School of Computer Sciences

CAT405 Intelligent Computing Major Project

Final Report

Vehicle Counting and Classification System

Mohammed Tayfour Abdalla Mohammed

140562

Supervisor: Dr. Mohd Nadhir Bin Ab Wahab

Examiner: Dr. Mohd Halim Bin Mohd Noor

Academic Session

2022/2023

DECLARATION

“I declare that the following is my own work and does not contain any ***unacknowledged*** work from any other sources. This report was undertaken to fulfill the requirements of the Undergraduate Major Project for the Bachelor of Science in Computer Science (Honors) program at Universiti Sains Malaysia”.

Signature :

Name : Mohammed Tayfour Abdalla Mohammed

Date : 18/06/2023

ABSTRAK

Pada masa ini, kenderaan di Malaysia, dan di banyak bahagian dunia tidak dikira dengan cekap. Ia selalunya sama ada dikira secara manual, yang boleh mengakibatkan banyak percanggahan dan ralat, atau di pondok tol lebuh raya. Jenis kenderaan juga dikesan dan dikelaskan, tetapi dalam kebanyakan kes, secara berasingan. Oleh itu, data dan cerapan yang dihasilkan adalah kurang mencukupi. Lebih banyak cerapan berharga boleh ditemui. Mengambil kira perkara ini, bagi projek tahun akhir CAT405 sesi 2022/2023, saya di bawah seliaan Dr Mohd Nadhir Bin Ab Wahab telah memilih untuk mengusahakan pembikinan Sistem Kiraan dan Klasifikasi Kenderaan. Sistem berpusat ini akan melaksanakan semua tugas ini. Satu Teknik Pembelajaran Mendalam akan dibangunkan untuk mengklasifikasikan jenis kenderaan dan mengiranya dalam satu set sampel rakaman yang diambil daripada kamera CCTV oleh Majlis Bandaraya Seberang Perai (MBSP) dan Majlis Bandaraya Pulau Pinang (MBPP). Data yang dihasilkan daripada pengelasan dan pengiraan akan dianalisis dan dipaparkan melalui sistem papan pemuka untuk tatapan umum. Setelah selesai pembangunannya, sistem ini akan membantu orang awam membuat ramalan dan pandangan yang lebih tepat dan boleh dipercayai. Projek ini terutamanya akan menangani Matlamat Pembangunan Mampan (SDG) 11, bertajuk "Bandar dan komuniti lestari".

ABSTRACT

At the moment, vehicles in Malaysia, and in many parts of the world aren't counted efficiently. They are often either counted manually, which can result in many discrepancies and errors, or on highway toll booths. Vehicle types are also detected and classified, but in most cases, separately. As such, the data and insights produced are less than sufficient. More valuable insights can be discovered. Taking this into account, for the CAT405 final year project session 2022/2023, I, under the supervision of Dr. Mohd Nadhir Bin Ab Wahab have chosen to work on making the Vehicle Counting and Classification System, a centralized system that will perform all of these tasks. A Deep Learning Technique will be developed in order to classify the vehicle types and count them in a set of sample footage taken from CCTV cameras by Majlis Bandaraya Seberang Perai (MBSP) and Majlis Bandaraya Pulau Pinang (MBPP). The data produced from the classification and counting will be analyzed and displayed through a dashboard system for the general public to view. Upon the completion of its development, the system will help the general public make more accurate and reliable predictions and insights. This project will mainly tackle the Sustainable Development Goal (SDG) 11, titled "Sustainable cities and communities".

ACKNOWLEDGEMENTS

In my life, I have many to thank, but I would like to begin by thanking Allah, for hearing and answering my prayers whenever I am in need, and for blessing me with the most wonderful company.

Mama, Baba, Ma'athir, Shaima, and Ahmed. Thank you all for being so patient and supportive of me through every decision that I've ever made. Everything I do in life is in your honor, and so I dedicate this to you.

I would like to express my deepest gratitude to my supervisor, Dr. Mohd Nadhir Ab Wahab, whose support during both my internship and final year has been tremendous and for giving me the opportunity to work on a project that helped me grow not only as a student, but as a person as well. I would, by extension, like to thank the School of Computer Sciences for giving me the experience of a life time, and for teaching me lessons that I will carry on with me wherever I go.

I would like to thank my brother, whom I love very much, Syed Wajhi Hassan, for being the most patient, and kind friend that any man could ask for.

And of course, I would like to that Safiya Nima Messaya, ma lune. Thank you for pushing me to work hard and for seeing something in me. I'm blessed to have you in my life. Merci for everything.

This is for all of you.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS AND SYMBOLS

YOLO – You Only Look Once

1 INTRODUCTION

1.1. Background

This project is aimed towards the field of Intelligent Computing and Data Analytics. Currently vehicles are being counted manually, with the exception of toll booths that are capable of keeping records of vehicles passing, without being able to know what type of vehicle they are. Vehicle detection and classification are also done separately. The insights discovered from each of these parts separately isn't sufficient. Therefore, in view of this issue, the Vehicle Counting and Classification system is proposed. The aim of this project is to unify all 3 tasks by performing the aforementioned functions. Using CCTV footage obtained from Majlis Bandaraya Seberang Perai (MBSP) and Majlis Bandaraya Pulau Pinang (MBPP), a deep learning technique was applied using Python programming language to help detect, count the number of vehicles, as well as classify the vehicles according to their types (Car, Motor, Truck, Lorry, etc.). The data obtained is then presented on a dashboard for the general public to see. This system would ideally be tackling on Sustainable Development Goal 11 titled "sustainable cities and communities", allowing universities and other companies to make use of this data for security purposes as well as government bodies to aid in alleviating traffic jams/accidents that may occur and allowing us to discover more valuable insights from the data, and better understand traffic situations overall.

1.2. Problem Statements

According to the New Straits Times, it was reported that the number of vehicles in Malaysia had completely overtaken the human population, with an increase of roughly one million vehicles per year since 2019, outpacing the human population growth for the first time. "The government needs to address worsening traffic congestion issue now, not wait for studies to be done" is what Malaysian politician and former transport minister Anthony Loke said on the 27th of May 2022. There is an impending issue of traffic congestion, increasing the risk of vehicular accidents. The increase in the number of vehicles may prove to be a security issue as it increasingly becomes more and more difficult to keep track of things. According to News Trail, 20 to 50 million injuries are sustained in car crashes each year worldwide. 3 million car accident fatalities happen annually worldwide. That equates to 3,287 deaths per day.

However, this data isn't enough. More insights and discoveries can be unearthed. Counting the number of vehicles in a highway or road, classifying them according to their types could allow us to discover more valuable insights on top of the ones already discover. This data could allow the government and general public better manage road safety issues as well as managing traffic. More trends can be discovered, allowing people to make better informed decisions and more accurate predictions. Currently, there are no signs of a system that detects, counts, classifies, and present all of the data in an online web dashboard. However it can be done.

1.3. Motivation

With the alarming rate of increase in the number of vehicles/traffic as well as the dramatic increase in traffic congestion and accidents as a result of that, a system is needed in order to analyze and connect all of the data gathered and present it on a dashboard for the general public to access and work with.

1.4. System Objectives

The vehicle counting and classification system will be divided into three main modules:

1. To develop a system that's able to detect and classify the types of vehicles (Car, Motor, Truck, Lorry, etc.).
2. To count the number of vehicles of each type and produce the data.
3. To provide visual information via a web-based dashboard based on the collected data, for people to see.

These modules outline how the parts are intertwined with one another and how they work together in order for the entire system to function properly.

1.5. Proposed Solutions

The vehicle counting and classification system is the proposed solution to tackle the issue of traffic congestion/accidents, security problems, and that fact that there isn't a dashboard available that provides accessible and reliable information for the general public to keep them informed.

The detection, counting, and classification modules will be developed using Python programming language. The CCTV footage is uploaded as input and then through the use of Yolo v5, the number and classification of vehicles will be detected and classified. A dashboard will be built to visualize the data acquired from the footage.

1.6. Benefits and Uniqueness of the Proposed Solutions

The official mission of the sustainable development goal 11 is to make cities more inclusive, safe, resilient, and sustainable. This is what the vehicle counting and classification system is intended to tackle. Smart traffic to optimize traffic flow in order to reduce congestion and the risk of collisions as well as better security are one of the many benefits that this project offers including providing valuable insights from the data produced.

Vehicle detection, counting, and classification isn't a new idea/system. It has been implemented before using a plethora of methods. However, upon doing research, there have been no signs of a web-based dashboard system integrating everything and displaying analytics and statistics for the public to view and make use of. The vehicle counting and classification system unifies all of these tasks and place them in a way that's accessible to the general public.

