SEAT BELT CONTROL SYSTEM: DROWSINESS DETECTOR SYSTEM

NUR AISYAH BINTI ABDUL RAHMAN

Bachelor of Engineering Technology (Electrical)
With Honours

UNIVERSITI MALAYSIA PAHANG

UNIVERSITI MALAYSIA PAHANG

DECLARATION OF THESIS AND COPYRIGHT

Author's Full Name	:	**NUR AISYAH BINTI ABDUL RAHMAN**
Date of Birth	:	
Title	:	**SEAT BELT CONTROL SYSTEM: DROWSINESS DETECTOR SYSTEM**
Academic Session	:	**SEMESTER 1 2021/2022**

I declare that this thesis is classified as:

☐	CONFIDENTIAL	(Contains confidential information under the Official Secret Act 1997)*
☐	RESTRICTED	(Contains restricted information as specified by the organization where research was done)*
☐	OPEN ACCESS	I agree that my thesis to be published as online open access (Full Text)

I acknowledge that Universiti Malaysia Pahang reserves the following rights:

1. The Thesis is the Property of Universiti Malaysia Pahang
2. The Library of Universiti Malaysia Pahang has the right to make copies of the thesis for the purpose of research only.
3. The Library has the right to make copies of the thesis for academic exchange.

Certified by:

_______________________	_______________________
(Student's Signature)	(Supervisor's Signature)
_______________________	TS. DR. ZAINAH BINTI MD. ZAIN
New IC/Passport Number	Name of Supervisor
Date: 18 FEB. 2022	Date: 18 FEB. 2022

NOTE : * If the thesis is CONFIDENTIAL or RESTRICTED, please attach a thesis declaration letter.

THESIS DECLARATION LETTER (OPTIONAL)

Librarian,
Perpustakaan Universiti Malaysia Pahang,
Universiti Malaysia Pahang,
Lebuhraya Tun Razak,
26300, Gambang, Kuantan.

Dear Sir,

CLASSIFICATION OF THESIS AS RESTRICTED

Please be informed that the following thesis is classified as RESTRICTED for a period of three (3) years from the date of this letter. The reasons for this classification are as listed below.

Author's Name

Thesis Title

Reasons (i)

(ii)

(iii)

Thank you.

Yours faithfully,

____
(Supervisor's Signature)

Date: 22 FEB. 2022

Stamp: TS. DR. ZAINAH BINTI MD ZAIN
PENSYARAH KANAN
FAKULTI TEKNOLOGI KEJURUTERAAN ELEKTRIK DAN ELEKTRONIK
UNIVERSITI MALAYSIA PAHANG
26600 PEKAN, PAHANG

Note: This letter should be written by the supervisor, addressed to the Librarian, *Perpustakaan Universiti Malaysia Pahang* with its copy attached to the thesis.

MAKLUMAT PANEL PEMERIKSA PEPERIKSAAN LISAN

(only for Faculty of Computer's student)

Thesis ini telah diperiksa dan diakui oleh
This thesis has been checked and verified by

Nama dan Alamat Pemeriksa Dalam :
Name and Address Internal Examiner

Nama dan Alamat Pemeriksa Luar :
Name and Address External Examiner

Nama dan Alamat Pemeriksa Luar :
Name and Address External Examiner

Disahkan oleh Penolong Pendaftar IPS :
Verified by Assistant Registrar IPS

Tandatangan :
Signature

Tarikh :
Date

Nama :
Name

SUPERVISOR'S DECLARATION

I hereby declare that I have checked this thesis, and, in my opinion, this thesis is adequate in terms of scope and quality for the award of the degree of Bachelor of Engineering Technology in (Electrical) With Hons.

____

(Supervisor's Signature)

Full Name : TS. DR. ZAINAH BINTI MD. ZAIN

Position : SENIOR LECTURER

Date : 18 FEB. 2022

____

(Co-supervisor's Signature)

Full Name :

Position :

Date :

STUDENT'S DECLARATION

I hereby declare that the work in this thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Universiti Malaysia Pahang or any other institutions.

____

(Student's Signature)

Full Name : NUR AISYAH BINTI ABDUL RAHMAN

ID Number : TB18023

Date : 18 FEB. 2022

SEAT BELT CONTROL SYSTEM:
DROWSINESS DETECTOR SYSTEM

NUR AISYAH BINTI ABDUL RAHMAN
TB18023

Thesis submitted in fulfillment of the requirements
for the award of the degree of
Bachelor Of Engineering Technology (Electrical) With Honours

Faculty of Electrical & Electronics Engineering
UNIVERSITI MALAYSIA PAHANG

22 FEBRUARY 2022

ACKNOWLEDGEMENTS

Assalamualaikum w.b.t and greetings,

In the name of Allah the most gracious and the most merciful. First and foremost, I want to express my gratitude to Allah S.W.T for granting me the patience, strength, and courage necessary to complete this project and giving bestowed for me to complete my thesis. Throughout to complete this project, I have obtained various unforgettable experiences and knowledge that were valuable to our life as students.

