TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING

PASCHIMANCHAL CAMPUS

A Major Project Mid-Term Progress Report On

Smart Traffic Lights

Submitted by:

Aashish Shrestha [PAS076BCT002]

Ritika Bhandari [PAS076BCT024]

Sameen Kunwar [PAS076BCT031]

Shirshak Tiwari [PAS076BCT039]

Submitted to:

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING

PASCHIMANCHAL CAMPUS

LAMACHAUR, POKHARA

MANGSIR, 2080

ACKNOWLEDGEMENT

Project would not have been feasible without the generous support and assistance of many people and organizations. It brings us great pleasure to share the report of the BCT Project (Part A) included in the syllabus. We would like to express our heartfelt gratitude to each and every one of them. We are really obliged to our supervisor Er. Rupesh Kumar Sah for his direction and regular monitoring as well as for giving vital information regarding the project & also for his cooperation in bringing the project to its current state. We also want to thank our friends for their support and encouragement in bringing this project to its current state. Last but not the least, We would like to thank everyone who is involved in the project directly or indirectly.

Sincerely,
Aashish Shrestha
Ritika Bhandari
Sameen Kunwar
Shirshak Tiwari

ABSTRACT

Traffic congestion is becoming one of the critical issues with increasing population and automobiles in cities. Traffic jams not only cause extra delay and stress for the drivers but also increase fuel consumption and air pollution. Although it seems to pervade everywhere, busy cities are the ones most affected by it. And its ever-increasing nature makes it necessary to calculate the road traffic density in real-time for better signal control and effective traffic management. The traffic controller is one of the critical factors affecting traffic flow. Therefore, the need for optimizing traffic control to better accommodate this increasing demand arises. Our proposed system aims to utilize live images from the cameras at traffic junctions for traffic density calculation following YOLO(You Only Look Once)v8:Real -Time Object Detection . The project also aims on the algorithm for switching the traffic lights based on the vehicle density to reduce congestion, thereby providing faster transit to people and reducing pollution

Keywords—Traffic control, Traffic light system, Traffic Junctions, Object detection, YOLO

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

1.1 Background

In the evolving urban landscape, the constant increase in automobile traffic has become an essential feature of urban life. Traditional traffic light systems that rely on fixed schedules and rudimentary timers are increasingly proving inadequate to meet the needs of modern urbanization. As cities grow, the complexity of traffic patterns also increases, leading to congestion, delays and increased commuter frustration. Addressing these challenges requires a paradigm shift in traffic management systems, requiring the adoption of cutting-edge technologies to pave the way for a new era of smart solutions and Adaptation.
The origin of the project “Smart Control Traffic Signals” lies in the awareness of the limitations of conventional traffic light systems. Traditional models lack the ability to adapt to fluctuating traffic conditions, leading to inefficiency, wasted time and increased vehicle emissions. The booming technology landscape, especially in the fields of computer vision and artificial intelligence, offers unprecedented opportunities to reinvent and improve traffic control mechanisms. Central to this project is the use of YOLO (You Only Look Once), an advanced object detection system known for its speed and accuracy. YOLO allows the system to instantly identify and track vehicles near intersections. This break from traditional methods that often struggle with real-time detection has put the project at the forefront of innovation in traffic management. The increasing availability and affordability of high-performance hardware, combined with the open source nature of YOLO, contribute to the project’s technical viability.
In addition, the project also recognizes the multidimensional nature of traffic congestion. By integrating advanced algorithms to estimate traffic intensity, the system not only aims to quantify traffic flow but also differentiate the impact of different vehicle types on traffic congestion dynamics. In addition to the immediate benefits of reducing congestion and improving traffic flow, the project also fits into broader goals such as environmental sustainability and economic efficiency. Implementing smart traffic light control systems has the potential to minimize fuel consumption, reduce carbon emissions and optimize the use of road infrastructure.

