ELECTRONIC ASSIGNMENT COVERSHEET

Murdoch UNIVERSITY

Student Number:	34445873
Surname:	Chew
Given name:	Jian Yue
Email:	cjianyue@gmail.com

Unit Code:	ICT619
Unit name:	Artificial Intelligence
Enrolment mode:	External
Date:	31 March 2023
Assignment number:	2
Assignment name:	Project
Tutor:	Dr. Loo Poh Kok

Student's Declaration:

1. Except where indicated, the work I am submitting in this assignment is my own work and has not been submitted for assessment in another unit.
2. This submission complies with Murdoch University's academic integrity commitments. I am aware that information about plagiarism and associated penalties can be found at

http://our.murdoch.edu.au/Educational-technologies/Academic-integrity/. If I have any doubts or queries about this, I am further aware that I can contact my Unit Coordinator prior to submitting the assignment.

3. I acknowledge that the assessor of this assignment may, for the purpose of assessing this assignment:
   a. reproduce this assignment and provide a copy to another academic staff member; and/or
   b. submit a copy of this assignment to a plagiarism-checking service. This web-based service may retain a copy of this work for the sole purpose of subsequent plagiarism checking, but has a legal agreement with the University that it will not share or reproduce it in any form.
4. I have retained a copy of this assignment.
5. I will retain a copy of the notification of receipt of this assignment. If you have not received a receipt within three days, please check with your Unit Coordinator.

I am aware that I am making this declaration by submitting this document electronically and by using my Murdoch ID and password it is deemed equivalent to executing this declaration with my written signature.

Optional Comments to Tutor:

e.g. If this is a group assignment, list group members here

Group assignment

1. Chew Jian Yue (34445873@student.murdoch.edu.au) 34445873
2. Bao Qikao (34386276@student.murdoch.edu.au) 34386276

*If you can, please insert this completed form into the body of **each** assignment you submit. Follow the instructions in the Unit Information and Learning Guide about how to submit your file(s) and how to name them, so the Unit Coordinator knows whose work it is.*

Start your assignment on the next page.

AI vehicle detection and classification with fuzzy system for controlling traffic light signal

Abstract

As cars become more and more affordable, the upsurge of vehicles in modern cities has a direct impact on traffic congestion. Traffic congestion is still a hard nut to crack for many traffic authorities, and there is still no “perfect” solution. This paper presents a model of real-time vehicle detection and traffic signal allocation with AI to mitigate traffic conditions at intersections. The use of YOLOv8 was proposed for vehicle detection and classification. The data is then fed to the fuzzy logic system to determine the adequate green light signal time to allow vehicles to pass the intersection efficiently. The results indicate that YOLOv8 has a higher vehicle detection and classification accuracy than with previous models used in previous studies, and fuzzy logic allocate signal time to manage the traffic condition at interactions better. The model is highly recommended for smart cities like Singapore and other metropolitans without incurring high costs on specialised road sensors.

Table of Contents

Introduction

Traffic congestion is a severe problem in modern cities around the world. For the traveller, traffic congestion results in longer travel times and commuters' frustration (Wen, 2008). Time wasted during travel deteriorates the travellers' quality of life and may sometimes incur more financial costs. This is especially true in times of changing situations, for example, in the recent post-pandemic world, where there is a surge in commuter numbers in Singapore daily due to more people coming back to work in the office. Traffic congestion would cause delivery delays and lost worker productivity (Wen, 2008). In metropolitan cities such as Singapore, there are possibly thousands of vehicles on the roads, and that could be more prone to fatal accidents (Joy et al., 2018).

Furthermore, traffic congestion has an impact on air pollution and fuel consumption, which affects health and the environment. A recent study has determined that Singapore is one of the best places for motorists (Tan, 2023). However, this comes at the cost to vehicle owners due to the measures implemented by the Singapore government to curb traffic congestion and the number of vehicles on the roads. These costs include higher High Additional Registration Fees for vehicles, the requirement of obtaining a "*Certificate of Entitlement*" to purchase cars (Meng et al., 2015), and increasing ERP charges. These costs may be unnecessary or could be reduced if there is a better traffic management solution or the ability to manage traffic. Detection of the number of vehicles and collecting of this data could be helpful understand and plan future traffic policies, and better managing traffic.

This paper proposed a model with real-time vehicle detection and transfer into traffic light signal management at intersections to smooth traffic flows in AI technology. It has been organised as follows: upcoming section describes the background of AI applications in traffic control from previous works. After that, an in-depth investigation of possible AI techniques and the justification for the chosen ones on our model is presented. Then, it elaborates on how the models are evaluated. Lastly, the results are presented and discussed.

Background

With the advancement of technology, many countries have taken advantage of that and alleviated the impact brought by the increasing number of vehicles on the roads. The popularity and successful implementation of AI in recent years also contributed to the acceleration of AI adoption in traffic control.

In Singapore, there are a lot of CCTV cameras installed at traffic junctions. There are “Junction Electronic Eyes” (J-Eyes) system of some 400 cameras (mounted on traffic light posts, lampposts, etc.) located at major traffic junctions in Singapore that monitor traffic conditions and verifies real-time incidents. Currently, one of the intelligent transport systems implemented by the Land Transport Authority (LTA) is the Green Link Determining System. With that, adjacent traffic signals are linked to allow vehicles to travel from one junction to another with minimal stops. However, most of the traffic lights at intersections are still managed by conventional static timers learned from historical data. The adoption of AI to manage traffic aligns with the government's smart nation plan in transport.

The studies on the feasibility of AI implementation in traffic control have given a fundamental understanding of them. Albatish and Abu-Naser (2019) suggested using a rule-based system to model and control traffic lights by detecting the number of vehicles using sensors that are handled by a microcontroller called Arduino. However, the accuracy of the sensor detection was not mentioned. In addition, they ignore the fact that such design requires special infrastructure. It is costly and requires periodic maintenance (Zaid et al., 2017).

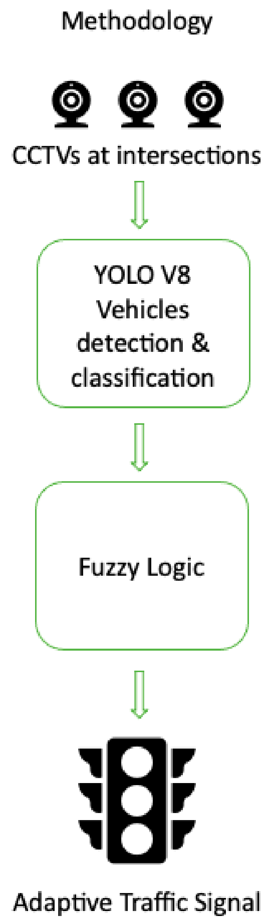
Vogel et al. (2018) developed a traffic control system by means of fuzzy logic. This showed that there is a major improvement over the examined traffic parameters on the primary driveway on the compromise of worsening the traffic on the secondary driveway. That is not able to solve the overall traffic condition.

Lakshmi and Kalpana (2017) proposed using image processing to compute the traffic density and used Arduino as a microcontroller to determine the traffic signal in a sequential manner. The drawback of it is that it did not take the types of vehicles on the roads into account. There is no priority given to emergency vehicles. The number of vehicles is computed in pixels from the video frame corresponding to the area occupied by vehicles. As a result, when a few trucks occupy a big area and irregular spaces exist between the vehicles, the accuracy of vehicle computing will be reduced. The inaccurate data will lead to incorrect traffic signal prediction.

In the early 1990s, Foy et al. (1992) discovered that genetic algorithms (GA) applied to traffic controls converge to a reasonable timing strategy. They presented this in the response of GA in finding a longer cycle time at intersections when arrival volume increases to the point of oversaturation.

Gandhi et al. (2020) found that live images from CCTV at traffic intersections can provide an excellent benefit for computing traffic density, and these data are useful for traffic light calculations. It showed a considerable improvement of 23% in crossing the intersections regardless of the number of cars in each lane.

Given the comprehensive coverage of CCTV at main traffic intersections, this project planned to leverage CCTVs to detect, count and identify the type of vehicles at the traffic intersections. We would take the commonly used vehicles in Singapore, such as cars, trucks, buses, and lorries for computation. These vehicles are either transporting humans, goods, delivery packages or other dangerous items. Our AI project aims to detect whether the incoming vehicles are either of the above types of vehicles. This data will be reprocessed to adjust the traffic green light duration in response to the road conditions to alleviate the traffic problems.



AI Techniques and Analysis

In this section, we would explore extensive research that had been carried out to investigate the possible AI techniques to use in our architecture. On vehicle detection, we considered Faster R-CNN, Single Shot Detector (SSD) and YOLOv8 and compared them. To determine the traffic light signal duration, we analysed fuzzy logic, genetic algorithms, and ANN.

Vehicle Detection and Classification

To find out what is the best for vehicle detection, we reviewed a comparative analysis of image detection algorithms studied by Srivastava et al. (2021) and the newly released YOLOv8.

1. Faster R-CNN

R-CNN stands for region-based Conventional Neural Networks. It is a mixture of regional proposals for objection segmentation and high-capacity CNNs for objection detection (Srivastava et al., 2021). Each image is extracted in 2000 region proposals and uses CNN to compute its distinct features to classify the output. Hence, it is a very time-consuming process. Faster R-CNN was developed to solve this drawback. There is a significant reduction in time for training and testing. However, its performance is still greatly affected by the region proposals (Srivastava et al., 2021). A study have concluded that faster R-CNN on vehicle detection with a simple application performs unimpressively (Q. Fan et al., 2016).

2. SSD

SSD approach is built on a feed-forward convolutional network that scores a collection of fixed-size bounding boxes after its production and leads to the final detection results in a way of non-maximum suppression (Srivastava et al., 2021). The early network layers are built on a standard architecture as the base network which is meant for high-quality image classification. Then, the detection will be produced by the auxiliary structure added to the truncated base network (Srivastava et al., 2021). It has higher accuracy than faster R-CNN, but it is still not perfect in detecting small objects. In our case, motorbikes may be not detected.

3. YOLOv8

YOLO is an object detection system that can handle real-time processing. As it is run based on a convolutional neural network (CNN) which predicts multiple bounding boxes and classes simultaneously. Moreover, it is highly accurate. It was reported that YOLO achieves 72% top-1 accuracy and 91.2% top-4 on ImageNet respectively (Hui, 2022). With the release of YOLO8, it is perceived to be higher accuracy and flexibility (Bhalerao, 2023).

Generally, YOLO system resizes input images to 448x448 and runs a single convolutional network on the image and thresholds the resulting detections by the model’s confidence.

Advantages and disadvantages

Below is the summary of comparison for the most three common object detection algorithms (Srivastava et al., 2021).

No.	Method	Advantage	Disadvantage
1	Faster R-CNN	Faster R-CNN reduces the time complexity. Its base network can self-learn the regional proposals	Faster R-CNN has a slow execution time. It is expensive to increase the overall feasibility of the process due to the large computational resources. It is quite hard to make improvements when training the model and delay in the proposition of different objects (Lenc & Vedaldi, 2015)
2	Single Shot Detector (SSD)	SSD runs at real-time speed with lost cost. There is accuracy improvement compared to the state-of-the-art Faster R-CNN	SSD decreases the resolution of images to lower quality. It underperforms on small-scale objects and is less efficient.
3	YOLO V8	The model is pre-trained. Although it processes and computes at high real-time speed, its computation cost is relatively low. With the reduction of background	YOLO V8 sometimes fails to detect smaller objects in an image or video due to the lower recall rate. It has limitations on bounding boxes, fail to detect

		errors, it achieves higher accuracy (J. Fan et al., 2021).	two objects when they are extremely close to each other.

YOLOv8 was the best fit and suitable for our model.

Reasons for choosing YOLOv8 for image detection and classification in our model:

1. YOLO has a better speed of object detection as it predicts objects in real time. This aligns with our strategic plan to have first-hand information on traffic conditions and turn it into an adequate traffic signal.
2. YOLO has excellent learning capabilities which enable it to learn the key features of an object and apply them to object detection. It plays a critical role in classify the type of vehicles waiting at the intersections
3. YOLO sees the entire image during test and training time and encodes contextual information about classes together with the appearance. Hence, it makes half the number of background errors compared to another top detection method - fast R-CNN (unable to see the larger context).
4. YOLO processes streaming video in real-time with low latency at a low cost.

Traffic Signal Calculation

After the vehicle detection, classification and calculating the number of each vehicles, there are several other methods for calculating the traffic light timings, the data may be fed into an algorithm to estimate the traffic light timings (to turn green, red or orange).

1. Genetic algorithm

Adopted the concept of genetic evolution in biological studies, Genetic algorithms provide the best solutions for the problem through the processes of selection, reproduction, and mutation. The fittest solution will last till the end of the process (Teo et al., 2010).

Solutions to the problems are akin to chromosomes in a population. There is a fitness function to select the fit solution or gene. Every solution will be filtered by the fitness function and measured how fit they are. They will be then ranked accordingly and select fitter ones. The selected solution would undergo another stage of process and reproduction. This mutation only stops when the stopping criteria are reached (Teo et al., 2010).

Genetic algorithm provides an alternative to traditional methods to solve problems with fewer constraints. However, it is not recommended in our model due to the slowness and cost of implementing.

2. Fuzzy logic

Fuzzy was developed in 1965 and it is the best solution to deal with real-time problems (Sawake & Borkar, 2017). A fuzzy system is built upon membership functions (MFs), comprising of several fuzzy sets. Generally, there are three codification shapes to choose from for MFs inputs. They are triangular, trapezoidal and Gauss respectively and trapezoidal is commonly used (Akhtar & Moridpour, 2021).

A review by Sawake and Borkar (2017) found that the fuzzy logic approach applied to traffic signal control can easily solve the traffic congestion issue and uneven traffic flow not only on single intersections but also multiple intersections.

3. Artificial Neural Network (ANN)

An ANN is the imitation of a biological neuron. It could consist of many inputs and one output. There are many simple processing elements that are intertwined with each other and at different layers (Mishra & Srivastava, 2014).

There have been some problems with ANN described (Mishra & Srivastava, 2014):

- No structured methodology available in ANN
- No single standardised paradigm for the development of ANN
- Output quality of ANN may be unpredictable

- ANN systems do not describe how they solve problems
- High computational burden

In this case, ANN will process the image capture to determine what class of the vehicles belong to for further analysis. As a result, the output of this method will heavily depend on tuning positive and negative training data using the sliding window method (Zaid et al., 2017) and the prediction was only on whether it was a car or not. A study by Soman and Radhakrishnan (2018) also found that there is an error rate of 2% with execution of 1.5 seconds.

In our case, fuzzy logic is selected to determine a suitable period for the red and green light at traffic intersections due to its high reliability. When the number of vehicles increases, the period of green light will increase.

