A

Project-I

On

# Smart Traffic Control Using Deep Learning

Submitted By

Mr. Naik Chintamani
Mr. Mudgal Yash
Mr. Patil Nikhil
Mr. Patil Vivek
Mr. Pandita Sandeep

In partial fulfilment for the award of the degree

of

**Bachelor of Engineering**

Under the Guidance of,

**Mr. J.S.Pujari**


**Department of Information Technology,**

**KIT's College of Engineering, Kolhapur.**

**2019-2020**

# Certificate

This is to certify that following **B.E. (IT)** students from KIT's College of Engineering, Kolhapur have completed Project-I successfully in the partial fulfilment for the award of degree of B.E. (Information Technology). They worked on the project **"Smart Traffic Control Using Deep Learning"** project during **SEM-I, 2019-2020** under the supervision of Mr.J.S.Pujari.

Mr. Naik Chintamani
Mr. Mudgal Yash
Mr. Patil Nikhil
Mr. Patil Vivek
Mr. Pandita Sandeep

Place: KIT Kolhapur
Date:

**Mr.J.S.Pujari** **Mr.T.B.Patil**

**Guide** **HOD IT** **External**

# Acknowledgement

We are highly grateful to Mr. T. B. Patil, HOD IT, KIT's College of Engineering, Kolhapur, for providing this opportunity to carry out the Project at the IT department. We would like to expresses our gratitude to other faculty members of the IT department for providing academic inputs, guidance & encouragement throughout this period. We would like to express a deep sense of gratitude and thanks to Mr. J. S. Pujari without whose permission, wise counsel and able guidance, it would have not been possible to carry out our project in this manner.

Finally, we express my indebtedness to all who have directly or indirectly contributed to the successful completion of our project.

# Abstract

This project is Smart Traffic Control which solves the problem of traffic congestion and traffic signal time allocation technique at a signal and its connected routes by taking live footage from camera installed at traffic signal and capturing count of vehicles present in a single lane to decide whether the time allocated to increase or decrease in order to increase utilization. Vehicle detection allows the use of various applications of artificial intelligence system for several purposes, especially: intelligent transportation, automatic monitoring, autonomous driving, and driver safety guarantee. In this work, we focus on the detection and recognition of vehicles in a video stream. For this reason, we have used the convolutional neural network technique (CNN) and a dataset that contains images to enable recognition and classification of vehicles.

We have identified types of vehicles in a video frame belonging to categories of car, truck to control traffic by adjusting traffic signal timing for each individual lane. Compared to traditional methods of object detection and classification, Deep learning methods are a new concept in the field of computer vision. Our model operates in two stages: a data preparation step, it consists of applying Treatments on the images composing the dataset in order to extract the characteristics, the second step is to apply the concept of convolutional neural networks to classify vehicles. After completion of the above two steps we got our own trained model for detection and classification of vehicles based on dataset of 200 car images and 200 truck images. This trained model is used for detecting vehicles in different images, videos and in a live video stream also.

# List of Figures



# List of Tables

# Table of Contents

# 1. Introduction

The problem of traffic congestion is increasing with the increasing number of vehicles and a solution is required which can adapt to the changing traffic. The traditional situation of handling traffic was with the help of a traffic policeman or officer. Later the traffic signal system was used but they have fixed time allocation techniques. Both of this solutions fail to provide the answer to ever changing incoming traffic handling. Therefore there is need for smart traffic control which can adapt to changing traffic and provide time allocation technique for all lanes at a single signal and alert other connected signal of incoming traffic ahead of time.

Vehicle detection allows the use of various applications of artificial intelligence system for several purposes as intelligent transportation, automatic monitoring, autonomous driving, and driver safety guarantee. In this work, we focus on the detection and recognition of vehicles in a video stream. For this reason, we have used the convolutional neural network technique (CNN) and a dataset that contains images to enable recognition and classification of vehicles.Convolutional neuron network efficiency depends to a large extent on the quality of the training data set; the network will produce good results only if the training data used contain sufficient important characteristics so that they can produce new predictions.

## 1.1 Problem Statement

The traffic handling schemes that are in use today are fixed time allocated traffic signal which do not change on incoming traffic or fail to provide time allocation scheme over changing traffic. Where traffic signals are not present this work is carried out by traffic policeman or traffic officer who is able to assess the incoming traffic and provide handling of traffic based on changing traffic. Both of this technique are not suitable where heavy traffic is present or its constantly changing. A solution is required to the traditional traffic signal which can provide better handling of incoming traffic and alert ahead of time of incoming traffic.

Thus there is need to make smart traffic control system which can identify types of vehicles in a video frame belonging to categories of car, truck, bikes and buses along with number of vehicles present to control traffic by adjusting traffic signal timing for each individual lane and send this data to its connected signals and alert them of incoming traffic to calculate respective time allocation for each individual lane by using deep learning algorithms and object detection.

## 1.2 Purpose

The purpose of this project is develop a desktop, stand-alone application for of Smart Traffic Control Using DL. An application which makes automatic allocation of time for traffic signal according to different conditions of traffic. The different software requirements of Smart Traffic Control Using DL are described in the Document which will outline all the technical aspects regarding development of Smart Traffic Control Using DL. It will illustrate the purpose and complete declaration for the development of system. It will also explain system constraints, interface and interactions with other external applications. This document is primarily intended to be proposed to a customer for its approval and a reference for developing the first version of the system for the development team. The software requirement specification in this document is an all time reference for the basic requirements for proceeding with the development.