1.7. Organization of the Report

Now that the idea of the project has been introduced. More details on it will be expanded on further. Background and related works will be discussed, followed by the system requirements and design where details of the development methodology and processes will be discussed with further descriptions of the modules of the system. The

2 BACKGROUND & RELATED WORK

As previously stated, vehicle detection, counting, and classification isn't a new idea/system. It has been implemented before using a plethora of methods. In this section, similar works will be discussed in depth.

In a paper titled "A Video based Vehicle Detection, Counting and Classification System" written by Sheeraz Memon and colleagues, of Mehran University of Engineering & Technology, Pakistan, a vision based vehicle counting and classification system was proposed. The system involved capturing frames from the videos to perform background subtraction in order detect and count the vehicles using Gaussian Mixture Model (GMM) background subtraction then it would further classify the vehicles by comparing the contour areas to the values. The main aim of the work was the comparison of two classification methods. Classification has been implemented using Contour Comparison as well as Bag of Features and Support Vector Machine (SVM) method. This proposed work was met with many limitations, namely with its inability to count and classify accurately without proper lighting or if there's any traffic congestion [1].

Luis Unzueta and colleagues also proposed a similar system in a paper titled "Adaptive Multicue Background Subtraction for Robust Vehicle Counting and Classification". The focus of the project was to present a robust vision-based system for vehicle tracking and classification. The system would perform in real time and in various conditions such as with moving casted shadows on sunny days, headlight reflections on the road, rainy days, and traffic jams, using only a single standard camera. The approach would, however, need further extension to deal with severe occlusions [2].

Shiva Kamkar and Reza Safabakhsh of the Department of Computer Engineering and Information Technology at Amirkabir University of Technology, Iran, proposed a vehicle detection method which selects vehicles using an active basis model and verifies them according to their reflection symmetry in a paper titled "Vehicle detection, counting and classification in various conditions". The vehicles were counted and classified by extracting two features: vehicle length in the corresponding time-spatial image and the correlation computed from the grey-level co-occurrence matrix of the vehicle image within its bounding box and a random forest was trained to classify vehicles into three categories: small, medium, and large. The experimental results showed a good performance, but some improvements such as using GPU programming or adding other texture-based features to the classification part were suggested to improve the algorithm performance and run-time [3].

Zhe Dai and colleagues proposed a video-based vehicle counting framework using a three-component process of object detection, object tracking, and trajectory processing to obtain the traffic flow information in a paper titled "Video-Based Vehicle Counting Framework". 2 datasets are established, and then using YOLO v3, they were able to detect moving vehicles. However, no specific algorithm to classify the vehicles was used and it was only limited to cars, buses, and trucks [4].

3 SYSTEM REQUIREMENTS / ANALYSIS

3.1. Project scope

In this section, more details on the system requirements and design are discussed. The vehicle counting and classification system is divided into three main modules:

1. Detect and classify the types of vehicles (Car, Motor, Truck, Lorry, etc.).
2. Count the number of vehicles of each type and produce data to be stored.
3. Visualize the data on a dashboard.

The main aim of this project was to use the provided video footage obtained from Majlis Bandaraya Seberang Perai (MBSP) and Majlis Bandaraya Pulau Pinang (MBPP) and use deep learning techniques to count and classify vehicles from that footage. From the data obtained, a dashboard was be built to visualize everything, making it more presentable to users.

3.2. Project management

It's important to manage and organize the work required to build the vehicle counting and classification system. Figure 1.0 is the work breakdown structure for the project.

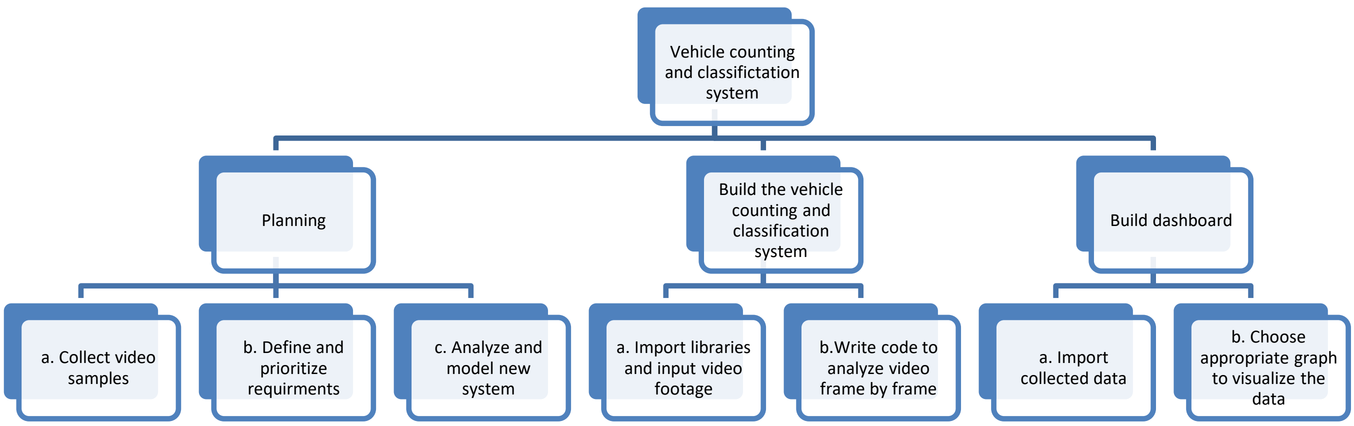


Fig 1.0: Work breakdown structure for the vehicle counting and classification system

A Gantt chart is a great way to measure the overall progress of the project and allows for better time management. Table 1.0 is the Gantt chart modelled after the Work Breakdown Structure that was introduced earlier.

Task	Start Date	End Date	Duration Work (Weeks)	Vehicle counting and classification system								
				October	November	December	January	February	March	April	May	June
Planning												
a.	17/10/2022	1/11/2022	2									
b.	1/11/2022	1/12/2022	4									
c.	1/12/2022	1/1/2023	4									
Building the vehicle counting and classification system												
a.	20/12/2022	15/2/2023	8									
b.	15/2/2023	1/4/2023	6									
Building dashboard												
a.	1/4/2023	14/5/2023	6									
b.	14/5/2023	1/6/2023	2									

Table 1.0: Gantt chart for the vehicle counting and classification system

The major milestones for developing the system are also shown in the figure below.

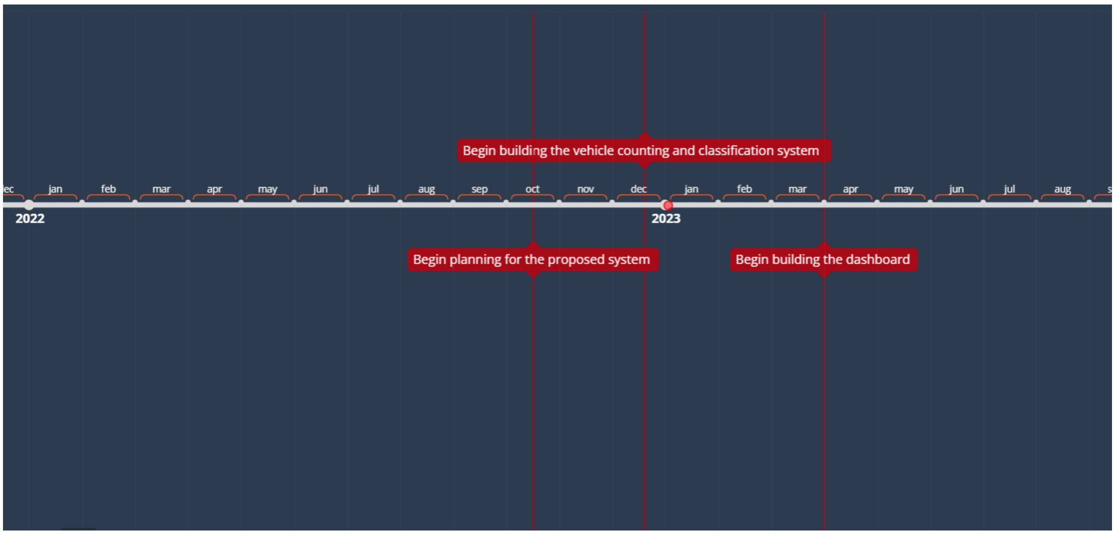


Figure 1.1: Milestone timeline for the vehicle counting and classification system

A SWOT (Strength, Weaknesses, Opportunities, and Threats) analysis was performed to assist focusing on the strengths, minimize the threats and take the greatest possible advantage of opportunities available.

