

# **VEHICLE OVER SPEED DETECTION SYSTEM**

A Mini project Report Submitted to  
**Jawaharlal Nehru Technical University Hyderabad**

*In Partial Fulfillment of the Requirements for the Award of the Degree  
of*

**BACHELOR OF TECHNOLOGY  
IN  
COMPUTER SCIENCE AND ENGINEERING**

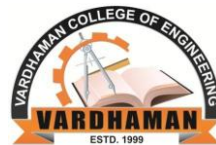
**Submitted by**

**Vivek Vardhan Asam    18881A0502**

**Sai Teja Sajja                18881A0553**

**SUPERVISOR**

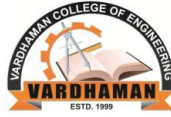
**Dr. H. Venkateswara Reddy  
Professor**



**Department of COMPUTER SCIENCE AND ENGINEERING  
VARDHAMAN COLLEGE OF ENGINEERING, HYDERABAD**

**An Autonomous Institute, Affiliated to JNTUH**

**July, 2021**



**VARDHAMAN COLLEGE OF ENGINEERING, HYDERABAD**

**An Autonomous Institute, Affiliated to JNTUH**

**Department of COMPUTER SCIENCE AND ENGINEERING**

## **CERTIFICATE**

This is to certify that the mini-project titled **VEHICLE OVER SPEED DETECTION SYSTEM** is carried out by

**Vivek Vardhan Asam    18881A0502**

**Sai Teja Sajja                18881A0553**

in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **COMPUTER SCIENCE AND ENGINEERING** during the year 2020-21.

**Signature of the Supervisor**  
**Dr. H. Venkateswara Reddy**  
**Professor**

**Signature of the HOD**  
**Dr. Rajanikanth Aluvalu**  
**Professor and Head, CSE**

## 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. H. Venkateswara Reddy**, Professor and Project Supervisor, Department of COMPUTER SCIENCE AND ENGINEERING, Vardhaman College of Engineering, for his able guidance and useful suggestions, which helped us in completing the mini-project in time.

We are particularly thankful to **Dr. Rajanikanth Aluvalu**, the Head of the Department, Department of COMPUTER SCIENCE AND ENGINEERING, his guidance, intense support and encouragement, which helped us to mould our mini-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 heartfelt thanks to **Dr. Teegala Vijender Reddy**, Chairman and **Sri Teegala Upender Reddy**, Secretary of VCE, for providing a congenial atmosphere to complete this mini-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.

**Vivek Vardhan Asam**

**Sai Teja Sajja**

## **Abstract**

The Main Idea of this project is to identify over speed vehicles, using Deep Learning and Machine Learning Algorithms. Now a days the accidents are Increasing day by day. Most of the accidents are happening due to over speed of the vehicles.Over speed Vehicles can be restricted by taking some actions based on the collected data of speed-limit crossed vehicles and actions like sending them a notice that you are exceeding the speed limit.Travel early reach safely,with this accidents and human loss can be handled.In this Project,Images of vehicles are extracted which are exceeding the speed-limit and provide the details of vehicles to the respected department to take actions against speed-limit exceeded vehicles.

# Table of Contents

Title	Page No.
<b>Acknowledgement</b> . . . . .	i
<b>Abstract</b> . . . . .	ii
<b>List of Figures</b> . . . . .	v
<b>Abbreviations</b> . . . . .	vi
<b>CHAPTER 1 Introduction</b> . . . . .	1
1.1 Motivation . . . . .	1
1.2 Problem Definition . . . . .	1
1.3 Objective of Project . . . . .	1
1.4 Limitations of Project . . . . .	2
<b>CHAPTER 2 Literature Survey</b> . . . . .	3
2.1 Introduction . . . . .	3
2.2 Existing System . . . . .	3
2.3 Disadvantages of Existing system . . . . .	3
2.4 Proposed System . . . . .	4
<b>CHAPTER 3 ANALYSIS</b> . . . . .	5
3.1 Software Requirement Specification . . . . .	5
3.1.1 Software requirements . . . . .	5
3.1.2 Hardware requirements . . . . .	5
3.2 Content diagram or Architecture of Project . . . . .	5
3.3 Algorithms and Flowcharts . . . . .	6
3.3.1 Image Acquisition . . . . .	6
3.3.2 Vehicles Detection . . . . .	6
3.3.3 Tracking the detected Vehicles using dlib library . . . . .	7
3.3.4 Speed Estimation . . . . .	7
<b>CHAPTER 4 DESIGN</b> . . . . .	9
4.1 Introduction . . . . .	9
4.2 DFD / ER / UML diagram (any other project diagrams) . . . . .	9
<b>CHAPTER 5 IMPLEMENTATION and RESULTS</b> . . . . .	11
5.1 Introduction . . . . .	11
5.2 Method of Implementation . . . . .	11

