Karnataka State Council for Science and Technology
(An autonomous organisation under the Dept. of Science & Technology, Govt. of Karnataka)
Indian Institute of Science Campus, Bengaluru – 560 012
Telephone: 080-23341652, 23348848, 23348849, 23348840
Email: office.kscst@iisc.ac.in, office@kscst.org.in ♦ **Website:** www.kscst.iisc.ernet.in, www.kscst.org.in

KSCST

Mr. H. Hemanth Kumar
Executive Secretary

11th May, 2022

Ref: 7.1.01/SPP/91

The Principal,
P.E.S. Institute of Technology and Management,
N. H. 206, Sagar Road,
Shivamogga – 577 204

Dear Sir/Madam,

Sub : Sanction of Student Project - 45th Series: Year 2021-2022

Project Proposal Reference No. : **45S_BE_0919**

Ref : Project Proposal entitled **DROWSINESS DETECTION SYSTEM WITH EMBEDED SYSTEM & IOT**

We are pleased to inform that your student project proposal referred above, has been approved by the Council under "Student Project Programme - 45th Series". The project details are as below:

Student(s)	Mr. SYED FAIZAN	**Department**	MECHANICAL ENGINEERING
	Mr. MOHAMMAD FAIZAN A ATTAR		
	Mr. SUNPREET SINGH		
	Mr. UZAIR KHAN AMJED	**Sanctioned Amount (in Rs.)**	8,000.00
Guide(s)	Mr. GANESH U L		

Instructions:

a) The project should be performed based on the objectives of the proposal submitted.

b) Any changes in the project title,objectives or students team is liable for rejection of the project and your institution shall return the sanctioned funds to KSCST.

c) Please quote your project reference number printed above in all your future correspondences.

d) After completing the project, 2 to 3 page write-up (synopsis) needs to be uploaded on to the following Google Forms link https://forms.gle/YMn9K7XETu96i8KbA. The synopsis should include following:

1) Project Reference Number

2) Title of the project

3) Name of the College & Department

4) Name of the students & Guide(s)

5) Keywords

6) Introduction / background (with specific reference to the project, work done earlier, etc) - about 20 lines

7) Objectives (about 10 lines)

8) Methodology (about 20 lines on materials, methods, details of work carried out, including drawings, diagrams etc)

9) Results and Conclusions (about 20 lines with specific reference to work carried out)

10) Scope for future work (about 20 lines).

e) In case of incompeted projects, the sanctioned amount shall be returned to KSCST.

f) The sanctioned amount will be transferred by NEFT to the bank account provided by the College/Institute.

g) The sponsored projects evaluation will be held in the Nodal Centre/Online Mode and the details of the same will be intimated shortly by email / Website announcement.

h) After completion of the project, soft copy of the project report duly signed by the Principal, the HoD, Guide(s) and studetn(s) shall be uploaded in the following Google Forms Link https://forms.gle/PciAaAVisn6bn8AM7. The report should be prepared in the format prescribed by the university.

Please visit our website for further announcements / information and for any clarifications please email to spp@kscst.org.in

Thanking you and with best regards,

Yours sincerely,

(H. Hemanth Kumar)

Copy to:

1) The HoD
MECHANICAL ENGINEERING
P.E.S. INSTITUTE OF TECHNOLOGY AND MANAGEMENT, SHIVAMOGGA

2) Mr. GANESH U L
MECHANICAL ENGINEERING
P.E.S. INSTITUTE OF TECHNOLOGY AND MANAGEMENT, SHIVAMOGGA

3) THE ACCOUNTS OFFICER
KSCST, BENGALURU

ABSTRACT

In recent years, driving has become an important part of our day-to-day life, especially in urban areas sleepiness-related accidents are occurring infrequent. Road accidents are apparently a global hazard. Based on the survey of the National Crime Records Bureau (NCRB - 2022), about 1,35,000 traffic-related demises happened every year in India. These factors lead to the development of Intelligent Transportation System (ITS). If the accident caused by abnormalities of the driver, it can be prevented by placing abnormality detecting system within the vehicle. Drowsiness detection system is introduced as a part of safety measure to avoid these types of accidents. Along with this, sensors for drink and drive and accident intimation are also can be used for sharing the real time updates of the vehicle. This project includes both Hardware and Software part of the design, the hardware components required are Raspberrypi4, USB camera, different types of sensors, Batteries, Chasis, Motors, Motor Drivers and Connecting wires. The Software requirements are Python, AI, ML, HTML, CSS, MyPHP, SQL Database. The objective of this project is to How to effectively monitor and prevent driver fatigue driving has much real significance to reduce traffic accidents and personnel mortality. The system through many studies of the driver's eyes, the research done here include the four parts i.e., Driver’s Face Detection, Driver’s eye location and Tracking, Driver’s eye state Recognition, Driver’s fatigue state identification.

CONTENTS

CHAPTER 5: EXPERIMENTATION

CHAPTER 6: RESULTS AND DISCUSSIONS

CHAPTER 7: CONCLUSIONS AND FUTURE SCOPE OF WORK

CHAPTER 8: COSTING

LIST OF FIGURES

LIST OF TABLES

Chapter 1

INTRODUCTION

In recent years, driving has become an important part of our day-to-day life, especially in urban areas sleepiness-related accidents are occurring infrequent. Road accidents are apparently a global hazard in our country. Based on the survey of the National Crime Records Bureau (NCRB) about 1,35,000 traffic-related demise happen every year in India. These factors lead to the development of Intelligent Transportation System (ITS). If the accident caused by abnormalities of the driver, it can be prevented by placing abnormality detecting system within the vehicle.

Tiredness can be caused by several phenomena like psychosocial factors, health factors, and physiological factors. Based on the survey of National Highway Traffic Safety Administration of the United States of America (USA), police studied around 1, 00, 000 crashes occurred due to the drowsiness of the driver, it causes major losses like 71,000 wounds, $12.5 billion budgetary mishap and 1,550 passings. The issues can be overcome by implementing several methods which include Support Vector Machines (SVM), fuzzy-based system, neuro-fuzzy approach and Artificial Neural Networks (ANN) for detecting the drowsiness. It is difficult to correctly say about an exact number of sleep-related accidents, but traffic research shows up to 20% of accidents happen due to fatigue of the driver. Drowsiness detection can be carried out by two techniques. The first technique is intrusive and second is nonintrusive. The intrusive technique involves computation of mind wave monitoring, heart-beat rate etc.

1.1 Fatigue can cause EEG changes:

The EEG is not on the performance of the same when the cerebral cortex is in excitement or inhibition. According to the EEG's frequency distribution and waveforms, assumes the function status of the brain activity, so as to speculate whether the driver is fatigued. However, the EEG is vulnerable to interference from external factors and there are so many differences in individual physiological response.

1.2 Head posture

When the driver is fatigued, the head will always downward-sloping. According to statistics, the correlation coefficient of head position and fatigue degree is about 0.8. However, some driver's head posture will not change basically, the correlation coefficient will be negative and the system's judge and early warning failed.

1.2.1 Steering wheel's rotation amplitude and handgrip strength: System detects the driver's mental state by monitoring steering wheel's movements and patterns. With the deepening of the driver fatigue, the number of greatly rotation will be increase; the handgrip strength will become larger.

1.2.2 Road tracker: This method monitors the time and the deviation degree of vehicles leaving from the white lines by installing camera in the same perspective with the driver on the vehicle. This measurement requires the white line must be exist and clear enough on the road, so the interference of outside conditions is very great.

1.3 Objective of the Fatigue Detection

How to effectively monitor and prevent driver fatigue driving has much real significance to reduce traffic accidents and personnel mortality. After Comparison of the above fatigue alarm, the system through many studies of the driver's eyes, the research work in this paper includes the four parts, i.e.,

- Driver’s face detection,
- Driver’s eye locating and tracking,
- Driver’s eye state recognition and
- Driver’s fatigue state identification.

Chapter 2

LITERATURE REVIEW

The main aim is to detect drowsiness of driver, it can be done in different ways like detecting facial expression of the driver and measuring Eye Aspect Ratio (EAR). Blinking pattern is different for each and every individual. The pattern gets varied in terms of squeezing degree of eye, blink duration and speed of closing and opening the eye. The proposed method involved with the following methodologies such as Haar Cascade Classifiers, Shape Predictor_68_facial landmark detection, Eye Aspect Ratio (EAR), Ubidots cloud service and Twilio API. 30 International Journal of Engineering & Technology

2.1 Haar Cascade Classifiers

In Haar Cascade Classifiers, a lot of similar and dissimilar images are trained in order to detect fatigue of the driver. OpenCV is a learning-based method, packed with a detector as well as a trainer. For training, a separate database is maintained for face and eye with several positive and negative images having eye closed and opened conditions and different set facial images [15]. In 2013, Patil et al suggested a vision-based drowsiness is carried out using Support Vector Machine and Haar Cascade Classifiers.

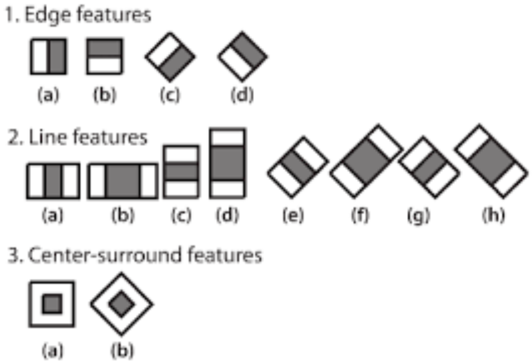


Fig. 2.1: Haar Cascade Classifier

2.2 Shape Predictor_68_Facial Landmark Detection and Eye Aspect Ratio (Ear)

In order to predict the face and eye region in the live video stream, shape predictor is used. Fig.1 shows the sleepiness which is measured by calculating the eye aspect ratio (Euclidean distance between the eyes are calculated), the arguments are passed to the predefined dataset and facial landmark detection is carried out. For every video sequence, the eye landmarks are located. The aspect ratio between width and height of the eye is calibrated.

$$\text{EAR} = \|\boldsymbol{p2}-\boldsymbol{p6}\|+\|\boldsymbol{p3}-\boldsymbol{p5}\| \ \mathbf{2}\,\|\boldsymbol{p1}-\boldsymbol{p4}\| \text{-------------- i}$$

Where p1, p6 are the two-dimensional landmark location, represented in Fig.2. The EAR is mostly stable when an eye is open and is getting close to zero while the eye is not in open state. If the person viewing the camera continuously, the Eye Aspect Ratio (EAR) is found to be normal and it reaches low value when he/she closing the eye for a longer time. When the lower value is reached, then drowsiness is detected. In 2012, Ubidots elaborated on connected software and hardware solutions to remotely control, automate processes for healthcare clients and monitor. Twilio is a cloud communication platform as a service (PaaS), it allows a software developer to programmatically send and receive text message using its web service APIs, make and receive phone calls. The paper introduces alerting process when the driver feels sleepy using cloud server and mobile API to send a message and at the same time providing an alarm signal to the driver. 3. Programming Algorithm Raspberry Pi 4 is effective with Python IDLE. It involves programming in software with extensions of OpenCV computer vision installed. The entire algorithm for drowsiness detection is shown with the help of a flowchart shown in Fig 2. The program will start to run and it can be terminated using command line interfaces in Raspberry Pi 4 or the system is turned off. In order to initiate the program execution, it will import the following libraries like numpy, OpenCV, play sound, argparse, dlib, distance, timer, client, ApiClient, and picamera

2.3 Programming Algorithm

Raspberry Pi 4 is effective with Python IDLE. It involves programming in software with extensions of OpenCV computer vision installed. The entire algorithm for drowsiness detection is shown with the help of a flowchart shown in Fig 2. The program will start to run and it can be terminated using command line interfaces in Raspberry Pi 4 or the system is turned off. In order to initiate the program execution, it will import the following libraries like numpy, OpenCV, play sound, argparse, dlib, distance, timer, client, ApiClient, and picamera

2.4 Introduction to PHP:

PHP is a server-side scripting language designed primarily for web development but is also used as a general-purpose programming language. Originally created by RasmusLerdorf in 1994,the PHP reference implementation is now produced by The PHP Development Team. PHP originally stood for *Personal Home Page*, but it now stands for the recursive acronym *PHP: Hypertext Preprocessor*.

