D.K.T.E. Society’s Textile and Engineering Institute, Ichalkaranji.

(An Autonomous Institute, Affiliated to Shivaji University, Kolhapur)

Department of Information Technology

2019-2020

**Project-I Report On**

Smart Guideline and Braking System for Safety Driving

**Under the Guidance Of**

Prof. V. V. Kheradkar

**Submitted By:**

1. Sumit D Mhetre 19UIT652

2. Sandesh D Bhujavadkar 16UIT12063XX

3. Ruturaj V Lokhande 16UIT12028XX

4. Hanamant P Kattimani 16UIT12022XX

5. Pruthviraj S Patil 17UIT72010XX

|  |  |  |
| --- | --- | --- |
| Prof. V. V. Kheradkar | Prof. (Dr). D.V. Kodavade | Prof. (Dr). P. V. KADOLE |
| **(Project Guide)** | **(H.O.D)** | **(Director)** |



**YEAR 2019-2020**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**CERTIFICATE**

This is to certify that the project report entitles “Smart Guideline and Braking System for Safety Driving” is record of project work carried out in this college by,

1.Mr. Sumit D Mhetre (19UIT652)

2.Mr. Sandesh D Bhujavadkar (16UIT12063XX)

3.Mr. Ruturaj V Lokhande (16UIT12028XX)

4.Mr. Hanamant P Kattimani (16UIT12022XX)

5.Mr. Pruthviraj S Patil (17UIT72010XX)

in fulfillment of the requirement for degree of BACHELOR OF TECHNOLOGY in INFORMATION TECHNOLOGY of SHIVAJI UNIVERSITY, KOLHAPUR. This project report is record of their own work carried out under my supervision and guidance during academic year 2019-2020.

Prof. V. V. Kheradkar Prof. (Dr.) D. V. Kodavade

**[Project Guide]** [**Head of the Department]**

Prof. (Dr.) P.V. KADOLE

**[Director]**

**DECLARATION**

We the undersigned students of B. Tech Information Technology declare that, the field work report entitled **Smart Guideline and Braking System for Safety Driving**written and submitted under the guidance of Prof. V. V. Kheradkar is our original work. The empirical findings in this report are based on the data collected by us. The matter assimilated in this report is not reproduction from any readymade report.

Date:

Place: Ichalkaranji.

**Members** Signature

1.Mr. Sumit D Mhetre

2.Mr. Sandesh D Bhujavadkar

3.Mr. Ruturaj V Lokhande

4.Mr. Hanamant P Kattimani

5.Mr. Pruthviraj S Patil

**Acknowledgment**

With great pleasure we wish to express our deep sense of gratitude to prof. V. V. Kheradkar for his valuable guidelines, support and encouragement in completion of project report.

Also, we would like to take opportunity to thank our head of department Dr. D. V. Kodavade for his co-operation in preparing this project report.

We feel gratified to record our cordial thank to other staff members of Information Technology department for their support, help and assistance which they extended as and when required.

Thank you,

1.Mr. Sumit D Mhetre 19UIT652

2.Mr. Sandesh D Bhujavadkar 16UIT12063XX

3.Mr. Ruturaj V Lokhande 16UIT12028XX

4.Mr. Hanamant P Kattimani 16UIT12022XX

5.Mr. Pruthviraj S Patil 17UIT72010XX

**Abstract**

Now-a-days accidents are mostly caused by delay of the driver to hit the brake or by the

negligence by the driver. The project aims to develop a system which can operate automatically

with the help of high Ultrasonic sensors based on relay circuit and some changes in traditional

braking system and slow down the vehicle automatically in emergency situation.

The resulting system can achieve measurements with high accuracy and improved short

distance measurement also. This distance measurement is used to control smart braking system

for safety applications. The actual intelligence of system is developed on raspberry pi

microcontroller.

The Ultrasonic sensors are the eyes of this system, which are cheaper and the system comprises

of a less demanding hardware.

**INDEX**

**CONTENTS PAGE**

**1. INTRODUCTION**

1.1 Problem Statement

1.2 Need of the project with motivating example

1.3 Objective of Project

1.4 Limitations and Scope

1.5 Timeline for project

1.6 Cost of project

**2. BACKGROUND STUDY AND LITERATURE REVIEW**

2.1 Technology review

2.2 Literature review

**3. REQUIREMENT ANALYSIS**

3.1 Functional requirements

3.2 System Requirements

3.2.1 Hardware Requirements

3.2.2 Operating System Requirements

3.2.3 Application or Web server requirements

3.2.4 Storage requirement

3.2.5 Tools and Technologies Requirements

**4. SYSTEM DESIGN**

4.1 Architecture Diagram

4.2 Use case Diagram

4.3 Algorithmic description of each module

4.4 Data Flow Diagram

4.5 Sequence diagram

4.6 Class Diagram

4.7 Activity Diagram

**5. IMPLEMENTATION**

5.1 Environmental settings for running the module

5.2 Implementation Details

**6. SOFTWARE TESTING**

6.1 Unit test cases generation and its testing reports

6.2 Integration test cases generation and its testing reports

6.3 System test cases generation and its testing reports

**7. OUTPUT SCREEN**

**8. PERFORMANCE ANALYSIS**

**9. APPLICATION**

**10. CONCLUSION:**

**11. INSTALLATION GUIDE AND USER MANUAL**

**12. PROJECT ETHICS**

**13. REFERENCES**

**1. INTRODUCTION**

Nowadays, a vehicle is one of the most important need of human being. Every

time human needs vehicle to accomplish his work. But these vehicles are not smart

enough to handle the situation such as accident. So, these vehicles require some smart

moves. At Present, vehicles are often equipped with active safety systems to reduce the risk of accidents, many of which occur in the urban environments

So, this system measures the distance between two vehicles and if that distance

is less than the threshold range then it slows down the vehicle.

In recent years there is increase in computing power have brought computer vision to consumer-grade applications. As computers offer more and more processing power, the goal of real-time traffic sign detection and recognition is becoming feasible. Some new models of high-class vehicles already come equipped with driver assistance systems which offer automated detection and recognition of certain classes of traffic signs. Traffic sign detection and recognition is also becoming interesting in automated road maintenance.

**1.1 Problem Statement**

To implement the system which provide guidelines to the driver for

safe driving and slow down the speed of vehicle emergency situation.

**1.2 Need of the Project with motivating example**

As, traffic is increasing day-by-day across the globe, the accidents are happening

at high level. Always, driver is not responsible for accident instead, there is some kind of

failure that leads to the same. To avoid these accidents there is need of smart kind of

system that monitors all the time and guides the driver to do some necessary actions.