Strength	Weaknesses
1. The general public has access to a platform that allows them to better understand traffic 2. The system would allow companies/government bodies to better understand traffic in specific areas	1. Requires a stable and strong internet connection in order to access and view the dashboard
Opportunities	**Threats**
1. Not many competitors focus on a web dashboard to visualize the data. Most of the systems similar to this are done for research purposes	1. Poor network connection can greatly affect the system as the dashboard can't be accessed 2. The right model needs to be selected in order to accurately detect and classify them

Table 1.1 SWOT analysis

3.3. Development Methodology

For the vehicle counting and classification system, the agile methodology was the most appropriate one to follow. It's an iterative methodology in that the project is broken up into phases. Its flexibility made it ideal when compared to a more rigid approach such as the Waterfall methodology as many changes can occur during the development process including, but not limited to, further improvements down the line.

3.4. Analysis of the new system

Use case diagram

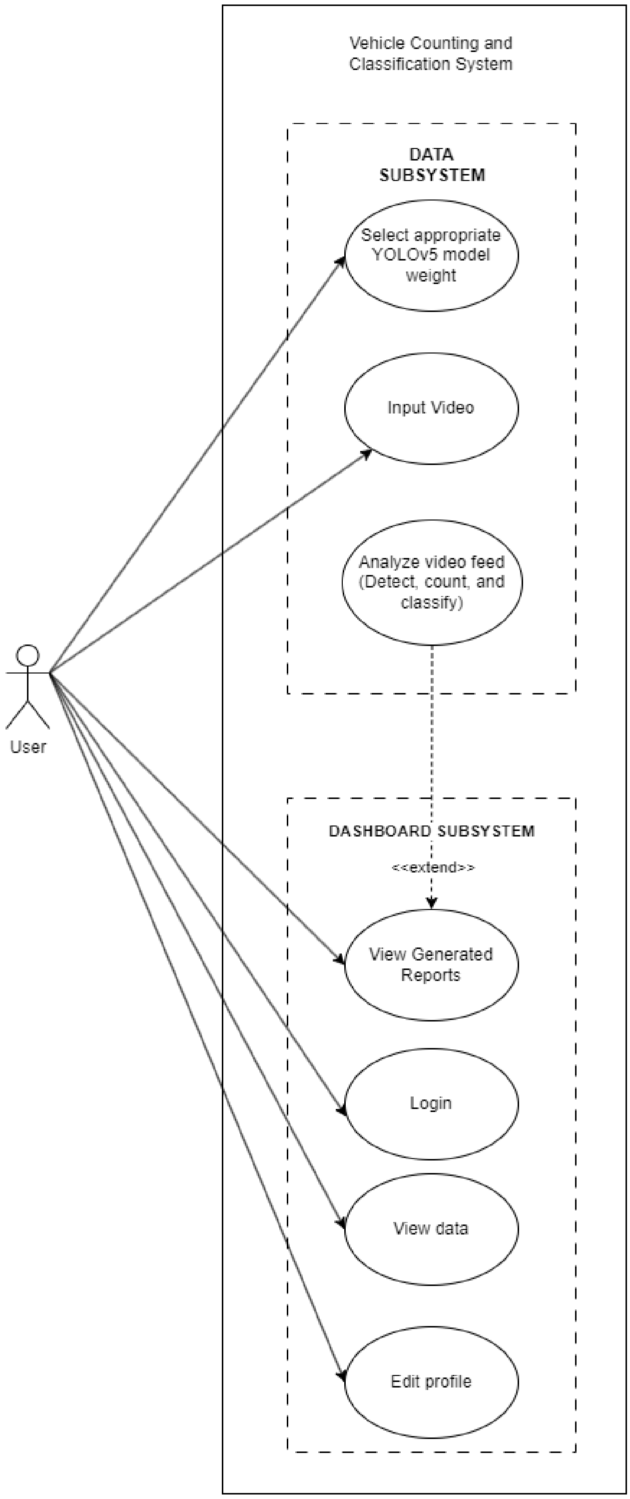


Fig 1.2 Use case diagram

For each of the use cases illustrated in figure 1.3, a brief description is provided:

Use Case	Description
Select appropriate YOLO v5 model weight	Users can decide the appropriate YOLO model for their footage
Input video	Users can input the video into the system
Analyze video feed	The video is monitored and analyzed.
View generated reports	The collected data will display results of the detection
Log in	Users can log into their created account
View data	Users can visit the data page to view the raw data
Edit profile	Users can edit their details in

Use case related models which include fully-developed use case description and system sequence diagram for use cases that are involved in the main function of the overall system are available further in the Appendix A.

Below are Sequence Diagrams built from the Use Case diagrams.

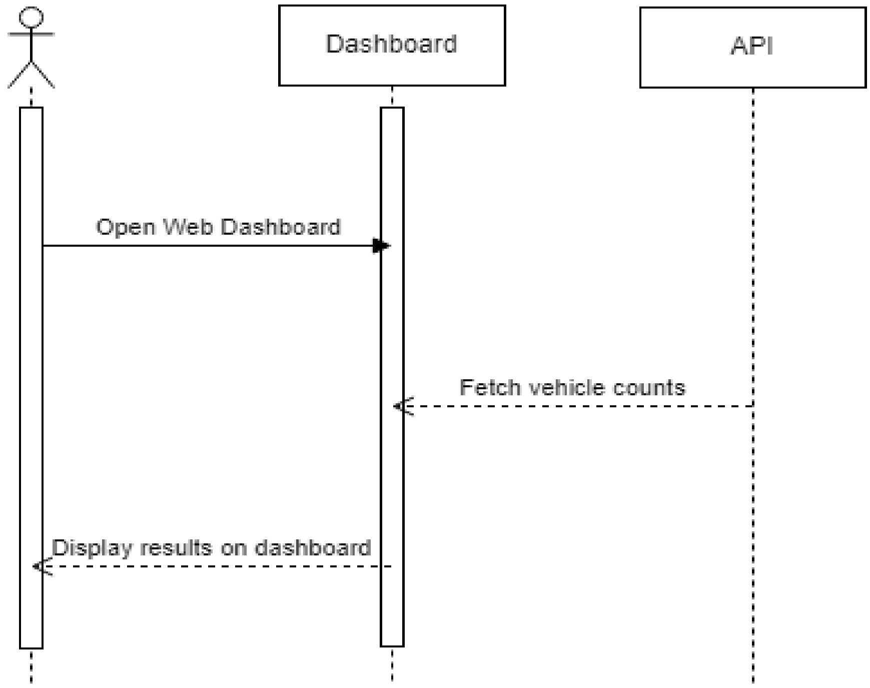


Fig 1.3 Sequence Diagram for dashboard

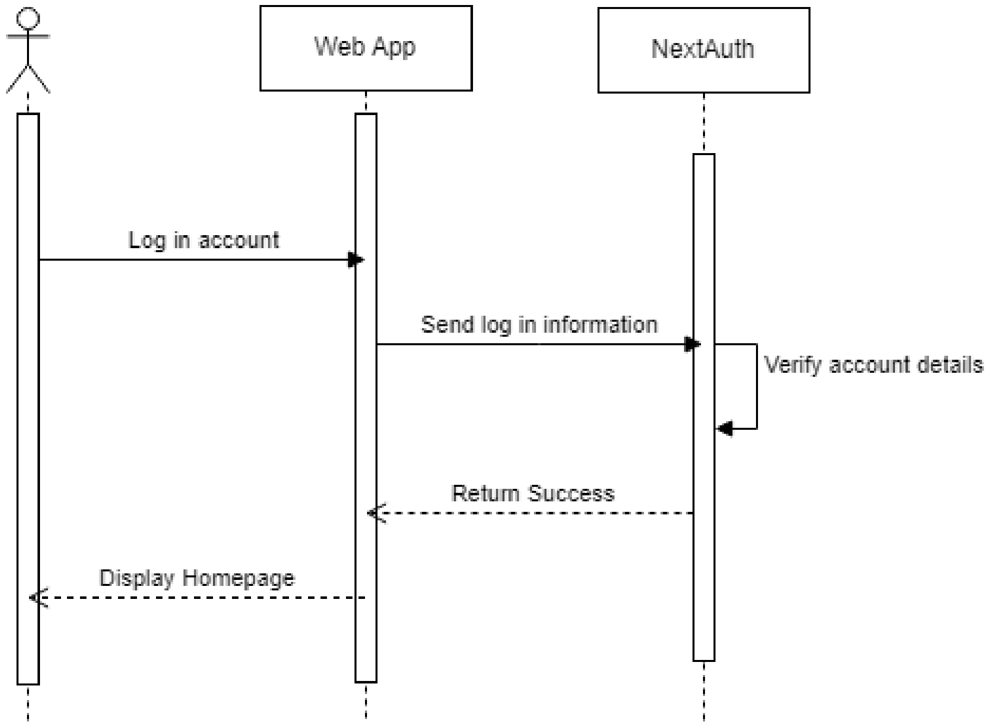


Fig 1.4. Sequence diagram for log in

3.5. Description of the proposed system

The vehicle counting and classification system is divided into three main modules:

1. Detect and classify the types of vehicles (Car, Motor, Truck, Lorry, etc.)
2. Count the number of vehicles of each type and produce data based on the defined time intervals
3. Visualize the data on a dashboard

The data will be uploaded to a web based dashboard to be visualized for everyone with an account.

