OBJECT DETECTION ON ROAD USING DEEP LEARNING APPROACH

Submitted by

Zahid Hasan

192-35-2873

Department of Software Engineering

Daffodil International University

Supervised by

Md. Shohel Arman

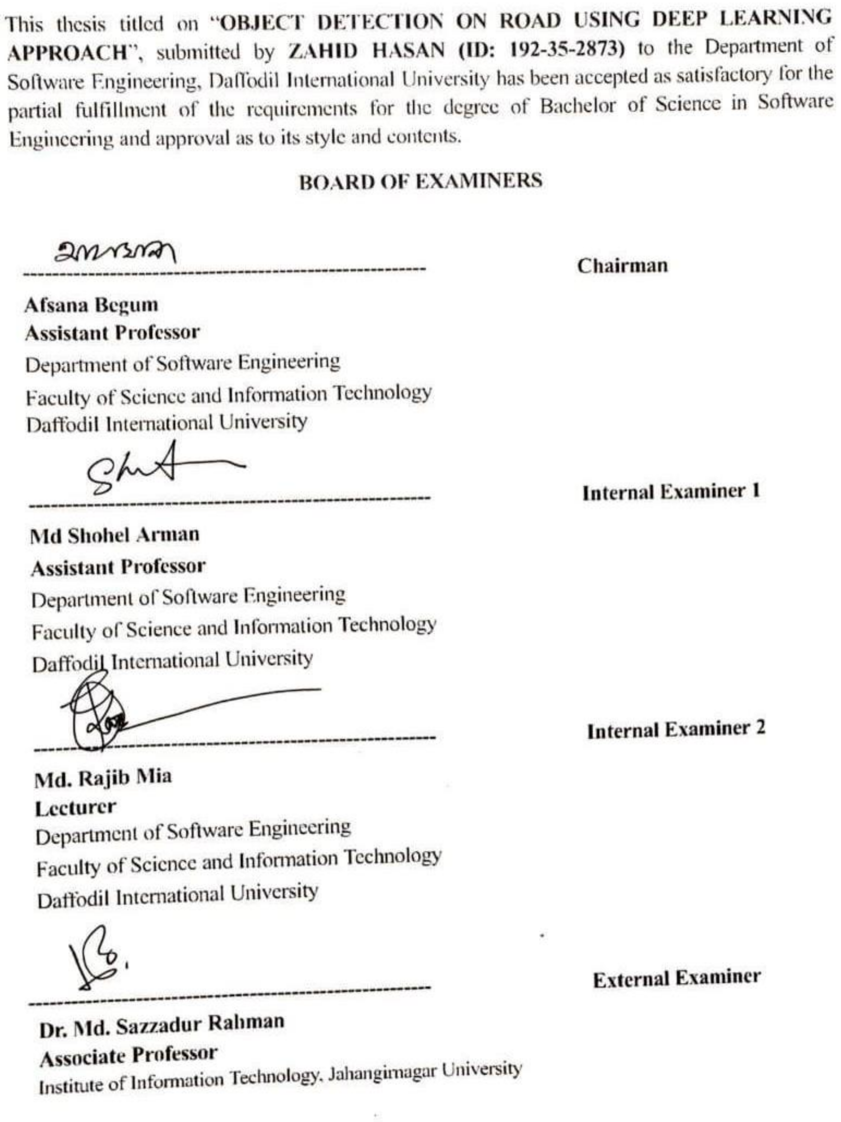
Assistant Professor

Department of Software Engineering

Daffodil International University

This Thesis report has been submitted in fulfillment of the requirements for the Degree of Bachelor of Science in Software Engineering

APPROVAL



This thesis titled on "**OBJECT DETECTION ON ROAD USING DEEP LEARNING APPROACH**", submitted by **ZAHID HASAN (ID: 192-35-2873)** to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Software Engineering and approval as to its style and contents.

BOARD OF EXAMINERS

-- **Chairman**

Afsana Begum
Assistant Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

-- **Internal Examiner 1**

Md Shohel Arman
Assistant Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

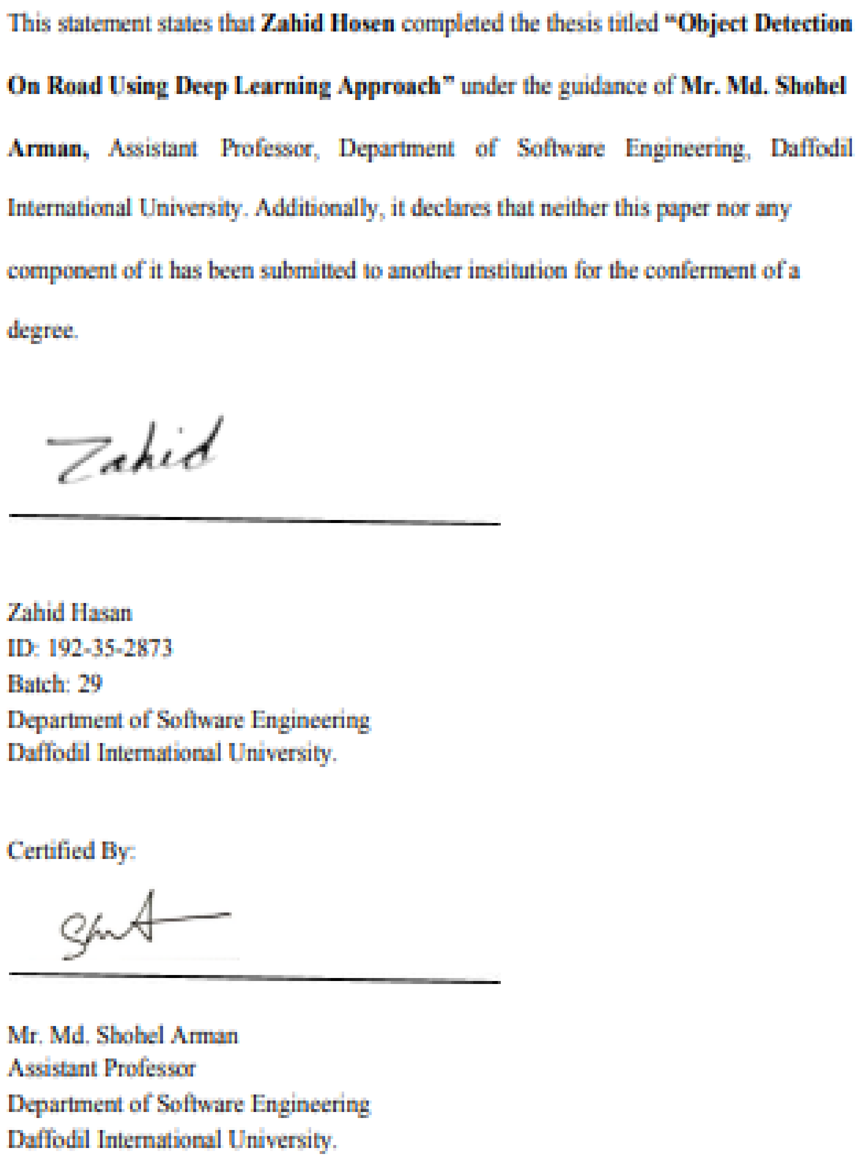
-- **Internal Examiner 2**

Md. Rajib Mia
Lecturer
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

-- **External Examiner**

Dr. Md. Sazzadur Rahman
Associate Professor
Institute of Information Technology, Jahangimagar University

DECLARATION



This statement states that **Zahid Hosen** completed the thesis titled **"Object Detection On Road Using Deep Learning Approach"** under the guidance of **Mr. Md. Shohel Arman,** Assistant Professor, Department of Software Engineering, Daffodil International University. Additionally, it declares that neither this paper nor any component of it has been submitted to another institution for the conferment of a degree.

Zahid

Zahid Hasan
ID: 192-35-2873
Batch: 29
Department of Software Engineering
Daffodil International University.

Certified By:

Mr. Md. Shohel Arman
Assistant Professor
Department of Software Engineering
Daffodil International University.

ACKNOWLEDGEMENT

First, to Almighty God, I express my heartiest thanks and gratefulness for His divine blessing in making it possible to complete the final year thesis successfully.

I am grateful and wish for our profound indebtedness to **Md. Shohel Arman**, **Assistant Professor**, Department of Software Engineering, Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of "*Deep Learning*" to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant

and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts, and correcting them at all stages have made it possible to complete this project.

I would like to express my heartiest gratitude to Dr. Imran Mahmud, Head In-Charge of the Software Engineering faculty for his kind help to finish our project and also to other faculty members and the staff of the SWE department of Daffodil International University.

Finally, I must acknowledge with due respect the constant support and patients of our parents.

ABSTRACT

For a developing nation like Bangladesh, traffic jams are a major problem. Systems for maintaining traffic manually are expensive and time-consuming. Making a system that can automatically identify traffic flow is therefore necessary in order to help the authorities determine whether roads are busier or less congested. Developed nations have already created a system that Bangladesh is unable to afford. Therefore, I've made my choice to create a system that will assist the authority in detecting, tracking, and obtaining traffic flow at a reasonable cost. In order to create a system to recognize and track vehicles at a minimal cost, I have employed transfer learning of convolutional neural networks. Transfer learning is a system where we may reuse the code. I used 2 convolutional neural network models(CNN), 1 algorithm, and transfer learning techniques throughout. They are YOLOv8, and YOLOv7. The entire model's mAP has been produced. The YOLOv8 model, however, provides the highest mAP of the two. In the future, I'll focus on obtaining data on how each vehicle on the road is affected by traffic flow and I'll update the video dataset to obtain more accurate data and mAP for traffic analysis.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

1. INTRODUCTION

1.1 BACKGROUND

Traffic Jam refers to a long line of vehicles on the road leading to a serious roadblock. Normally, traffic jams are created by primitive traffic control systems, a huge number of vehicles, narrow roads, road damage, insufficient traffic signs and signals, poor road structures, reckless parking alongside pavements, and unhealthy competition among drivers. Traffic jams are an old scar in Dhaka city. It has been intensifying day by day. It has become a curse for people. It has become a common phenomenon in Dhaka city.

Now According to the Dhaka Tribune, in Dhaka cities, most of the roads have not been properly controlled by the traffic system, and the roads are narrow and damage a huge number of vehicles. Every day, about 18 people are killed in road accidents due to uncontrolled vehicle movement. Among the world, South Asian countries faced the problem of traffic most. Bangladesh is one of those countries. For solving traffic jams, proper monitoring is needed. Because without monitoring it's hard to detect where there is a foot over bridge, flyover and traffic surgeon. After detecting vehicles, management needs to understand the type of vehicle and which precaution is mostly needed to control traffic. If the management does not monitor properly, then it will become hard for them to detect the vehicle and understand traffic flow. So, controlling traffic jams is a time-consuming and costly process but it is very necessary to maintain traffic because traffic jams have an impact on the people of the country.

In other countries, they have already developed some processes to detect and classify vehicles to understand the type of road accidents. There is a global native Bangladeshi vehicle (Poribohon-BD**,** 2020) dataset which is available online. It contains 9058 images. The dataset has been made on road images from Bangladesh.

Worldwide people are researching different types of vehicle image dataset.

There are several algorithms and techniques which are used to solve the challenges.
In different works, researchers have used different types of algorithms and most of their works are based on deep learning. Because it's image processing-based work. So, with deep learning, it's easy to classify, detect and count vehicles from a road surface.

So, for classifying vehicles from images, deep learning is a very popular method. Why is classification technique important? Because it is very important to teach a model which one is a small car and which is not. What type of vehicles cause traffic jam? One can get all the question-answers from these questions. In a standard classifying

format, there are at first many more types of native Bangladeshi vehicles. All of these vehicle types are classified with different class names. They are decided into 15 classes named Bicycle, Bike, Boat, Bus, Car, Cng, Easy-bike, Horse-cart, Lauch, Leguna, Multi-class Vehicles(table-1). For road surfaced vehicles, we don't need Boat and Launch classes. For classifying vehicles peoples approach with both machine learning models and deep learning methods. Different researchers classified road damages into different types. Some researchers only classified the cracks and some researchers classified them into other types.

Vehicle Type	Detail	Class Name
On Road Surface	Two wheeled (Small)	bicycle
	Two wheeled	bike
	Four wheeled (large)	bus
	Four wheeled (Small)	car
	Three wheeled	cng
	Three wheeled (Medium)	rickshaw
	Four wheeled>= (Larger)	truck

	Road Damage	road_damage
	Traffic Sign	traffic_sign
	Person	person

TABLE 1: STANDARD CLASSIFICATION OF ClASSES

I utilize a categorized dataset in the work I do. I have approached a model to detect vehicles, traffic signs, road damage and persons from video data. From the detected vehicle, for counting we needed to track the vehicle.