Reasons for choosing fuzzy logic for traffic light time calculation in our model:

1. It is a robust system where no precise inputs are required. In this case, the system will give acceptable reasoning even if there is a small error in the data.
2. The system structure itself is simple and justifiable. It is easy for users to configure and modify the rules.
3. It works more prone to human reasoning and decision-making. The outputs are more justifiable to the situation.
4. It accepts different types of inputs, including qualitative or quantitative, which gives more flexibility to the model in the event of any issue with the data formats.

Advantages and disadvantages

No.	Method	Advantage	Disadvantage
1	Genetic algorithm	Genetic algorithms provide better solutions than traditional methods, able to solve	Genetic algorithms are too slow, expensive to implement and difficult to understand, debug and optimize (Teo et al., 2010)

		problems with multiple objectives and constraints.	
2	Fuzzy logic	Fuzzy logic can manage words and sentences in natural language for transparent model formulation. It is designed to handle not only quantitative but also qualitative data. It works more prone to human reasoning and decision-making, which makes it easy to build straightforward and flexible control structure (Hoogendoorn et al., 1999)	Fuzzy logic is highly dependent on human knowledge and expertise. It requires a regular update of the rules in the system. Currently, it still lacks complete methodology or theory for stability analysis (Hoogendoorn et al., 1999).
3	ANN	ANN is good at learning and modelling non-linear and complex relationships. It does parallel processing and deals with insufficient data and information with high fault tolerance.	ANNs are computationally expensive, likely overfits and have a fixed number of input layers.

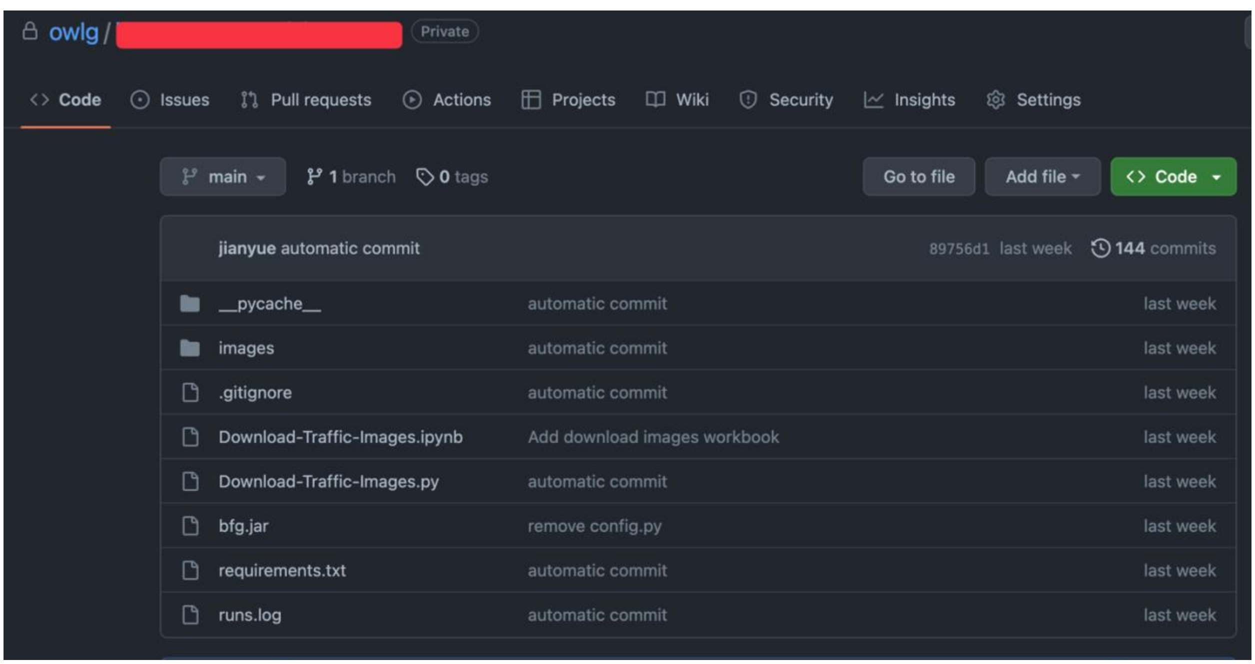
Generally, after the completion of vehicles detection by YOLO, fuzzy logic will be applied on the data output of the YOLO algorithm to estimate the traffic green light duration to allow the vehicles to travel across the traffic junction.

After consideration of the AI techniques found in our research, our chosen AI technique is YOLO V8 for object detection on CCTV traffic image data and fuzzy system for determination of traffic green light signal duration. This is for the following reasons:

Data and knowledge acquisition

Acquisition of data

There are traffic camera images of main expressways across entire Singapore that are available from Land Transport Authority website, which is updated at every five minutes intervals, found at https://onemotoring.lta.gov.sg/content/onemotoring/home/driving/traffic_information/traffic-cameras.html. A program was written to download the images periodically from all the traffic cameras or for a specific traffic camera over a specified time. A video demonstration is found here: https://www.youtube.com/watch?v=HVymH2Uv_I0. Additionally, a computer is set-up to execute the script every few minutes to download images from all traffic cameras available in Singapore for a few days. The computer is also configured to automatically upload the images periodically to a private GitHub repository.



Overall, there were 144 commits made. For our paper, it would be too tedious to analyse all traffic camera images from all angles across too many days. We have decided to narrow our focus to just a specific traffic camera with camera ID of 3795 and on a particular day 23 Mar 2023. A program is written to focus on a particular date and a particular camera ID, and these are configurable on the *config.py* file.

The dataset of traffic CCTV images was collected over a day for a particular traffic light camera from the above website. In particular, we collected the images dataset for testing from 11:55 PM on 22 Mar 2023 to 11:45 PM on 23 Mar 2023. In total, we filtered out 123 images that included the night and day datasets.

Below are common issues when or after acquisition of traffic data.

Issue 1: Traffic images at traffic junctions are difficult to obtain

Although most of the road networks are installed with CCTV cameras, not all of them are of high definition. Traffic junction images are not publicly available; however, this problem be circumvented with the use of publicly available traffic images to infer the traffic conditions leading up to the traffic junctions (see below figure for illustration). The image captured for incident management may not be suitable for object detection.

An important consideration and challenge are that vehicles detected could be travelling to multiple expressway exits and hence to multiple traffic light junctions. Hence, the data we are feeding to the fuzzy system may not be accurate enough to control that particular junction's traffic green light duration. Although it could be infeasible to conduct in this short study, combining knowledge of other detections and metrics for other traffic light cameras and other forms of data could improve the accuracy of the system.

Issue 2: Other forms of data are unavailable

A possible challenge is the inaccuracy in identifying the traffic conditions with just the reliance of traffic camera images that are only updated at intervals of five minutes at the expressways. Video streams are superior, as we are able to more accurately detect the number of moving vehicles over a period of time. There could be an instance at expressway (fast moving vehicles) that when the camera captures, the vehicles are little but in true fact, they have already driven away from the frame of the camera. There could be instances whereby important vehicles are not detected such as ambulance.

However, in situations where there are slow moving traffic or traffic jams, the model should perform well.

Issue 3: Incompatible data

There could be issues with the incompatible image formats being used for analysis. Traffic cameras may output data to incompatible formats. This applies even for image data, formats may vary, conversion of data formats from one image format to other image formats may result in loss of data. Some image photos may not be compatible for use in prediction. We have standardised the use of using jpeg as this image format is widely adopted and compatible in most instances.

Issue 4: Missing data

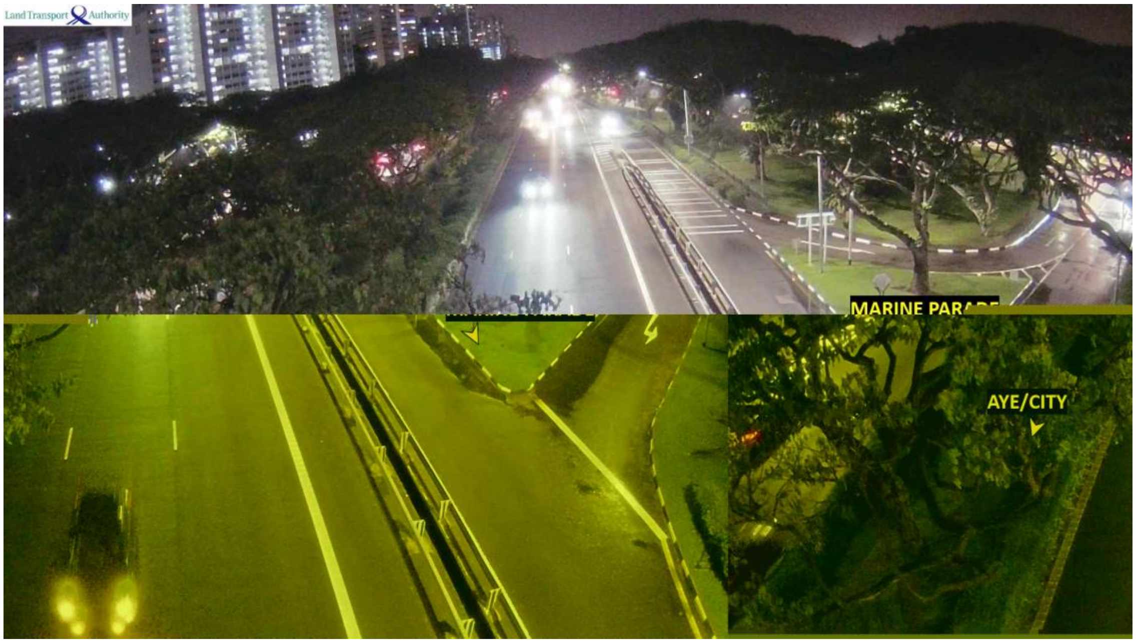
When traffic cameras are under maintenance, there could be a period of downtime, whereby the density of the traffic cannot be predicted. There are some occurrences of these, in future, our project may account for this problem by detecting such scenarios and the traffic light junction can be programmed to use historical data as an estimation to the signal duration.

Issue 5: Data corruption

Since traffic image data are being downloaded from a network, there could be issues with data corruption. Image data are smaller than video feed used in some projects, hence this problem should be considered as minor. Furthermore, the data should be downloaded and processed at some server instead of at the traffic junction, as there are no issues with retrieving the traffic images from anywhere in the world with Internet.

Issue 6: Poor quality of data

There are some traffic camera images that are of poorer quality than others or could have been subjected to unexpected changes not previously specified. For example, there is little to no indication of images that are of the following quality for the traffic camera view from Marine Parade Flyover taken at 03/03/2023 23:21 PM:



The above image differs largely from usual images collected and this could be unexpected. These images can only be resolved by the Land Authority of Singapore, and performing such corrections ourselves are usually to costly and ineffective. From our observations, such occurrences are rare.

The images during the night are at times blurry and there are instances bright light of vehicle headlights leads to inability to detect and classify moving vehicles. There is a possibility that the model wrongly detects (such as detecting cars as lampposts) or does not detect the vehicles.

However, the quality of images during bright day light are superior, we expect that the model is able to detect vehicles quite effectively during the afternoons and early evenings. Examples of images collected from daily CCTV traffic images:

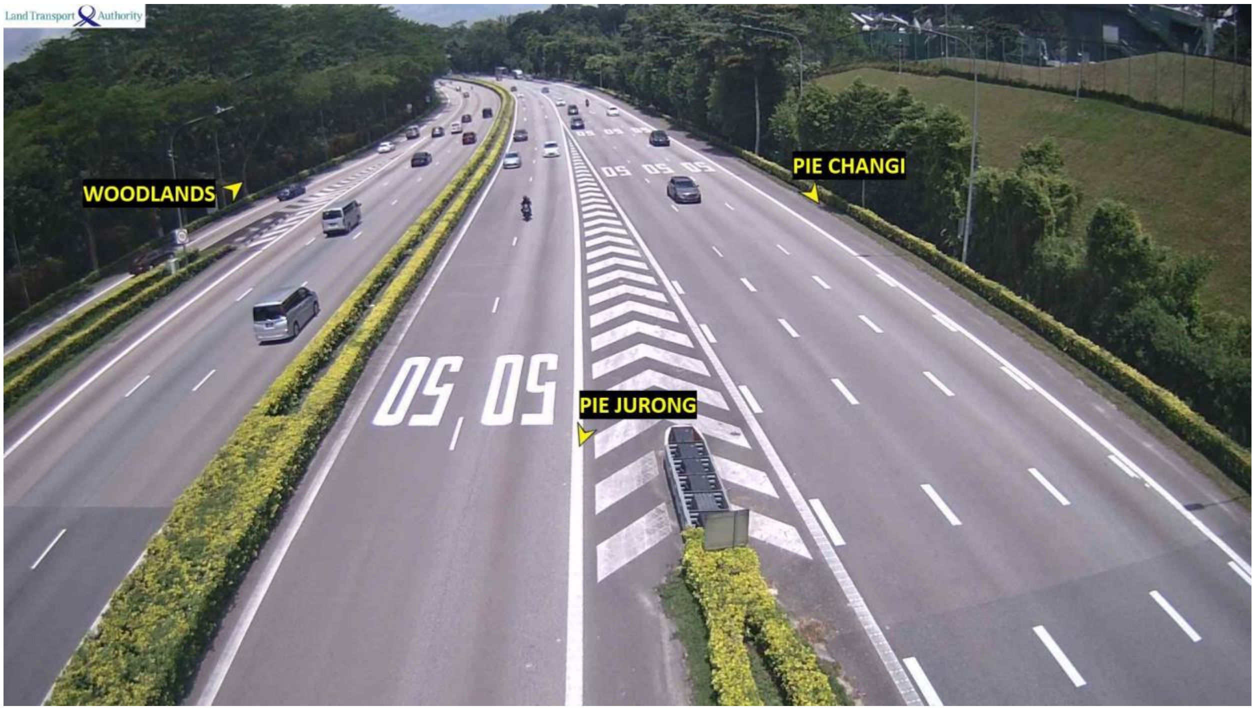


Figure 1 - Image collected at 19 Feb 2023 at 12.40pm (Camera ID 2703) - _2023-02-19_12-40_2703_1238_20230219124108_529752.jpg