3.6. Description of data source

As previously stated, footage taken from CCTV cameras by Majlis Bandaraya Seberang Perai (MBSP) and Majlis Bandaraya Pulau Pinang (MBPP) as shown in the sample screenshot below, was used in order to count and classify the vehicles. In doing so, data including the type of vehicles and their total number will be acquired. This data is then visualized on a dashboard. Figure 1.6 is a screenshot of the acquired sample footage from MBPP.

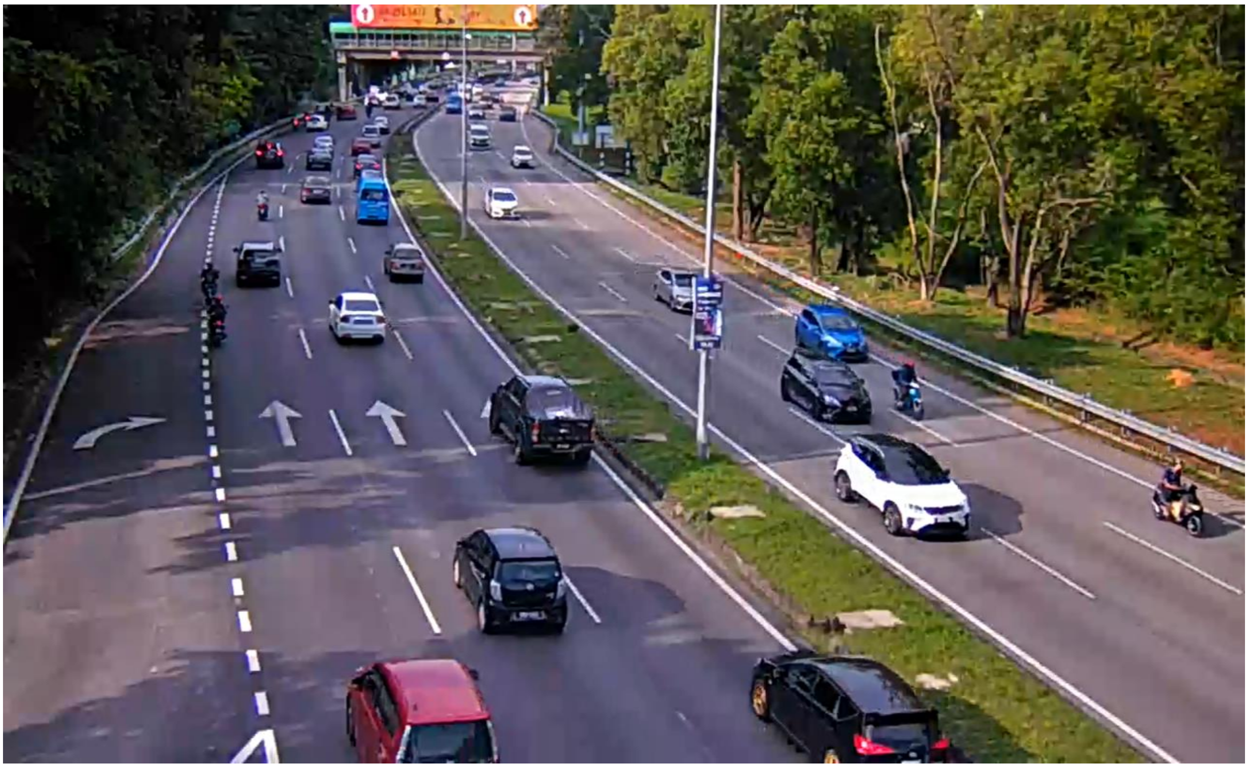


Fig 1.5: Screenshot of the acquired sample CCTV footage from MBPP

3.7. Technology deployed

The vehicle counting and classification system developed using the programming language Python using the IDE Pycharm. The CCTV footage will be uploaded in the program as input, after which the number and type of vehicles will be counted, classified, and then the data acquired will be stored in an excel sheet for further use. The web development framework, Next.js, accompanied by the programming language, Typescript on VS Code is used to build a dashboard in order to visualize the stored data in a neat and accessible way for users. All of this will be done on a laptop with the following specifications. Specifications of the operating system are also included:

Hardware specifications:

Processor	Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz 2.00 GHz
Installed	RAM 16.0 GB
System type	64-bit operating system, x64-based processor

Software specifications:

Edition	Windows 11 Home
Version	22H2
OS build	22621.963
Experience	Windows Feature Experience Pack 1000.22638.1000.0

4 SYSTEM DESIGN & IMPLEMENTATION

4.1. Architecture

As shown in the figure below, the Vehicle Counting and Classification System deploys many technologies in order to work.

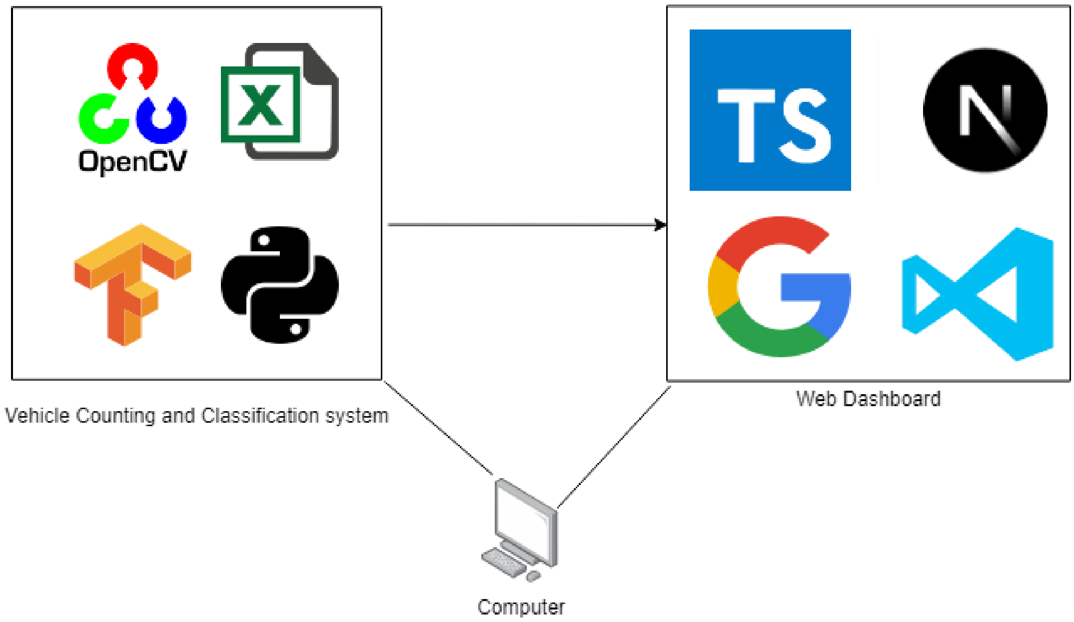


Fig 1.7 System Architecture

4.2. Implementation

The Vehicle Counting and Classification system as a whole can be broken down to 2 separate systems. The first is the counting and classification system. Written in Python language in the PyCharm IDE, the implementation of intelligent computing methods in this system revolves around leveraging the YOLOv5 model for object detection and employing algorithms for tracking, filtering, and counting vehicles. These methods facilitate automated and efficient vehicle counting and tracking within the provided video.

The video frames are processed using the YOLOv5 model to detect objects, particularly vehicles. Each frame is analyzed by the model, generating detection results that consist of bounding box coordinates, confidence scores, and object class labels. The detection results undergo filtering based on 2 thresholds. A confidence threshold, with only detections with a higher confidence than the threshold are considered, and a frame disappearance threshold. Moreover, non-maximum suppression (NMS) is applied to eliminate redundant bounding boxes that significantly overlap.

A tracking mechanism is implemented in or track each vehicle across frames. It compares newly detected bounding boxes with those from the previous frame, utilizing Intersection over Union (IoU) calculations. When a bounding box overlaps or is in close proximity to an existing tracked vehicle, the tracked vehicle's information is updated.

Visualizing the tracking results, bounding boxes are drawn around the tracked vehicles in each frame. Additionally, the frame displays the vehicle count for each vehicle type. Finally, the code saves the vehicle counts to a CSV file for further use.

The second system is the web-based data dashboard using Next.js, a JavaScript framework, Material UI (MUI), a design system, and Typescript on VS Code. The web dashboard is built by installing all of the necessary dependencies, creating pages, components, and subroutes. For logging in, NextAuth JS was used in order to allow for authentication.