1.2 MOTIVATION OF THE RESEARCH

Traffic Jam is one of the major problems which causes many problems for the people of a country. Moreover, a lot of uncontrolled vehicles cause a lot of people’s death every year. Roads which have the most traffic create a problem for people during daily moving. It is very hard to maintain every road traffic free so fast for a developing country like Bangladesh. Getting traffic flow manually is time

consuming and costly. If there is a system for the authorities to detect the vehicles which cause traffic more and know the traffic flow, traffic issues and accidents can be prevented. In other countries, they have already developed some systems which are used to detect and track vehicles. There is a global Bangladeshi Native Vehicle (Poribohon-BD, 2020) dataset to use for vehicle detection and tracking. People from other countries use their country’s dataset to develop the system. The developed systems are automated, low-cost and time-saving. The authority is using the system and correcting the road at a low cost and easily by saving time. We have found

that there is very little work based on our country’s traffic jams. By inspiring that we have to think of making a system that will detect vehicles, track them and show traffic flow. Dhaka is a populated city and there are many traffic issues which are waiting to be cured by the authorities. It's time and money-consuming for the authority. So, to help the authority to detect the cause of traffic we have made a system which will help the authority to get traffic flow easily. By tracking vehicles, they will be able to find out which one is more in need of services. From our work, they will be able to get the traffic flow where there is no need for foot over the bridge or the Surgeon.

So that they will be able to easily rank the most busy roads and free roads and can use the road easily. So, the motivation behind the work is to make a low-cost system for our country which will give the best results on the basis of the dataset from the country’s road.

1.3 PROBLEM STATEMENT

We know that traffic issues are quite problematic for people of any country. Early, many works on traffic jams have already been done. Many authors of the research work with different types of algorithms such as Deep Convolutional Neural Network, CNN, R-CNN, embedded systems, and many more. Some approaches with state to art solutions also. Some approaches detect vehicles with common vehicles. For classifying the type of vehicles, they applied SVM, Random Forest, decision tree, and many more. They collected the data with different systems. Some used customized datasets and some used Global data which is available. Other countries

worked with their road’s dataset but the Bangladeshi climate and situation is different from others. So, the research is applicable for their own country. For our country Bangladesh we want to apply deep learning models on the dataset created from our country’s vehicles because Bangladeshi native vehicles are in different classes. They are Bicycle, Bike, Bus, Car, Cng,Van, Truck, Leguna, Rickshaw, Truck as well Multi-class Vehicles in road surface road damage and traffic signs. Because in our country's perspective other types of vehicles are not so much available.

1.4 RESEARCH QUESTIONS

The research question was

- Q1: Which model performs better for tracking vehicles, road damage and traffic signs ?
- Q2: Which model gives faster results?

1.5 RESEARCH OBJECTIVES

My paper's significant objective is to pinpoint, track, and count vehicles on a road surface, as well as to identify road damage and traffic signs at a minimal cost. Additionally, we sought a more favorable result to broaden the application of our methodology. Our thesis goals are:

- Building an automatic system.
- For creating a low-cost infrastructure that a developing country like Bangladesh might afford.
- Detect, track and count vehicles to get Traffic flow on Dhaka City Roads.

1.6 RESEARCH SCOPE

The primary focus of the research is as follows:

- To create a system that can recognize and track vehicles in videos in order to count and determine traffic flow. This method will be based on the Bangladeshi native vehicle dataset.

- Will support the authorities in Bangladesh in creating a system that is affordable and simply understands traffic movement on congested routes and free roads. Consequently, it will enable the government to lessen people's suffering.

1.7 THESIS ORGANIZATION

In the first chapter, a particular part on the Vehicle counting system and its usage, the background behind the work, motivation of the research, problem statement, research questions, and research objectives are discussed. The other parts related to our research are as below:

In the next chapter I will discuss the literature review where we can see some researcher’s studies which have already been done on the same field of traffic issues, their used methodology, lacking and on the basis of their work comparison among my work and their work. In their chapter, we will discuss the methodology of our work. In the methodology of my work, I will discuss data collection, a better result from data increpre-processing and analysis of our tasks. In chapter four, the methodology's findings will be addressed. The final chapter acts as the ending. In this section, I will present the conclusion, which will include a comprehensive summary of my efforts. I've discussed the work I'll do in the future to develop my career.

CHAPTER 2

2. LITERATURE REVIEW

2.1 INTRODUCTION

In an overview of literature, a researcher examines previous work, research, conference papers, books and articles, and so on. It may be used to learn about past research on the issue, provide a basic overview of it, and identify any gaps in the study. Following analysis, they may concentrate on limitations and devise methods to work around them in order to get more favorable outcomes.

2.2 PREVIOUS LITERATURE

The concept of counting vehicles and traffic analysis started from the time when people started facing accidents for primitive manual traffic systems. Bangladeshi roads are not having enough space, there is a lot of road damage and insufficient traffic signs or signals. Moreover, huge numbers of vehicles go through. For this reason, a huge number of accidents and traffic jams occur every year. On vehicle counting and traffic analysis, to categorize road damage, many researchers have already conducted their studies and used various machine learning methods. Detection and classification tasks are included in some of the works but only classification, cannot track the vehicle to give traffic flow. Therefore, I have mostly concentrated on the detection and tracking components for my work.

Cheng-Jian Lin and et al [3] evaluated the conditions of the vehicles counting and traffic analysis by a Deep neural network. To automatically count and detect vehicles with localisation, they employed the YOLO GMM and a quicker R CNN deep neural network technique. They retrieved image data from traffic videos recorded with online cameras installed along various roads in Taiwan. They also estimate the speed of vehicles and count in particular zones.

Amit Ghosh and et al [5] worked for vehicles counting and traffic analysis. They used the BackgroundSubtractionMoG2 algorithm, OpenCV, Java SE Development Kit 8 to detect vehicle classification counting and vehicle detection, speed measurement automatically. They used the collected data through hardware was detecting and classify vehicles and measure speed.This research could use machine learning algorithms to mine traffic patterns.

T.M Amir-ul-Haque Bhuiyan and et. al. [3] approaches with a computer based traffic monitoring and analyzing system. With HOG, Haar features an Adaboost classifier that monitors the whole system and they apply SVM (Support Vector Machine) for vehicle classification. They collected data from different roads of Dhaka. This research could give the prediction of Traffic flow.

Zillur Rahman and et al [3] worked for the wrong way vehicle counting detection using a deep neural network. For that work, they used camera footage. They used the YOLO Centroid tracking algorithm for detecting wrong way vehicles. In order to prevent the ID number from being switched owing to object overlap, the centroid of the object must be near together between future frames.

With HOG, SVM, YOLO Wenming Cao and et al. [4] detected the real time video for object detection and also made a classification by fast Deep Neural Networks. They use sample data for detecting objects from the real time video. They achieved a new method which is developed for faster object detection. They intend to optimize network architecture and explore further performance-enhancing methods in the future.

Ohidujjaman and et al. [8] worked with faster R-CNN and MOG algorithms for applying a deep learning approach on vehicle detection and counting. They obtained their data from videos captured by cameras for detecting vehicles and counting. Their main aim was to traffic management system control and maintenance through vehicle detection and counting.

Lili Jia and et al [6] worked on a real-time vehicle detection and tracking system in street scenarios.They use continuously adaptive MeanShift (CamShift) algorithms to detect vehicle tracking and also classify them. The dataset was derived from captured video from different highways. The videos were captured from China. They tried to detect the vehicles and track them. They said that it could track and detect multiple objects in different environments. Ya-Li Hou and et al [3] explore the tricks for counting people in a challenging situation. They used Gaussian Mixture Model (GMM), Kanade-Lucas-Tomasi (KLT), EM algorithms and clusters for human counting. Their dataset was collected from a 4-hour video that was captured by a camera. The system in research successfully detects humans. The intention of the research was to detect and count humans, and this could give an estimation of gathering.

For Human detection in surveillance videos, its application reflection and classification Manoranjan Paul and et al [3] used HMM, SVM algorithms. They also used Mixture of Gaussian model, Non-parametric background model, Temporal differencing, Hierarchical background model classification and human. They used KTH human motion dataset, Weizmann human action dataset, PETS dataset, INRIA XMAS multi-view dataset, CASIA Gait database. Then the system in the research successfully detects humans. They also said that it could exploit a multi-view approach and adopt an improved model based on localized parts of the image.

Using Kalman filter detection Damien Leflocha and et al [5] wanted to detect real time people counting and classify them. They also used Automatic segmentation algorithms for video analysis, video surveillance, background estimation, segmentation object tracking. For this research, they used PETS2006 datasets to detect humans and count them as well. They wanted to detect and classify people using a single video camera. This research was detecting and counting humans and also shadows in crowds. They also said that it can't detect humans without a crowd.

For "Vehicle detection and classification system based on virtual detection zone" et al [21] used GMM and KNN. The suggested system is divided into four stages: foreground extraction, vehicle detection, vehicle feature extraction, and vehicle categorization. The Gaussian mixture model (GMM) is used to detect a moving vehicle initially. Then, to acquire accurate foreground objects, numerous approaches such as region of interest selection, adaptive morphological operation, and contour processing are used. When a vehicle's centroid is on the VDZ, vehicle characteristics are determined. Finally, the k-nearest neighbor classifier is used to classify vehicles.

For "Native Vehicles Classification on Bangladeshi Roads" et al [22] used CNN and YOLO.Vehicle detection or identification is a crucial problem in bringing an intelligent transportation system to a developing country like Bangladesh in order to study the distinctive characteristics of the local cars. Following on from their great breakthrough in object detection using deep convolutional neural networks (CNN), they addressed the recognition of native vehicles on Bangladeshi roads. In this study, a CNN-based transfer learning strategy for vehicle categorization is applied to the popular You Only Look Once (YOLO) framework. The aim is to reuse the YOLO framework's pre-trained convolutional layers and transfer information to recognize 15 unique cars in Bangladesh and collect data from traffic scenes.

X. Hu, Z. Wei, and W. Zhou et al [19] worked with YOLO.To accomplish safe vehicle driving, information about the vehicle's surroundings must be seen, and computer vision is one of the major technologies for solving this challenge. This work provides an improved YOLOv4-based video stream vehicle target identification technique to address the detection speed issue. This study first theoretically describes the YOLOv4 method, then offers an algorithmic technique to improve detection speed, and lastly performs real road trials.

S. Tabassum, S. Ullah, N. H. Al-nur, and S. Shatabda et al [23] worked with CNN, YOLO, VGG-16 and R-CNN algorithm for “Poribohon-BD: Bangladeshi local vehicle image dataset with annotation for classification”.This article introduces the 'Poribohon-BD' dataset for vehicle categorization in Bangladesh. Images of vehicles are gathered from two sources: i) a smartphone camera, and ii) social media. The collection includes 9058 tagged and annotated photos of 15 native Bangladeshi vehicles, including a bus, motorcycle, three-wheeler rickshaw, truck, and wheelbarrow. Data augmentation techniques were used to keep the quantity of photos comparable for each vehicle type. Tzuta Lin's LabelImg tool was used to label the photos. To ensure privacy and secrecy, human faces have also been obscured.

With ANN, AdaBoost algorithm Soumen Kumar and et al. [3] detected the vehicles detection, count and classification. They used the SVM, MOG, GMG for image classification, video tracking, information gathering, information analysis, target detection and vehicle count. Their dataset was from captured video from Indian roads. They aimed to detect and label videos and classify images in a timely manner. The research was for detecting and labeling videos and classifying images. This also could give the prediction of traffic flow.

2.3 CONCLUSION

Different kinds of algorithms are employed. They used feature extraction, augmentation, annotation, and numerous other techniques to achieve a decent outcome. Forget about real-time performance and increased precision. We also attempted to count and track vehicles, road damage and traffic signs while working on real-time traffic analysis in our work.

CHAPTER 3

3. RESEARCH METHODOLOGY

3.1 RESEARCH METHODOLOGY

For our research, we used YOLOv8 and YOLOv7 on a dataset gathered in Dhaka City.

3.2 DATA COLLECTION

On the internet, there is already a dataset that has undergone a great deal of effort. Datasets have been gathered from countries like Japan, the Czech Republic, and India. On the internet, there are more dataset resources that come from many nations. But there is some Bangladeshi native vehicle dataset available on Roboflow workspace. However, the video data is not accessible online. We therefore considered gathering video data from the streets of Dhaka. I just gathered video datasets from Dhaka City for this research (Figure 1).

Figure 1: Vehicle sample videos

3.3 DATA PREPROCESSING

That dataset has about 4000 images of Bangladeshi native vehicles and I annotate about 3000 images . The annotated values are stored in XML files. The data files are divided into 10 folders. The model's accuracy improved as a result of the processed data. So, the dataset has 10 classes. I separated the data into train , test and validation sets after gathering road photos. There are 80% of the data in the training dataset, 13% in the validation set, 7% in the test set, and in both training datasets, there are ten folders of 10 types of datasets. There are a total of 20 videos in the test dataset. The dataset contains a total of 2.5 thousand data, 408 data for validation, and 205 data points for testing. Consequently, the dataset contains 3141 data in total (Figure 2).

