Title of the Thesis

Intelligent Traffic Management System and Density Analysis with Artificial Intelligence

Submitted By

Nishat Ahmed Samrin

ID – 183-16-380

Department of Computing & Information System

Daffodil International University

Supervised By

Mr. Abdullah Bin Kasem Bhuiyan

Lecturer

Department of Computing & Information System

Daffodil International University

Department of Computing and Information System
Daffodil International University.
Dhaka, Bangladesh

Submission Date: 14/01/2023

APPROVAL

This thesis titled "**Intelligent Traffic Management System and Density Analysis with Artificial Intelligence"**, Submitted by **Nishat Ahmed Samrin**, ID No: 183-16-380, to the Department of Computing & Information Systems, Daffodil International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computing & Information Systems and approved as to its style and contents. The presentation was held on 14/01/2023.

BOARD OF EXAMINERS

Mr. Md Sarwar Hossain Mollah — **Chairman**
Associate Professor and Head
Department of Computing & Information Systems
Faculty of Science & Information Technology
Daffodil International University

Mr. Md. Mehedi Hasan — **Internal Examiner**
Lecturer
Department of Computing & Information Systems
Faculty of Science & Information Technology
Daffodil International University

Mr. Syed Tangim Pasha — **Internal Examiner**
Lecturer
Department of Computing & Information Systems
Faculty of Science & Information Technology
Daffodil International University

Dr. Saifuddin Mr. Tareeq

External Examiner

Professor & Chairman

Department of Computer Science and Engineering

Dhaka University, Dhaka

DECLARATION

I, **Nishat Ahmed Samrin**, hereby declare that the work in this dissertation titled **"Intelligent Traffic Management System and Analysis density with Artificial Intelligence"**; I have done this thesis under the supervision of **Abdullah Bin Kasem Bhuiyan**, Lecturer, Department of Computing and Information System (CIS) of Daffodil International University. I am also declaring that this thesis or any part of there has never been submitted anywhere else for the award of any educational degree like B.Sc., M.Sc., Diploma or other qualifications.

Supervised By

Mr. Abdullah Bin Kasem Bhuiyan
Designation: Lecturer
Department of CIS
Daffodil International University

Submitted By

Nishat Ahmed Samrin
ID: 183-16-380
Department of CIS

ACKNOWLEDGEMENT

It was my pleasure to research and complete the thesis paper on "**Intelligent Traffic Management System and Density Analysis with Artificial Intelligence**" I am grateful to Allah for his grace in my education.

I want to express my gratitude to my supervisor, **Abdullah Bin Kasem Bhuiyan**, Lecturer in the CIS Department at Daffodil International University, for his valuable time, advice, and inspiration to give this paper and my education my all. I want to thank my sir and send him my best wishes for his belief, steadfast guardianship, sound judgment, suitable outlook, and comprehension of various factors that helped make my thesis possible. He has done everything he can to ensure I know what is necessary to finish this thesis.

I also want to convey my heartfelt respect to **Mr. Sarwar Hossain Mollah**, the head of my department, for his helpful help at the beginning of the study and for making sure I understood the topic. His kind behaviors and helpful nature always encourage me to do better work.

Finally, I must respectfully thank **Mr. Mehedi Hassan**, Lecturer in the CIS Department at Daffodil International University, and all of my lecturers for their unfailing help and patience in getting this thesis work done.

DEDICATION

I Want to dedicate my work to my father **late Zafar Ahmed,** mother **Laila Zafar,** Without the blessings and unwavering support of my parents, the **late Zafar Ahmed and Laila Zafar, and** my bother, **Mr. Shaidul Islam**, I would not have been able to pursue this opportunity to research or study in order to make a positive impact on society.

ABSTRACT

Traffic density is turning into one of the major concerns issues around the world. Of excessive traffic, people are facing health and mental problems. Around the world, Bangladesh is 4^{th} country that has the larger traffic. It is continuously increasingly time-consuming, causing fuel consumption, stress, delays, mental health, physical health, and pollution. For improved signal management and efficient traffic control, it is also important to determine real-time traffic density on the roads due to its ever-increasing nature. One of the crucial components influencing traffic flow is the traffic controller. The result is that it becomes necessary to optimize traffic management to meet this rising demand better. Our suggested method intends to use real-time photos from the cameras at traffic intersections for image processing and AI-based traffic density calculation. It also emphasizes the algorithm for adjusting traffic signals depending on the number of vehicles. We have collected around 1727 data locally, which we have trained. Our class was Car, Bike, Bus, Truck, and CNG. We have used the YOLO3 model to solve this. And it achieved the best accuracy of 86%. We also did a simulation to understand our model better. Our top results are for all class precision of 86%, Recall of 0.741%, and mean average of 0.805%.

Table of Contents

List of Figures

CHAPTER 1

INTRODUCTION

1.1 Introduction

Traffic obstruction is becoming a significant critical issue today. Modernism is becoming more prevalent with time, and we are making progress in every industry, but how far have we come in traffic? It plays a big part in every country. The purpose of this thesis is to give focus on the current traffic scenario in Bangladesh. In Bangladesh, according to research, the average traffic speed has decreased over the past ten years from 21 kph to 7 kph, and by 2035, It might barely reach 4 kph, which is less than the speed of walking. Excessive traffic many causes occurring, such as excessive delays, physical health, fuel waste, mental health, losing track of time, and many more. Manually controlling and fixed time signals are major reasons for traffic. The other reason is that parking on the side of vehicles, over-tacking, don't move for a long time, and over vehicles are also reasons for massive traffic. Nowadays, most developed countries like Singapore, the United States, Australia, South Korea, Japan, and Thailand have already developed intelligent traffic management systems. The development of the system is different from another one, but they have the same motive to improve the traffic management system. It is a big challenge for the Bangladesh Govt. to expand the intelligent traffic system to every level.

In light of this, using Artificial Intelligence can give a better solution to monitoring and controlling traffic density in a better way. I have tried to give a solution using computer vision which will get real-time data from CCTV cameras installed at traffic signals after that, and the cameras will feed video. Then it will process the deep neural network to detect traffic and calculate the traffic density. After analysis, it will estimate and adjust the time schedule of traffic. The traffic light will change according to the estimated time. For betterment, traffic police can monitor the traffic. Therefore, the major contributions I have proposed in this work are:

- The economic effect may greatly impact how well traffic density is implemented. As a result, forecasting traffic density levels based on these variables enables decision-makers to make informed decisions and take appropriate action to address problems.
- An analysis of the model's performance is made in order to comprehend the proposed model's predictive power.
- In the proposed system, I have used the Vehicle detection module, Signal Switching Algorithm, Yolo5, Vehicle calculation, and Signal time changing to predict the traffic density and give the best solution for traffic management.

Several parts of this paper are organized below ways: Problem background, research objectives, motivation, research questionaries, expected output result, literature reviews of the existing works, the scope of the problems, challenges of this implementation, Proposed system overview, vehicle detection algorithm, Signal detection item, implementation requirement, working of the algorithm, simulation module, using of Yolo, Experimental results and discussion, comparison of models, performance evaluation, Conclusion and Future works in the last section.

1.2 Research Objective

- To analyze the density of traffic
- To develop the best intelligent traffic density control and monitoring system possible using computer vision
- To make decisions by the video collected from the CCTV camera.
- Implementing artificial intelligence and CCTV cameras in major cities to improve traffic control.

1.3 Motivation

Traffic is one of the most critical aspects of our regular lives. To do our regular duties, I have to travel every day. Every day, I had to deal with traffic problems like being late, stress, delays, mental and physical health effects, etc. Excessive traffic irritates the majority of office workers, students, and workers. After I ran into this problem, I was determined to learn more about traffic management, develop long-term solutions for analyzing traffic density, and find a good solution. This thinking, therefore, inspires me to do this.

1.4 Rationale of the Study

After doing a lot of research and analysis on traffic management systems and traffic density, I discovered the limitations of other people's work and the scope of our own. Research on traffic density with artificial intelligence is very rare around the world. This has not yet occurred in Bangladesh. I also saw that many researchers had already worked on their theoretical traffic solution research, but there was very little research on traffic density. So, if we can make a smart traffic system that can tell when there is much traffic, it will be very important for our country.

1.5 Research Questions

Research on traffic is very difficult because of tracking properly, data, and many more things. We also faced problems building models and finding hidden patterns in the traffic data. So, before beginning our research, we chose a few questions to focus on in order to complete our work.

1. Why do we need an automated traffic system?
2. To develop the best intelligent traffic density control and monitoring system possible using computer vision
3. How beneficial are the systems for Bangladesh?
4. What are the challenges of developing the system?
5. What specifications must be met while creating a system for traffic density analysis?
6. What is the comparison between the proposed model to existing research work?

1.6 Expected Output

Throughout this research, we will analyze traffic density and develop an automated system for better traffic management. Expect to discover based on the research questions is -

1. To analyze the density of traffic
2. To develop the best intelligent traffic density control and monitoring system possible using computer vision
3. Improving traffic control by installing CCTV cameras and artificial intelligence in key cities.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

In this CHAPTER 2, I'll discuss literature reviews, the challenges of our research, and how it differs from earlier studies. I'll compare our study to other research studies' methodologies, end results, conclusions, accuracy, etc. The difficulties we ran into while conducting our research, as well as the lessons we learned, are discussed in the difficulties section. In the part on difficulties, I'll go over how I developed a density analysis model for better traffic management.