1.2 Problem Statement

In the changing urban landscape, the continuous increase in automobile traffic has become an essential feature of urban life. Traditional traffic light systems that rely on fixed schedules and rudimentary timers are increasingly proving unsuitable for the needs of modern urbanization. As cities grow, the complexity of traffic patterns increases, leading to congestion, delays and increased commuter frustration. Addressing these challenges requires a paradigm shift in traffic management systems, requiring the adoption of advanced technologies to usher in a new era of intelligent solutions and adaptability .

1.3 Objectives

The main objective of the ”Smart Control Traffic Signal” project is:
• To develop an intelligent traffic control system using YOLO for real-time vehicle detection, estimate traffic intensity, and adjust traffic signals for optimal traffic flow in urban intersections.

1.4 Feasibility Analysis

The feasibility of the project “Smart Control Traffic Signals” is evaluated on the basis of technical, economic and operational aspects.

1.4.1 Technical Feasibility

The project relies on proven technologies such as YOLO for object detection, making it technically feasible. Additionally, the availability of affordable open source hardware and software improves technical feasibility.

1.4.2 Economic Feasibility

Implementation of this project is economically viable because it leverages existing technologies and aims to improve traffic efficiency, potentially reducing costs maintenance and fuel in the long run.

1.4.3 Operational Feasibility

The project is operationally feasible because it focuses on improving traffic signal control efficiency without disrupting existing infrastructure.

1.5 Scope of Project

The scope of the "Smart Control Traffic Signal" project includes urban and suburban intersections with high traffic density. The system will be designed to adapt to different road configurations and accommodate different vehicle types. Additionally, the project scope includes scalability for potential integration with existing traffic management systems in the future.

CHAPTER 2: LITERATURE REVIEW

Numerous studies in the literature have explored the application of computer vision for smart traffic management. The literature review for this project involves researching and analyzing existing studies, articles, and publications related to using deep learning for intelligent traffic lights.

Popular tools and technologies used in our project include OpenCV[1], YOLOv8[2], Weights and Biases[3], Supervision[4], Fast API[5],and Uvicorn[6].

The following papers and articles were referred for research for this project:

Image Processing Based Smart Traffic Control System for Smart City
In this paper, a real-time traffic management system to control the traffic flow by calculating percentage match using Canny Edge detection technique, which is an image processing method, along with a digital image processing tool in MATLAB is proposed.[7].

AI Based Smart Traffic Management
In this article, a smart traffic management system utilizing the Internet of Things (IoT) is presented to address a variety of challenges for managing traffic on roadways and to aid authorities in efficient planning .[8].

Real-Time Smart Traffic Control and Simulation: An Approach for Urban Congestion Management
In this article , the authors present a review of traffic congestion management in smart cities and propose a novel, lightweight simulator for visualizing real-time traffic and an optimisation mechanism based on realtime congestion data gathered from the simulator is also proposed. [9].

Smart traffic lights switching and traffic density calculation using video processing
The method to use live video feed from the cameras at traffic junctions for real time traffic density calculation using video and image processing is presented, thereby aiming at reducing the traffic congestion on roads which will help lower the number of accidents.[10].

. .

CHAPTER 3: METHODOLOGY

The methodology for implementing the project ”Smart Control Traffic Signal” involves several key steps, including data collection, model training, and deployment. These steps are essential to ensure the successful development and implementation of a smart traffic signal system.

3.1 Data Collection

The first step is to collect the necessary datasets for training the vehicle detection model. This involves gathering existing vehicle detection datasets from sources like Roboflow Universe, which provides labeled image datasets of different vehicles. Additionally, custom datasets will be created by capturing images of vehicles. It is important to ensure that the datasets are diverse, containing samples representing various vehicle categories, lighting conditions, and orientations. This will help the model generalize effectively and accurately detect different types of vehicles. The images are taken by CCTV, which is a common source for capturing vehicle data for detection and monitoring purposes.

