An industrial oriented mini project report

on

VEHICLE DETECTION AND SPEED TRACKING

Submitted by

M. SREE HARISH KUMAR	**19W91A05C5**
L. V. LASYA	**19W91A05C1**
M. NIKITHA	**19W91A05D4**
N. SHIVA KUMAR	**20W95A0516**

Under the Esteemed Guidance of

Dr. SASIKUMAR P

TO

Jawaharlal Nehru Technological University, Hyderabad

In partial fulfilment of the requirements for award of degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

MALLA REDDY INSTITUTE OF ENGINEERING AND TECHNOLOGY

(UGC AUTONOMOUS)

(Sponsored by Malla Reddy Educational society)

(Affiliated to JNTU, Hyderabad)

Maisammaguda, Dhulapally post, Secunderabad-500014.

2022-2023

MALLA REDDY
INSTITUTE OF ENGINEERING & TECHNOLOGY
(Autonomous Institution - UGC, Govt. of India)
(Sponsored by Malla Reddy Educational Society)
Approved by AICTE, New Delhi, Recognized Under 2(f) & 12(B)
Affiliated to JNTU, Hyderabad, Accredited by NBA & NAAC with 'A' Grade

Department of Computer Science and Engineering

BONAFIDE CERTIFICATE

This is to certify that this is the bonafide certificate of an industrial oriented mini project report titled "**VEHICLE DETECTION AND SPEED TRACKING"** is submitted by **M. SREE HARISH KUMAR (19W91A05C5), L. V. LASYA (19W91A05C1), M. NIKITHA (19W91A05D4), N. SHIVA KUMAR (20W95A0516)** of B. Tech in the partial fulfilment of the requirements for the degree of **Bachelor of Technology in Computer Science and Engineering**, Dept. of Computer Science & Engineering and this has not been submitted for the award of any other degree of this institution.

Internal Guide Sign **Head of the Department Sign**

External Examiner Sign

DECLARATION

I hereby declare that the Mini Project report entitled **“VEHICLE DETECTION AND SPEED TRACKING”** submitted to Malla Reddy Institute of Engineering and Technology(Autonomous), affiliated to Jawaharlal Nehru Technological University Hyderabad (JNTUH), for the award of the degree of Bachelor of Technology in Computer Science & Engineering is a result of original industrial oriented mini project done by me.

It is further declared that the seminar report or any part thereof has not been previously submitted to any University or Institute for the award of degree or diploma

M. SREE HARISH KUMAR	**19W91A05C5**
L. V. LASYA	**19W91A05C1**
M. NIKITHA	**19W91A05D4**
N. SHIVA KUMAR	**20W95A0516**

ACKNOWLEDGEMENT

First and foremost, I am grateful to the Principal **Dr. M. ASHOK**, for providing me with all the resources in the college to make my project a success. I thank him for his valuable suggestions at the time of seminars which encouraged me to give my best in the project.

I would like to express my gratitude to **Dr. ANANTHA RAMAN G R,** Head of the Department, Department of Computer Science and Engineering for his support and valuable suggestions during the dissertation work.

I offer my sincere gratitude to my project – coordinator **Dr. M. NARENDHAR** and internal guide **Dr. SASIKUMAR P Professor** of Computer Science and Engineering department who has supported me throughout this project with their patience and valuable suggestions.

I would also like to thank all the supporting staff of the Dept. of CSE and all other departments who have been helpful directly or indirectly in making the project a success.

I am extremely grateful to my parents for their blessings and prayers for my completion of project that gave me strength to do my project

M. SREE HARISH KUMAR	**19W91A05C5**
L. V. LASYA	**19W91A05C1**
M. NIKITHA	**19W91A05D4**
N. SHIVA KUMAR	**20W95A0516**

TABLE OF CONTENTS

ABSTRACT

Manual and legacy technology is used to detect over speeding in developing countries like India. This process can be easily automated by leveraging the powers of image processing and machine learning. Moving object detection is one of the crucial tasks in image processing because of its important role in many real-world applications. Vehicle speed detection can be achieved by employing image and video processing methods to determine vehicular speed. In this work, we propose a novel approach for vehicle speed detection based on an integrated approach to detect moving objects in video sequence frames as opposed to the most commonly used RADAR and LIDAR devices for traffic law enforcement.

Video data is collected and analysed for speed in real-time without any sensor calibrations, thereby removing any hardware requirements. A Haar classifier has been trained and integrated into the application for detecting the vehicles. To make the detection more robust, we adopt an efficient object detection system by training the classifier on 1500 positive images.0 Moving vehicles are segmented out by using frame subtraction and masking techniques. The algorithm tracks the time taken by the car to cover a pre-determined distance in order to calculate its speed. This application can be used by the traffic police to automatically detect the speed of each car and note down any license numbers from the video stream in cases of over speeding.

LIST OF FIGURES

LIST OF SCREENS

CHAPTER-1

INTRODUCTION

Detection of vehicle and tracking of speed is the crucial part of town planning. In the last decade, vision-based traffic monitoring system has received considerable attention. This can be done with the help of vehicle detection and speed monitoring. The monitoring system gives various information about, vehicle count, traffic congestion and speed of the vehicle. One of the root cause of road accidents is speed.

Extracting frames from the video and comparing the speed between two given points can be used to determine whether the car is moving above the permissible limit or not. There are many algorithms available for extraction of vehicles from the background. Traditionally, radar systems were used for such applications but had some limitations. So to overcome the limitations in existing methods, various techniques have been developed for vehicle speed determination using image processing.

But the main factors that would affect these image processing algorithms is, waving of tree branches, camera noise, illuminations. The goal of this current project is to develop an automatic vehicle counting system along with the detection of speed, which can process videos recorded from stationary cameras over roads e.g. CCTV cameras installed near traffic intersections / junctions and counting the number of vehicles passing a spot in a particular time for further collection of vehicle / traffic data.