2.2 Related Work

(Gandhi et al., 2020) researchers are investigating the intelligent management of traffic lights using computer vision. They aimed to employ image processing and AI to analyze real-time data from the intersection cameras to determine the amount of traffic. They have tried calculating the real-time traffic density and effective solutions for better traffic management. They demonstrated how traffic is now managed manually, citing the usage of timers and electronic sensors in traffic signals as one of the main causes for the system's inability to manage traffic congestion effectively. To solve this problem, they proposed a system; first, they will fit CCTV cameras at traffic junctions to help take live images and calculate the real-time density by processing images and identifying objects. Traffic density will be sent to the server for calculating signal time like waiting time, running time, etc. after analysis; signal time will update time by time. They have tried to help optimize better signal timing, cleared traffic properly, made the System dynamic, and reduced unwanted delays, congestion, and waiting time. The author of (Alpatov et al., 2018a) of attention is on analyzing road situations to detect vehicles and count them to secure traffic control. They have used several algorithms like – Road marking detection method and vehicle detection and counting algorithm. Using smart cameras, they designed algorithms to process images with the help of cameras and create a proper detection system. On the other hand, (Osman et al., 2017) we are referring to; they attempt to make an advanced management traffic system with computer vision for a cross-section of roads. They suggest two methods with low cost. Hardware was used to build the first one, but no hardware was used in the second design. Their proposed system will count the number of vehicles and determine the ideal amount of time to wait before flashing a red signal light and running a green signal light based on the number of vehicles. Another research work (Pramanik et al., 2021) uses the google maps datto model for a developing country's congestion. As technology advances, they can now collect data from the Google Maps traffic layer for the least amount of money. They employ models like Support Vector Regression with Graph, Historical Averages Support Vector Regression, and Auto-Regressive Integrated Moving Average (ARIMA). This paper first collects data from the google maps traffic layer and then automatically captures customized traffic layer images. Then it goes to manual annotation and takes out density data pixel-wise after processing the traffic data matrix. Lastly, it gives results by using statistical and machine learning models. (Nam et al., 2020a) developed the LSTM approach to estimate proper traffic density, which is also a non-parametric method. They evaluate some algorithms also, but they found LSTM is best. They demonstrate that their model provides a better traffic management and density solution. To estimation of density, they update times, concentrate on the dataset's composition, moving-horizon size, and congestion criteria.(Nemade, 2016)The suggested approachaddresses the main traffic issues while offering customized solutions for Indian roads and automobiles. The 2-lines method and vehicle classification algorithm work together to classify and keep track of the number of different vehicle types on Indian roads in addition to estimating traffic density.

2.3 Bangladesh Perspective

Currently, around the world, Bangladesh traffic is in the ninth position. Our country is now developing in the many sectors but still traffic didn't show any magnificent changes. Over-population, over-vehicle is the major factor of traffic mismanagement. Now it's necessary to convert our manual system to automated system. Therefore, taking the benefited. We can now state that the following technology will be a fantastic invention from Bangladesh's point of view.

2.4 Scope of the Problem

The poor traffic management could not able to handle the vehicle. Traffic increases for many reasons like road defects, manual traffic control, higher population, higher rate of private vehicles, etc. According to our research, people nowadays are very concerned about traffic. Manually traffic handling is not managed correctly. The working scope of our study is very broad-ranging. We can use CCTV cameras with deep learning processes to measure the density of traffic, and according to the analysis, we can change traffic signals. As Bangladesh does not have an automated traffic automation system and this kind of work has not been done yet, there is a big opportunity to find out the problems and solve them. By developing this system, we can make decisions and handle traffic management. It can be a future prediction system in traffic density detection and quality measurements.

2.5 Challenges of this Work

I had to overcome numerical challenges to complete this thesis. Our first challenge was data collection. It's a difficult part where I can measure the exact traffic. With a small amount of data, it was hard to train the model properly. Unfortunately, we were unable to locate a dataset that might satisfy our need for situational traffic data. Then we choose to manually gather data from the internet and supplement it in order to create a sufficient volume of data to train a YOLO model. The next challenge is to select the optimal algorithm or method for building a model that could support image categorization for traffic situations. After analysis, I found YOLO, R-CNN, HOG, R-FCN, which can assist us in performing this task. The next obstacle we had to overcome was training our model on a reliable computer. Since it takes many hours to train a model on the local CPU, we looked for a solution since we lacked a high-end GPU. Finally, we discovered a solution called Pygame, Google Collab, and Visual Studio, and used a paid cloud GPU to train our model quickly. So, while our assignment was challenging, we were able to finish it.

CHAPTER 3

METHODOLOGY

3.1 Proposed Traffic System Overview

Our proposed system for traffic captures an image from the CCTV cameras at traffic crossings to assess the real-time traffic density using object detection and image processing. This image is supplied to the vehicle detection and vehicle count algorithm, which incorporates YOLO, as seen in Fig. 1. To find adequate traffic density, the number of each form of vehicle, such as car, bike, bus, CNG, and trucks, is counted. The signal-switching algorithm uses this density along with other variables to set the green signal timer for each lane. Accordingly, the red signal times are changed. To prevent starving of a given route, the length of the green signal is regulated to a maximum and lowest value. A simulation is also developed to demonstrate the system's effectiveness and contrast it with the present static system.

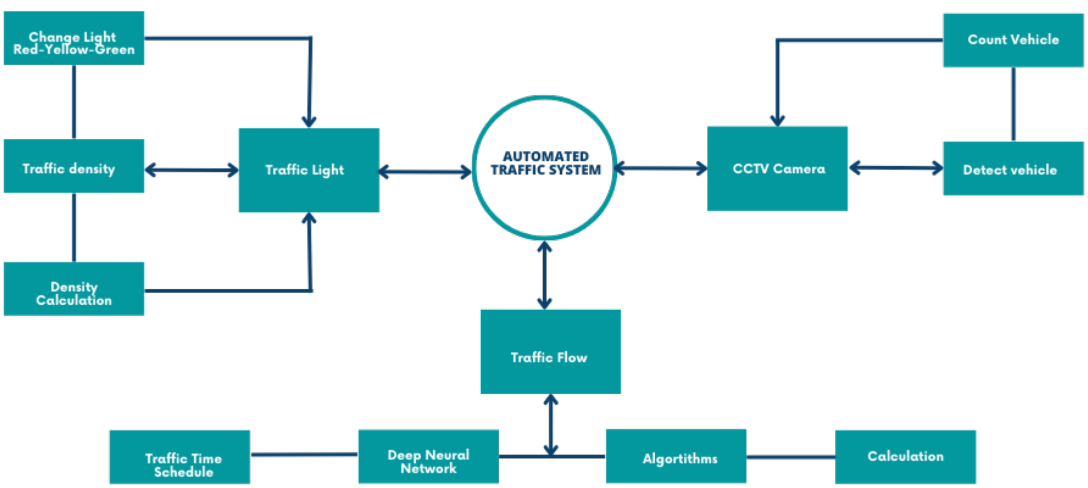


Fig 1: Proposed Traffic System Overview

3.2 Data Collection of Traffic

Data collection was the most challenging part. I used some new and old datasets to train our proposed model for traffic management study. We search for as much data as possible because we know that minimal data cannot provide a decent model. Unfortunately, we could not locate a dataset that might satisfy our need for situational traffic data. Then we manually gather data from the internet and supplement it to get a sufficient volume of data to build a Yolo model. Our dataset has five classes: car, bus, truck, CNG, and bike. There are 1727 total images in this dataset; 1401 are used for training, while 326 are used for testing. The remaining 20% of the data were used for testing, while the remaining 80% were used for training.

3.3 Data Preprocessing

Our team gathered the entire dataset manually using a cell phone while the remaining information was obtained online. For us, this project's difficult portion involved pre-trained tasks with the same dimensions and requirements. For matching, we had to use quad dimensions. The image is cropped to a fixed size of 416 × 416, calculated from the bottom type's height to the image. To reduce the number of objects in the graphic, we first cropped the entire image in a variable quad dimension. The needed size for the cropped image that incorporates label images in the preprocess is the full fling. The Yolo program examines an image of the car during this phase.

3.4 Data Augmentation of Traffic

Data augmentation is used in our model because without augmentation model will not perform well. Although data augmentation in computer vision is not new, it has recently moved to the forefront of cutting-edge modeling. YOLOv3 uses several data augmentation strategies and a brand-new, cutting-edge image detection model to improve the model's performance on COCO, a well-known image detection benchmark that includes a wide variety of images and object classes. By improving the picture classification procedure, a significant amount of overfitting and underfitting of the model to the training data can be avoided. I have to use the Roboflow tool for data augmentation.

It is a very powerful tool where we can load data, labeling data, data augmentation, data process, train, and deploy.

3.3 Data Labelling of Traffic

By manually labeling images that were scraped from Google and added to the dataset of traffic for the model's training, LabelIMG. LabelIMG is a graphical image annotation tool that helps to label data. Next, pre-trained weights that were obtained from the YOLO website were used to train the model. The setting of the.cfg file used for training was changed in accordance with the specifications of our model. By altering the 'classes' variable, the number of output neurons in the final layer was changed to correspond to the number of classes the model is intended to recognize. Our system has five of these: a car, a bicycle, a bus, a truck, and CNG. After these setup changes, the model was trained until the loss was significantly smaller and no longer seemed to be decreasing. The weights have been changed to suit our demands after the workout. The OpenCV library was then used to detect automobiles using these program-entered weights. A threshold specifies the least level of assurance required for efficient detection. After the model has been loaded and fed an image, it outputs the results in a JSON format or as key-value pairs, where labels serve as the keys and the values are the confidence and coordinates of the labels. Again, using the provided labels and coordinates, OpenCV may produce the bounding boxes on the photographs.