3.2 Data Pre-Processing

Although the dataset was pre-labeled, some data pre-processing was necessary. The dataset was divided into training, validation, and test sets. The images were annotated for classification into various classes such as car, motorbike, bus, and truck. Data augmentation was performed to enhance the diversity of the available dataset, ensuring an adequate number of samples. All of these tasks were carried out on the Roboflow website, which is intuitive and easy to use.

3.3 Model Training

The next crucial step is to train a model for vehicle detection and classification using the gathered and labeled datasets. YOLOv8 (You Only Look Once version 8), a sophisticated model renowned for its real-time object detection capabilities, will be employed. The training process will involve optimizing the model’s parameters through iterations to enhance its accuracy in swiftly identifying and classifying different vehicles on the road. This adaptation aligns seamlessly with our overarching goal of developing a smart traffic signal control system powered by YOLOv8 for efficient and real-time vehicle detection. Various versions of pre-trained YOLOv8 models were tested to find the balance between speed and accuracy.

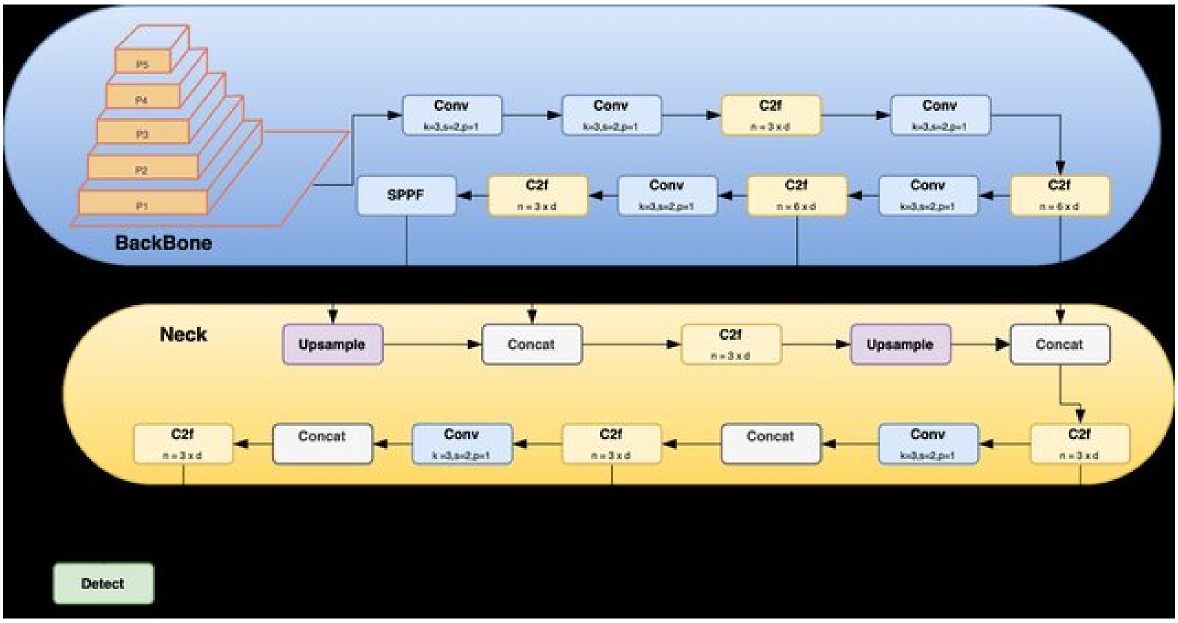

Figure 1: YOLOv8 Architecture

3.4 API Development

A simple web application has been developed, employing frameworks such as FastAPI and Uvicorn. This web app is designed with the capability to receive video input and execute predictions using pre-trained models. The use of FastAPI and Uvicorn contributes to the efficiency and responsiveness of the web application, ensuring a seamless user experience. The application serves as an accessible interface for users to submit video data, allowing the underlying trained models to analyze and generate predictions based on the provided inputs.

