mAP50**
Pickup Truck	0.746
Private Car	0.939
Motorcycle	0.658
Bicycle	0.658
Bus	0.797
Micro Bus	0.917
Covered Van	0.825

Table 6.7: Classification Report (continued)

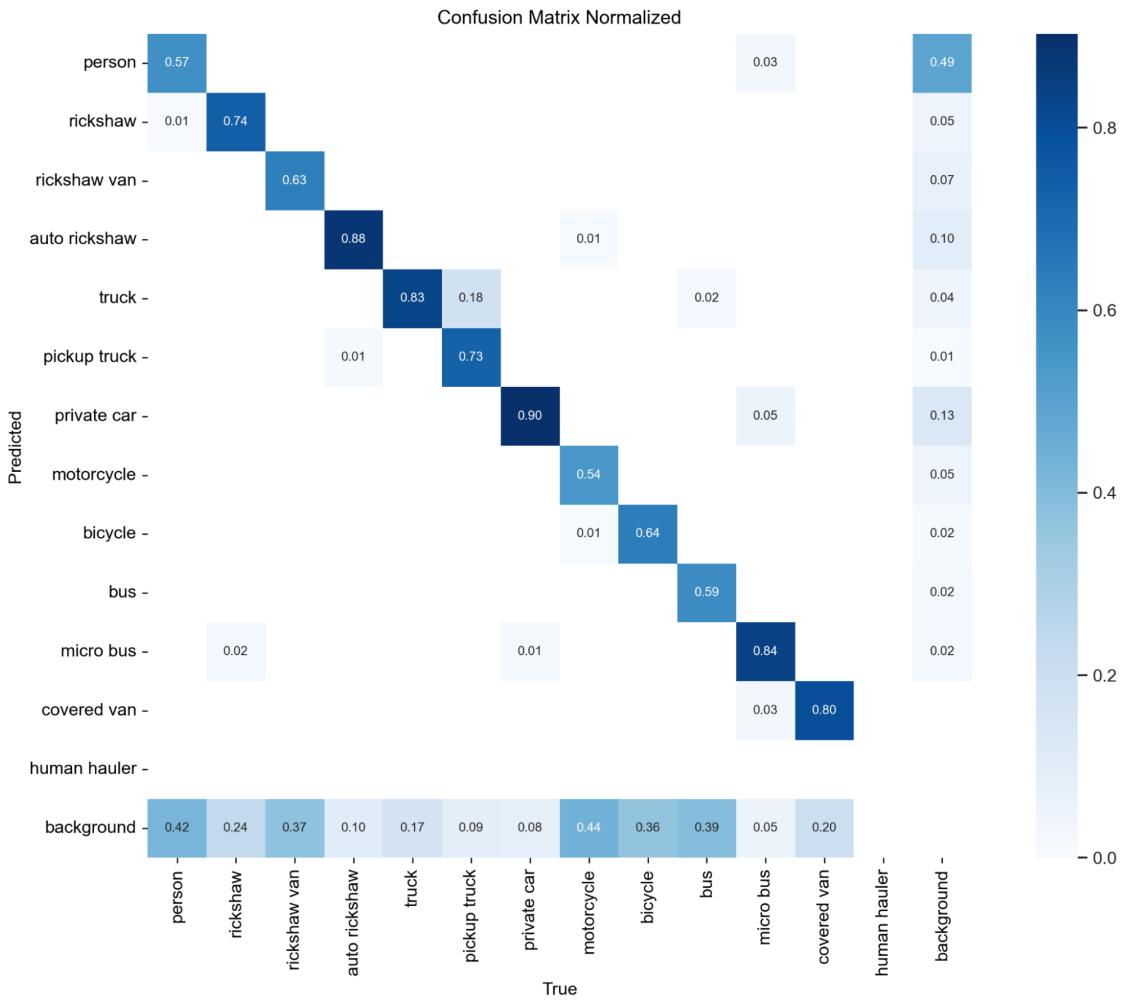


Figure 6.6: Confusion Matrix YOLOv10

Some key significance from the Confusion Matrix are shown in Figure 6.6:

- There is a High TP (True Positives) for Rickshaws (0.74), Auto Rickshaws (0.88), Private Cars (0.90), Buses (0.83), Covered Van (0.90) and Micro buses (0.84). So these classes are well-learned by the model
- Trucks and Pickup Trucks had some noticeable confusion. It is because they have similar features. Trucks were misclassified (0.18) as Pickup Trucks.
- Person class was misclassified (0.49) as background.

6.1.4 YOLO v11 Summary

In the Figure 6.7, the training and validation accuracy/loss plot has been illustrated. Both of the Tables 6.8 and 6.9 , represent the classification report of this model.

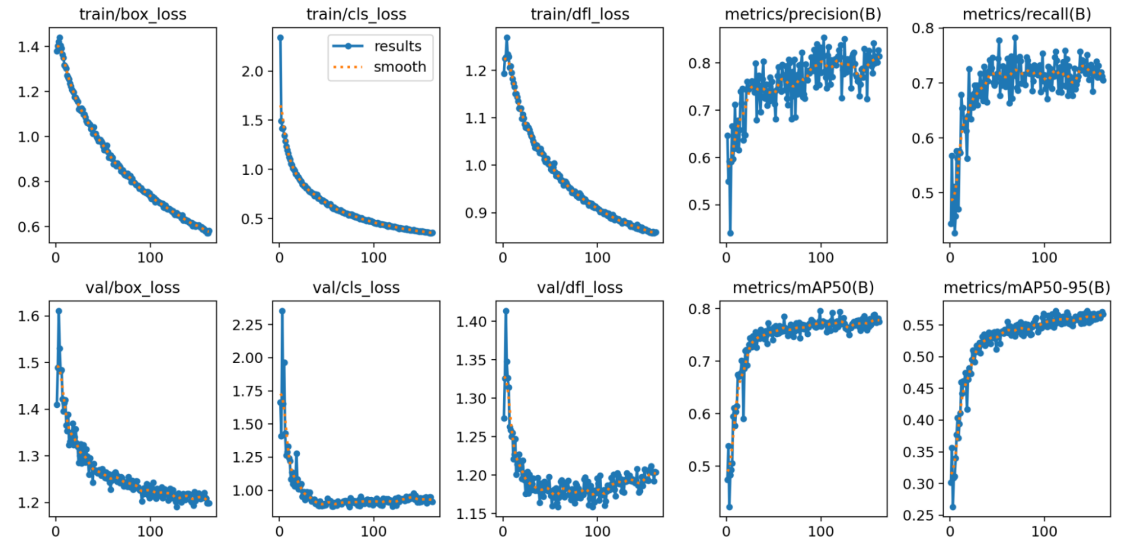


Figure 6.7: YOLOv11m Training and Validation Accuracy and Loss Plot

Class	mAP50
All	0.793
Person	0.636
Rickshaw	0.849
Rickshaw Van	0.636
Auto Rickshaw	0.929
Truck	0.921