3.5 Vehicle Detection Module

In our suggested traffic system, I have used YOLO, which means You only look once. It uses for detecting objects in real time. After analysis, some algorithm I found it the best because it provides better performance in accuracy and gives better processing time. To detect the vehicle, used to custom YOLO model, which can identify a variety of vehicles, including cars, buses, bikes, trucks, and CNG. To recognize objects instantly, a clever convolutional neural network (CNN) called YOLO is used. The algorithm divides the image into regions, predicts bounding boxes and probabilities for each region, and applies a single neural network to the entire image. Utilizing anticipated probabilities, these bounding boxes are weighted. YOLO is well-liked because it can operate in real-time and achieve excellent accuracy. The algorithm "only looks once" at the image in the sense that it only needs to run one forward propagation run through the neural network to generate predictions. Following non-max suppression, it outputs the bounding boxes and the detected items. Only one CNN employs YOLO to predict multiple bounding boxes and class probabilities for each box simultaneously. The YOLO backbone CNN can be further streamlined to speed up processing.

An open-source neural network framework called Darknet was created in C and CUDA. It supports CPU and GPU calculation, is quick to set up, and is fast. Using Dark Net, YOLO tops out at 72.9 percent. The traffic dataset used to train the model was produced by manually extracting the images from Google and categorizing them using the graphical image annotation program LabelIMG. The model was then trained using pre-trained weights that were downloaded and trained using pre-trained weights from the YOLO website. According to the requirements of our model, the .cfg file configuration used for training was modified. By altering the 'classes' variable, the number of output neurons in the final layer was changed to correspond to the number of classes the model is intended to recognize. Our system has four of these: a car, a bike, a bus, a truck, and CNG. Following that, the program's weights were utilized to detect automobiles using the OpenCV library. A threshold specifies the least level of assurance required for efficient detection. After the model has been loaded and fed an image, it outputs the results in a JSON format or as key-value pairs, where labels serve as the keys and the values are the confidence and coordinates of the labels. Once more, using the names and locations given, OpenCV can create the bounding boxes on the photos.

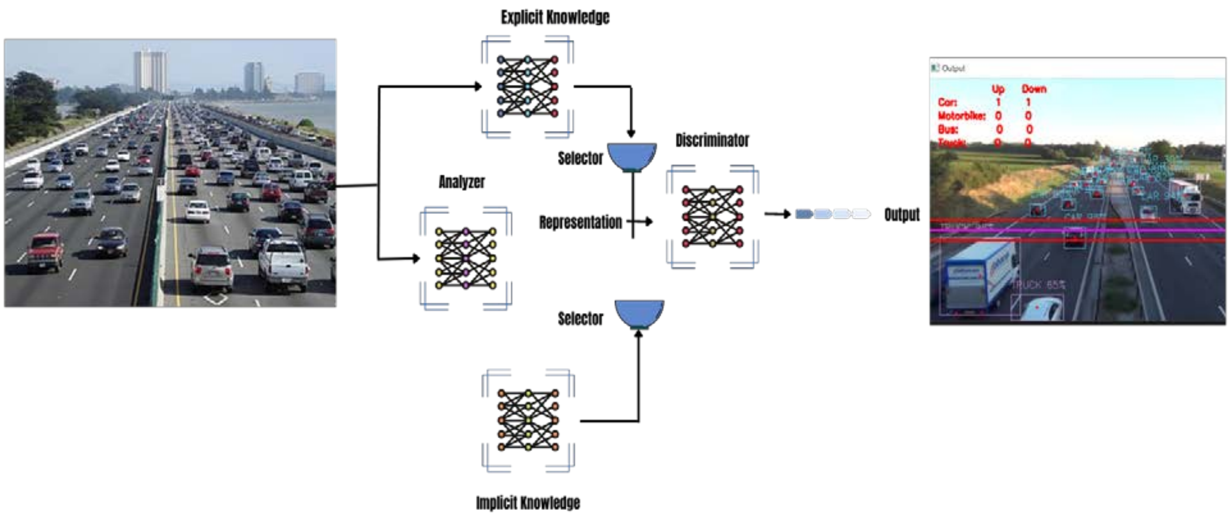


Fig 2: Our model block diagram

Fig. 2 demonstrates test pictures where our car detection model was used. The figure's right half, which includes bounding boxes and accompanying labels, is the outcome after the image has been processed with the vehicle detection model. The original image may be seen on the figure's left side.

3.6 Signal Switching Algorithm

The vehicle detecting module returns traffic density data to the signal switching algorithm, which sets the green signal timer and updates neighboring lights red signal timers in accordance. The timers are used to cycle through the signals as well. The algorithm receives information regarding the vehicles that the detection module picked up as input, as was detailed in the prior section. The key is the label of the detected item, and the data in JSON format are the confidence and coordinates. The total number of vehicles in each class is then calculated using this data. The red signal times of the other signals are then modified as appropriate. Next, the signal is given its green signal time. Any number of signals at a junction can be accommodated by scaling the algorithm up or down. Red to

Green to Yellow to Red is the signal order in the suggested traffic system. The signal switching algorithm's formula is -

$$\mathbf{GST} = \sum_{\text{vehicleClass}} \frac{\left(\text{NumOfVehicles}^{\text{vehicleClass}} * \text{AverageOfTime}^{\text{vehicleClass}}\right)}{(\text{NoOfLanes} + 1)}$$

Below is the whole meaning explains of the formula:

GST = Green Signal Time

NumOfVehicles = The number of cars, bikes, CNG, trucks and buses in each class NumOfVehicles represent the vehicle detection module that counted at the signal.

AverageTime = AverageTime measures the typical amount of time it takes a vehicle in that class to cross a junction.

NoOfLanes = NoOfLanes has how many lane there are at the traffic point.

During algorithm development, the following elements were taken into traffic management:

1) The algorithm's processing time to determine traffic density and the subsequent length of the green light determines when a picture should be collected.

2) Number of Total lanes of traffic

3) A Total number of each type of vehicle, such as cars, bicycles, CNG vehicles, trucks, and buses.

4) Time added due to the over amount of vehicles, but it has limited time for reducing traffic of other lanes.

5) The typical speed of each type of vehicle at the moment the green light turns on has been noted for more precise traffic assessments.

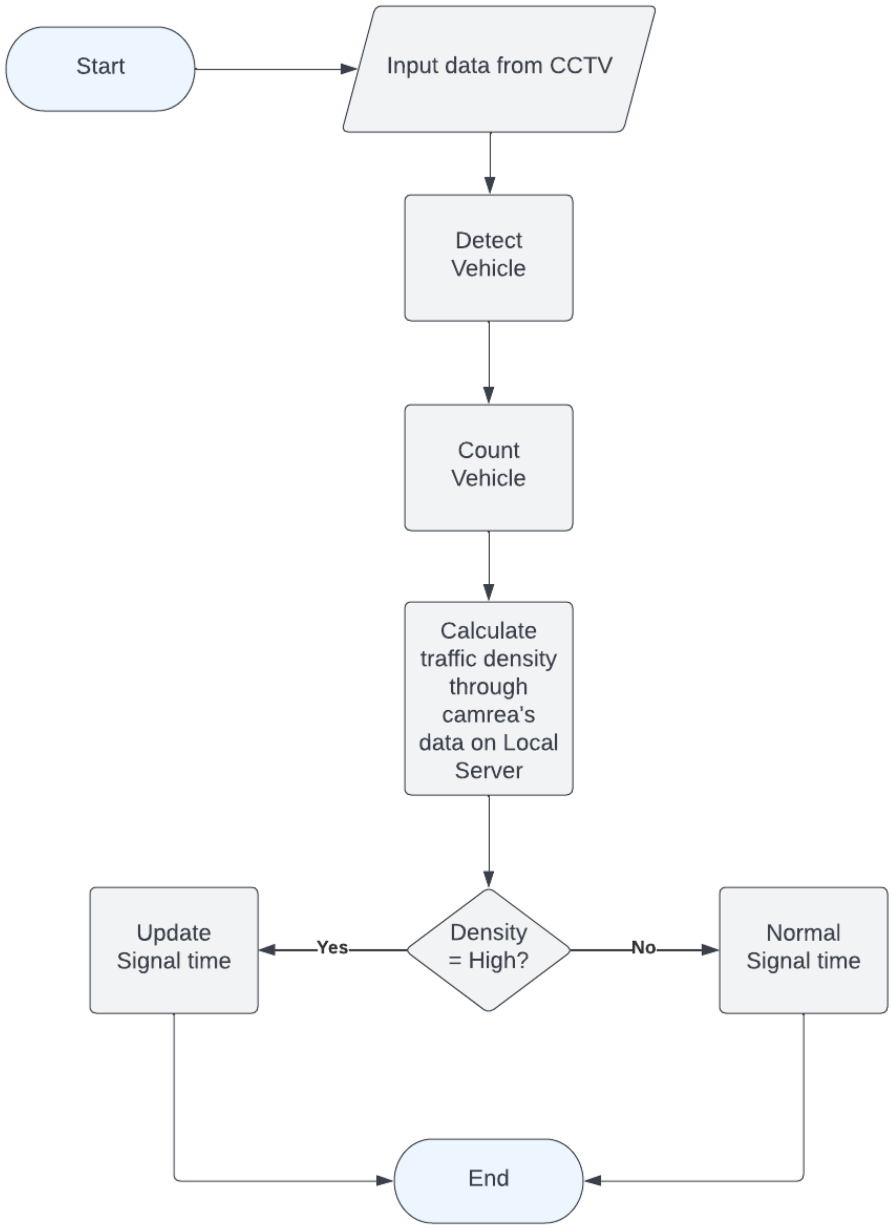


Fig 3: Flowchart of signal switching of traffic

3.7 Implementation Requirement

After carefully analyzing theoretical concepts and techniques, there is a set of requirements for this traffic density detection task. The following things will likely be needed:

Hardware and Software Prerequisites

- Running System (I have used Windows)
- Hard Disk (16 GB)
- Random Access Memory (Ram minimum 16 GB)
- Graphics Processor Unit (GPU)

Developing Tools

- Python Environment
- PyGame
- Visual Studio Code
- Google Collab

3.8 Simulation Module

To better understand our project, we have developed a simulation to simulate real-time traffic with scratch using PyGame. It is simpler to visualize the proposed model and compare with the static model that is already in place. There are 4 traffic signals at a 4-way intersection in total. A timer on top of each signal indicates how long it will be before changing from green to yellow, red to green, or yellow to red. Alongside each light, a number of vehicles that have gone through the intersection are also shown.Vehicles of various shapes and sizes are cars, buses, bikes, trucks, CNGs, and others arrive from every angle. Some of the vehicles in the rightmost lane turn and cross the intersection to increase the realism of the simulation. Random numbers are also utilized to determine whether a vehicle will turn when it is generated. A timer with an indication how much time has passed since the simulation started is also included. We have adjusted time elapsed: 300 second. Maximum red light will be remained 60 seconds and minimum will be 15 second. Here all times are, default red 150, default yellow 5, default green 20, default minimum 15, number of signals are 4. Speed of vehicles are car 2 sec, bike 1 sec, cng 2.25 sec, bus 2.5sec and truck 2.5 sec.

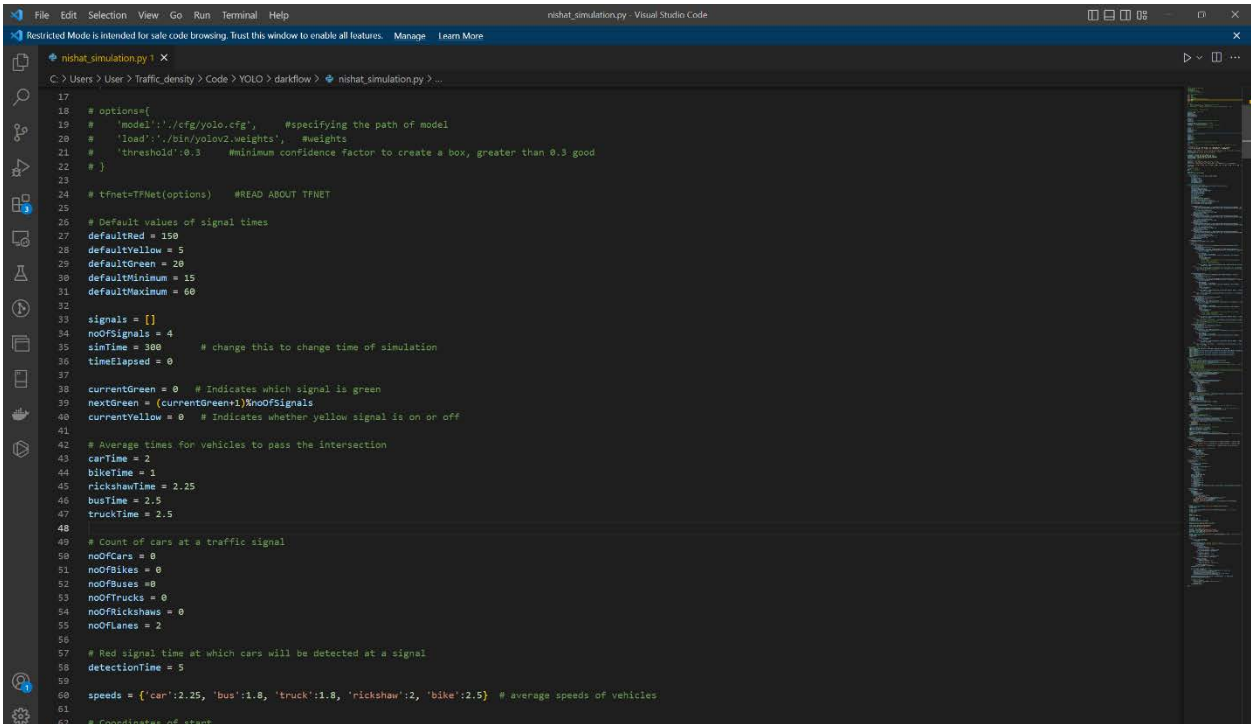


Fig 4: Code sample of simulation

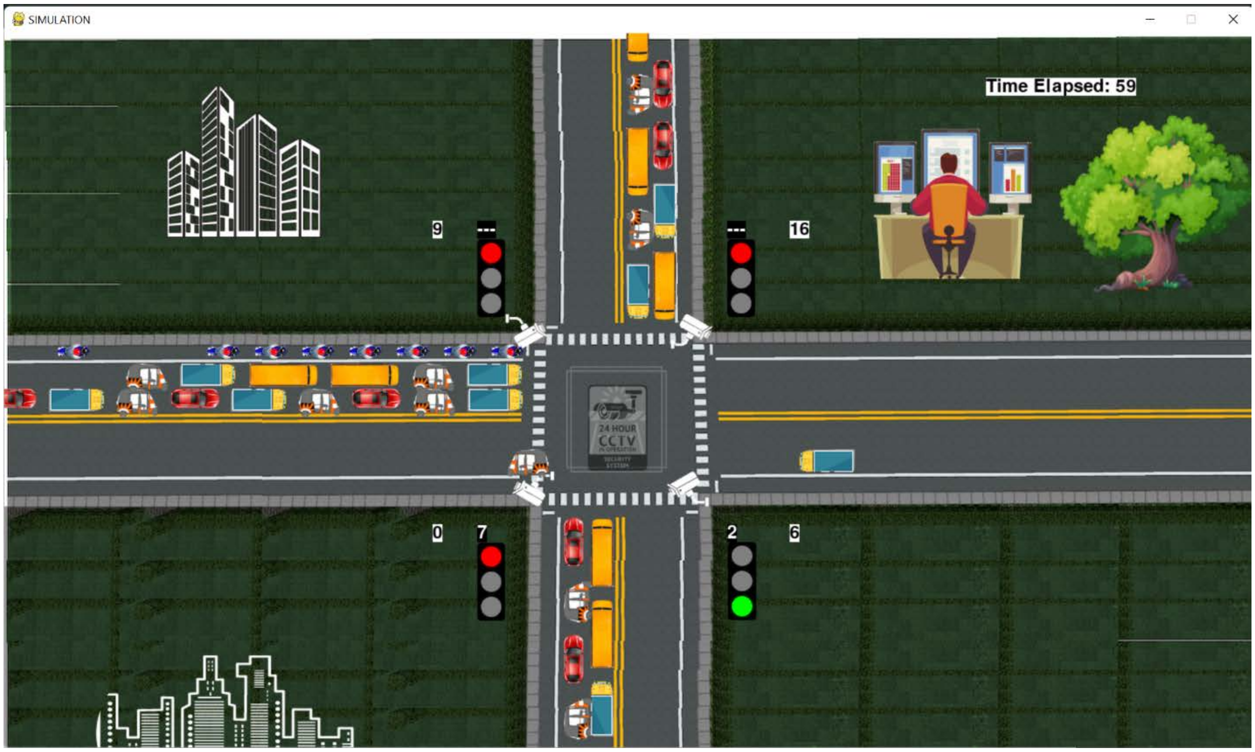


Fig 5: Simulation of the traffic management system

3.9 Real-Time Module

To better understand detection, we have tried the real-time module also, where the vehicle is detecting and counting. We have used vehicle algorithms and counting algorithms to track vehicles. We collect some data from Bangladesh road to put the technique into practice and run our experiments. The mobile cameras were used to collect footage of traffic. Figure 4 shows. A calculation was used to determine the distance between two parallel lines drawn perpendicular to the direction of traffic.