Volvo, Honda companies have such kind of system.

**1.3 Objectives of Project**

The objective of this project is to design the automatic braking system in order to

avoid the accident. To develop a safety vehicle braking system using ultrasonic sensor and

to design a vehicle with less human attention to the driving.

The objectives of the project are listened below:

* To detect an object in front of the vehicle.
* To calculate the distance between the object and the vehicle.
* To slow down vehicle to avoid an accident.
* To capture the road signs on besides of roads.
* To give guidelines according to captured road sign.

This project is necessary to be attached to every vehicle. Mainly it is used when

drive the vehicles in night time. Mostly the accident occurred in the night time due to long

travel the driver may get tired. So, the driver may hit the front vehicle or road side trees.

By using this project, the speed of vehicle get slowdown. So, we can avoid the accident.

**1.4 Limitations and Scope**

The scope of this project is to develop an ultrasonic sensor to detect the

obstacles and to process the output from the ultrasonic sensor to drive the servomotor

an actuator.

Vehicles can automatically brake due to obstacles when the sensor senses the

obstacles. The focus of this project is designing an automatic braking system that can

help driver to control the braking system of a vehicle. The automatically braking system also

needs to work with an ultrasonic sensor, which produces light by Arduino. The ultrasonic

wave is generated transmitter and sends to a Driver.

The accuracy of the guideline system is depending on training data. Less amount

of data gives less accuracy. And it is difficult to collect real-time datasets in huge

amounts. During capturing road signs camera sensors can loss little accuracy due any

environmental situations.

**1.5 Timeline for Project**

|  |  |  |
| --- | --- | --- |
| **TOPIC** | **START DATE** | **END DATE** |
| Domain Selection | 30/07/2019 | 06/08/2019 |
| Domain Finalization | 06/08/2019 | 13/08/2019 |
| Selection of Problem Statement | 13/08/2019 | 20/08/2019 |
| Finalization of Problem Statement | 20/08/2019 | 03/09/2019 |
| Study on Research Paper | 03/09/2019 | 07/09/2019 |
| Documentation of Synopsis | 07/09/2019 | 14/09/2019 |
| Requirement Analysis | 14/09/2019 | 17/09/2019 |
| System Requirement | 17/09/2019 | 21/09/2019 |
| Module Identification | 21/09/2019 | 24/09/2019 |
| System Architecture | 24/09/2019 | 29/09/2019 |
| Implementation 25% | 29/07/2019 | 12/10/2019 |
| Testing 25% | 12/10/2019 | 15/10/2019 |
| Implementation 50% | 07/01/2020 | 28/01/2020 |
| Testing 50% | 28/01/2020 | 31/01/2020 |
| Implementation 75% | 31/01/2020 | 14/02/2020 |
| Testing 75% | 14/02/2020 | 22/02/2020 |
| Implementation 100% | 22/02/2020 | 02/03/2020 |
| Testing 100% | 02/03/2020 | 10/03/2020 |
| Report Making | 10/03/2020 | 21/03/2020 |

**1.6 Cost of project:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Sr No. | Equipment |  | Unit |  | | Price (Rs) |
| 1. | Arduino |  | 1 |  | | 345 |
| 2. | Breadboard |  | 1 |  | | 70 |
| 3. | LED |  | 4 | |  | 10 |
| 4. | Jumping wire |  | 1 Packet | |  | 100 |
| 5. | 3-wheel Robo |  | 1 | |  | 400 |
| 6. | Battery |  | 4 | |  | 120 |
| 7. | Raspberry pi |  | 1 | |  | 5000 |
| 8. | Raspberry pi camera |  | 1 | |  | 1000 |
| 9. | HDMI cable |  | 1 | |  | 300 |
| 10. | Data cable |  | 1 | |  | 200 |
| 11. | Speaker |  | 1 | |  | 175 |
| 12. | Ultrasonic sensor |  | 1 | |  | 250 |
|  |  |  | **TOTAL** | |  | 7970 |

Estimated cost by considering other factors will be approx. **- Rs. 8000/-**

1.6.1 COCOMO Model

In this project, the Cost Estimation based on COCOMO (Constructive Cost Model) the formula for this Model is follows:

Effort = Constant × (Size) scale factor× Effort Multiplier

– Effort in terms of person-months

– Constant: 2.45 in 1998 based on Organic Mode

– Size: Estimated Size in KLOC

– Scale Factor: combined process factors

– Effort Multiplier (EM): combined effort factors

Functional Point Table

The function point range in between 1-10

Conversion of Functional point to Lines of Code (LOC)

Total function points = 6

■ Estimated Size – 300 LOC

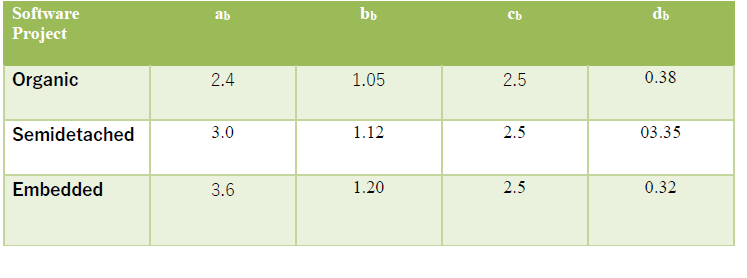
The basic COCOMO equations take the form

Effort Applied (E) = ab (KLOC) bb [man-months]

Development Time (D) = cb (Effort Applied) db [months]

People required (P) = Effort Applied / Development Time [count]

Where, KLOC is the estimated number of delivered lines (expressed in thousands) of code for project. The coefficients ab, bb, cb and db are given in the following table.



Organic Mode:

Effort Applied (E) = 2.5\*(300) \*1.15=862.5

Development Time (D) = 2.5\*(4) \*0.3= 3.0

People Required (P) = 862.5/3.0 = 3 people

**2. BACKGROUND STUDY AND LITERATURE REVIEW**

**2.1 Technology review**

In the beginning, the system implemented with Arduino UNO, and it’s going well. But there is some difficulty in between camera modules. So, in the next phase system implemented with Raspberry Pi. Now the whole system is comfortable with Raspberry Pi. We have a choice between distance sensors i.e. ultrasonic sensor and IR sensor. The system builds with an Ultrasonic sensor because it unaffected by Interference from light sources (e.g. sunlight, fluorescent tubes, etc.) and detection range is more than 1 meter, which is as compare to IR is very high (IR detects more appropriate an object closer than 10mm).