Table 6.8: Classification Report 1

Class	mAP50
Pickup Truck	0.787
Private Car	0.942
Motorcycle	0.665
Bicycle	0.598
Bus	0.813
Micro Bus	0.920
Covered Van	0.814

Table 6.9: Classification Report (continued)

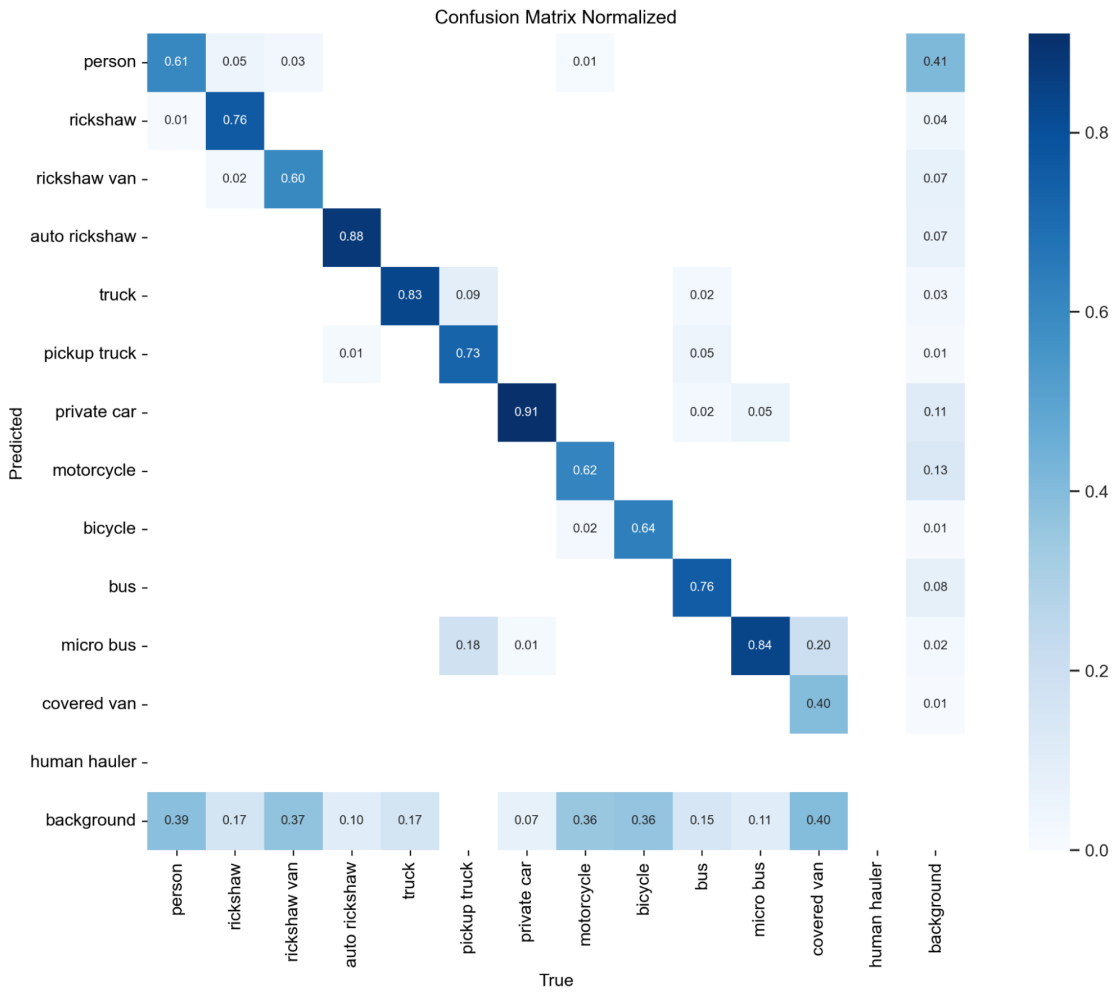


Figure 6.8: Confusion Matrix YOLOv11

Some key significance from the Confusion Matrix are shown in Figure 6.8:

- There is a High TP (True Positives) for Rickshaws (0.76), Auto Rickshaws (0.88), Private Cars (0.91) and Bus (0.76). So these classes are well-learned by the model.
- Micro Buses had some noticeable confusion with Pickup Trucks and Covered Van. It is because they have similar features. Micro Buses were misclassified (0.18) as Pickup Trucks and. Also it was misclassified (0.20) as Covered Van.
- Person class was misclassified (0.41) as background.

6.2 Road segmentation

The models were evaluated based on their performance on new unseen images. This included creating segmentation masks for three different categories of roads. The predictions were once again compared against the ground-truth masks using accuracy, IoU, F1-score, recall, and precision methods. Additionally, Inference time and frames per second (FPS) were measured.

