Intelligent Traffic Control

A Project Report Submitted in the
Partial Fulfillment of the Requirements
for the Award of the Degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted by

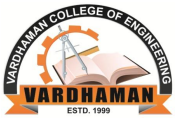
G.Soumya sree	**19881A05D5**
M.Kavya	**19881A05F9**
P.Aditya Sasank	**19881A05G4**

SUPERVISOR
Dr.R.Karthikeyan
Proffesor

Department of Computer Science and Engineering

VARDHAMAN COLLEGE OF ENGINEERING, HYDERABAD
An Autonomous Institute Affiliated to JNTUH

May, 2022



VARDHAMAN COLLEGE OF ENGINEERING, HYDERABAD

An Autonomous Institute Affiliated to JNTUH

Department of Computer Science and Engineering

CERTIFICATE

This is to certify that the project titled **Intelligent Traffic Control** is carried out by

G.Soumya sree	**19881A05D5**
M.Kavya	**19881A05F9**
P.Aditya Sasank	**19881A05G4**

in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science and Technology** during the year 2021-22.

Signature of the Supervisor
Dr.R.Karthikeyan
Proffesor

Signature of the HOD
Dr. Ramesh Karnati
Professor and Head, CSE

Kacharam (V), Shamshabad (M), Ranga Reddy (Dist.)–501218, Hyderabad, T.S.
Ph: 08413-253335, 253201, Fax: 08413-253482, www.vardhaman.org

Acknowledgement

The satisfaction that accompanies the successful completion of the task would be put incomplete without the mention of the people who made it possible, whose constant guidance and encouragement crown all the efforts with success.

We wish to express our deep sense of gratitude to **Dr.R.Karthikeyan**, Proffesor and Project Supervisor, Department of Computer Science and Technology, Vardhaman College of Engineering, for his able guidance and useful suggestions, which helped us in completing the project in time.

We are particularly thankful to **Dr. Ramesh Karnati**, the Head of the Department, Department of Computer Science and Technology, his guidance, intense support and encouragement, which helped us to mould our project into a successful one.

We show gratitude to our honorable Principal **Dr. J.V.R. Ravindra**, for providing all facilities and support.

We avail this opportunity to express our deep sense of gratitude and heartful thanks to **Dr. Teegala Vijender Reddy**, Chairman and **Sri Teegala Upender Reddy**, Secretary of VCE, for providing a congenial atmosphere to complete this project successfully.

We also thank all the staff members of Computer Science and Engineering department for their valuable support and generous advice. Finally thanks to all our friends and family members for their continuous support and enthusiastic help.

G.Soumya sree

M.Kavya

P.Aditya Sasank

Abstract

The goal of this project is to manage traffic, which is a major concern in many metropolitan areas. Because of the expanding number of vehicles and the limited resources given by current infrastructures, traffic problems are becoming more prevalent. The most straight forward method of operating a traffic signal is to utilise a timer for each phase. Another option is to utilise electronic sensors to detect vehicles and generate a cycled signal. We present an object detection-based traffic light control system. Instead of employing electronic sensors placed in the pavement, the system will detect automobiles using photographs. The photographs will be captured using a CCTV camera stationed alongside the traffic light. Image sequences will be captured.The Object Detection here is carried out using a package in Python called Imageai. Imageai consists of some pre-trained models. Vehicles like cars, Trucks, Motorcycles are counted and density on each line is calculated and signals change accordingly.

Keywords: metropolitan; Imageai; Sensors

Table of Contents

List of Figures

Abbreviations

Abbreviation	Description
HTML	Hyper Text Markup Language
CSS	Cascading Style Sheet
JS	Java Script
COCO	Common Objects in Context
AI	Artificial Intelligence
YOLO	You Only Look Once
API	Application Programming Interface
HTTP	Hyper Text Transfer Protocol
CCTV	Closed-Circuit Television
OPENCV	Open Computer Vision

CHAPTER 1

Introduction

1.1 Motivation

As the problem of urban traffic congestion spreads, there is a pressing need for introduction of advanced technology to improve the state of art of traffic control. In current situation, the signal remain green even if the vehicles have passed. Detection and classification of vehicle is essential for effective traffic control, for which, traffic related information needs to be collected and analyzed. Some other techniques are Magnetic Wireless Sensor detectors, Radio Frequency, Regression Analysis, Motion Vector Technique, etc. Standard traffic control system is is Manual Controlling, in which more man-power is required.

1.1.1 Basic Definition

To develop a web application that displays image instances of vehicles at traffic lights. The signal is then adjusted to reflect the amount of vehicles on each side. To begin, traffic density is recorded at signals, and time delays for traffic lights on the side with the most traffic are adjusted correspondingly. Transportation is very important in today's period, and as a result, traffic has increased. To reduce this, we need a better traffic control system that can cope with traffic situations and adjust control as needed. This clever method will reduce traffic while having no negative impact on transportation.

1.1.2 Traffic Junction

Figure 1.1: Traffic at a junction

1.2 Problrm Statement

To create an application that calculate the amount of vehicles on each side of the signal and displays the signal accordingly. It is impossible to efficiently control traffic by hand. Automatic Controlling is another approach that uses a timer for each phase. Electronic sensors are used to detect and produce signals from automobiles. A green light on an empty road may waste time. All of these drawbacks are meant to be addressed through Image Processing, which involves detecting cars using image instances.

CHAPTER 2

Methodologies and System Designs

2.1 Existing Methodologies

Usually junctions are the roads with heaviest traffic, there were several existing models which are designed to control traffic at the junction. Here are some of those existing models.

2.1.1 Manual Controlling

Requires extra man power. Moreover, only skilled operators can make judgements, hence, there is very high work load on skilled operators. There are many disadvantages in manual controlling mainly if the operator takes a wrong decision may cause many accidents.

Figure 2.1: Police officer controlling traffic

2.1.2 Micro controller

Controls the traffic lights at the zebra crossing or we can say traffic junction, which is not a flexible method, rather, there is fixed on and off timings for yellow, green and red lights.

Figure 2.2: Traffic light manual control post

2.1.3 Automatic Traffic Light

Performs its functioning through sensors and timers. Depending on timer values, lights are automatically getting ON and OFF. It will not solve the problem of waiting for a long time if one side has less traffic the timer is set previously may be long. Hence there may be more waiting for the people at that side of traffic junction.

Figure 2.3: Automatic Traffic Light

2.1.4 Vehicle Actuated Control

Attempt to alter the green light times on a regular basis. One of the key flaws is that it fails to account for automobiles waiting at red lights. A detector is placed ahead of the stop line and delivers signals to the controller sensitive to signals. Only if the predicted traffic flow matches the real traffic flow will the system work.