Styling and theming on the dashboard was important as it needs to be visually appealing for the users. MUI inline styles were used, accompanied by CSS modules and SCSS. Which was how it was possible to create a light and dark theme for the web-dashboard too.

To protect the dashboard from being accessed by anyone, Route Guarding with NextAuth Middleware was used. With that, nobody could access the information in the data dashboard until they register for an account or log in.

Lastly, the data ribbons at the top of the dashboard including the charts and data cards with their statistics was made using Charts.js.

As previously stated, a CSV file was made at the end of the program that did the counting and classification. The next job was to place that data in the charts and ribbons and cards that were made for it. This was done by installing the ‘read-excel-file’ package. A new API route in next.js was also created to handle the file reading. All that was required was for the components to fetch the vehicle counts from the API endpoint.

Once that was done, the Vehicle Counting and Classification System was officially completed.

4.3. Design Modelling

4.3.1 Pseudocode for counting and classification:

```
Import required libraries

Load YOLOv5 model

Open video capture

Initialize vehicle_counts dictionary with initial counts for different vehicle types

Initialize prev_boxes as None

Initialize tracked_vehicles as an empty dictionary

Set confidence_threshold to 0.6

Set disappeared_threshold to 10

Set frame_count to 0

Read the first frame from the video capture

while True:

    Read next frame from the video capture

    if end of video is reached:

        break

    Pass frame through YOLOv5 model and get detection results

    Filter detections based on confidence threshold and remove overlapping boxes
```

Update tracked vehicles based on detected boxes and increment counts

Remove disappeared vehicles

Draw bounding boxes for tracked vehicles on the frame

Draw vehicle count on the frame

Display the frame

Check for user input to quit the process

Increment frame count

Create a DataFrame to store vehicle counts

Save the vehicle counts to a CSV file

Close windows and release resources

End

4.3 User Interface

The Vehicle Counting and Classification system has more than one user interface. Both need to work in tandem with one another for the entirety of the system to function. One the one hand, the counting and classification part of the project is done in the backend using Python. The user needs to put the path to the footage. They could decide the minimum confidence threshold for counting the vehicles as well depending on the level of accuracy they require. When a user visits the web dashboard, they are prompted to sign in as shown in the figure below. Once the user has registered/logged into their account, they are met with the home page which shows the dashboard. The users can interact with different parts of the system. They may also tailor the aesthetic of the site to fit their need, with a toggle button that allows them to shift from dark to light theme. Beyond that, there are other pages that the user can visit. The user can visit the data page. This page allows the user to upload the raw data and view it raw, without any visualization.

Another page that the user can visit is the profile section. As shown in the figure below, it allows the user to adjust details on their own profile. If they want, by ticking "Receive Traffic Analytics", the user can receive traffic analytics on their own email. The last page is the settings page, which allows you to adjust different features of the dashboard by allowing you to add or remove data that you may or may not want to see.

Screenshots of the user interface can be found in Appendix B.

4.4 Implementation Strategy

The implementation strategy used for the vehicle counting and classification system is a bottom up approach. Each module is reduced into smaller ones, focusing more on their specific task or functionalities. This would be beneficial in identifying where problems/errors occur when compared to the top-down approach where it would be harder to find. Each module is tested and debugged individually to ensure its correctness and reliability. This step is crucial as it helps identify and fix any issues or errors within the module. Once all the individual modules have been implemented and tested, they are combined and integrated to form the complete solution. The integration process involves connecting the modules together, passing data between them, and ensuring their cooperation.

4.5 Overview

To sum everything up, the Vehicle counting and classification system combines many modules to form a big one. Its design is very simple and accessible to anyone. It doesn't exhaust someone with too much data. Only what is necessary and important, and has so much more room to grow.

5 SYSTEM TESTING & EVALUATION

Software testing involves evaluating a software application in order to make sure that it meets the requirements that were specified and in order to identify any defects. In this section, we're going to see how the Vehicle Counting and Classification System was tested.

5.1 Testing Strategy

For the Vehicle Counting and Classification System, 2 testing strategies will be applied:

1. Black Box testing: This form of testing is performed under the premise that there is no knowledge of the software's internal code structure.
2. Regression testing: This involves making changes, to the software, and then making sure that those changes don't introduce any defects in the software

5.2 Test Cases

After the testing strategies were identified, the test cases were established. With regards to this specific system, an overall of 5 test cases were made. 3 are in relation to black box testing, with the other 2 being regression testing cases. The following table introduces those test cases.

The test case IDs were determined based on the testing strategy:

AB – Black Box Testing

AR – Regression testing

Detailed test reports can be found in Appendix C

Test Case ID	Test Case Name
AB0001	Vehicle Counting and Classification
AB0002	Log in
AB0003	View Dashboard

AB0004	View Raw Data
AR0001	Vehicle Counting but with adjusted confidence threshold and frame disappearance threshold
AR0002	View Dashboard after confidence and frame disappearance threshold adjusted
AR0003	Vehicle Counting and Classification with a YOLO v5 model with a lower weight (Yolov5n)
AR0004	View Dashboard after model weight adjustment

5.3 Summary of results

As previously stated, detailed test reports can be found in Appendix C. To summarize those results, the black box testing went successfully. No changes were made to the original program's code and it ran smoothly. It was easy to capture the number of vehicles in the sample CCTV footage, and the dashboard represented the data without any issues.

However it was a different case when Regression testing was performed. The confidence threshold that was initially set at 0.6. This was determined to be the most ideal threshold as it yielded better results overall compared to lower or higher thresholds. The threshold in this test was set to 0.2. Additionally the disappearance threshold, which refers to the number of consecutive frames a vehicle should be missing to be considered disappeared, was also adjusted. Initially, it was 10 frames, but was set to 1. The result was the counter overshooting and counting more cars than their actually were.

A similar issue was also observed when the model weight was adjusted to a lower weight. With their being lesser layers in the model, the frame rate was higher, at the expense of its accuracy as the counter overshot and counted more vehicles than there were.

5.4. Evidence of functionality

Based on the overall test results, it can be concluded that the system is operational and does meet the requirements, however it may require specific adjustments due to different factors that may arise. The model weight, the threshold of confidence and frame disappearance need to be taken into account.

5.5. Critical Evaluation

5.5.1. Advantages

- Upon refreshing, the dashboard automatically adjusts itself to new data
- The dashboard offers many features, including aesthetic ones where the user can toggle between dark mode and light mode
- The web app is secure as only people with accounts can see the data

5.5.2. Disadvantages

- The results aren’t always accurate, and adjustments to the confidence and disappearance threshold need to be made

5.5.3. Strengths

- Simple and easy to use
- Does not require manual data entry/upload
- It can adjust itself to the size of the users screen, so it looks appealing in Tabs and Smartphones as well

5.5.4. Limitation

- An active internet connection is required by the user in order to access and view the dashboard

- The model weight as well as other adjustments need to be made in order to boost accuracy of the results

6 CONCLUSION & FUTURE WORK

As previously stated, there is an impending issue of traffic congestion, increasing the risk of vehicular accidents. The increase in the number of vehicles may prove to be a security issue as it increasingly becomes more and more difficult to keep track of things. A system is needed to help keep the general public, or anyone interested informed on the latest developments in traffic. The aim of the Vehicle Counting and Classification is that.

A comprehensive study was conducted to better understand about the various existing systems, methodologies and algorithms used for Vehicle Counting and Classification. Upon research, while they were done in unique ways, the proposed system is far superior in that it works like its predecessors, but expands on itself by also being a web dashboard that’s interactive. The system managed to achieve its goals, successfully fulfilling all 3 modules that were planned, and right on time. It showed that it can detect, count, and classify vehicles in CCTV footage of all types. It stores them all in a CSV file where, it is fetched by an API and is presented on a web dashboard.

Despite that, there’s always room for improvement. The system worked at its best after having to stabilize the bounding boxes, defining a confidence threshold as well as a disappearing frame threshold as well. In the future, the system will be able to operate more efficiently, by attempting to work with more advanced models such as YOLO v8. Another improvement that the system will have in the future is allowing users to upload CCTV footage directly through the system. Centralizing everything instead of having to do them separately.

In conclusion, the Vehicle Counting and Classification System has successfully managed to achieve the goals that were defined earlier, with a very bright future ahead of it as well.

REFERENCES

[1] Memon, S. *et al.* (2018) "A video based vehicle detection, counting and classification system," *International Journal of Image, Graphics and Signal Processing*, 10(9), pp. 34–41. Available at: https://doi.org/10.5815/ijigsp.2018.09.05.

[2] Unzueta, L. *et al.* (2012) "Adaptive multicue background subtraction for robust vehicle counting and classification," *IEEE Transactions on Intelligent Transportation Systems*, 13(2), pp. 527–540. Available at: https://doi.org/10.1109/tits.2011.2174358.