5.2.1	Output Screens and Result Analysis.....	11
<b>CHAPTER 6</b>	<b>TESTING.....</b>	<b>14</b>
6.1	Introduction .....	14
6.2	Design of test cases.....	14
6.2.1	Scenario – 1 . . . . .	14
6.2.2	Scenario – 2 . . . . .	14
6.2.3	Scenario – 1 . . . . .	14
6.2.4	Scenario – 2 . . . . .	14
<b>CHAPTER 7</b>	<b>CONCLUSION .....</b>	<b>16</b>
<b>REFERENCES</b>	<b>.....</b>	<b>17</b>

## **List of Figures**

<b>3.1</b>	<b>Basic Architecture Diagram of proposed system .....</b>	<b>6</b>
<b>3.2</b>	<b>Basic Flow Diagram .....</b>	<b>8</b>
<b>4.1</b>	<b>Use-Case diagram of proposed system.....</b>	<b>10</b>
<b>4.2</b>	<b>ER diagram of proposed system .....</b>	<b>10</b>
<b>5.1</b>	<b>Output Screen-1 .....</b>	<b>11</b>
<b>5.2</b>	<b>Output Screen-2 .....</b>	<b>12</b>
<b>5.3</b>	<b>Analysis Of Existing System Output (Arduino IOT Based) .....</b>	<b>12</b>
<b>5.4</b>	<b>Analysis Of Proposed System Output (OpenCV+ Dlib Based) .....</b>	<b>12</b>
<b>5.5</b>	<b>Barchart for Existing System vs Proposed System.....</b>	<b>13</b>
<b>5.6</b>	<b>Piechart of Number of accidents in 2018.....</b>	<b>13</b>
<b>6.1</b>	<b>Test-Cases.....</b>	<b>15</b>

## **Abbreviations**

<b>Abbreviation</b>	<b>Description</b>
VCE	Vardhaman College of Engineering
AI	Artificial Intelligence
OpenCV	Open Source Computer Vision Library
PC	Personal Computer
CPU	Central Processing Unit
RAM	Random Access Memory
OS	Operating System
XML	Extensible Markup Language
PPM	Pixels per meter
FPS	Frames per second
UML	Unified Modeling Language
ER MODEL	Entity-Relationship model
IOT	Internet of Things
IEEE	Institute of Electrical and Electronics Engineers



# **CHAPTER 1**

## **Introduction**

### **1.1 Motivation**

Machine learning is an application of artificial intelligence (AI) which allows computer programs to progressively learn and improvise from its experience with the data. It automates analytic by using algorithms that learn iterative, to make predictions. Its simple technique of self-learning rather than rule-based programming has found a wide application across multiple scenarios. So, this technology has pervaded everyday lives, whether bringing ease of living with navigation recommendations or warning you of market volatility for best investment decisions. Therefore, machine learning matters; as it shapes your ease of living or decision-making. It has integrated so deeply into daily life that you will most likely not notice its application.

### **1.2 Problem Definition**

Now a days the accidents are Increasing day by day. Most of the accidents are happening due to over-speed of the vehicles. If we restrict that over-speed Vehicles by taking some actions based on the collected data like sending them a notice that you are driving with over-speed instead of over-speed travel before starting time so that the accidents and Human 's loss will reduce.

The Main Goal of this project is to identify vehicles which is exceeding the described speed limit with the help of Machine Learning Algorithms and Python libraries.

### **1.3 Objective of Project**

i. To Detect the speed of the Vehicles and estimate the vehicle which is exceeding the speed limit.

ii. To Design an automatic and versatile system that addresses the issues of over speeding without use of human resources.

## **1.4 Limitations of Project**

The limitations of this project are:

- i. As we are using camera, there may be a chance of blur images.
- ii. As We are using the Machine learning algorithm and Python libraries there may be some errors or accuracy fault.

## **CHAPTER 2**

### **Literature Survey**

#### **2.1 Introduction**

The practical applications of machine learning drive business results which can dramatically affect a company's bottom line. New techniques in the field are evolving rapidly and expanded the application of machine learning to nearly limitless possibilities. Industries that depend on vast quantities of data—and need a system to analyze it efficiently and accurately, have embraced machine learning as the best way to build models, strategies, and plan.

With increasing technologies like Machine Learning Algorithms and Powerful Python Libraries, the usage of the hardware devices to estimate the speed of the vehicles is less. Because those technologies and libraries gives good accuracy than hardware devices as hardware devices have the drawback of Replacement of the parts which are defected. So, we are using these Technologies like Machine Learning Algorithms and Powerful Python Libraries like Open-CV, Dlib to estimate the Speed of the Vehicle.