3.5 Hardware Integration

Simultaneously, the trained model can be integrated with existing CCTV cameras and traffic lights. A simulation demonstrating the implementation of the system will be created in the next phase of this project. We will conduct tests on low-end devices to determine the effectiveness and speed of the trained model. The real-time feed from the CCTV cameras will be input into the model to detect, classify, and count the number of vehicles in each direction of the junction. The count of vehicles in each direction of the junction will determine the timings of traffic signals. Priority will be given to the direction with a higher vehicle count, and more time will be assigned to that direction. This is how we can seamlessly integrate existing hardware with the capabilities of machine learning to optimize the flow of traffic.

The figure explains how the system works.

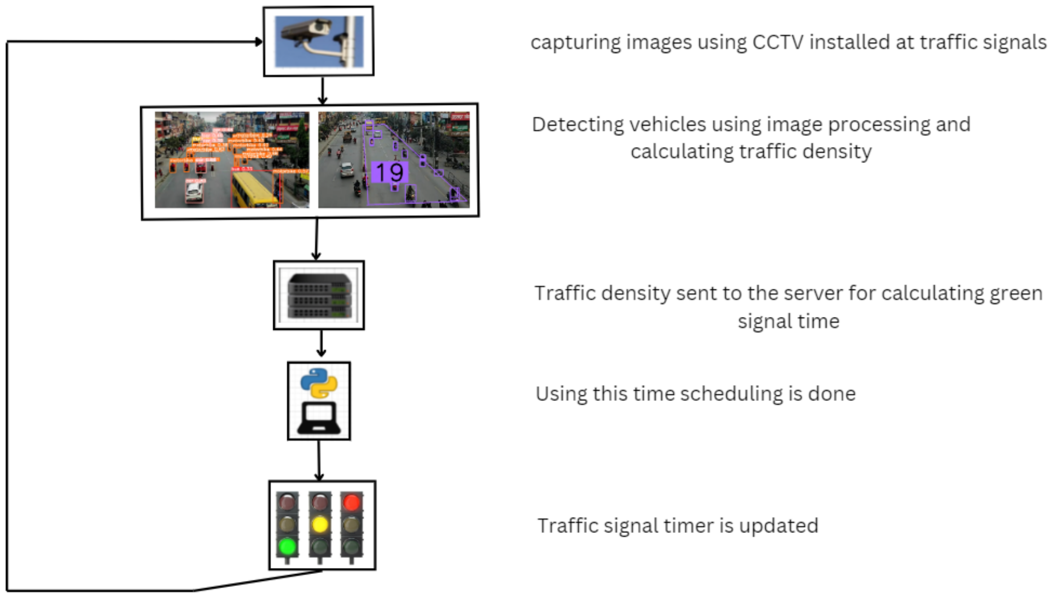

Figure 2: Block diagram

CHAPTER 4: TOOLS AND TECHONOLOGIES

4.1 Software

4.1.1 Python

Python is a popular programming language used for developing various applications, including machine learning and computer vision. It serves as the primary programming language for implementing the waste classification and sorting system. Python provides a wide range of libraries and frameworks that facilitate image processing, deep learning, and communication with hardware components.

4.1.2 TensorFlow

TensorFlow is an open-source machine learning framework widely used for building and training deep learning models. It provides a comprehensive set of tools and functionalities for creating and deploying neural networks. In this project, TensorFlow is utilized for training and implementing the YOLO (You Only Look Once) model.

4.1.3 OpenCV

OpenCV is a library that provides various computer vision functions and algorithms. It offers a wide range of tools for image processing, feature detection, and object tracking. In this project, OpenCV is used for calculating the coordinates of the bounding box from a predicted image.