6.2.1 U-Net:

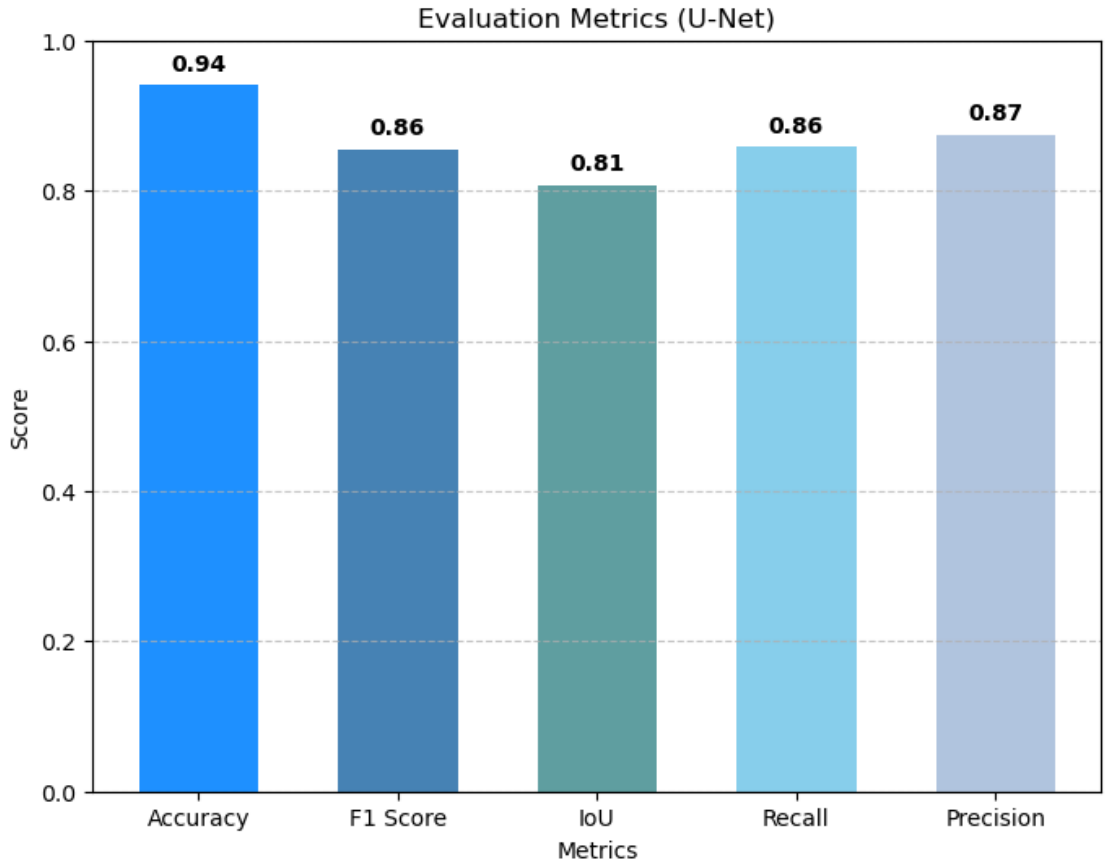


Figure 6.9: U-Net Evaluation Metrics

Figure 6.9 shows the measurement of performance of U-Net in the task of road segmentation. The model exhibited excellent reliability where, accuracy was around 0.94. The F1 score (0.86) is a reliable indicator of precision (0.87) and recall (0.86) and as a result, road region identification is performed efficiently. The IoU score (0.81) suggests a significant amount of overlap was between the predicted mask and ground-truth mask, confirming the segmentation capabilities of the model.

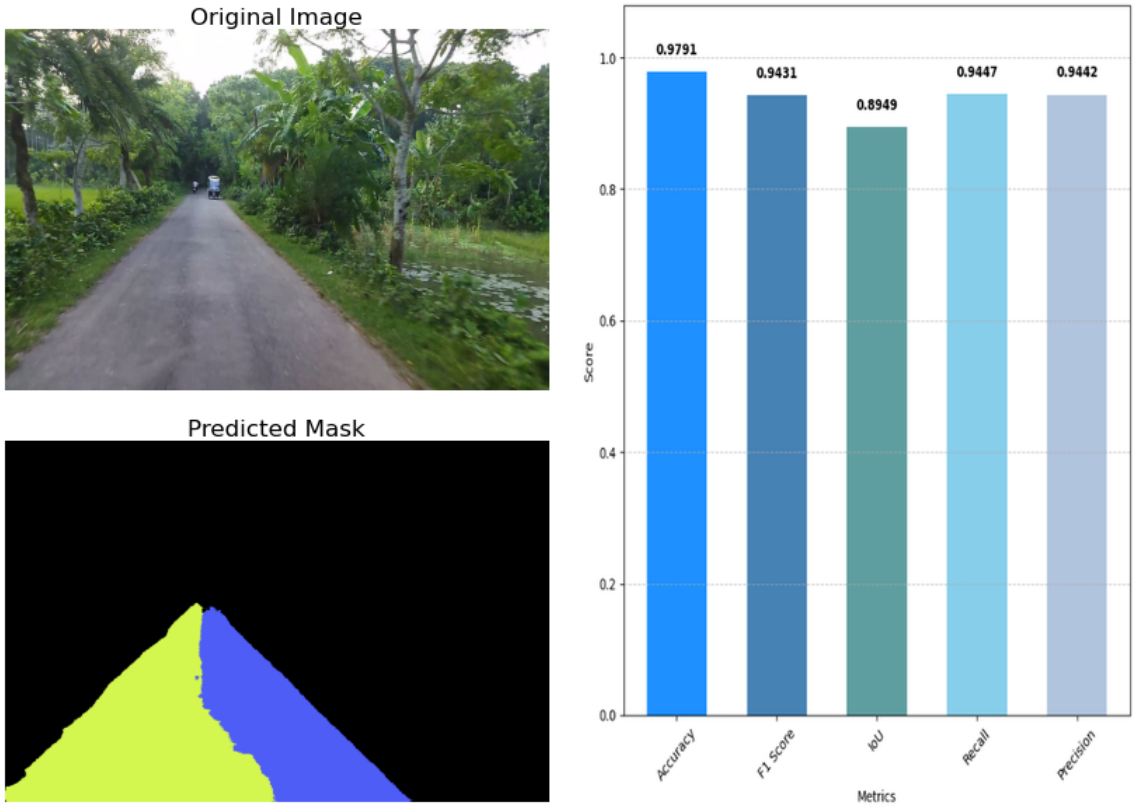


Figure 6.10: Original vs Predicted mask example (U-Net)

Figure 6.10 shows an original road image alongside its predicted segmentation mask using U-Net. For the predicted mask, the model achieved an accuracy score of 0.9791, IoU of 0.9499, F1 score of 0.9431, recall of 0.9447 and precision of 0.9442.