[3] Kamkar, S. and Safabakhsh, R. (2016) "Vehicle detection, counting and classification in various conditions," *IET Intelligent Transport Systems*, 10(6), pp. 406–413. Available at: https://doi.org/10.1049/iet-its.2015.0157.

[4] Dai, Z. *et al.* (2019) "Video-based Vehicle Counting Framework," *IEEE Access*, 7, pp. 64460–64470. Available at: https://doi.org/10.1109/access.2019.2914254.

[5] Dhaduk, H. (2022) *An Ultimate Guide of Web Application Architecture*, *Simform*. Available at: https://www.simform.com/blog/web-application-architecture/

APPENDICES

Appendix A – Use Case Descriptions appearing in the order they are listed in

Use Case Name	Select appropriate YOLO v5 model	
Scenario	User would like to use the system	
Triggering Event	User would like to detect and count vehicles	
Brief Description	User would like to detect, count, and classify data obtained from CCTV footage	
Actors	User	
Related Use Case	N/A	
Stakeholders	Users/General Public	
Preconditions:	N/A	
Postcondition	N/A	
Flow of activities	User	System
	1. User starts the system 2. User selects appropriate model	1. System accepts the chosen model 2. System tells Yolov5 to load the model

Exception Conditions	1. Typo where he mistook one model for another 2.

Use Case Name	Input
Scenario	Users can input the video into the system
Triggering Event	When user has the system open
Brief Description	User wants to input CCTV footage for the model to catch
Actors	User
Related Use Case	N/A
Stakeholders	Users/General Public
Preconditions:	N/A
Postcondition	Video is successfully input
Flow of activities	1. User puts the file path to the video into the program
Exception Conditions	User input the wrong file path

Use Case Name	Input
Scenario	Users can input the video into the system
Triggering Event	When user has the system open
Brief Description	User wants to input CCTV footage for the model to catch them
Actors	N/A
Related Use Case	N/A
Stakeholders	Users/General Public
Precondtions:	N/A
Postcondition	Video is successfully input
Flow of activities	1. User puts the file path to the video into the program
Exception Conditions	Video is corrupted

Use Case Name	View Generated Report	
Scenario	User would like to get a general overview of traffic	
Triggering Event	User opens web dashboard	
Brief Description	The user just wants to view the generated visual statistics	
Actors	User	
Related Use Case	Analyze video feed	
Stakeholders	User	
Precondtions:	N/A	
Postcondition	N/A	
Flow of activities	Actor	System
	User access the data results via the web dashboard	System generates the report via the web dashboard based on the information processed in the database
Exception Conditions	No exception case	

Use Case Name	Log in	
Scenario	User would like to log into the application	
Triggering Event	User signing into the system	
Brief Description	User will be using their registered account to log into the application	
Actors	User	
Related Use Case	View Generated Reports	
Stakeholders	Users/General Public	
Precondtions:	Application requires the user to have internet connection for validation. User account must be pre-registered in the system	
Postcondition	Log into application successfully	
Flow of activities	Actor	System
	1. User inputs the valid information and clicks on the log in button 2. User is successfully	1. System validates the account information

	able to login into their account	
Exception Conditions	1. Invalid Input 2. No internet connection	

Use Case Name	View data
Scenario	User can go on the data page and view data from excel sheet
Triggering Event	When user is on the data page
Brief Description	User wants to visit the data page and view data from excel sheet
Actors	User
Related Use Case	N/A
Stakeholders	Users/General Public
Precondtions:	N/A
Postcondition	Page successfully loaded

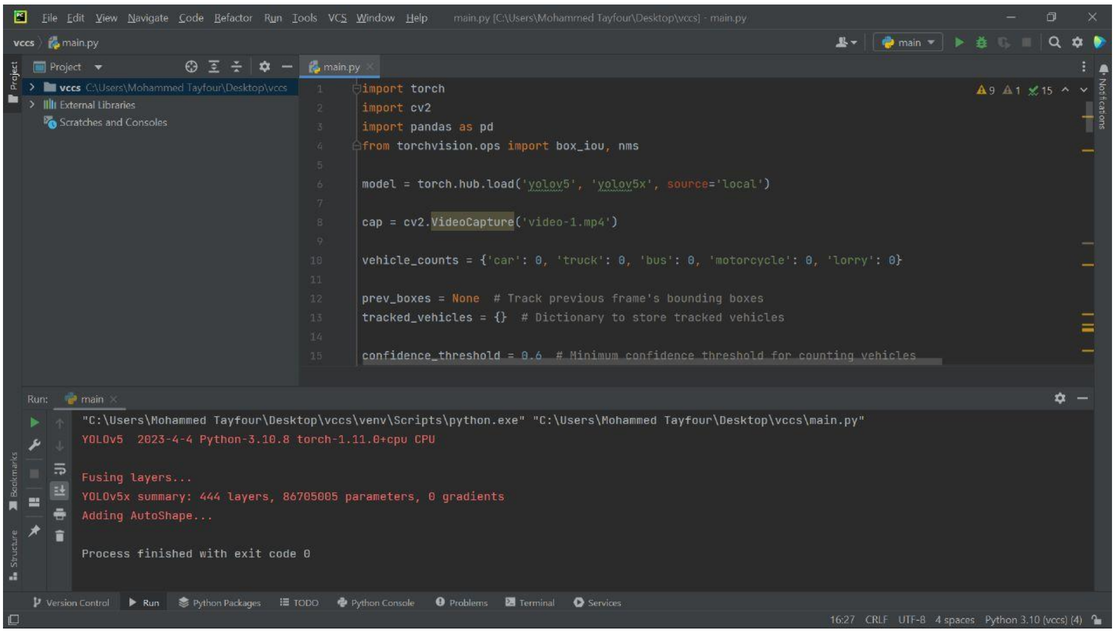
Flow of activities	1. User clicks on choose file 2. User selects file
Exception Conditions	Page is corrupted

#7

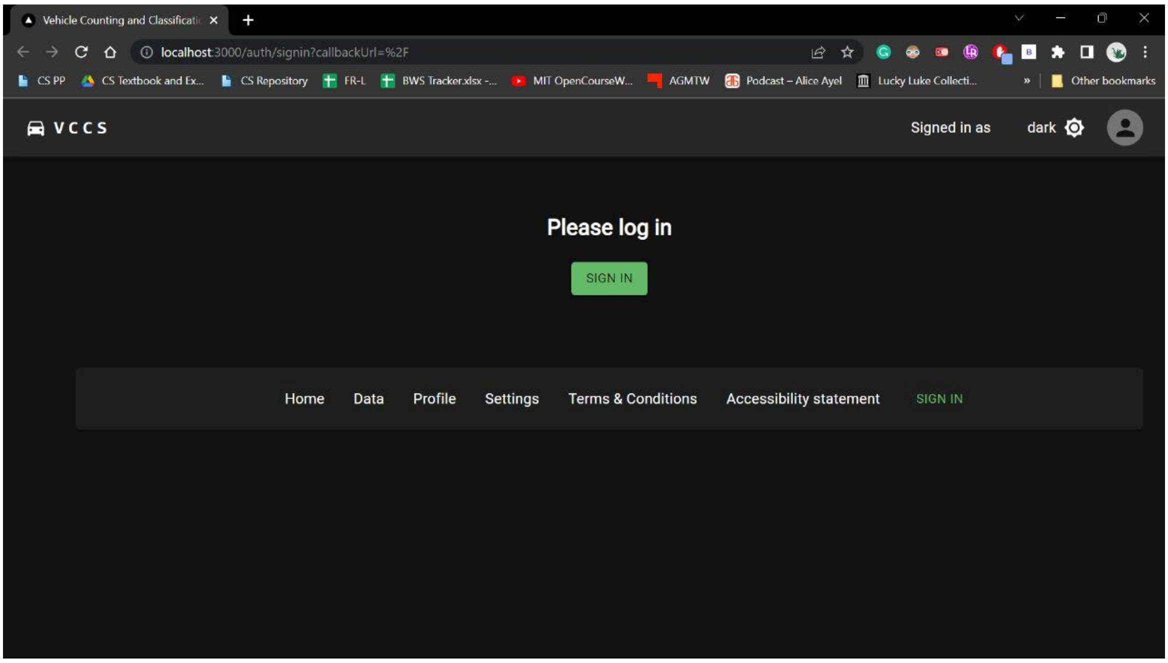
Use Case Name	Edit profile
Scenario	User would like to edit profile details
Triggering Event	When user is on the profile section
Brief Description	User wants to visit the profile page to change their details
Actors	User
Related Use Case	N/A
Stakeholders	Users/General Public
Precondtions:	N/A
Postcondition	Page successfully loaded