## 1.3 Scope

The project deals with the development of desktop, stand-alone application for the purpose of Smart Traffic Control Using DL. The application is supposed to change the way the manual signal assignment is carried out for different situations. The project can help government to achieve increase in efficiency of signal allocation scheme as defined and used till date by the traffic regulations authorities. We focus on the detection and recognition of vehicles in a video stream. For this reason, we have used the convolutional neural network technique (CNN) and a dataset that contains images to enable recognition and classification of vehicles. The input video stream is CCTV at signal which provides live images. To identify types of vehicles in a video frame belonging to categories of car, truck, bikes and buses along with number of vehicles present to control traffic by adjusting traffic signal timing for each individual lane and send this data to its connected signals and alert them of incoming traffic to calculate respective time allocation for each individual lane by using deep learning algorithms and object detection. The video is converted into frames for processing and further calculations are carried out accordingly and results are sent to traffic signal timer which updates the different types of signals for different lanes. Also to make the system capable of carrying out its work on live CCTV footage.

## 1.4 System Analysis

### 1.4.1 Existing System

1) Traffic Officer - a single officer in charge of all lanes of traffic

2) Traffic Signal - a fix time allocation based traffic signal

The traffic handling schemes that are in use today are fixed time allocated traffic signal which do not change on incoming traffic or fail to provide time allocation scheme over changing traffic. Where traffic signals are not present this work is carried out by traffic policeman or traffic officer who is able to assess the incoming traffic and provide handling of traffic based on changing traffic. Both of this technique are not suitable where heavy traffic is present or its constantly changing.

### 1.4.2 Limitations of Existing System

1. The traffic officer has to manually allocate handling of traffic.

2. The traffic officer may not always give the best time for all traffic lanes.

3. The traffic signal has fixed time for all lanes which does not change.

4. Traffic signal fail to adapt to different situations of incoming traffic.

### 1.4.3 Proposed System

The proposed system is a smart traffic control using deep learning will identify traffic present and give time allocation for each lane according to different situations that will rise. The proposed system is dynamic and adaptable to changing incoming traffic. The proposed system will be able to handle traffic in dynamic manner than the existing static one. The proposed system deals with the development of desktop, stand-alone application for the purpose of Smart Traffic Control Using DL.

### 1.4.4 Advantages of Proposed System

- Solution to problem of fixed time allocation scheme for traffic signal
- More practical solution to handling of traffic
- Alert of incoming traffic ahead of time to take measures to prevent congestion of traffic
- The system will be able to monitor traffic all the time
- The efficiency of system is more than traditional system in use

## 1.5 Definitions, Acronyms and Abbreviation

The following conventions are used through the document and should be understood accordingly. The table below shows a list of all the conventions used in the document given below.

| | |
|---|---|
| DL | Deep Learning |
| CNN | Convolutional Neural Network |
| RCNN | Region Based CNN |
| CCTV | Close Circuit Television |

## 1.6 Overview

This project is a concept of smart traffic control system proposed as a solution to the conventional problem of fixed traffic signal and the problems faced by its manual behaviour. The Project has been implemented under the guidance of our professors and is used and implemented using the data set put forward by students. Thus the application developed is for college project purpose.

# 2. The Overall Description

## 2.1 Product Perspective

For development of the proposed system we have used the convolutional neural network technique and a dataset that contains images to enable recognition and classification of vehicles. Convolutional neural network efficiency depends to a large extent on the quality of the training data set; the network will produce good results only if the training data used contain sufficient important characteristics so that they can produce new predictions.

The proposed solution will have an object detection model which can identify different types of vehicles and it is built in Tensorflow. To develop an object detection model we need to give training images and apply pre-processing then pass the dataset to CNN algorithm. The model will be used for identifying different vehicles. The proposed project is supposed to work on as a desktop application or standalone as far as the project is concerned.

## 2.2 Product Functions

- The system will take input as a live video stream and apply object detection on it.
- The system will display the different types of objects detected in a single frame and label them as cars, trucks, bikes, etc.
- The system will initialize the counter of signal which is depend on number of object detected.
- The system will be send the count of the objects to next signal.
- The system will initialize the counter of the signal on basis of the data received from the previous signal or on its own

## 2.3 User Characteristics

User of the system in this case is the field officer who should be able to login using the ID provided. The ID is unique to the field officer who will be monitoring the traffic signal. The field officer can login to application and should be able to perform the following operations

- Login into the system against his ID to keep at track of his login status
    - Select input CCTV footage
    - Observe and monitor vehicles detected
    - Monitor traffic signal timers
    - Get results for traffic analysis and signal timer
    - Login using a single window on the server
    - Check the details of the current traffic
    - Check if proper lane assignment is carried out

# 3. SPECIFIC REQUIREMENTS

This section contains all the software requirements at a level of detail, that when combined with the system context diagram, use cases, and use case descriptions, is sufficient to enable designers to design a system to satisfy those requirements, and testers to test that the system satisfies those requirements.

## 3.1 External Interfaces

The Smart Traffic Control using Deep Learning will use the standard input/output devices for a personal computer. This includes the following:

- Keyboard
- Mouse
- Monitor
- Web camera/CCTV

### 3.1.1 User Interfaces

| | |
|---|---|
| Registration | New user registration |
| Login | Log into the system |
| Home | This Contains Services that user can access |
| Control Panel | Observe traffic and time allocation schemes |

### 3.1.2 Software Interfaces

| | |
|---|---|
| OpenCV | It mainly focuses on image processing, video capture and analysis including features like object detection. |
| Tenserflow | It is an open source artificial intelligence library, using data flow graphs to build models. |
| Anaconda Spyder | IDE for python application development |
| Anaconda | For compiling python files as we will be using python native to desktop application. |

### 3.1.3 Hardware Interfaces

The proposed project is supposed to work on any Desktop Application as far as the project is concerned. The system having configuration specified in operating environment and other requirement section of this document. There is no such hardware requirement for the Desktop application part.