6.2.2 ResU-net

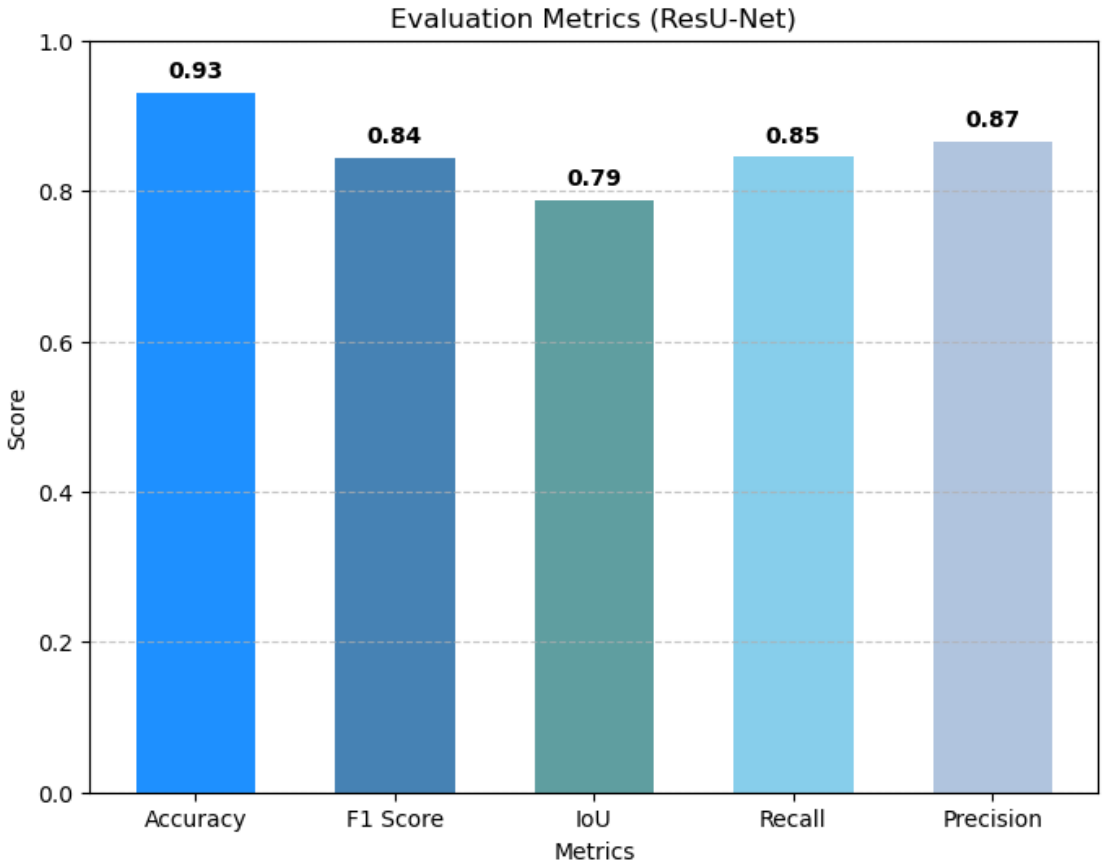


Figure 6.11: ResU-Net Evaluation Metrics

Figure 6.11 presents evaluation metrics for ResU-Net. The model was able to achieve an accuracy score of 0.93 which indicates highly reliable classification at the pixel level. The F1 score (0.84) provides good information about balance between precision (0.87) and recall (0.85), meaning identified road regions will gave lower false positive rates. The IoU score (0.79) is lower compared to U-Net but sufficiently high for the predicted masks and the ground truth masks.

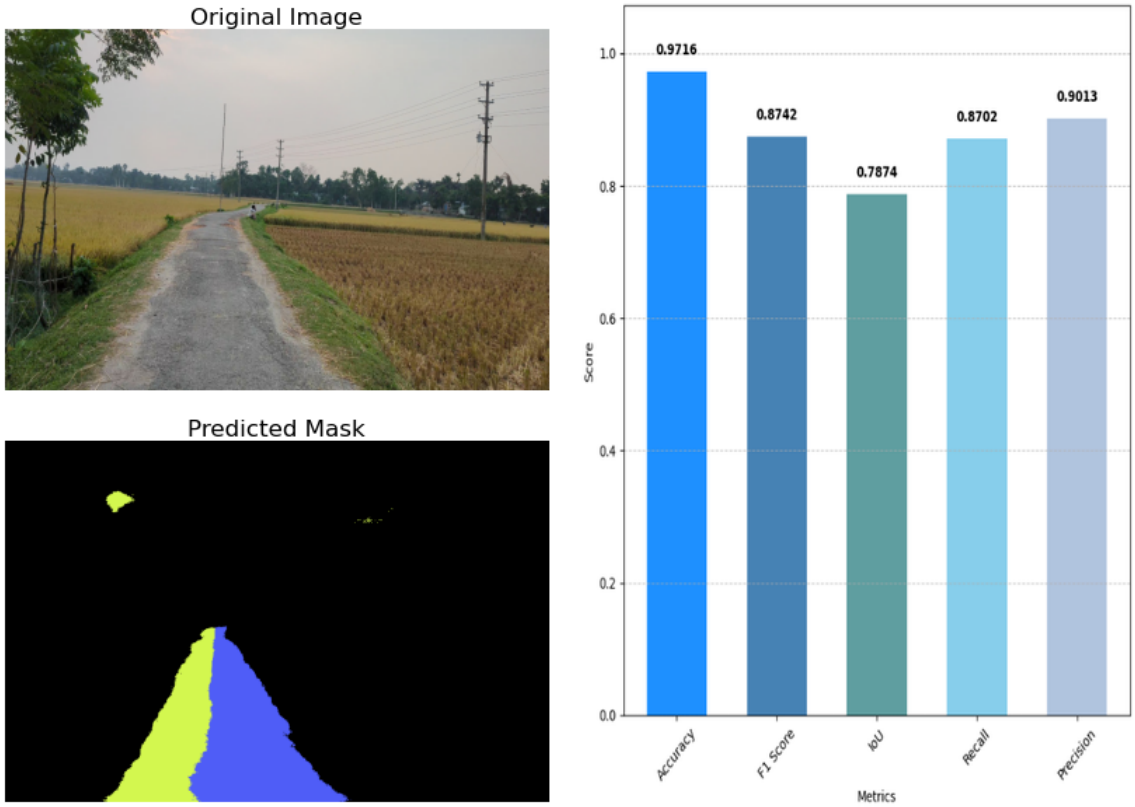


Figure 6.12: Original vs Predicted mask example (ResU-Net)