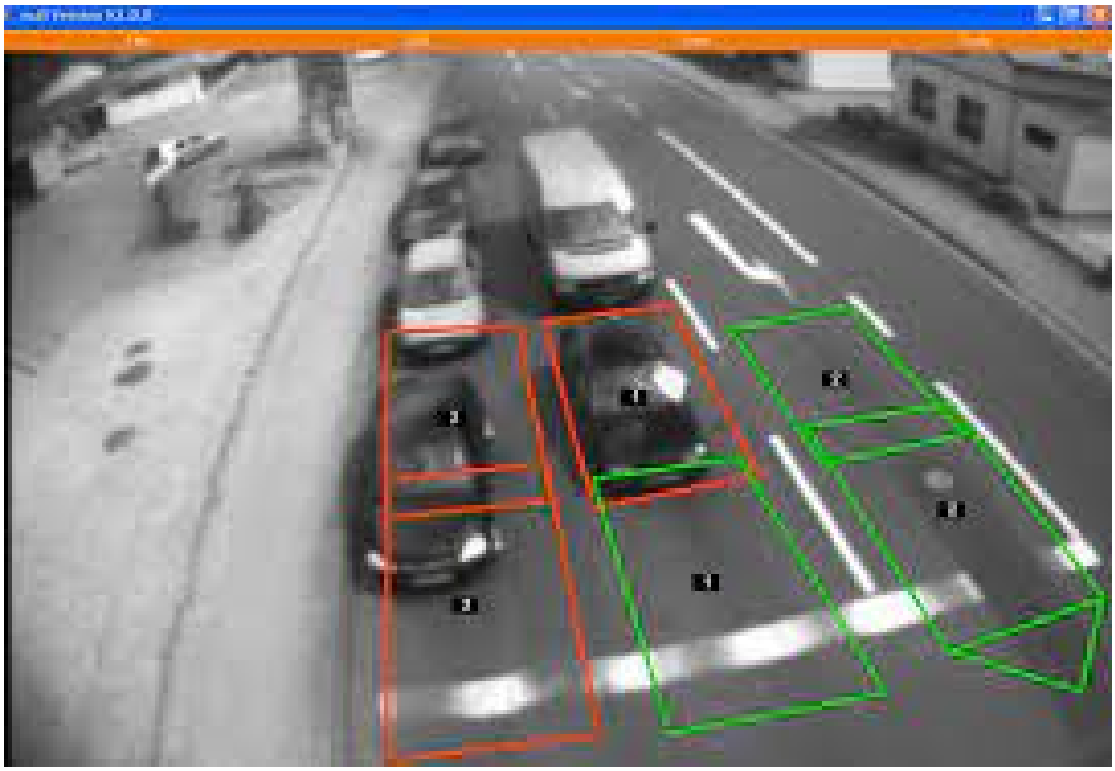


Figure 2.4: Design of vehicles acuted signal

2.1.5 Sensors

Another method is to use sensors to save time by shining a constant green light on a bare road. As a result, even with the use of Automatic Traffic Lights, traffic congestion remains an issue.

2.2 Proposed Methodology

The computer system is given image instances of vehicles at traffic signals. Then, using Python's object detection, it calculates the number of vehicles on each side and displays the signal accordingly.

We used the imageai package, which is a Python package. It has an object detection class built in. There are several pre-trained models in the imageai package. The COCO dataset was used to train these pre-trained models. Microsoft's COCO dataset is a large-scale object detection, segmentation, and captioning dataset. The COCO dataset is widely used by machine learning and computer vision engineers for a variety of computer vision projects. You can use the ObjectDetection class to detect objects in any image or set of images. The supported models are RetinaNet, YOLOv3, and TinyYOLOv3. You'll be able to detect and identify 80 different types of common household items as a result of this. When we give it an image, it processes it and adds to the count whenever it finds bikes, cars, trucks, and other vehicles. It does the same for the other three images and gives a green signal based on count value.

2.3 Architecture Of Proposed System

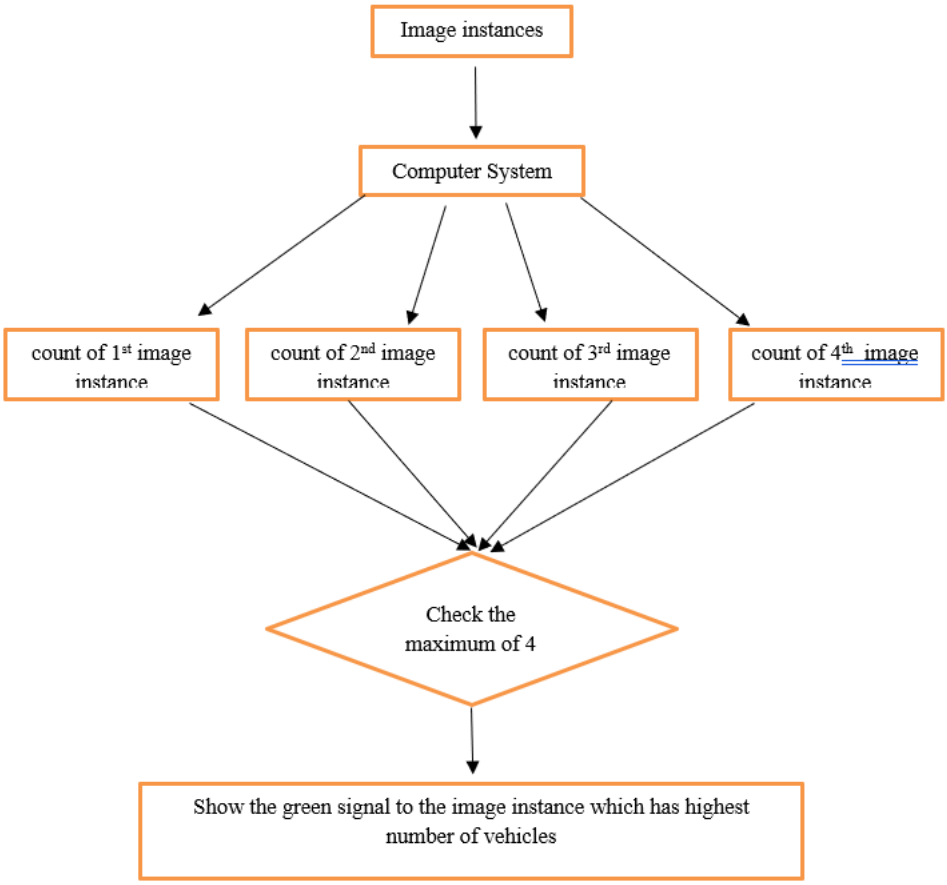


Figure 2.5: Architecture of the proposed system

As shown in Fig 2.1 The computer system is given image instances of vehicles at traffic signals. Then, using Python's object detection, it calculates the number of vehicles on each side and shows the signal accordingly. Using pre-trained models trained on the COCO dataset, this ObjectDetection class allows you to perform object detection on any image or series of images. Each image's count of vehicles is counted and represented as count1,count2,count3, and count4 respectively. We take the total of count1, count2, count3, and count4 and find the highest number. Display the green signal to the image instance with the most vehicles.