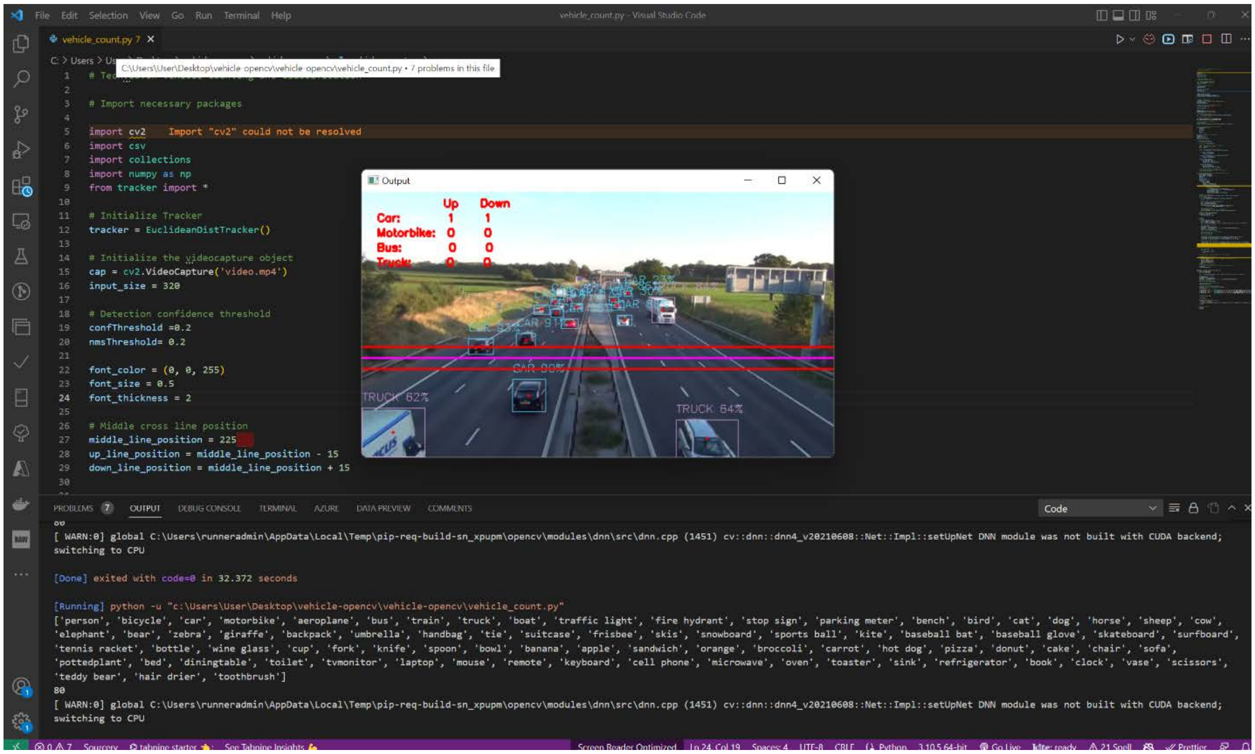


Fig 6: Real-Time detection of the traffic management system

Fig 7: Before Real-Time detection

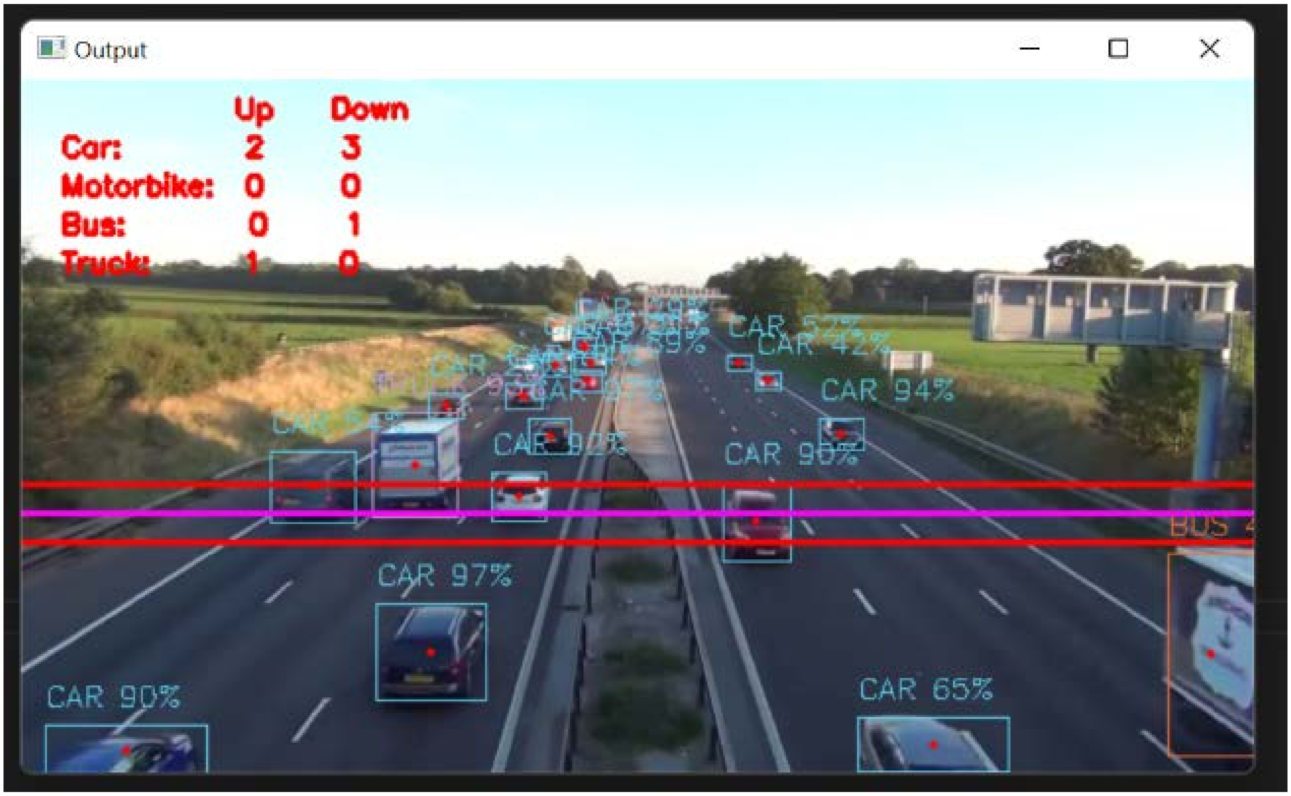


Fig 8: After Real-Time detection

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

In this CHAPTER 4, we'll go through with our suggested traffic model that how it was created or trained and the results it produced. We will contrast the performance of our model with some already trained models, including CNN, LSTM, Yolo, and OpenCV. All our work has been done on Google Collab, Visual studio, PyGame, and Roboflow.

4.2 Experimental Dataset

Our dataset has 1727 training images on our dataset. We have tested 326 images, and remain images were used for the train. In images, I have tried to maintain all class images like bus, car, bike, CNG, and truck.

```
Model Summary: 261 layers, 61518970 parameters, 0 gradients
               Class     Images     Labels          P          R     mAP@.5 mAP@.5:.95: 100% 21/21 [00:05<00:00,  4.03it/s]
                 all        326       2731      0.834       0.76      0.809      0.406
                bike        326        392      0.803      0.605       0.68      0.274
                 bus        326        735      0.852       0.82      0.881      0.534
                 car        326        934      0.837      0.784      0.833      0.402
                 cng        326        489      0.855       0.81      0.857      0.428
               truck        326        181       0.82      0.779      0.794      0.394
wandb: Waiting for W&B process to finish... (success).
```

Fig 9: Sample of Model summary

4.2 Evaluation of Proposed Traffic System

The vehicle detection module was tested using several test video and image sets with various numbers of vehicles, such as buses, cars, bikes, CNG, and trucks. The detection accuracy was 86%. In Fig. 3, several test results are displayed, like vehicle count, vehicle detection, density analysis, and update signal timer. This is adequate but not the best. The absence of a suitable dataset is the main cause of the low accuracy. Using data from genuine traffic camera footage to feed the model will increase the system's accuracy.