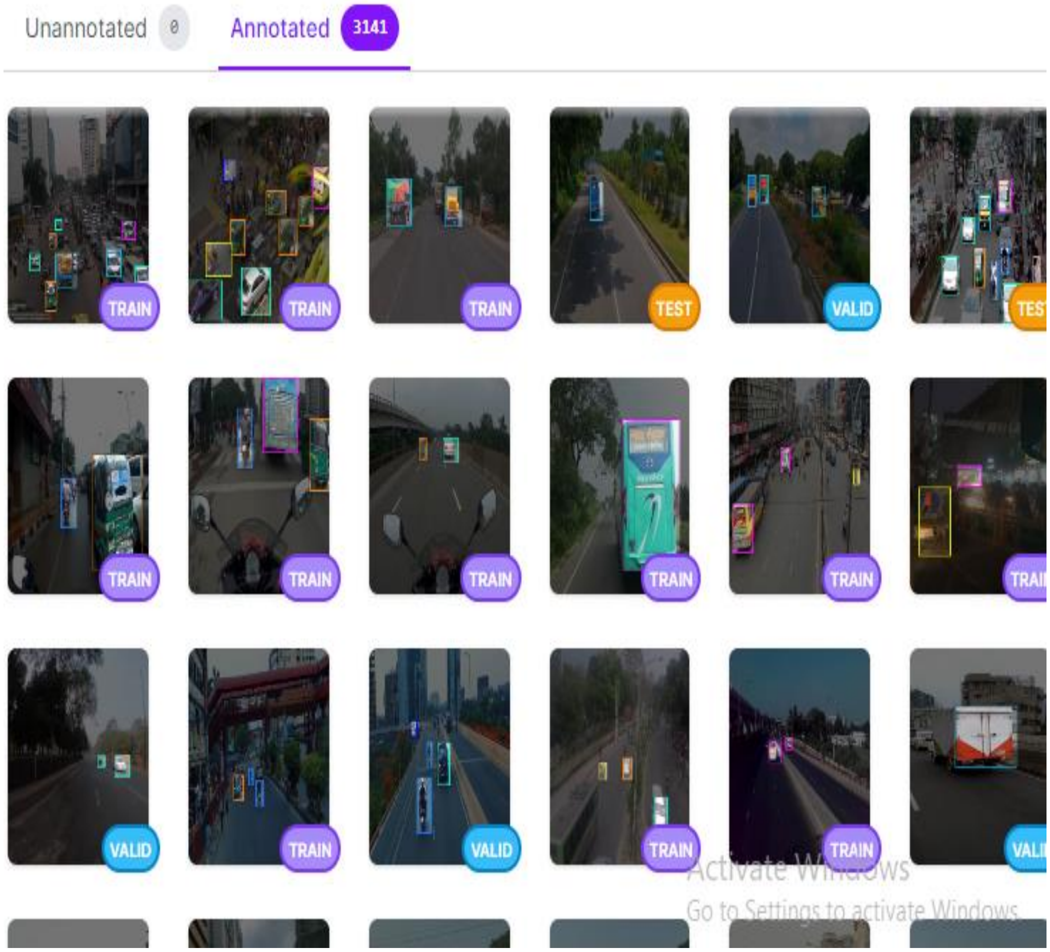


Figure 2: 3 Types dataset sample

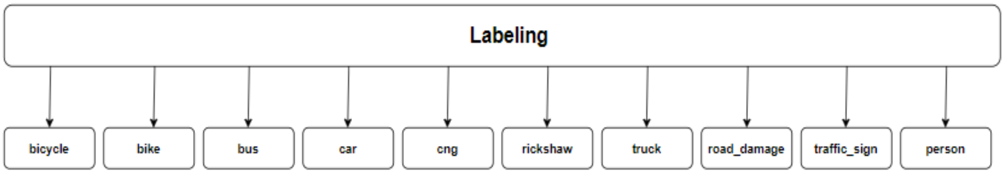


Figure 3: Dataset Labeling

For data preprocessing we follow this methodology-

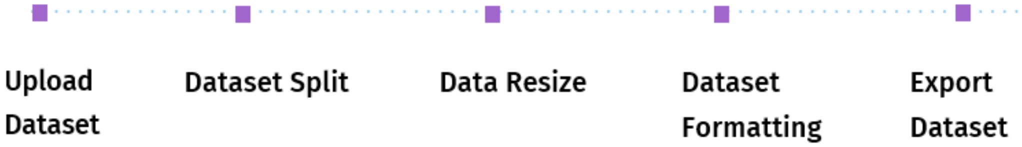


Figure 4: Data Preprocessing Methodology

3.4 CONVOLUTIONAL NEURAL NETWORK(CNN)

I'm using all the convolutional neural network models for our data tracking and detection. So, this explains the fundamental concept behind CNNs (Convolutional Neural Networks) (figure 5)

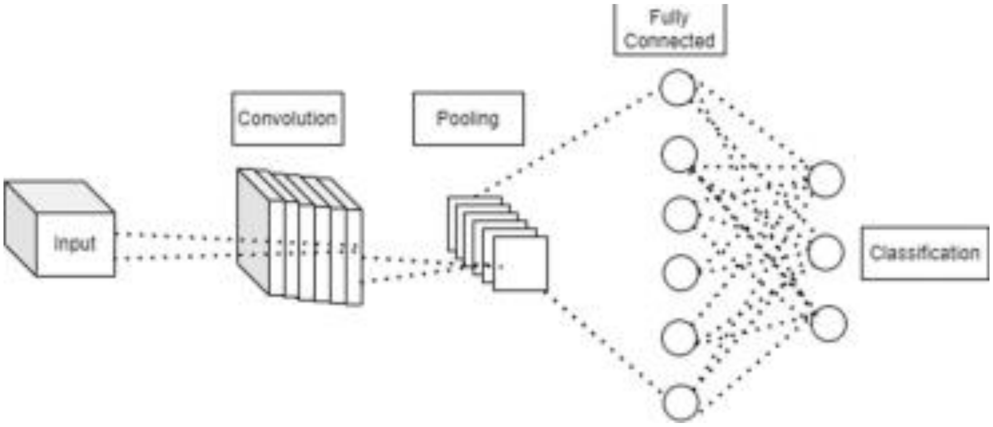


Figure 5: Convolutional neural network's basic architecture

The convolutional layer is the initial layer utilized for feature extraction after receiving input. The image is scaled down to (M*M) size at this point.

The pooling layer comes after that. Here, the size of the previous image has been scaled back to save on computation fees. Then the biggest element is taken to create a max-pooling layer from the feature map.

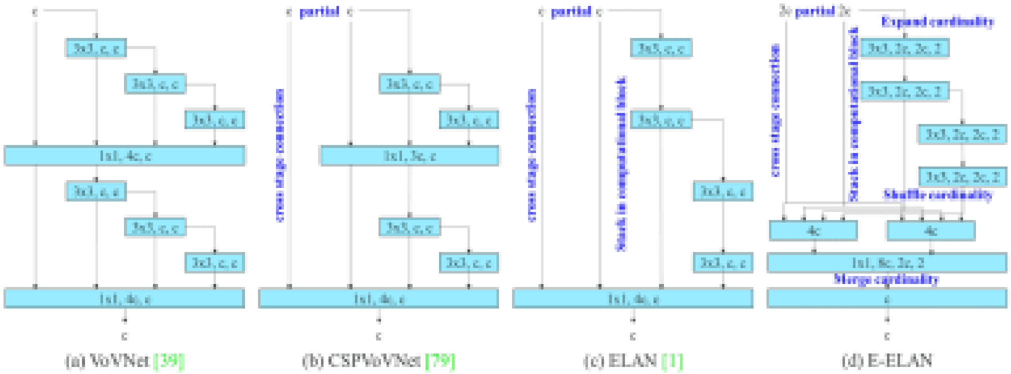
Then it's time for a layer that is completely connected. Before the output layer, they are the last layer. The photos, which are taken from the preceding layer, are flattened here. The classifying process is only getting started here.

Overfitting may happen after connecting to the completely connected layer. Dropout is therefore employed as a solution to this issue. Here, it is chosen which data will move on to the next stage and which data may be fired. The activation function is one of the most important characteristics. Activation functions like ReLu, Softmax, and tanh are some of the more popular ones.

I have utilized the Softmax activation Function for my work.

3.5 YOLOv7

The authentic object identification model for computer vision programs that is quickest and most accurate is ‘YOLOv7’. Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao published the authorized YOLOv7 article titled "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors" in July 2022. The computer vision and machine learning community are buzzing about the “YOLOv7” algorithm. The most recent ‘YOLO’ algorithm outperforms all earlier object detection methods and YOLO iterations in terms of speed and precision. It can be taught significantly quicker on tiny datasets without any pre-learned weights than for other neural network models and requires technology that is several times less expensive. As a result, ‘YOLOv7’ is anticipated to overtake ‘YOLOv4’, the previous state-of-the-art for real-time applications, to become the accepted standard for object recognition in the near future. The “real-time object detection” performance is significantly increased by ‘YOLOv7’ without raising inference expenses. ‘YOLOv7’ effectively outperforms other well-known object detectors by reducing about 40% of the parameters and 50% of the computation required for state-of-the-art real-time object detection. This allows it to perform inferences more quickly and with higher detection accuracy. In summary, ‘YOLOv7’ offers a quicker and more robust network architecture that offers a better feature integration approach, more precise object recognition performance, a more robust loss function, and an improved label assignment and model training efficiency. Because of this, ‘YOLOv7’ uses far less expensive computational hardware than other deep-learning models. It accepts the COCO dataset. The intended re-parameterized convolution design in ‘YOLOv7’ employs RepConv without identity connection (RepConvN). When re-parameterized convolution is used to replace a convolution process with residue or concatenation, the intention is to prohibit contact points from emerging.



(a) VoVNet [39]
(b) CSPVoVNet [79]
(c) ELAN [1]
(d) E-ELAN

Figure 6: YOLOv7 Architecture

3.6 YOLOv8

YOLO(You only look once) v8 is an object detection algorithm. The most recent and cutting-edge YOLO model, YOLOv8, can be utilized for applications including objects identification, image categorization, and instance segmentation. Ultralytics, who also produced the influential YOLOv5 model that defined the industry, developed YOLOv8. Compared to YOLOv5, YOLOv8 has a number of architectural updates and enhancements. It is a novel convolutional neural network(CNN) that detects objects in real time with great accuracy. This approach uses a single neural network to process the entire picture, then separates it into parts and predicts bounding boxes and probability of components (figure 7).

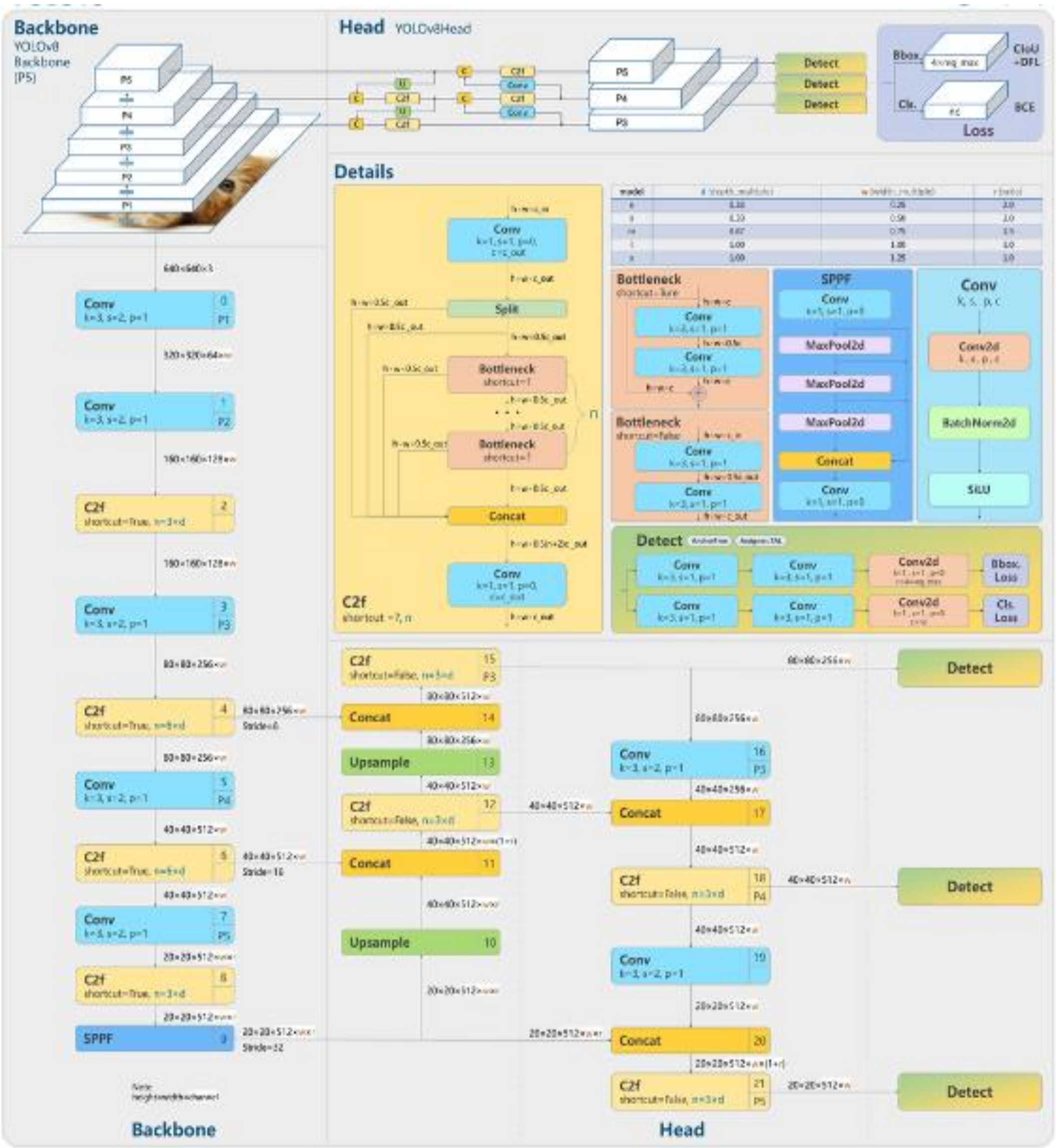


Figure 7: YOLOv8 Architecture

3.7 Transfer Learning:

All the proposed methods in the work are the transfer Learning of vision models. Making a shortcut for re-using the model weights which is pre-trained is known as transfer learning. It is the system of a neural network to solve one problem using the trained model of another problem. The transfer learning processes can detect the generic features of photographs. It achieved the state performance of state to art and still, the system remains effective.