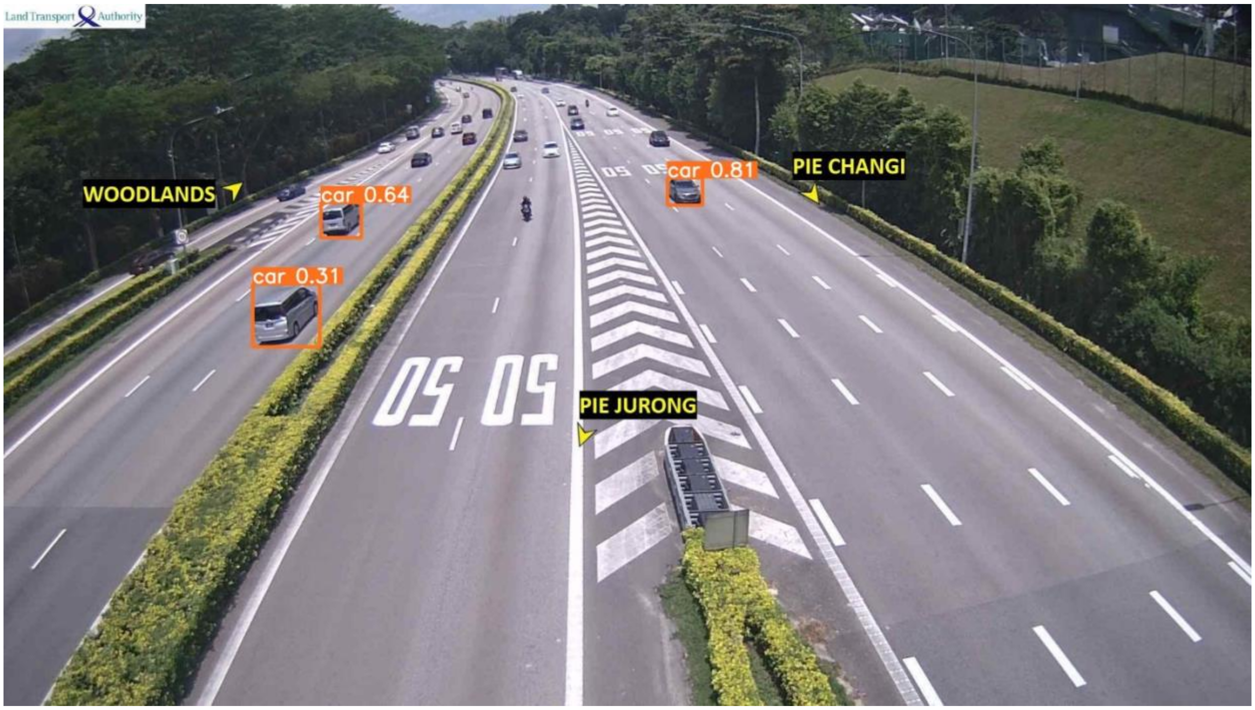


Figure 2 - Predicted on available pre-trained model

The above figure shows a preliminary test conducted on the image, small vehicles and images that are too far from the camera could hardly be detected with the pre-trained model being applied to the image directly. Hence, we expect this may result in under-counting the number of vehicles in the traffic.

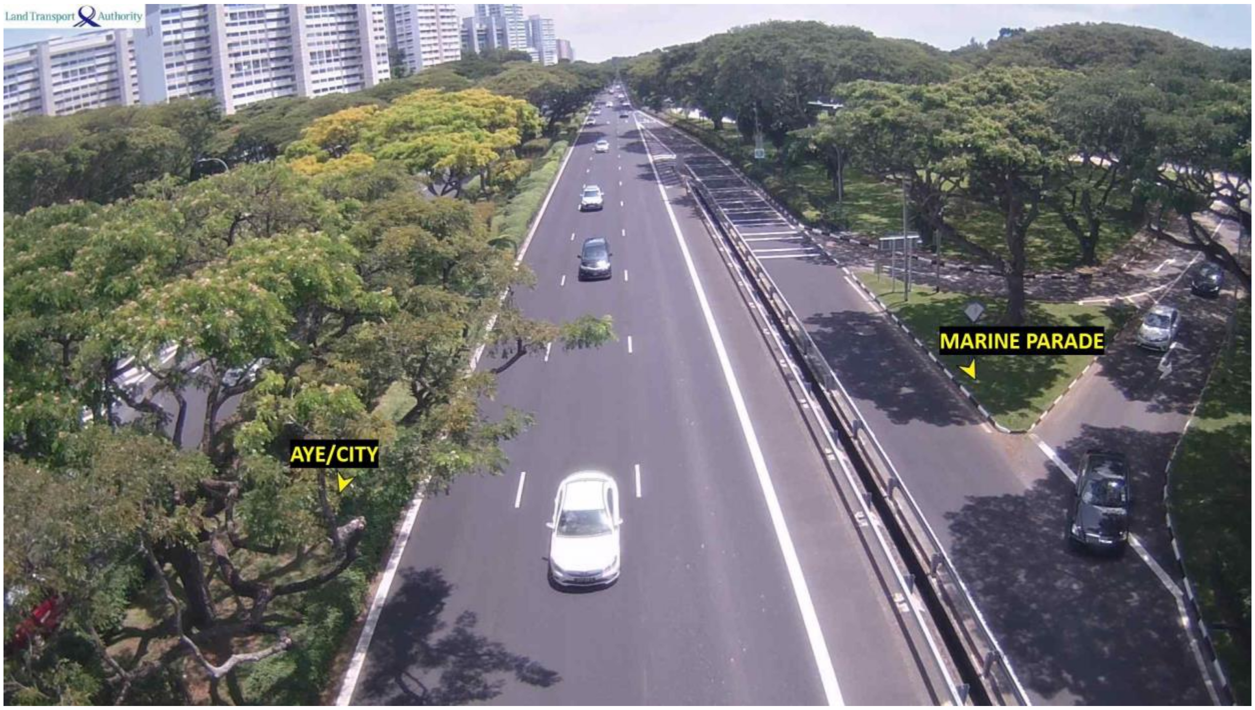


Figure 3 - Image collected at 19 Feb 2023 at 12.40pm (Camera ID 3795) - _2023-02-19_12-40_3795_1239_20230219124021_504899.jpg

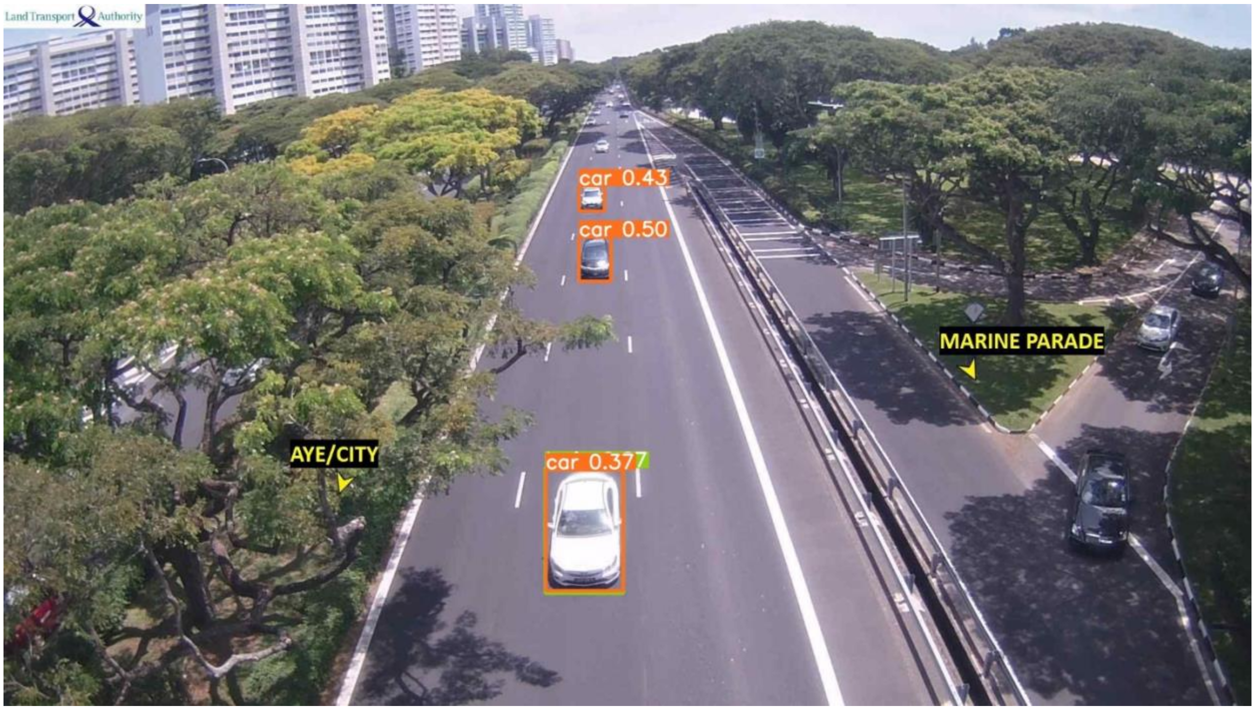


Figure 4 - _2023-02-19_12-40_3795_1239_20230219124021_504899.jpg (Predicted on available pre-trained model)

As seen in **Error! Reference source not found.**, some vehicles on darker regions or shadows cannot be detected as cars using the pretrained model directly applied to the earlier image. Furthermore, the car (0.37) is being detected as something else at the same time.

In this project, we aim to select one of the CCTV camera images collected over a period of time. The collection of CCTV images can be configured at an interval of around 10 minutes. The above images are samples of the CCTV traffic images at certain regions in Singapore collected on 19 Feb 2023 at 12:39 PM. The location of the CCTV is approximately at the location shown on Google Maps below, which is at the Marine Parade Flyover (how this is determined is demonstrated here: https://www.youtube.com/watch?v=z3ipUt8BgOE):

Issue 7: Differences in data across different countries and regions

Even within Singapore, there are differences in the image data being collected, such as different designs of buses, cars and other vehicles. The variations in how the traffic system works vary even more across different countries although the variation within a small Singapore like Singapore can be assumed to be minimal.

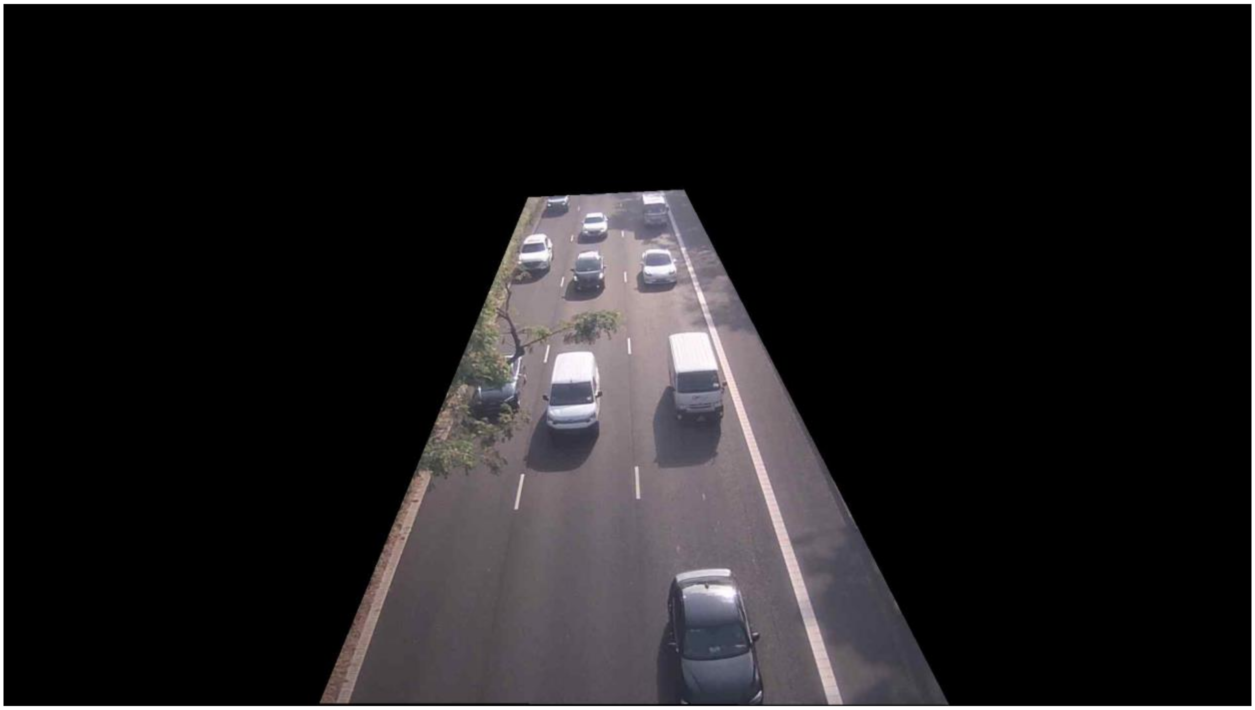
There may be some traffic regulations that we may not be aware of, since we are not an expert in road traffic, including all the traffic regulations in many countries will be too lengthy in our project or report.

Data preparation

From the consideration of the above problems, we found that the most important issue is to allow the model to only detect a certain direction of vehicle movement, image data was cropped to focus on just the road that we are interested in detecting the number of vehicles. This may increase the accuracy of detection and counting vehicles on the roads. Through image processing and object detection, the data is used as input for real-time traffic density calculation.

In this case, we are interested in knowing the volume-importance metric of vehicles travelling to AYE/CITY. We can only include the portion of the images that we can tell with some certainty the class of vehicles (more information in user guide).

After using our pre-processing script to crop the image, we can see the below and after pictures as follows:



Building up the model

Image detection using YOLO algorithm

Image detection is being applied to traffic camera images either at traffic junctions or at readily available traffic images that can be from Singapore's Land Transport Authority website. The number of vehicles and the type of vehicles are predicted after the image is fed into the model.

Table 1 - Table of types of vehicles found in Singapore and their importance

Vehicle type	Number of passengers	Importance Weight
Car	1	1
Bus	35 (varies)	35
Truck	10	10
Ambulance	3	500
Motorbike	1	1

A demonstration is performed here: https://www.youtube.com/watch?v=leNAv8qYlqo on how the script works and the computation of the volume-importance metric is being shown in the video.

The output from the prototype Python script (which is attached to this report) will be the volume-importance metric (variable volume_*imp*) which will be the input of the fuzzy system below as "*vehicleCount*".

```
In [81]: volume_imp = car_w * car_no + bus_w * bus_no + truck_w * truck_no

         volume_imp

Out[81]: 35
```

The importance value (termed as volume-importance vehicle metric) can be tuned or modified by a subject-matter expert or through further research with machine learning (e.g., using linear regression, etc.). The importance value can be adjusted by the expert on a separate configuration file. This volume-importance of vehicles metric is important, and this value will be sent to the fuzzy logic system as an input.

The number of people waiting at the traffic junctions can be also fed to the fuzzy system as an input, and the traffic from intersecting roads. These inputs (or more, in our project) will determine the duration of the traffic light showing red or green, as an improvement to the traditional traffic control system.