2.4 UML Diagrams

2.4.1 Activiy Diagram

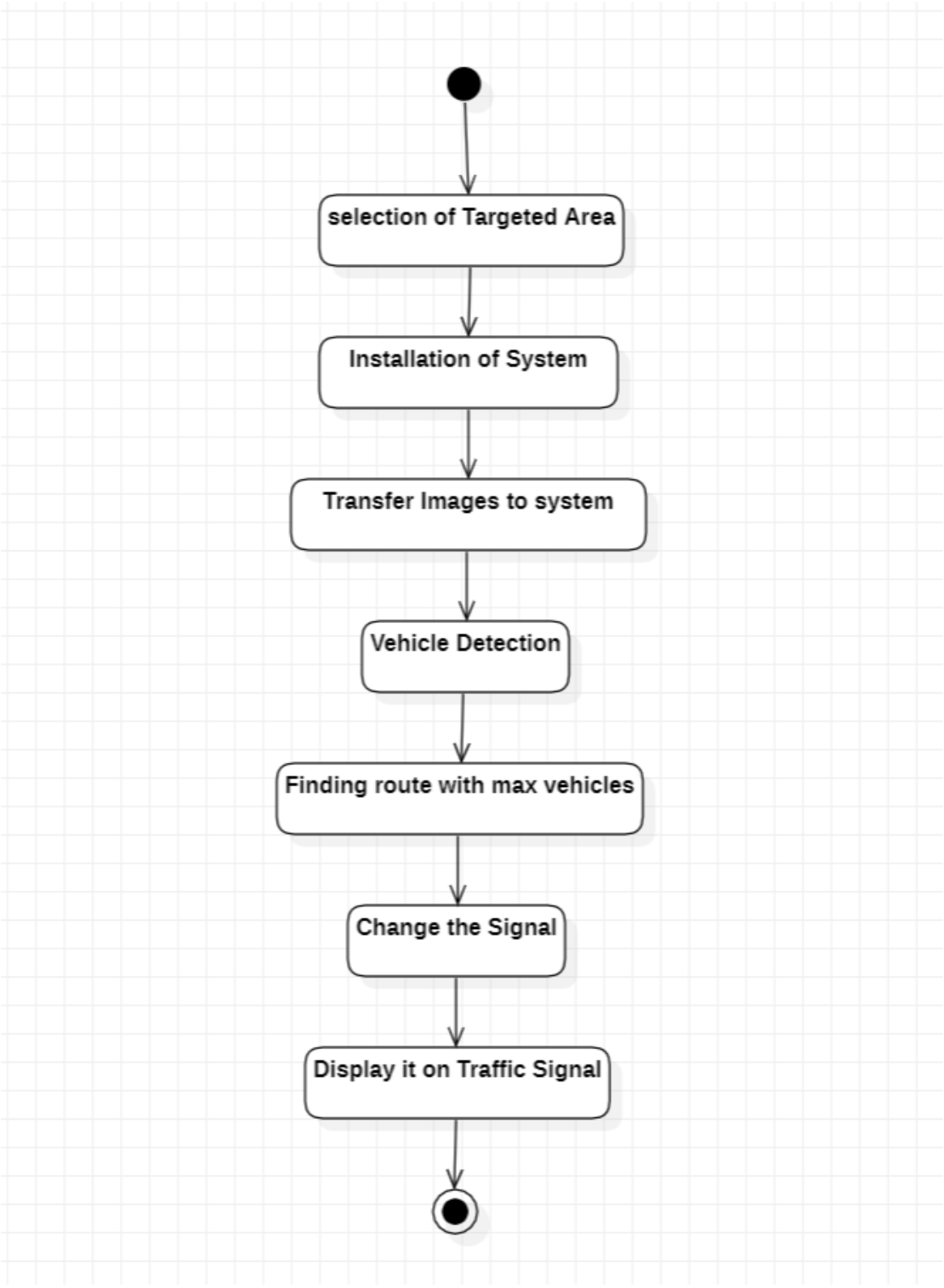


Figure 2.6: Activity Diagram

The activity diagram is a key UML diagram for describing the system's dynamic characteristics.Fig-2.6 represents an activity diagram for the Intelligent Traffic Control System. It is a flowchart that depicts the movement of information from one action to the next. The action can be described as a system operation.From one action to the next, the control flow is depicted.Every Activity diagram starts with a initial state and ends at a final point.This flow can be sequential, branching, or running at the same time. Different elements such as fork, join, and others are used in activity diagrams to cope with various types of flow control.The basic purposes of activity diagrams is to captures the dynamic behavior of the system.Activity diagrams are used not only to visualise a system's dynamic nature, but also to build the executable system utilising forward and reverse engineering approaches. The only portion of the activity diagram that is lacking is the messages.

2.4.2 Sequence Diagram

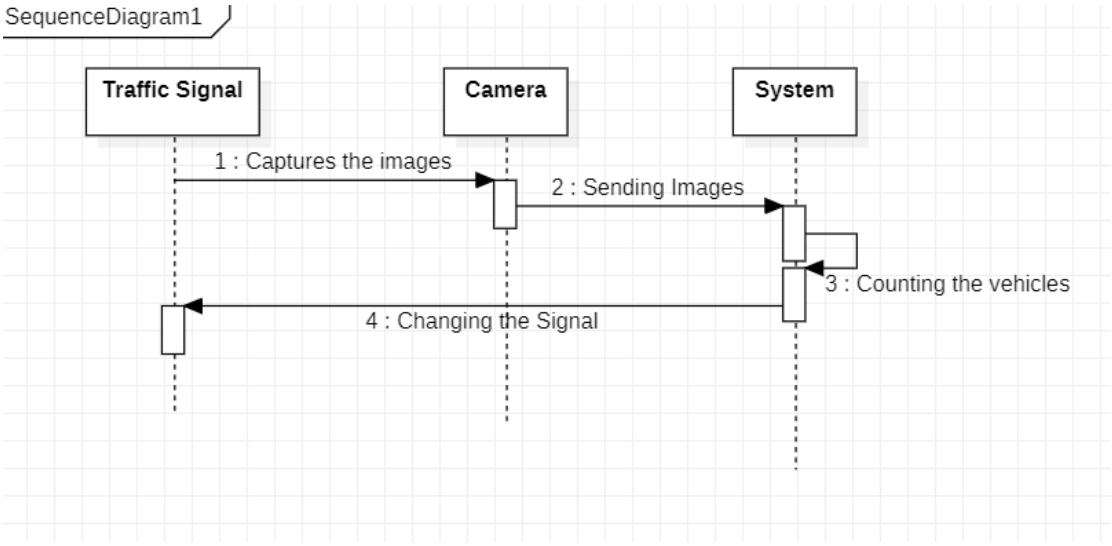


Figure 2.7: Sequence Diagram

A sequence diagram simply displays the order within which things interact, or the order during which these interactions occur.Fig-2.7 represents the Sequential diagram for the Intelligent Traffic Control System.A sequence diagram can even be spoken as an occasion diagram or an occasion scenario.

Sequence diagrams show how and in what order the components of a system work together.Software engineers frequently use these diagrams to document and understand requirements for brand new and current systems.

CHAPTER 3

SOFTWARE AND HARDWARE REQUIREMENTS

3.1 Software Requirements

3.1.1 HTML

HTML stands for Hyper Text Markup Language, and it is the most widely used language for developing web pages on the Internet. Berners-Lee created HTML in late 1991, but the first standard HTML specification, "HTML 2.0," was published in 1995. HTML 4.01 was announced in early 1999 as a major version of HTML. Though the HTML 4.01 version is still widely used, we now have the HTML-5 version, which is an extended version to HTML 4.01 and was officially launched in 2012.

The web browser uses the < !DOCTYPE > declaration tag to figure out what version of HTML is being used in the document. HTML 5 is the most recent version. Depending on the HTML version, there are a wide range of other declaration types that can be used in an HTML document.

HTML is a markup language that uses tags to format content, as previously stated. Angular braces enclose these tags ¡Tag Name¿ Except for maybe a few tags, the large percentage of tags have a closing tag.

A starting tag helps to identify an HTML element. If the element contains other content, it must end with a closing tag, with the element name preceded by a forward slash, as shown in the example below. HTML elements such as <img.../>, <hr.../>, and <br.../> do not have to be closed. These are referred to as void elements. HTML documents are made up of a tree of these elements that specify how HTML documents should be built and what type of information should go where in the document.

A starting tag helps to identify an HTML element. If the element contains

other content, it must end with a closing tag, with the element name preceded by a forward slash, as shown in the example below. HTML elements such as <img.../>, <hr.../>, and <br.../> do not have to be closed. These are referred to as void elements. HTML documents are made up of a tree of these elements that specify how HTML documents should be built and what type of information should go where in the document.

3.1.2 CASCADING STYLE SHEET

CSS (Cascading Style Sheets) describe how documents are displayed on screens, in print, or even how they are spoken. CSS allows you to specify various attributes for HTML tags in a simple and effective way. You can specify a number of style properties for an HTML element using CSS. A colon separates the name and value of each property (:). A semi-colon separates each property declaration (;). In your HTML document, you can use CSS in three different ways.