#### **2.2 Existing System**

In the Existing System, Detection of speed is done by using the hardware tools like micro-controller systems, sensors and etc to calculate the speed of the vehicles.[1]

#### **2.3 Disadvantages of Existing system**

The Main Disadvantage of the Existing System is hardware may not work in Efficient manner as it requires replacements of the parts to function accordingly and also some of the hardware parts may not function up to the required functionality.[2]

## **2.4 Proposed System**

In this proposed system, we detect the speed of the vehicle and get the vehicles of images who is exceeding the speed limit as we design an automatic and versatile system that addresses the issues of over speeding without use of human resources.

## **CHAPTER 3**

### **ANALYSIS**

#### **3.1 Software Requirement Specification**

##### **3.1.1 Software requirements**

- i. Download IDLE python from this site: '<https://www.python.org/downloads/>'
- ii. Install OpenCV and dlib libraries from: '<https://www.learnopencv.com/install-dlib-on-windows/>'
- iii. Install these other libraries as well: NumPy, time, datetime, and os using pip in command line.

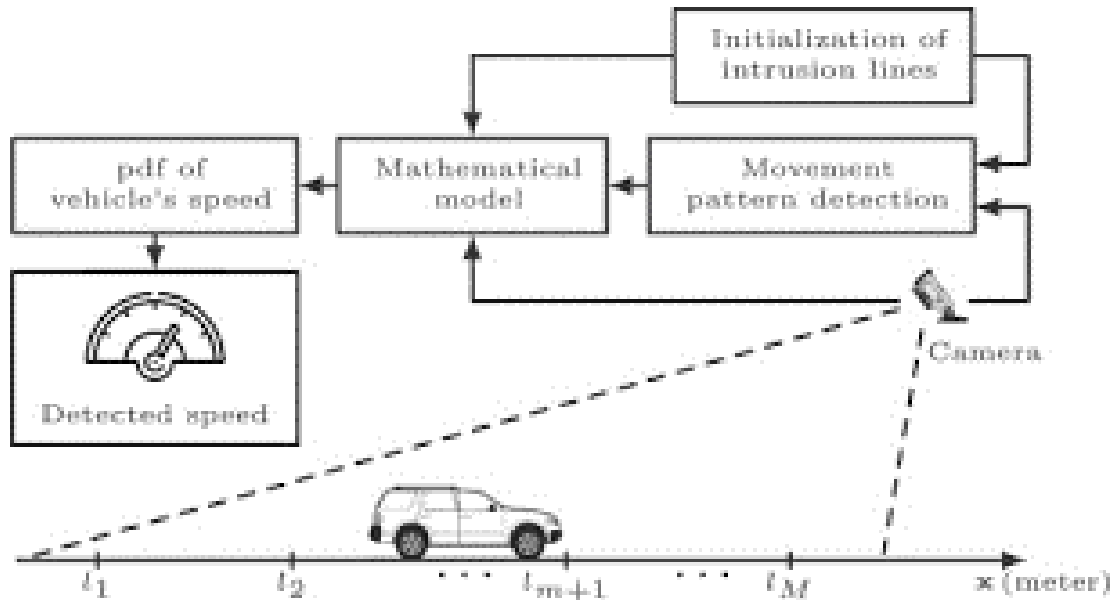
##### **3.1.2 Hardware requirements**

- i. Laptop/ Personal Computer (PC).
- ii. Random Access Memory (RAM): 4 GB or above.
- iii. Central Processing Unit (CPU): 1.7 GHz Processor and above.
- iv. Operating System (OS): Windows 8 and above.

#### **3.2 Content diagram or Architecture of Project**

- i. In this proposed system, we get the Images of vehicles those are exceeding the limit of speed and provide them to the required department to take actions on them.
- ii. Now a days the accidents are Increasing day by day. Most of the accidents are happening due to over-speed of the vehicles. If we restrict that over-speed Vehicles by taking some actions based on the collected data like sending them a notice that you are driving with over-speed instead of over-speed travel before starting time so that the accidents and Human 's loss will reduce.

iii. We detect the speed of the vehicle and get the vehicles of images who is exceeding the speed limit as we design an automatic and versatile system that addresses the issues of over speeding without use of human resources.



**Figure 3.1: Basic Architecture Diagram of proposed system**

### 3.3 Algorithms and Flowcharts

The following are the methods, algorithms and Flowcharts to obtain the output:

#### 3.3.1 Image Acquisition

Extracting series of images from a video one by one, then reading them using cv2 (Open-Source Computer Vision) library.[3]

#### 3.3.2 Vehicles Detection

After acquisition of series of images from the video, vehicles are detected using Haar Cascade Classifier.[4]

Haar Cascade Classifier [5] Algorithm includes 4 stages:

1. Haar Feature Selection.

2. Creating Integral Images. 3. Adaboost (Adaptive boosting) Training.
4. Implementing Cascading Classifiers.

Functions Used: Cascade Classifier and Multi-scale detection.