Vehicle Speed surveillance is a predominant factor in enforcing traffic laws. Traditionally vehicle speed surveillance was done using a radar technology which consists of a radar gun and a radar detector. Radar is an acronym that stands for Radio Detection and Ranging. Radar systems create radio waves, a form of electromagnetic energy that can be directed out into the air where the signals produced travel at the speed of light – roughly 186,000 miles per second, or 3.08 x 10^8 meters per second.

The transmission of these signals and the collection of returned energy that bounces off of objects in the path of the radar's transmission (called returned pulses) is what allows radar to be used to detect objects and range them, meaning establish their position and distance relative to the radar system's location. When a radar is used to detect the speed of an object (for example, when a police officer with a stationary radar gun is detecting the rate at which a car is moving), it does so by taking advantage of a phenomenon that occurs whereby the frequency of the radio wave for the return signal is altered because of the car's motion relative to the radar. If the car is moving toward the radar device, the return signal radio wave frequency increases.

The radar gun can then use this change in frequency to determine the speed at which the car is moving. This principle, which establishes that the difference between the frequency of the emitted pulse and the frequency of the return pulse varies with the relative motion of the source to the object, is called the Doppler effect. So, while the distance of an object can be established by the amount of time that it takes to detect the return pulse, the speed of an object can be detected by establishing the change in the pulse characteristics between transmitted and received echo.

This provides a velocity along the direction in which the radar is pointing, termed the radial velocity. One point to note is that the pulse characteristic changes used to establish the speed of a moving object like a car will depend on the relative position of the car to the radar. The measured speed will be accurate if the car moves directly towards the radar. But if the car's motion is at an angle relative to the radar gun's line of sight, the speed being measured will be a component of the actual speed of the vehicle. This principle is known as the cosine error effect. Because of these errors, the United States of America's law keeps a buffer of 8km which is caused because of the above-mentioned error. Also, radar technology can track one vehicle at a time. The project deals with vehicle detection and speed tracking which is explained further.

CHAPTER-2

LITERATURE SURVEY

There are many researches with the same topic for estimating vehicle speed based on image processing.

Survey 1: A Vehicle Speed Estimation Algorithm Based on Dynamic Time Warping Approach -IEEE SENSORS JOURNAL, VOL. 17, NO. 8, APRIL 15, 2017 [1]. Advantages of this paper is Experiment results show that the algorithm detection accuracy is better than 98%. DTW is the most relevant distance for time series analysis. But in DTW, heavy computational burden is required to find the optimal time alignment path. Also DTW has the Quadratic Complexity - Performance is directly proportional to the squared size of the input data set.

Survey 2: A Novel Motion Plane-Based Approach to Vehicle Speed Estimation [2]. Advantages are The centre point of vehicle's license plate is considered as the reference point for the car and speed of each vehicle is estimated by the displacement of its license plate in the time seen by camera. Estimates the motion plane using 3D position of license plates, which are estimated by Shape-from-Template (SFT) technique. And Drawbacks are Template matching techniques applicability is limited mostly by the available computational power, as the identification of big image patterns is time-consuming.

Survey 3: A Deep Learning Approach for Localization Systems of High-Speed Objects [3]. Advantages is Deep learning approach effectively suppresses the potential divergence of the modified EKF. Drawbacks are this approach is not used for the commercial purpose on Roadways. It is suitable to use in Aircrafts and Military purposes. High cost.

CHAPTER-3

SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

In recent times, there has been a drastic change in people's lifestyles and with an increase in incomes and lower cost of automobiles there is a huge increment in the number of cars on the roads which has led to traffic and commotion. The manual efforts to keep people from breaking traffic rules such as the speed limit are not enough. There is not enough police and man force available to track the traffic and vehicles on roads and check them for speed control. Hence, we require technologically advanced speed calculators installed that effectively detect cars on the road and calculate their speeds.

Radar is used to detect the speed of an object (for example, when a police officer with a stationary radar gun is detecting the rate at which a car is moving), it does so by taking advantage of a phenomenon that occurs whereby the frequency of the radio wave for the return signal is altered because of the car's motion relative to the radar. If the car is moving toward the radar device, the return signal radio wave frequency increases. The radar gun can then use this change in frequency to determine the speed at which the car is moving.

3.1.1 Disadvantages:

- Manual efforts required
- More manpower required
- The radar gun is used for speed calculation
- Low accuracy

3.2 PROPOSED SYSTEM

We have developed a Haar cascade to detect cars on the roads, whose velocities are then measured using a python script. this project can be done by creating a bigger haar cascade since bigger the haar cascade developed, more the number of vehicles that can be detected on the roads. Better search algorithms can allow a faster search and better detection of these vehicles for better efficiency. This project is to develop an algorithm to calculate the speed of the object(vehicle) detected. We have implemented the algorithm using Python Script.

3.2.1 Advantages:

- Lower manual intervention,
- Fast reaction,
- High detection rate.

3.3 FEASIBILITY STUDY

The feasibility of the project is analysed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are,

- Economic Feasibility
- Technical Feasibility
- Social Feasibility

3.3.1 ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

3.3.2 TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement; as only minimal or null changes are required for implementing this system.

3.3.3 SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

CHAPTER-4

SYSTEM REQUIREMENTS SPECIFICATION

4.1 REQUIREMENT ANALYSIS

The project involved analysing the design of few applications so as to make the application more users friendly. To do so, it was really important to keep the navigations from one screen to the other well-ordered and at the same time reducing the amount of typing the user needs to do. In order to make the application more accessible, the browser version had to be chosen so that it is compatible with most of the Browsers.