External Style Sheet: If you need to apply your style sheet to multiple pages, it's always a good idea to keep a separate file for it. The extension of a cascading style sheet file is.CSS, and it is included in HTML files using the <link> tag.

If you want to apply Style Sheet rules to a single document, use the <style> tag in the header section of the HTML document. The rules defined in the internal style sheet take precedence over those defined in the external CSS file.

Using the style attribute of the relevant tag, you can apply style sheet rules directly to any HTML element using inline style sheets. This should only be used when you want to make a specific change to an HTML element. The rules defined inline with the element take precedence over the rules defined in an external CSS file as well as the rules defined in the <style> element.

3.1.3 JAVA SCRIPT

A script is a small program that can be used to make your website more interactive. JavaScript or VBScript could be used to create this script. Any scripting language can be used to create a series of small functions known as event handlers, which can then be triggered using HTML attributes. Most web developers nowadays use only JavaScript and related frameworks; VBScript is not even supported by most major browsers. You can either keep JavaScript code in a separate file and then include it wherever it's needed, or define functionality directly in the HTML document. Let's take a glance at both cases one by one, with suitable example.

External JavaScript: It's best to keep functionality defined in a separate JavaScript file and then include it in your HTML documents if you're going to use it in multiple HTML documents. A JavaScript file does have the .js extension and is included in HTML files via the <script> tag.

You can write your script code directly into your HTML document using internal JavaScript. Script code is usually kept in the document's header using the <Script> tag, but there are no restrictions and you can put your source code anywhere in the document except inside the <Script> tag.

3.1.4 LAYOUT

A good webpage layout is essential for giving your website a professional appearance. Designing a website's layout with a great look and feel takes a long time. All modern websites now use CSS and JavaScript-based frameworks to create responsive and dynamic websites, but you can still create a good layout with simple HTML tables or division tags in combination with other formatting tags. The HTML tag is the most basic and widely used method of creating layouts. Because these tables are set up in columns and rows, you can use the rows and columns in any way you want. You can use a Multiple Column Layout to divide your web content into multiple pages. You can keep your content in the middle column and in the right column.You can put your

content in the middle column, use the left column for a menu, and use the right column for advertising or other items.

3.1.5 PYTHON

Python is a high-level, interpreted programming language that can be used for a variety of tasks. Python was created by Guido van Rossum and first launched in 1991. Its design philosophy demonstrates code readability and makes extensive use of whitespace. Its language constructs and object-oriented approach are aimed at assisting programmers in writing clear, logical code for both small and large-scale projects.

3.1.6 FLASK

Flask is a WSGI web application framework that's lightweight. It's built to create getting started simple and quick, with the pliability to rescale to more sophisticated projects. It started off as a basic wrapper for Werkzeug and Jinja and has since grown into one in every of the foremost popular Python web application frameworks.

3.2 Hardware Requirements

3.2.1 UBUNTU 16.04 OR LATER

Ubuntu is a free and open source operating system that runs on your computer, on the cloud, and on all of your internet-connected devices.

3.2.2 WINDOWS 7 OR LATER

Microsoft Windows is a collection of graphical operating system families that Microsoft has created, promoted, and sold. Windows NT and Windows IOT are active Microsoft Windows families, with subfamilies such as Windows Server and Windows Embedded Compact (Windows CE). Windows 9x, Windows Mobile, and Windows Phone are all defunct Microsoft Windows family.

3.2.3 RASPBIAN 9.0 OR LATER

The Foundation's official supported operating system is Raspbian. You can use NOOBS to install it, or you can download the image below and follow our installation instructions. Raspbian comes pre-installed with a wide range of educational, programming, and general-purpose software. Python, Scratch, Sonic Pi, Java, and other programming languages are included.

3.2.4 I3 PROCESSER AND ABOVE

The latest 10th Gen Intel® Core™ i3 processors with Intel® Iris® Plus graphics bring broad-scale artificial intelligence (AI) to the PC for the first time. With approximately 2.5x accelerated AI performance1, approximately 2x graphics performance2, nearly 3x faster wireless speeds3 with Intel® Wi-Fi 6 (Gig+) and the fastest4 and most versatile port available via Thunderbolt™ 3 technology, these processors bring a new level of integration to power PC experiences for today and the future.

Random papers are referenced here[1],here[2], here[3], here[4] and here[5]. Go check the References page.

CHAPTER 4

Implementation

4.1 Developing Backend Using Python

4.1.1 ImageAI

ImageAI is a python library built to empower developers, reseachers and students to build applications and systems with self-contained Deep Learning and Computer Vision capabilities using simple and few lines of code. This documentation is provided to provide detailed insight into all the classes and functions available in ImageAI, coupled with a number of code examples.ImageaAIis a project developed by Moses Olafenwa and John Olafenwa , the DeepQuest AI team.

ImageAI package consists of several pre-trained models like RetinaNet, YOLOv3, and TinyYOLOv3.There are several charecteristics for each model making them unique from each other.

Network Structure of RetinaNET is shown in Fig-4.1

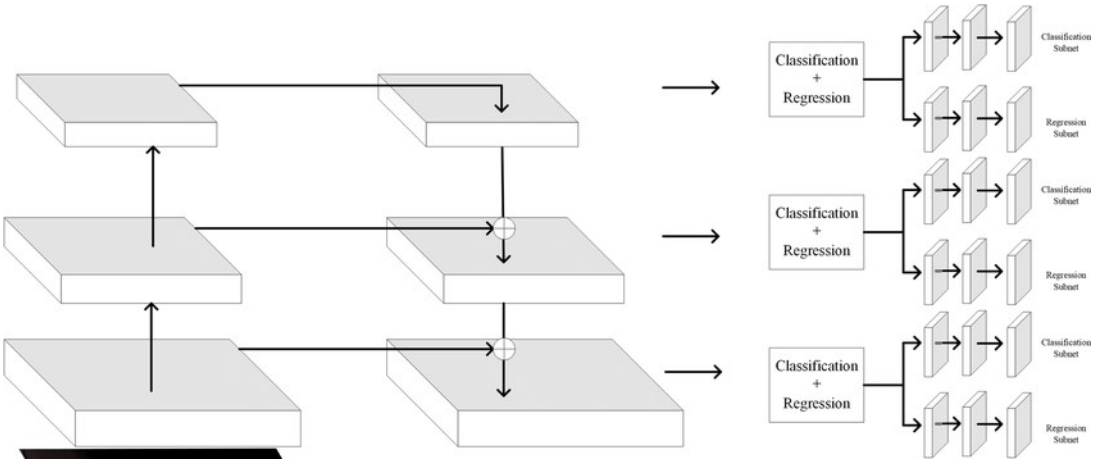


Figure 4.1: Structure of RetinaNET

Yolov3:Prior detection methods perform detection by repurposing classifiers or localizers. They apply the model to an image at different scales and places. Detections are parts of the image with a high score.

Yolo takes a really unique approach. Here Yolo uses one neural network to process the complete image. This network divides the image into regions and assigns bounding boxes and probabilities to every one of them. The projected probabilities here are wont to weight these bounding boxes.

Compared to classifier-based systems, our model has significant advantages. At test time, it examines the whole image, thus its predictions are influenced by the image's overall context. In contrast to systems like R-CNN, which require thousands of network evaluations for one image, it provides predictions with only 1. This makes it 1000 times quicker than R-CNN and 100 times faster than Fast R-CNN.

YOLOv3 uses a few tricks to improve training and increase performance, including multi-scale predictions, a better backbone classifier, and more. The full details are in our paper!