**2.2 Literature review**

We found some of the literature papers based on smart vehicles that uses some technologies and devices that aims to this project. Some of the literatures are as follows-

**2.2.1** Honda’s idea of ABS which helps the rider get hassle free braking experience

in muddy and watery surfaces by applying a distributed braking and prevents

skidding and wheel locking.

**2.2.2** Volvo launched XC60 SUV which was equipped with laser assisted braking.

This is capable to sense a collision up to 50 MPS and apply brakes automatically.

Drawbacks in the existing approaches:

**2.2.3** ABS can only help if the rider applies it in right time manually and maintains

the distance calculations. ABS have its own braking distance. Moreover, most of

the commuter bikes in India don’t have ABS because it’s very expensive.

**2.2.4** Volvo’s laser assisted braking could not work effectively in rainfall and

snowfall season and laser are easily affected by atmospheric conditions.

**3. REQUIREMENT ANALYSIS**

* 1. **Functional requirements**

**3.1.1** **Module: Object Detection**

  Input: Ultrasonic waves

  Output: Object Detected

  Description:

1. The Ultrasonic waves transmitter transmits ultrasonic waves.
2. if any object detected then ultrasonic wave sends back to the ultrasonic receiver receives.

**3.1.2** **Module: Distance Calculation**

  Input: Ultrasonic waves

  Output: Distance in cm.

  Description:

1. The module will take duration (time) between time from which ultrasonic transmitter transmits waves to ultrasonic receiver receives waves.
2. That duration (time) then divided by 58.2 to calculate the distance.

**3.1.3** **Module: Brake system**

  Input: Distance in cm

  Output: slowdown vehicle

  Description:

1. Inputted distance compared to decided threshold
2. If distance is less as compared to standard distance then call braking system

**3.1.4** **Module: Collect data for guideline.**

  Input: R Camera Module

  Output: Collect image data

  Description:

1. Pi-Camera are placed in front of vehicle or on top of vehicle.
2. Pi-Camera are continuously capturing images and forwarded to system.

**3.1.5** **Module: Generate alert signal.**

  Input: Collected data as images from module ‘Collect data for guideline’

  Output: Generate alert signal according to single

  Description:

1. Inputted image compared with trained models
2. If get matched with any model then generate alert according to it.
3. Else give ‘Clear Road’ alert.

**3.2 System Requirements**

**3.2.1 Hardware Requirements**

* Raspberry Pi Camera Module
* Ultrasonic sensor
* Raspberry Pi 3 Model B
* ROM 500 MB
* RAM 8 GB (Intel Core i7-10700T 10th GEN)

**3.2.2 Operating System Requirements**

* Windows 10
* Raspbian

**3.2.3 Application or Tools and Technologies**

* Python 2.7
* Raspberry Pi IDE- Thonny
* Libraries -OpenCV
* Fatkun extension - For collecting Dataset
* Cascade-Trainer GUI – For training the dataset

**3.2.4 Storage requirement**

* 2 GB memory for storage purpose (dataset + code)
* Dataset: 1.5 GB
* 4 GB RAM/2GB graphics for the dataset training purpose.

**4**. **SYSTEM DESIGN**

**4.1 Architecture Diagram**

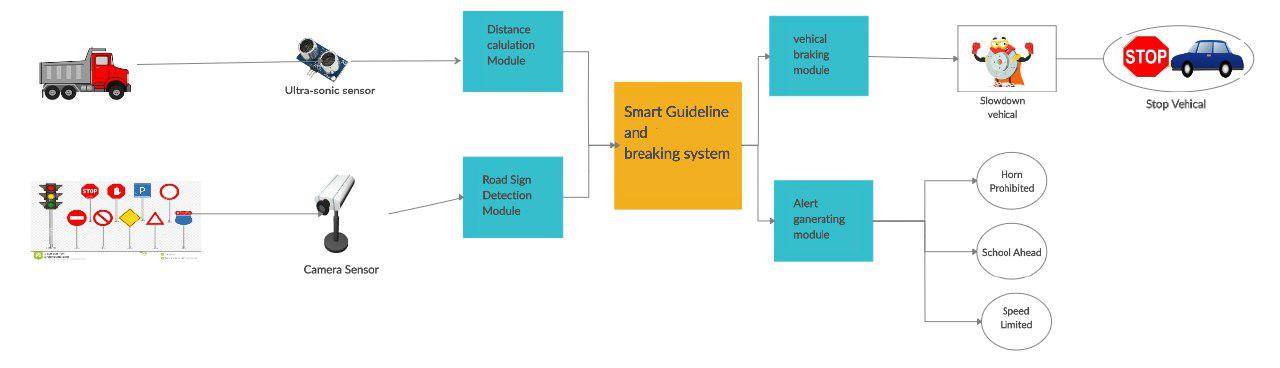


Fig 4.1 Architecture Diagram of System

This Architecture Diagram elaborate overall structure of system. The main objectives of system are to avoid and decrease road accidents. A vehicle braking module and signal guideline module are helping to achieve objective.

**4.1.1 Components in Architecture diagram**

In Architecture Diagram, A four modules of system are presented with required component.

A vehicle, an ultra-sonic sensor, cameras etc. components are required.

1)Vehicle - A vehicle could be any four-wheeler or two-wheeler.

2)Ultra-sonic sensors - An ultrasonic sensor is an instrument that measures the distance to an object using ultrasonic sound waves.

An ultrasonic sensor uses a transducer to send and receive ultrasonic pulses that relay back information about an object’s proximity.

Ultrasonic sensors work by sending out a sound wave at a frequency above the range of human hearing.

The transducer of the sensor acts as a microphone to receive and send the ultrasonic sound.

The sensor determines the distance to a target by measuring time lapses between the sending and receiving of the ultrasonic pulse.

3)Camera Sensor - A camera sensor uses an array of millions of tiny light cavities or "photosets" to record an image.

A camera sensor is a device that allows the camera to convert photons i.e. light into electrical signals that can be interpreted by the device.

**4.2 Use Case Diagram**

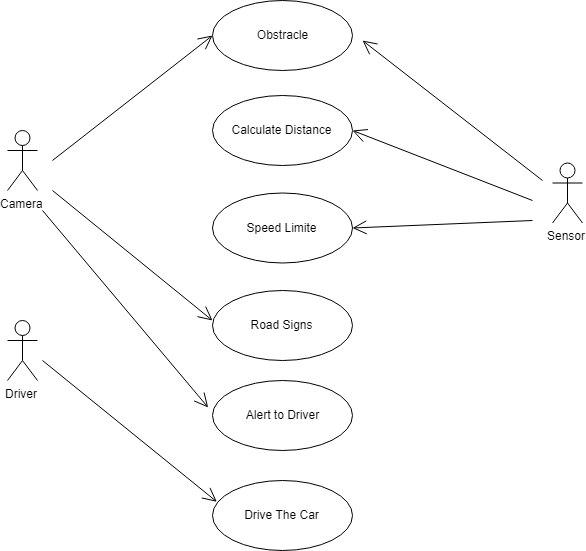


Fig 4.2 Use Case Diagram

Above diagram shows main functionalities are shown.