4.2 FUNCTIONAL REQUIREMENTS

Functional requirement should include function performed by a specific screen outline work-flows performed by the system and other business or compliance requirement the system must meet. Functional requirements specify which output file should be produced from the given file they describe the relationship between the input and output of the system, for each functional requirement a detailed description of all data inputs and their source and the range of valid inputs must be specified. The functional specification describes what the system must do, how the system does it is described in the design specification. If a user requirement specification was written, all requirements outlined in the user requirements specifications should be addressed in the functional requirements.

4.3 NON FUNCTIONAL REQUIREMENTS

Describe user-visible aspects of the system that are not directly related with the functional behaviour of the system. Non-Functional requirements include quantitative constraints, such as response time (i.e. how fast the system reacts to user commands.) or accuracy (i.e. how precise are the systems numerical answers.).

4.4 INPUT & OUTPUT DESIGN

4.4.1 INPUT DESIGN

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

- What data should be given as input?
- How the data should be arranged or coded?
- The dialog to guide the operating personnel in providing input.
- Methods for preparing input validations and steps to follow when error occur.

OBJECTIVES

1. Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.
2. It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.
3. When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow.

4.4.2 OUTPUT DESIGN

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.
2. Select methods for presenting information.
3. Create document, report, or other formats that contain information produced by the system.

The output form of an information system should accomplish one or more of the following objectives.

1. Convey information about past activities, current status or projections of the future.
2. Signal important events, opportunities, problems, or warnings.
3. Trigger an action.
4. Confirm an action.

4.5 SYSTEMS REQUIREMENT AND SPECIFICATION

4.5.1 HARDWARE REQUIREMENTS

The most common set of requirements defined by any application or software application for virtual computer applications, also known as hardware, Hardware Requirements list is usually accompanied by a hardware compliance list (HCL), especially if there are applications.

The HCL list checks hardware devices that are tested, compatible, and sometimes not compatible with a specific application or application. The following sections discuss various aspects of hardware requirements.

- RAM: 4GB and Higher
- Processor: Intel i3 and above
- Hard Disk: 500GB: Minimum

4.5.2 Software Requirements

Software requirements address the definition of software application requirements and pre-requisites that require computer installation to provide System performance. These prerequisites or requirements are not usually included in the software installation package and need to be installed separately before the software can be installed.

- Operating system : Windows 10.
- Coding Language : Python.
- Back-End : Python

CHAPTER- 5

SYSTEM DESIGN

5.1 SYSTEM ARCHITECTURE

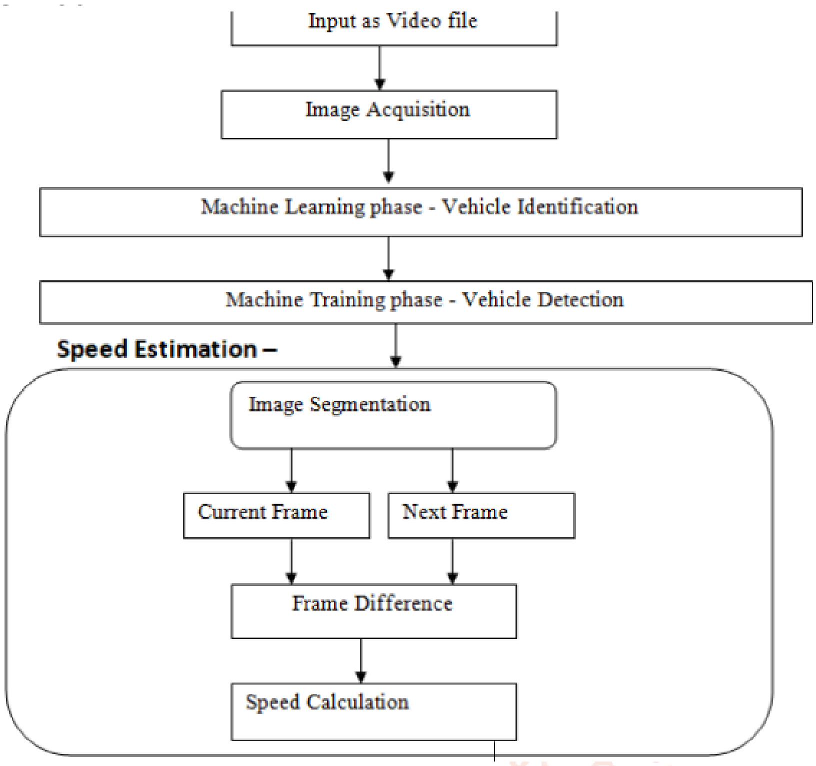


Figure 5.1 System Architecture of Vehicle Detection and Speed Tracking

5.2 INTRODUCTION

Software design sits at the technical kernel of the software engineering process and is applied regardless of the development paradigm and area of application. Design is the first step in the development phase for any engineered product or system. The designer's goal is to produce a model or representation of an entity that will later be built. Beginning, once system requirement has been specified and analysed, system design is the first of the three technical activities -design, code and test that is required to build and verify software.

The importance can be stated with a single word "Quality". Design is the place where quality is fostered in software development. Design provides us with representations of software that can assess for quality. Design is the only way that we can accurately translate a customer's view into a finished software product or system. Software design serves as a foundation for all the software engineering steps that follow. Without a strong design we risk building an unstable system – one that will be difficult to test, one whose quality cannot be assessed until the last stage. The purpose of the design phase is to plan a solution of the problem specified by the requirement document.

This phase is the first step in moving from the problem domain to the solution domain. In other words, starting with what is needed, design takes us toward how to satisfy the needs. The design of a system is perhaps the most critical factor affection the quality of the software; it has a major impact on the later phase, particularly testing, maintenance. The output of this phase is the design document. This document is similar to a blueprint for the solution and is used later during implementation, testing and maintenance. The design activity is often divided into two separate phases System Design and Detailed Design.