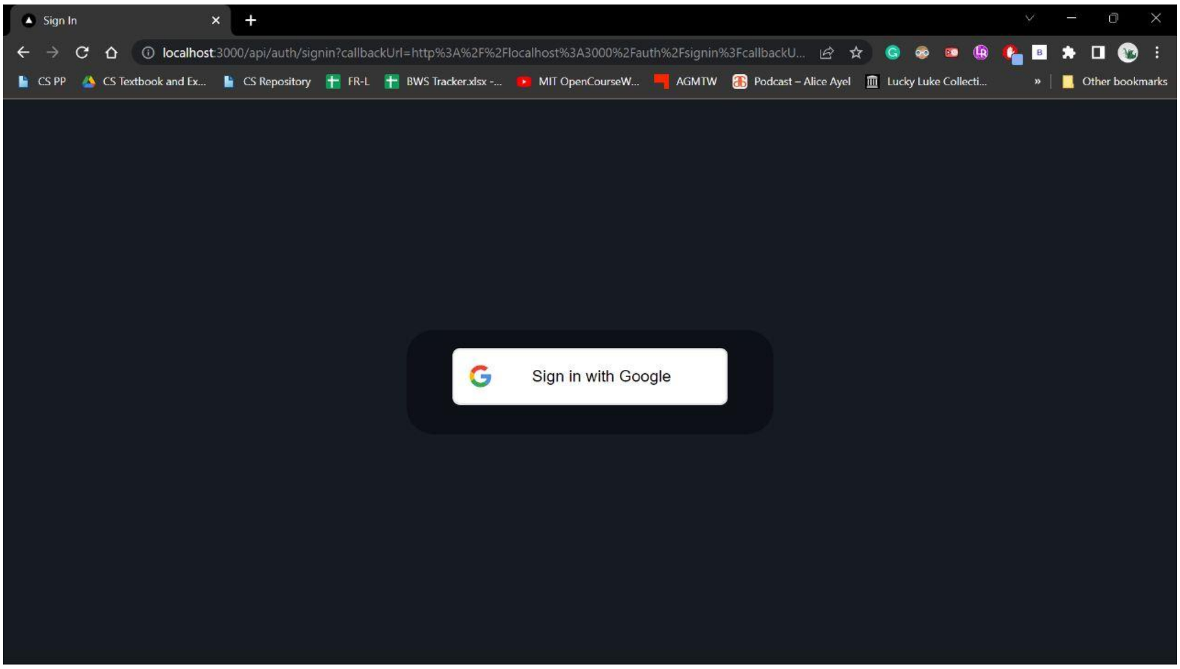
Flow of activities	1. User clicks on box 2. User fills out the box with new information 3. User clicks Save Changes
Exception Conditions	Page is corrupted