3.8 Evaluation Methods

I created a confusion matrix to evaluate the results. It requires the true positive, true negative, false positive, and false negative values for evaluation.

True-positive means that the actual number was appropriately anticipated.
True-negative is rejected appropriately.
False-positive means that the value has been estimated to be positive when it is not.
False-negative is rejected mistakenly.

3.8.1 Accuracy:

The model's accuracy defines how well a computer can anticipate the results. It is significant when all classes are equally important. All classes are equally significant in my line of work. As a result, the accuracy is also significant in determining the model's correctness.

$$Accuracy = \frac{TrueNegatives + TruePositive}{TruePositive + FalsePositive + TrueNegative + FalseNegative}$$

3.8.2 Precision:

It is a term used to measure how well a machine learning model performs.Precision is calculated by dividing the true positive value by the all-positive value.

$$Precision = \frac{True Positive}{True Positive + False Positive}$$

3.8.3 Recall:

Recall is the measurement of the true positive that is precisely identified. The recall is calculated by dividing the true positive value by the total number of related documents.

$$Recall = \frac{True Positive}{True Positive + False Negative}$$

3.8.4 F1 Score

An F1 score is a measurement of test accuracy. F1 score is computed using recall and precision.

$$\text{F1 Score} = \frac{2 \times (\text{Precision} \times \text{Recall})}{\text{Precision} + \text{Recall}}$$

I have displayed the mAP along with model accuracy for both of the two models, YOLOv8 and YOLOv7. In model accuracy mAP, training accuracy and validation accuracy are considered for determining accuracy. Additionally, the loss for the model is discovered in terms of training accuracy and validation accuracy.

3.8.4 Mean Average Precision(mAP)

Mean Average Precision(mAP) is a metric used to evaluate object detection models such as Fast R CNN, YOLO, Mask R-CNN, etc. The mean of average precision (AP) values are calculated over recall values from 0 to 1.

The mAP formula is based on the following submetrics:

- Confusion Matrix
- Intersection over Union (IoU)
- Recall
- Precision

3.8.5.1 Confusion Matrix

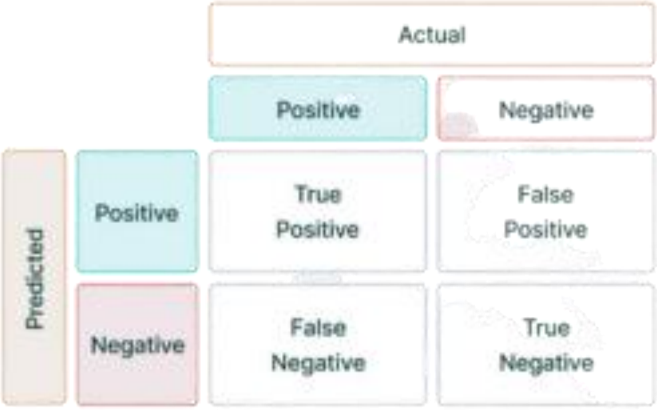
To create a Confusion Matrix, we need four attributes:

- **True Positives (TP)**: The model predicts a label and matches it correctly as per ground truth.

- **True Negatives (TN)**: The model does not predict the label and is not part of the ground truth.

- **False Positives (FP)**: The model predicted a label, but it is not part of the ground truth (Type I Error).

- **False Negatives (FN)**: The model does not predict a label, but it is part of the ground truth (Type II Error).



		Actual	
		Positive	Negative
Predicted	Positive	True Positive	False Positive
	Negative	False Negative	True Negative

Figure 8: Confusion Matrix

3.1.5.2 Intersection over Union (IoU)

The intersection over Union indicates the overlap of the predicted bounding box coordinates with the ground truth box. A higher IoU indicates the predicted bounding box coordinates closely resemble the ground truth box coordinates.

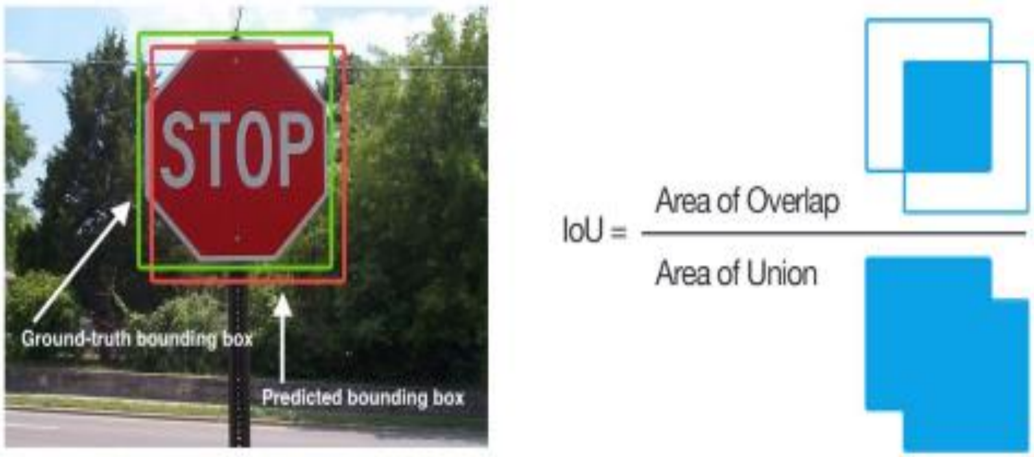


Figure 9: Intersection over Union

The mAP is calculated by finding the average precision (AP) for each class and then averaging it over a number of classes.

$$\mathrm{mAP} = \frac{1}{N}\sum_{i=1}^{N}\mathrm{AP}_i$$

Figure 10: Formula of mAP

CHAPTER 4

4. RESULTS AND DISCUSSION

4.1 INTRODUCTION

Models and methods for vehicle identification, tracking, and counting are discussed. Following data collection and preparation, I described the models I utilized to create my detection approaches. In that part, I'll go over the model's outputs.

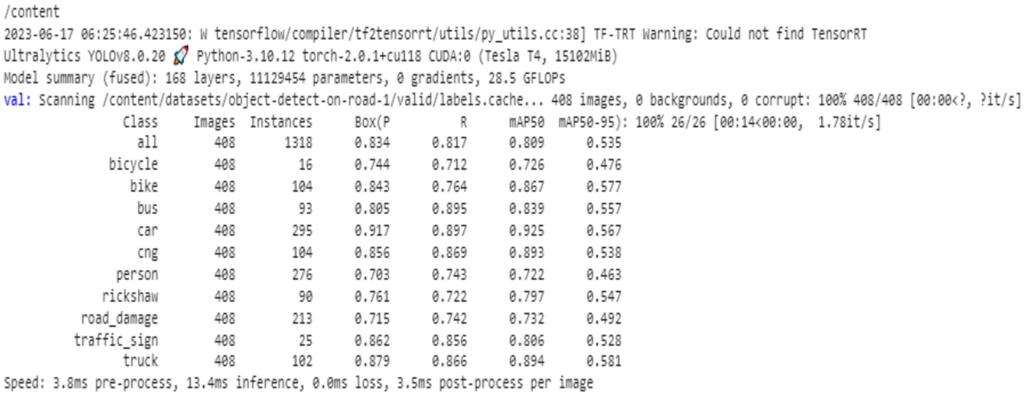
4.2 RESULT DISCUSSION

We have got the model's mAP (Mean Average Precision), accuracy, and loss plotting graph after applying the YOLOv8 and YOLOv7 models to our dataset. Training and validation accuracy are compared to generate a graph displaying the accuracy and loss of each model.

4.2.1 YOLOv8

Our custom-trained model has achieved 80.9% mAP rate after accomplishing 100 epochs. After training that custom model, we have a custom weight file which will help with further detection. For detecting purposes, We assign our confidence-level at 0.5 and image-size at 800. With the help of detect.py and custom-weighted files, we always received the highest accuracy in our results.

Result with description at a glance:



```
/content
2023-06-17 06:25:46.423150: W tensorflow/compiler/tf2tensorrt/utils/py_utils.cc:38] TF-TRT Warning: Could not find TensorRT
Ultralytics YOLOv8.0.20 🚀 Python-3.10.12 torch-2.0.1+cu118 CUDA:0 (Tesla T4, 15102MiB)
Model summary (fused): 168 layers, 11129454 parameters, 0 gradients, 28.5 GFLOPs
val: Scanning /content/datasets/object-detect-on-road-1/valid/labels.cache... 408 images, 0 backgrounds, 0 corrupt: 100% 408/408 [00:00<?, ?it/s]
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100% 26/26 [00:14<00:00,  1.78it/s]
                   all        408       1318      0.834      0.817      0.809      0.535
               bicycle        408         16      0.744      0.712      0.726      0.476
                  bike        408        104      0.843      0.764      0.867      0.577
                   bus        408         93      0.805      0.895      0.839      0.557
                   car        408        295      0.917      0.897      0.925      0.567
                   cng        408        104      0.856      0.869      0.893      0.538
                person        408        276      0.703      0.743      0.722      0.463
              rickshaw        408         90      0.761      0.722      0.797      0.547
           road_damage        408        213      0.715      0.742      0.732      0.492
          traffic_sign        408         25      0.862      0.856      0.806      0.528
                 truck        408        102      0.879      0.866      0.894      0.581
Speed: 3.8ms pre-process, 13.4ms inference, 0.0ms loss, 3.5ms post-process per image
```