PHP code may be embedded into HTML code, or it can be used in combination with various web template systems, web content management systems and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a module in the web server or as a Common Gateway Interface (CGI) executable. The web server combines the results of the interpreted and executed PHP code, which may be any type of data, including images, with the generated web page. PHP code may also be executed with a command-line interface (CLI) and can be used to implement standalone graphical applications.

The standard PHP interpreter, powered by the Zend Engine, is free software released under the PHP License. PHP has been widely ported and can be deployed on most web servers on almost every operating system and platform, free of charge.

The PHP language evolved without a written formal specification or standard until 2014, leaving the canonical PHP interpreter as ade facto standard. Since 2014 work has gone on to create a formal PHP specification.

During the 2010s there have been increased efforts towards standardization and code sharing in PHP applications by projects such as PHP-FIG in the form of PSR-initiatives as well as Composer dependency manager and the Packagist repository.

PHP development began in 1995 when RasmusLerdorf wrote several Common Gateway Interface (CGI) programs in C, which he used to maintain his personal homepage. He extended them to work with web forms and to communicate with databases, and called this implementation "Personal Home Page/Forms Interpreter" or PHP/FI.

PHP/FI could help to build simple, dynamic web applications. To accelerate bug reporting and to improve the code, Lerdorf initially announced the release of PHP/FI as "Personal Home Page Tools (PHP Tools) version 1.0" on the Usenet discussion group comp.infosystems.www.authoring.cgi on June 8, 1995.This release already had the basic functionality that PHP has as of 2013. This included Perl-like variables, form handling, and the ability to embed HTML. The syntax resembled that of Perl but was simpler, more limited and less consistent.

Lerdorf did not intend the early PHP to become a new programming language, but it grew organically, with Lerdorf noting in retrospect: "I don’t know how to stop it, there was never any intent to write a programming language […] I have absolutely no idea how to write a programming language, I just kept adding the next logical step on the way. "A development team began to form and, after months of work and beta testing, officially released PHP/FI 2 in November 1997.

The fact that PHP lacked an original overall design but instead developed organically has led to inconsistent naming of functions and inconsistent ordering of their parameters. In some cases, the function names were chosen to match the lower-level libraries which PHP was "wrapping", while in some very early versions of PHP the length of the function names was used internally as a hash function, so names were chosen to improve the distribution of hash values.

- PHP 3 and 4:

ZeevSuraski and Andi Gutmans rewrote the parser in 1997 and formed the base of PHP 3, changing the language's name to therecursive acronym PHP: Hypertext Preprocessor.Afterwards, public testing of PHP 3 began, and the official launch came in June 1998. Suraski and Gutmans then started a new rewrite of PHP's core, producing the Zend Engine in 1999. They also founded Zend Technologies in Ramat Gan, Israel.

On May 22, 2000, PHP 4, powered by the Zend Engine 1.0, was released.As of August 2008 this branch reached version 4.4.9. PHP 4 is no longer under development nor will any security updates be released.

- PHP 5:

On July 13, 2004, PHP 5 was released, powered by the new Zend Engine II. PHP 5 included new features such as improved support for object-oriented programming, the PHP Data Objects (PDO) extension (which defines a lightweight and consistent interface for accessing databases), and numerous performance enhancements. In 2008 PHP 5 became the only stable version under development. Late static binding had been missing from PHP and was added in version 5.3.

Many high-profile open-source projects ceased to support PHP 4 in new code as of February 5, 2008, because of the GoPHP5 initiative, provided by a consortium of PHP developers promoting the transition from PHP 4 to PHP 5.

Over time, PHP interpreters became available on most existing 32-bit and 64-bit operating systems, either by building them from the PHP source code, or by using pre-built binaries. For the PHP versions 5.3 and 5.4, the only available Microsoft Windows binary distributions were 32-bit x86 builds requiring Windows 32-bit compatibility mode while using Internet Information Services (IIS) on a 64-bit Windows platform. PHP version 5.5 made the 64-bit x86-64 builds available for Microsoft Windows.

- PHP 6 and Unicode:

PHP received mixed reviews due to lacking native Unicode support at the core language level. In 2005, a project headed by Andrei Zmievski was initiated to bring native Unicode support throughout PHP, by embedding the International Components for Unicode (ICU) library, and representing text strings as UTF-16 internally. Since this would cause major changes both to the internals of the language and to user code, it was planned to release this as version 6.0 of the language, along with other major features then in development.

However, a shortage of developers who understood the necessary changes, and performance problems arising from conversion to and from UTF-16, which is rarely used in a web context, led to delays in the project. As a result, a PHP 5.3 release was created in 2009, with many non-Unicode features back-ported from PHP 6, notably namespaces. In March 2010, the project in its current form was officially abandoned, and a PHP 5.4 release was prepared containing most remaining non-Unicode features from PHP 6, such as traits and closure re-binding. Initial hopes were that a new plan would be formed for Unicode integration, but as of 2014 none have been adopted.

- PHP 7:

During 2014 and 2015, a new major PHP version was developed, which was numbered PHP 7. The numbering of this version involved some debate. While the PHP 6 Unicode experiment had never been released, several articles and book titles referenced the PHP 6 name, which might have caused confusion if a new release were to reuse the name. After a vote, the name PHP 7 was chosen.

The foundation of PHP 7 is a PHP branch that was originally dubbed PHP next generation (phpng). It was authored by Dmitry Stogov, Xinchen Hui and Nikita Popov, and aimed to optimize PHP performance by refactoring the Zend Engine while retaining near-complete language compatibility. As of 14 July 2014, WordPress-based benchmarks, which served as the main benchmark suite for the phpng project, showed an almost 100% increase in performance. Changes from phpng are also expected to make it easier to improve performance in the future, as more compact data structures and other changes are seen as better suited for a successful migration to a just-in-time (JIT) compiler. Because of the significant changes, the reworked Zend Engine is called Zend Engine 3, succeeding Zend Engine 2 used in PHP 5.

Uses:

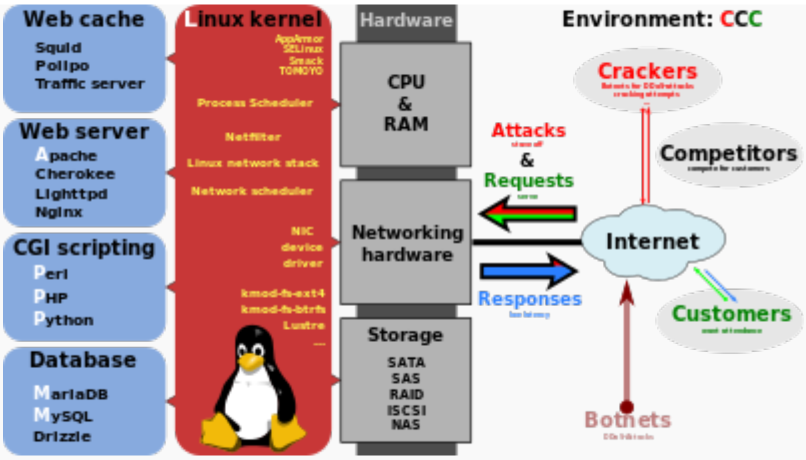


Fig. 2.2: Uses of PHP diagram

PHP is a general-purpose scripting language that is especially suited to server-side web development, in which case PHP generally runs on a web server. Any PHP code in a requested file is executed by the PHP runtime, usually to create dynamic web page content or dynamic images used on websites or elsewhere. It can also be used for command-line scripting and client-side graphical user interface (GUI) applications. PHP can be deployed on most web servers, many operating systems and platforms, and can be used with many relational database management systems (RDBMS). Most web hosting providers support PHP for use by their clients. It is available free of charge, and the PHP Group provides the complete source code for users to build, customize and extend for their own use.

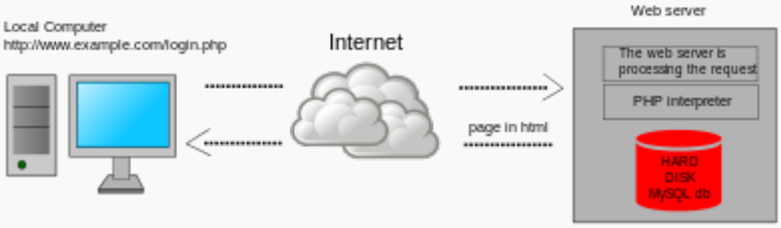


Fig. 2.3: Data Exchange diagram

PHP acts primarily as a filter, taking input from a file or stream containing text and/or PHP instructions and outputting another stream of data. Most commonly the output will be HTML, although it could be JSON,XML or binary data such as image or audio formats. Since PHP 4, the PHP parser compiles input to produce bytecode for processing by the Zend Engine, giving improved performance over its interpreter pre decessor.

Originally designed to create dynamic web pages, PHP now focuses mainly on server-side scripting, and it is similar to other server-side scripting languages that provide dynamic content from a web server to aclient, such as Microsoft's ASP.NET, Sun Microsystems' JavaServer Pages, and perl. PHP has also attracted the development of many software frameworks that provide building blocks and a design structure to promote rapid application development (RAD). Some of these include PRADO, CakePHP, Symfony,CodeIgniter, Laravel, YiiFramework, Phalcon a nd Zend Framework, offering features similar to other web frameworks.

2.5 Introduction to MySQL

MySQL is an open-source relational database management system (RDBMS). Its name is a combination of "My", the name of co-founder Michael Widenius' daughter, and "SQL", the abbreviation for Structured Query Language. The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation. For proprietary use, several paid editions are available, and offer additional functionality.

MySQL is a central component of the LAMP open-source web application software stack (and other "AMP" stacks). LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python". Applications that use the MySQL database include: TYPO3,MODx,Joomla,WordPress, phpBB, MyBB, and Drupal. MySQL is also used in many high-profile, large-scale websites, including Google (though not for searches), Facebook, Twitter, Flickr, and YouTube.

MySQL was created by a Swedish company, MySQL AB, founded by David Axmark, Allan Larsson and Michael "Monty" Widenius. Original development of MySQL by Widenius and Axmark began in 1994. The first version of MySQL appeared on 23 May 1995. It was initially created for personal usage from mSQL based on the low-level languageISAM, which the creators considered too slow and inflexible. They created a new SQL interface, while keeping the same API as mSQL. By keeping the API consistent with the mSQL system, many developers were able to use MySQL instead of the (proprietarily licensed) mySQL antecedent...

2.6 Connecting to Database in xamp:

PHP MySQL commands:

- mysqli_connect
- mysqli_query
- mysqli_fetch_array
- mysqli_close

The Plan

- make the connection and select the database
- perform the query on the table
- print out the data
- close the connection

Chapter 3

SCOPE OF THE WORK

3.1 Problem Defination

In recent years, driving has become an important part of our day-to-day life, especially in urban areas sleepiness-related accidents are occurring infrequent. Road accidents are apparently a global hazard in our country. Based on the survey of the National Crime Records Bureau (NCRB) about 1,35,000 traffic-related demise happen every year in India. These factors lead to the development of Intelligent Transportation System (ITS). If the accident caused by abnormalities of the driver, it can be prevented by placing abnormality detecting system within the vehicle.