Network Structure of Yolov3 is shown in Fig-4.2

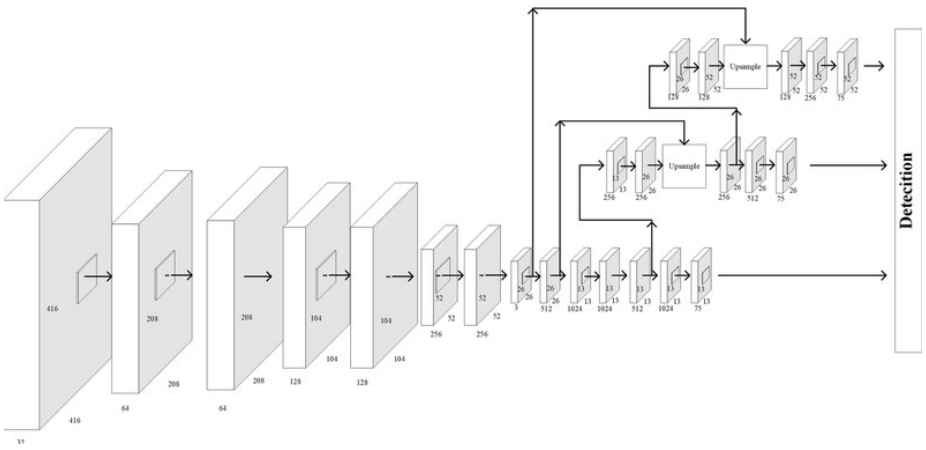


Figure 4.2: Structure of Yolov3

4.1.2 Procedure To Implement Imageai

Creating necessary folders

Our first task here is to create the necessary folders. To use imageai package we need the following folders.

1.Object detection:It is the root folder.In this folder we create all the remaining folders related to the project.This folder contains all the data related to project.

2.Models : This folder is to store the pre-trained models like Yolov3,Tinyolo and RetinaNET.

3.Input : In this folder we store image files on which we want to perform object detection

4.Output : This folder is to store image files with detected objects.

5.template : This folder contains HTML file .In flask we use **render_template**. It is used to generate output from a template file based on the Jinja2 engine that is found in the application's templates folder.

Note : That render_template is typically imported directly from the flask package.

Specify the path

After importing **objectdetecion()** class from imageai package and creating an instance we need to set the path for the model,input and output files. We need create variables for each folder then assign the path.

Ex :

model_path = "./model/modelname"

input_path = "./input/inputname"

output_path = "./output/name of the new image"

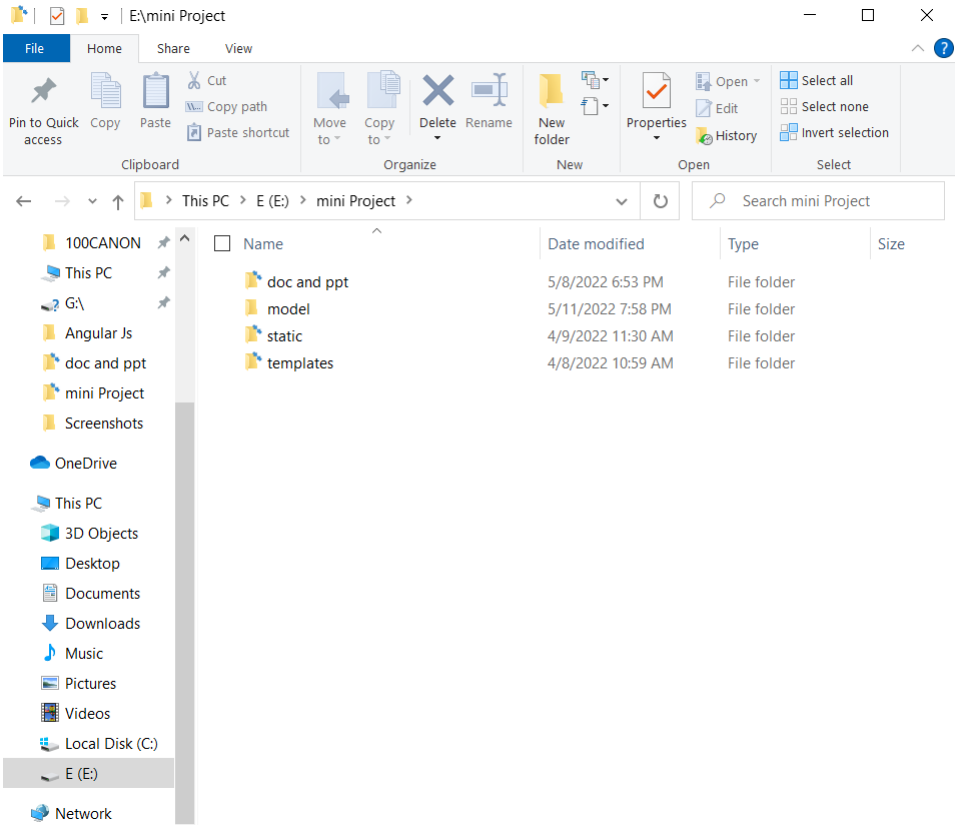


Figure 4.3: Structure of folders

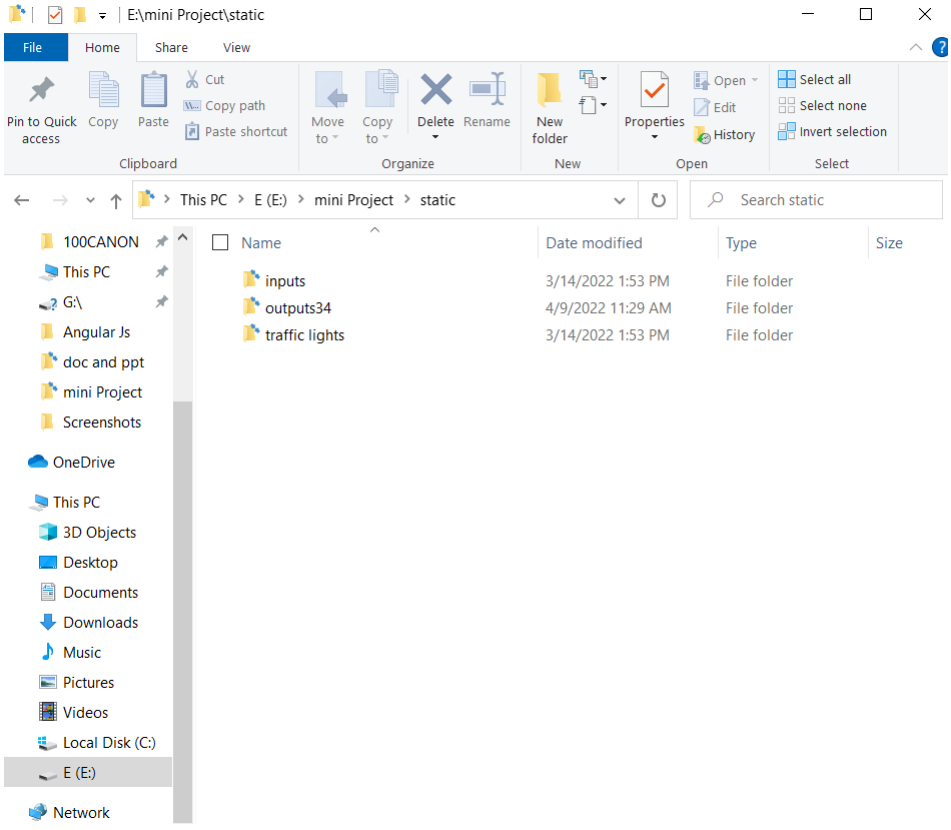


Figure 4.4: Structure of folders

As shown in Fig-4.3 and Fig-4.4, We need to create the neccesary folders for a project.

4.1.3 Function to Load Random Image from a File

.loadchoice():Image is randomly loaded using random.loadchoice in random module.