In diagram three users are shown i.e. Camera, Driver and Sensor. Out of those camera and sensor are related to each other.

**4.3 Algorithmic description of each module: -**

1. **Module 1: - Object detection** 
   1. Start
   2. Transfer ultrasonic waves from transformer of sensor
   3. If path of ultrasonic wave is broken by object
   4. Then ultrasonic wave sends back to the receiver.
   5. End
2. **Module 2: - Distance calculation**
   1. Start
   2. Calculate duration of time from transformation state of wave to receiving state of wave.
   3. Duration (time) divided by 58.2
   4. End
3. **Module 3: - Vehicle brake system** 
   1. Start
   2. If distance less than threshold value
   3. Then slowdown vehicle
   4. Else clear road for vehicle
   5. End
4. **Module 4: - Collect image data of signal**
   1. Start
   2. Capture every image in front of camera
   3. Send image to alert generating module
   4. End
5. **Module 5: - Generating alert generator** 
   1. Start
   2. Receive image
   3. Compare with trained model
   4. If matched with trained model
   5. Then generate alert according to it
   6. End

**4.4 Data Flow Diagram**

**4.4.1 DFD level 0**

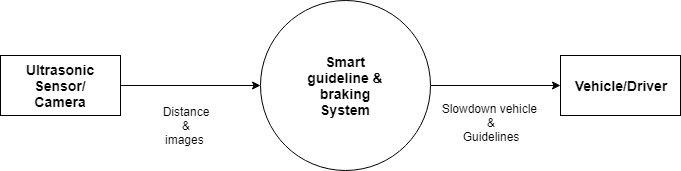


Fig 4.4.1 Diagram of DFD Level 0

Above diagram shows first the sensor calculates the distance and pi

camera captures the image and transfer that data which is input to system which

then guides the driver/vehicle.

**4.4.2 DFD level 1**

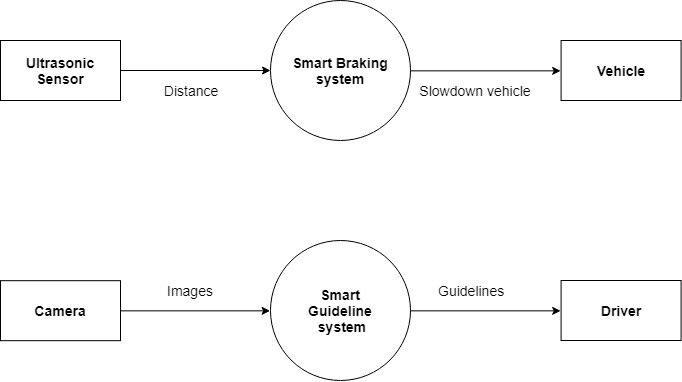


Fig 4.4.2 Diagram of DFD level 1

As shown in above the specific functionality of each module is

given which performs their respective tasks.

**4.4.3 DFD level 2**

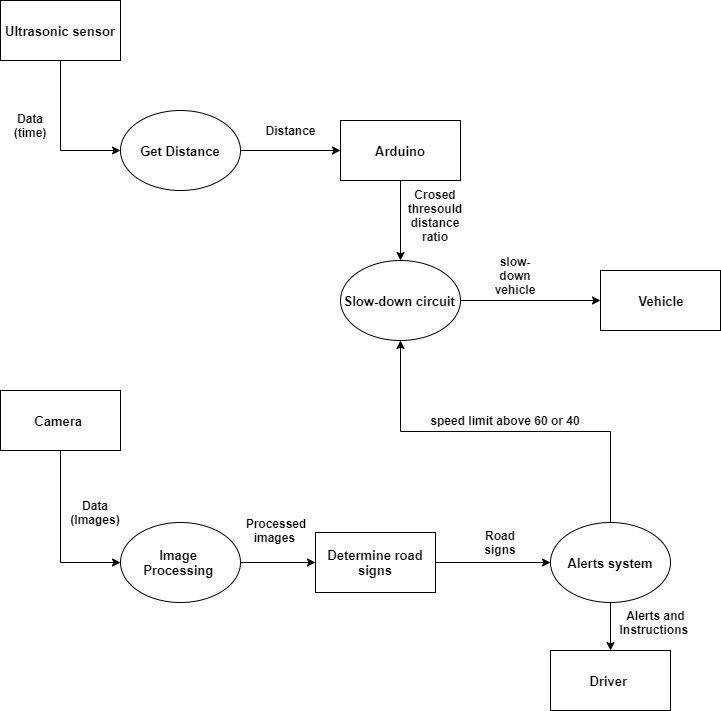


Fig 4.4.3 Diagram of DFD level 2

As shown in diagram the specific functionality of each module is

given which performs their respective tasks.

**4.5 Sequence diagram**

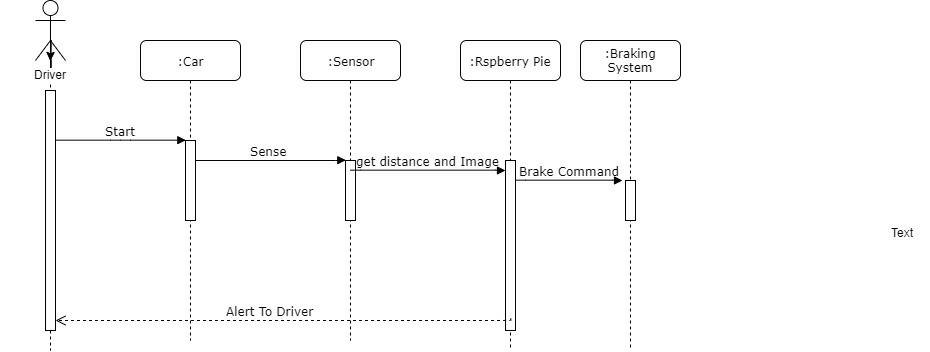


Fig 4.5 Sequence Diagram

As shown in above diagram the sequence of several activities is

shown, which are carried out in specific manner.