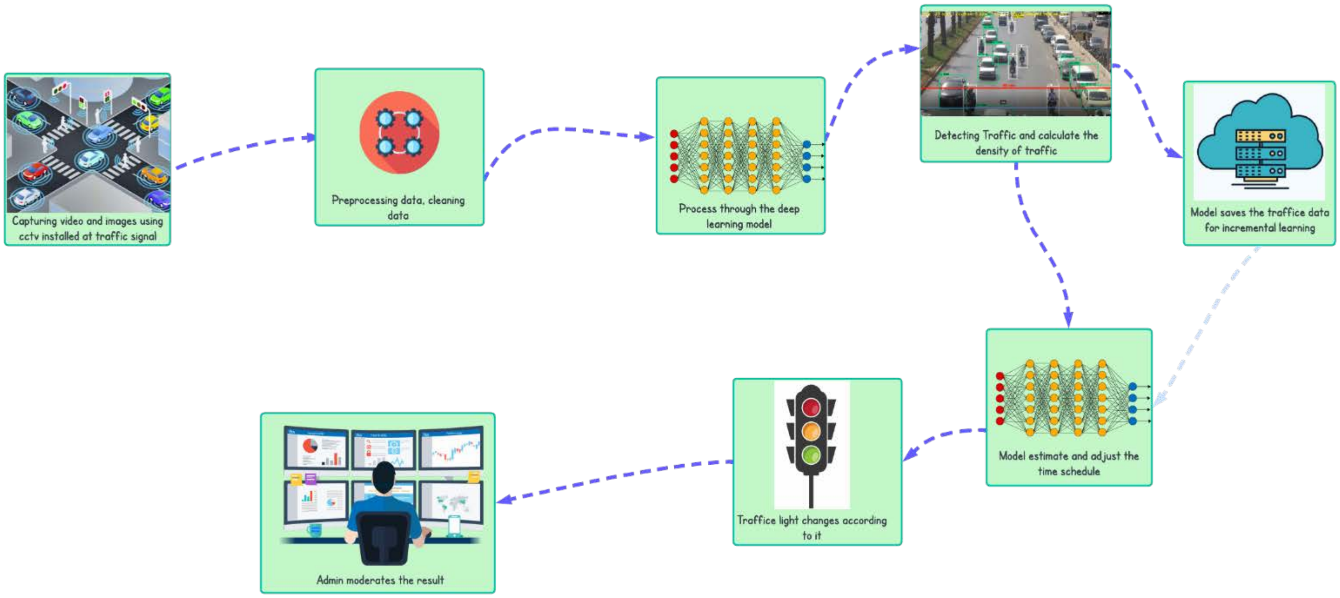
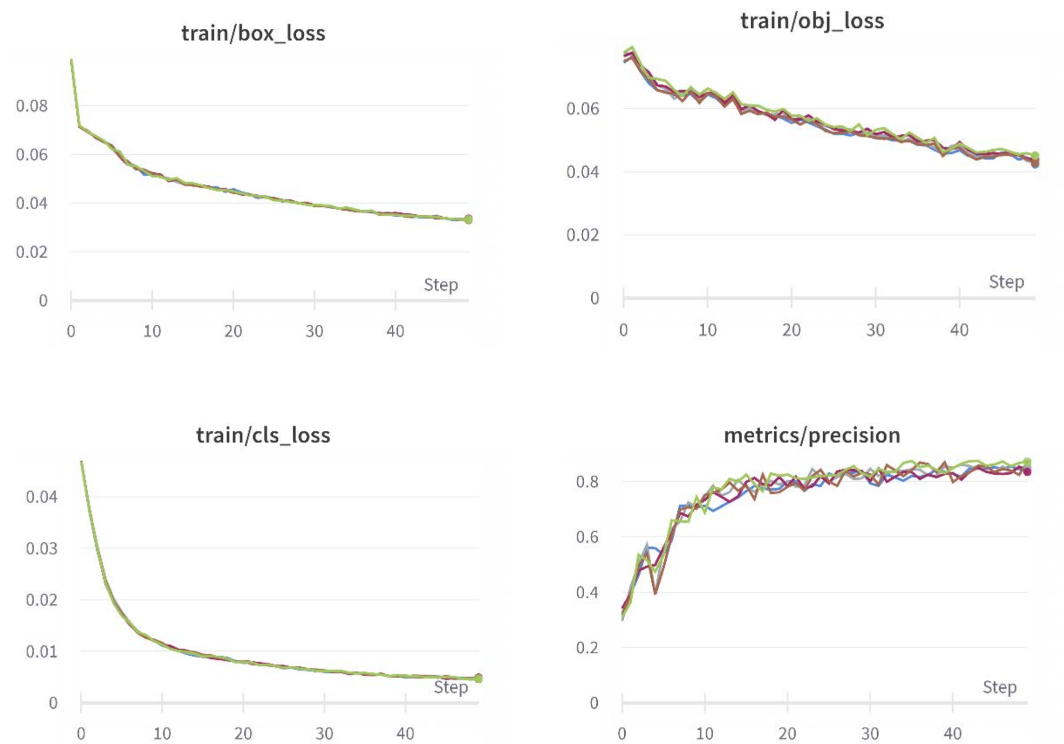


Fig 10: Methodology of proposed traffic system

4.3 Performance Evaluation



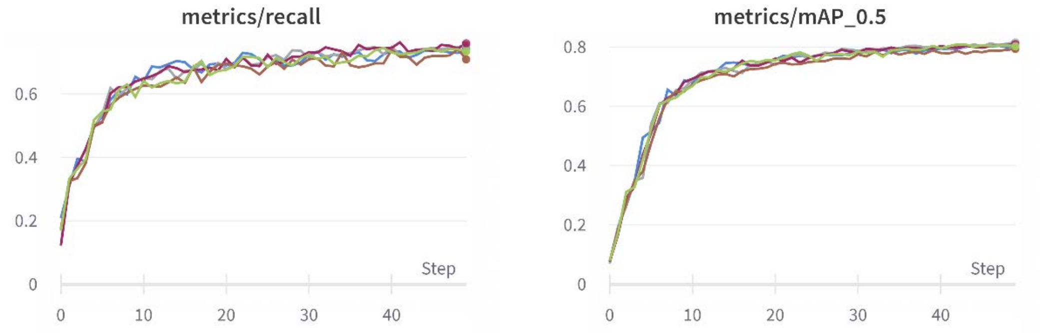


Fig 11: Model Metrics Result

In our dataset, 80% of the images were training sets, and the remaining 20% were test sets. So, we have 326 images of testing and a total of 1401 images for training. We coded and trained our model entirely in Google Collab notebook using the premium cloud GPU. GPU with tensor cores from Nvidia. We use Wandb software for data visualization. Wandb arranges and evaluates your trials in machine learning. It weighs less than a toolbox for tensor boards. Wandb records your model's hyperparameters and output metrics with a few lines of code and provides you with all visual charts for training, model comparison, accuracy, etc. Our training program's objective was to minimize cross-entropy. Therefore, we chose a batch size of 8 and 50 epochs to train the model and evaluate its performance. On the training set, our suggested system provides 86 percent accuracy, and on the test set, 85 percent accuracy. Figure 5 shows the metrics and outcome of our model's accuracy, Recall, means of precision, box loss, and object loss.

Fig 12: Vehicle detection result of multiple type

4.4 Result discussion of model

This section describes the results that each classifier produced. The following provides detailed descriptions and analyses.

Here,

P means Precision

R means Recall

MaP means Mean Average Precision

Precision: The percentage of similar instances among the recovered examples is known as precision. An algorithm has returned much more significant outcomes than unimportant ones when it has a high prediction precision.

Precision formula is -

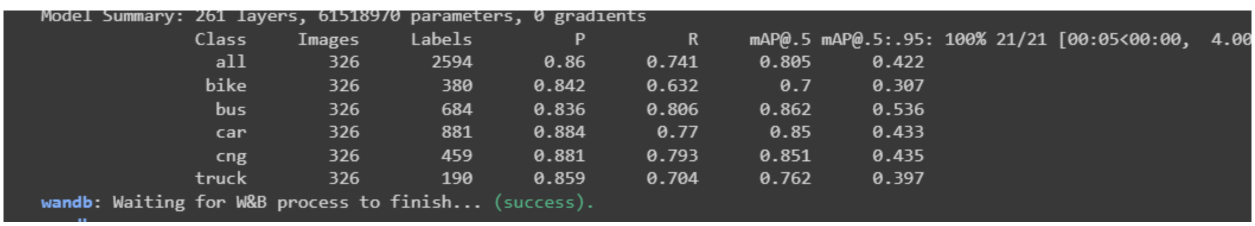
$$\text{Precision} = \frac{TP}{TP+FP}$$

Recall: Recall healps to measure positivle values proportion of correctly classified to all actual class-yes observations.

Recall formula is -

$$\text{Recall} = \frac{TP}{TP+FN}$$

MAP: To evaluate object identification methods like Fast R-CNN, YOLOand many more. Mean Average Precision (mAP) is utilized.



```
Model Summary: 261 layers, 61518970 parameters, 0 gradients
               Class     Images     Labels          P          R     mAP@.5 mAP@.5:.95: 100% 21/21 [00:05<00:00,  4.00
                 all        326       2594       0.86      0.741      0.805      0.422
                bike        326        380      0.842      0.632        0.7      0.307
                 bus        326        684      0.836      0.806      0.862      0.536
                 car        326        881      0.884       0.77       0.85      0.433
                 cng        326        459      0.881      0.793      0.851      0.435
               truck        326        190      0.859      0.704      0.762      0.397
wandb: Waiting for W&B process to finish... (success).
```

Fig 13: Sample of result

Table 1: RESULT OF CROSS VALIDATION FOLD 1

Fold	Class	Images	Labels	Precision	Recall	MaP.5
	all		2594	0.86	0.741	0.741
	bike		380	0.842	0.632	0.7

1	bus	326	684	0.836	0.806	0.862
	car		881	0.884	0.77	0.85
	cng		456	0.881	0.793	0.851
	truck		190	0.859	0.704	0.762

Table 1**,** displays the outcome of fold 1. We have tested 20 percent and trained our model 80 percent . We have total of 5-fold to test. Our batch size was 8, and Epoch was 50. In fold 1, we have taken all classes and given individual outcomes. For entire class, labels were 2594; accuracy was 0.86, Recall was 0741, MaP. 5 was 0.741. For all bikes, labels were 380, accuracy was 0.842, Recall was 0.632, MaP. 5 was 0.7. For entire bus, labels were 684, accuracy was 0.836, Recall was 0806, MaP. 5 was 0.862. For every car, labels were 881, accuracy was 0.884, Recall was 0.77, MaP. 5 was 0.85 For all cng, labels were 456, accuracy was 0.881, Recall was 0.793, MaP. 5 was 0.851. For entire truck, labels were 190, accuracy was 0.859, Recall was 0704, MaP. 5 was 0.762.