Figure 11: Mapping of YOLOv8

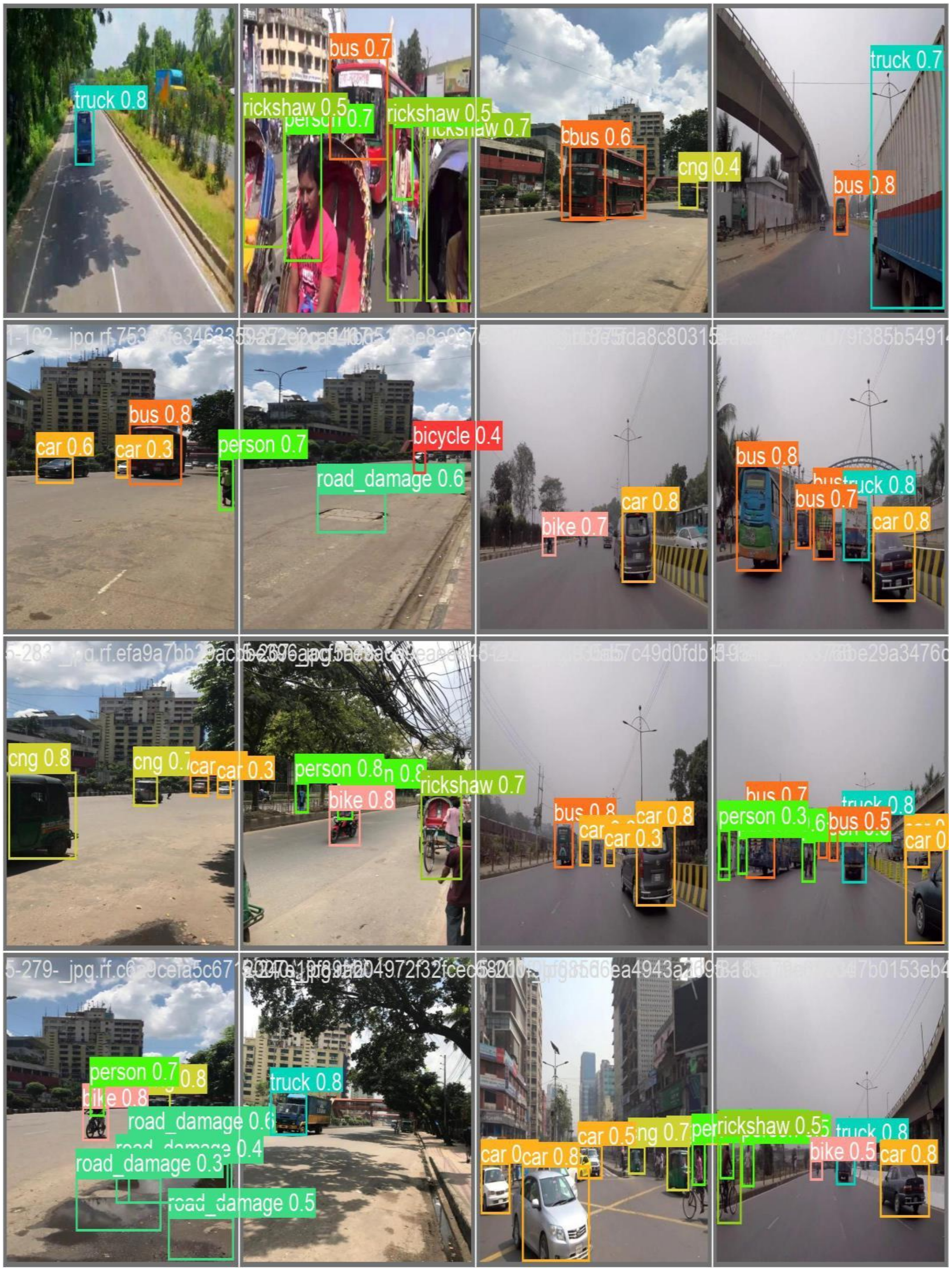


Figure 12: ‘Output of YOLOv8’ - Result 1

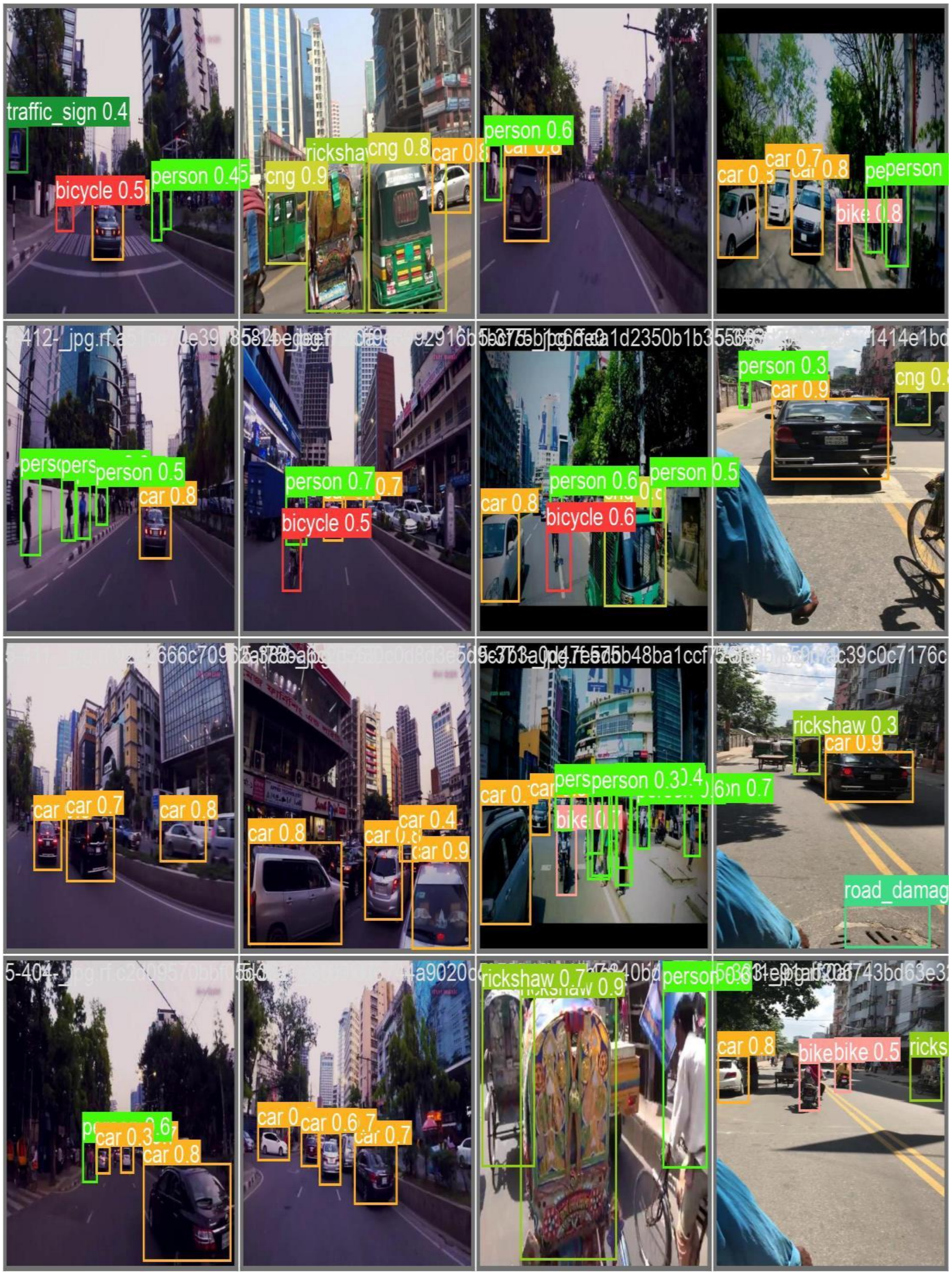


Figure 13: 'Output of YOLO8' - Result 2

With the model's train and validation value associated, some plotting diagrams are produced.