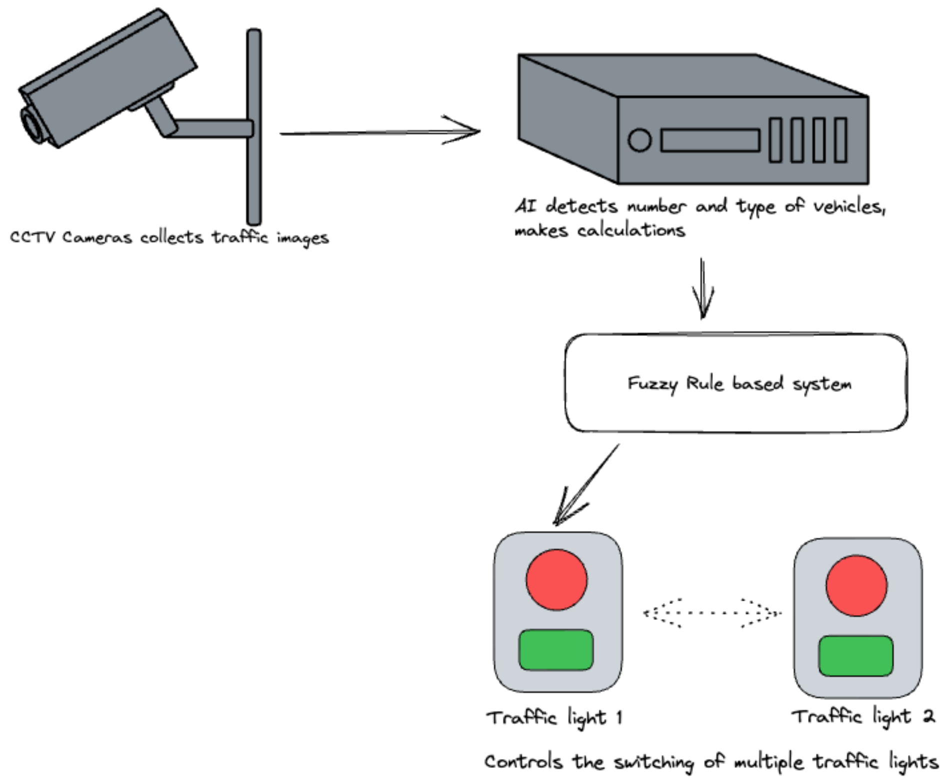


Figure 5 - General idea of the prototype

Fuzzy logic system to determine the traffic green light duration

A demo of the traffic green light duration obtained from the fuzzy rule-based system to control the traffic light from the point-of-view of the vehicles at the traffic junction can be seen in the video I have created here: https://www.youtube.com/watch?v=3xeNdoAVezc

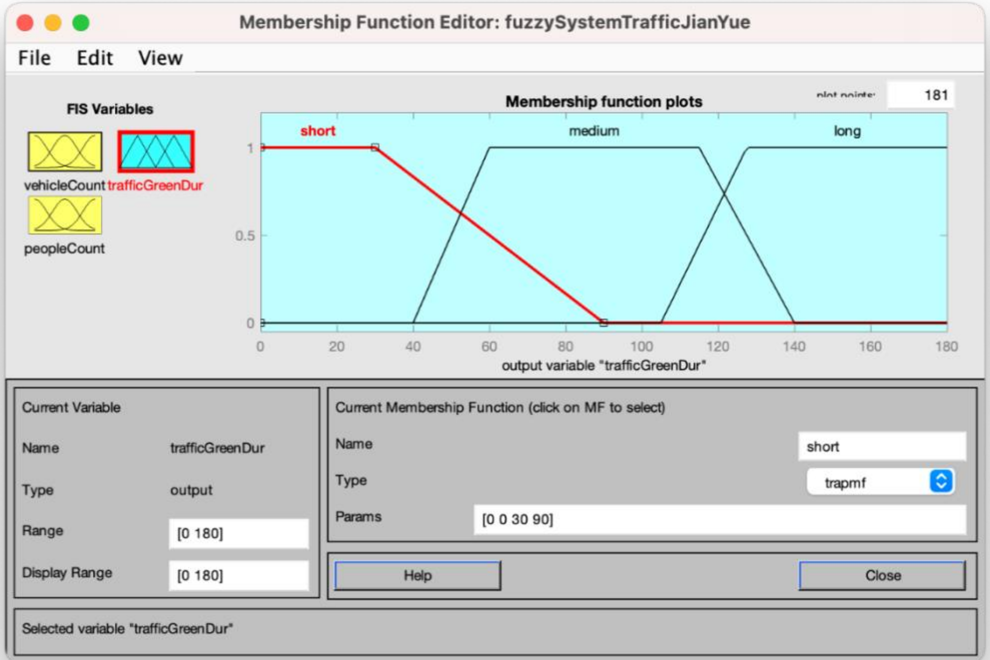


Figure 6 - Example of membership function indicating the linguistic variable, names used are "short", "medium" and "long".

Figure 7 - Fuzzy rule viewer example

For example, for relatively low volume-importance of vehicles (vehicleCount), and few number of people (peopleCount), the traffic green light duration is determined to be 72.5 seconds.

The output can be obtained from the MATLAB interface and the output (in number of seconds) may be fed back to the traffic light controller system.

```
>> greenLightTime = evalfis(fuzzySystemTrafficJianYue, [200,70])

greenLightTime =

    65.4804
```

Generally, there is only a need to control the duration of the green light on a traffic light, the time it takes from green to amber and to red is generally standardised based on the roads and country, and there is rarely a need to adjust. The intersecting traffic light will be the “reverse” of the traffic light that we are controlling with our fuzzy system duration. Although the traffic conditions on the intersecting junction may influence the traffic light duration, we have chosen to simplify this instead of also integrating this into our fuzzy system. This could be explored in our future projects.

Evaluation Method

Our proposal involves vehicle detection and classification, the calculation of vehicle volume-importance metric, and the fuzzy system that computes the amount of time for the traffic signal at intersections.

Vehicle Detection

During the training phase, a machine has been set-up with the appropriate environment for labellers’ easy access on labelling the image data. A step-by-step guide can be found on the User Guide. In total we have around 34 hand labelled images. To evaluate the trained model, for vehicle detection, we used eight images (some in the labelled images) for validation and testing (data splitting).

The data from vehicle detection determine how much time the signal is allocated to it. Hence, the accuracy of the data is critical in this model (Zaid et al., 2017) compared manual and automatic calculations of cars based on the ANN algorithm. It showed an average error rate of 2% on the six test samples, which could affect the traffic light period of 1.5 seconds. We find this method is not stringent enough. For our model, we apply a confusion matrix to calculate the accuracy of vehicle detection. Other evaluations and results are found in appendix.

Fuzzy system - Average deviation score

$$AvgDeviation = \frac{\sum |FuzzyScore - HumanOpinionScore|}{TotalDataPoints}$$

HumanOpinionScore is the level of congestion estimated by the people we may survey. In this paper, we did not survey people as it is too time-consuming. The *HumanOpinionScore* could be the average maximum time that each of the passengers are willing to wait, if we were to survey people in the vehicles. The fuzzy score is the traffic green light duration that our fuzzy system generates. From there, we may tweak the fuzzy system values, the universe of discourse, whether the need for the changes in membership functions.

Results

Vehicle detection

There were 8 images retrieved from LTA road camera to simulate the vehicle detection and classification by YOLOv8.

The eight images are subjected to at least two vehicle detection models to test the accuracy of detection and the results are present below.

<u>YOLOv8 pre-trained model</u>

		Predicted Values		
		Car	Truck	Bus
Actual Values	Car	34	0	0
	Truck	0	4	2

Bus	0	0	0

Accuracy: (34+4) / 40= 0.95

Error rate = 1- 0.95 = 0.05

Trained YOLOv8 model

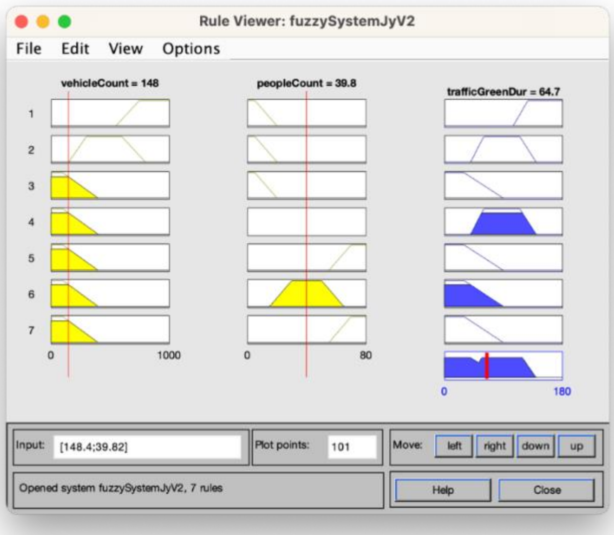
Predicted Values

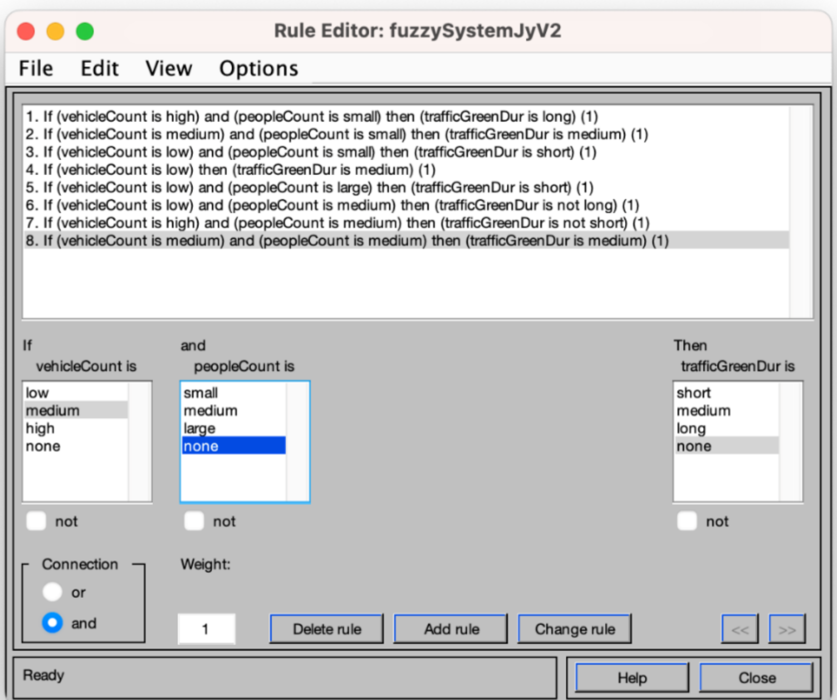
		Car	Truck
Actual Values	Car	34	1
	Truck	2	3

Accuracy: (34+3) / 40= 0.925

Error rate = 1- 0.925 = 0.075

Fuzzy system





The rules are re-evaluated and refined, according to logical reasoning. Some of the values including the range, rules and the membership parameters were modified.

Command line output on MATLAB:

```
>> greenLightTime = evalfis(fuzzySystemJyV2, [100,80])

greenLightTime =

   64.4595
```

```
>> greenLightTime = evalfis(fuzzySystemJyV2, [400,10])

greenLightTime =

   95.2040
```

```
>> greenLightTime = evalfis(fuzzySystemJyV2, [100,20])

greenLightTime =

   63.5059
```

Discussion

Interpretation of results

YOLO V8 was able to detect the number of vehicles in all images in both models. In terms of classification, Model 1 achieved higher accuracy of 0.95, while Model 2 achieved with a result of 0.925. This result has proved YOLOV8 is highly reliable for object detection, and it is ideal to use in traffic control systems. We used the higher accuracy model and fed the data into fuzzy logic.

Fuzzy logic shows the flexibility of the traffic light duration adjustment. There are peoplecount and vehiclecount membership functions. When few cars with fewer people count, the traffic green light will be shorter, and vice versa. This meets our goal for a smoother traffic condition at the intersections.

Our learnings from experiment

As AI is becoming pervasive in all lines of work and people's daily life, the research on AI adoption in traffic management is exponential. The traffic data are obtained either by the sensor or camera detection. The inconsistent data source makes it complicated to evaluate the probe data (Akhtar & Moridpour, 2021). In addition, the data captured in a few days will not truly present the actual dynamic traffic situation on the road. After doing model training, our results may be highly accurate, but more factors should be taken into account in real life to solve the problem.

The surrounding is another factor contributing to traffic congestion. National events and public holidays, and school holidays play a big role in traffic congestion. For instance, the roads to Tuas and Woodlands checkpoints are crowded before public holidays and school holidays; road closure for the Indian festive Thaipusam celebration at Serangoon Road; the influx of people to Orchard Road during the Christmas shopping frenzy; and F1 racing. Those factors must be considered in traffic forecasting.

In our model, the images from the LTA cameras did not truly reflect the real traffic situation due to the angle of the camera. The trees on the roads may block the view from the camera to capture the image. It is important to make sure the CCTVs are installed at the correct angle to have a full view of the road traffic condition and there are regular checks to make sure the view from the cameras that are clear.

Nevertheless, the results from our model indicate the high accuracy of vehicle detection, which are then passed to process at fuzzy logic to allocate proper signal time duration at traffic intersections.

Conclusion

No other project that we could find has used the new metric introduced called the volume-importance metric and such a plan to detect vehicles customised to Singapore may be new. Furthermore, Singapore has been using sensors which is usually more expensive to install and maintain, without much known reliance on the availability of traffic cameras. This project could integrate with existing technologies to provide smoother and better travellers' experiences, provide an avenue for future research and studies using the current resources on traffic.

Other than the problems as described earlier, our project could further include more functionalities which are not limited to:

1. Vehicles violating traffic rules: If we engage experts in road safety and laws in our system across multiple countries, vehicles violating traffic laws could be identified by high-speed photo and video capture. This is important as to potentially reduce the road traffic accidents which causes longer travelling times. This may require involvement in image processing techniques (Gandhi et al., 2020).
2. Better methods by combining different sources of data and testing to find the optimal green light duration in different situations such as combining existing technologies already implemented and discussed above. However, this could be an issue as many of such data are not publicly available.
3. Model shall include more vehicles that are not found in the available pre-trained model, this will involve possible training on annotated and labelled images.

Acknowledgements

We would like to thank the availability of images dataset collected from Land Transport Authority Singapore (LTA) website throughout the whole day and for any particular time of the day, and how easy it was to be able to retrieve image data. We appreciate the tool available freely, Google Maps, which helps us in illustrating some ideas on this paper, find out where the cameras are located across Singapore. We would like to thank the authors and the organisation (Ultralytics) that made available the pre-trained YOLOv8 model, YOLOv5 model that are available freely online for our initial testing, comparison and results. The Python libraries for running the models onto the images, including the library that allows us to collect our images and data source, library that enables us to parse configuration files, libraries already available in Python version 3.10.

For word processing, we would like to thank Microsoft Word online for collaboration on the assignment and project. For image storage and sharing of dataset amongst us, we would like to thank Github and Microsoft OneDrive.