The model for the classifier is trained using lots of positive and negative images to make an XML file.

This is followed by tracking down the vehicles and estimating their speeds with the help of their respective locations, ppm (pixels per meter) and fps (frames per second).

### **3.3.3 Tracking the detected Vehicles using dlib library**

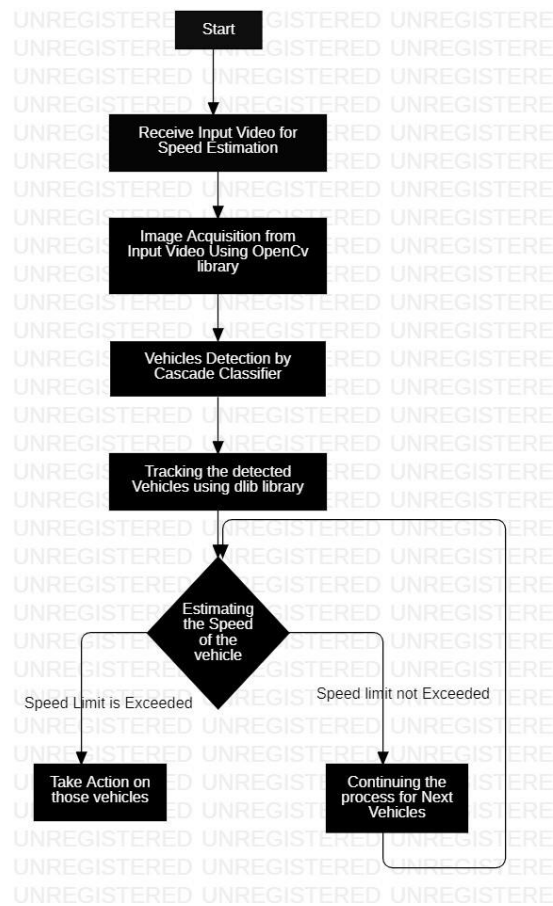
Functions Used:

- a. `dlib.correlation_tracker()`
- b. `dlib.correlation_tracker().start_track()`
- c. `dlib.rectangle`

### **3.3.4 Speed Estimation**

Speed (in km/hr) = Distance Travelled by the detected trucks (in meters)  
\* fps \* 3.6.[6]

Now, the cropped images of the identified Vehicles are stored in our device, so that we can take certain action on them.



**Figure 3.2: Basic Flow Diagram**



## **CHAPTER 4**

### **DESIGN**

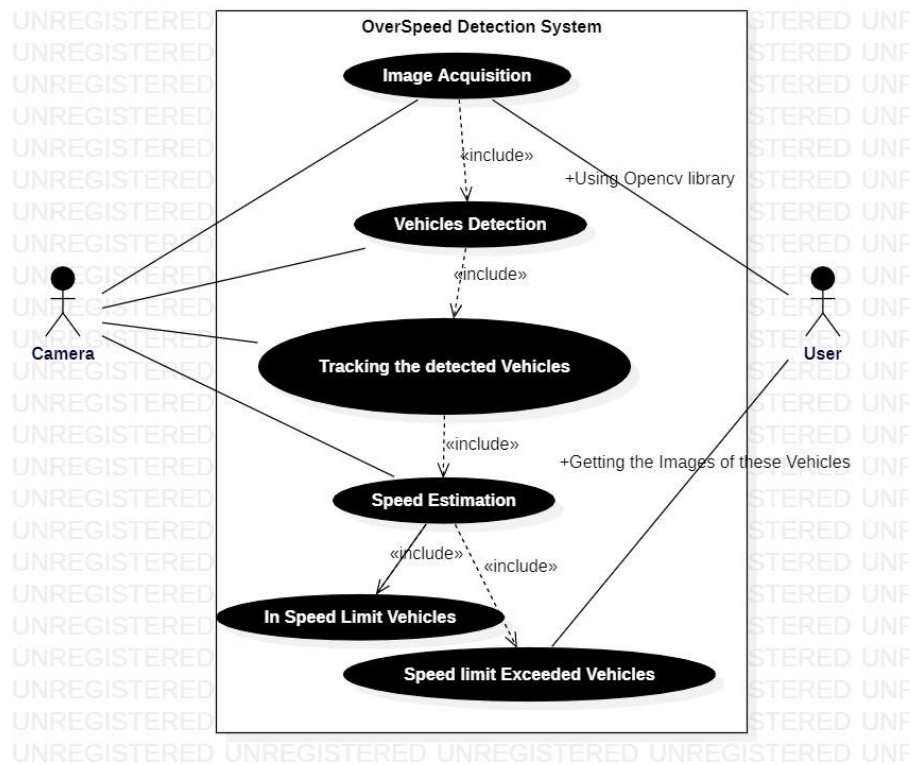
#### **4.1 Introduction**

Here in this proposed system, we have developed our project using python libraries and Machine learning algorithms to estimate the speed of the vehicle and detect the over speed vehicles. As Most of the accidents are happening due to over speed of the vehicles. If we restrict that over speed Vehicles by taking some actions based on the collected data like sending them a notice that you are driving with over speed instead of over speed travel before starting time so that the accidents and Human 's loss will reduce.