System Design also called top-level design aims to identify the modules that should be in the system, the specifications of these modules, and how they interact with each other to produce the desired results. At the end of the system design all the major data structures, file formats, output formats, and the major modules in the system and their specifications are decided.

During, Detailed Design, the internal logic of each of the modules specified in system design is decided. During this phase, the details of the data of a module is usually specified in a high-level design description language, which is independent of the target language in which the software will eventually be implemented.

In system design the focus is on identifying the modules, whereas during detailed design the focus is on designing the logic for each of the modules. In other works, in system design the attention is on what components are needed, while in detailed design how the components can be implemented in software is the issue.

Design is concerned with identifying software components specifying relationships among components. Specifying software structure and providing blue print for the document phase. Modularity is one of the desirable properties of large systems. It implies that the system is divided into several parts. In such a manner, the interaction between parts is minimal clearly specified.

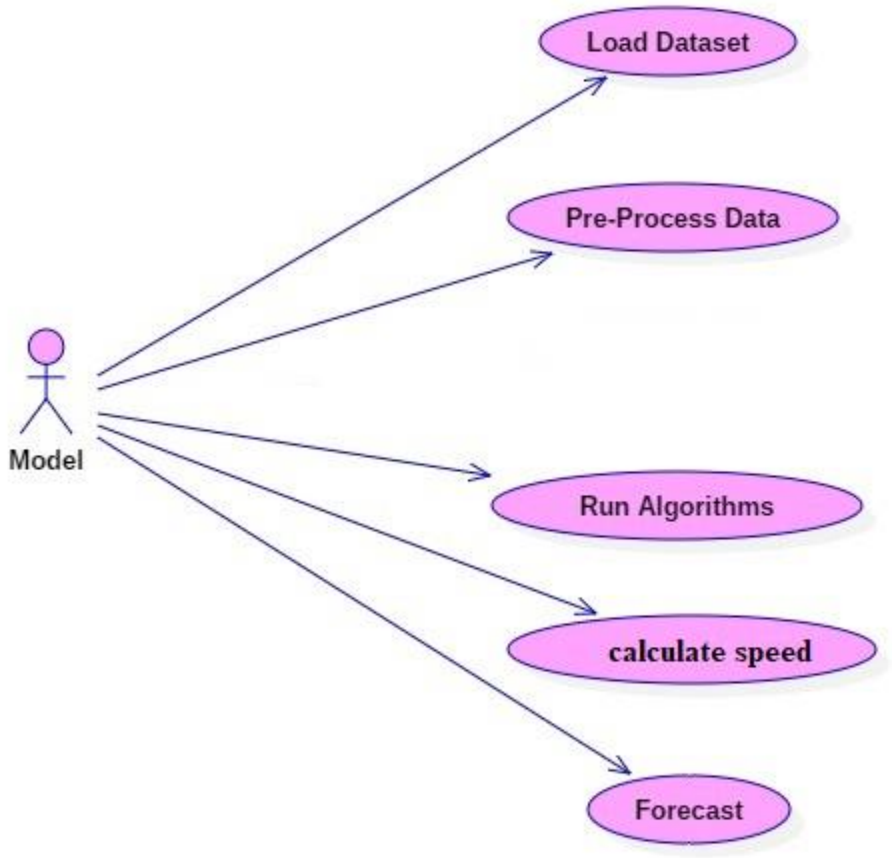
During the system design activities, Developers bridge the gap between the requirements specification, produced during requirements elicitation and analysis, and the system that is delivered to the user.

Design is the place where the quality is fostered in development. Software design is a process through which requirements are translated into a representation of software.

5.3 UML DIAGRAMS

5.3.1 USECASE DIAGRAM

A use case diagram is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.