3.1.1 Existing System:

An important factor for causing accidents in traffic that the driver's fatigue. Many countries are engaged in research in this area actively now. How to effectively monitor and prevent driver fatigue driving has much real significance to reduce traffic accidents and personnel mortality.

When the driver is fatigued, the head will always downward-sloping. According to statistics, the correlation coefficient of head position and fatigue degree is about 0.8. However, some driver's head posture will not change basically, the correlation coefficient will be negative and the system's judge and early warning failed.

System detects the driver's mental state by monitoring steering wheel's movements and patterns. With the deepening of the driver fatigue, the number of greatly rotation will be increase; the handgrip strength will become larger.

This method monitors the time and the deviation degree of vehicles leaving from the white lines by installing camera in the same perspective with the driver on the vehicle. This measurement requires the white line must be exist and clear enough on the road, so the interference of outside conditions is very great.

3.2 Proposed System

3.2.1 Eyes Location: In order to make the image smoothing, doing some treatments before eyes location, including image denoising and enhancement, which is a prerequisite to ensure precise eyes location achieve the better result.

The first step: Locating the eye region roughly

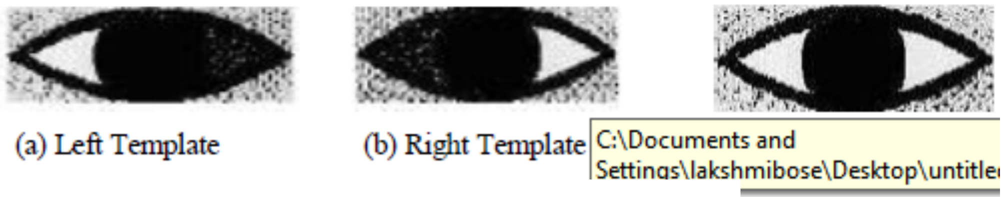
The edge feature analysis method means, making use of the vertical gray-scale projection curve of the image determined the left and right borders of the face according to the convex peak width, then making use of the horizontal gray-scale projection curve of the gotten region determined roughly the up and down border of the eye's location region. The region that corresponds to a face is a convex peak with a certain width by observing the vertical gray-scale projection curve of a number of different single-face images.

The left and right borders of the convex peak generally represented the left and right borders of the face . When the left and right borders of the face are established, take the region of the face between the left and right borders as the study object, and then make the horizontal gray-scale projection curve of the image, something will be found by observing. The first minimum point of the horizontal gray-scale projection curve corresponds to the crown of the head, the maximum point corresponds to one of the foreheads, the secondary maximum point corresponds to the central of the nose, and take the region between the central of the nose and the crown of the head as the rough located region (See Fig.1).

The second step: sifting the similar eye points collection

The primary problem is selecting the appropriate template prior to the template matching. In the follow-up algorithm, it is necessary to use the relative position between two eyes to locate the two eyes from a number of similar points, so long as to ensure that there are two real eye-points among a number of similar eye-points. 9

In order to reduce the two eyes' sensitivity to the eye template and improve the robustness, the system adopts the synthetic eye template of the two eyes (See Figure 2.3).



(a) Left Template

(b) Right Template

Fig. 3.1: Eye Template diagram

In order to select the similar eye-points, it is desirable first to establish the similarity metric. The general way is doing the relevant operation to the local image and the image template, the cross-correlation coefficient obtained in this way is regarded as the similarity metric (See Formula 1). Two parameters are used to describe the synthetic template: template height M , width template N .

$$\rho_{xy}=\frac{\sum_{i,j=1}^{M,N}\left(T(i,j)-\overline{T}\right)\left(S_r(i+x,j+y)-\overline{S}_r\right)}{M\times N\times\sqrt{\frac{1}{M\times N}\sum_{i,j=1}^{M,N}\left(T(i,j)-\overline{T}\right)^2}\times\sqrt{\frac{1}{M\times N}\sum_{i,j=1}^{M,N}\left(S_r(i+x,j+y)-\overline{S}_r\right)^2}} \quad (1)$$

Therein, N is the synthetic eye template, the size is $M \times N$;T is the average of the eye template image; r S T is the average of the local image that matches with the template in the expected face recognition image; (x, y) is the coordinates of search points in the face image.

According to the above formula, operating P_{xy}, always have P_{xy} □□□□, and the greater the P_{xy} □, the higher the matching. However, due to the synthetic eye template exists a certain error and image acquisition will be affected by external conditions, when the interference, these may lead to the greatest similarity is not the real eye point, so locating the eye point can not only be determined by the size of the similarity

In order not to miss the real eye point, the way is selecting roughly a similar eye point collection including the two real eye points (See Figure 2.4) Xi , Yi) i= 1,2,…n} , and then obtains the two real eye points through prior knowledge calibration. n is a optional coefficient.

3.2.2 Eye Tracking

This system adopts the improved target tracking algorithm when it traces the eyes. The essence of target tracking is that it carries on the pinpoint while recognizing target in the image sequence.

The target tracking algorithm realized in this system divides into two parts: the primary algorithm and the modified algorithm. The primary algorithm is based on the template matching technology, namely, after pinpointing the eye point to the first frame image, it selects this eye point in the image as the tracking object and extracts appearance information of this eye point as the new eye template, in the following sequence image, it will match the candidate image region and this new eye template, then take the most similar image region as the position that this eye point in the current image.

The modified algorithm adopts the method of selecting candidate image region. It reduces the match times greatly, and then reduces the computation complexity of the system. The system uses the image gathering card for gathering image, and the rate is 25 frame per second, while the pilot driving, the head's amount of exercise is very small, therefore the position difference between the two neighboring frame images is very small, namely, it can obtain the roughly position of the eye point in the next image after pinpointing real eye point.

After adopts the target tracking algorithm, the system does not need to carry on an eye pinpointing for every frame image in the image sequence, but only repositions the eye point to the image which loses the tracking object, thus it improves operating efficiency of the system greatly and satisfies real-time request of the system too.

3.3 Feasibility Study:

Feasibility Study aims to objectively and rationally uncover the strengths and weaknesses of the existing business or proposed adventure, opportunities and threats as presented by the environment, the resources required to carry through, and ultimately the prospects of success .It is an important step in any software development process, The main functionality of feasibility study is the analysis of cost required for developing and executing the system, time required for each phase of the system etc.

3.3.1 Operational Feasibility:

This type of feasibility deals with the working and functionality that the software provides to its user. Proposed systems are beneficial only if they can be turned into information System that will meet the organizations operating requirements.

Our project provides much different functionality like Adding, Updating, Deleting, the photos, and complete authorization of the administrator, etc...

Operational feasibility is a measure of how well a proposed system solves the problems, and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development.

The operational feasibility assessment focuses on the degree to which the proposed development projects fits in with the existing business environment and objectives with regard to development schedule, delivery date, corporate culture, and existing business processes.

3.3.2 Technical feasibility:

This study was conducted to determine whether the software and hardware components are capable of supporting our application which our system did so by using Centralize database for the easy storage and fast retrieval of data and its architecture adds to its advantage making it simple, fast and secure.

Our Project was build keeping in mind of all these feasibilities to make it a true, complete, trustworthy and user-friendly application.

3.3.3 Economic feasibility:

This type of feasibility is also known as Financial Feasibility and is most frequently used method for evaluating the effectiveness and cost of the candidate system. More commonly known as cost/benefit analysis, the procedure is to determine the benefit and saving that are expected from the candidate system and compare them of time by automating the process generation.

Our projects Development cost is less as it makes use of smart and powerful software's (PHP, MySQL, HTML, CSS, JAVA SCRIPT, BOOTSTRAP). It is a long-lasting type of application that can be used and maintained for many years. Since it's a server-side application built using the mentioned tools it is also well secured and safe to use…

Chapter 4

METHODOLOGY AND EXPERIMENTATION

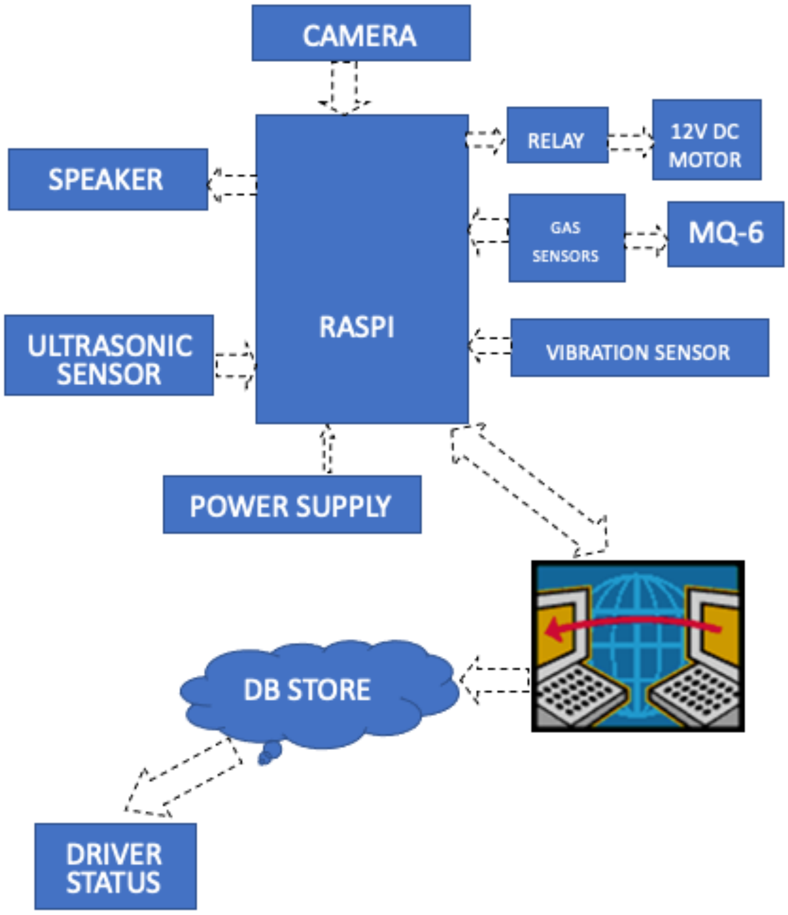


Fig. 4.1: Block Diagram of proposed system

4.1 Hardware requirement:

- Raspberrypi4 model B
- Mmc, Motors
- Motor-driver
- PCB board
- Sensors (Ultrasonic sensor, Vibration sensor, MQ-6 sensor)
- USB camera
- 12v Battery
- Pi power supply

4.2 SRS(Software Requirement Specification):

A software requirements specification (SRS) is a document that captures complete description about how the system is expected to perform. It is usually signed off at the end of requirements engineering phase

Types of Requirements:

The below diagram depicts the various types of requirements documented via the SRS.

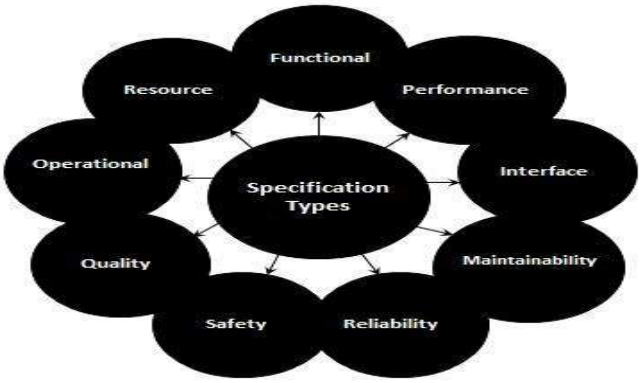


Fig. 4.2: Software requirement

4.3 System Design

Design is the first step in the development phase for any techniques and principles for the purpose of defining a device, a process or system in sufficient detail to permit its physical realization.