Table 2: RESULT OF CROSS VALIDATION FOLD 2

Fold	**Class**	**Images**	**Labels**	**Precision**	**Recall**	**MaP.5**
2	all	326	3071	0.872	0.709	0.793
	bike		464	0.831	0.531	0.624
	bus		758	0.882	0.792	0.876
	car		1144	0.861	0.755	0.845
	cng		528	0.901	0.74	0.809
	truck		177	0.877	0.725	0.811

Table 2, displays the outcome of fold 2. We have tested 20 percent and trained our model 80 percent . We have a total of 5-fold to test. Our batch size was 8, and Epoch was 50. In fold 2, we have taken all classes and given individual outcomes. For all classes, labels were 3071; accuracy was 0.872, Recall was 0709, MaP. 5 was 0.793. For all bike, labels were 464; accuracy was 0.831, Recall was 0.531, MaP. 5 was 0.624. For all bus, labels were 758, accuracy was 0.882, Recall was 792, MaP. 5 was 0.876. For every car, labels were 1144, accuracy was 0.861, Recall was 0.755, MaP. 5 was 0.845 For all cng, labels were 528, accuracy was 0.901, Recall was 0.74, MaP. 5 was 0.809. For all truck, labels were 177, accuracy was 0.877, Recall was 0725, MaP. 5 was 0.811.

Table 3: RESULT OF CROSS-VALIDATION FOLD 3

Fold	**Class**	**Images**	**Labels**	**Precision**	**Recall**	**MaP.5**
3	all	326	2731	0.834	0.76	0.809
	bike		392	0.803	0.605	0.68
	bus		735	0.852	0.82	0.881
	car		934	0.837	0.784	0.833
	cng		489	0.855	0.81	0.857
	truck		181	0.82	0.779	0.794

Table 3, shows the outcome of fold 3. We have tested 20 percent and trained our model 80 percent . We have a total did 5-fold to test. Our batch size was 8, and Epoch was 50. In fold 2, we have taken all courses and displayed individual result results also. For all classes, labels were 2731, accuracy was 0.834, Recall was 0.76, MaP. 5 was 0.809. For all bikes, labels were 392, accuracy was 0.803, Recall was 0.605, MaP. 5 was 0.68. For entire bus, labels were 735, accuracy was 0.852, Recall was 0.82, MaP. 5 was 0.881. For every car, labels were 934, accuracy was 0.837, Recall was 0.784, MaP. 5 was 0.833 For all cng, labels were 489, accuracy was 0.855, Recall was 0.81, MaP. 5 was 0.857. For all vehicle, labels were 181, accuracy was 0.82, Recall was 779, MaP. 5 was 0.794.

Table 4: RESULT OF CROSS VALIDATION FOLD 4

Fold	Class	Images	Labels	Precision	Recall	MaP.5
4	all	326	2942	0.853	0.755	0.815
	bike		452	0.852	0.586	0.671
	bus		788	0.848	0.809	0.875
	car		1040	0.842	0.769	0.832
	cng		487	0.895	0.828	0.873
	truck		175	0.825	0.783	0.825

Table 4, shows the outcome of fold 4. We have tested 20 percent and trained our model 80 percent . We have a total did 5-fold to test. Our batch size was 8, and Epoch was 50. In fold 4, we have taken all classes and given individual outcomes. For all class, labels were 2942, accuracy was 0.853, Recall was 0.755, MaP. 5 was 0.815. For all bikes, labels were 452, accuracy was 0.852, Recall was 0.586, MaP. 5 was 0.671. For all bus, labels were 788, accuracy was 0.848, Recall was 0.809, MaP. 5 was 0.875. For every car, labels were 1040, accuracy was 0.842, Recall was 0.769, MaP. 5 was 0.832. For all cng, labels were 487, accuracy was 0.895, Recall was 0.828, MaP. 5 was 0.873. For all truck, labels were 175, accuracy was 0.825, Recall was 0.783, MaP. 5 was 0.825.

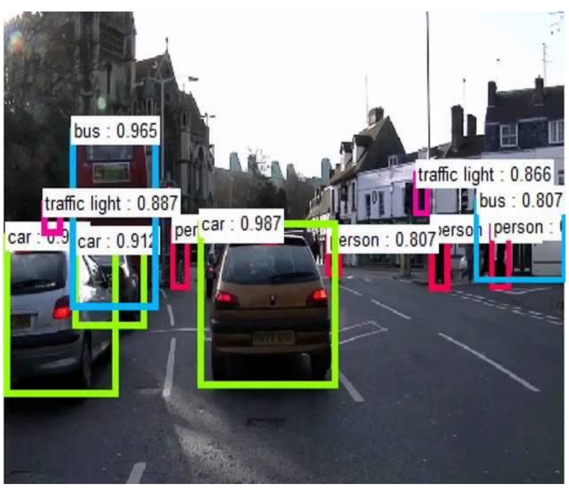
Table 5: RESULT OF CROSS VALIDATION FOLD 5

Fold	Class	Images	Labels	Precision	Recall	MaP.5
5	all	326	2925	0.843	0.741	0.803
	bike		460	0.84	0.606	0.691
	bus		775	0.829	0.809	0.863
	car		986	0.858	0.726	0.815
	cng		500	0.875	0.822	0.865
	truck		204	0.812	0.74	0.782

Table 5, shows the outcome of fold 5. We have tested 20 percent and trained our model 80 percent . We have totally done the 5-fold test. Our batch size was 8, and Epoch was 50. In fold 5, we have taken all classes and given individual outcomes. For entire class, labels were 2925, accuracy was 0.843, Recall was 0741, MaP. 5 was 0.803. For all bike, labels were 460, accuracy was 0.803, Recall was 0.741, MaP. 5 was 0.803. For entire bus, labels were 775, accuracy was 0.829, Recall was 809, MaP. 5 was 0.863. For every car, labels were 986, accuracy was 0.858, Recall was 0.726, MaP. 5 was 0.815 For all cng, labels were 500, accuracy was 0.875, Recall was 0.822, MaP. 5 was 0.865. For all truck, labels were 204, accuracy was 0.812, Recall was 0.74, MaP. 5 was 0.782.

4.5 Comparative models of object detection

Model Name	**mAP**	**Real Time**
Fast YOLO	53.5%	1
YOLO	64.7%	1
Fast R-CNN	66.4%	0
Faster R-CSS VGG-16	70.0%	0
Faster R-CNN ZF	73.2%	0



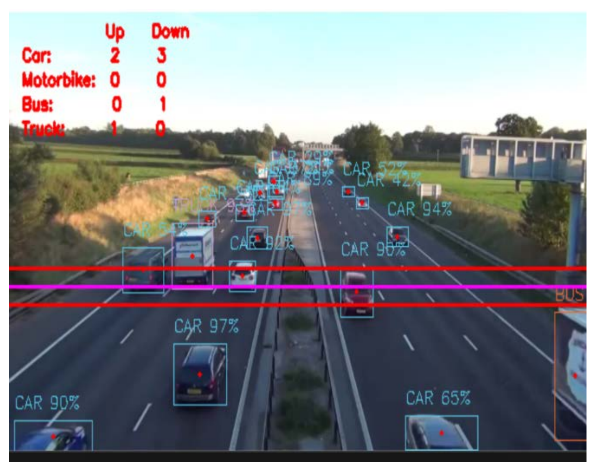


Fig. 11 Vehicle detection with Fast-CNN and Yolo

4.6 Comparison to some previously completed works:

There were numerous valuable papers about this system. However, not all of them had great success. We know that this system uses various techniques, including trajectory-based, GPS, FASTER R CNN, Yolo-V3, and R CNN. Here, we can observe several previously completed recognitions and their level of accuracy.

Table 6: COMPARISON WITH OTHER RELATED WORKS.

Author Name	Main Aim	Used Model	Best Model	Accuracy
(Gandhi et al., 2020)	The use of AI to calculate traffic congestion using CCTV cameras in traffic intersections	Yolo	Yolo	82%
(Osman et al., 2017)	They aimed to create two methods. One of the approaches was built with hardware assistance, while the other was developed without it.	Theory-based	Theory-based	none
(Alpatov et al., 2018b)	Their aim was detecting the vehicle,	CNN	CNN	76%

	counting them, and road marking detection.			
(Nam et al., 2020b)	Using Information Obtained from Connected and Autonomous Probes, Estimating Traffic Density	LSTM	LSTM	65%
(Manchanda et al., 2019)	Detect traffic extreme density and accident rate in India	Theory-based	Theory-based	none
(Pramanik et al., 2021)	This study suggests a novel method for inexpensively gathering data of traffic. To demonstrate the effectiveness of the collected historical data in predicting	SVR HA ARIMA	ARIMA	63%

	future congestion.			
(Mrazek et al., 2020)	Traffic control by measuring density	Theory-based	Theory-based	none
(Raj et al., 2016)	Application of data mining techniques for estimating and predicting traffic density	TMS KNN ANN	ANN	84%
(Nemade, 2016)	Intelligent traffic with video tracking	YOLO	YOLO	69%
(Liu et al., 2017)	A real-time, adaptive car counting system for urban roads based on video	YOLO2	YOLO2	74%
(Meng et al., 2021)	This work proposed adaptive traffic light control system that uses image processing and image matching technique successfully	Fast CNN	Fast CNN	72%

	controlling the traffic by capturing photographs of each lane at a junction			
(Ariffin et al., 2021)	Demonstrates a sophisticated system that dynamically adjusts each lane's cycle time according to the number of vehicles present at an intersection.	IOT Device	IOT Device	none
(Asmara et al., 2020)	Aims to measure how the error rate obtained from the results of forecasting	CNN YOLO	YOLO	81%
(Mahmood et al., 2020)	Aims to use a computational process called YOLO to detect vehicles with infrared photographs in road traffic.	YOLO2	YOLO2	74%

(Corovic et al., 2018)	Aims the demonstration the usage of the newest YOLOv3 algorithm for the detection of traffic participants	Yolo3	Yolo3	78%

4.7 Comparison with static system and proposed system:

Table 7: Static system result

No	**Lane1**	**Lane2**	**Total Lane**
1	60	56	116
2	105	22	127
3	42	134	176
4	58	112	170
6	63	43	106
7	72	34	106

Table 8: Simulation system result

No	**Lane1**	**Lane2**	**Total Lane**
1	89	112	201
2	55	68	123
3	72	96	168
4	43	102	145
6	58	73	131
7	96	33	129

CHAPTER 5

CONCLUSION

In this thesis, we have focused most adaptable and realistic traffic solution. In summary, by adaptively changing time of green signal light based on the traffic density, the proposed method ensures that the direction with more traffic is given a green signal for a longer time compared to the route with fewer traffic. It also helps to help socio-demographic changes by less pollution of weather and fuel use. The technology exhibits a considerable improvement over the current method regarding the quantity of vehicles crossing the road intersection, with simulation results showing an improvement of roughly 25%. This system can be enhanced to accomplished even finer with additional calibration utilizing real-world data of camera for training of our proposed model. Our proposed system predicts 86%. We have used YOLO3, and for better understanding, we did a simulation, code, and graph. Our study has revealed the overall perceptions about traffic management systems. This work could be beneficial for properly controlling traffic and help decision-makers in the traffic management to get a proper idea about the current traffic system and find a better solution.