4.1.4 Function to create the instance of object detection

ObjectDetection(): An instance is created of objectdetection class.Once you have created an instance of the class, you can use the functions below to set your instance property an start detecting objects in images.

4.1.5 Function to set the Model type of the Object Detection

.setModelTypeAsYolov3(): This function changes the model type of the object detection instance you created to the Yolov3 model, which means you'll be using the pre-trained "Yolov3" model to perform your object detection tasks.

4.1.6 Function to Load the Model from the Specified Path

.loadModel(): This function loads the model into your object detection instance from the path you gave in the function call above.

4.1.7 Function That Performs Oject Detection

.detectObjectsFromImage(): After the model has been loaded, this function performs the object detection operation. It may be used repeatedly to recognise items in a variety of photos.

4.2 Creating Interface Using HTML and JINJA

For Python programmers, Jinja2 is a modern templating language. It was built using the Django template. It's used to generate HTML, XML, and other markup types, which are then sent to the user through HTTP. Jinja2 is compatible with Python versions 2.6.x, 2.7.x, and ¿= 3.3. You can use an older version of Jinja2 (2.6) if you're running Python 3.2, as support for Python 3.2 was removed in Jinja2 version 2.7. HTML allows scripting languages like JavaScript to insert programs that influence the behaviour and content of online pages. CSS determines the appearance and layout of material. The World Wide Web Consortium (W3C), which oversees both the HTML and CSS standards, recommends CSS over explicit presentational HTML.
Random papers are referenced here[6], here[1] and here[7]. Go check the References page.

4.3 Dependencies

To use ImageAI in an application, we must have installed the following dependencies:
1.Python 3.7.6
2.Tensorflow 2.4.0
3.OpenCV
4.Keras 2.4.3

Tensorflow: TensorFlow's high-level APIs are based on the Keras API standard for defining and training neural networks. Keras enables fast prototyping, state-of-the-art research, and production—all with user-friendly APIs.

OpenCV (Open Source Computer Vision Library): OpenCV is a free software library for computer vision and machine learning. OpenCV was created to provide a common infrastructure for computer vision applications and to let commercial goods incorporate machine perception faster.

Keras: Keras is a high-level neural networks API, written in Python and capable of running on top of TensorFlow, CNTK, or Theano. It was developed with a focus on enabling fast experimentation.

CHAPTER 5

Results

5.1 Outputs using different Models

5.1.1 Retina net

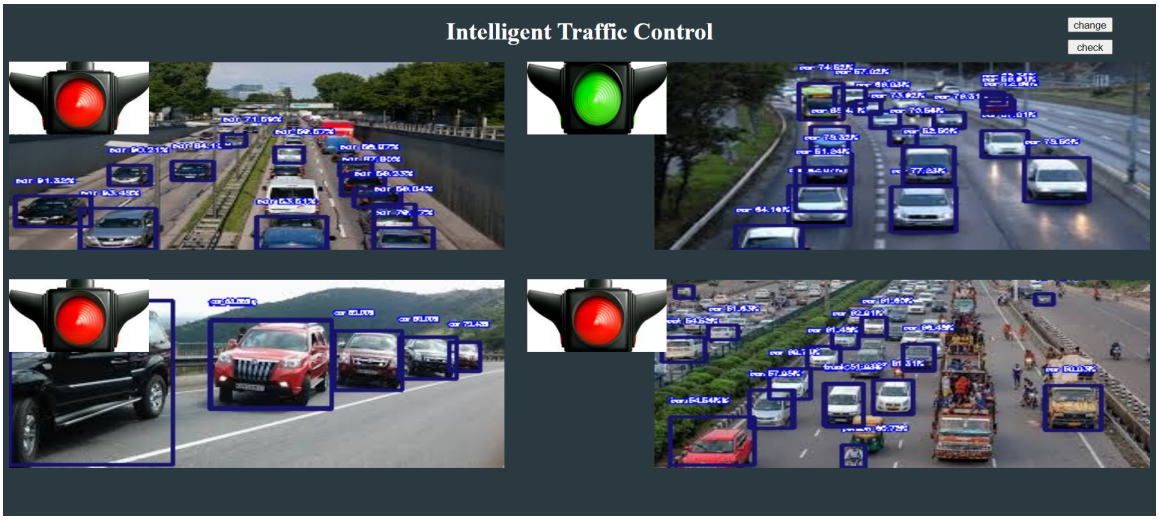


Figure 5.1: Output using ResNet50

RetinaNet is one of the best one-stage object detection models that has proven to work well with dense and small scale objects. But as we can observe in the Fig 5.1 retinaNET is not best suitable for vehicle identification in this particular cases, few vehicles on the corners of the pictures were left unidentified.

5.1.2 TinyYolo

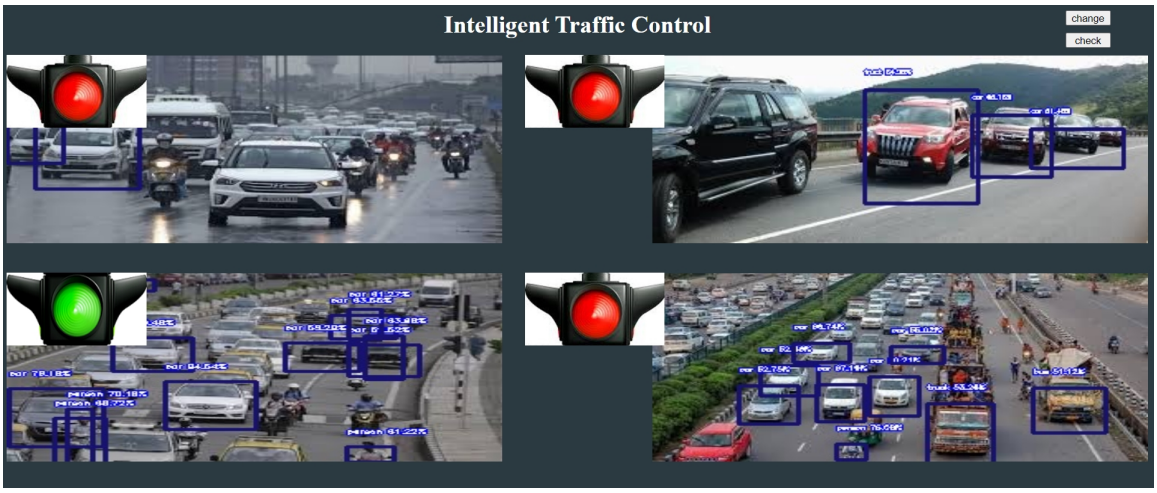


Figure 5.2: Output using TinyYolo

Tiny-Yolov3 is a simplified version of YOLOv3, which has a much smaller number of convolution layers than YOLOv3, which means that tiny-yolov3 does not need to occupy a large amount of memory, reducing the need for hardware. And it also greatly speeds up detection, but lost some of the detection accuracy. what stops us to use the tinyyolo is the less accuracy,which is most important for the project.