This system adopts the Embedded system, IOT, AI & ML algorithms to provide the output in this prototype Raspberrypi4 is used as microprocessor which process the complete progress of this project & the different types of sensors like - Ultrasonic, MQ-6 & Vibration sensors with a USB camera are embedded to the Raspberrypi4 with motor drivers, Power bank, Battery and motor wheels. Along with this system adopts web technologies to develop application where the drowsiness status can be updated.

Once the software requirements have been analysed and specified the software design involves three technical activities - design, coding, implementation and testing that are required to build and verify the software.

The design activities are of main importance in this phase, because in this activity, decisions ultimately affecting the success of the software implementation and its ease of maintenance are made. These decisions have the final bearing upon reliability and maintainability of the system. Design is the only way to accurately translate the customer's requirements into finished software or a system.

Design is the place where quality is fostered in development. Software design is a process through which requirements are translated into a representation of software. Software design is conducted in two steps. Preliminary design is concerned with the transformation of requirements into data

UML Diagrams:

Actor:

A coherent set of roles that users of use cases play when interacting with the use cases.

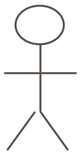


Fig. 4.3: UML diagram

Use case:

A description of sequence of actions, including variants, that a system performs that yields an observable result of value of an actor.

UML stands for Unified Modeling Language. UML is a language for specifying, visualizing and documenting the system. This is the step while developing any product after analysis. The goal from this is to produce a model of the entities involved in the project which later need to be built. The representation of the entities that are to be used in the product being developed need to be designed.

There are various kinds of methods in software design:

They are as follows:

- Use case Diagram
- Sequence Diagram
- Collaboration Diagram
- Activity Diagram

- State chat Diagram

USE CASE DIAGRAMS:

Use case diagrams model behaviour within a system and helps the developers understand of what the user require. The stick man represents what's called an actor.

Use case diagram can be useful for getting an overall view of the system and clarifying who can do and more importantly what they can't do.

Use-case for Drowsiness Detection System

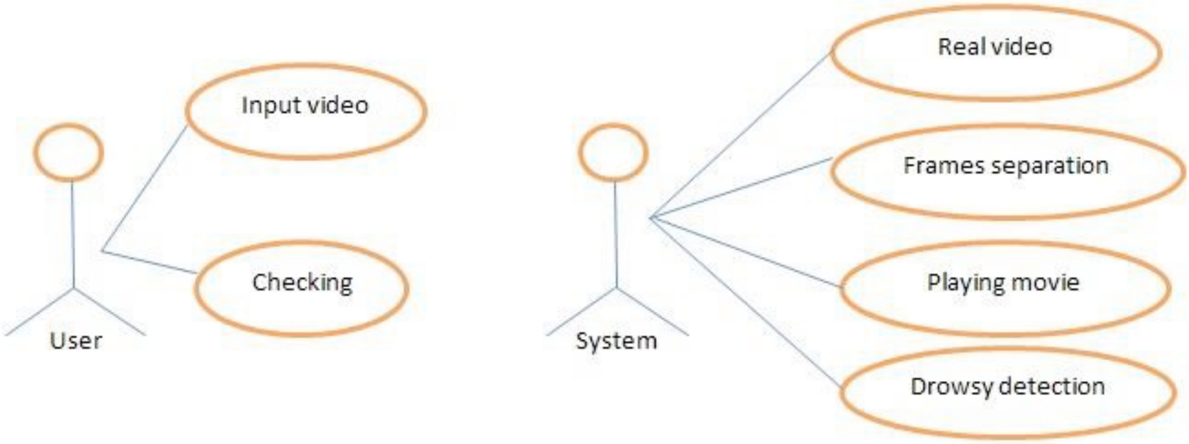


Fig. 4.4: Use- case for drowsiness

Sequence Diagram:

1. It is a much formal and object interaction representational diagram which describes the interaction between various objects involved in the project.
2. A sequence diagram is a kind of interaction diagram that shows how the processes operate with one another and in what order. It describes an interaction by focusing on the sequence of messages that are exchanged, along with their corresponding occurrence specifications on the lifelines.
3. Life line is a named element which represents an individual participating in the interaction. A lifeline is showed by a rectangle forming its "head" followed by a vertical line representing the lifetime of the participant.

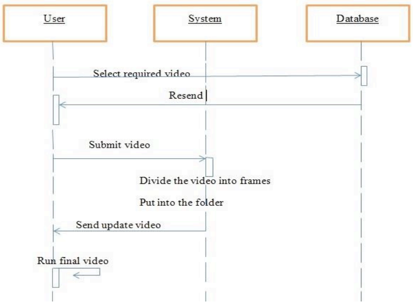


Fig. 4.5: Sequence Diagram

Chapter 5

EXPERIMENTATION

Implementation is the carrying out, execution or practice of a plan, a method, or any design.

Implementation is the action that must follow any preliminary thinking. A good plan helps to optimize the use of the project resources and limits the time spent on resolving problems during implementation.

Project implementation can be referred to as a process whereby “project inputs are converted to project outputs as a set out in the project framework”. The process involves a series of activity, which needs to be planned, operated and controlled, and which will inevitably involve the utilization of resources. It leads to the realization of the project output and immediate objectives.

Project implementation may be looked at as:

- Putting in action the activities of the project.
- Putting into practice what was proposed in the project document i.e.; transforming the project proposal into the actual project.
- Management of the project or executing the project intentions.

5.1 Source Code

1) Pi_Detect_Drowsiness.

```
 # USAGE

# python detect_drowsiness.py --shape-predictor
shape_predictor_68_face_landmarks.dat

# python detect_drowsiness.py --shape-predictor
shape_predictor_68_face_landmarks.dat --alarm alarm.wav

# import the necessary packages

from scipy.spatial import distance as dist

from imutils.video import VideoStream

from imutils import face_utils

from threading import Thread
```

```
import numpy as np

import os

import argparse

import imutils

import time

import dlib

import cv2

import RPi.GPIO as GPIO

import pyttsx3

from urllib.request import urlopen

m1 = 25

m2 = 8

m3 = 7

m4 = 1

vib = 27

alco= 22

GPIO_TRIGGER = 23

GPIO_ECHO = 24

GPIO.setwarnings(False)

GPIO.setmode(GPIO.BCM)       # Use BCM GPIO numbers

GPIO.setup(m1, GPIO.OUT)  #

GPIO.setup(m2, GPIO.OUT)

GPIO.setup(m3, GPIO.OUT)

GPIO.setup(m4, GPIO.OUT)

GPIO.setup(vib, GPIO.IN) #

GPIO.setup(alco, GPIO.IN) #

GPIO.setup(GPIO_TRIGGER, GPIO.OUT)
GPIO.setup(GPIO_ECHO, GPIO.IN)

engine = pyttsx3.init()

newVoiceRate = 145

engine.setProperty('rate',newVoiceRate)

voices = engine.getProperty('voices')

engine.setProperty('voice', voices[2].id)

engine.say("A-I based Drowziness detection system")

engine.runAndWait()


def eye_aspect_ratio(eye):

    # compute the euclidean distances between the two sets of

    # vertical eye landmarks (x, y)-coordinates

    A = dist.euclidean(eye[1], eye[5])

    B = dist.euclidean(eye[2], eye[4])

# compute the euclidean distance between the horizontal

  # eye landmark (x, y)-coordinates

    C = dist.euclidean(eye[0], eye[3])

 # compute the eye aspect ratio

    ear = (A + B) / (2.0 * C)

 # return the eye aspect ratio

    return ear

def mot_stop():

            GPIO.output(m1, False);

            GPIO.output(m2, False);

            GPIO.output(m3, False);

            GPIO.output(m4, False);

def mot_start():
```

```
            GPIO.output(m1, True);
            GPIO.output(m2, False);
            GPIO.output(m3, True);
            GPIO.output(m4, False);


def distance():
    # set Trigger to HIGH
    GPIO.output(GPIO_TRIGGER, True)
     # set Trigger after 0.01ms to LOW
    time.sleep(0.00001)
    GPIO.output(GPIO_TRIGGER, False)
    StartTime = time.time()
    StopTime = time.time()
  # save StartTime
    while GPIO.input(GPIO_ECHO) == 0:
        StartTime = time.time()
# save time of arrival
    while GPIO.input(GPIO_ECHO) == 1:
        StopTime = time.time()
 # time difference between start and arrival
    TimeElapsed = StopTime - StartTime
    # multiply with the sonic speed (34300 cm/s)
    # and divide by 2, because there and back
    distance = (TimeElapsed * 34300) / 2
  return distance
# construct the argument parse and parse the arguments
ap = argparse.ArgumentParser()
```

```
ap.add_argument("-p", "--shape-predictor", required=True,
    help="path to facial landmark predictor")
ap.add_argument("-a", "--alarm", type=str, default="",
    help="path alarm .WAV file")
ap.add_argument("-w", "--webcam", type=int,default=0,
    help="index of webcam on system")
args = vars(ap.parse_args())
 # define two constants, one for the eye aspect ratio to indicate
# blink and then a second constant for the number of consecutive
# frames the eye must be below the threshold for to set off the
# alarm
EYE_AR_THRESH = 0.2
EYE_AR_CONSEC_FRAMES = 10
# initialize the frame counter as well as a boolean used to
# indicate if the alarm is going off
COUNTER = 0
ALARM_ON = False
# initialize dlib's face detector (HOG-based) and then create
# the facial landmark predictor
print("[INFO] loading facial landmark predictor...")
detector = dlib.get_frontal_face_detector()
predictor = dlib.shape_predictor(args["shape_predictor"])
# grab the indexes of the facial landmarks for the left and
# right eye, respectively
(lStart, lEnd) = face_utils.FACIAL_LANDMARKS_IDXS["left_eye"]
(rStart, rEnd) = face_utils.FACIAL_LANDMARKS_IDXS["right_eye"]
# start the video stream thread
```

```
print("[INFO] starting video stream thread...")
vs = VideoStream(src=args["webcam"]).start()
time.sleep(1.0)
# loop over frames from the video stream
while True:
   mot_start()
   z=20
   y=distance()
   print("Distance = ")
   print(y)
   if y<10:
         mot_stop()
         print("Alert! Alert!  Obstacle detedted in front")
         engine = pyttsx3.init()
         engine.say("ALERT! Alert! Obstacle detedted in front")
         engine.runAndWait()
         time.sleep(4)
         z=10
   a=GPIO.input(alco)
   if a==0:
         print (a)
         mot_stop()
         print("alcohol detected")
         engine = pyttsx3.init()
         engine.say("ALERT! Alert!  Alcohol detected")
         engine.runAndWait()
         time.sleep(5)
```

```
        z=10
b=GPIO.input(vib)
if b==0:
        print (b)
        mot_stop()
        print("vibration detected")
        engine = pyttsx3.init()
        engine.say("ALERT! Alert!  vibration detected")
        engine.runAndWait()
        time.sleep(4);
        z=10
frame = vs.read()
frame = imutils.resize(frame, width=450)
gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

# detect faces in the grayscale frame
rects = detector(gray, 0)

# loop over the face detections
for rect in rects:
    # determine the facial landmarks for the face region, then
    # convert the facial landmark (x, y)-coordinates to a NumPy
    # array
    shape = predictor(gray, rect)
    shape = face_utils.shape_to_np(shape)

    # extract the left and right eye coordinates, then use the
    # coordinates to compute the eye aspect ratio for both eyes
    leftEye = shape[lStart:lEnd]
    rightEye = shape[rStart:rEnd]
    leftEAR = eye_aspect_ratio(leftEye)
    rightEAR = eye_aspect_ratio(rightEye)

    # average the eye aspect ratio together for both eyes
    ear = (leftEAR + rightEAR) / 2.0

    # compute the convex hull for the left and right eye, then
    # visualize each of the eyes
    leftEyeHull = cv2.convexHull(leftEye)
    rightEyeHull = cv2.convexHull(rightEye)
    cv2.drawContours(frame, [leftEyeHull], -1, (0, 255, 0), 1)
    cv2.drawContours(frame, [rightEyeHull], -1, (0, 255, 0), 1)

    # check to see if the eye aspect ratio is below the blink
    # threshold, and if so, increment the blink frame counter
    if ear < EYE_AR_THRESH:
        COUNTER += 1

        # if the eyes were closed for a sufficient number of
        # then sound the alarm
        if COUNTER >= EYE_AR_CONSEC_FRAMES:
            # draw an alarm on the frame
            mot_stop()
            cv2.putText(frame, "DROWSINESS ALERT!", (10, 30),
            cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)
            time.sleep(4);
            engine = pyttsx3.init()
            engine.say(" Alert! DROWSINESS ALERT!")
            engine.runAndWait()
            f=open('wf','w')
            f.write('1')
            f.close()
                    # otherwise, the eye aspect ratio is not below the blink
    # threshold, so reset the counter and alarm
    else:
        f=open('wf','w')
        f.write('0')
        f.close()
        COUNTER = 0
        ALARM_ON = False
    # draw the computed eye aspect ratio on the frame to help
    # with debugging and setting the correct eye aspect ratio
    # thresholds and frame counters
    cv2.putText(frame, "EAR: {:.2f}".format(ear), (300, 30),
        cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)
 # show the frame
 cv2.imshow("Frame", frame)
 key = cv2.waitKey(1) & 0xFF
# if the `q` key was pressed, break from the loop
 if key == ord("q"):
    break
```

```
# do a bit of cleanup

cv2.destroyAllWindows()

vs.stop()
```