•

Figure 5.2 Use Case Diagram for Vehicle Detection and Speed Tracking

5.3.2 SEQUENCE DIAGRAM

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

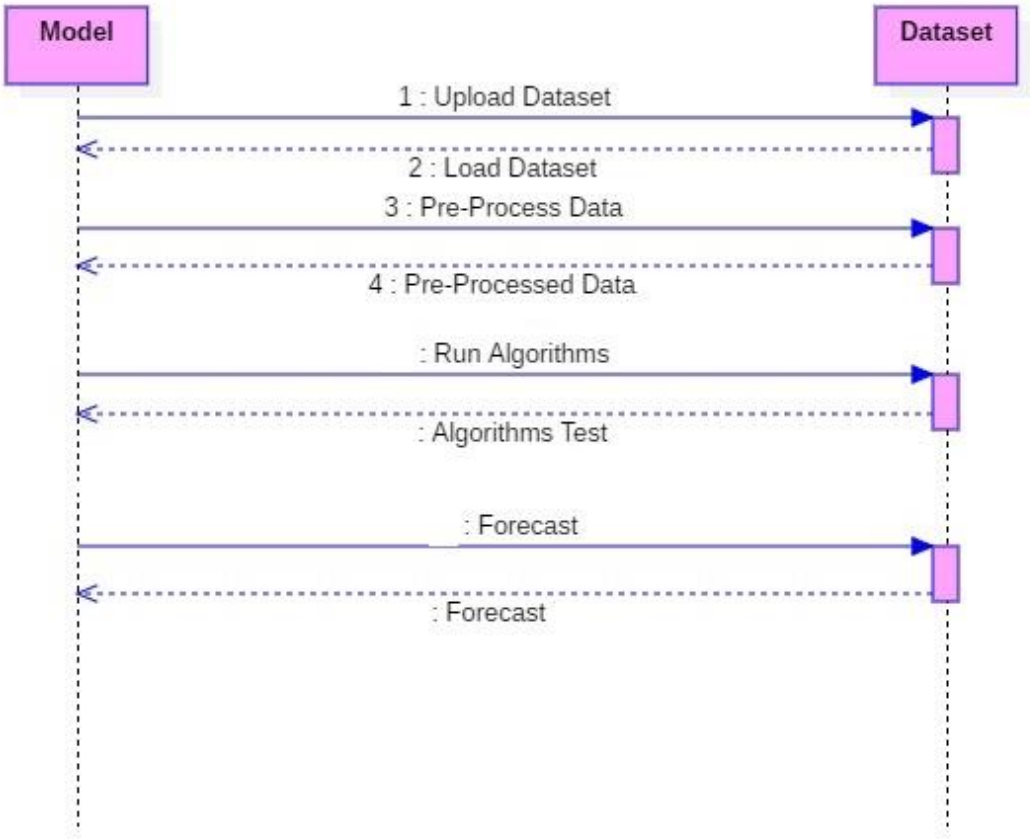


Figure 5.3 Sequence Diagram for Vehicle Detection

5.3.3 ACTIVITY D1AGRAM:

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modelling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

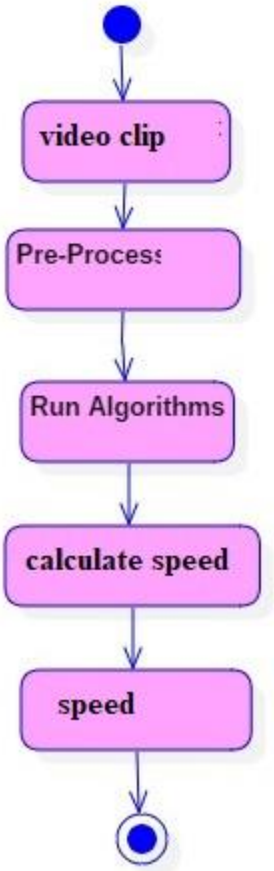


Figure 5.4 Activity Diagram for Vehicle Detection and Speed Tracking

5.3.4 CLASS DIAGRAM

A class diagram is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

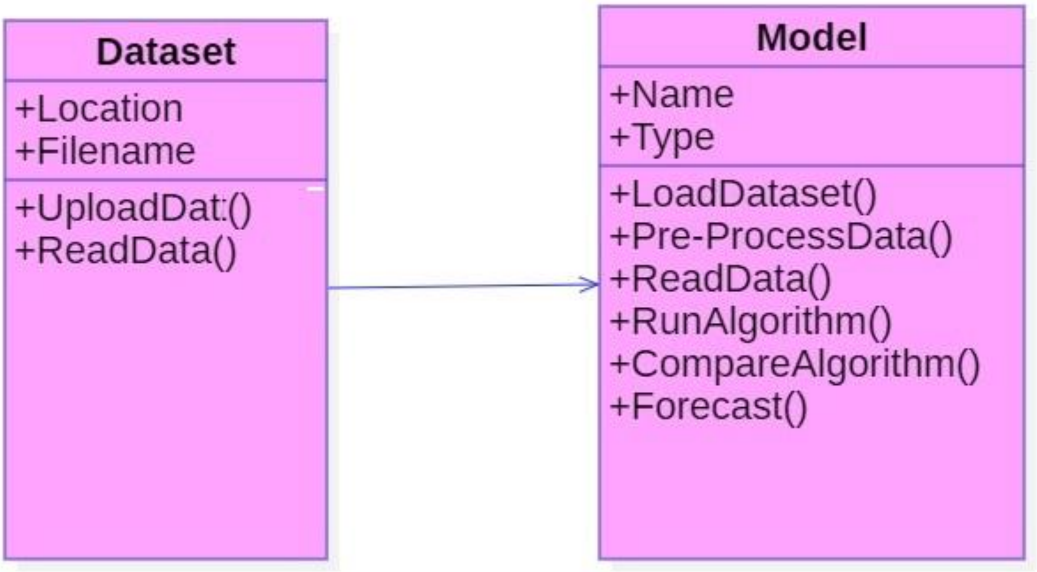


Figure 5.5 Class Diagram for Vehicle Detection

CHAPTER-6

IMPLEMENTATION

6.1 MODULES

- Vehicle Detection
- Speed Calculation

MODULES DESCRIPTION

6.1.1 Car Detection:

Object Location utilizing Haar highlight based course classifiers is a compelling item discovery strategy that uses a machine learning based approach where a course capacity is prepared from a considerable measure of positive and negative pictures. It is then used to recognize protests in different pictures.

• Initially, the calculation needs a considerable measure of positive (pictures of autos) and negative (pictures without autos) to prepare the classifier. At that point, we have to concentrate highlights from it. For this, haar highlights appeared in beneath picture are utilized. They are much the same as our convolutional part. Each component is a solitary esteem acquired by subtracting total of pixels under white rectangle from aggregate of pixels under dark rectangle.

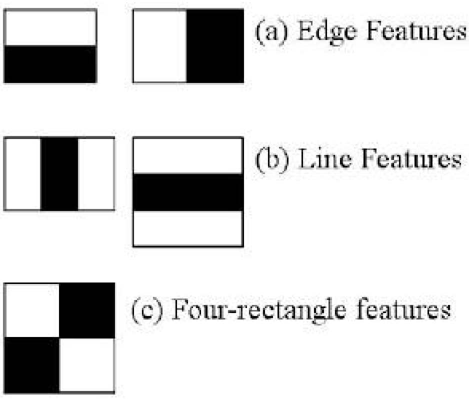


Figure 6.1 Haar Highlights

Now every single conceivable size and areas of every part is utilized to ascertain a lot of components. (Simply envision what amount of calculation it needs? Indeed, even a 24x24 window comes about more than 160000 components). For each component computation, we have to discover whole of pixels under white and dark rectangles. To tackle this, they presented the necessary pictures.