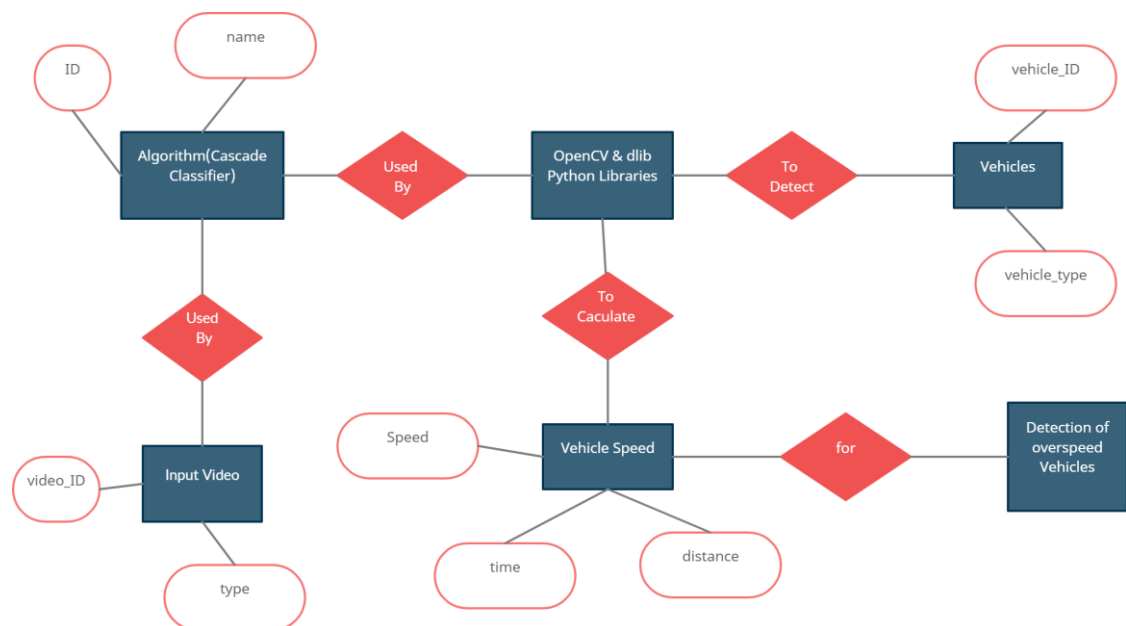
We get the Images of vehicles those are exceeding the limit of speed and provide them to the required department to take actions on them.

#### **4.2 DFD / ER / UML diagram (any other project diagrams)**

Below is the Use-Case diagram, ER diagram of our system as it helps to capture the functional requirements and easily traceable. Also helps to development guidelines to programmers, to a test case and finally into user documentation.



**Figure 4.1: Use-Case diagram of proposed system**



**Figure 4.2: ER diagram of proposed system**

# CHAPTER 5

## IMPLEMENTATION and RESULTS

### 5.1 Introduction

We have developed application, compatible for windows OS and implemented on different files varying from 1MB to 30 MB of file size to get the vehicles pictures from camera. We have developed a web application, compatible for any OS and can run on any platform to detect the vehicles which are exceeding the speed limit by using python libraries (Open CV and Dlib) and machine learning algorithm.