### 3.1.4 Operating Environment

The following operating environment is required for the system to operate.

- A system with following specifications

    1. 8 GB RAM (16 GB recommended)
    2. Core i3/i5 processor
    3. GPU compatible with tesnsoflow-gpu if used
    4. Operating system Windows

### 3.1.5 Communication Interfaces

The project supports all configurations of devices fulfilling minimum requirements specified in operating environment section. The project supports windows operating system as well as Linux. We will be using web camera to communicate with the Desktop application.

### 3.1.5 Other Requirements

For training the model a computer with following specification is required

a) For **Tensorflow-CPU**
   1. Tensorflow cpu library 1.14.0 version
   2. Pentium Dual-Core or above CPU's
   3. RAM 4 GB minimum
   4. Disk Space 1 GB minimum

b) For **Tensorflow-GPU**
   1. Tensorflow library 1.14.0 version
   2. Pentium Dual-Core or above
   3. RAM 4 GB minimum
   4. Disk Space 1 GB minimum
   5. NVIDIA Graphics Driver with Compute Capability over 3.5
   6. NVIDIA CUDA 10.0 Toolkit
   7. NVIDIA cuDNN SDK

## 3.2 Functional Requirements

**1 User Module:**

**1.1 User Registration**

- User Register with Name, RTO ID, Phone Number, E-mail address.

**1.2 User Login**

- User Login with Username and Password.
- Forgot Password facility for Retrieving Password

### 1.3 Product Functions

The major functions of the product to be developed are shown in the block diagram below.

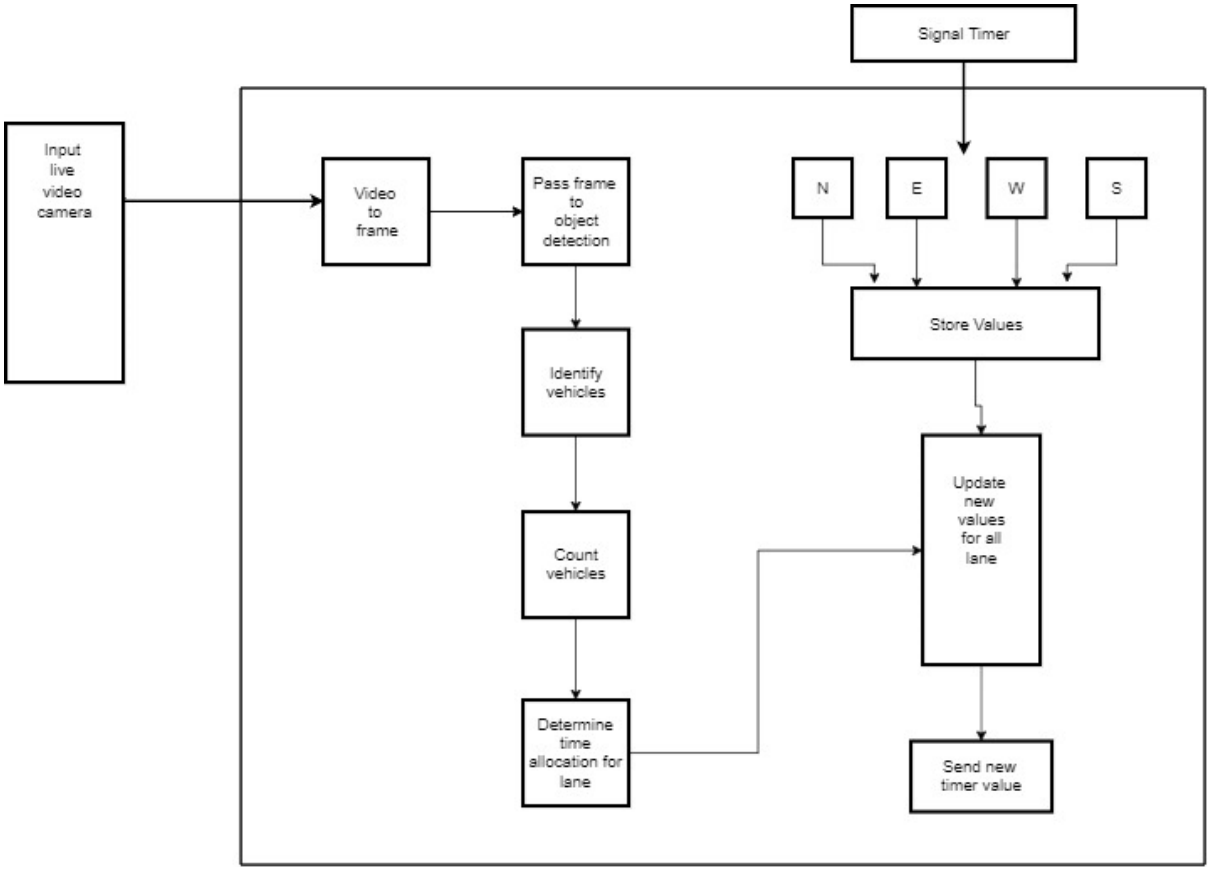


Figure: Block diagram of application

## 3.3 Nonfunctional Requirements

### 3.3.1 Performance Requirements

- The application should not crash under any circumstances. Care must be taken to avoid breaks in input stream.
- The application should be compatible across various operating versions of Windows, Linux
- Easy login and logout sessions for officers
- Application should process images in faster manner and in low amount of time

### 3.3.2 Safety Requirements

The make sure the data is not lost due to hard disk crashes or any other failures regular backup of collected data should be maintained. This requires regular backups of the data to be made every day to prevent any data loss of collected price.

### 3.3.3 Security Requirements

The password should be encrypted to prevent any data hacks so that incorrect assignment or sensitive data won't be compromised. Additionally the timer values calculated needs to be accurate.