CHAPTER 6

FUTURE WORKS

In the future, we will try to explore more in the traffic sector. We can integrate Google maps to analyze more accurately. In our dataset, we have our data and some from google. But in the future, we can create our dataset bigger, we also try some low-resolution images, day and night images, and blur images to perform our model more accurately. This work may be included in a sophisticated traffic management system. In this scenario, our model will function as an Artificial Intelligence (AI) system to control traffic lights intelligently. With the aid of our suggested approach, it is possible to employ artificial intelligence (AI) monitoring to detect accidents as they are happening. A sophisticated traffic management system may include this work. In this scenario, our model will function as an Artificial Intelligence (AI) system to control traffic lights intelligently. With the aid of our suggested approach, it is possible to employ artificial intelligence (AI) monitoring to detect accidents as they are happening.

REFERENCES

[1] Alpatov, B. A., Babayan, P. v., & Ershov, M. D. (2018b). Vehicle detection and counting system for real-time traffic surveillance. 2018 7th Mediterranean Conference on Embedded Computing (MECO), 1–4. https://doi.org/10.1109/MECO.2018.8406017

[2] Ariffin, W. N. S. F. W., Keat, C. S., Suriyan, T. P. A., Nore, N. A. M., MohamadIrfanSyaaramiMohdLazim, HasneezaLizaZakaria, Hashim, N. B. M., & Zain, A. S. M. (2021). Real-time Dynamic Traffic Light ControlSystem with Emergency Vehicle Priority. Journal of Physics: Conference Series, 1878(1), 012063. https://doi.org/10.1088/1742-6596/1878/1/012063

[3] Asmara, R. A., Syahputro, B., Supriyanto, D., & Handayani, A. N. (2020). Prediction of Traffic Density Using YOLO Object Detection and Implemented in Raspberry Pi 3b + and Intel NCS 2. 2020 4th International Conference on Vocational Education and Training (ICOVET), 391–395. https://doi.org/10.1109/ICOVET50258.2020.9230145

[4] Corovic, A., Ilic, V., Duric, S., Marijan, M., & Pavkovic, B. (2018). The Real-Time Detection of Traffic Participants Using YOLO Algorithm. 2018 26th Telecommunications Forum (TELFOR), 1–4. https://doi.org/10.1109/TELFOR.2018.8611986

[5] Gandhi, M. M., Solanki, D. S., Daptardar, R. S., & Baloorkar, N. S. (2020). Smart Control of Traffic Light Using Artificial Intelligence. 2020 5th IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE), 1–6. https://doi.org/10.1109/ICRAIE51050.2020.9358334

[6] Liu, F., Zeng, Z., & Jiang, R. (2017). A video-based real-time adaptive vehicle-counting system for urban roads. PLOS ONE, 12(11), e0186098. https://doi.org/10.1371/journal.pone.0186098

[7] Mahmood, M. T., Ahmed, S. R. A., & Ahmed, M. R. A. (2020). Detection of vehicle with Infrared images in Road Traffic using YOLO computational mechanism. IOP

Conference Series: Materials Science and Engineering, 928(2), 022027. https://doi.org/10.1088/1757-899X/928/2/022027

[8] Meng, B. C. C., Damanhuri, N. S., & Othman, N. A. (2021). Smart traffic light control system using image processing. IOP Conference Series: Materials Science and Engineering, 012021. https://doi.org/10.1088/1757-899X/1088/1/012021

[9] Mrazek, J., Mrazkova, L., Hromada, M., & Vavra, J. (2020). Traffic Control Through Traffic Density (pp. 253–258). https://doi.org/10.2507/daaam.scibook.2020.22

[10] Nam, D., Lavanya, R., Jayakrishnan, R., Yang, I., & Jeon, W. H. (2020a). A Deep Learning Approach for Estimating Traffic Density Using Data Obtained from Connected and Autonomous Probes. Sensors, 20(17), 4824. https://doi.org/10.3390/s20174824

[11] Nemade, B. (2016). Automatic Traffic Surveillance Using Video Tracking. Procedia Computer Science, 79, 402–409. https://doi.org/10.1016/j.procs.2016.03.052

[12] Osman, T., Psyche, S. S., Shafi Ferdous, J. M., & Zaman, H. U. (2017). Intelligent traffic management system for cross section of roads using computer vision. 2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC), 1–7. https://doi.org/10.1109/CCWC.2017.7868350

[13] Pramanik, Md. A., Rahman, M. M., Anam, A. S. M. I., Ali, A. A., Amin, M. A., & Rahman, A. K. M. M. (2021). Modeling Traffic Congestion in Developing Countries Using Google Maps Data (pp. 513–531). https://doi.org/10.1007/978-3-030-73100-7_36

[14] Raj, J., Bahuleyan, H., & Vanajakshi, L. D. (2016). Application of Data Mining Techniques for Traffic Density Estimation and Prediction. Transportation Research Procedia, 17, 321–330. https://doi.org/10.1016/j.trpro.2016.11.102

183-16-380

ORIGINALITY REPORT

13%	9%	8%	%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	dspace.daffodilvarsity.edu.bd:8080 Internet Source	5%
2	Mihir M. Gandhi, Devansh S. Solanki, Rutwij S. Daptardar, Nirmala Shinde Baloorkar. "Smart Control of Traffic Light Using Artificial Intelligence", 2020 5th IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE), 2020 Publication	4%
3	Md. Imtiaz Hossain Subree, Md. Rakib Hasan, Maksuda Haider Sayma. "Design and Implementation of an Unreal Engine 4-Based Smart Traffic Control System for Smart City Applications", International Journal of Advanced Computer Science and Applications, 2022 Publication	1%
4	www.jesr.ub.ro Internet Source	1%
5	www.semanticscholar.org Internet Source	<1%

6	Divya Meena, Hari Krishna P, Chakka Naga Venkata Jahnavi, Patri Lalithya Manasa, J Sheela. "Efficient Wildlife Intrusion Detection System using Hybrid Algorithm", 2022 4th International Conference on Inventive Research in Computing Applications (ICIRCA), 2022 Publication	<1 %
7	Md. Selim Hossain, M.M. Kamruzzaman, Shuvo Sen, Mir Mohammad Azad, Mohammad Sarwar Hossain Mollah. "Hexahedron core with sensor based photonic crystal fiber: An approach of design and performance analysis", Sensing and Bio-Sensing Research, 2021 Publication	<1 %
8	gupea.ub.gu.se Internet Source	<1 %
9	ui.adsabs.harvard.edu Internet Source	<1 %
10	ir.unimas.my Internet Source	<1 %
11	core.ac.uk Internet Source	<1 %
12	Bhushan Nemade. "Automatic Traffic Surveillance Using Video Tracking", Procedia Computer Science, 2016	<1 %

Publication

13 Rosa Andrie Asmara, Bimo Syahputro, Dodit Supriyanto, Anik Nur Handayani. "Prediction of Traffic Density Using YOLO Object Detection and Implemented in Raspberry Pi 3b + and Intel NCS 2", 2020 4th International Conference on Vocational Education and Training (ICOVET), 2020 <1%
Publication

14 orca.cardiff.ac.uk <1%
Internet Source

15 Asma Ait Ouallane, Assia Bakali, Ayoub Bahnasse, Said Broumi, Mohamed Talea. "Fusion of Engineering Insights and Emerging Trends: Intelligent Urban Traffic Management System", Information Fusion, 2022 <1%
Publication

16 www.biologicalengineering.in <1%
Internet Source

17 www.diva-portal.org <1%
Internet Source

18 "Innovative Data Communication Technologies and Application", Springer Science and Business Media LLC, 2022 <1%
Publication

19 "Intelligent Technologies and Applications", Springer Science and Business Media LLC, <1%

2019
Publication

20	etd.aau.edu.et Internet Source	<1 %
21	"Artificial Intelligence for Communications and Networks", Springer Science and Business Media LLC, 2019 Publication	<1 %
22	Traisirin Saengseangfa, Suradet Tantrairatn. "Design and development of a vision system for identifying a workpiece quality and defects using object detection", 2021 25th International Computer Science and Engineering Conference (ICSEC), 2021 Publication	<1 %

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	On		