References

Akhtar, M., & Moridpour, S. (2021). A review of traffic congestion prediction using artificial intelligence. *Journal of Advanced Transportation*, *2021*, 1–18. https://doi.org/10.1155/2021/8878011

Albatish, I. M., & Abu-Naser, S. S. (2019). Modeling and controlling smart traffic light system using a rule based system. *2019 International Conference on Promising Electronic Technologies (ICPET)*, 55–60. https://doi.org/10.1109/ICPET.2019.00018

Bhalerao, C. (2023, January 17). YOLO v8! The real state-of-the-art? *Medium*. https://medium.com/mlearning-ai/yolo-v8-the-real-state-of-the-art-eda6c86a1b90

Fan, J., Lee, J., Jung, I., & Lee, Y. (2021). Improvement of object detection based on faster R-CNN and YOLO. *2021 36th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC)*, 1–4. https://doi.org/10.1109/ITC-CSCC52171.2021.9501480

Fan, Q., Brown, L., & Smith, J. (2016). A closer look at Faster R-CNN for vehicle detection. *2016 IEEE Intelligent Vehicles Symposium (IV)*, 124–129. https://doi.org/10.1109/IVS.2016.7535375

Foy, M. D., Benekohal, R. F., & Goldberg, D. E. (1992). Signal timing determination using genetic algorithms. *Transportation Research Record*, *1365*, 108.

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

Hoogendoorn, S., Hoogendoorn-Lanser, S., & Schuurman, H. (1999). Fuzzy perspectives in traffic engineering. *Workshop on Intelligent Traffic Management Models*.

Hui, J. (2022, September 6). Real-time Object Detection with YOLO, YOLOv2 and now YOLOv3. *Medium*. https://jonathan-hui.medium.com/real-time-object-detection-with-yolo-yolov2-28b1b93e2088

Joy, J. J., Bhat, M., Verma, N., & Jani, M. (2018). Traffic management through image processing and fuzzy logic. *2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)*, 52–55. https://doi.org/10.1109/ICCONS.2018.8662968

Lakshmi, C. J., & Kalpana, S. (2017). Intelligent traffic signaling system. *2017 International Conference on Inventive Communication and Computational Technologies (ICICCT)*, 247–251. https://doi.org/10.1109/ICICCT.2017.7975196

Lenc, K., & Vedaldi, A. (2015). *R-CNN minus R* (arXiv:1506.06981). arXiv. https://doi.org/10.48550/arXiv.1506.06981

Meng, Q., Lu, Z., & Ohtman, M. I. B. (2015). Determinants of Certificate of Entitlement premium for cars under vehicle quota system in Singapore. *Journal of the Eastern Asia Society for Transportation Studies*, *11*, 126–140. https://doi.org/10.11175/easts.11.126

Mishra, M., & Srivastava, M. (2014). A view of artificial neural network. *2014 International Conference on Advances in Engineering & Technology Research (ICAETR-2014)*, 1–3.

Sawake, V. V., & Borkar, P. (2017). Review of traffic signal timing optimization based on fuzzy logic controller. *2017 International Conference on Innovations in Information, Embedded* *and Communication Systems (ICIIECS)*, 4–4. https://doi.org/10.1109/ICIIECS.2017.8276171

Soman, R., & Radhakrishnan, K. (2018). Traffic light control and violation detection using image processing. *IOSR Journal of Engineering*, *8*(4).

Srivastava, S., Divekar, A. V., Anilkumar, C., Naik, I., Kulkarni, V., & Pattabiraman, V. (2021). Comparative analysis of deep learning image detection algorithms. *Journal of Big Data*, *8*(1), 1–27. https://doi.org/10.1186/s40537-021-00434-w

Tan, C. (2023, February 24). Singapore ties with Japan as best place for motorists: Study. *The Straits Times*. https://www.straitstimes.com/singapore/transport/singapore-ties-with-japan-as-best-place-for-motorists-study

Teo, K. T. K., Kow, W. Y., & Chin, Y. (2010). Optimization of traffic flow within an urban traffic light intersection with genetic algorithm. *2010 Second International Conference on Computational Intelligence, Modelling and Simulation*, 172–177. https://doi.org/10.1109/CIMSiM.2010.95

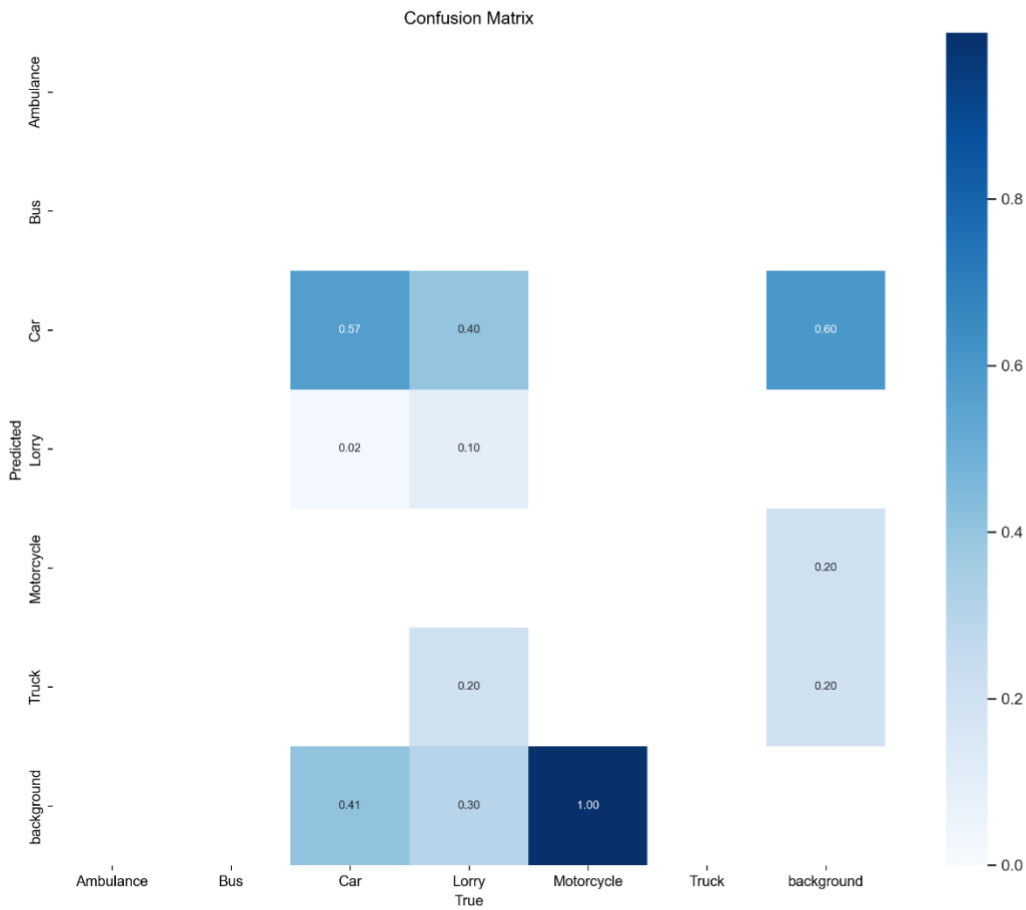
Vogel, A., Oremović, I., Šimić, R., & Ivanjko, E. (2018). Improving traffic light control by means of fuzzy logic. *2018 International Symposium ELMAR*, 51–56. https://doi.org/10.23919/ELMAR.2018.8534692

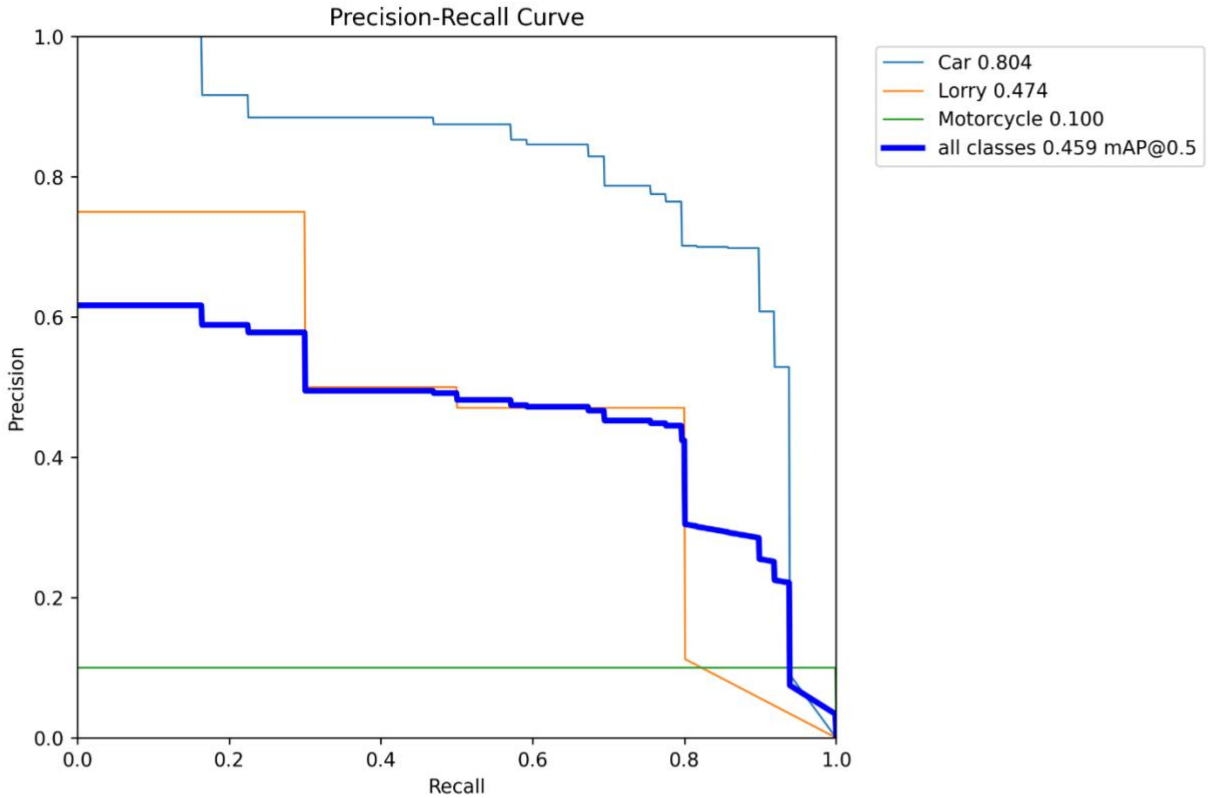
Wen, W. (2008). A dynamic and automatic traffic light control expert system for solving the road congestion problem. *Expert Systems with Applications*, *34*(4), 2370–2381. https://doi.org/10.1016/j.eswa.2007.03.007

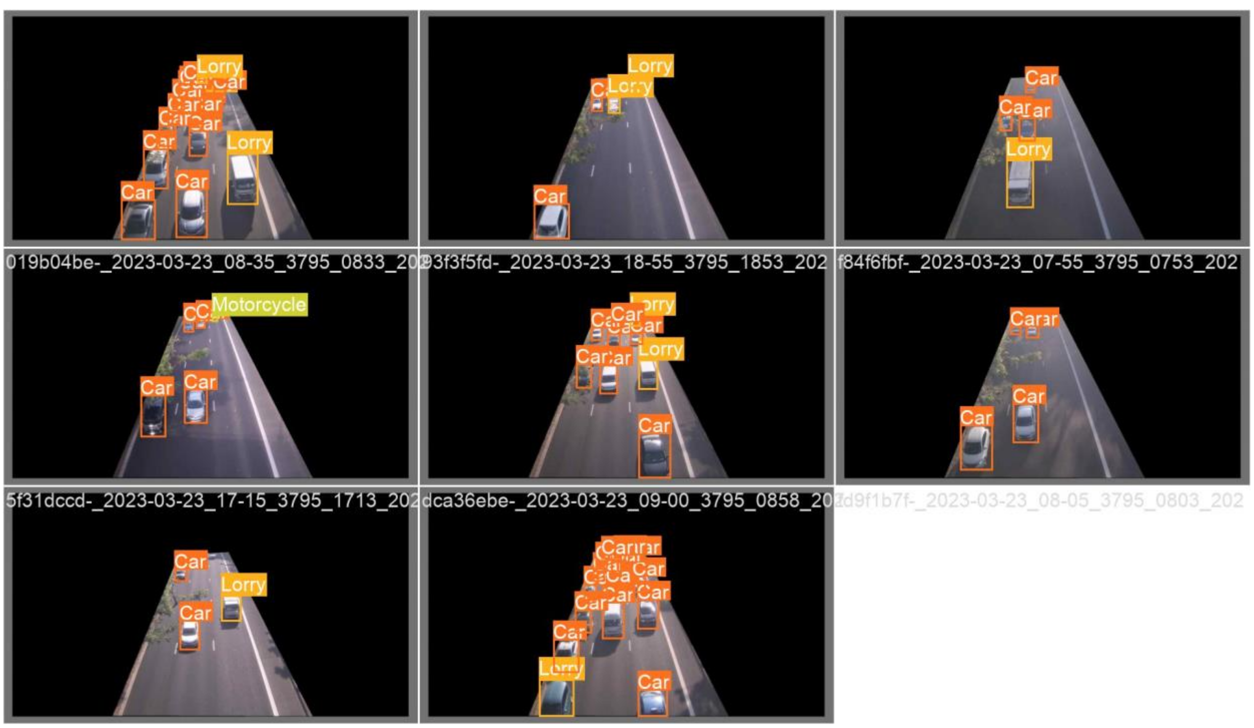
Zaid, A. A., Suhweil, Y., & Yaman, M. A. (2017). Smart controlling for traffic light time. *2017 IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT)*, 1–5. https://doi.org/10.1109/AEECT.2017.8257768