**4.6 Class Diagram**

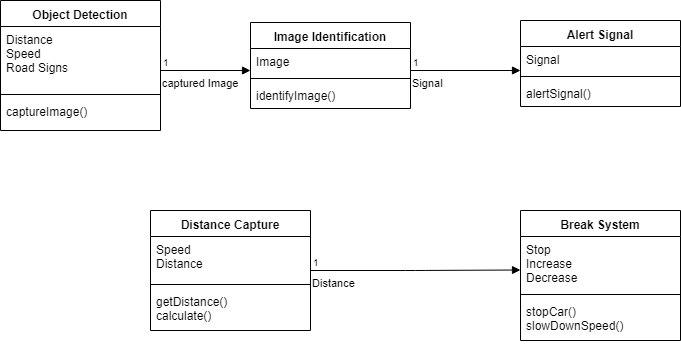


Fig 4.6.1 Class Diagram

Above diagram elaborate the class structure of each module of system.

The data members and functionality of each module is described.

**5. IMPLEMENTATION**

* 1. Environmental settings for running the module
* Install Raspbian OS on Raspberry Pi.
* Install Thonny IDE for Raspberry Pi.
* Install required all libraries.
* Install Cascade Trainer-GUI tool on training machine
* Add extension to Chrome extension named as Fatkun Batch.
  1. Implementation Details
     1. Brake System: -
* Mount Ultrasonic sensor on Vehicle-Model and connect to distance calculation module.
  + Logic: The ultrasonic sensor connected to vehicle will send and receive waves and forward it into the transferable form to the distance calculation module.
* Distance calculation, object detection module connects to braking system
  + Logic – Depends on the speed of sound the obtained waves(pulse) are divided by certain value to calculate the exact distance.
* ‘Robo chassis’ helps brake system to slowdown the vehicle.
  + Logic: The calculated distance will be forwarded to the braking system module where depending on distance braking system will react.
    1. Alert Generating system: -
* Mount Pi camera on Vehicle-Model
  + Logic- To capture the road signs pi camera is attached to the vehicle which will help to capture some alert, warning and some precaution signals.
* Collect dataset of signals and Train dataset by using Cascade Trainer-GUI tool
  + Logic - To achieve some good accuracy in result the real time data is being trained which will help to react quick and more accurate.
* Import that trained model in program.
  + Logic - The trained data is imported in program to test it on some alert, warning signals.
* Generate program for generating alert signals.
  + Logic - Finally it will generate some alert or give some instructions accordingly.

**6. SOFTWARE TESTING**

**6.1 Unit test cases generation and its testing reports**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Test  case No |  | Test case |  | Input | Expected  Output |  | Actual Output |  | Status |
| 01 |  | Ultrasonic wave transfer |  | Ultrasonic wave | Ultrasonic wave transfers successfully |  | Ultrasonic wave transfers successfully |  | Pass |
| 02 |  | Ultrasonic wave receives |  | Ultrasonic wave | Ultrasonic wave receives successfully |  | Ultrasonic wave receives successfully |  | Pass |
| 03 |  | Calculate time between transfer and receive wave |  | Ultrasonic wave | Time between wave transfer and wave receive calculated successfully |  | Time between wave transfer and wave receive calculated successfully |  | Pass |
| 04 |  | Capture image |  | Camera module | Capture image successfully |  | Capture image successfully |  | Pass |
| 05 |  | Call vehicle stop module |  | Call  to module | Vehicle stop module working successfully |  | Vehicle stop module working successfully |  | Pass |
|  |  |  |  |  |  |  |  |  |  |

**6.2 Integration test cases generation and its testing reports**

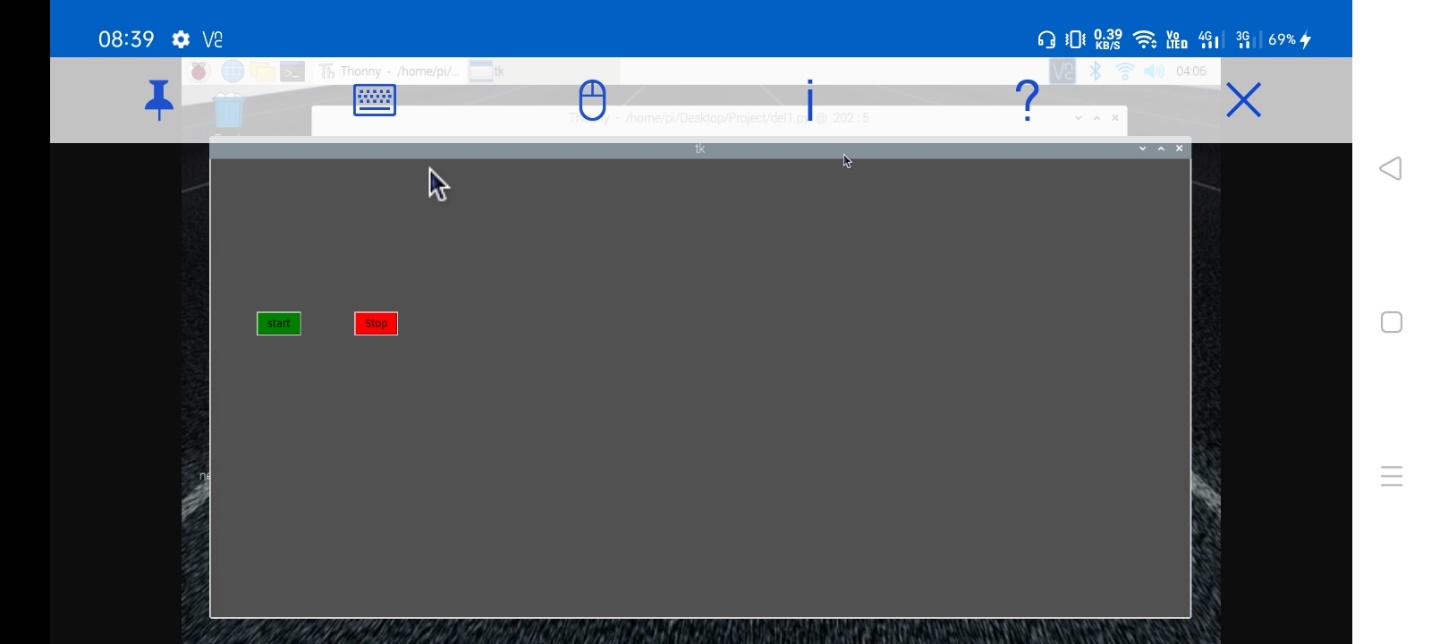
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test case No | Test case | Input | Expected  Output | Actual Output | Status |
| 01 | Object detection (within threshold) | Ultrasonic wave | Object detected  Successfully | Object detected  Successfully | Pass |
| 02 | Object detection  (not within threshold) | Ultrasonic wave | Object does not detect  Successfully | Object does not detect  Successfully | Pass |
| 03 | Distance calculation | Ultrasonic wave | Distance calculated successfully | Distance calculated successfully | Pass |
| 04 | Check captured image has traffic signal image or not. | Captured image | Image has traffic signal. | Image has traffic signal. | Pass |
|  |  |  |  |  |  |