Figure 6.12 displays an original road image alongside its predicted segmentation mask using ResU-Net. The model's performance on this image achieved an accuracy of 0.9716, F1 score of 0.8742, IoU of 0.8784, recall of 0.8702 and precision of 0.9013.

6.3 Analysis and Discussion

6.3.1 Object Detection

Based on the results, accuracy scores and confusion matrices, YOLOv11 has given us the best results. YOLOv11 handled misclassifications better than other YOLO versions. Specifically, the Medium Sized version (YOLOv11m) achieved the highest accuracy. It has provided the best balance between feature learning, misclassification reduction and better accuracy in all classes.

Additionally, the model size of YOLO versions that is suitable for our road scenario depends on various factors such as: Inference speed, Accuracy, Real time-constraints and Hardware Resources. The Nano sized model has better inference speed but lower accuracy. The Small sized model strikes as a perfect balance for speed and accuracy. However, both of them have lower accuracy than the Medium Sized model. As for road scenarios, one important factor we need to consider is road safety where accuracy plays an important role. Therefore, if you have high-performance devices (NVIDIA RTX 4080 etc.), it is best to use the Medium Sized YOLOv11 model.

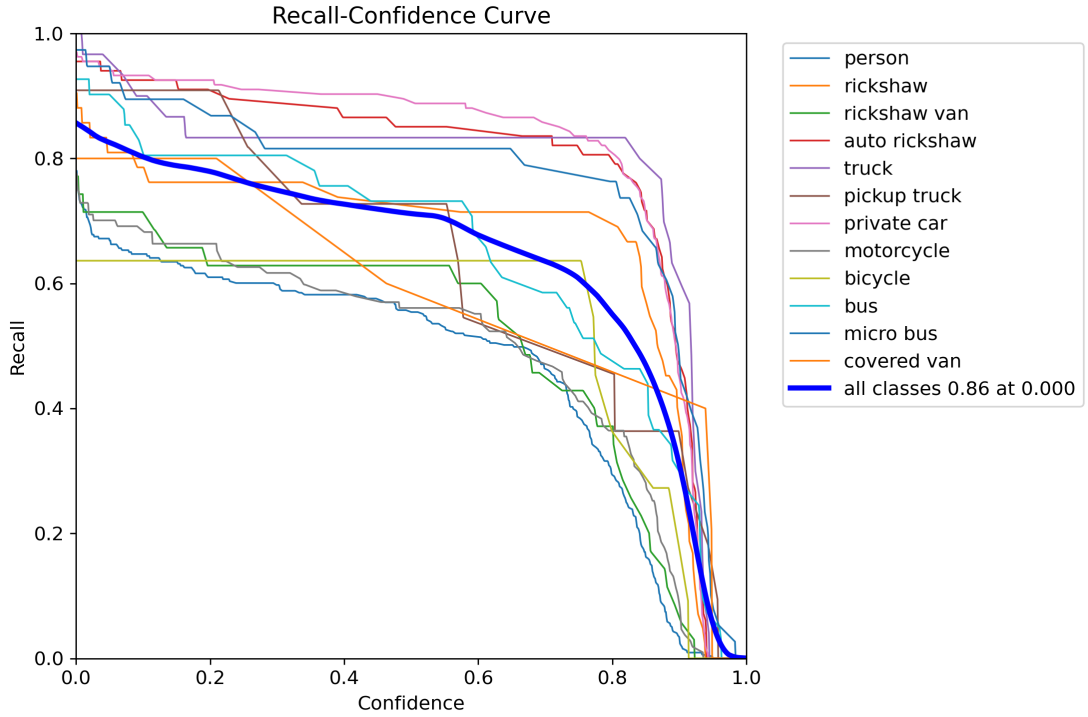


Figure 6.13: Recall-Confidence curve

Furthermore, in Figure 6.13, we have depicted the Recall-Confidence curve from the best working YOLO model. It is relatively higher with a percentage of 86%, than all other versions. This indicates that our recall value is strong and it successfully detects most of the objects. Moreover, the best model maintains a satisfactory recall at a fair confidence level. From this recall, we can break down the classes individually to ensure further improvements.