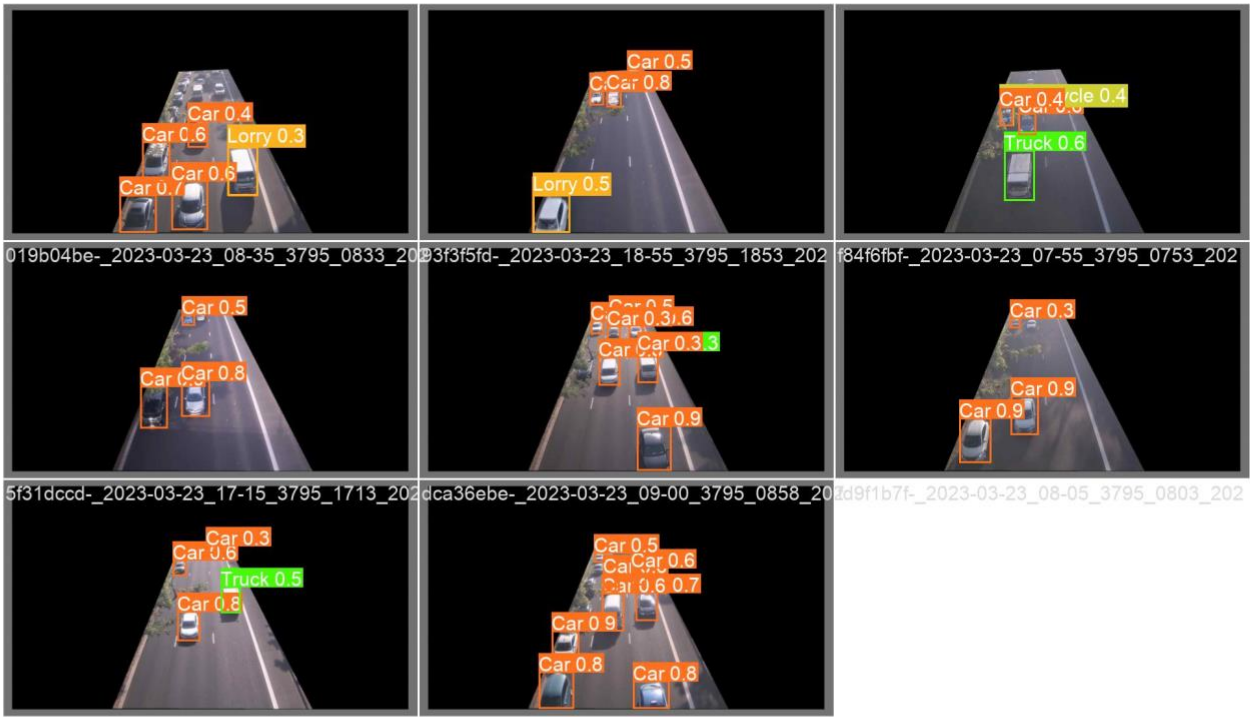
Appendix

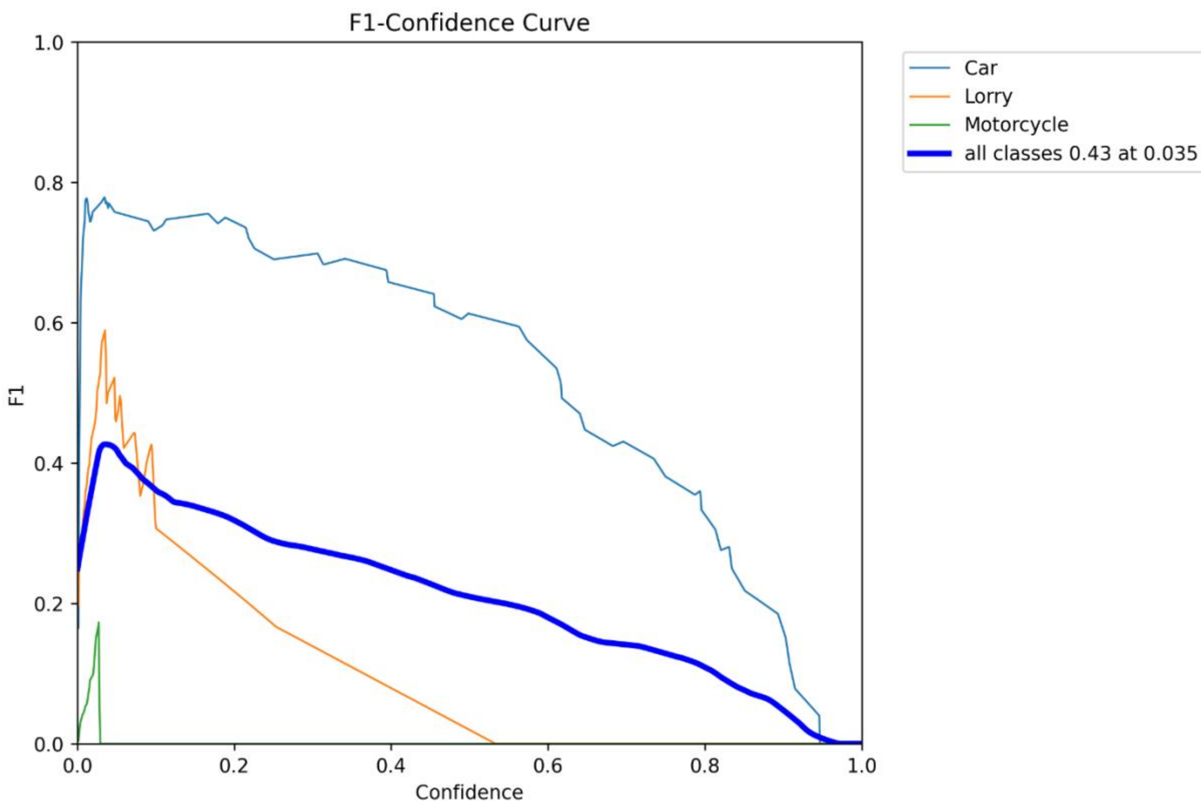
Other results for trained YOLOv8 model

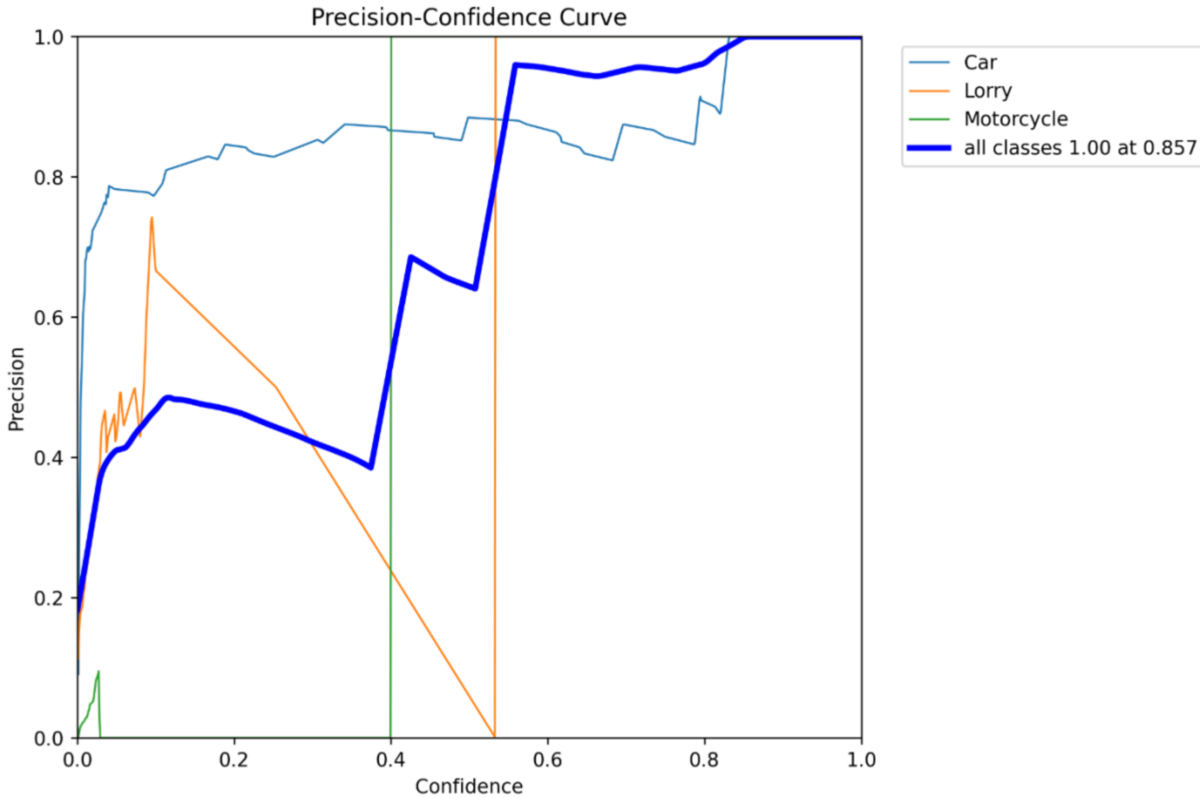


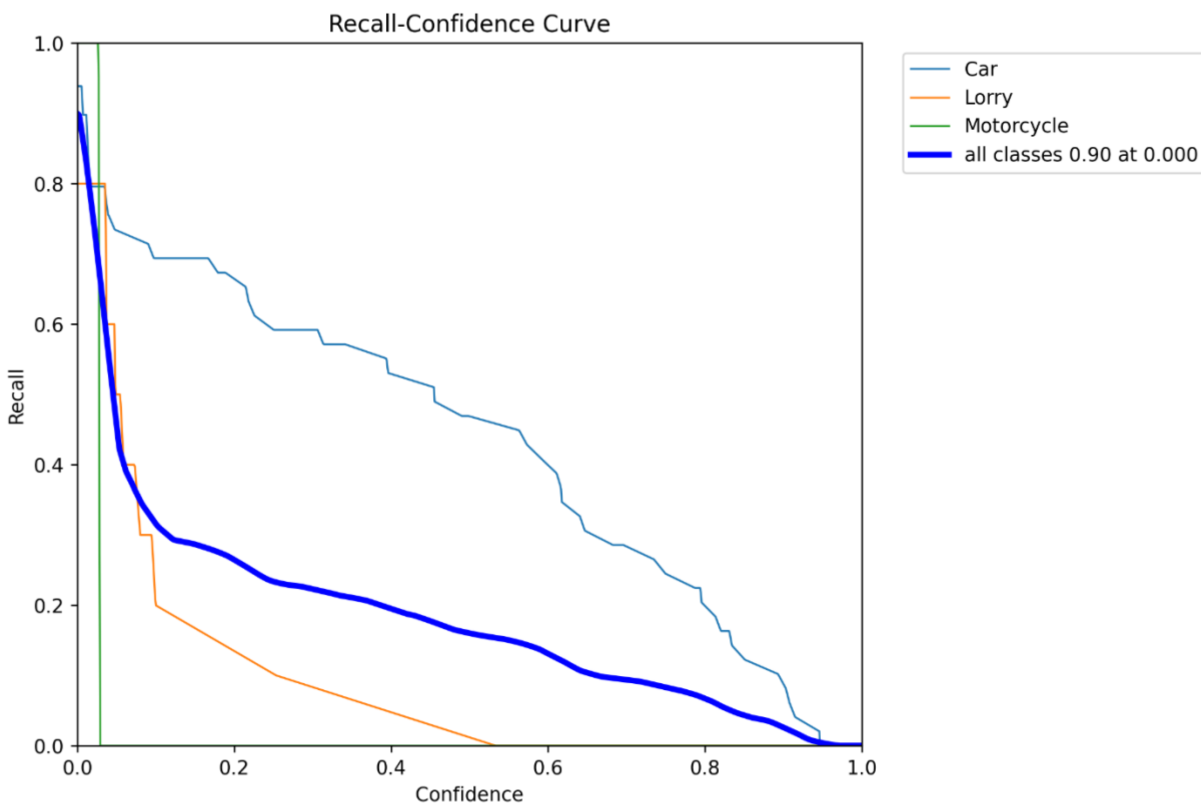

Precision-Recall Curve
Car 0.804
Lorry 0.474
Motorcycle 0.100
all classes 0.459 mAP@0.5
Precision
Recall
0.0
0.2
0.4
0.6
0.8
1.0


019b04be-_2023-03-23_08-35_3795_0833_202
93f3f5fd-_2023-03-23_18-55_3795_1853_202
f84f6fbf-_2023-03-23_07-55_3795_0753_202
5f31dccd-_2023-03-23_17-15_3795_1713_202
dca36ebe-_2023-03-23_09-00_3795_0858_202
fd9f1b7f-_2023-03-23_08-05_3795_0803_202
Car
Lorry
Motorcycle

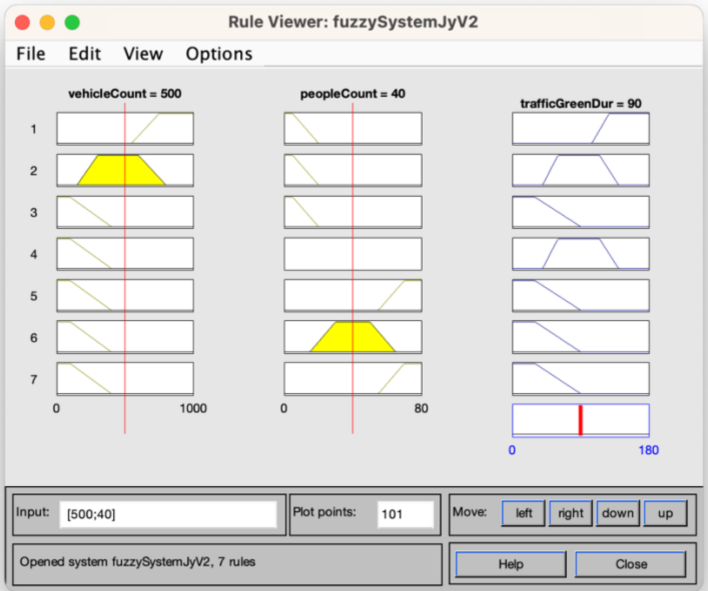

019b04be-_2023-03-23_08-35_3795_0833_202
93f3f5fd-_2023-03-23_18-55_3795_1853_202
f84f6fbf-_2023-03-23_07-55_3795_0753_202
5f31dccd-_2023-03-23_17-15_3795_1713_202
dca36ebe-_2023-03-23_09-00_3795_0858_202
d9f1b7f-_2023-03-23_08-05_3795_0803_202

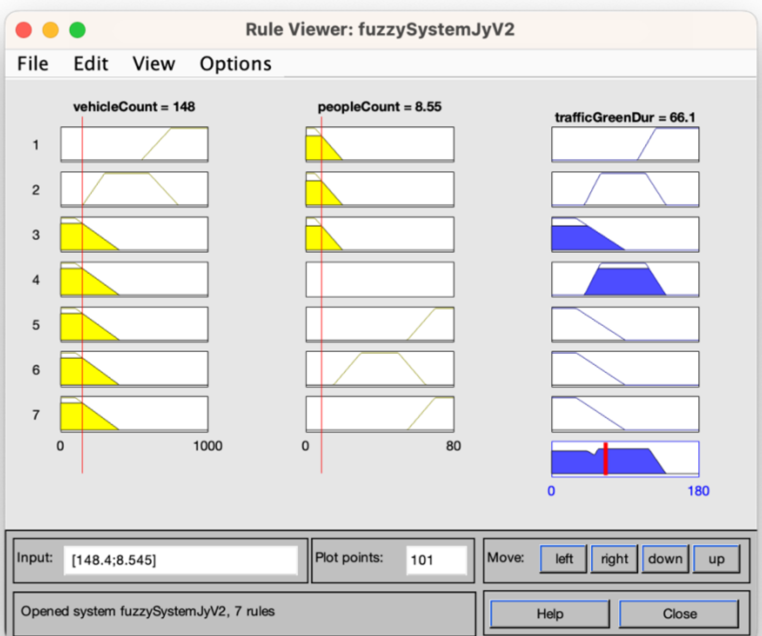

F1-Confidence Curve
Car
Lorry
Motorcycle
all classes 0.43 at 0.035
F1
Confidence
0.0
0.2
0.4
0.6
0.8
1.0