### 5.2 Method of Implementation

#### 5.2.1 Output Screens and Result Analysis

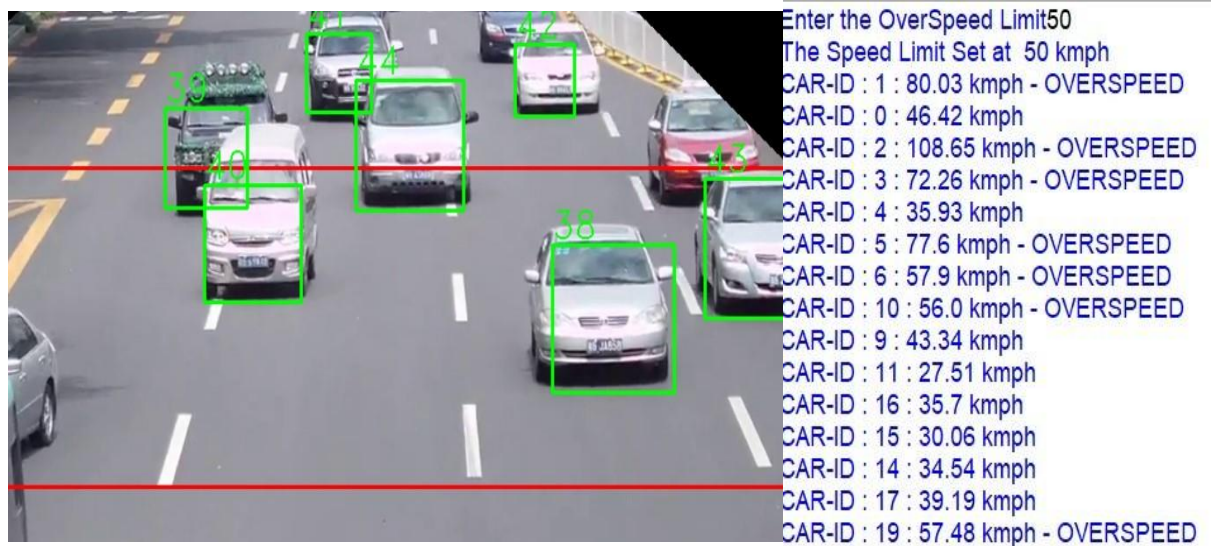
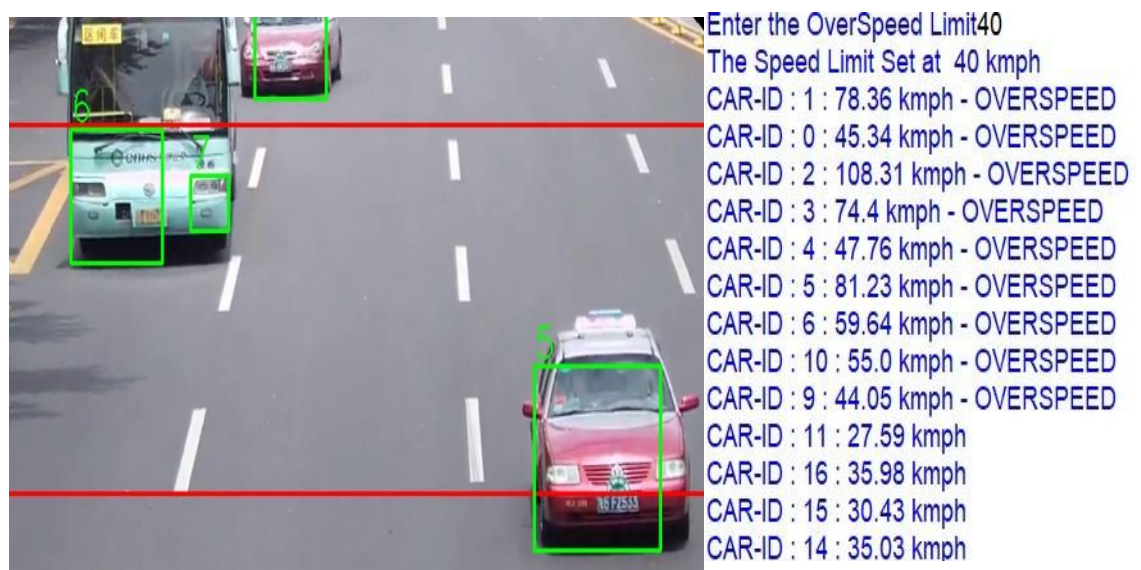