6.3.2 Road Segmentation

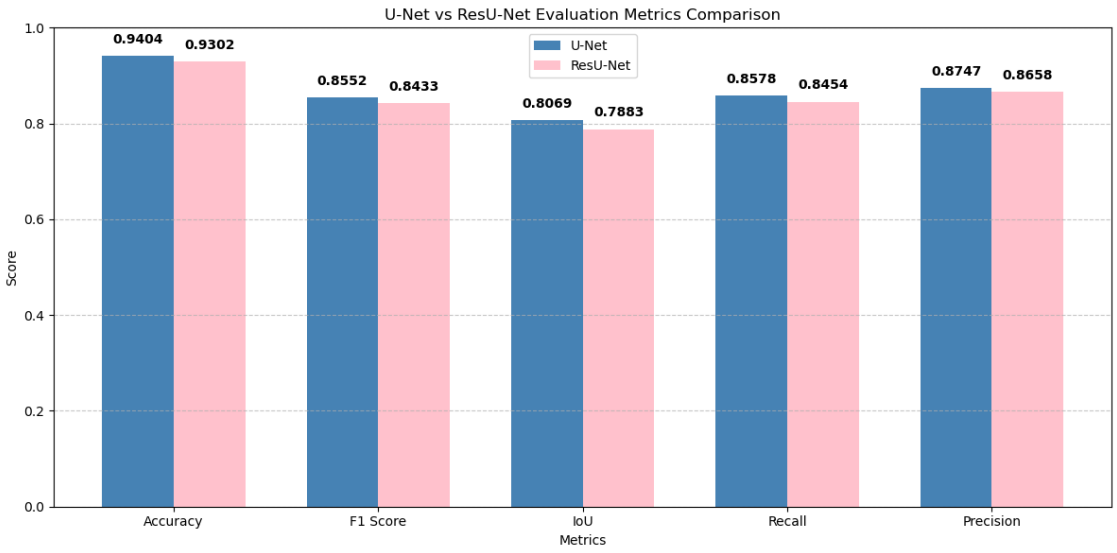


Figure 6.14: U-Net vs ResU-Net evaluation metrics

From the analysis of the results of U-Net and ResU-Net on road segmentation, it can be observed that U-Net scored somewhat better in almost all evaluation metrics (see Figure 6.14). For instance, U-Net attained an accuracy of 0.9404, whereas ResU-Net's accuracy score was 0.9302. Hence, U-Net had a slightly higher accuracy as compared to ResU-Net. The F1-Score for U-Net is also slightly higher at 0.8552 compared to ResU-Net's 0.8433 which indicates that United Net has slightly better precision and recall. As for the Intersection over Union (IoU), U-Net scored 0.8069, beating ResU-Net's score of 0.7883, which means that U-Net has better correlation with the ground-truth compared to ResU-Net. The recall values are close, with U-Net scoring 0.8578 in comparison to ResU-Net's score of 0.8454, which is more than 0.81. Both systems have amazing performance for detecting road pixel. Similarly, Precision fared the same as U-Net recording 0.8747 and ResU-Net achieving 0.8658. All in all, while both models performed very well, U-Net continued to score higher which is why this model was selected for road segmentation in this study out of the two models researched.

6.4 Merged Overview

The merging is carried out by combining the best-performing object detection model and road segmentation model which are Yolo 11m and U-Net in our case. YOLO detects objects, drawing bounding boxes around them, while the U-Net model classifies road areas figuratively into 'My Way', 'Other Way', and 'Non-Drivable Area'. The merged results are consolidated from both object detection and semantic segmentation, as shown in Figure 6.15, where the two techniques are integrated to provide a comprehensive view.

(a) Merged example 1 (b) Merged example 2

Figure 6.15: Merged examples

Chapter 7

Website Deployment

Using the Next.js framework, we developed a web-based interface to integrate the best object detection and road segmentation machine learning models. Using the developed web interface, any user can input an image and generate three outputs: an object detection, a road segmentation output, and a merged visualization output. This implementation allows users to view the results of object detection and road segmentation in real time within a single platform. Figures 7.1 and 7.2 represent the landing page and result page of the website respectively.