**6.3 System test cases generation and its testing reports**

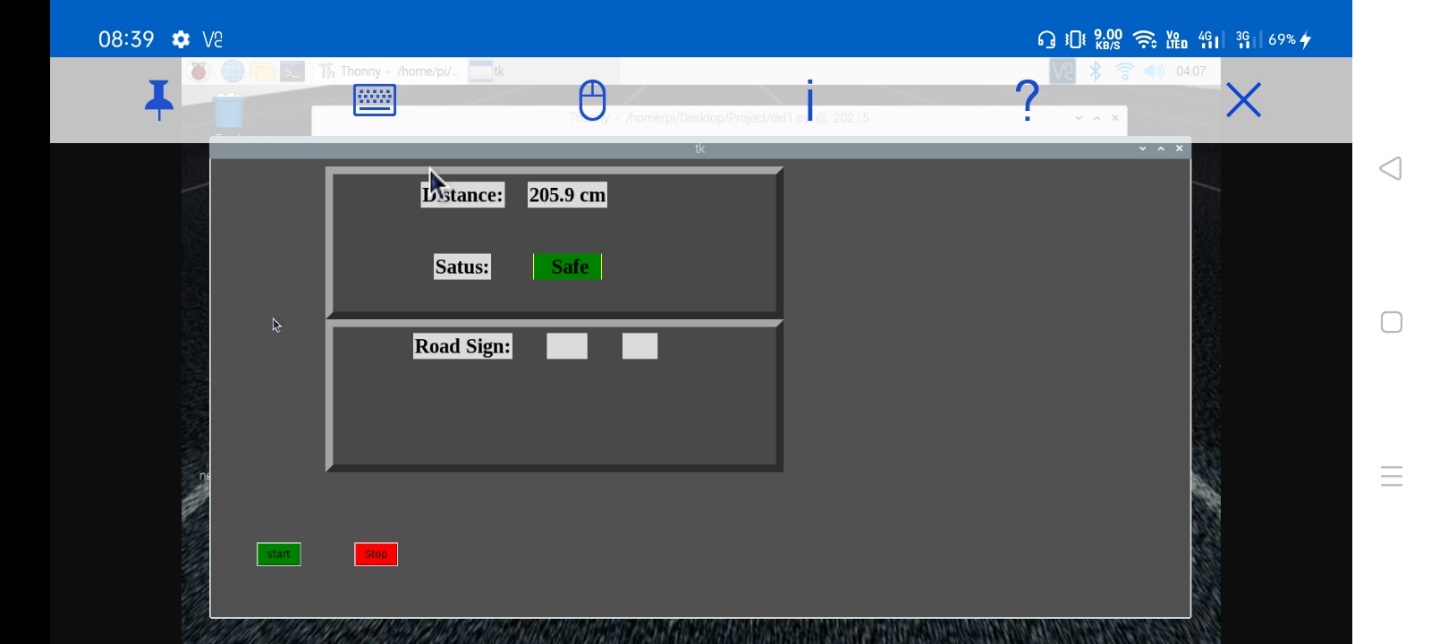
|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Test case No | Test case |  | Input |  | Expected  Output |  | Actual Output |  | Status |
| 01 | if object detected, Stop vehicle |  | Ultrasonic wave, call to stop vehicle module |  | Vehicle stopped successfully |  | Vehicle stopped successfully |  | Pass |
| 02 | if object not detected,  go ahead with vehicle |  | Ultrasonic wave |  | Vehicle not stopped |  | Vehicle not stopped |  | Pass |
| 03 | Detect traffic signal from inputted image |  | Captured image |  | Traffic signal detected successfully |  | Traffic signal detected successfully |  | Pass |
| 04 | if traffic signal detected,  Generate alert for traffic signal |  | Captured image |  | Generate alert for detected traffic signal |  | Generate alert for detected traffic signal |  | Pass |
| 05 | if traffic signal not detected,  does not generate alert for traffic signal |  | Captured image |  | Dose not generate any alert for traffic signal |  | Dose not generate any alert for traffic signal |  | Pass |
| 06 | Detected traffic sign: - stop and generate alert for stop |  | Captured image |  | Generate alert for ‘stop’ signal successfully |  | Generate alert for ‘stop’ signal successfully |  | Pass |
| 07 | If two traffic signals detected: -  Generate alerts for both signal |  | Captured image |  | Generate alerts for both traffic signals  successfully |  | Generate alerts for both traffic signals  successfully |  | Pass |