### 3.3.4 Software Quality Attributes

#### 3.3.4.1 Reliability

The system will consistently perform its intended function. For e.g. The time allocation scheme for traffic.

#### 3.3.4.2 Efficiency

Unnecessary data will not be transmitted on the network. Application should process images in faster manner and in low amount of time

#### 3.3.4.3 Reusability

The system can be reused in any organization master definition under software license agreement.

#### 3.3.4.4 Integrity

Only system can change the timer values. Each user will be having rights to access the control panel for observing but not changing the data.

### 3.3.4.5 Security

Some of the factors that are identified to protect the software from accidental or malicious access, use, modification, destruction, or disclosure are described below. Specific requirements in this area could include the need to:

- Keep specific log or history data sets.
- Check data integrity for critical variables.
- Incorrect assignment or sensitive data won't be compromised.
- Additionally the timer values calculated do not get changed or modified.

# 4. SOFTWARE MODEL

For this project we have used prototyping model. Prototype is a working model of software with some limited functionality. The prototype does not always hold the exact logic used in the actual software application and is an extra effort to be considered under effort estimation. Prototyping is used to allow the users evaluate developer proposals and try them out before implementation. It also helps understand the requirements which are user specific and may not have been considered by the developer during product design.

Following is a stepwise approach used to develop a software prototype.

## 4.1 Basic Requirement Identification

This step involves understanding the very basics product requirements especially in terms of user interface. We collected object detection requirement for different types of vehicles.

## 4.2 Developing the initial Prototype

The initial Prototype is developed in this stage, where the very basic requirements are showcased and user interfaces are provided. These features may not exactly work in the same manner internally in the actual software developed. We developed a working prototype for object detection with cars and trucks.

## 4.3 Review of the Prototype

The prototype developed is then presented to the customer and the other important stakeholders in the project. The feedback is collected in an organized manner and used for further enhancements in the product under development.

## 4.4 Revise and Enhance the Prototype

The feedback and the review comments are discussed during this stage and some negotiations happen with the customer based on factors like – time and budget constraints and technical feasibility of the actual implementation. The changes accepted are again incorporated in the new Prototype developed and the cycle repeats until the customer expectations are met.