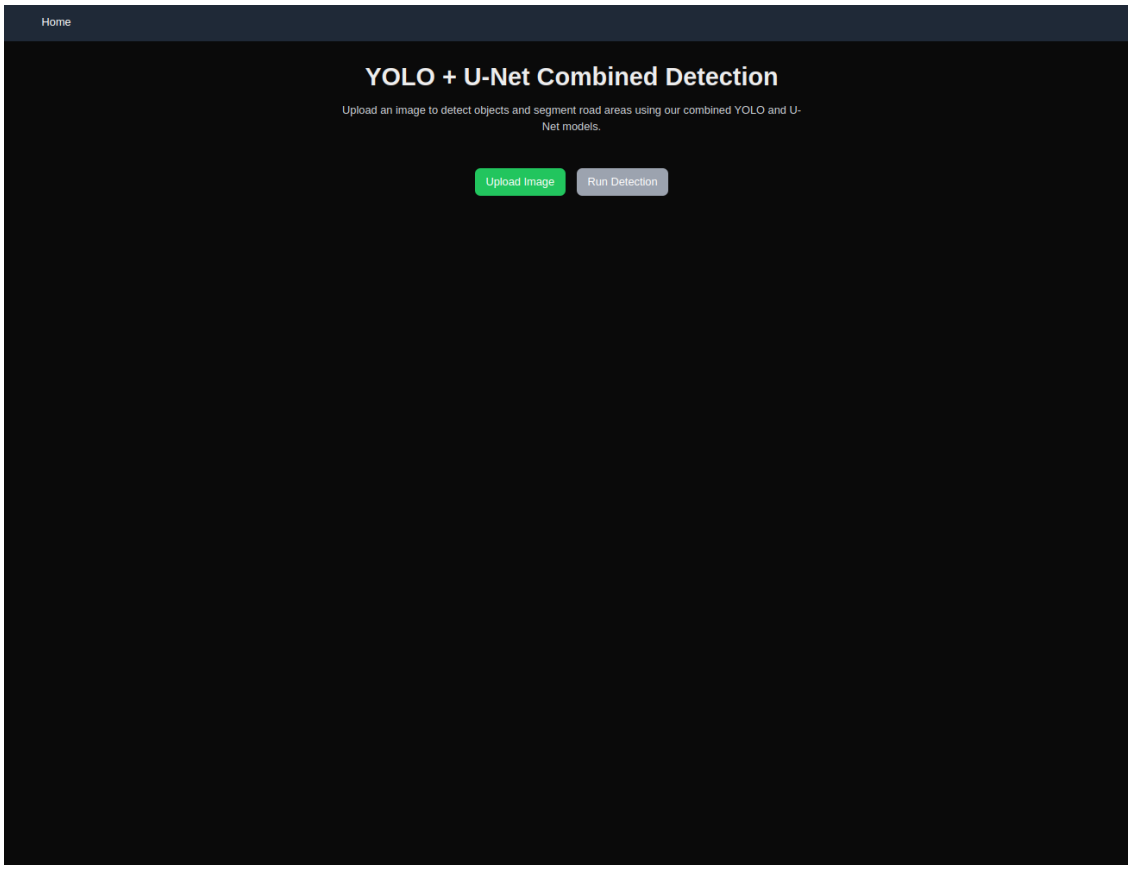


Figure 7.1: Web landing interface

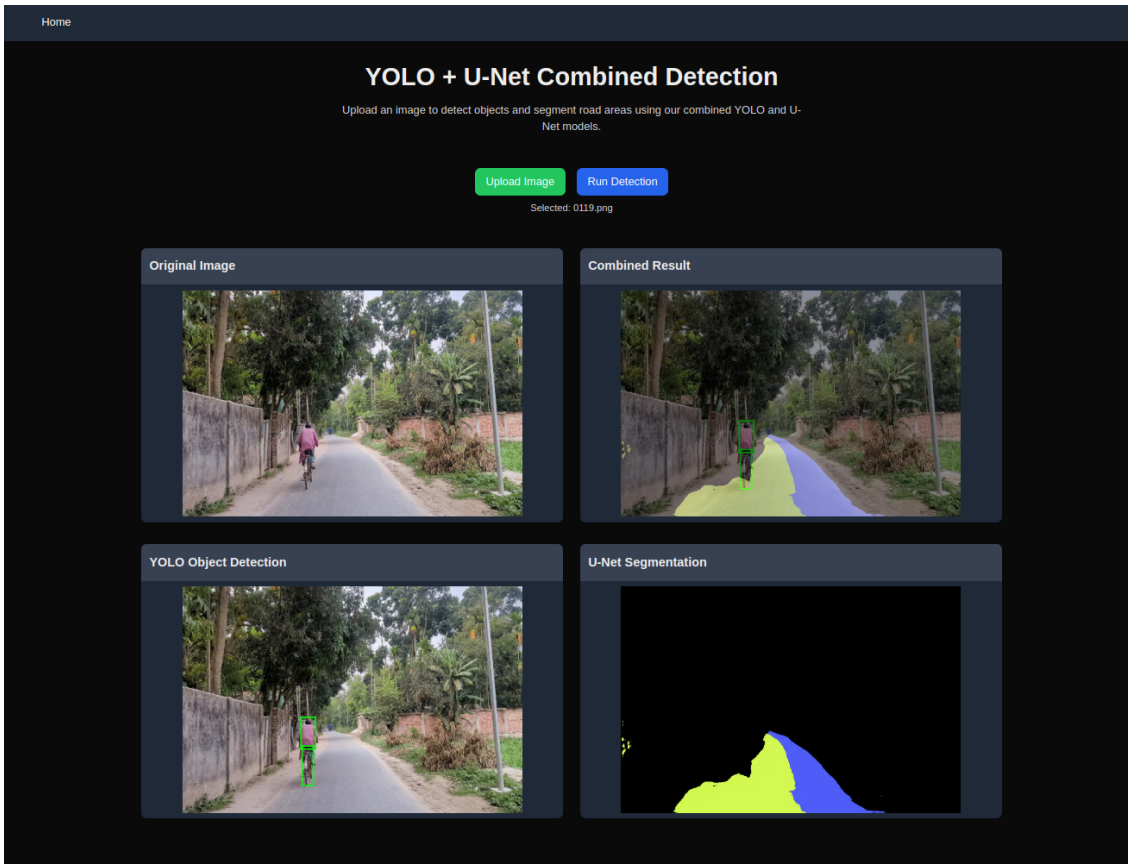


Figure 7.2: Web result interface

7.1 Limitations

The web platform is only available locally, and so this limits access to a wider audience. The platform has no cloud server, which means people can only access it from a local area network. Further, these results are not saved permanently—processed images are lost if the page is refreshed. Future changes could be made by deploying the system on a cloud platform for internet access, as well as providing an option to save the results.

Chapter 8

Conclusion

8.1 Conclusion

In conclusion, we have successfully developed an optimized object detection and road segmentation model for autonomous driving in Bangladesh. This model works not only in good weather conditions but also in adverse weather conditions such as: Rainy and Foggy weather. Our research on these models have shown us that YOLOv11m provides high overall accuracy alongside best balance with speed. This makes it the most suitable and optimized model for road object detection in Bangladesh. Especially when road safety is an important factor in autonomous driving we can use this model with hard performance hardware. In addition to this, U-net effectively outperformed ResU-Net in road segmentation by exhibiting higher overall accuracy across various evaluation metrics that has been used in this study. From this research our dataset has been proven to be invaluable for these models to work which can be used for further enhancements regarding similar studies. Lastly, our research might help us greatly in fighting road accidents by implementing these models in autonomous driving as these robust models are tailored for real-world driving scenarios.

8.2 Future Work

In order to improve the performance metrics of our models, future work will be centered around augmenting the real-time inference speed and precision. This includes:

- Optimization of YOLOv11m for lower-end hardware
- Studying the model quantization process
- Merging sensor fusion with LiDAR depth information
- Wider and local road samples for Road Segmentation

Moreover, augmenting the datasets with more extreme weather conditions and more intricate ensemble urban settings will help in fostering more generalization. Finally, the prototype of the autonomous vehicle will serve as a platform for a real-life testing phase in order to confirm the model's efficiency in real driving scenarios.

Bibliography

[1] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, Accessed: 2025-02-06, 2015. [Online]. Available: https://lmb.informatik.uni-freiburg.de/Publications/2015/RFB15a/.

[2] J. Liu, Q. Qin, J. Li, and Y. Li, "Rural road extraction from high-resolution remote sensing images based on geometric feature inference," *ISPRS International Journal of Geo-Information*, vol. 6, no. 10, p. 314, 2017. [Online]. Available: https://www.mdpi.com/2220-9964/6/10/314.

[3] S. M. Grigorescu, B. Trasnea, T. Cocias, and G. Macesanu, "A survey of deep learning techniques for autonomous driving," *ResearchGate*, 2019. [Online]. Available: https://www.researchgate.net/publication/337264008_A_survey_of_deep_learning_techniques_for_autonomous_driving.

[4] A. Saha, "Rural road extraction using object-based image analysis (obia): A case study from assam, india," 2019. [Online]. Available: https://www.researchgate.net/publication/334217077_Rural_Road_Extraction_using_Object_Based_Image_Analysis_OBIA_A_case_study_from_Assam_India.