2) Update

```
import import RPi.GPIO as GPIO  #Import GPIO library
import requests
import time     #Import time library
def update():
     payload = {"status":1}
     r = requests.get("http://www.airobosoft.com/2022/dexto/api/update_data.php", params=payload)
     print(r.url)


while True:
 f=open('wf','r')
 wf = f.read()
 print (wf)
 if wf == '1':
  print (wf)
  update()
```

3) Robo:

```
# USAGE
# python detect_drowsiness.py --shape-predictor
shape_predictor_68_face_landmarks.dat
# python detect_drowsiness.py --shape-predictor
shape_predictor_68_face_landmarks.dat --alarm alarm.wav

# import the necessary packages

import time
import RPi.GPIO as GPIO
import pyttsx3
import urllib.request
import requests

m1 = 25
m2 = 8
m3 = 7
m4 = 1
```

```
GPIO_TRIGGER = 23
GPIO_ECHO = 24
GPIO.setwarnings(False)

GPIO.setmode(GPIO.BCM)       # Use BCM GPIO numbers
GPIO.setup(m1, GPIO.OUT)  #
GPIO.setup(m2, GPIO.OUT)
GPIO.setup(m3, GPIO.OUT)
GPIO.setup(m4, GPIO.OUT)
GPIO.setup(GPIO_TRIGGER, GPIO.OUT)
GPIO.setup(GPIO_ECHO, GPIO.IN)

engine = pyttsx3.init()
newVoiceRate = 145
engine.setProperty('rate',newVoiceRate)
voices = engine.getProperty('voices')
engine.setProperty('voice', voices[3].id)
engine.say("A-I based Drowziness detection system")
engine.runAndWait()

def mot_stop():
        GPIO.output(m1, False);
        GPIO.output(m2, False);
        GPIO.output(m3, False);
        GPIO.output(m4, False);

def mot_start():
        GPIO.output(m1, True);
        GPIO.output(m2, False);
        GPIO.output(m3, True);
        GPIO.output(m4, False);

def mot_left():
        GPIO.output(m1, True);
        GPIO.output(m2, False);
        GPIO.output(m3, False);
        GPIO.output(m4, True);

def mot_right():
        GPIO.output(m1, False);
        GPIO.output(m2, True);
        GPIO.output(m3, True);
        GPIO.output(m4, False);
```

```
def mot_back():
            GPIO.output(m1, False);
            GPIO.output(m2, True);
            GPIO.output(m3, False);
            GPIO.output(m4, True);

def distance():
    # set Trigger to HIGH
    GPIO.output(GPIO_TRIGGER, True)

    # set Trigger after 0.01ms to LOW
    time.sleep(0.00001)
    GPIO.output(GPIO_TRIGGER, False)

    StartTime = time.time()
    StopTime = time.time()

    # save StartTime
    while GPIO.input(GPIO_ECHO) == 0:
        StartTime = time.time()

    # save time of arrival
    while GPIO.input(GPIO_ECHO) == 1:
        StopTime = time.time()

    # time difference between start and arrival
    TimeElapsed = StopTime - StartTime
    # multiply with the sonic speed (34300 cm/s)
    # and divide by 2, because there and back
    distance = (TimeElapsed * 34300) / 2

    return distance

# loop over frames from the video stream
while True:
    z=20
    y=distance()
    print("Distance = ")
    print(y)
    if y<30:
        mot_stop()
        print("Alert! Alert!  Obstacle detedted in front")
        engine = pyttsx3.init()
        engine.say("ALERT! Alert! Obstacle detedted in front")
        engine.runAndWait()
        time.sleep(4)
        z=10
    url = "https://airobosoft.com/2022/dexto/api/motor.php"
    user_agent = 'Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.9.0.7)
Gecko/2009021910 Firefox/3.0.7'
    headers={'User-Agent':user_agent,}
    request=urllib.request.Request(url,None,headers) #The assembled request
    response = urllib.request.urlopen(request)
    data = response.read() # The data u need
    data = data.decode()
    d=int

(data)
    print(data)
    print('__________________________________________________')
    if d == 1:
        mot_start()
        print(d)
    elif d == 2:
        mot_back()
        print(d)
    elif d == 3:
        mot_right()
        print(d)
    elif d == 4:
        mot_left()
    elif d == 0:
        mot_stop()
```

5.2 Screenshots & Web Application Images

Terminal Operation

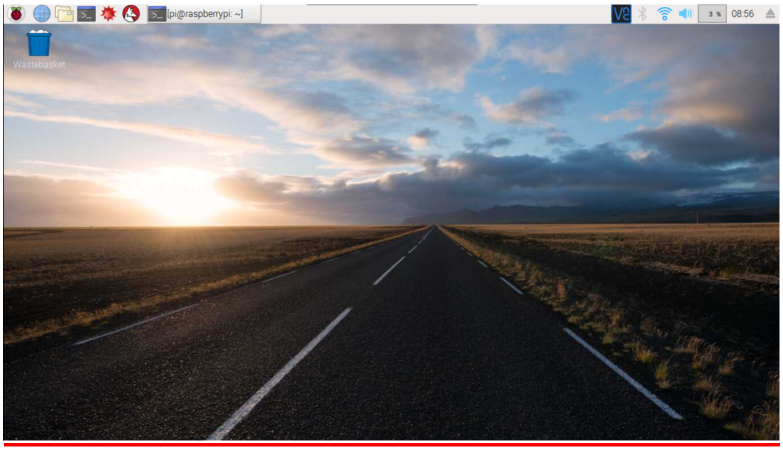


Figure 5.1 - Desktop Raspberry pi

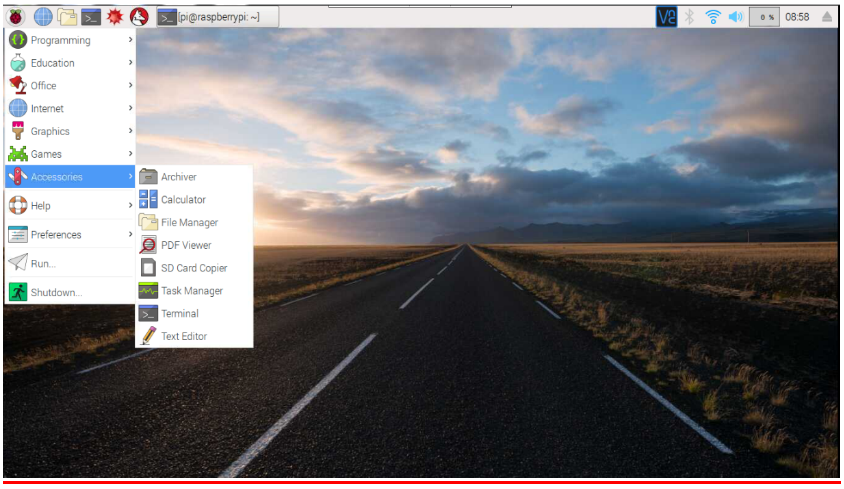
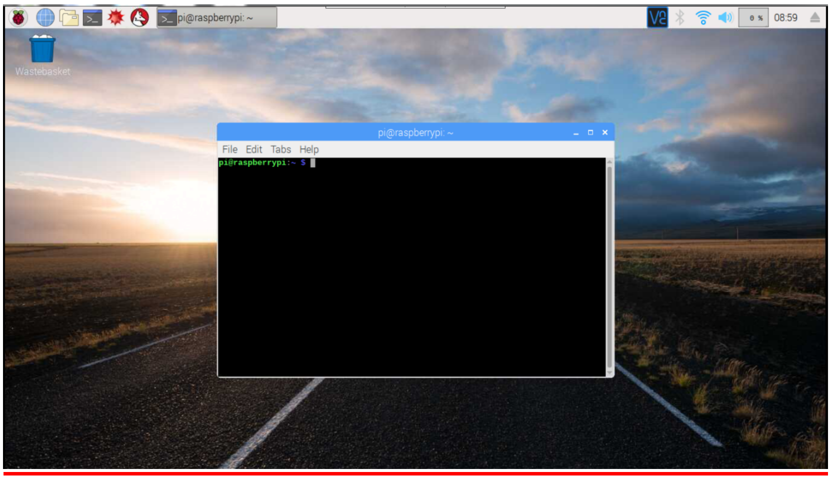


Figure 5.2 - Menu to Open Terminal.



```
pi@raspberrypi:~ $
```

Figure 5.3 - Terminal To Run Python Script

File Manager operation

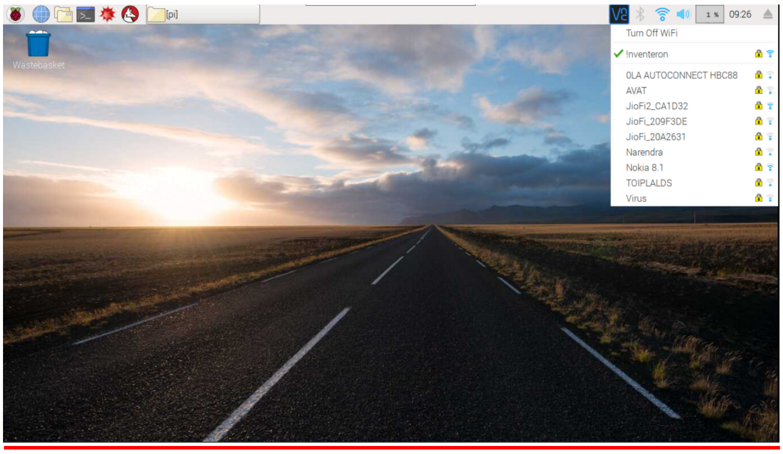


Figure 5.4 - Connect To Wifi.