# 5. PROJECT PLANNING

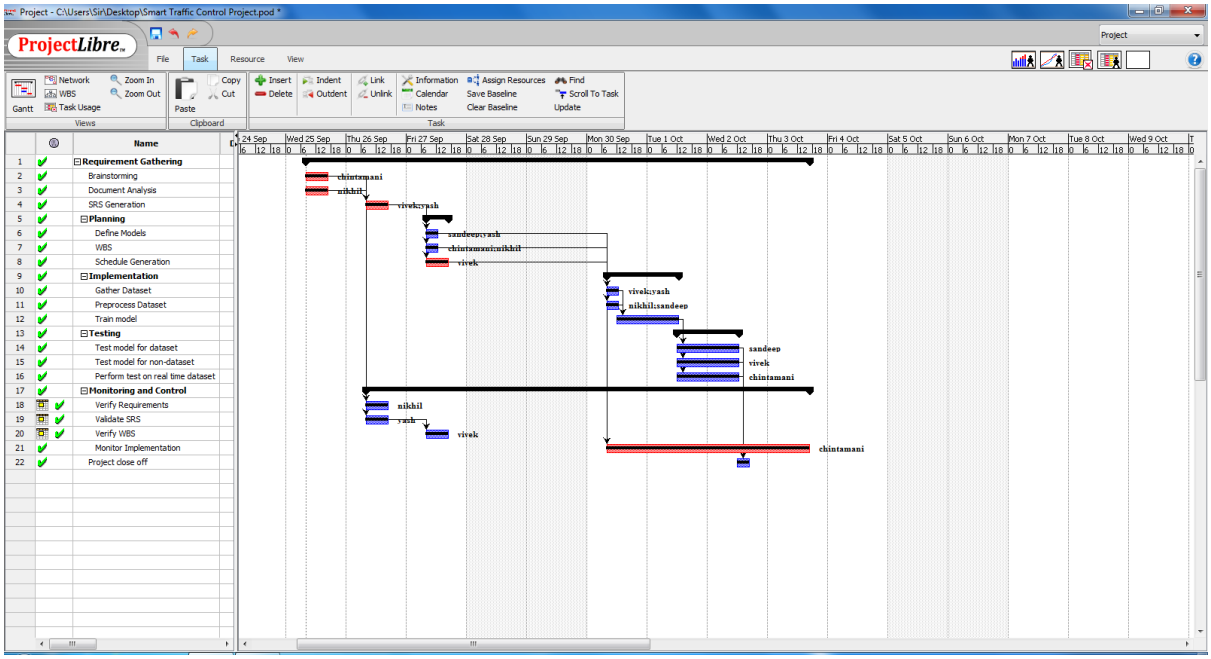


Figure – Project Planning in GANNT chart

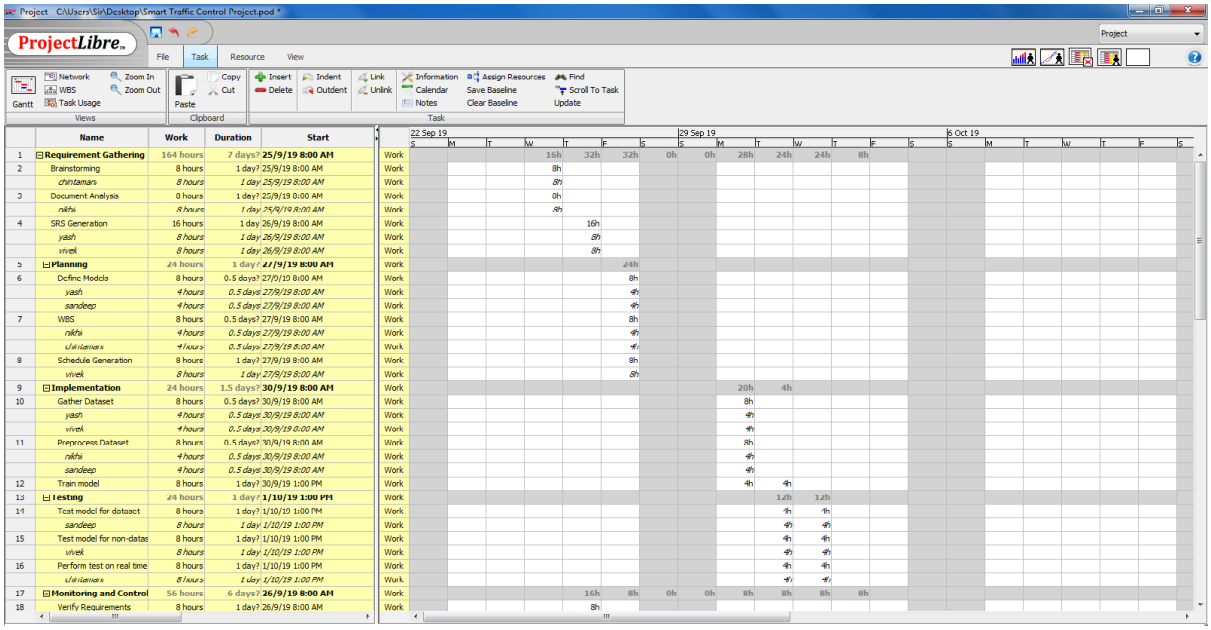


Figure – Project Planning Schedule

# 6. ANALYSIS MODULES

## 6.1 Data Flow Model

### 6.1.1 0 level DFD

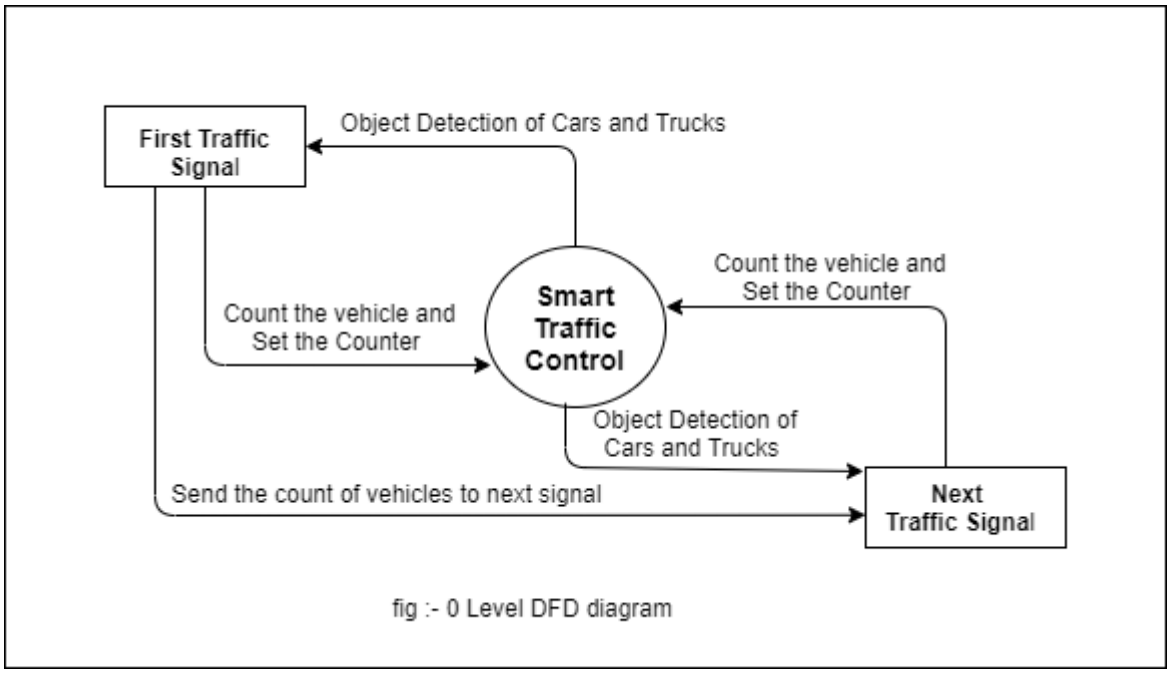


fig :- 0 Level DFD diagram

### 6.1.2 1 level DFD

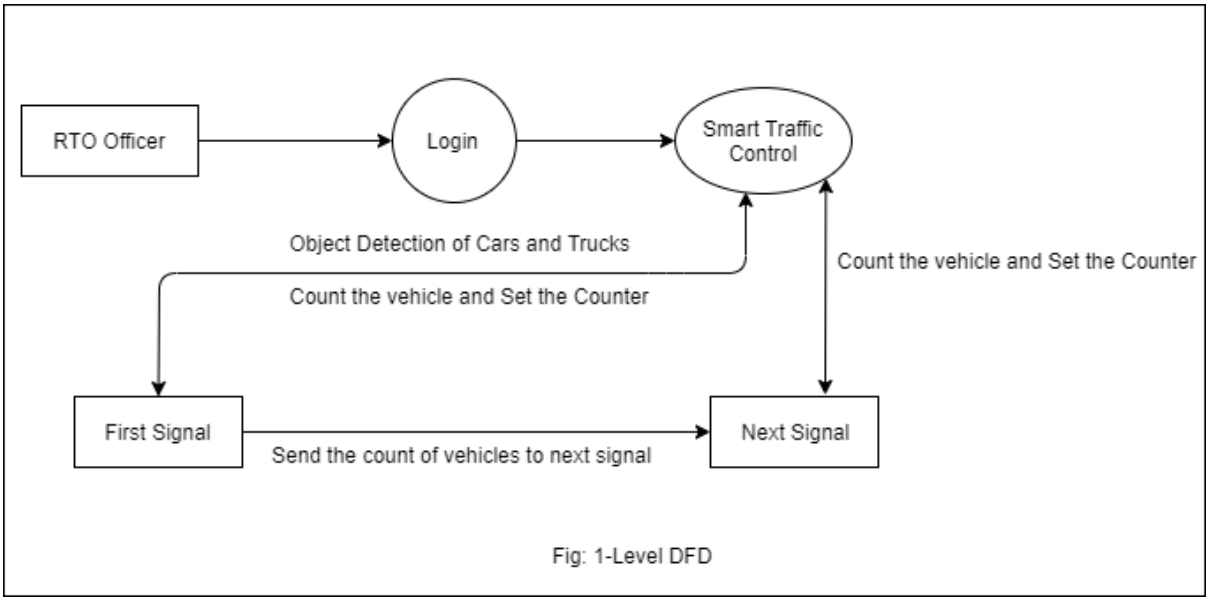


Fig: 1-Level DFD

## 6.2 Activity Diagram

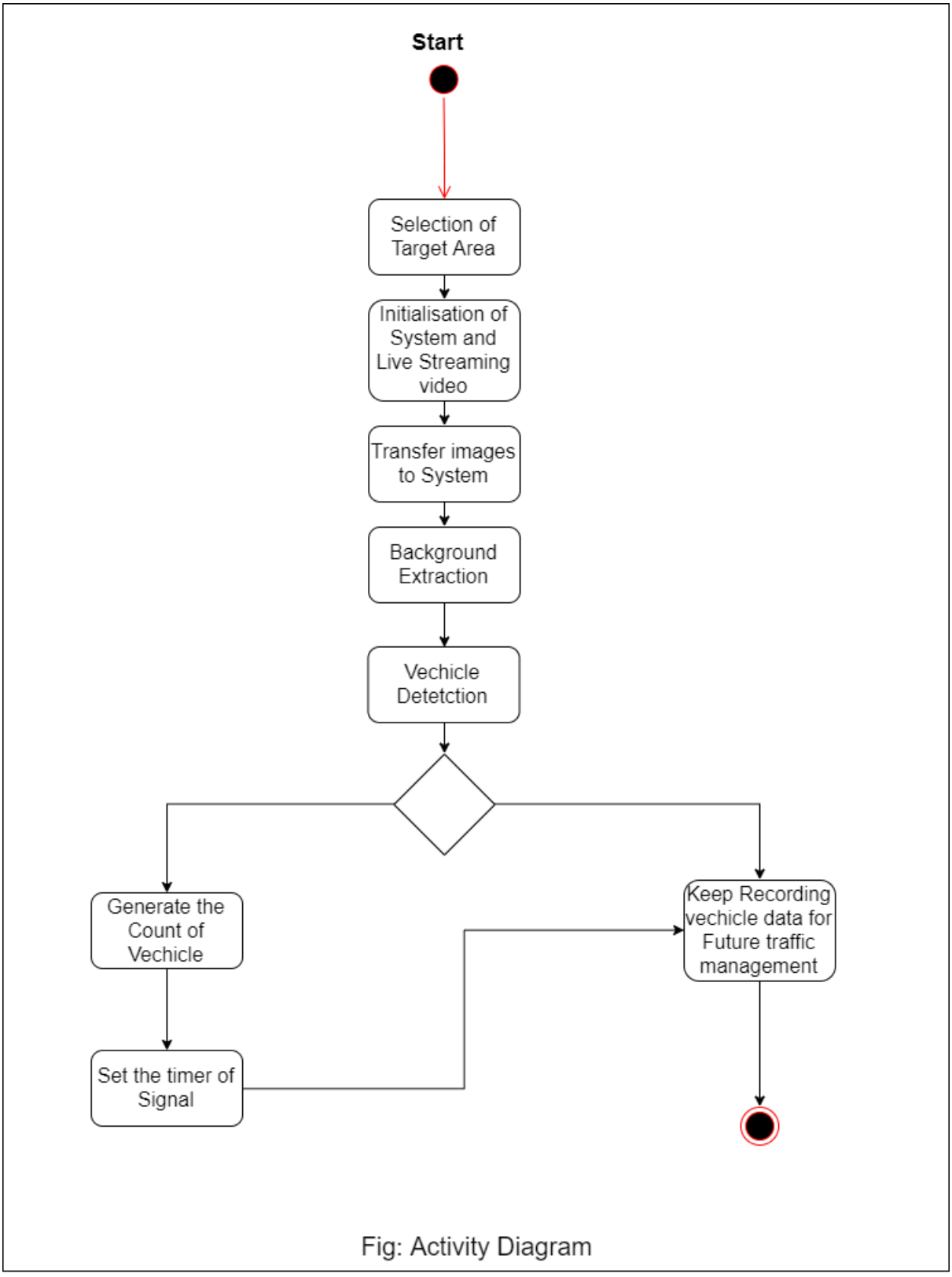