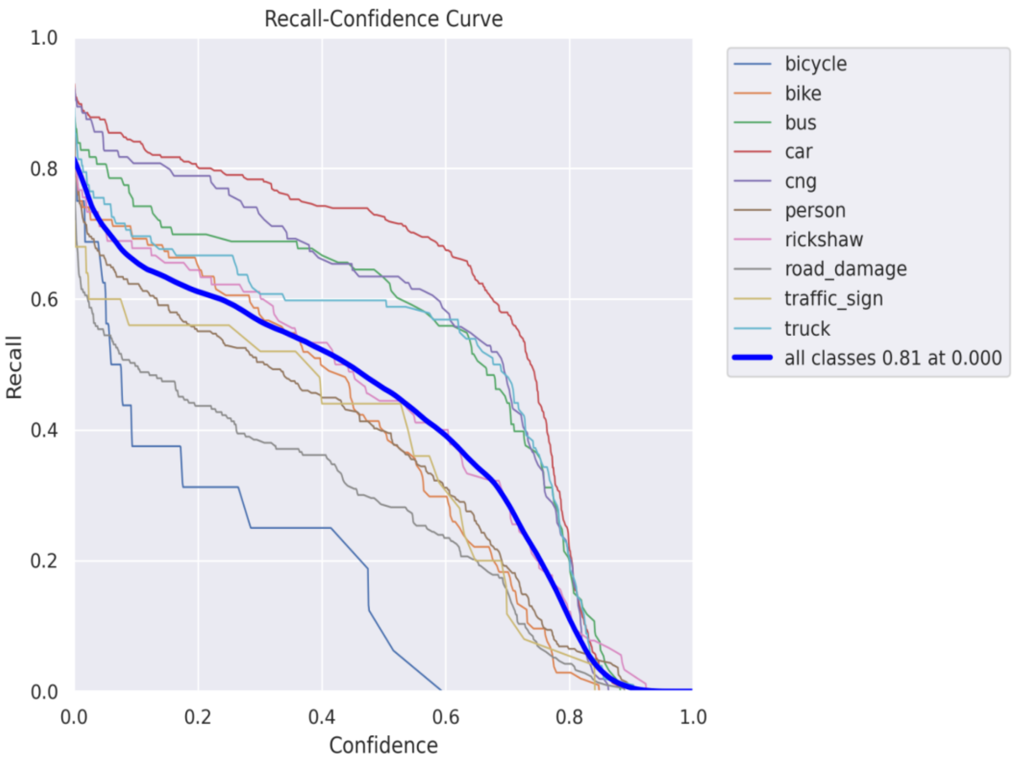


Figure 14: Recall Curve

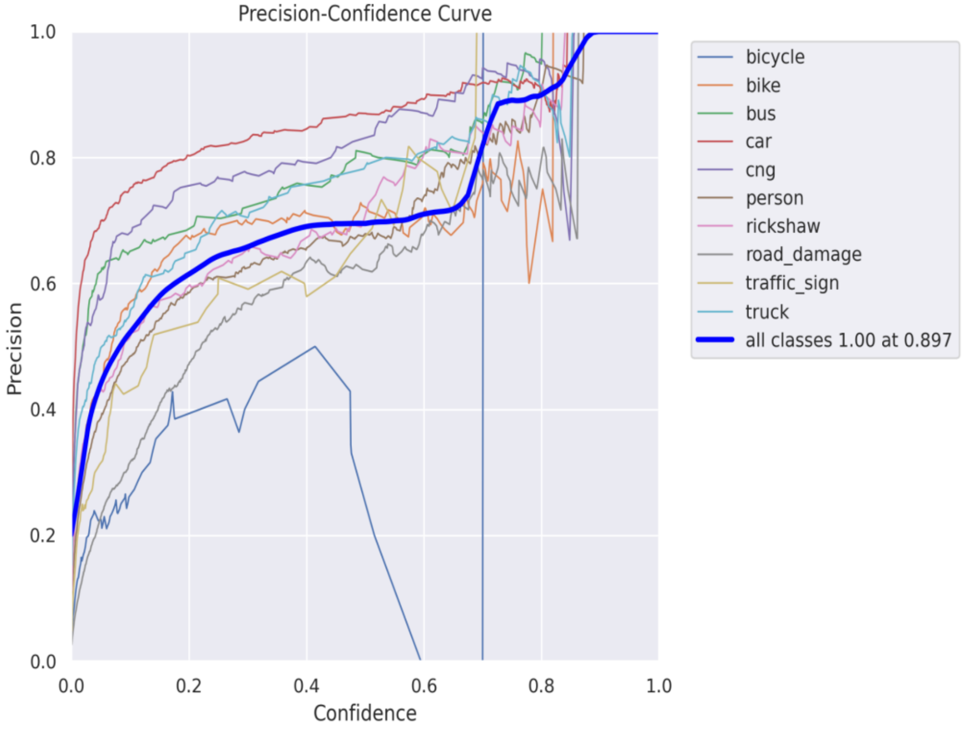


Figure 15: Precision Curve

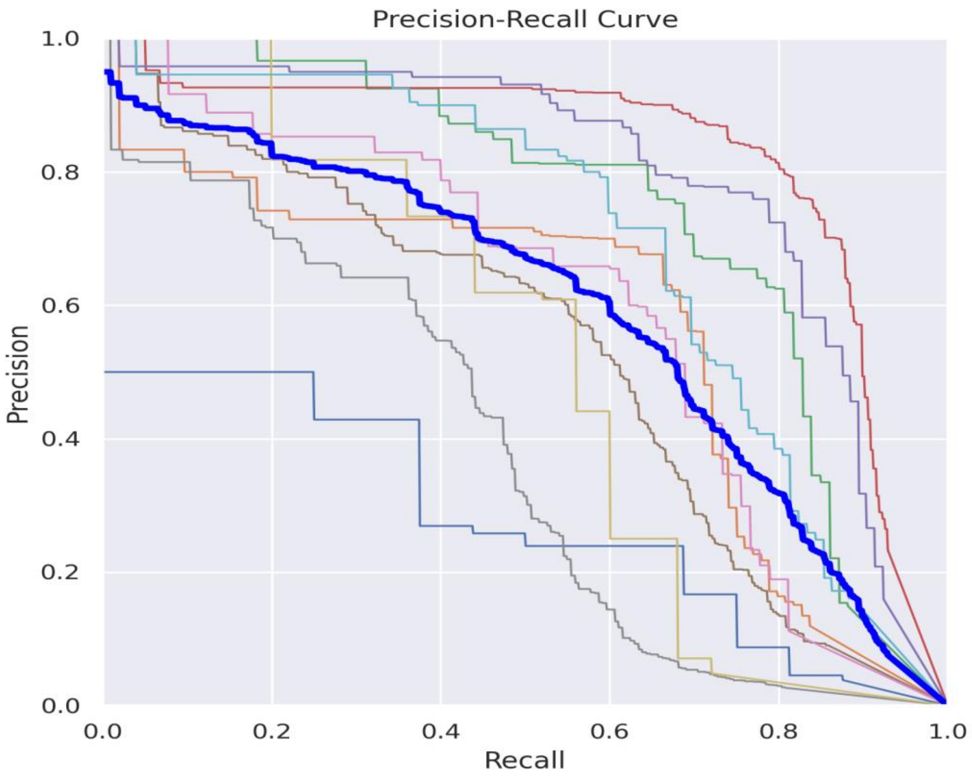


Figure 16: Precision Recall Curve

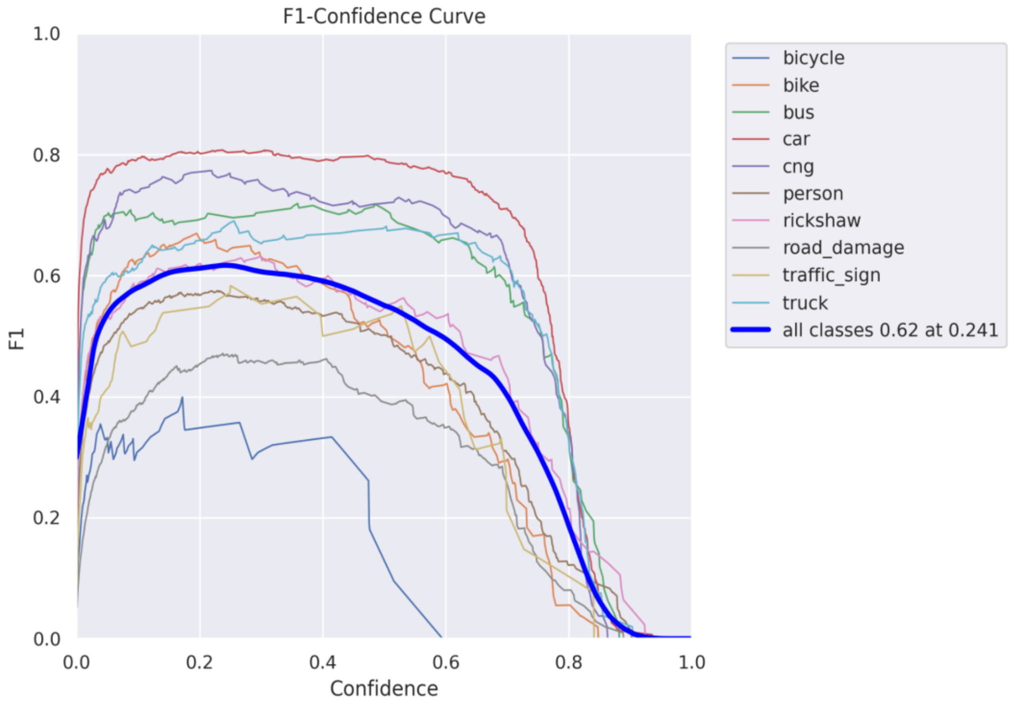
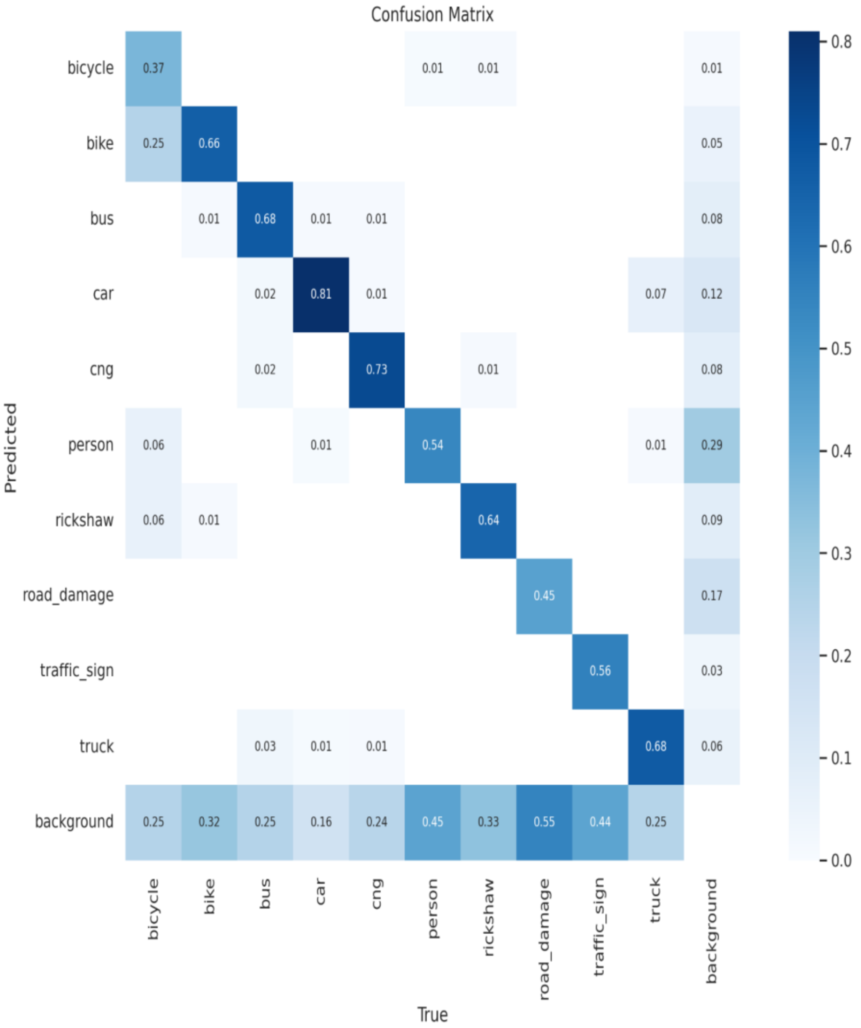


Figure 17: F1 - Confidence Curve



	bicycle	bike	bus	car	cng	person	rickshaw	road_damage	traffic_sign	truck	background
bicycle	0.37					0.01	0.01				0.01
bike	0.25	0.66									0.05
bus		0.01	0.68	0.01	0.01						0.08
car			0.02	0.81	0.01					0.07	0.12
cng			0.02		0.73		0.01				0.08
person	0.06			0.01		0.54				0.01	0.29
rickshaw	0.06	0.01					0.64				0.09
road_damage								0.45			0.17
traffic_sign									0.56		0.03
truck			0.03	0.01	0.01					0.68	0.06
background	0.25	0.32	0.25	0.16	0.24	0.45	0.33	0.55	0.44	0.25	

Figure 18: Confusion matrix

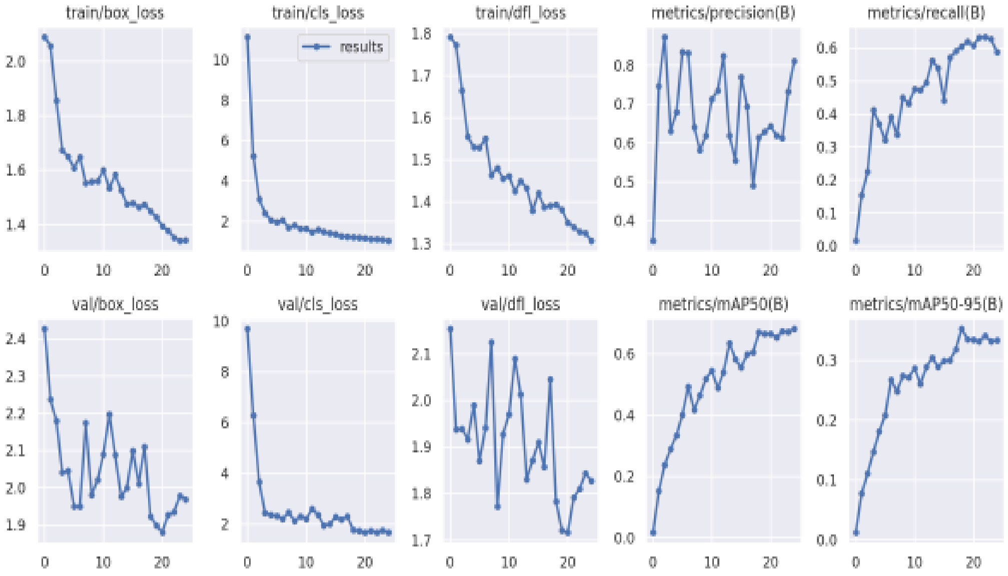
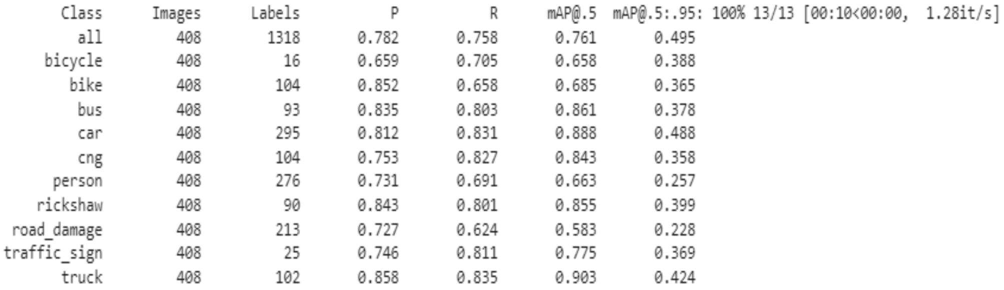


Figure 19: Result at a Glance

4.2.2 YOLOv7

Our custom-trained model has achieved a 76.1% mAP rate after accomplishing 30 epochs. After training that custom model, we have a custom weight file which will help with further detection. For detecting purposes, We assign our confidence-level at 0.5 and image-size at 640. With the help of detect.py and custom-weighted files, we always received the highest accuracy in our results.

Result with description at a glance:



```
         Class      Images      Labels           P           R      mAP@.5  mAP@.5:.95: 100% 13/13 [00:10<00:00,  1.28it/s]
           all         408        1318       0.782       0.758       0.761       0.495
       bicycle         408          16       0.659       0.705       0.658       0.388
          bike         408         104       0.852       0.658       0.685       0.365
           bus         408          93       0.835       0.803       0.861       0.378
           car         408         295       0.812       0.831       0.888       0.488
           cng         408         104       0.753       0.827       0.843       0.358
        person         408         276       0.731       0.691       0.663       0.257
      rickshaw         408          90       0.843       0.801       0.855       0.399
   road_damage         408         213       0.727       0.624       0.583       0.228
  traffic_sign         408          25       0.746       0.811       0.775       0.369
         truck         408         102       0.858       0.835       0.903       0.424
```