• Now, we apply each component on all the preparation pictures. For each component, it finds the best limit which will characterize the countenances to positive and negative. Be that as it may, clearly, there will be blunders or misclassifications. We select the elements with least mistake rate, which implies they are the elements that best orders the auto and non-auto pictures.

• So now you take a picture. Take each 24x24 window. Apply 6000 elements to it. Check on the off chance that it is auto or not.

6.1.2 Speed Calculation

• Once a car is detected, using the cascade Classifier () function on the haar cascade developed.

• Now the time is started which was initialized to 0.

• Using the ratio in the image for each cm travelled by the detected image and real-time distance in meters, the actual distance covered by the car is calculated.

• As soon as the car reaches the center of the detection window whose distance is already known to us the time is stopped.

• Now the actual distance calculated is divided by the time calculated and velocity is obtained.

• This velocity and the distance of the camera in feet from the car (i.e. the height of camera above the car) is printed on the output screen.

For this use multiple object detection algorithms could have been used but the algorithm of developing the Haar cascade and its implementation proves to be the best since it is the least time consuming, most efficient and highly reliable.

6.2 METHODOLOGY

6.2.1 Video Input and Image Acquisition

Videos taken from the surveillance camera are being used as the input. Nowadays in most of the all roads we could get the vehicles passed by videos from those cameras easily. The first stage of video processing is the computer vision system is the image acquisition stage. After the image has been obtained, number of methods of processing can be applied to the image to carry out the many different vision tasks that are required. This stage includes the operations such as Pre-processing, Back ground subtraction, Smoothing and shadow removal. However, if the image has not been acquired satisfactorily then the intended tasks needs to be iterated. Speed is a relative measure of the moving objects and these objects needs to be tracked in each frame. Hence, to calculate speed video images are broken into frames, then the object being travelled from one frame to another frame is traced out. The distance and time taken between these frames are calculated to estimate the speed of the object. A main advantage of a digital image processing, versus an analogue image, we may have number of copies and loss data of very less... Extracting series of images from a video one by one, then reading them using cv2 (Open Source Computer Vision) library.

6.2.2 Machine Learning:

Vehicle Identification Vehicle Identification refers to the ability of computer and software systems to trace objects in an image/scene and recognize each object. Object detection has plays a vital role in face detection, vehicle detection, pedestrian counting, web images, security systems and driverless cars. There are number of ways for object detection, can be used well in many fields of practice. Like every other fast growing computer technology, a wide range of creative and amazing uses of object detection. In this paper, object detection is done by using Machine learning algorithm called Haar cascade classifier.

Training Dataset

Collecting Data like Vehicle and Non Vehicle Images have collected thousands of images i.e. Positive Images – True images which needs to be identified by the system. Negative Images – False Images, which system needs to discard. In this Haar cascade classifier system gets learned by applying positive image on Negative Images. Vehicle detection has been done using Haar Cascade Classifier. Cascade Classifier and detect Multi Scale Train the classifier using 'training set', tune the parameters using 'validation set' and XML () file is the output for this process.

6.2.3 Tracking

Tracking is a process that gives an ID for the detected object to know the same object at next frame. This can be done using the correlation tracker in Dlibs library. It's being used to track the multiple objects at the same time and each ID monitors the object that is being shifted from one frame to another. While the object is in ROI, the object will be traced in every frame and system will calculate the distance travelled and then calculated the estimated speed between two consecutive frames. This estimated speed calculation is iterated for number of frame in ROI. So system keeps calculated the speed for number of frames and stored the internal value and then we can publish the average speed of the vehicle. When the vehicle is moving out of ROI, then particular Correlation ID removed from the tracking process. This is being done to avoid the multiple tracking process of the same vehicle. So the result of this phase is system is able to identify the vehicle and assign an ID to trace the path and distance that is being travelled between the frames. Finally, average speed of the vehicle can be obtained. Dlibs from python are used to track the Multi objects in the frame.

6.2.4 Speed Estimation

Speed estimation is done by calculate the distance travelled by the object for two consecutive frames. The conventional method of using Euclidean distance calculation is being used and then applies it to pixels per meter travelled by the vehicle. First of all, we need to have the WIDTH of the road in meters. Speed limit can be fixed by the width of the road. Hence I would like to consider the Width of the road as one of the factor for this speed calculation. To calculate the distance travelled by the object for two consecutive frames using Euclidean distance, let

Ct (a, b) and Ct+1(c, d) is centroid point of the object in frame t and t + 1 respectively. The distance d calculated by Euclidean distance is given below.

$$d = \sqrt{(a-c)^2 + (b-d)^2}$$

Then Pixels per meter is calculated as ppm. This value can be estimated manually for the particular road using video processing techniques. Therefore, values may be varying from Road to Road and this values needs to be adjusted. The actual width of the road is taken in similar way, each country follows some common value for the width and this can be obtained by doing the field study when the desired road's width needs to be taken.

From the above Euclidean calculation, d_pixels are the distance travelled by the object between frames. Hence for the standard conversion i.e. from converting Pixels to meter d_pixels need to converted to d-meters,

d-meters = d_pixels / ppm

So to calculate the final speed of the vehicle, in Km/hr we need to calculate the fps. This can be done by using video processing technique in pre-processing stage.

Speed = d_meters * fps*3.6

which gives the speed of the vehicle in km/hr. Each vehicle is identified by the correlation tracker and bounding box is applied to trace the moving vehicles. And speed of the vehicle is mentioned over the bounding box. Hence by seeing the output video, one can identify the speed of the vehicle.