4.1.4 YOLO

YOLO is a Convolutional Neural Network (CNN) for doing object detection in real-time. The algorithm applies a single neural network to the full image, and then divides the image into regions and predicts bounding boxes and probabilities for each region. It is widely used for applications such as object recognition, autonomous driving, and surveillance systems. In this project, YOLO is employed for waste classification, allowing the system to identify and categorize different types of waste items, such as paper, plastic, glass, metal, and biodegradable waste. Different versions of YOLO are available. We will be using YOLO v8 which is the latest version in our project.

4.1.5 Weights and Biases

Weights and Biases is a platform designed for streamlined machine learning model training and optimization. It offers tools for experiment tracking, visualization, and collaboration. Users can log and compare model performance, analyze training processes, and collaborate seamlessly. The platform is adaptable, supporting integration with popular machine learning frameworks. Weights and Biases enhances productivity throughout the model development lifecycle.

4.1.6 Roboflow

Roboflow is a user-friendly platform for simplifying computer vision projects. It provides tools for image annotation, dataset versioning, and data augmentation. With an intuitive interface, Roboflow supports collaboration and accommodates users of varying expertise levels. The platform is designed to streamline dataset management and preprocessing tasks, making it a valuable resource for researchers, developers, and data scientists working on computer vision projects.

4.1.7 Supervision

Supervision is an open-source Python package developed by Roboflow, that provides utilities for building computer vision applications. It offers tools for common tasks such as loading datasets, drawing detections and annotations, converting labels, and counting detections in zones. By using Supervision, developers can save time and focus on more critical aspects of their projects. The package supports various computer vision tasks, including object detection, image classification, and semantic segmentation. It can be installed with pip in a Python environment and is distributed under an MIT license, allowing free use in both public and private projects.

4.1.8 Fast API

FastAPI is a high-performance web framework for building APIs with Python 3.7 or higher. It is fully compatible with well-known API standards such as OpenAPI and JSON Schema, and it offers significant speed for development while being completely production-ready. FastAPI is particularly useful for creating RESTful APIs and web applications, and it automatically generates interactive API documentation using the OpenAPI standard, simplifying testing and understanding the API.

4.1.9 Uvicorn

Uvicorn is a fast and efficient ASGI-compatible web server commonly used with Python web frameworks like FastAPI. It implements ASGI standards, provides support for HTTP/2 and WebSockets, and offers high-performance, concurrent support for web applications. When used with FastAPI, Uvicorn serves as the development server, listening on a socket, receiving incoming connections, and then handing the requests over to FastAPI according to the ASGI interface. This combination allows for the creation of high-speed, well-documented APIs, making it suitable for various web applications and deployments.

CHAPTER 5: EPILOGUE

5.1 Work Progress

After the model completed training, it achieved a mAP50 score of 0.895. The metrics for evaluating the model are shown below:

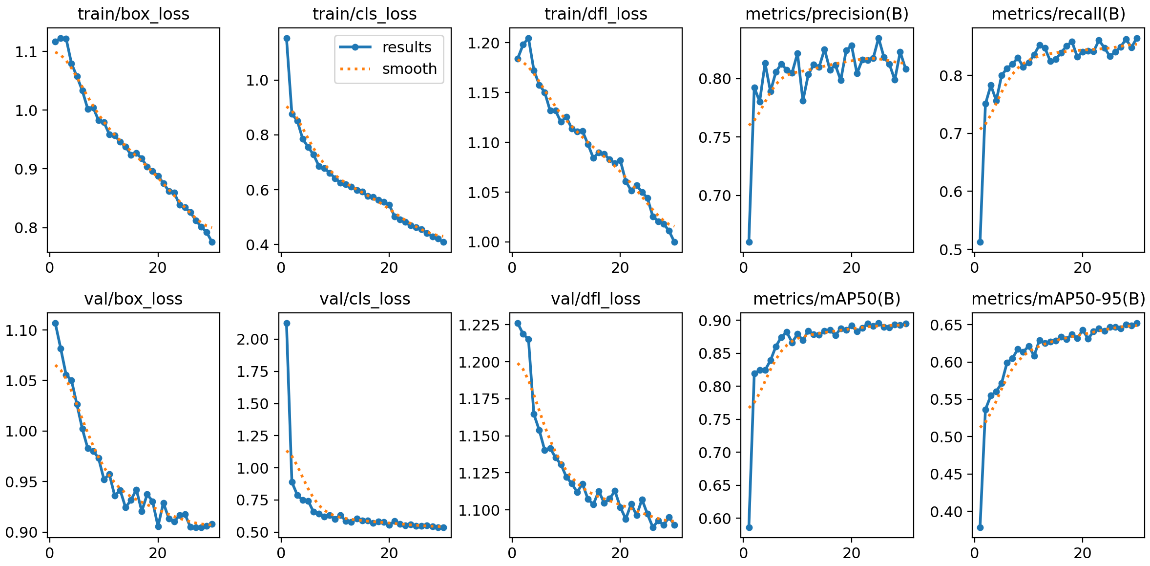

Figure 3: Metrics for model evaluation

The trained model was used to predict and classify vehicles in an image.