Figure 20: Mapping of YOLOv7

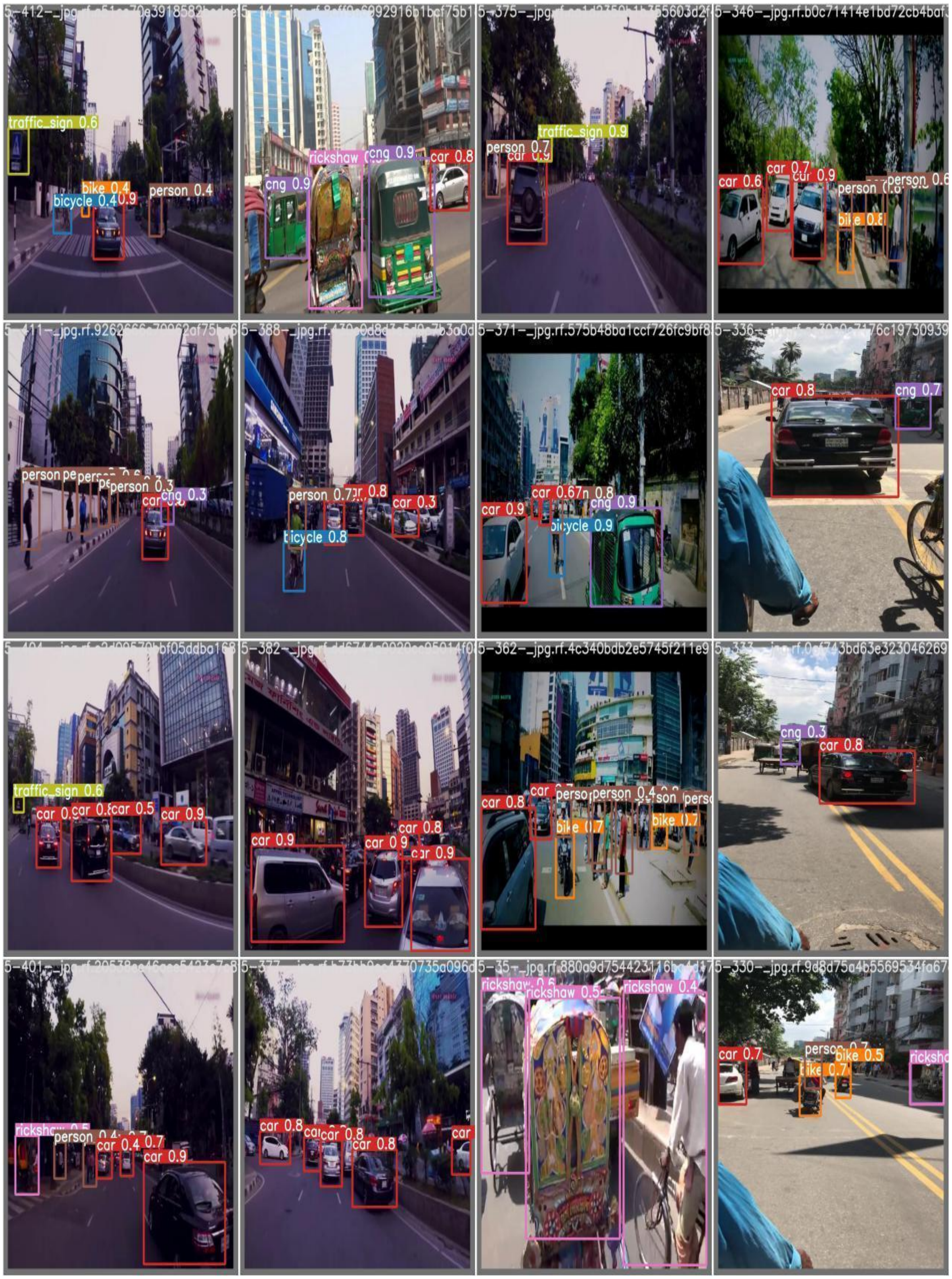


Figure 21: ‘Output of YOLOv7’ - Result 1

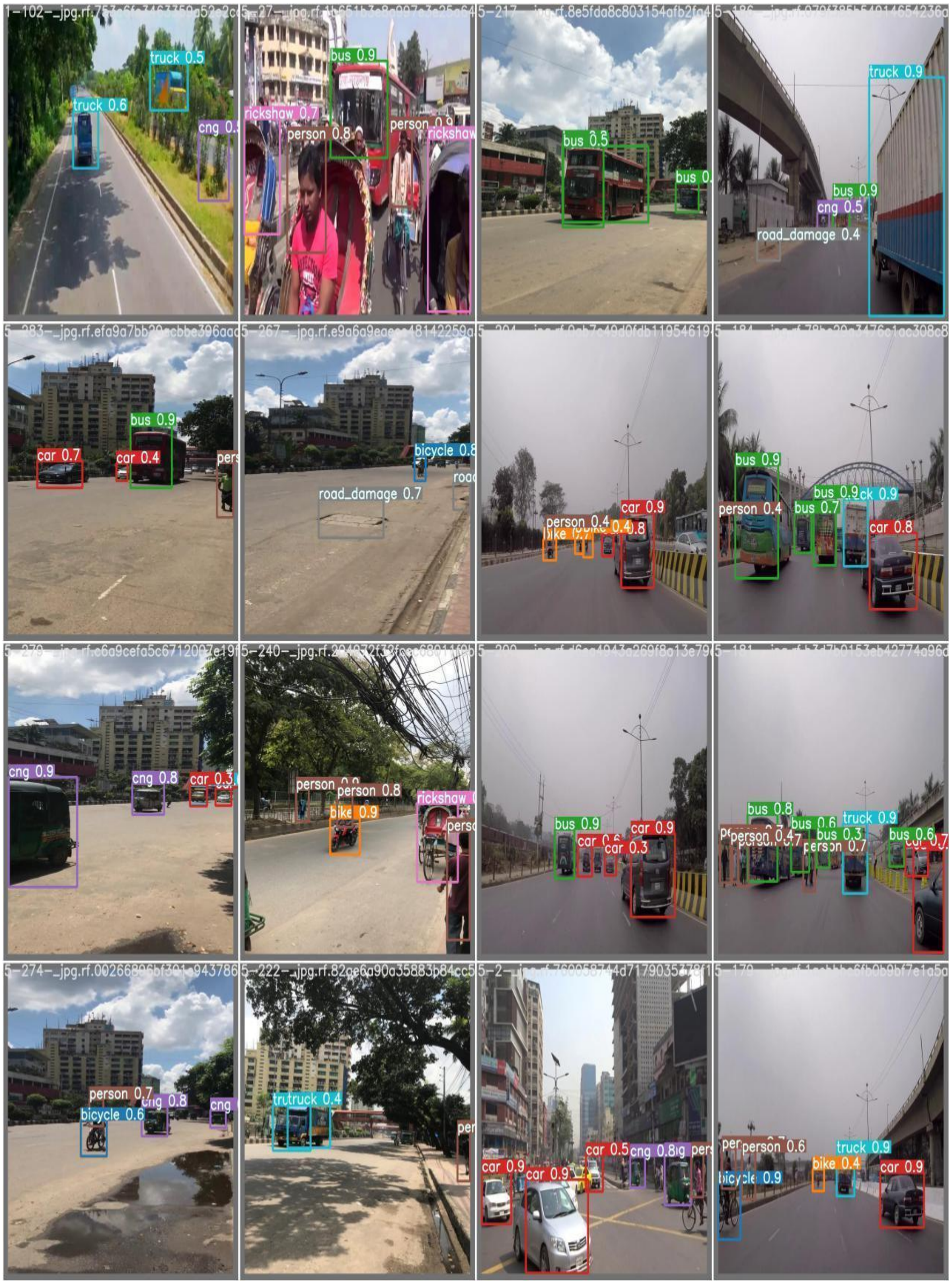


Figure 22: **‘**Output of Yolov7’ - Result 2

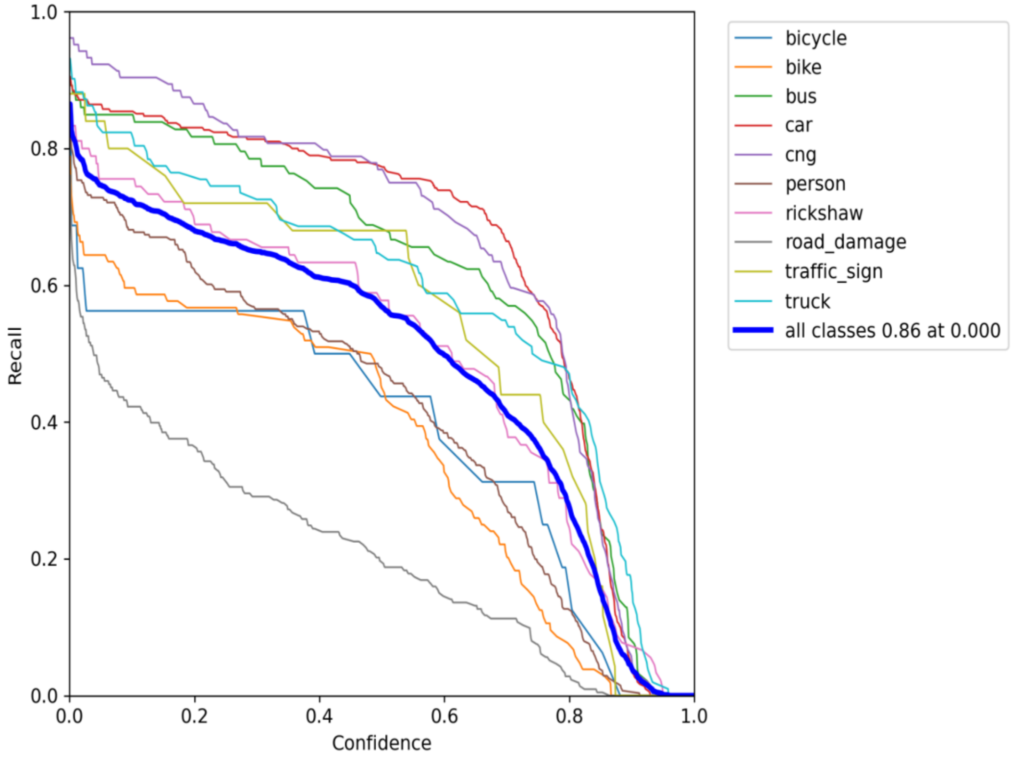


Figure 23: Recall Curve

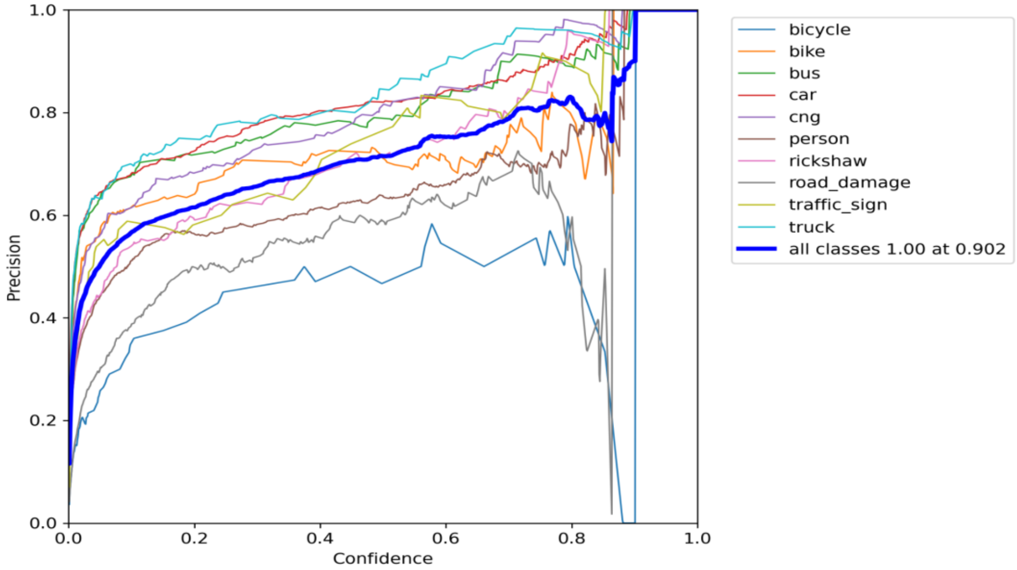


Figure 24: Precision Curve

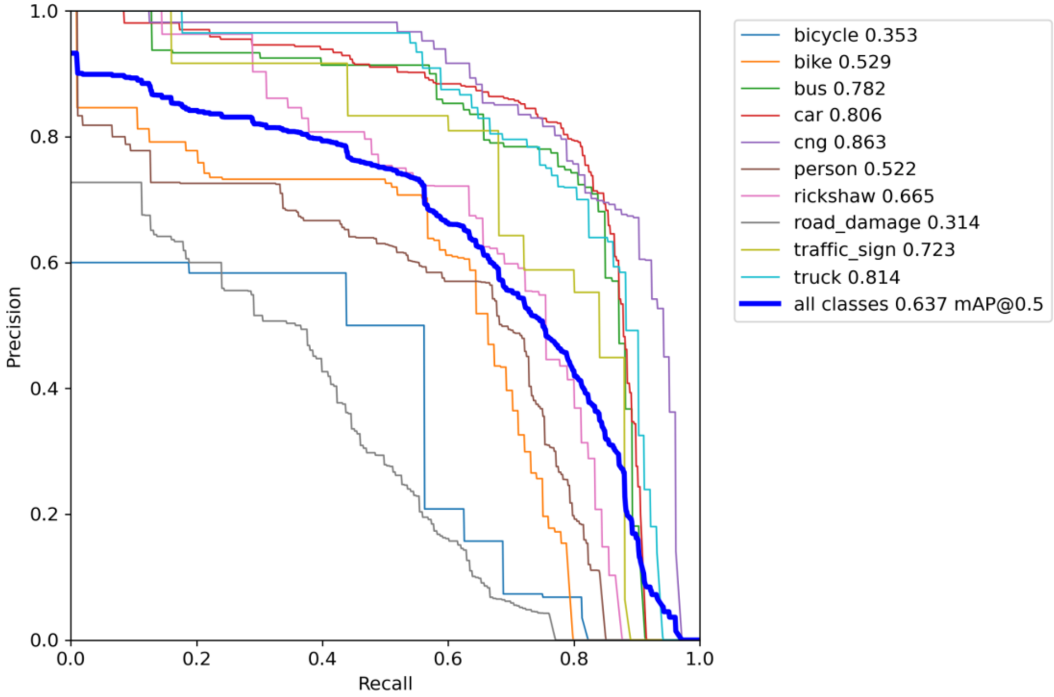


Figure 25: Precision Recall Curve

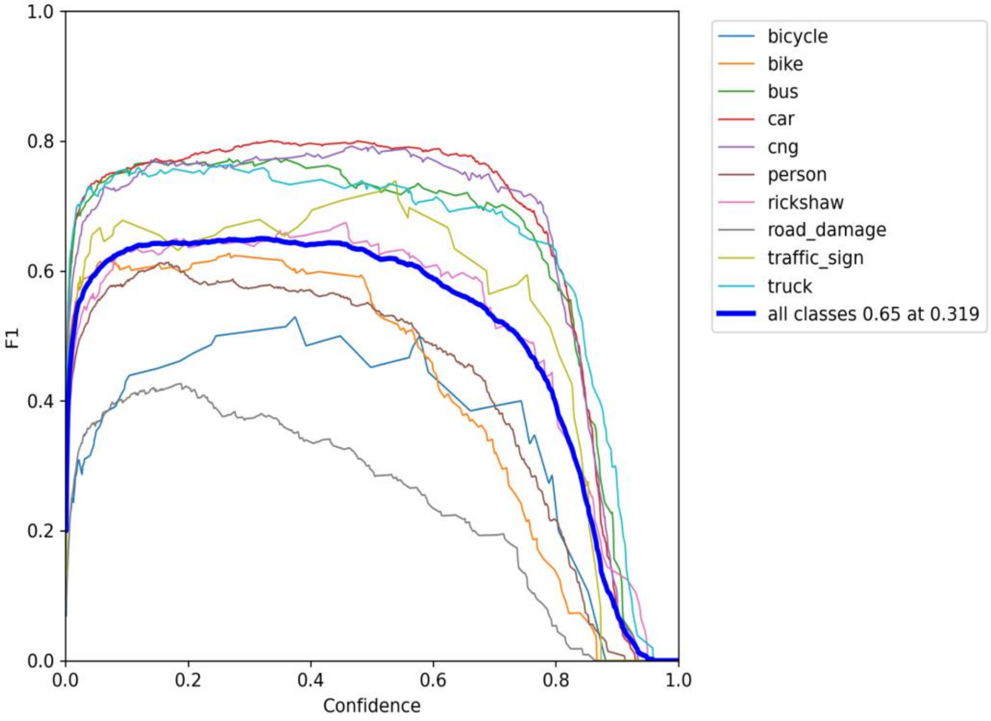


Figure 26: F1-Confidence Curve

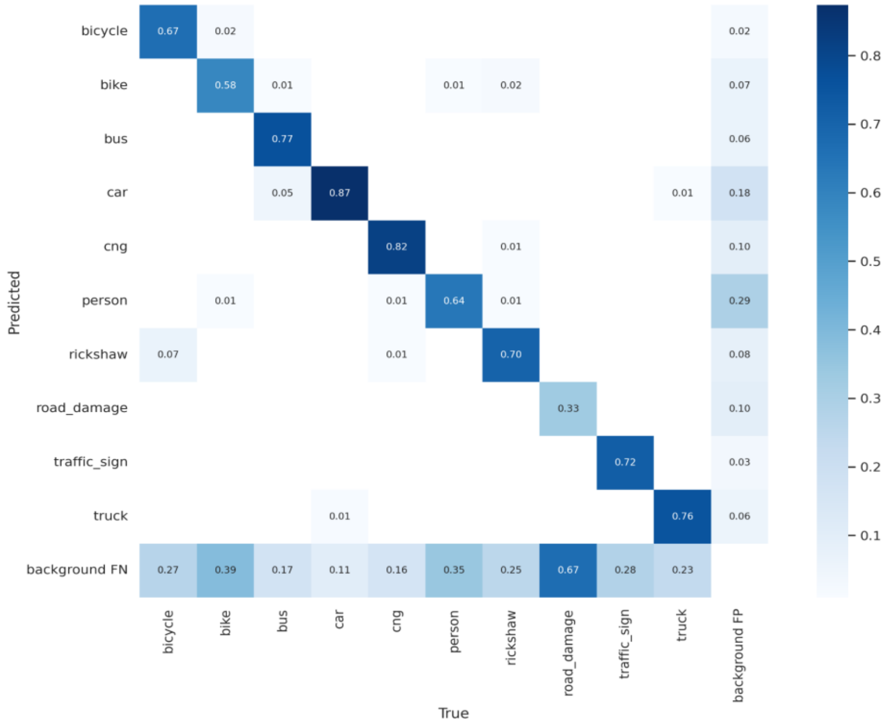


Figure 27: F1-Confidence Curve

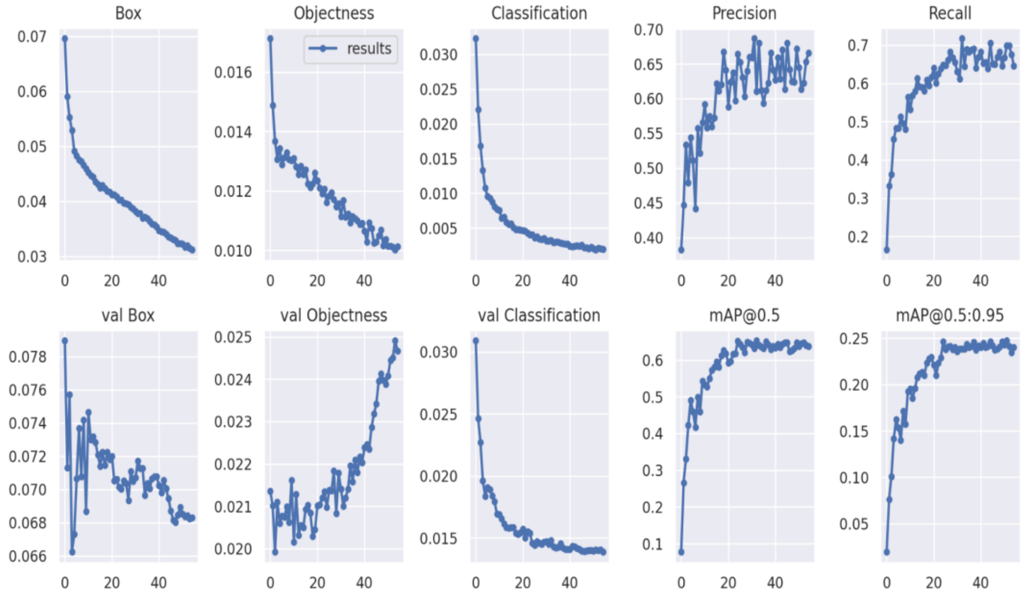


Figure 28: Result at a Glance

CHAPTER 5

5. CONCLUSION AND LIMITATIONS

5.1 CONCLUSION

Heavy traffic is a highly concerning issue because in Bangladesh.Road damage that is manually repaired is expensive and time-consuming. The ability to use an automated system to identify and track vehicles on which roads have more and less traffic will be beneficial for authorities in maintaining the traffic system. Additionally, it will assist motorists and pedestrians in avoiding congested roads because it saves time consuming for the people. So, from that idea, they can decide which road is mostly in need of a foot over bridge, and that is less necessary. Other nations already have procedures in place. It is challenging for Bangladesh to employ a system with significant costs. Therefore, I have attempted to create a system using my work that is minimal cost. I have employed deep learning for professional purposes. I have applied a convolutional neural network's transfer learning technique from deep learning. Because the code can be reused with transfer learning, the system will become inexpensive. Two convolutional neural network models, one algorithm, and all of them were transfer learning techniques employed by me. They are YOLOv8 and YOLOv7. All the model's mAP has been generated but among all of the two models, YOLOv8's mAP is higher.

5.2 LIMITATIONS

The result of the work is not very satisfying. It can be as a result of the video's viewpoint, which is based on Dhaka. To acquire a higher mAP, I will also try to take videos at the ideal angle and along with this work, I will try to get the flow of each vehicle separately and use other models to compare the mAP. I will also try to count vehicles in range and detect and recognize number plates on vehicles.

CHAPTER 6

6. REFERENCES

[1] Lin, C. J., Jeng, S. Y., & Lioa, H. W. (2021). A real-time vehicle counting, speed estimation, and classification system based on virtual detection zone and YOLO. *Mathematical Problems in Engineering*, *2021*, 1-10.

[2]Ghosh, A., Sabuj, M. S., Sonet, H. H., Shatabda, S., & Farid, D. M. (2019, June). An adaptive video-based vehicle detection, classification, counting, and speed-measurement system for real-time traffic data collection. In 2019 IEEE region 10 symposium (TENSYMP) (pp. 541-546). IEEE.

[3] Bhuiyan, T. A. U. H., Das, M., & Sajib, M. S. R. (2019). Computer vision based traffic monitoring and analyzing from on-road videos. *Glob. J. Comput. Sci. Technol*, *19*(2).

[4] Rahman, Z., Ami, A. M., & Ullah, M. A. (2020, June). A real-time wrong-way vehicle detection based on YOLO and centroid tracking. In *2020 IEEE Region 10 Symposium (TENSYMP)* (pp. 916-920). IEEE.

[5] Cao, W., Yuan, J., He, Z., Zhang, Z., & He, Z. (2018). Fast deep neural networks with knowledge guided training and predicted regions of interests for real-time video object detection. *IEEE Access*, *6*, 8990-8999.

[6] Tuhin, O., Siddika, F., Hossain, S., Azam, T., Mahmud, S., Hossain, S., ... & Islam, M. M. (2021). Traffic Management System Through Vehicle Detection and Counting. *Science*, *2*(4), 61-66.

[7]Jia, L., Wu, D., Mei, L., Zhao, R., Wang, W., & Yu, C. (2012). Real-time vehicle detection and tracking system in street scenarios. In Communications and Information Processing: International Conference, ICCIP 2012, Aveiro, Portugal, March 7-11, 2012, Revised Selected Papers, Part II (pp. 592-599). Springer Berlin Heidelberg.