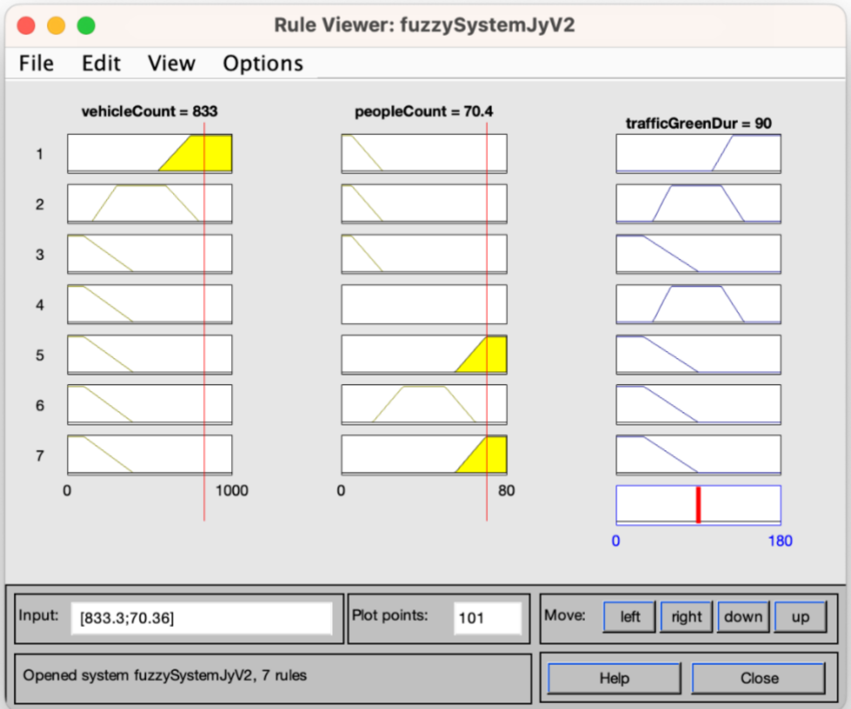

Precision-Confidence Curve
Precision
Confidence
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
Car
Lorry
Motorcycle
all classes 1.00 at 0.857

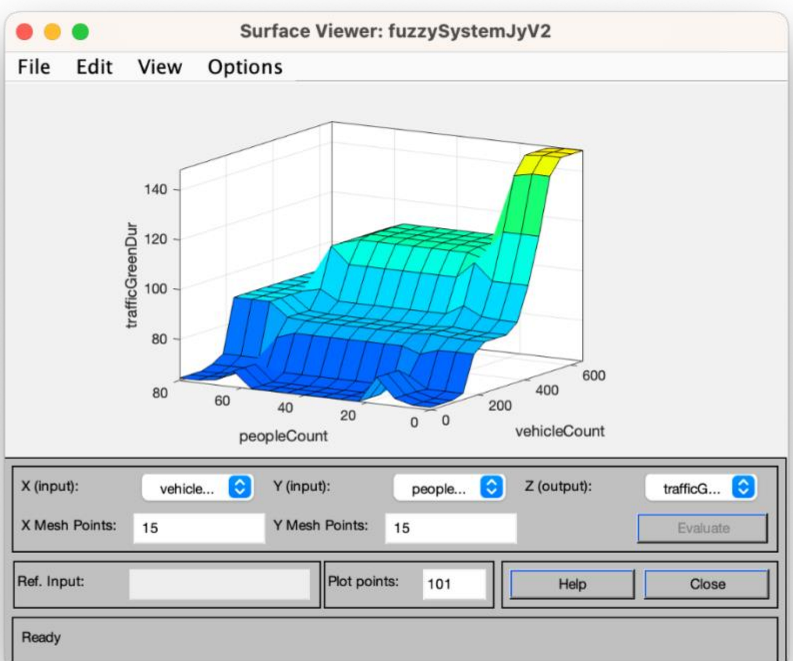

Recall-Confidence Curve
Recall
Confidence
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
Car
Lorry
Motorcycle
all classes 0.90 at 0.000

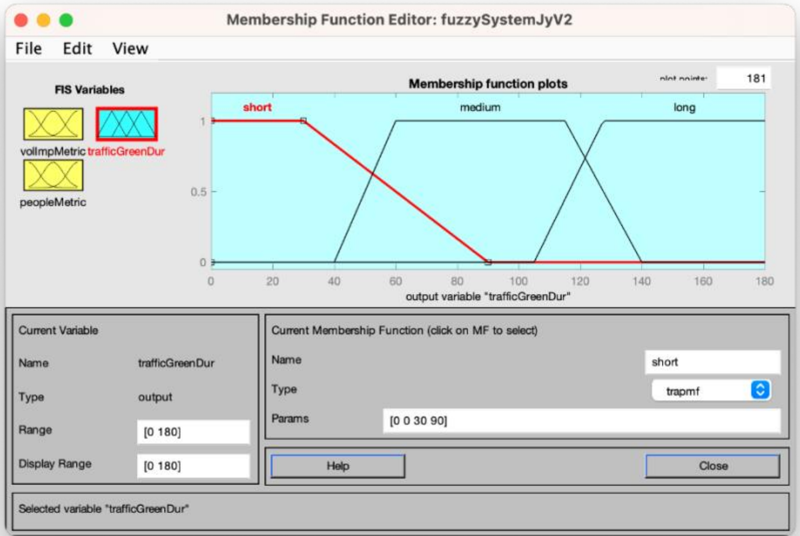
Fuzzy system

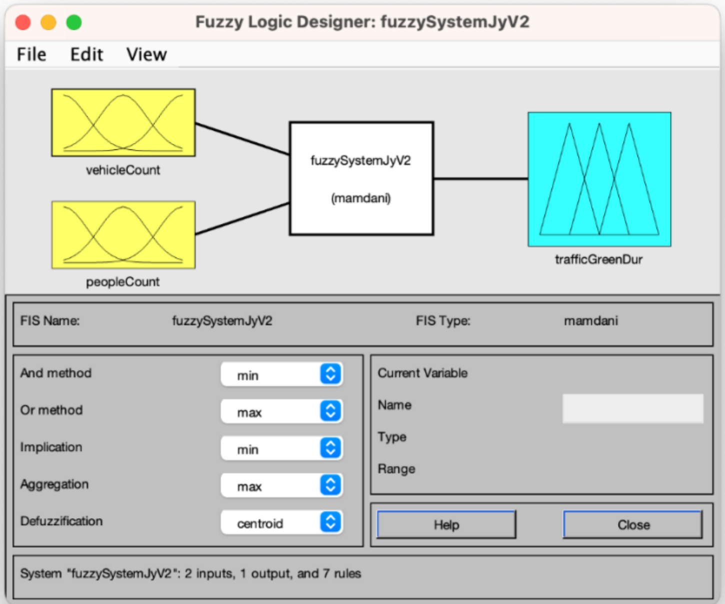


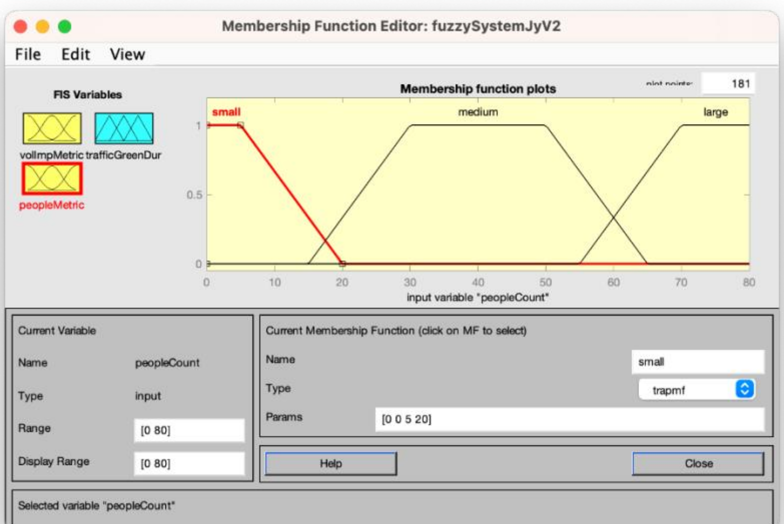


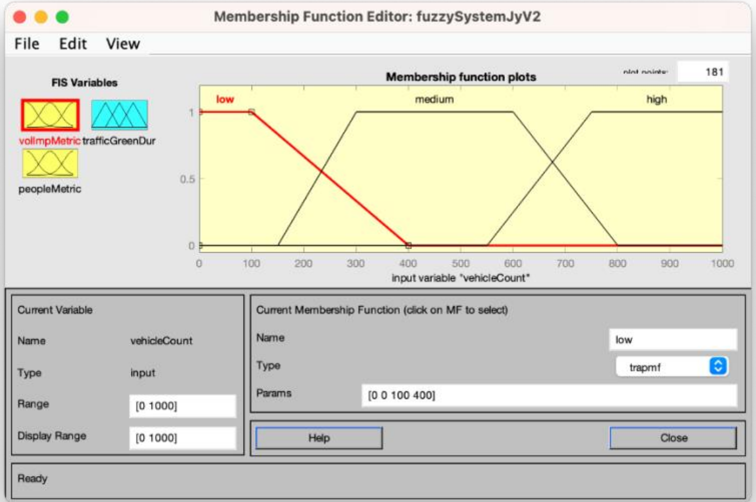

Rule Viewer: fuzzySystemJyV2
File Edit View Options
vehicleCount = 833
peopleCount = 70.4
trafficGreenDur = 90
1
2
3
4
5
6
7
0
1000
0
80
0
180
Input: [833.3;70.36]
Plot points: 101
Move: left right down up
Opened system fuzzySystemJyV2, 7 rules
Help
Close


Surface Viewer: fuzzySystemJyV2
File Edit View Options
trafficGreenDur
140
120
100
80
80
60
40
20
0
peopleCount
0
200
400
600
vehicleCount
X (input): vehicle...
Y (input): people...
Z (output): trafficG...
X Mesh Points: 15
Y Mesh Points: 15
Evaluate
Ref. Input:
Plot points: 101
Help
Close
Ready


Membership Function Editor: fuzzySystemJyV2
File
Edit
View
FIS Variables
volImpMetric
trafficGreenDur
peopleMetric
Membership function plots
181
short
medium
long
output variable "trafficGreenDur"
Current Variable
Name
trafficGreenDur
Type
output
Range
[0 180]
Display Range
[0 180]
Current Membership Function (click on MF to select)
Name
short
Type
trapmf
Params
[0 0 30 90]
Help
Close
Selected variable "trafficGreenDur"


Fuzzy Logic Designer: fuzzySystemJyV2
File Edit View
vehicleCount
peopleCount
fuzzySystemJyV2
(mamdani)
trafficGreenDur
FIS Name: fuzzySystemJyV2
FIS Type: mamdani
And method min
Or method max
Implication min
Aggregation max
Defuzzification centroid
Current Variable
Name
Type
Range
Help
Close
System "fuzzySystemJyV2": 2 inputs, 1 output, and 7 rules


Membership Function Editor: fuzzySystemJyV2
File Edit View
FIS Variables
volImpMetric
trafficGreenDur
peopleMetric
Membership function plots
181
small
medium
large
input variable "peopleCount"
Current Variable
Name peopleCount
Type input
Range [0 80]
Display Range [0 80]
Current Membership Function (click on MF to select)
Name small
Type trapmf
Params [0 0 5 20]
Help
Close
Selected variable "peopleCount"


Membership Function Editor: fuzzySystemJyV2
File
Edit
View
FIS Variables
volImpMetric
trafficGreenDur
peopleMetric
Membership function plots
181
low
medium
high
input variable "vehicleCount"
Current Variable
Name
vehicleCount
Type
input
Range
[0 1000]
Display Range
[0 1000]
Current Membership Function (click on MF to select)
Name
low
Type
trapmf
Params
[0 0 100 400]
Help
Close
Ready

User Guide

This user guide will show how the user can simply run a program and retrieve traffic information from traffic camera images from roads in Singapore. At the same time, the data collected can be used to determine the traffic green light duration by feeding the volume-importance metric to the fuzzy system.

How to use our AI solution?

Our AI solution consists of primarily two parts: the vehicle detection, classification and counting and the fuzzy system used to determine the duration of the traffic green light at the point of view of the vehicles.

Global configuration file

Below is a screenshot of the file and I have explained in good detail what each of them does.

```
## Image Pre-Processing Configuration
FocusDate = "2023-03-23"

## Traffic Camera Configuration
CameraID = "3795"
ImageCropArea = "486,1075,807,297,1046,285,1454,1078"

## Weights Configuration for Detection and Volume-Importance
mode = "validate"   # predict or validate

## Predict Configuration
conf_threshold = 0.25

## Validation Configuration
model = "jianyue_model.pt"
val_tr = "val_tr_config.yaml"

## Vehicle weights
ambulance_w = 500
bus_w = 35
car_w = 4
lorry_w = 10
motorcycle_w = 2
truck_w = 20
```

Data to focus on after data acquisition step

Camera to focus on after data acquisition

Coordinates to specify the crop area

The mode during real-life scenarios is usually "predict". User can change to validate to see performance of model

Any predictions below this will be removed

The model you want to use

Unless using "validate", this is not required

Each of the weights of the vehicles, entirely configurable, but remember to configure the fuzzy system too!

Retrieving real-time images

To retrieve real-time traffic images, this can be done by running the script with a required account key. All the real-time traffic images will be downloaded into a folder “images” and in the folder of the date of when the script is being run. This can be done on a computer specialising in data acquisition, while another computer retrieves the images. In my solution, the computer is set-up to execute the script every few minutes to download images from all traffic cameras available in Singapore for a few days and configured to automatically upload the images periodically to a private GitHub repository.

As to not expose my account key, the code is not included in this paper.

Vehicle detection and counting

Users can configure the importance weights for each type of vehicle in a separate configuration file (*config.py*) without directly touching the Python script. As mentioned, these weights are important as they determine the volume-importance vehicle metric used to estimate the green light signal time. These weights can be changed by using other AI or machine learning methods to fit the situation or scenario, as different roads and traffic conditions and different countries' traffic conditions may require different weights configured.