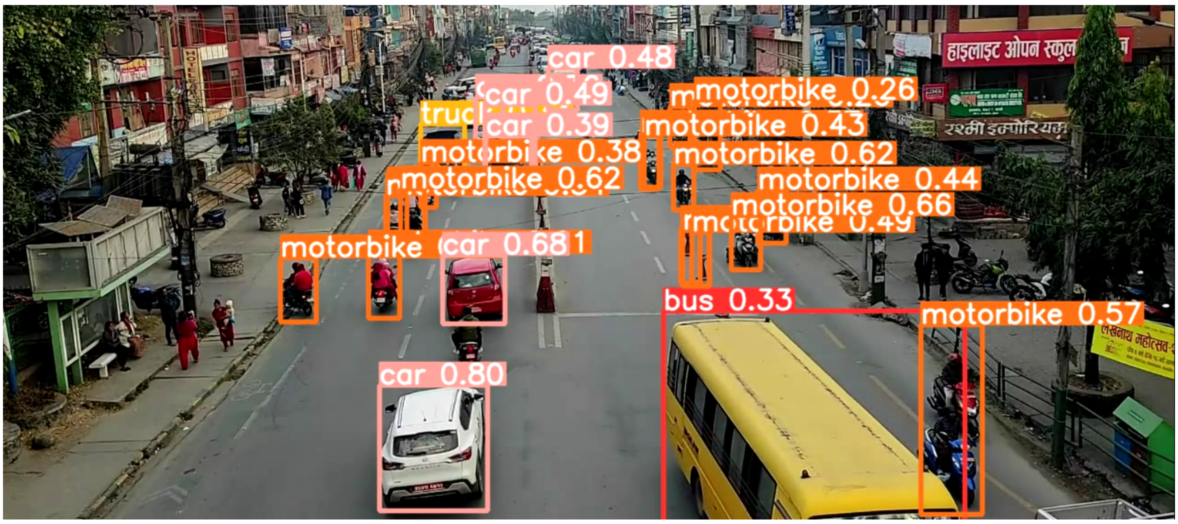

Figure 4: Predicted Result

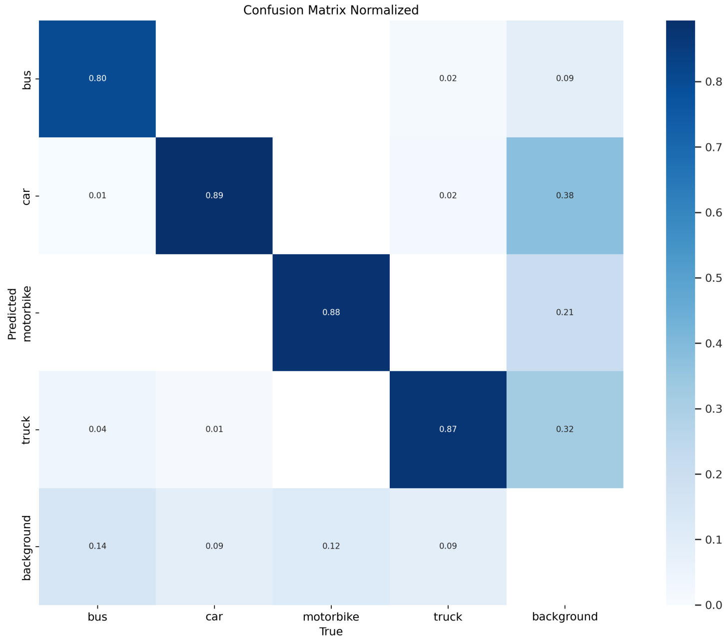

Figure 5: Normalized Confusion Matrix

The model was used to predict the number of vehicles present inside a zone. The zone creation and counting was done with the help of Supervision.