Fig: Activity Diagram

## 6.3 Sequence Diagram

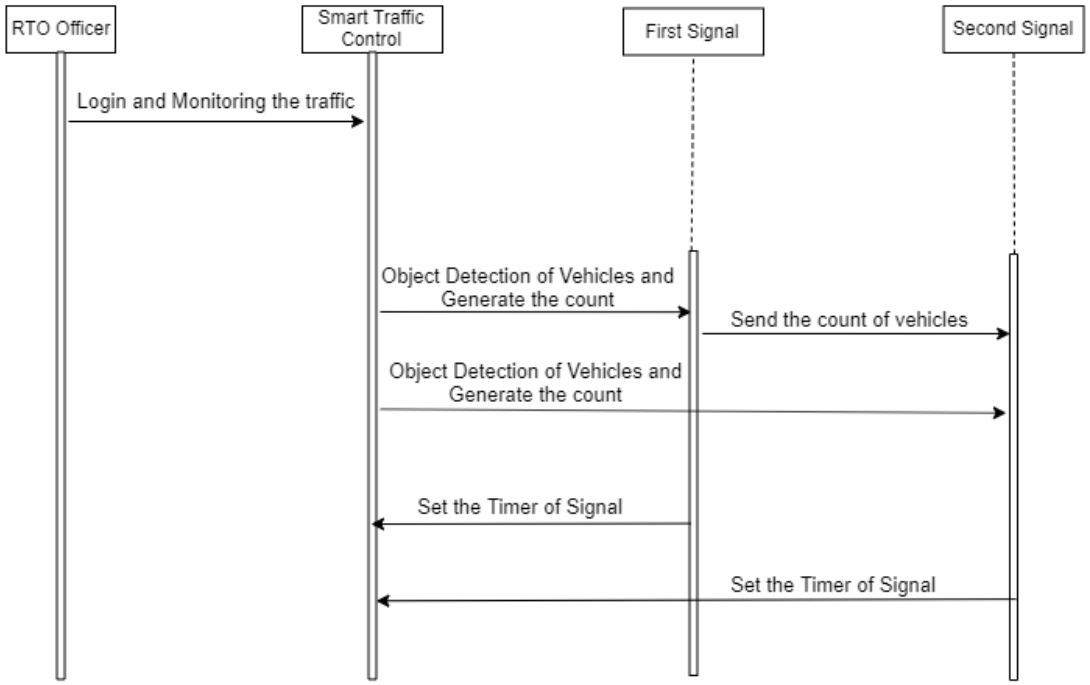


Fig: Sequence Diagram

# 8. IMPLEMENTATION

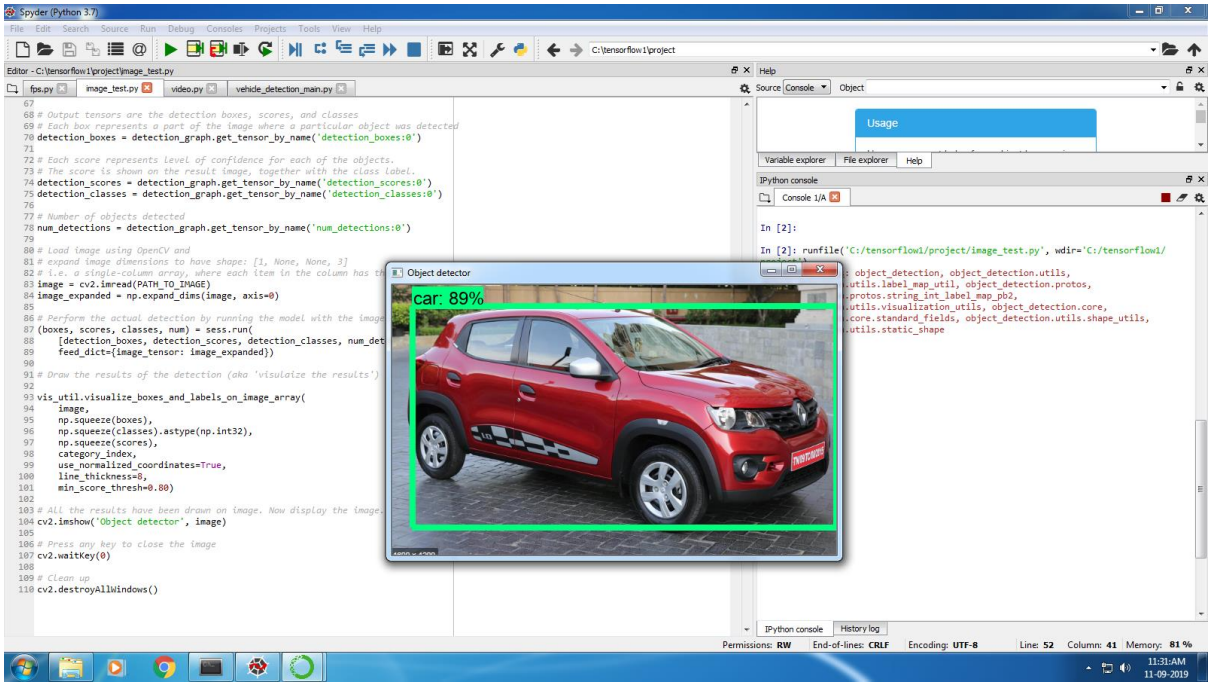


Figure-Implementation Screenshot 1

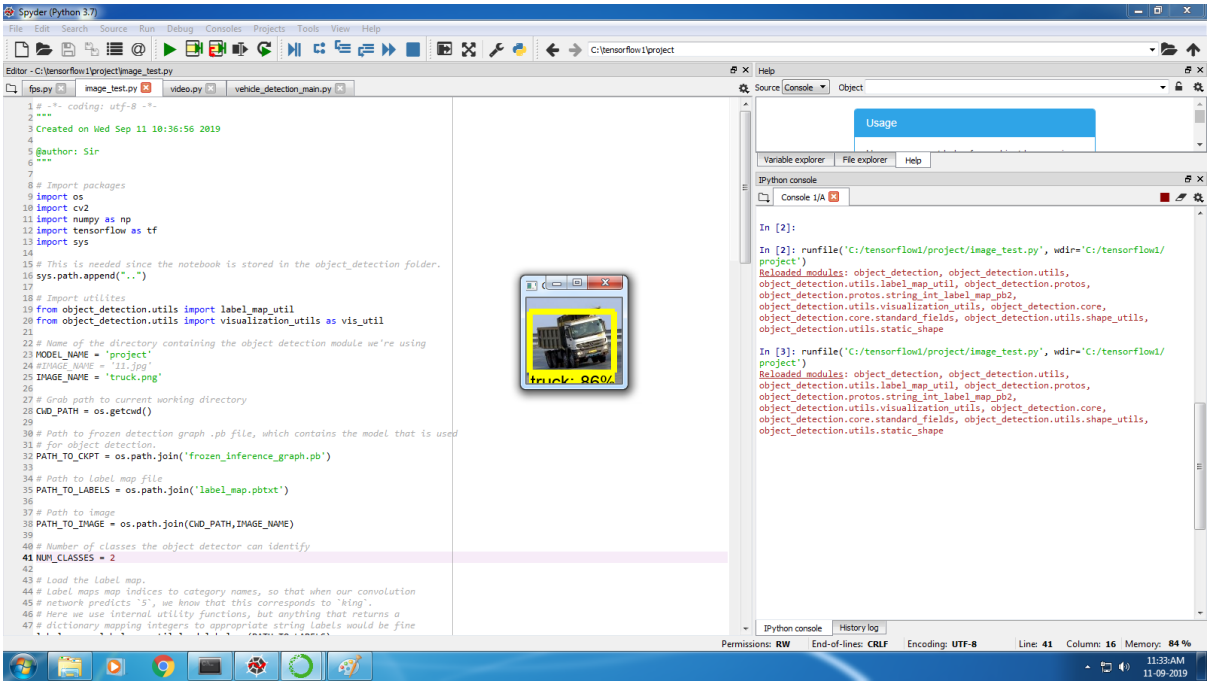


Figure-Implementation Screenshot 2

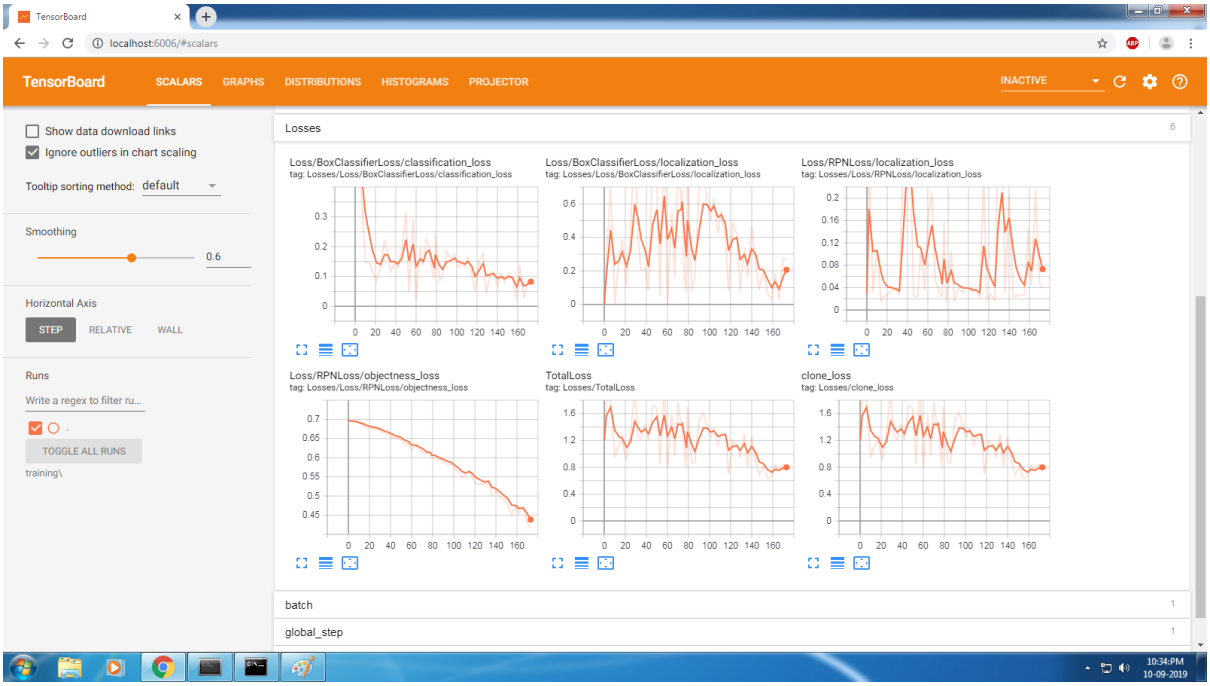


Figure-Implementation Screenshot 3

# 9. CONCLUSION

In this work vehicles are categorized into different class such as car, truck, bike, and bus based on our own dataset which contains labeled image dataset of 400 images. This classification and object detection model can be used for traffic detection, vehicle detection and other respective fields of vehicle detection. We have detected number of exact vehicles and used it in the timing allocation scheme for traffic signal to dynamically control the traffic.

## 10.REFERENCES

1) https://tensorflow-object-detection-api-tutorial.readthedocs.io/en/latest/install.html#tf-install

2) https://tensorflow-object-detection-api-tutorial.readthedocs.io/en/latest/install.html#tf-models-install

3) https://tensorflow-object-detection-api-tutorial.readthedocs.io/en/latest/install.html#labelimg-install

4) http://www.ijcstjournal.org/volume-6/issue-3/IJCST-V6I3P5.pdf

5) http://www.ijlemr.com/papers/volume4-issue2/4-IJLEMR-44009.pdf

6) http://www.tensorflow.org