**7. Output Screen**

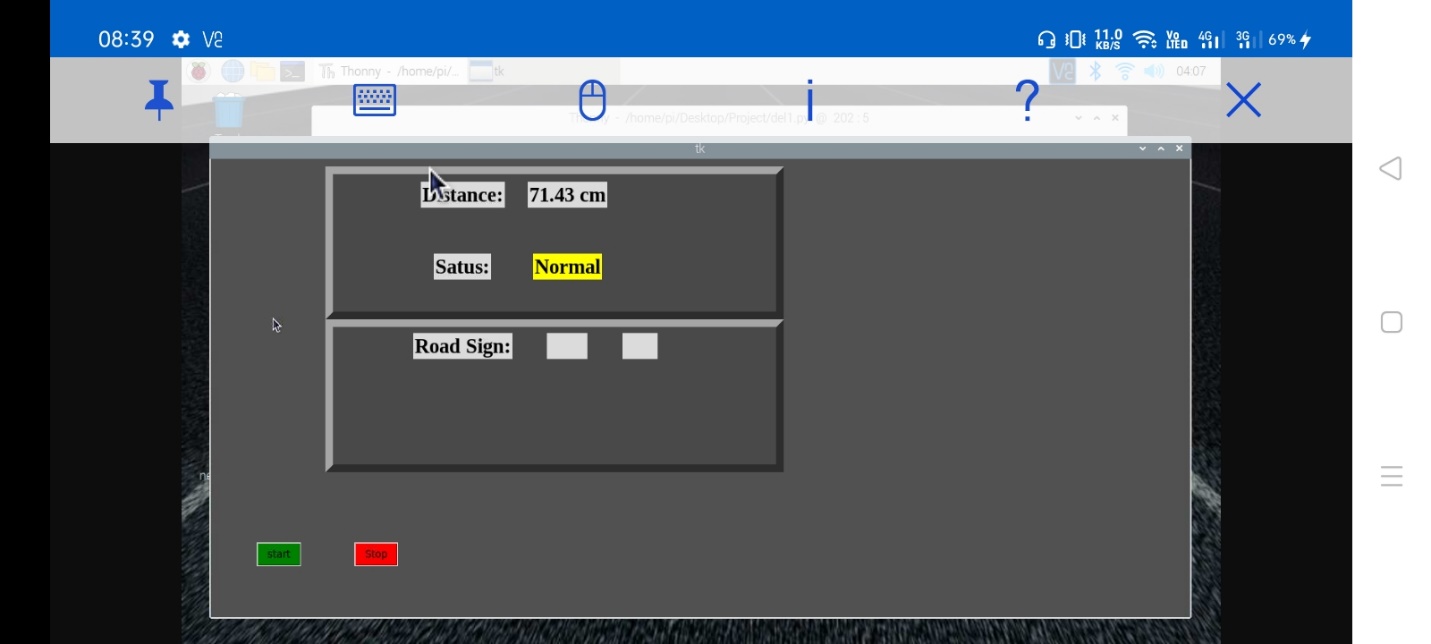
7.1 **Output-figure-1**: - Starting page of module. We give one button to start system and another one to stop system.



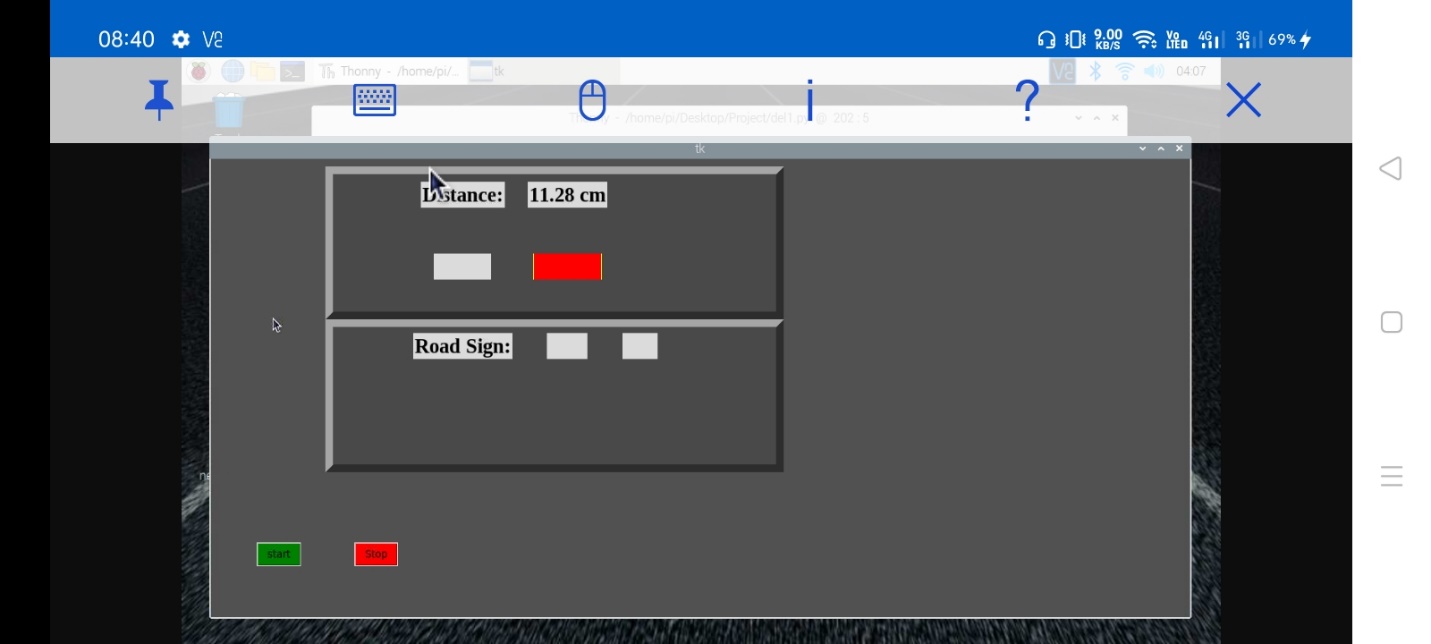
7.2 **Output-figure -2: -** Here live distance are shown and according to that distance status of vehicle was shown. In current screenshot system shoes status: - ‘safe’ and color is green which indicate safe state.



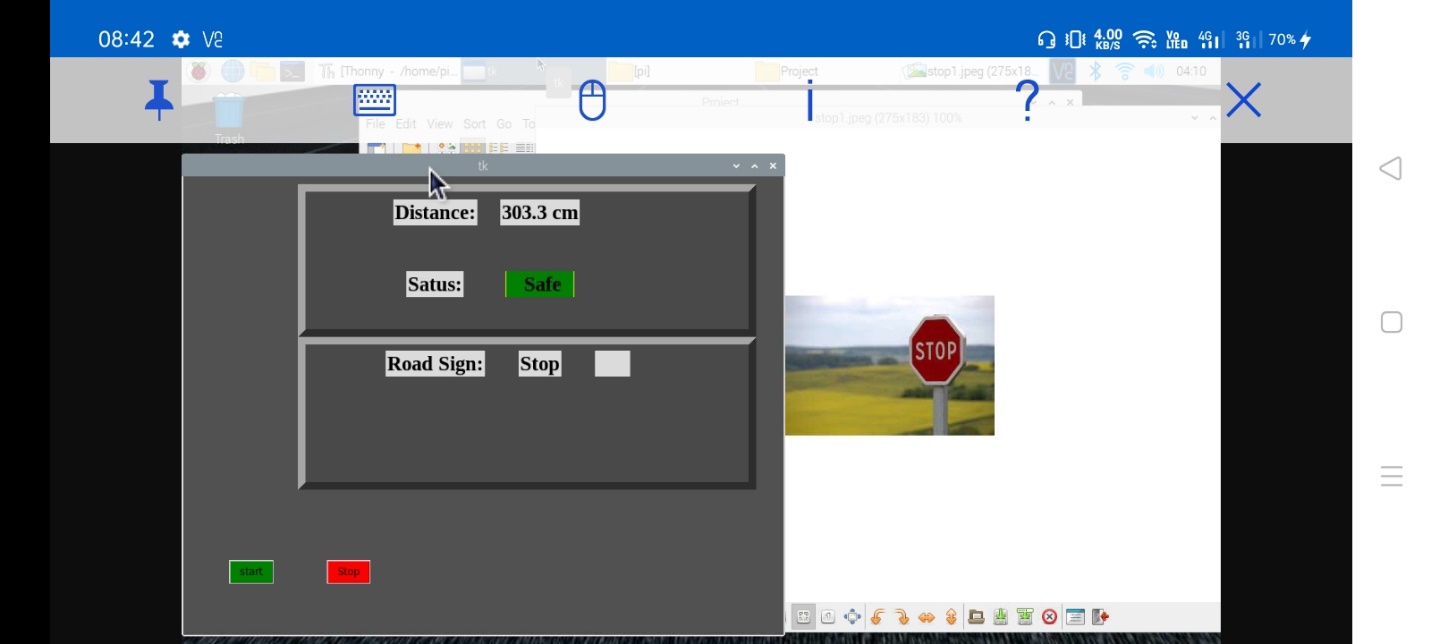
7.3 **Output-figure -3: -** Following screenshot have distance in normal rage, so system shows vehicle status: - ‘Normal’ and color is yellow which indicate normal state.