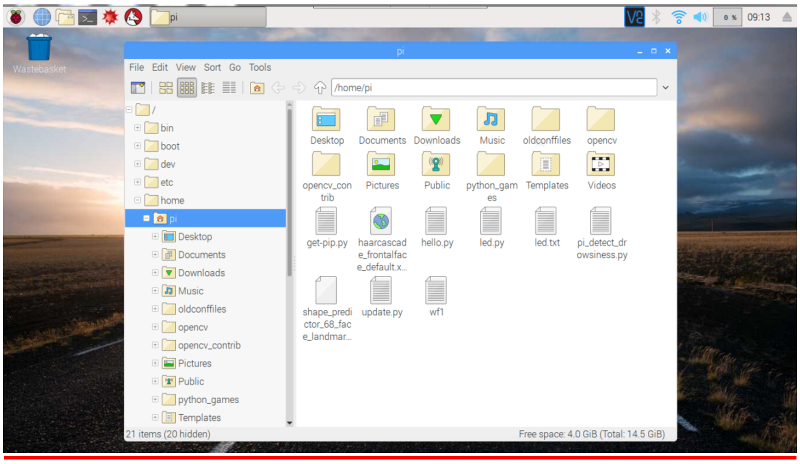
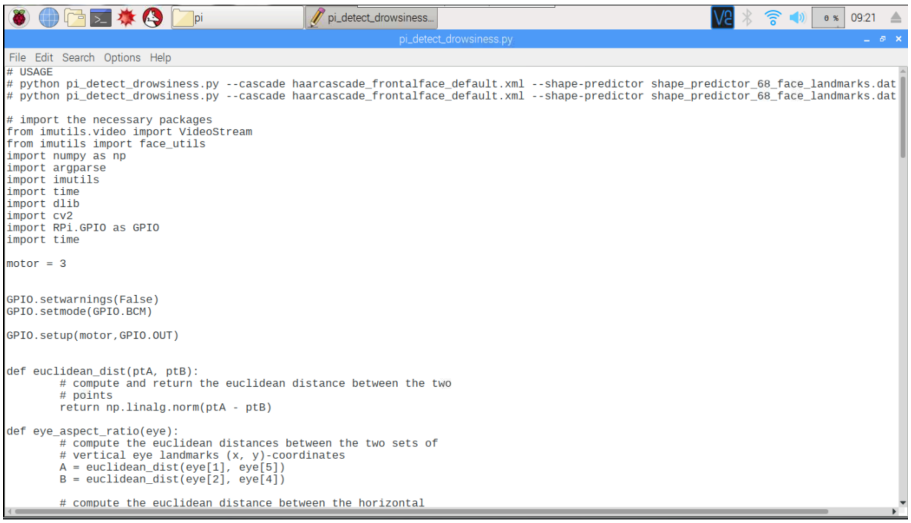


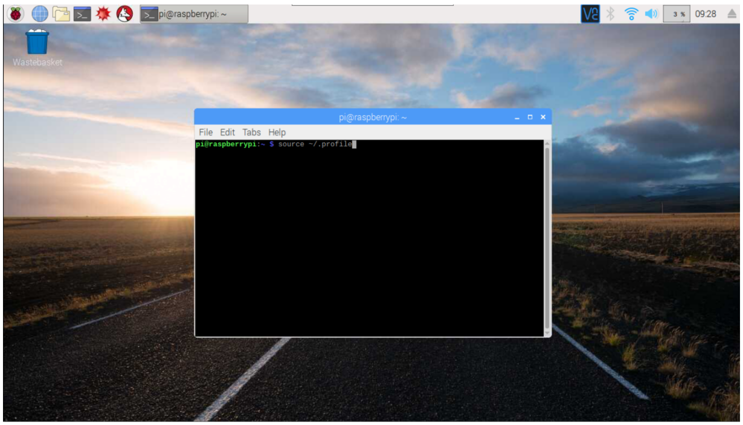
Figure 5.5 - File Manager In Raspbian OS



```
# USAGE
# python pi_detect_drowsiness.py --cascade haarcascade_frontalface_default.xml --shape-predictor shape_predictor_68_face_landmarks.dat
# python pi_detect_drowsiness.py --cascade haarcascade_frontalface_default.xml --shape-predictor shape_predictor_68_face_landmarks.dat

# import the necessary packages
from imutils.video import VideoStream
from imutils import face_utils
import numpy as np
import argparse
import imutils
import time
import dlib
import cv2
import RPi.GPIO as GPIO
import time

motor = 3


GPIO.setwarnings(False)
GPIO.setmode(GPIO.BCM)

GPIO.setup(motor,GPIO.OUT)


def euclidean_dist(ptA, ptB):
        # compute and return the euclidean distance between the two
        # points
        return np.linalg.norm(ptA - ptB)

def eye_aspect_ratio(eye):
        # compute the euclidean distances between the two sets of
        # vertical eye landmarks (x, y)-coordinates
        A = euclidean_dist(eye[1], eye[5])
        B = euclidean_dist(eye[2], eye[4])

        # compute the euclidean distance between the horizontal
```

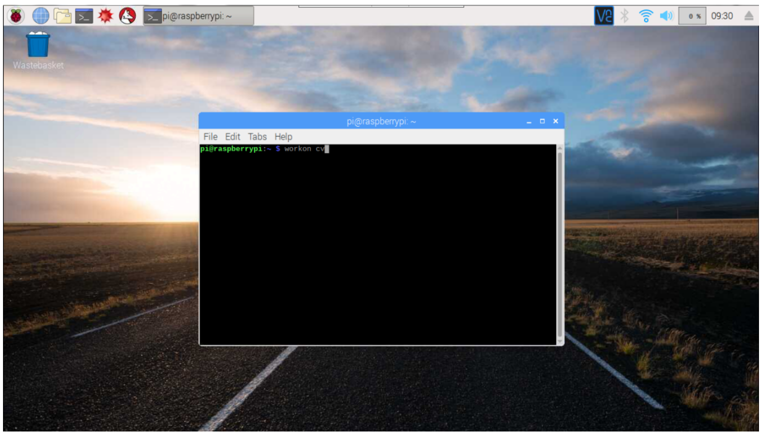
Figure 5.6 - Python File For Detecting Drowsiness.

Run The Code



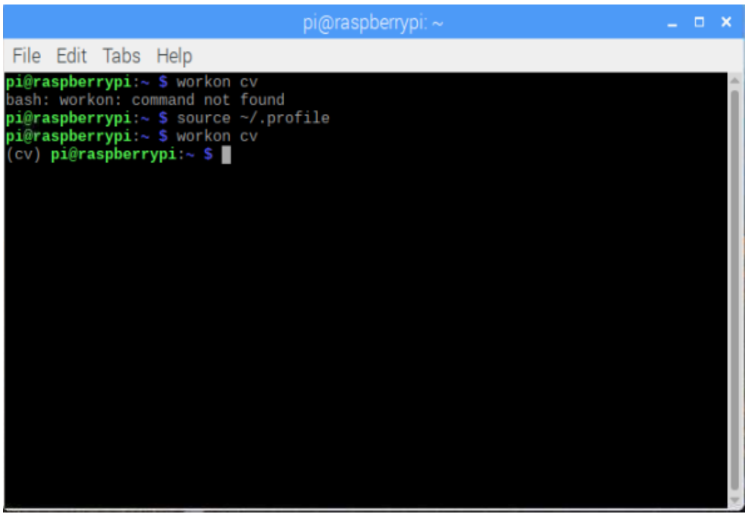
```
pi@raspberrypi:~ $ source ~/.profile
```

Figure 5.7 - Source The Profile Folder For Libraries.



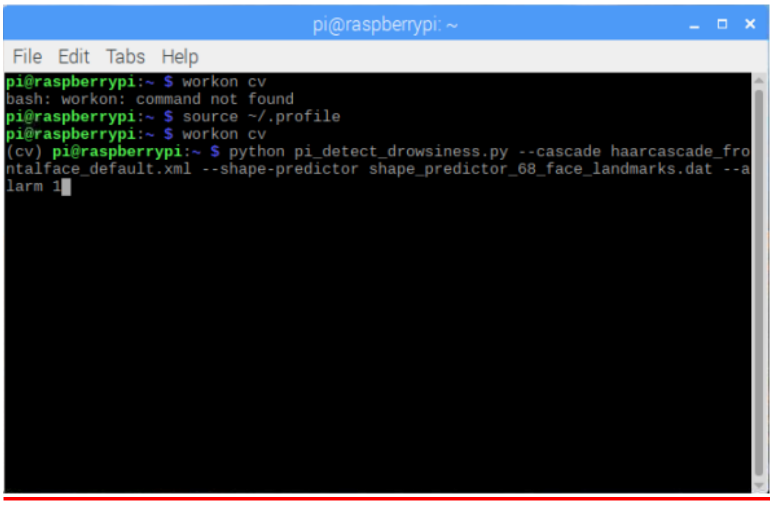
```
pi@raspberrypi:~ $ workon cv
```

Figure 5.8 - Start The Open Cv Environment.



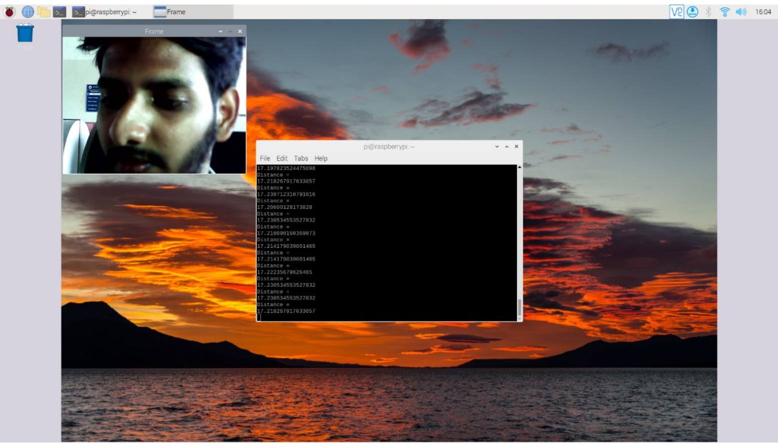
```
pi@raspberrypi:~ $ workon cv
bash: workon: command not found
pi@raspberrypi:~ $ source ~/.profile
pi@raspberrypi:~ $ workon cv
(cv) pi@raspberrypi:~ $ 
```

Figure 5.9 - Entered The Environment.



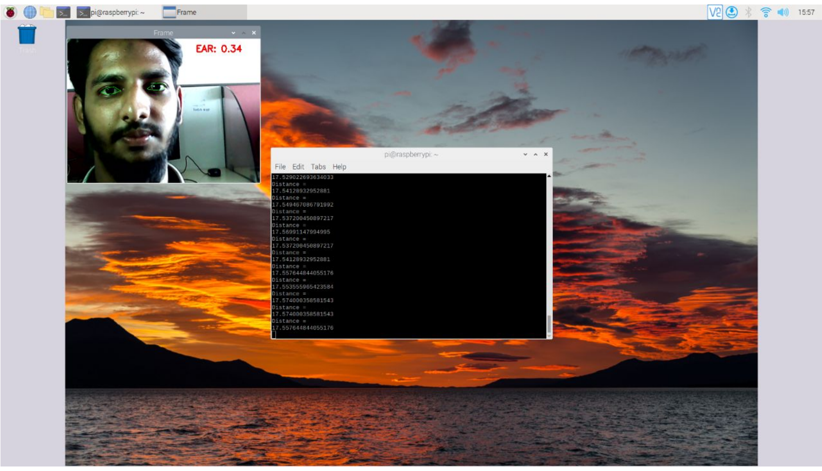
```
pi@raspberrypi:~ $ workon cv
bash: workon: command not found
pi@raspberrypi:~ $ source ~/.profile
pi@raspberrypi:~ $ workon cv
(cv) pi@raspberrypi:~ $ python pi_detect_drowsiness.py --cascade haarcascade_frontalface_default.xml --shape-predictor shape_predictor_68_face_landmarks.dat --alarm 1
```

Figure 5.10 - Run The Open Cv File.