[5] Y. Liu, T. Qiu, J. Wang, and W. Qi, "A nighttime vehicle detection method with attentive gan for accurate classification and regression," *Entropy*, vol. 23, no. 11, p. 1490, 2021. [Online]. Available: https://www.mdpi.com/1099-4300/23/11/1490.

[6] M. Mohammed and S. Shidlovskiy, "Computer vision with semantic segmentation for autonomous ground vehicle," 2021. [Online]. Available: https://vital.lib.tsu.ru/vital/access/services/Download/koha:000846376/SOURCE1.

[7] Y. Park, L. M. Dang, S. Lee, D. Han, and H. Moon, "Multiple object tracking in deep learning approaches: A survey," *MDPI Journal*, vol. 10, no. 19, p. 2406, 2021. [Online]. Available: https://www.mdpi.com/2079-9292/10/19/2406.

[8] S. Cakir, M. Gauß, K. Häppeler, and Y. Ounajjar, "Semantic segmentation for autonomous driving: Model evaluation, dataset generation, perspective comparison, and real-time capability," 2022. [Online]. Available: https://www.researchgate.net/publication/362276836_Semantic_Segmentation_for_Autonomous_Driving_Model_Evaluation_Dataset_Generation_Perspective_Comparison_and_Real-Time_Capability.

[9] H. Li, C. Liu, and A. Basu, "Semantic segmentation based on depth background blur," *MDPI Journal*, vol. 12, no. 3, p. 1051, 2022. [Online]. Available: https://www.mdpi.com/2076-3417/12/3/1051.

[10] Sabiha Akter Seema. "Road accident: A major concern of bangladesh." (2022), [Online]. Available: https://cgs-bd.com/article/9009/Road-Accident--A-Major-Concern-of-Bangladesh.

[11] P. Bhatt, D. Makwana, A. Joshi, and K. Raol, "Object extraction and detection using u2-net and yolov7," 2023. [Online]. Available: https://www.researchgate.net/publication/377020966_Object_Extraction_and_Detection_UsingU2-Net_and_YOLOv7.

[12] P. S. Chary, "Real-time object detection using yolov4," 2023. [Online]. Available: https://www.researchgate.net/publication/376893987_Real_Time_Object_Detection_Using_YOLOv4.

[13] Hidetomo Sakaino. "Proceedings of the IEEE conference on computer vision and pattern recognition workshops." (2023), [Online]. Available: https://www.computer.org/csdl/proceedings-article/cvprw/2023/024900d591/1PBxyR1TjZS.

[14] Joe A. King, Jr. "What percentage of car accidents is caused by human error?" (2023), [Online]. Available: https://www.mkhlawyers.com/blog/what-percentage-of-car-accidents-is-caused-by-human-error/.

[15] K. KALRA, "Image segmentation a comprehensive guide," 2023. [Online]. Available: https://pub.aimind.so/image-segmentation-f988e847af8c.

[16] G. Lavanya and S. D. Pande, "Enhancing real-time object detection with yolo algorithm," *ResearchGate*, 2023. [Online]. Available: https://www.researchgate.net/publication/376252973_Enhancing_Real-time_Object_Detection_with_YOLO_Algorithm.

[17] Martin Placek. "Autonomous vehicle technology." (2023), [Online]. Available: https://www.statista.com/topics/3573/autonomous-vehicle-technology/#topicOverview.

[18] X. Meng, Y. Liu, L. Fan, and J. Fan, "Yolov5s-fog: An improved model based on yolov5s for object detection in foggy weather scenarios," *Sensors*, vol. 23, no. 11, p. 5321, 2023. [Online]. Available: https://www.mdpi.com/1424/-8220/23/11/5321#:~:text=Compared%20to%20the%20baseline%20model,weather%2C%20for%20autonomous%20driving%20vehicles..

[19] S. Pandey, K.-F. Chen, and E. B. Dam, "Comprehensive multimodal segmentation in medical imaging: Combining yolov8 with sam and hq-sam models," in *Proceedings of ICCV Workshop*, 2023. [Online]. Available: https://openaccess.thecvf.com/content/ICCV2023W/CVAMD/papers/Pandey_Comprehensive_Multimodal_Segmentation_in_Medical_Imaging_Combining_YOLOv8_with_SAM_ICCVW_2023_paper.pdf.

[20] S. Rauch. "Ai in the automotive industry: A 2024 outlook." (2023), [Online]. Available: https://www.simplilearn.com/ai-in-automative-article.

[21] A. H. Sadhin, S. Z. M. Hashim, H. Samma, and N. Khamis, "Yolo: A competitive analysis of modern object detection algorithms for road defects detection using drone images," *ResearchGate*, 2023. [Online]. Available: https://www.researchgate.net/publication/375761847_YOLO_A_Competitive_Analysis_of_Modern_Object_Detection_Algorithms_for_Road_Defects_Detection_Using_Drone_Images.

[22] T. A. Victoire, D. A.karunamurthy, S. Sathish, and R. Sriram, "Ai-based self-driving car," *ResearchGate*, 2023. [Online]. Available: https://www.researchgate.net/publication/372960513_AI-based_Self-Driving_Car.

[23] d4r6j, *Yolov10: Model review*, Accessed: 2025-02-06, 2024. [Online]. Available: https://velog.io/@d4r6j/YOLOv10-1.-Model-Review.

[24] D. P. Hidayatullah and R. Tubagus, *Yolov9 architecture: Everything you need to know*, Accessed: 2025-02-06, 2024. [Online]. Available: https://article.stunningvisionai.com/yolov9-architecture.

[25] S. N. Rao, *Yolov11 explained: Next-level object detection with enhanced speed and accuracy*, Accessed: 2025-02-06, 2024. [Online]. Available: https://medium.com/@nikhil-rao-20/yolov11-explained-next-level-object-detection-with-enhanced-speed-and-accuracy-2dbe2d376f71.

[26] N. K. Tomar, *What is resunet?* Accessed: 2025-02-06, 2024. [Online]. Available: https://idiotdeveloper.com/what-is-resunet/.

[27] "Yolov8 network architecture diagram." (), [Online]. Available: https://www.researchgate.net/figure/YOLOv8-Network-Architecture-Diagram_fig5_381803486.