Appendix B – User Interface

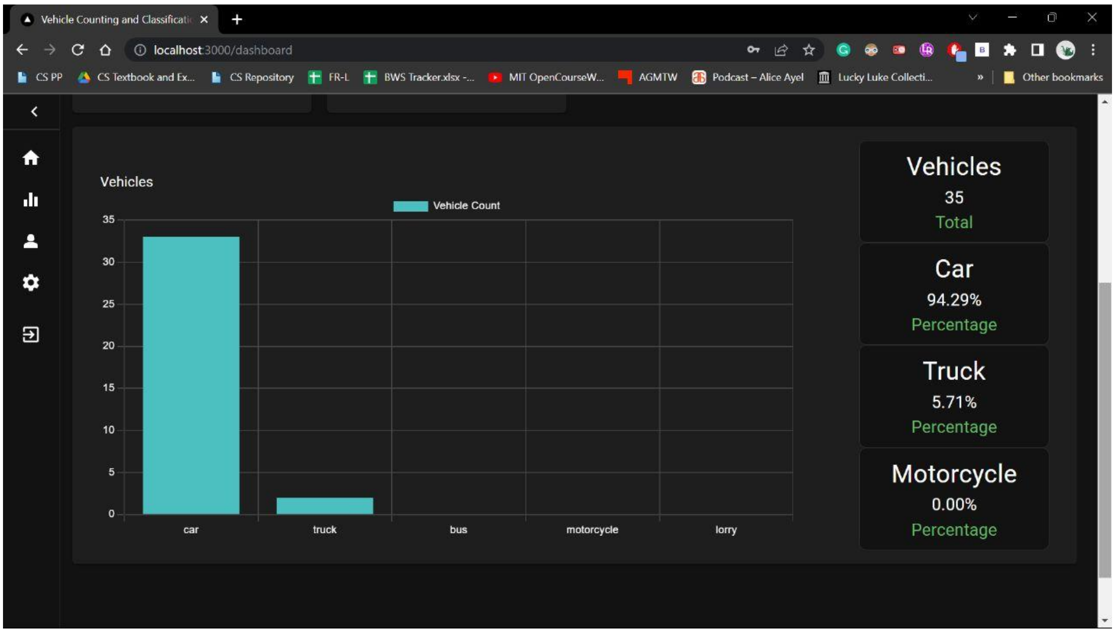


#1 Counting and Classification interface

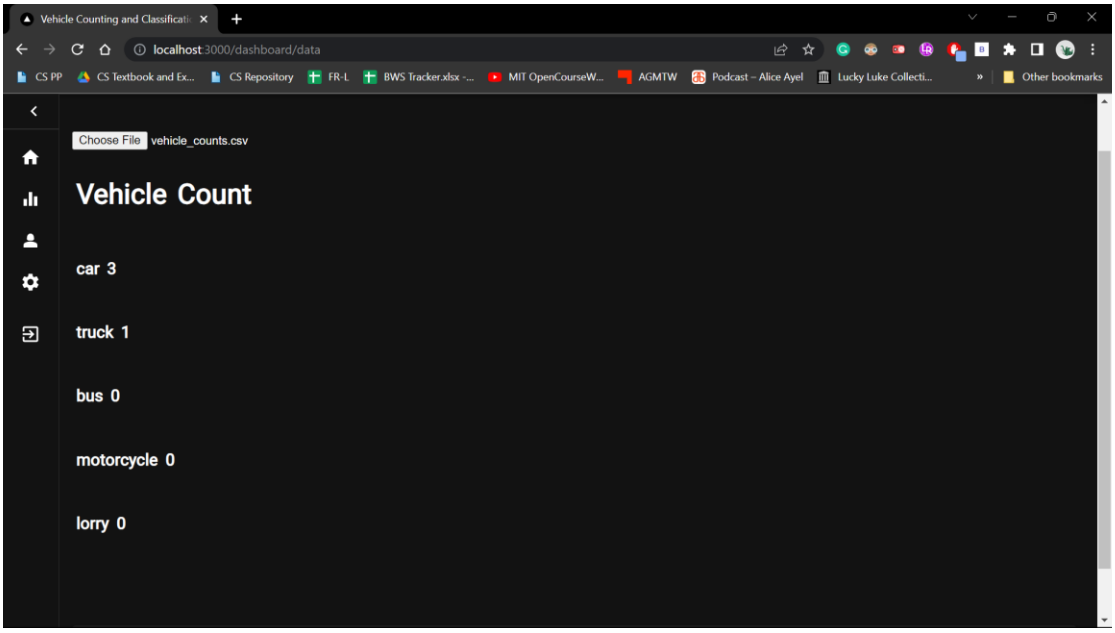




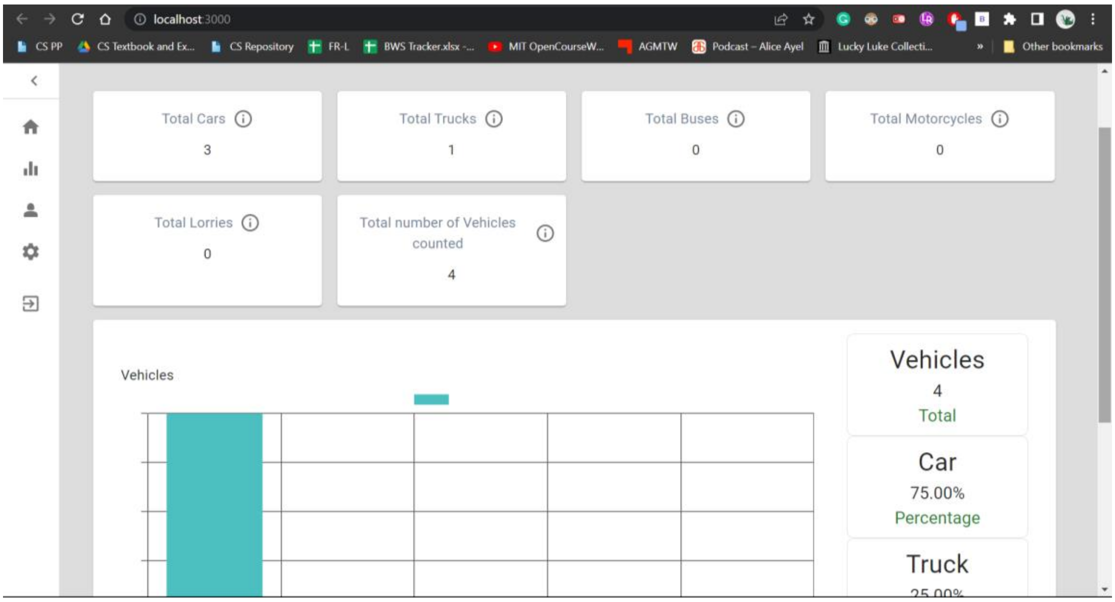
#2 Web dashboard log in process



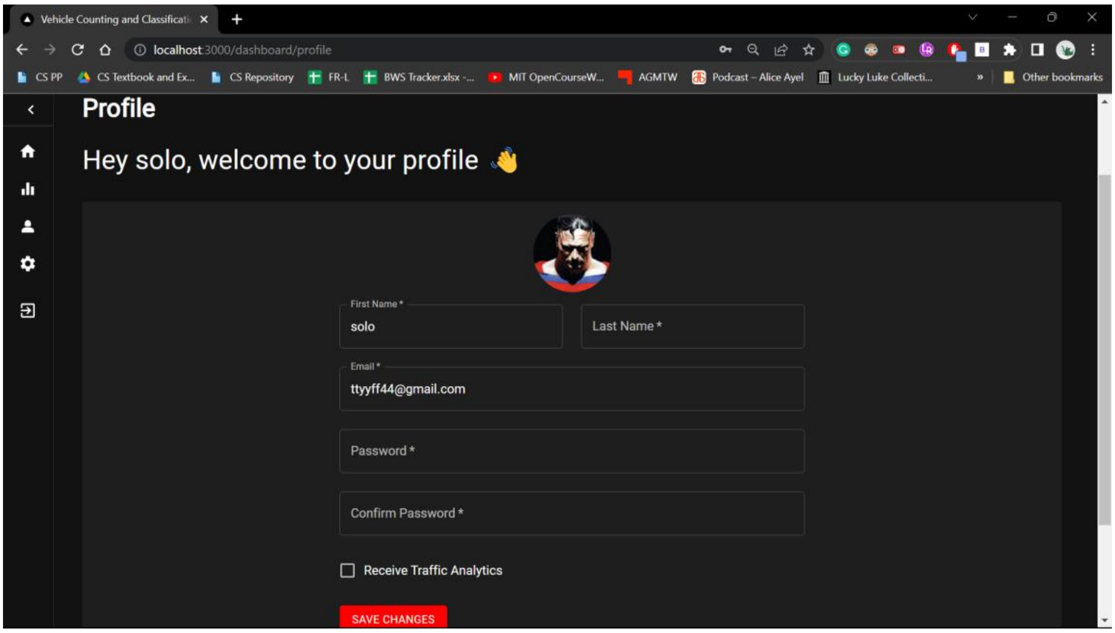
Dashboard



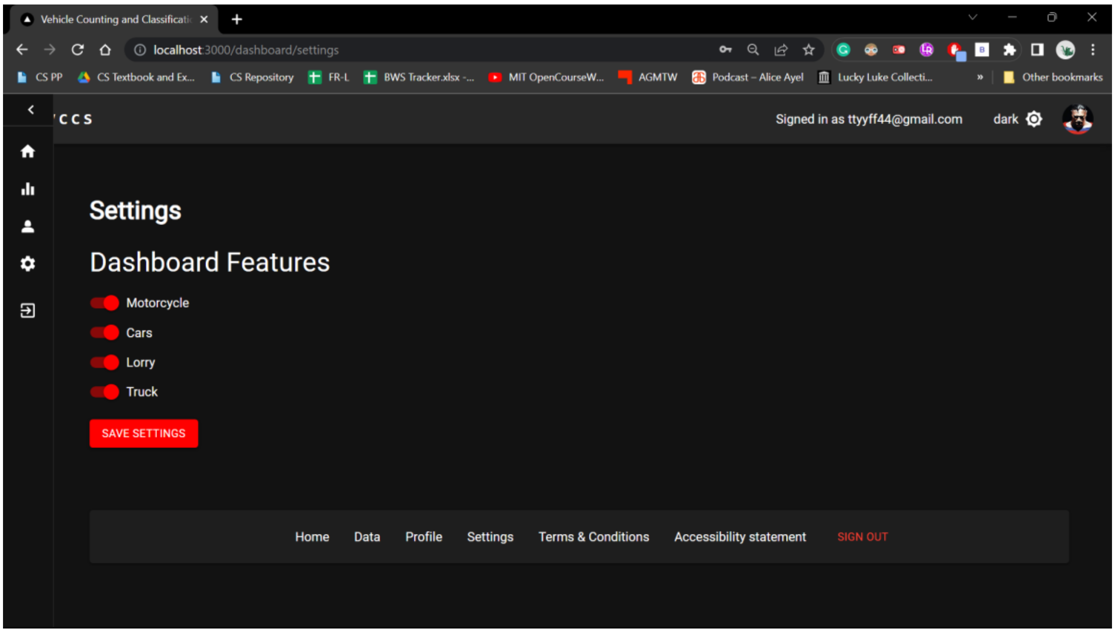
Data page for user to upload CSV file



Dashboard with light mode toggled showing data ribbons



Profile page



Settings page

Appendix C – Test Cases

By order in which they were mentioned:

Test Case ID	AB001				
Test priority	High				
Test Case Name	Vehicle Counting and Classification				
Description	Press run				
Preconditions	-				
Steps	**Steps involved**	**Input Data**	**Expected Result**	**Actual Result**	**Pass/Fail**
1	Press run	Video-1.mp4	Accurate counting and classification	Accurate counting and classification	Pass

Test Case ID	AB002				
Test priority	High				
Test Case Name	Log in				
Description	Enter valid email and password				
Preconditions	-				
Steps	**Steps involved**	**Input Data**	**Expected Result**	**Actual Result**	**Pass fail**

Test Case ID	AB003				
Test priority	High				
Test Case Name	View Dashboard				
Description	Click Home Button				
Preconditions	Logged in				
Steps	**Steps involved**	**Input data**	**Expected Result**	**Actual Result**	**Pass/Fail**
1	User clicks home button	Vehiclecounts.csv	Vehicle counts presented	Vehicle count presented	Pass

Test Case ID	AB004				
Test priority	Moderate				
Test Case Name	View Raw Data				
Description	Click Data button to go to data page				
Preconditions	Logged in				
Steps	**Steps involved**	**Input Data**	**Expected Result**	**Actual Result**	**Pass/Fail**
1	User clicks choose file	Any .csv file	Raw data uploaded	Raw data uploaded	Pass

	and uploads a csv file		and visualized	and visualized	

Test Case ID	AR001				
Test priority	High				
Test Case Name	Vehicle Counting but with adjusted confidence threshold and frame disappearance threshold				
Description	User changes confidence threshold to 0.2 and frame disappearance threshold to 1				
Preconditions	-				
Steps	**Steps involved**	**Input Data**	**Expected Result**	**Actual Result**	**Pass/Fail**
1	User clicks run after changing the threshold values	Video-1.mp4	System counts and classifies accurately	System counts and classifies inaccurately	Fail

Test Case ID	AR002
Test priority	High
Test Case Name	View Dashboard after confidence and frame disappearance threshold adjusted
Description	User clicks Dashboard

Preconditions	Logged in				
Steps	**Steps involved**	**Input Data**	**Expected Result**	**Actual Result**	**Pass/Fail**
1	User clicks on home button	Vehiclecounts.csv	Dashboard would present data, but inaccurately	Dashboard presented data inaccurately	Pass

Test Case ID	AR003				
Test priority	High				
Test Case Name	Vehicle Counting and Classification with a YOLO v5 model with a lower weight				
Description	User clicks run after changing the models weight from 'x' to 'n'				
Preconditions	-				
Steps	**Steps involved**	**Input Data**	**Expected Result**	**Actual Result**	**Pass/Fail**
1	User clicks run after changing model weight	Video-1.mp4	Inaccurate results due to lesser layers	Inaccurate results due to lesser layers	Fail

Test Case ID	AR004				
Test priority	High				
Test Case Name	View Dashboard after model weight adjustment				
Description	User inspects dashboard after model weight adjustment				
Preconditions	Logged in				
Steps	**Steps involved**	**Input Data**	**Expected Result**	**Actual Result**	**Pass/Fail**
1	User clicks home button	-	Dashboard presents the inaccurate data	Dashboard presents the inaccurate data	Pass

Appendix D – Sequence Diagram