5.2 WebPage Results

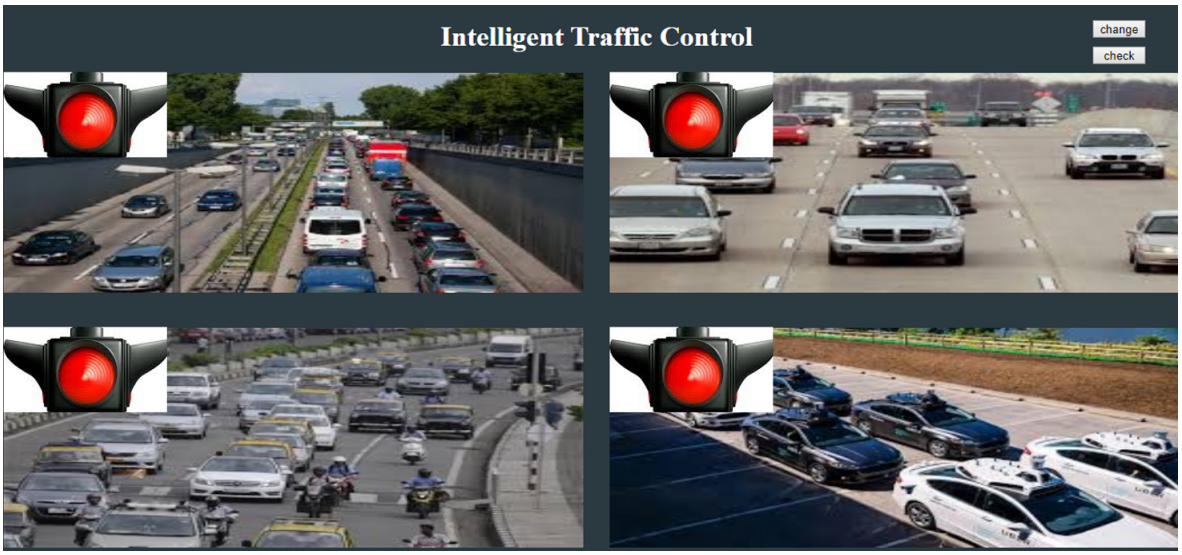


Figure 5.3: Initial WebPage

As shown in Fig 5.3 will be loaded when you start the web page ,it has four image instances of the vehicles at four different sides at traffic signal junction. It has two buttons on the top right, change an check. If you click change the webpage is loaded with the other 4 image instances of the vehicles and if you click check it will check the number of vehicles in each image instance and show the green signal to the image which has the highest number of vehicles.

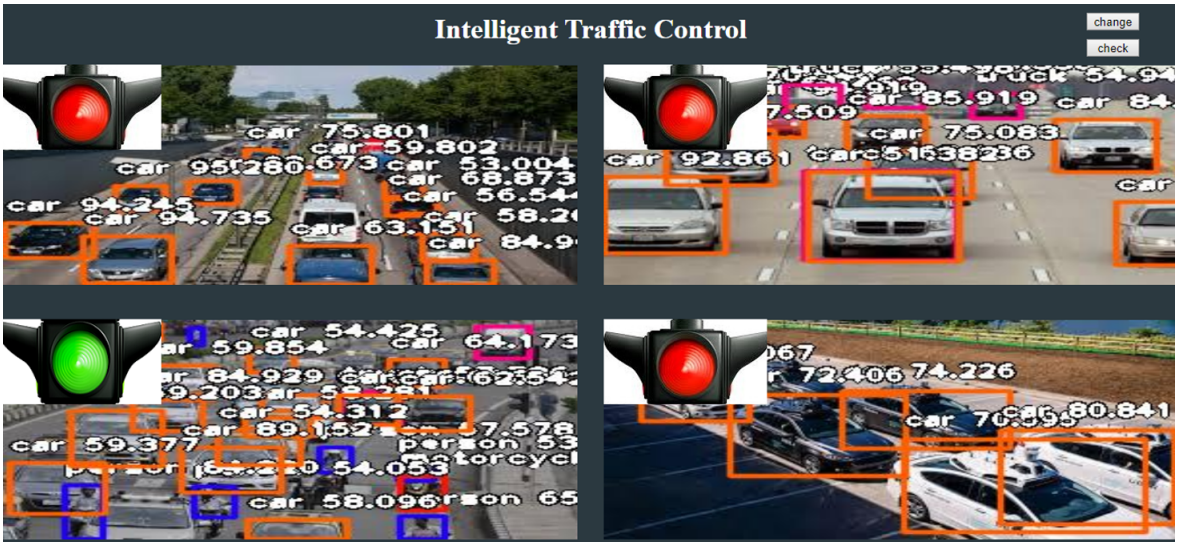


Figure 5.4: Result WebPage

As shown in Fig 5.4 after clicking the check button then you will get the above page. It will show the green signal to the road lane which has highest number of vehicles. Also it will show the type of the vehicle recognized and the probability of the correctness of the vehicle recognized.

5.3 CommandPromt Results

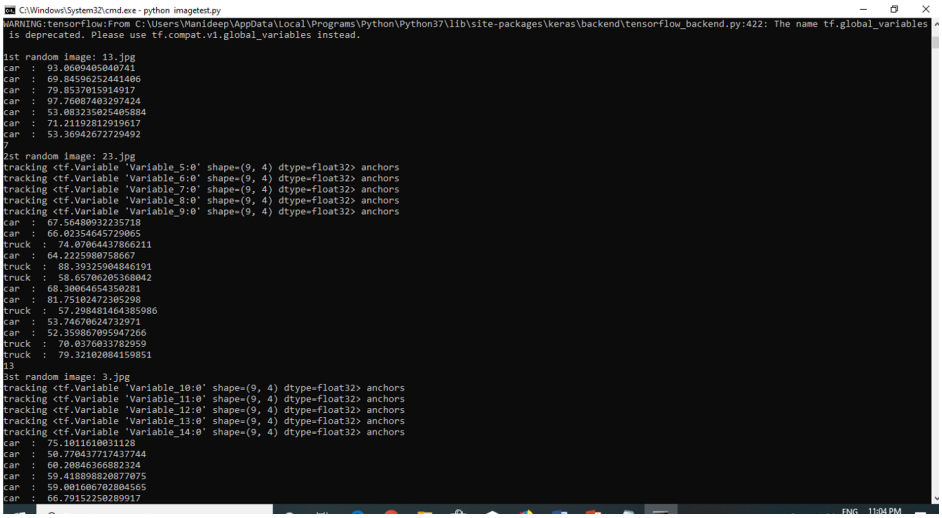


Figure 5.5: Loading the random images-1,2 and computing number of vehicles.

As shown in Fig-5.5 the loading of the first and second random images given in the input directory and the vehicles and persons recognized.It also shows the probability of the correctness of the vehicle recognized.

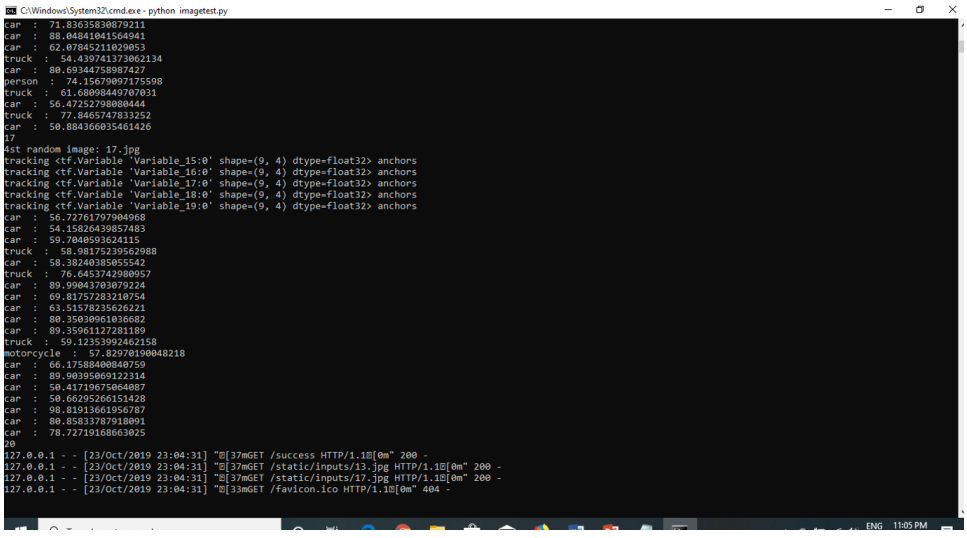


Figure 5.6: Loading the random images-3,4 and computing number of vehicles.