[8] Hou, Y. L., & Pang, G. K. (2010). People counting and human detection in a challenging situation. IEEE

transactions on systems, man, and cybernetics-part a: systems and humans, 41(1), 24-33.

[9] Paul, M., Haque, S. M., & Chakraborty, S. (2013). Human detection in surveillance videos and its applications-a review. *EURASIP Journal on Advances in Signal Processing*, *2013*(1), 1-16.

[10]Lefloch, D., Cheikh, F. A., Hardeberg, J. Y., Gouton, P., & Picot-Clemente, R. (2008, February). Real-time people counting system using a single video camera. In *Real-Time Image Processing 2008* (Vol. 6811, pp. 71-82). SPIE.

[11] Kanrar, S., Agrawal, S., & Sharma, A. (2021). Vehicle detection and count in the captured stream video using machine learning. *Machine Learning Approaches for Urban Computing*, 79-112.

[12] Zuraimi, M. A. B., & Zaman, F. H. K. (2021, April). Vehicle detection and tracking using YOLO and DeepSORT. In *2021 IEEE 11th IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE)* (pp. 23-29). IEEE.

[13] Alpatov, B. A., Babayan, P. V., & Ershov, M. D. (2018, June). Vehicle detection and counting system for real-time traffic surveillance. In *2018 7th Mediterranean Conference on Embedded Computing (MECO)* (pp. 1-4). IEEE.

[14] Mo, Y., Han, G., Zhang, H., Xu, X., & Qu, W. (2019). Highlight-assisted nighttime vehicle detection using a multi-level fusion network and label hierarchy. *Neurocomputing*, *355*, 13-23.

[15]Feng, D., Haase-Schütz, C., Rosenbaum, L., Hertlein, H., Glaeser, C., Timm, F., ... & Dietmayer, K. (2020). Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges. *IEEE Transactions on Intelligent Transportation Systems*, *22*(3), 1341-1360.

[16] Ghosh, A., Sabuj, M. S., Sonet, H. H., Shatabda, S., & Farid, D. M. (2019, June). An adaptive video-based vehicle detection, classification, counting, and speed-measurement system for real-time traffic data collection. In *2019 IEEE region 10 symposium (TENSYMP)* (pp. 541-546). IEEE.

[17] Zhang, X., & Zhu, X. (2019, July). Vehicle detection in the aerial infrared images via an improved Yolov3 network. In *2019 IEEE 4th International Conference on Signal and Image Processing (ICSIP)* (pp. 372-376). IEEE.

[18] Yang, W., Li, Z., Wang, C., & Li, J. (2020). A multi-task Faster R-CNN method for 3D vehicle detection based on a single image. *Applied Soft Computing*, *95*, 106533.

[19] Hu, X., Wei, Z., & Zhou, W. (2021, March). A video streaming vehicle detection algorithm based on YOLOv4. In *2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)* (Vol. 5, pp. 2081-2086). IEEE.

[20] C.-Y. Chen, Y.-M. Liang, and S.-W. Chen, "Vehicle classification and counting system," in Proceedings of the 2014 International Conference on Audio, Language and Image Processing (ICALIP), pp. 485–490, Shanghai, China, July 2014.

[21]Seenouvong, N., Watchareeruetai, U., Nuthong, C., Khongsomboon, K., & Ohnishi, N. (2016, July). Vehicle detection and classification system based on virtual detection zone. In *2016 13th International Joint Conference on Computer Science and Software Engineering (JCSSE)* (pp. 1-5). IEEE.

[22] Tabassum, S., Ullah, M. S., Al-Nur, N. H., & Shatabda, S. (2020, June). Native vehicles classification on Bangladeshi roads using CNN with transfer learning. In *2020 IEEE Region 10 Symposium (TENSYMP)* (pp. 40-43). IEEE.

[23] Tabassum, S., Ullah, S., Al-Nur, N. H., & Shatabda, S. (2020). Poribohon-BD: Bangladeshi local vehicle image dataset with annotation for classification. *Data in brief*, *33*.

[24] Han, Y., Jiang, T., Ma, Y., & Xu, C. (2018). Pretraining convolutional neural networks for image-based vehicle classification. *Advances in multimedia*, *2018*.

[25] Vaddi, R. S., Boggavarapu, L. N. P., Anne, K., & Siddhartha, V. (2015). Computer vision based vehicle

recognition on indian roads. *International Journal of Computer Vision and Signal Processing*, *5*(1), 8-13.