Figure 5.1: Output Screen-1



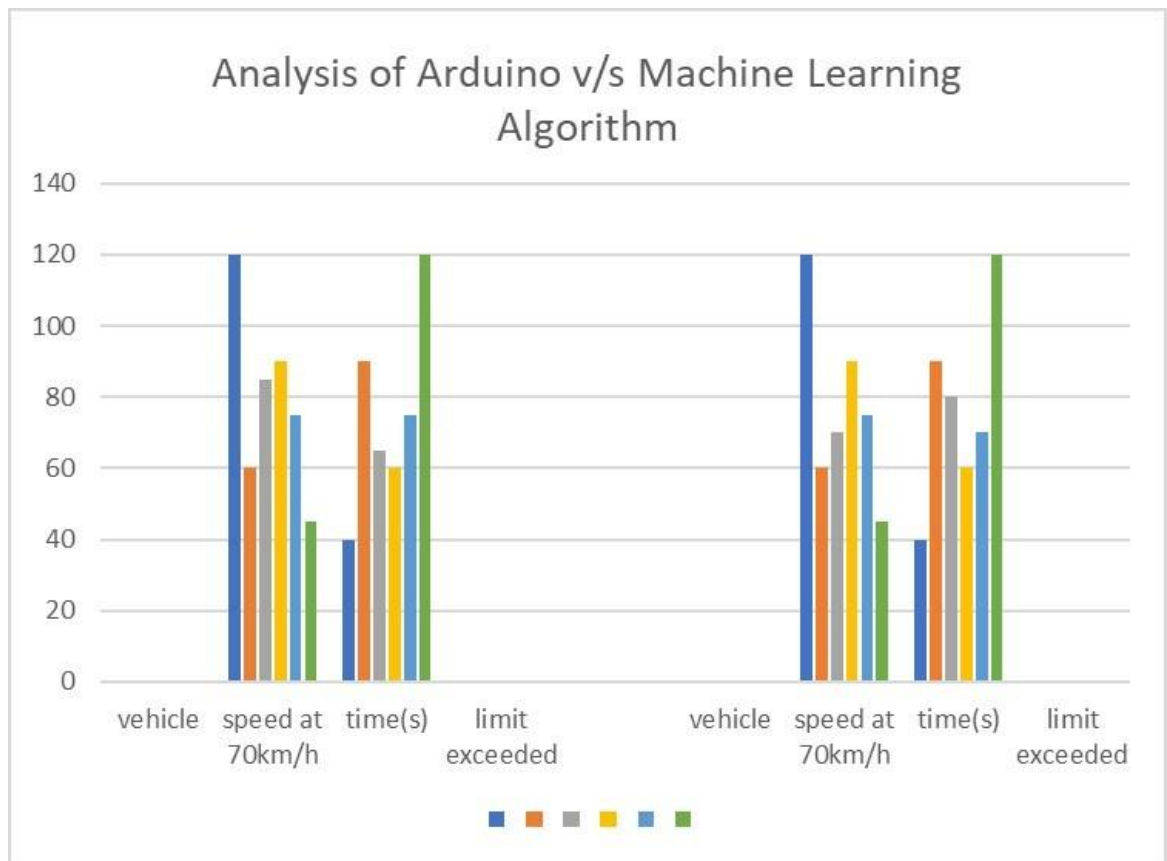
**Figure 5.2: Output Screen-2**

Vehicle	Speed at 70km/h	Time(s)	Limit exceeded
vehicle-1	120	40	yes
vehicle-2	60	90	no
vehicle-3	85	65	yes
vehicle-4	90	60	yes
vehicle-5	75	75	yes
vehicle-6	45	120	no

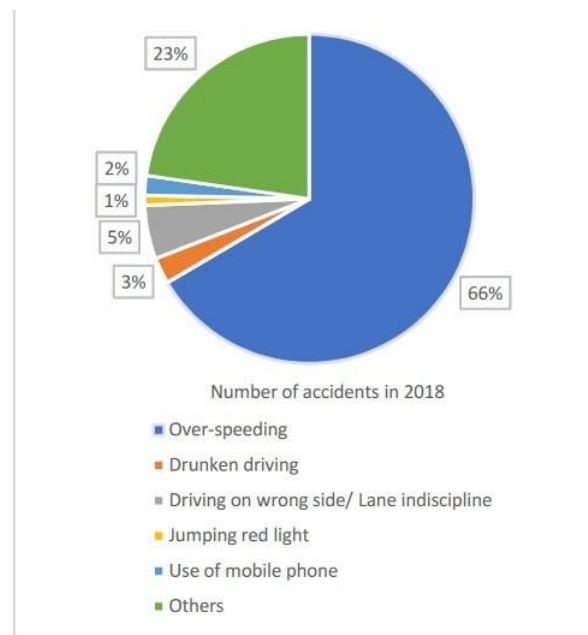
**Figure 5.3: Analysis Of Existing System Output (Arduino IOT Based)**

vehicle	Speed at 70km/h	Time(s)	Limit exceeded
vehicle-1	120	40	yes
vehicle-2	60	90	no
vehicle-3	70	80	no
vehicle-4	90	60	yes
vehicle-5	75	70	yes
vehicle-6	45	120	no

**Figure 5.4: Analysis Of Proposed System Output (OpenCV+ Dlib Based)**



**Figure 5.5: Barchart for Existing System vs Proposed System**



**Figure 5.6: Piechart of Number of accidents in 2018.**

# **CHAPTER 6**

## **TESTING**

### **6.1 Introduction**

In this testing phase, we need our application to run only when Camera is capturing the images of the vehicles and detect the Over-speed Vehicles and when the Camera is not capturing the images of the vehicles then detection of over-speed vehicles is not possible.

### **6.2 Design of test cases**

#### **6.2.1 Scenario – 1**

If the Camera Available for capturing the pictures, then pictures are captured.

#### **6.2.2 Scenario – 2**

If the Camera is not Available for capturing the pictures, then pictures are not captured.