Generating detected images with bounding boxes

For us to manually evaluate the results as with another similar study, we would need images with bounding boxes drawn onto each of the original images.

Additionally, I have chosen to only include classes that are relevant in our paper. These are the classes that are found in the YOLO model and are required in our study.

For example, the command after the run:

```
C:\Users\jiany\Downloads\github\[redacted]\images\2023-03-23_3795>yolo predict model=
yolov8x.pt save=True source=* classes=[2,3,5,7]
C:\Users\jiany\AppData\Local\Programs\Python\Python310\lib\site-packages\requests\__init__.py:102: Re
questsDependencyWarning: urllib3 (1.26.12) or chardet (5.0.0)/charset_normalizer (2.0.12) doesn't mat
ch a supported version!
  warnings.warn("urllib3 ({}) or chardet ({})/charset_normalizer ({}) doesn't match a supported "
Ultralytics YOLOv8.0.40  Python-3.10.8 torch-1.13.1+cu117 CPU
YOLOv8x summary (fused): 268 layers, 68200608 parameters, 0 gradients, 257.8 GFLOPs

image 1/9 C:\Users\jiany\Downloads\github\[redacted]\images\2023-03-23_3795\_2023-03-
23_07-55_3795_0753_20230323075635_494954_cropped.jpg: 384x640 2 cars, 1 truck, 669.6ms
image 2/9 C:\Users\jiany\Downloads\github\[redacted]\images\2023-03-23_3795\_2023-03-
23_08-05_3795_0803_20230323080536_515398_cropped.jpg: 384x640 4 cars, 691.9ms
image 3/9 C:\Users\jiany\Downloads\github\[redacted]\images\2023-03-23_3795\_2023-03-
23_08-35_3795_0833_20230323083523_515050_cropped.jpg: 384x640 5 cars, 1 bus, 585.1ms
image 4/9 C:\Users\jiany\Downloads\github\[redacted]\images\2023-03-23_3795\_2023-03-
23_08-45_3795_0843_20230323084619_100545_cropped.jpg: 384x640 9 cars, 1 bus, 1 truck, 638.1ms
image 5/9 C:\Users\jiany\Downloads\github\[redacted]\images\2023-03-23_3795\_2023-03-
23_09-00_3795_0858_20230323090105_984950_cropped.jpg: 384x640 7 cars, 1 truck, 663.6ms
image 6/9 C:\Users\jiany\Downloads\github\[redacted]\images\2023-03-23_3795\_2023-03-
23_09-35_3795_0933_20230323093600_102561_cropped.jpg: 384x640 3 cars, 1 truck, 647.6ms
image 7/9 C:\Users\jiany\Downloads\github\[redacted]\images\2023-03-23_3795\_2023-03-
23_17-15_3795_1713_20230323171529_102101_cropped.jpg: 384x640 4 cars, 1 bus, 737.8ms
image 8/9 C:\Users\jiany\Downloads\github\[redacted]\images\2023-03-23_3795\_2023-03-
23_18-55_3795_1853_20230323185508_981025_cropped.jpg: 384x640 2 cars, 1 truck, 643.1ms
image 9/9 C:\Users\jiany\Downloads\github\[redacted]\images\2023-03-23_3795\bus.jpg:
640x480 1 bus, 895.6ms
Speed: 0.2ms pre-process, 685.8ms inference, 1.0ms postprocess per image at shape (1, 3, 640, 640)
Results saved to runs\detect\predict9
```

The script will output the volume-importance metric as the final output that should be copied and pasted to the fuzzy-rule based system below.

In [81]:
```
volume_imp = car_w * car_no + bus_w * bus_no + truck_w * truck_no

volume_imp
```

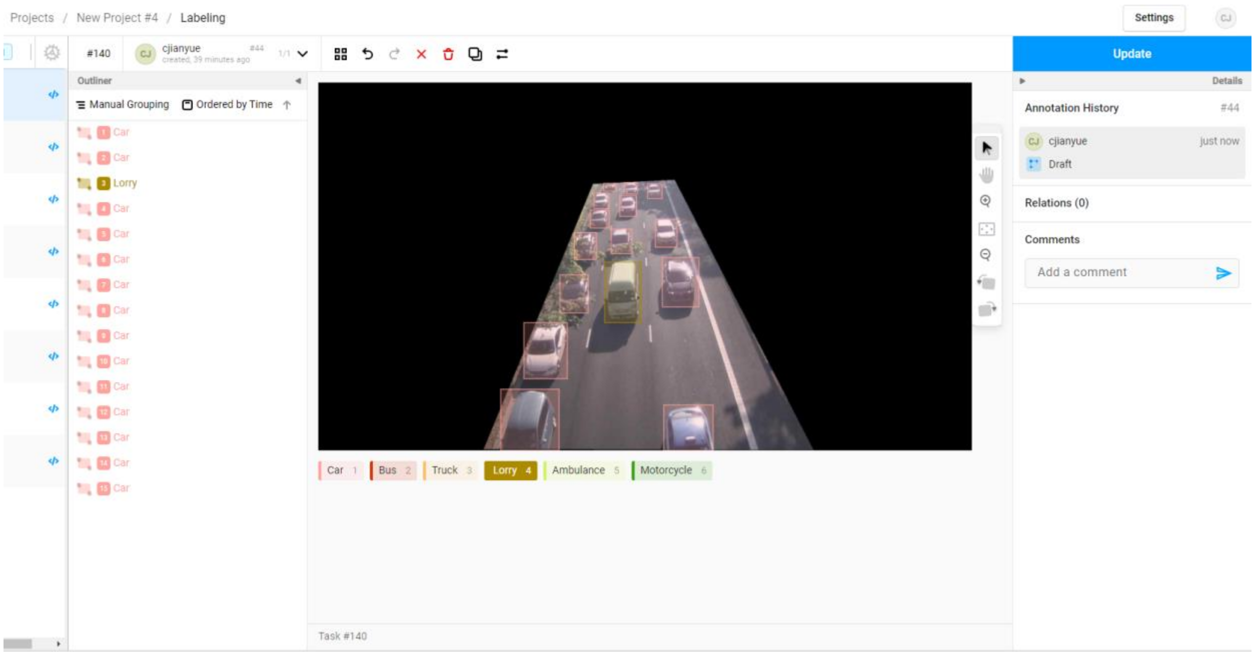
Out[81]:
```
35
```

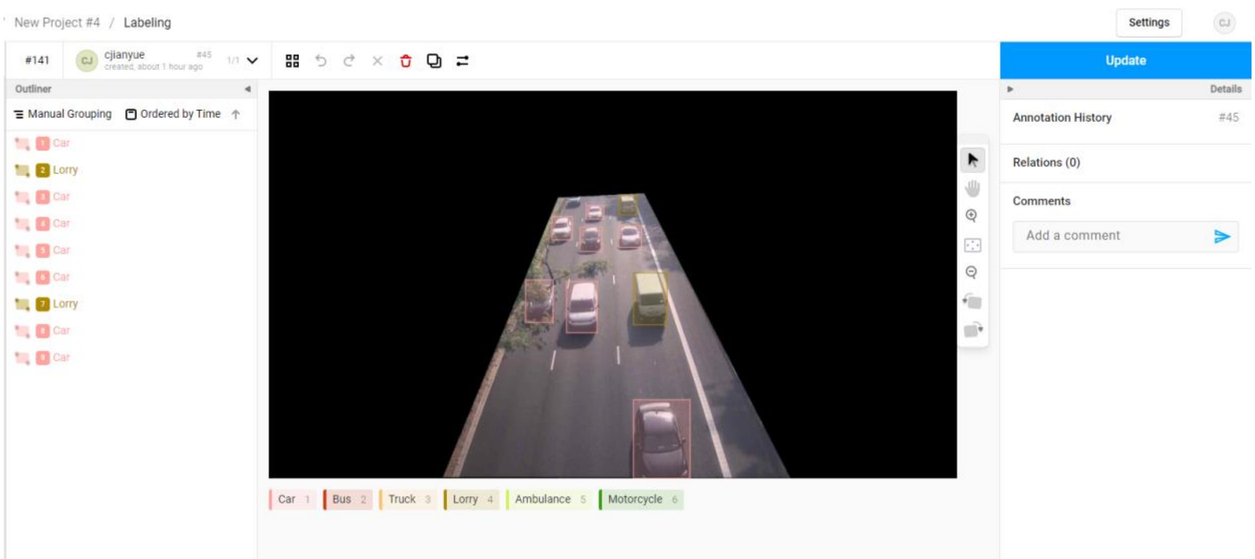
Training other vehicle types

User can train other vehicle types if they have labelled image data.

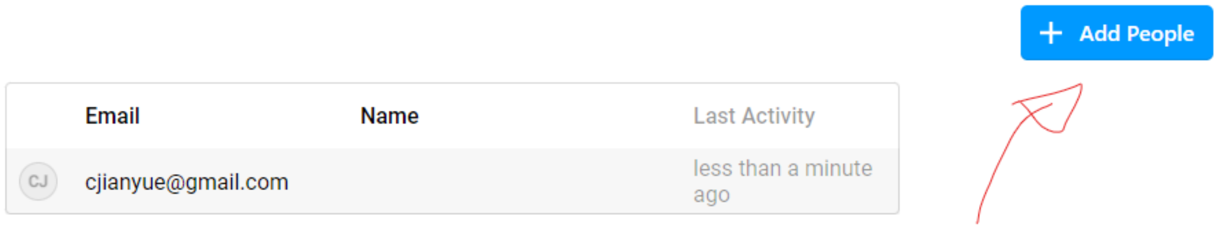
Labelling data

Data labelling is important for the user to make the model better at detection and classification in model training. There are many platforms that support data labelling available. The user can include other labels that are not shown and used in our paper. One of such platforms are shown below, which was utilised to validate our model performance, and take note of the different colours representing different vehicles. Try to draw bounding boxes as tightly to the vehicles as possible.



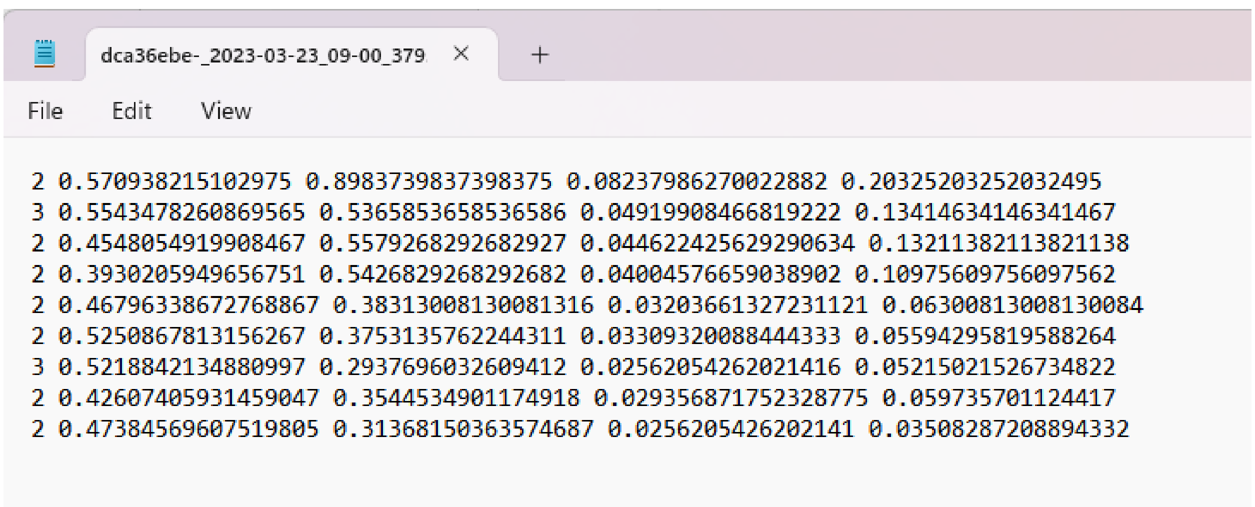


In general, the user should select the type of vehicle and draw bounding boxes around the vehicle on the image. Our guideline to data labelling is that to only label the entire vehicle if they are seen in the image. In that way, we are surer of the vehicle that is being drawn in the bounding box and the vehicle type. However, labelling data is a tiring and tedious process and is time consuming, there could be challenges in hiring sufficient manpower to process the large number of images and especially if there are many vehicles in the image, you can assign more people to help with the process of labelling:



Hence, the training and validation portion is a scalable solution, as long as there is enough manpower and the documentation is clear on how it should be labelled.

Sample image labels are provided, and a screenshot of what it looks like is as follows:




```
2 0.570938215102975 0.8983739837398375 0.08237986270022882 0.20325203252032495
3 0.5543478260869565 0.5365853658536586 0.04919908466819222 0.13414634146341467
2 0.4548054919908467 0.5579268292682927 0.044622425629290634 0.13211382113821138
2 0.3930205949656751 0.5426829268292682 0.04004576659038902 0.10975609756097562
2 0.46796338672768867 0.38313008130081316 0.03203661327231121 0.06300813008130084
2 0.5250867813156267 0.3753135762244311 0.03309320088444333 0.05594295819588264
3 0.5218842134880997 0.2937696032609412 0.02562054262021416 0.05215021526734822
2 0.42607405931459047 0.3544534901174918 0.029356871752328775 0.059735701124417
2 0.47384569607519805 0.31368150363574687 0.0256205426202141 0.03508287208894332
```

The first value of each row represents the index of the class while the four other values are the coordinates of the labelled bounding boxes.

Training and validation configuration file

As the YOLO model that we are using requires a separate configuration file, at the moment, we are unable to integrate it in our global configuration file. However, a sample configuration file is provided and needs to be modified based on the workstation used.

Fuzzy rule-based system

The volume-importance vehicle metric and the people count metric at the traffic junction are the variables used in the fuzzy system model to determine the traffic green light duration.

The fuzzy rule-based system values can be configured with the MATLAB program. This includes the de-fuzzification techniques, the range and the membership functions' shape and values. There are two ways to extract the green traffic light duration. The easier way to obtain the numerical value to be fed to a traffic controller system is via the MATLAB command line. The examples are as follows:

```
>> greenLightTime = evalfis(fuzzySystemJyV2, [100,80])

greenLightTime =

   64.4595
```

```
>> greenLightTime = evalfis(fuzzySystemJyV2, [400,10])

greenLightTime =

   95.2040
```

In particular, the change the value where the red arrow is pointing at to configure the volume-importance vehicle metric (a measurement of traffic conditions) and the people count or metric which could be detected at the traffic junction.

```
>> greenLightTime = evalfis(fuzzySystemJyV2, [100,80])

greenLightTime =

   64.4595
```

Volume-Importance vehicle metric

People count/metric

Product Demo Video

- Full presentation and slides are uploaded separately
- Demo Video is found in the same zip folder.