6.3 TECHNOLOGY DESCRIPTION

6.3.1 Introduction of Python

Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. An interpreted language, Python has a design philosophy that emphasizes code readability (notably using whitespace indentation to delimit code blocks rather than curly brackets or keywords), and a syntax that allows programmers to express concepts in fewer lines of code than might be used in languages such as C++ or Java. It provides constructs that enable clear programming on both small and large scales. Python interpreters are available for many operating systems. CPython, the reference implementation of Python, is open source software and has a community-based development model, as do nearly all of its variant implementations. CPython is managed by the non-profit Python Software Foundation. Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library

6.3.2 What is Python

Python is a popular programming language. It was created by Guido van Rossum, and released in 1991.

6.3.3 Applications of Python

- web development (server-side),
- software development,
- mathematics,
- system scripting.

6.3.4 What can Python do

- Python can be used on a server to create web applications.
- Python can be used alongside software to create workflows.
- Python can connect to database systems. It can also read and modify files.
- Python can be used to handle big data and perform complex mathematics.
- Python can be used for rapid prototyping, or for production-ready software development.

6.3.5 Why Python

- Python works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc.).
- Python has a simple syntax similar to the English language.
- Python has syntax that allows developers to write programs with fewer lines than some other programming languages.
- Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick.
- Python can be treated in a procedural way, an object-orientated way or a functional way.

.

6.3.6 Python Syntax compared to other programming languages

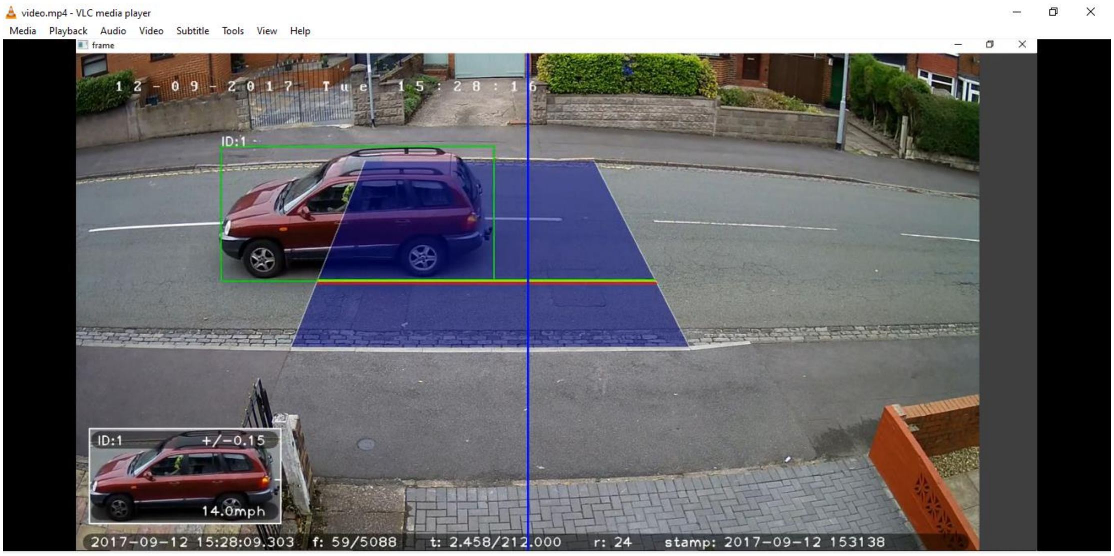
- Python was designed for readability, and has some similarities to the English language with influence from mathematics.
- Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses.
- Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly-brackets for this purpose.

6.4 CODING

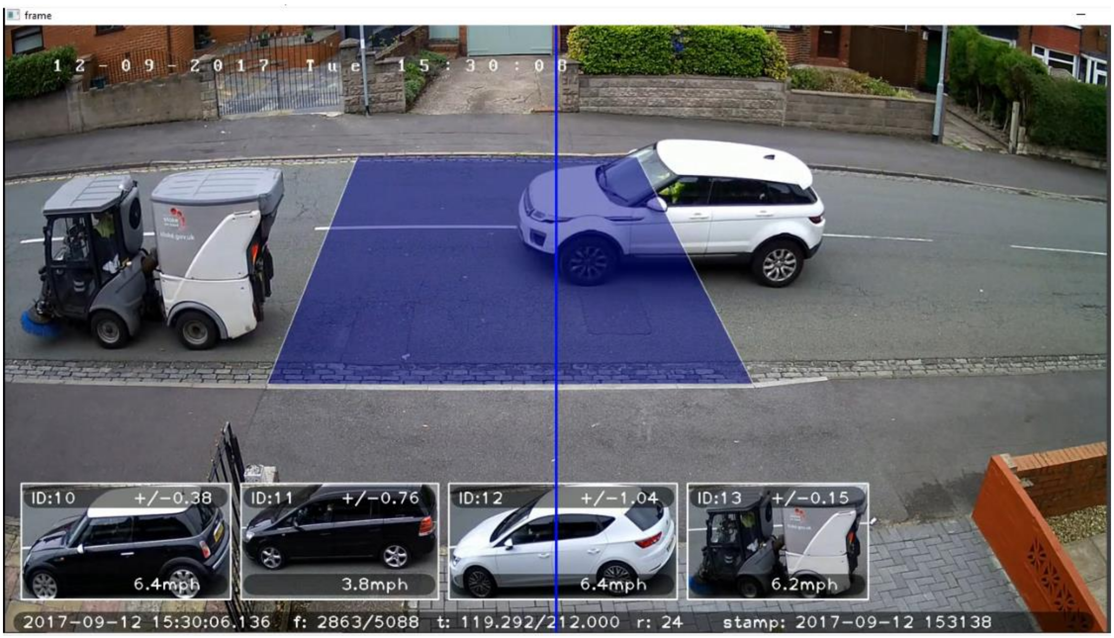
```
import numpy as np
import cv2
import time
car_cascade = cv2.CascadeClassifier('ha')
cap = cv2.VideoCapture('car.mp4')
wide=0.1   #depends upon size of car (~2.5)
flag=True
start=end=0
time_diff=0
while(cap.isOpened()):
    ret, img = cap.read()
    height,width,chan=img.shape
    #print(height,width,chan)
   cars = car_cascade.detectMultiScale(gray, 1.3, 5)
    #crp=gray[0:480,0:int(width/2) +10]
    for(x,y,w,h) in cars:
        cv2.rectangle(img, (x,y), (x+w,y+h), (0,255,0),2)
        #print(center_x,center_y)
        dist1=((wide*668.748634441)/w)
        #print ("Distance from car:",round(dist1,2),"m")
        roi_gray = gray[y:y+h,x:x+w]
        roi_color = img[y:y+h,x:x+w]
        dist0=((wide*668.748634441)/w)
        actual_dist=dist0*(width/2)/668.748634441
        #print ("Actual Distance:”, actual_dist)
        if flag is True and int(round(center_x)) in (range (0,80) or range (400,480)):
            start=time.time()
            flag=False
            #print("Start:",start)
```

```
        if flag is False and int(round(center_x)) in range(int(round(width/2))-10,
int(round(width/2)) +10):
            end=time.time()
            time_diff=end-start
            #print("End:",end)
            flag=True
            s_flag=True
    #print ("Time Difference:",time_diff)
    if time_diff>0 and s_flag==True:
        velocity=actual_dist/time_diff
        #print(round(start), round(end))
        vel_kmph=round(velocity*3.6,2)
        print ("Speed:", vel_kmph,"kmph")
        print ("Distance from car:",round(dist1,2),"m")
        s_flag=False
    cv2.line(img, (int(width/2),0), (int(width/2), height), (255,0,0),2)
    cv2.imshow('frame', img)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break
cap.release()
cv2.destroyAllWindows()
```