7.4 **Output-figure -4: -** Following screenshot have distance in danger rage, so system shows vehicle status: - ‘danger’ and color is red which indicate danger state.



7.5 **Output-figure -5: -** Captured image inputted to the system and system analyze traffic signal, and display traffic sign name on screen and also gives voice alert according to sign.



**8. PERFORMANCE ANALYSIS**

Traffic sign detection accuracy depends upon training of dataset. We provide large dataset for taring then it’s given maximum accuracy. So, in point of accuracy the amount of dataset plays vital role. As first stap we provide 20 images and we got no much good accuracy. Then we go with increasing number of images, and analyse accuracy. After taking 200 positive and negative images we got good accuracy. In the end, we provide dataset with 500 images of each positive and negative, and we achieve better accuracy as compared previous.

Following figure describes graph about dependency of dataset and accuracy: -

Fig 8.1 dataset vs accuracy

The above diagram shows the graph over accuracy for the number of images as a dataset. For below 100 images accuracy provides by the system is nearby 20%. As the number of images as datasets increases also accuracy increases.

Testing is important step during whole lifecycle of system. We train our traffic sign detection on cascade trainer. Firstly, we test that model on cascade tester which is by default tester for cascade trainer. Then we deploy on actual model and analyze accuracy.

Following graph elaborate about all observation of testing of traffic sign detection models.

Fig.8.2 Testing

Above line graph describes the relationship between testing on pc-tool and testing on the actual module. Testing on the actual module provides 5% to 10 % less accurate than testing on pc-tool.

**9.** **APPLICATION**

1.The main application to drivers for achieving the autonomous driving is the reduction of traffic accidents by eliminating human error, increasing road capacity and traffic flow by reducing the distance between cars and making use of traffic management information, relieving the car occupants from driving and navigation activities and allowing them to engage in other activities or rest.

2.The automatic braking system can be used in both light moving vehicles such as two wheelers as well as in heavy moving vehicles such as buses and trucks etc.

3.The system can be implemented in institutional vehicles, taxis, driving school vehicles, etc. Solenoid valves are used in a wide variety of industries. They are used in machinery, devices, and equipment such as refrigerators and automatic faucets.

4.Also from the alert system it will alert to the driver for safety driving like, if camera catches the no horn zone then it alerts to the driver. Also, like speed limiter, pedestrian crossing, school ahead, one way, Etc.

Some of the applications of this project are:

1) Used in high speed trains.

2) Used in military application such as spy robot.

3) Used in heavy trucks.

4) Used in heavy vehicles as well as light vehicles

**10. CONCLUSION:**

We built a system where a driver will experience less risk in driving the vehicle.

It was a first step towards building an autonomous vehicle or driver-less vehicle.

At the primary level the driver will have full control over vehicle but if situations like brake fail,

engine failure means there are some scenarios like engine fails suddenly, in such cases this system will do some extraordinary activities like sudden brake.

Also, this system gives alerts related to traffic signals such as stop, left, right and also warning signals which passes to the vehicle.

Driver will have full control over vehicle but because of this system the driver will ensure no accident will happen if he fails somewhere.

**11. INSTALLATION GUIDE AND USER MANUAL**

**Installation Steps:**

* Raspbian OS installed is on Raspberry Pi 3
* python is already available in Raspbian OS
* Installation of required external libraries
  + . OpenCV- for image processing
  + Tkinter library -for Graphical User Interface
  + pygame library -to give output through sound
  + importsys, signal, time, Rpi.
* Add chrome extension - Fatkun Batch
* Install Cascade Trainer-GUI tool

**12. PROJECT ETHICS**

As an Information Technology student, we believe it is unethical to,

1. Surf the Internet for personal and non-class related purpose during classes.
2. Make a copy of software for personal or commercial use.
3. Make a copy of software for friend.
4. Loan CDs of software to friend.
5. Download pirated software from the internet.
6. Distribute pirated software from internet.
7. Buy software on single user license and then install it on multiple computers.
8. Share a pirated copy of Software.
9. Install a pirated copy of Software.

**14. REFERENCES**

**14.1 RESEARCH PAPERS: -**

* M. Rajyalakshmi, B. Kranthi Kumar, A. Krishna Vaibhav, Md Ariefindex Khan **“Design and Fabrication of Intelligent Mechatronic Braking System”** IJSTE - International Journal of Science Technology & Engineering | Volume 4 | Issue 10 | April 2018 ISSN (online): 2349-784X
* Prachi Gawande**,” Traffic Sign Detection and Recognition Using Open CV”** International Research Journal of Engineering and Technology (IRJET) Volume: 04 Issue: 04 | Apr -2017 e-ISSN: 2395 -0056, p-ISSN: 2395-0072
* Hemalatha B K, P Pooja, Chaithra M, Megha S, Rakshitha R T

**“Automatic Braking System for Automobiles Using IR Sensor”**

**14.2 Other references:**

* Coursera platform courses:
  + Cyber Security and the Internet of Things
  + Traffic Sign Classification using deep learning in python/keras
* <https://www.raspberrypi.org/documentation/>
* <https://amin-ahmadi.com/cascade-trainer-gui/>

Training tool for dataset training and testing.