#### **6.2.3 Scenario – 1**

After Capturing of the vehicle images if the images are clarity, then detection is possible and Over-speed vehicles are detected.

#### **6.2.4 Scenario – 2**

After Capturing of the Vehicles images if the images are blur, then detection is not possible and Over-speed vehicles are not detected.

PROJECT NAME	VEHICLE OVER SPEED DETECTION SYSTEM							
MODULE NAME	VEHICLE SPEED DETECTION							
TEST CASE ID	SYSTEM REQUIREMENT SPECIFICATIONS							
CREATED BY	VIVEK VARDHAN ASAM & SAITEJA SAJJJA							
DATE OF CREATION	02-06-2021							
DATE OF REVIEW	05-06-2021							
TEST CASE ID	TEST CASE SCENARIO	TEST CASE	PRE CONDITION	TEST STEPS	TEST DATA	EXPECTED RESULT	POST CONDITION	STATUS
CS_MP_01	Detection of Vehicles	Camera Availability	Camera With Good Resolution	Getting Images	Camera Working	Image of a Vehicle is Captured	Image processed for detection	PASS
CS_MP_02	Detection of Vehicles	Camera Availability	Camera With Good Resolution	Getting Images	Camera Not Working	Image of a Vehicle is not Captured	Image not processed for detection	FAIL
CS_MP_03	Detection of Vehicles	Camera Resolution	Camera With Good Resolution	Detecting Images with high resolution	Image of a Vehicle is Captured with clarity	Detection is perfectly done	Image processed for speed detection	PASS
CS_MP_04	Detection of Vehicles	Camera Resolution	Camera With Good Resolution	Detecting Images with medium resolution	Image of a Vehicle is Captured with some clarity	Detection is partially done	Image processing of vehicle is failed	FAIL
CS_MP_05	Detection of Vehicles	Camera Resolution	Camera With Good Resolution	Detecting Images with low resolution	Image of a Vehicle is not Captured with clarity	Detection is failed	Image processing of vehicle is failed	FAIL
CS_MP_06	Speed Estimation	Speed detection	Fixed Speed Limit	Comparing the Vehicle speed with limit	Vehicle speed is under limit	Vehicle Image is not Considered for next step	Certain Actions is not taken	PASS
CS_MP_07	Speed Estimation	Speed detection	Fixed Speed Limit	Comparing the Vehicle speed with limit	Vehicle speed exceeded limit	Vehicle Image is Considered for next step	Certain action is taken on vehicle	FAIL

**Figure 6.1: Test-Cases**

## **CHAPTER 7**

### **CONCLUSION**

Overall, By implementing this project accidents can be reduced with the help of powerful libraries (Open-CV and Dlib) in python to detect the vehicles and to estimate the speed of Vehicles. At last, by acquiring the speed of the vehicles restriction of the vehicles could be possible which are Exceeding the speed limit. Accidents can be reduced with some extent by taking preferable actions against those vehicles which are exceeding the speed-limit.



## REFERENCES

- [1] Sarmad Majeed Malik, M Asad Iqbal, Zohaib Hassan, Tauseef Tauqeer, Rehan Hafiz, and Usman Nasir. "Automated over speeding detection and reporting system". In: *2014 16th International Power Electronics and Motion Control Conference and Exposition*. IEEE. 2014, pp. 1104–1109.
- [2] Adi Nurhadiyatna, Benny Hardjono, Ari Wibisono, Wisnu Jatmiko, and Petrus Mursanto. "ITS information source: Vehicle speed measurement using camera as sensor". In: *2012 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*. IEEE. 2012, pp. 179–184.
- [3] Ahmad Mosayebi and Hossein Khosravi. "A Simple Approach for Real Time Speed Estimation of On Road Vehicles Using Video Sequences". In: *First International Conference in applied research on Electrical, Mechanical and Mechatronics Engineering*. 2016.
- [4] Olivier Barnich and Marc Van Droogenbroeck. "ViBe: A universal background subtraction algorithm for video sequences". In: *IEEE Transactions on Image processing* 20.6 (2010), pp. 1709–1724.
- [5] Adi Nurhadiyatna, Wisnu Jatmiko, Benny Hardjono, Ari Wibisono, Ibnu Sina, and Petrus Mursanto. "Background subtraction using gaussian mixture model enhanced by hole filling algorithm (gmmhf)". In: *2013 IEEE international conference on systems, man, and cybernetics*. IEEE. 2013, pp. 4006–4011.
- [6] Ali Tourani, Asadollah Shahbahrami, Alireza Akoushideh, Saeed Khazaee, and Ching Y Suen. "Motion-based vehicle speed measurement for intelligent transportation systems". In: *International Journal of Image, Graphics and Signal Processing* 11.4 (2019), p. 42.