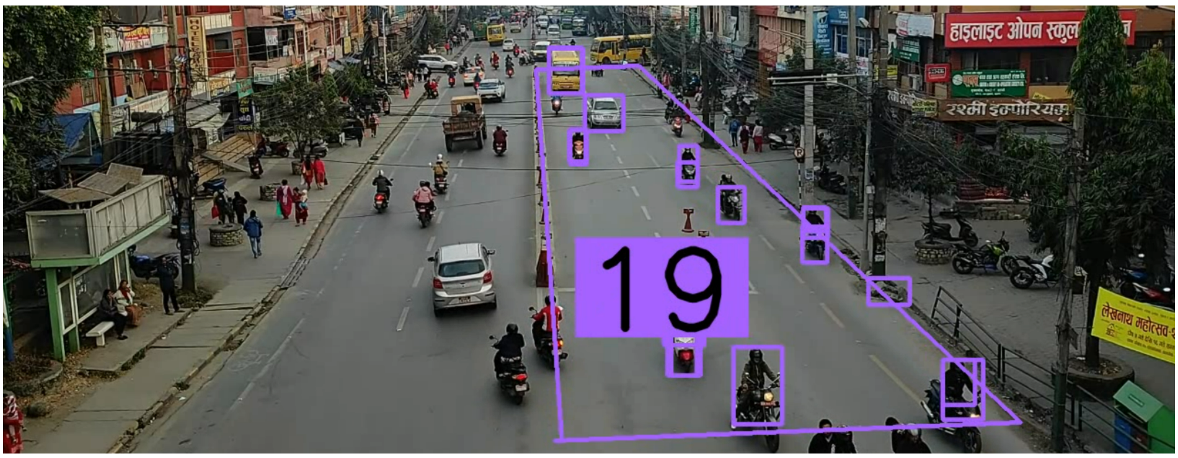

Figure 6: Counting objects in a zone

5.2 Work Schedule

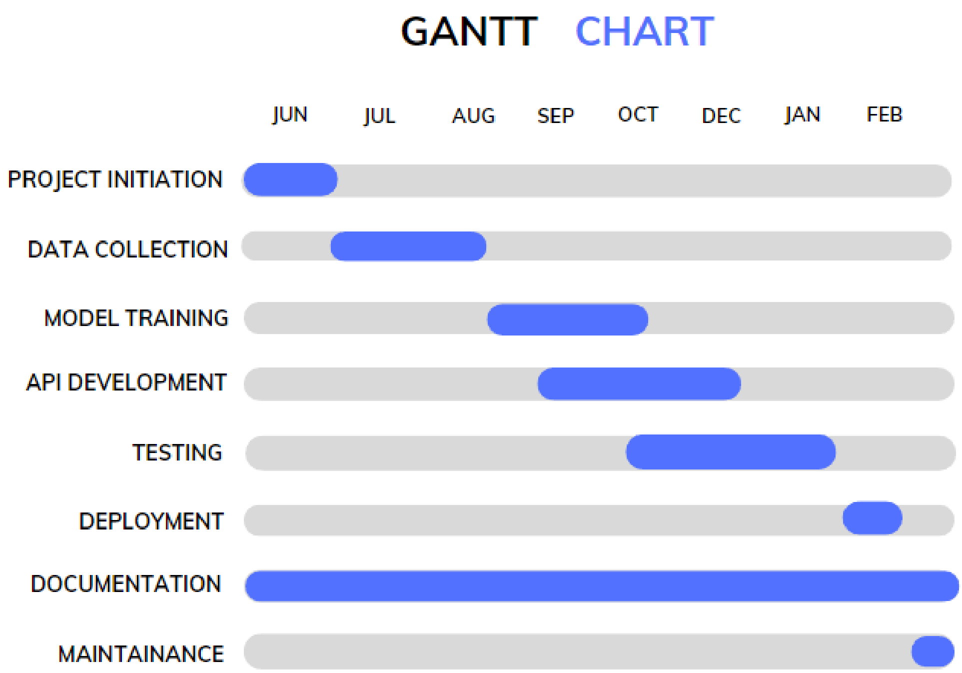

Figure 7: Gantt Chart

This means that the entire project should take approximately 8 months to complete. However, the exact duration will depend on the complexity of the system and the availability of resources.

5.3 Remaining Tasks

Approximately half of our work has been completed, yet some tasks and optimizations still need to be done.

5.3.1 Hyperparameter Tuning

We have previously conducted experiments with varying the number of epochs. However, to further enhance the model's performance, we intend to explore additional hyperparameters such as batch size, learning rate, and the number of layers during the model training process. Most of the current parameters in the model are set to default values, and adjusting these factors will allow for a more thorough optimization of the model.

5.3.2 Developing Simulation

Once the model has undergone fine-tuning to enhance its accuracy, we will create a simulation of the Smart Control Traffic Signal. This simulation will illustrate the impact on signal timings based on the number of vehicles present in each direction.

5.3.3 Deployment

The deployment of the model involves integrating it with the current CCTV Cameras and Traffic Lights. Utilizing low-end processing devices such as Raspberry Pi, we can seamlessly incorporate the model into the existing infrastructure.

REFERENCES

[1] "OpenCV Documentation," OpenCV. [online]. available: https://docs.opencv.org/. [accessed: 05-aug-2023]."

[2] "Ultralytics YOLOv8 Docs.," Ultralytics. [online]. available: https://docs.ultralytics.com/. [accessed: 08-aug-2023]."

[3] "Weights and Biases Documentation," Weights and Biases. [online]. available: https://docs.wandb.ai/. [accessed: 05-dec-2023]."

[4] "Supervision," Roboflow [online]. available: https://supervision.roboflow.com/. [accessed: 06-dec-2023]."

[5] "Fast API Documentation," DevDocs [online]. available: https://devdocs.io/fastapi/. [accessed: 08-aug-2023]."

[6] "Uvicorn Introduction," Uvicorn. [online]. available: https://www.uvicorn.org/. [accessed: 08-dec-2023]."

[7] V. V. Vedansh Bhardwaj, Yaswanth Rasamsetti, "Image Processing Based Smart Traffic Control System for Smart City," *2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT), Kharagpur, India*, pp. 1–6, 2021.

[8] R. D. V. R. N. Bharath R.V.1, Gils Paul, "AI Based Smart Traffic Management," *Indian Scientific Journal Of Research In Engineering And Management*, vol. 7, pp. 1–5, 2023.

[9] N. Jain, R. Parwanda, and A. Chauhan, "Real-time smart traffic control and simulation: An approach for urban congestion management," in *2023 IEEE IAS Global Conference on Emerging Technologies (GlobConET)*, 2023, pp. 1–6.

[10] A. Kanungo, A. Sharma, and C. Singla, "Smart traffic lights switching and traffic density calculation using video processing," in *2014 Recent Advances in Engineering and Computational Sciences (RAECS)*, 2014, pp. 1–6.