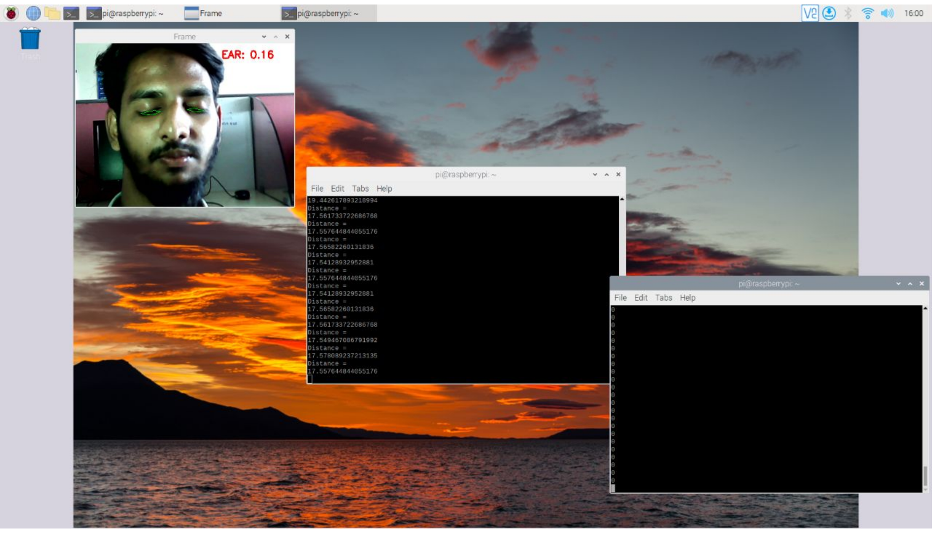
```
17.197823524475098
Distance =
17.218267917633057
Distance =
17.238712310791016
Distance =
17.20600128173828
Distance =
17.230534553527832
Distance =
17.210090160369873
Distance =
17.214179039001465
Distance =
17.214179039001465
Distance =
17.22235679626465
Distance =
17.230534553527832
Distance =
17.230534553527832
Distance =
17.218267917633057
```

Figure 5.11 - video processing started.PNG



```
17.529022693634033
Distance =
17.54128932952881
Distance =
17.549467086791992
Distance =
17.537200450897217
Distance =
17.56991147994995
Distance =
17.537200450897217
Distance =
17.54128932952881
Distance =
17.557644844055176
Distance =
17.553555965423584
Distance =
17.574000358581543
Distance =
17.574000358581543
Distance =
17.557644844055176
```

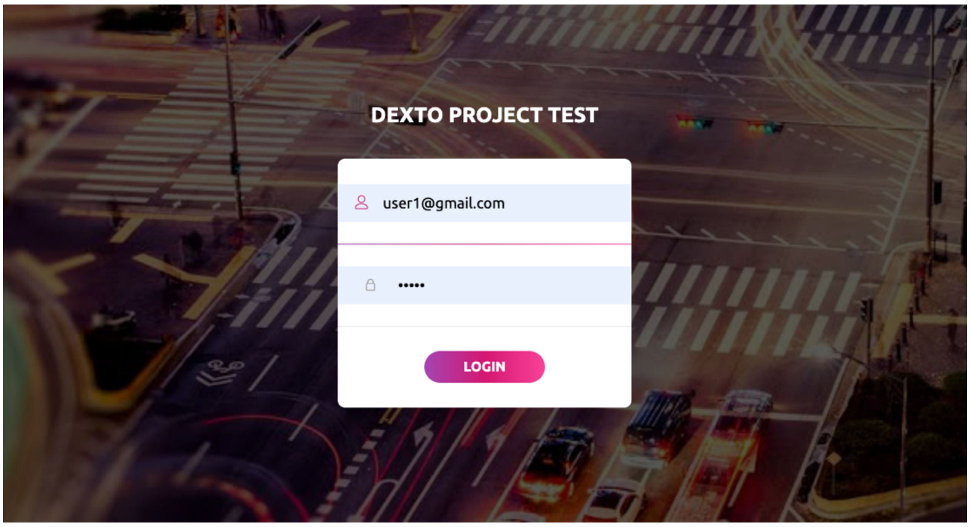
Figure 5.12 - E.A.R (eye aspect ratio).



```
19.442617893218994
Distance =
17.561733722686768
Distance =
17.557644844055176
Distance =
17.56582260131836
Distance =
17.54128932952881
Distance =
17.557644844055176
Distance =
17.54128932952881
Distance =
17.56582260131836
Distance =
17.561733722686768
Distance =
17.549467086791992
Distance =
17.578089237213135
Distance =
17.557644844055176
```

```
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
```

Figure 5.13 - Drowsiness detected.



user1@gmail.com

•••••

Figure 5.14 - Web application login page

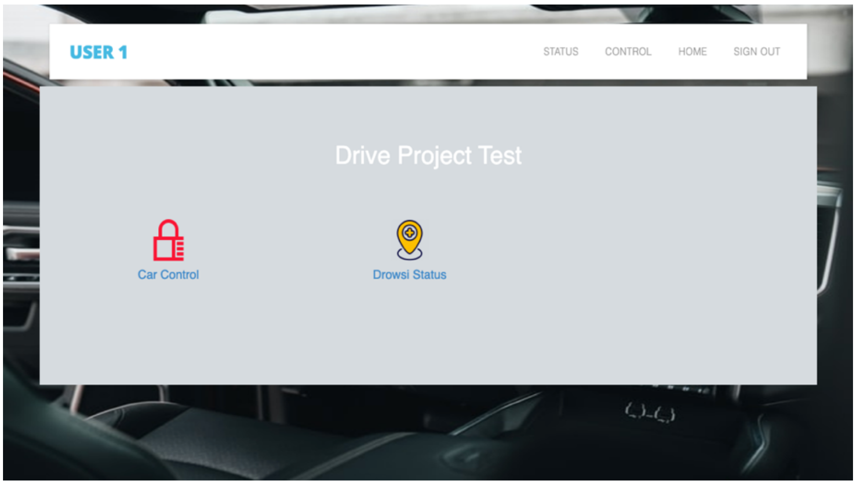
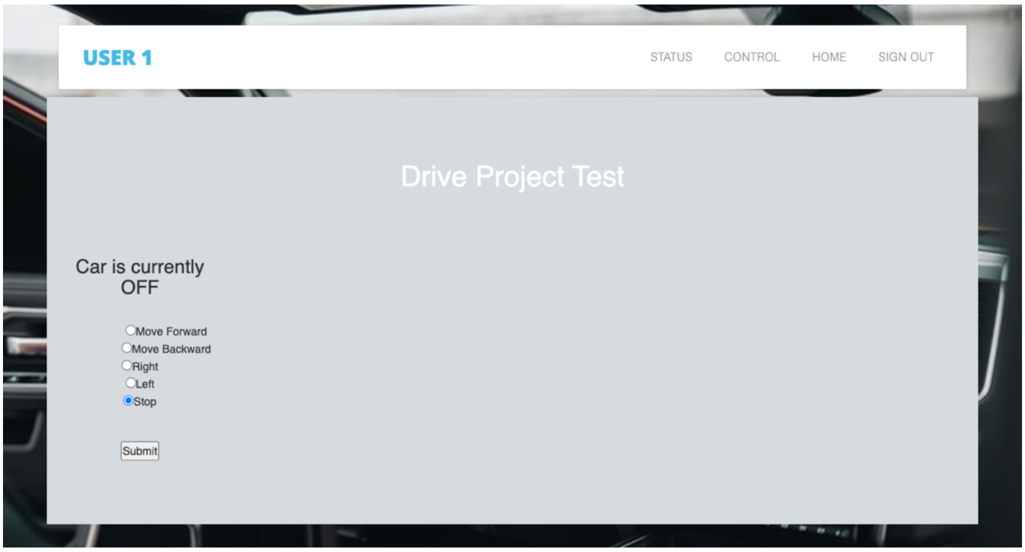


Figure 5.15 - Web application Attributes

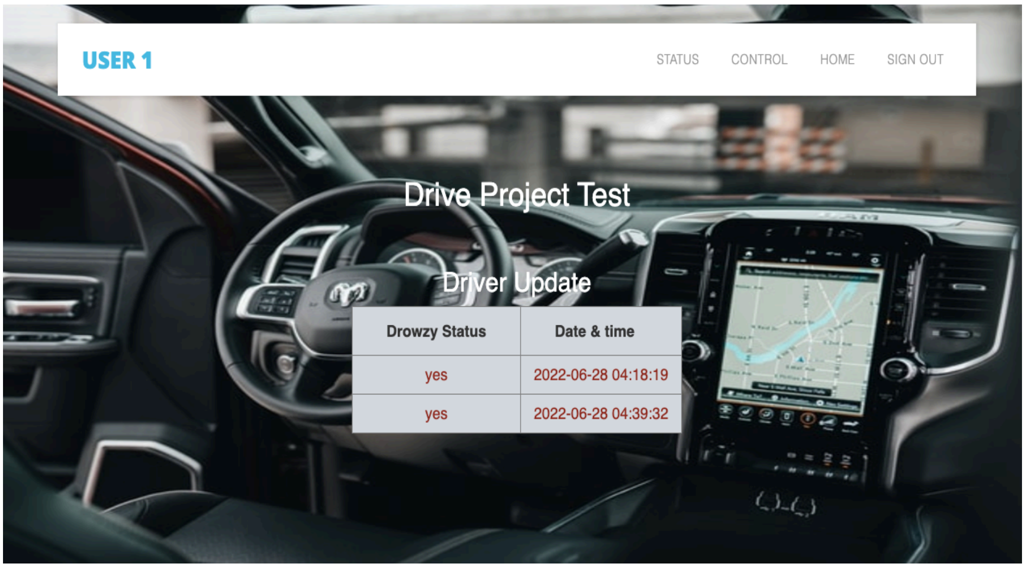


Drive Project Test

Car is currently OFF

○ Move Forward
○ Move Backward
○ Right
○ Left
● Stop

Figure 5.16 - Prototype Control buttons in the web application



Drive Project Test

Driver Update

Drowzy Status	Date & time
yes	2022-06-28 04:18:19
yes	2022-06-28 04:39:32

Figure 5.17 - Drowsiness status update on the web application.

Chapter 6

RESULTS AND DISCUSSIONS

6.1 Results:

The system is capable of accurate positioning eye point. Using four parameters of eye states can effectively detect the driver's fatigue status. In order to improve the accuracy grade, our system should be using some other methods as a supplementary means, such as: Road tracking, Vehicle theft intimation, Accident theft intimation.

Software testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. Software testing can also provide an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation. Test techniques include the process of executing a program or application with the intent of finding software bugs (errors or other defects).

It involves the execution of a software component or system component to evaluate one or more properties of interest. In general, these properties indicate the extent to which the component or system under test:

- meets the requirements that guided its design and development,
- responds correctly to all kinds of inputs,
- performs its functions within an acceptable time,
- is sufficiently usable,
- can be installed and run in its intended environments, and
- Achieves the general result its stakeholder’s desire.

6.2 Test Cases:

The test procedure used in testing process is black box testing or functional coding. Test cases are analyzed accordingly. This involves manual evaluation of the flow from one page to other and check accordingly for the process flow.