6.5 OUTPUT SCREEN



Screen 6.1 Screenshot of Vehicle Detection using Haar cascade classifier



Screen 6.2 Screenshot of Speed tracking

CHAPTER-7

SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

7.1 TYPES OF TESTS

7.1.1 Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

7.1.2 Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

7.1.3 Functional test

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centred on the following items:

- Valid Input: identified classes of valid input must be accepted.
- Invalid Input: identified classes of invalid input must be rejected.
- Functions: identified functions must be exercised.
- Output: identified classes of application outputs must be exercised.

Systems/Procedures:

Interfacing systems or procedures must be invoked. Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

7.1.4 System Test

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

7.1.5 White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

7.1.6 Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box. you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

7.1.7 Unit Testing

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

7.2 Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

7.2.1 Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

7.2.2 Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

7.3 Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one steps up – software applications at the company level – interact without error.

7.3.1 Test Results:

All the test cases mentioned above passed successfully. No defects encountered.

7.4 Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

7.4.1 Test Results:

All the test cases mentioned above passed successfully. No defects encountered.

CHAPTER-8

CONCLUSION AND FUTURE SCOPE

8.1 CONCLUSION

Road safety and reducing accidents is a very crucial issue and must be considered at utmost priority. Technological tools and tracking devices which help in monitoring the motion and speed of vehicles can help reduce the number of accidents on roads as well as trace the origins of the mishap. In this paper, we have discussed the challenges and obstacles faced while implementing a system which detects a vehicle and monitors its speed and motion. The separation of foreground and background objects are commonly preferred approaches to solve this issue. We have used openCV and haar cascade classifiers for object detection. Haar cascade is a approach based on machine learning where a cascade function is trained from a series of images which includes positive and negative. After the training it is used to detect objects in other images/videos. We have thus analysed various methods for speed tracking and vehicle detection.

8.2 FUTURE SCOPE

This can be further developed to identify all kinds of vehicles as well as to check anyone who breaks a traffic light.

Several nations are already using such systems to detect the speed and direction of vehicle. Moreover, some systems have advanced to the capacity of detecting the number plates of vehicles which are blurred for normal cameras and uses image processing algorithms to sharpen the image and extract the number plate which makes it even easier to locate the vehicle.

CHAPTER-9

REFERENCES

1. Zusheng Zhang, Tiezhu Zhao, Xin Ao, and Huaqiang Yuan A Vehicle Speed Estimation Algorithm Based on Dynamic Time Warping Approach IEEE SENSORS JOURNAL, VOL. 17, NO. 8, APRIL 15, 2017
2. Mahmoud Famouri, Zohreh Azimifar, Member, IEEE, and Alexander Wong, Senior Member, IEEE A Novel Motion Plane-Based Approach to Vehicle Speed Estimation IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS
3. W. Czajewski and M. Iwanowski, "Vision-based vehicle speed measurement method," in Computer Vision and Graphics. ICCVG (Lecture Notes in Computer Science), vol. 6374, L. Bolc, R. Tadeusiewicz, L. J. Chmielewski, and K. Wojciechowski, Eds. Berlin, Germany: Springer, 2010, pp. 308–315.
4. Lan, J. Li, G. Hu, B. Ran, and L. Wang, "Vehicle speed measurement based on gray constraint optical flow algorithm," Optik-International Journal for Light and Electron Optics, vol. 125, no. 1, pp. 289–295, 2014.
5. H. Sundoro and A. Harjoko, "Vehicle counting and vehicle speed measurement based on video processing," Journal of Theoretical and Applied Information Technology, vol. 84, no. 2, pp. 233–241, 2016.
6. A. G. Rad, A. Dehghani, and M. R. Karim, "Vehicle speed detection in video image sequences using cvs method," International Journal of Physical Sciences, vol. 5, no. 17, pp. 2555–2563, 2010.