As shown in Fig-5.6 the loading of the third and fourth random images given in the input directory and the vehicles and persons recognized.It also shows the probability of the correctness of the vehicle recognized.

5.4 Output Directory Results

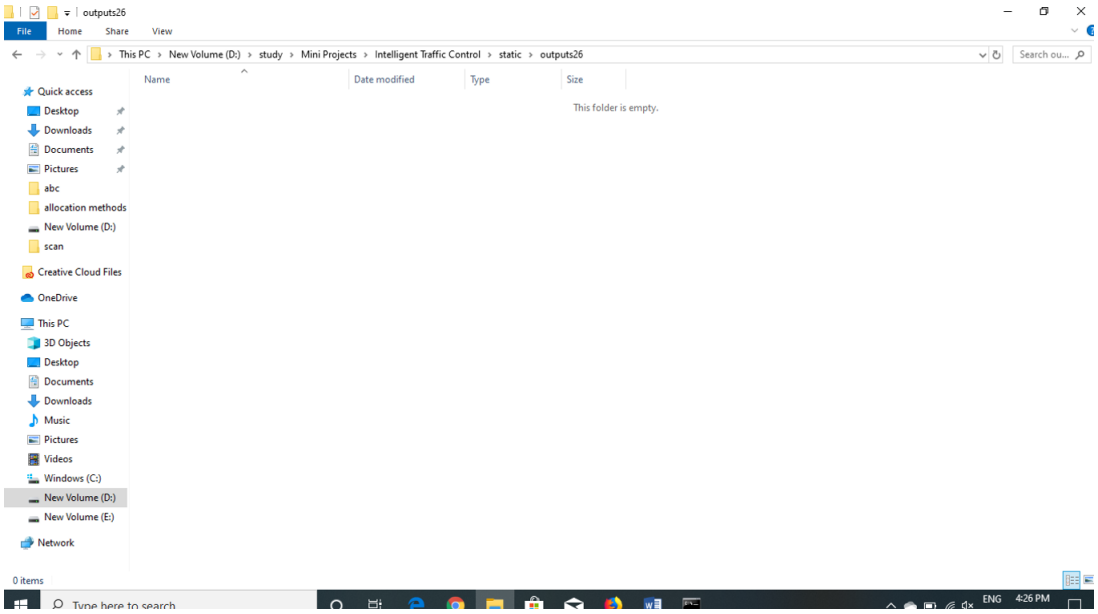


Figure 5.7: output directory before running of the code.

As shown in Fig-5.7 the directory of the output images before the execution of the program. It is empty as there are no images loaded and computed in the server.

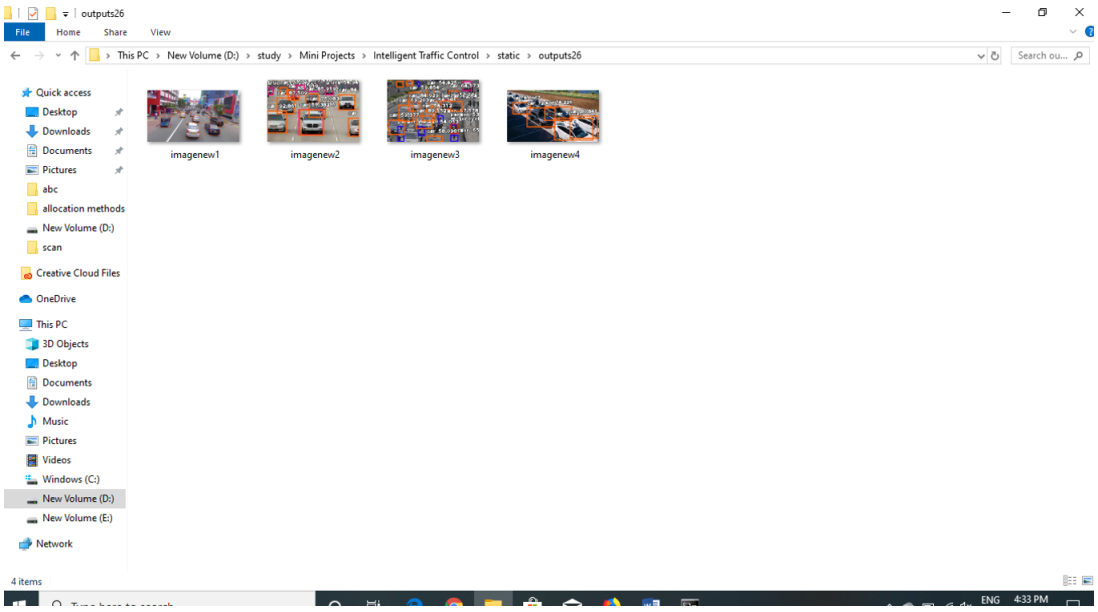


Figure 5.8: output directory after running of the code.

As shown in Fig-5.8 the directory of the output images after the execution

of the program .The four images are loaded and computed, the final images after recognized are saved in this folder.

CHAPTER 6

Conclusions and Future Scope

6.1 Conclusion

A density-based traffic light control system was designed for traffic control at the '+' road intersection in this design work to eliminate unnecessary time wastage and reduce road traffic casualties, which the current conventional traffic light control system failed to do. The design has demonstrated that the proposed system is a feasible traffic control tool, and the addition of a surveillance system would assist reduce road fatalities. Finally, the design's goals were accomplished. Using Python's Imageai, this model demonstrated how to direct traffic at a '+' road intersection. It specifically shows a workable software solution for traffic control depending on the density of traffic on each lane at the intersection.It gives an alternative to the traditional traffic light, which is connected with even timing of lanes of traffic regardless of the number of vehicles on the lanes per kilometre, which is the lane's density. The technology might also be set up to send the captured vehicle plate numbers of defaulters to the appropriate traffic agencies in real time. Finally, the design can be tweaked to handle traffic in more than four lanes.

6.2 Future Scope

6.2.1 Dynamic Inputs

Instead of photos, we may provide direct video feeds from traffic signal cameras, which can be used to compute the number of vehicles and adjust the signal accordingly. The system is not given dynamic image instances. Using CC-cameras on four sides of the traffic junction, we can dynamically provide photographs of the instances. Ambulances, as we all know, are the most vital medical mode of transportation in any country, as they deliver patients to local

hospitals. However, due to high traffic, ambulances are frequently detained in traffic for extended periods of time, putting patients' lives in jeopardy. As a result, the goal of our project extension is to address the issue of ambulances.

6.2.2 Emergency Vehicles

Ambulance are the most important vehicles,Due to the heavy traffic at junctions they get stuck and put the lives of patients at risk. we can further improve the intelligent traffic control to detect the ambulance and give priority to the roads with ambulance stuck,so that they could reach the hospital faster.

Figure 6.1: Ambulance stuck in Traffic

Figure 6.2: Detection of Ambulance

REFERENCES

[1] *Flask*. URL: http://flask.palletsprojects.com/en/1.1.x/.

[2] *Keras*. URL: https://github.com/keras-team/keras.

[3] *Tesnsorflow*. URL: https://github.com/tensorflow/tensorflow.

[4] *documentation*. URL: https://github.com/OlafenwaMoses/ImageAI.

[5] *Imageai*. URL: https://imageai.readthedocs.io/en/latest/.

[6] *javascript*. URL: https://www.w3schools.com/js/.

[7] *HTML*. URL: https://developer.mozilla.org/en-US/docs/Learn/HTML.