Test Cases	Check Item	Objective	Execution	Test Data	Expected Result	Actual Output
1	Login Page	Fields are Empty	Login		"Fill the correct data"	Yes
2	username	Invalid username	Login	admin@gmail.com	"username or pwd is wrong"	No
3	Password	Valid	Login	******	"Login successful"	Yes

Table 6.1: Test- case

Adding Packages:

Test Cases	Check Item	Objective	Execution	Test Data	Expected Result	Actual Output
1	Add package page	Fields are Empty	Save		"Fill the data"	No
2	Add package page	Packagename added	Save	Switzerland	"successfully added"	Yes

Table 6.2: Adding Packages

Update Users:

Test Cases	Check Item	Objective	Execution	Test Data	Expected Result	Actual Output
1	Select user	Fields are Empty and valid entries	show	Admin selected	Show admin details	Yes
2	Password	Enter valid password	update	New-valid password	Accepted	Yes
3	Confirm pasword	Enter Valid password	update	Same password as above	Accepted	Yes
4	Type of user	Select type of user	update	General/admin	Succesfully Updated	Yes

Table 6.3: Update user

6.3 Prototype Images:

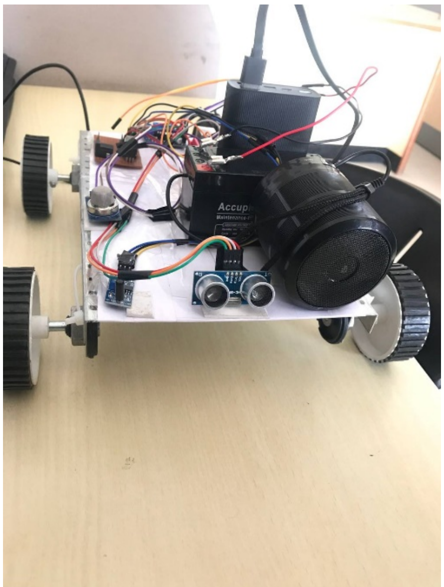


Fig 6.1: Prototype front-view

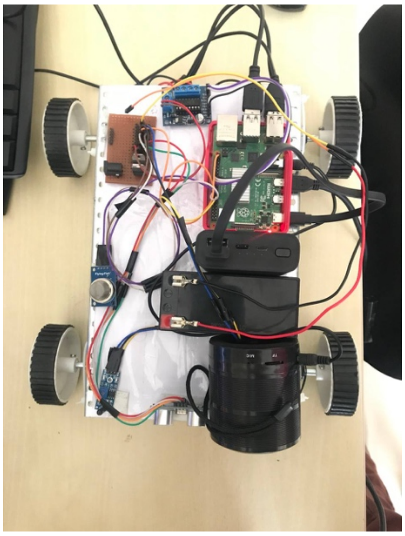


Fig 6.2: Prototype top-view

Chapter 7

CONCLUSIONS AND FUTURE SCOPE OF WORK

7.1 Conclusion

The system is capable of accurate positioning eye point. Using four parameters of eye states can effectively detect the driver's fatigue status. In order to improve the accuracy grade, our system should be using some other methods as a supplementary means, such as

- Road tracking,
- Head position
- The rotation rate and the grip force of the steering wheel, which are the main directions to improve our system accuracy.

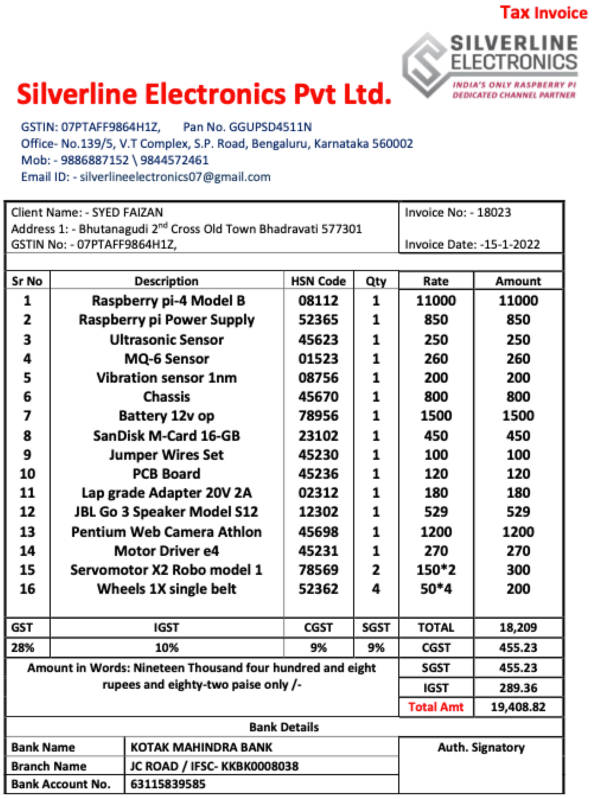
7.2 Future Scope of Work

A non-intrusive system to localize the eyes and monitor fatigue was developed. Information about the head and eyes position are obtained through various self-developed image processing algorithms. During the monitoring, the system is able to decide whether the eyes are opened or closed. When the eyes have been closed for two seconds, a warning signal is issued. In addition during monitoring, the system is able to automatically detect any eye localizing error that might have occurred. In case of this type of error, the system is able to recover and properly localize the eyes. The proposed system was tested on the real driver images. The video image [480 x 640 pixels] of 75 different test persons has been recorded during several day, night and complex background at different places. The proposed system has two key phases such as preprocessing and detecting eye from video images described in Chapter 5 and 6 respectively. In preprocessing, new enhanced technique is used to enhance the contrast of dark regions and tested with existing algorithm. As per the results obtained in section 5.5, all the noises in the video image are removed successfully. In second phase new techniques are used to extract eye from the preprocessed image. It is also tested with standard existing method and the comparison results are shown in Table 6.1 and 6.2. The eye pair can be selected successfully in most cases, shown in Fig 6.1, 6.2 and 6.3. From the Chapter 6, the false finding rate of drowsiness of Color cue and projection function, BTMED and KDKBM is 21.7%, 14.7% and 10.3%. Since the false finding rate is only 10.3%, the proposed system is effective than other methods. Please purchase PDF Split-Merge on www.verypdf.com to remove this watermark. 93 We achieved the following: • DDDS achieves highly accurate and reliable detection of drowsiness. • DDDS offers a non-intrusive approach to detect drowsiness without the annoyance and interference. • Processing, judges the driver's alertness level on the basis of continuous eye closures. • The proposed system works in both day time and night time conditions. All the drawbacks mentioned in section 2.5 have been eliminated. In future, this prototype can be extended to give alarm before sleeping by calculating the heart beat measure without physical disturbance i.e., non-intrusive method using modified ECG methods. Usually in ECG method key points of body (For example chest, head, wrist etc.,) are sticked with wire. In the extended method, sticking wire may be avoided. This will lead us to a way to find out the optimum level of drowsiness. Further, this prototype will be extended to monitor the reflect ray from eye using nano camera. If the reflection ray is absent, then eye is closed otherwise eye is opened. We believe that this will create a better opportunity to detect drowsiness.

Chapter 8

COSTING

8.1 BILLS



Tax Invoice

SILVERLINE ELECTRONICS
INDIA'S ONLY RASPBERRY PI DEDICATED CHANNEL PARTNER

Silverline Electronics Pvt Ltd.

GSTIN: 07PTAFF9864H1Z, Pan No. GGUPSD4511N
Office- No.139/5, V.T Complex, S.P. Road, Bengaluru, Karnataka 560002
Mob: - 9886887152 \ 9844572461
Email ID: - silverlineelectronics07@gmail.com

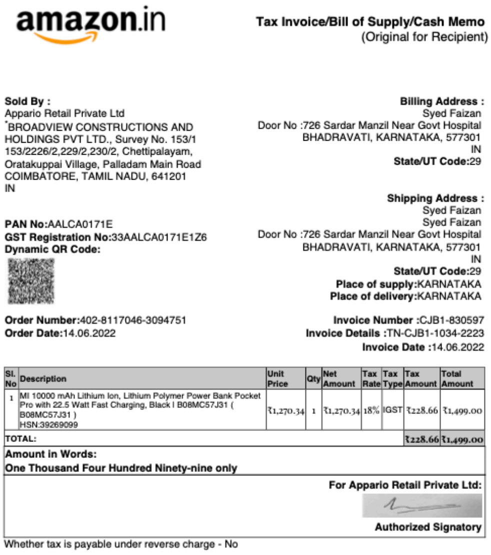
Client Name: - SYED FAIZAN Address 1: - Bhutanagudi 2nd Cross Old Town Bhadravati 577301 GSTIN No: - 07PTAFF9864H1Z,	Invoice No: - 18023 Invoice Date: -15-1-2022

Sr No	Description	HSN Code	Qty	Rate	Amount
1	Raspberry pi-4 Model B	08112	1	11000	11000
2	Raspberry pi Power Supply	52365	1	850	850
3	Ultrasonic Sensor	45623	1	250	250
4	MQ-6 Sensor	01523	1	260	260
5	Vibration sensor 1nm	08756	1	200	200
6	Chassis	45670	1	800	800
7	Battery 12v op	78956	1	1500	1500
8	SanDisk M-Card 16-GB	23102	1	450	450
9	Jumper Wires Set	45230	1	100	100
10	PCB Board	45236	1	120	120
11	Lap grade Adapter 20V 2A	02312	1	180	180
12	JBL Go 3 Speaker Model S12	12302	1	529	529
13	Pentium Web Camera Athlon	45698	1	1200	1200
14	Motor Driver e4	45231	1	270	270
15	Servomotor X2 Robo model 1	78569	2	150*2	300
16	Wheels 1X single belt	52362	4	50*4	200
GST	IGST	CGST	SGST	TOTAL	18,209
28%	10%	9%	9%	CGST	455.23
Amount in Words: Nineteen Thousand four hundred and eight rupees and eighty-two paise only /-				SGST	455.23
				IGST	289.36
				Total Amt	19,408.82

Bank Details

Bank Name	KOTAK MAHINDRA BANK	Auth. Signatory
Branch Name	JC ROAD / IFSC- KKBK0008038	
Bank Account No.	63115839585	

Fig 8.1: Bill 1



amazon.in

Tax Invoice/Bill of Supply/Cash Memo
(Original for Recipient)

Sold By :
Appario Retail Private Ltd
*BROADVIEW CONSTRUCTIONS AND HOLDINGS PVT LTD., Survey No. 153/1 153/2226/2,229/2,230/2, Chettipalayam, Oratakuppai Village, Palladam Main Road
COIMBATORE, TAMIL NADU, 641201
IN

Billing Address :
Syed Faizan
Door No :726 Sardar Manzil Near Govt Hospital
BHADRAVATI, KARNATAKA, 577301
IN
State/UT Code:29

Shipping Address :
Syed Faizan
Syed Faizan
Door No :726 Sardar Manzil Near Govt Hospital
BHADRAVATI, KARNATAKA, 577301
IN
State/UT Code:29
Place of supply:KARNATAKA
Place of delivery:KARNATAKA

PAN No:AALCA0171E
GST Registration No:33AALCA0171E1Z6
Dynamic QR Code:

Order Number:402-8117046-3094751
Order Date:14.06.2022

Invoice Number :CJB1-830597
Invoice Details :TN-CJB1-1034-2223
Invoice Date :14.06.2022

Sl. No	Description	Unit Price	Qty	Net Amount	Tax Rate	Tax Type	Tax Amount	Total Amount
1	MI 10000 mAh Lithium Ion, Lithium Polymer Power Bank Pocket Pro with 22.5 Watt Fast Charging, Black I B08MC57J31 (B08MC57J31) HSN:39269099	₹1,270.34	1	₹1,270.34	18%	IGST	₹228.66	₹1,499.00
TOTAL:							₹228.66	₹1,499.00

Amount in Words:
One Thousand Four Hundred Ninety-nine only

For Appario Retail Private Ltd:

Authorized Signatory

Whether tax is payable under reverse charge - No

Fig 8.2: Bill 2