I owed a huge debt of gratitude to my supervisor, TS. DR Zainah Binti Md Zain, for helping me to accomplish this thesis and project. She is a calm and considerate supervisor. Her suggestions and advice, as well as her support were important to me.

Consequently, I owe a debt of gratitude to Ahmad Hapizal and Mohammad Firdaus, who are both members of my group. Their insightful advice and support reassured me that I would not be alone in completing this project. Throughout the millions of their efforts, we won't be able to complete this project flawlessly without the commitment and tolerance of the teammates. I dedicate a heartfelt thanks to my classmates, who have always been kind and helpful in discussing ideas and giving a pleasant learning environment.

To my parents, Noriah binti Kadar and Abdul Rahman bin Mokhtar, deserve special thanks for their emotional support, unconditional love, and advice during my anxious and restless phase. Thank you to my brother also lent his hand by offering suggestions and ideas.

ABSTRAK

Dalam era permodenan ini, aspek keselamatan telah menjadi keutamaan pada setiap kenderaan yang dihasilkan. Natijahnya, setiap pereka dan pakar automotif telah menjadikan perkara tersebut menjadi keutamakan bagi setiap penghasilan. Hal ini kerana, jumlah pemandu kenderaan yang terkorban dalam kemalangan jalan raya meningkat kerana tidak memakai tali pinggang keledar dan keletihan serta mengantuk. Sejajar dengan itu, penggunaan tali pinggang keledar mampu mengurangkan kadar kemalangan dan boleh mengelakkan kemalangan maut kepada penumpang. Dari hasil penyelidikan menunjukkan pengurangan sebanyak 60% kepada 44% pada setiap penggunaan tali pinggang keledar. Teknologi ini terdiri daripada hubungan antara tiga sub-sistem yang dibangunkan dalam prototaip. Pertama, kami memilih "*magnetic reed switch*" yang dipasangkan didalam tali pinggang keledar sebagai pengesan tali pinggang keledar. Kemudian, apabila tali pinggang keledar diikat dan dibuka, ia akan mengukur keluaran tinggi dan rendah. Jika tali pinggang keledar dibuka selama 10 saat, penggera akan berbunyi, dan 5volt motor akan mulai perlahan, sementara sistem disambungkan bersama pengesan alkohol. Sensor MQ3 telah dipasang pada paparan skrin "*oled*" untuk mengesanan alkohol. Apabila kandungan alkohol melebihi tahap normal, paparan akan membaca kepekatan alkohol. Selepas itu, kelajuan motor akan berkurangan secara beransur-ansur. Berikutan itu ialah sistem yang memberi amaran kepada pemandu apabila ia mengesan perubahan pada tingkah laku muka, menunjukkan bahawa pemandu mengantuk. Selain itu, untuk sistem mengantuk, memerlukan proses pemasangan kamera yang membolehkan pengesanan siang dan malam serta pemberi amaran. Sistem ini akan mengesan pergerakan mata dan mulut keadaan tertutup dan terbuka. Oleh itu, "*shape predictor facial landmark*" dan "*Haar cascade clarifier*" algoritma telah digunakan. Tambahan pula, sistem pengesanan kenderaan direka untuk mengesan lokasi kenderaan dan menghantar maklumat melalui mesej kepada pewaris sekiranya "*limit switch*" diaktifkan. Tambahan itu, "*String $GPGGA*" adalah salah satu algoritma yang digunakan, ia untuk mengekstrak koordinat daripada GPS untuk penerima, ia menghantarnya ke GSM dari segi koordinat seperti latitud dan longitud algoritma terlibat. Pada kesimpulannya, sistem yang dibangunkan boleh dilaksanakan dalam model kenderaan lama untuk mengurangkan kadar kemalangan dan kematian di Malaysia, serta meningkatkan keselamatan pada pemandu.

ABSTRACT

Throughout this era of transformation, the safety of every vehicle produced has become a highest concern. Consequently, every automotive design engineer and specialist has made this a primary focus for all future productions. This is due to rising in the percentage of vehicle drivers died in road accidents as a matter of fact with not wearing a seat belt, fatigue as well as drowsiness. In accordance with this, the use of seat belts can reduce number of road accidents and decrease fatalities among passengers. According to the collected data, every use of seat belt significantly reduces the risk by 60% to 44%. Several initiatives have been undertaken in this regard to develop vehicle safety systems. This technology is consisted of interlink between three sub-system develop in prototype. Firstly, for seat belt detection system we preferred magnetic reed switches fitted within the seat belt buckle for the seat belt detector. Then, as the seat belt is buckled and unbuckled, it will measure a high and low output. If the seat belt is unbuckled for 10 seconds, an alarm will be triggered, and the 5V motor will begin to slow down, all while the system is linked to alcohol detectors system. The MQ3 Sensor was attached to an oled screen display for the alcohol detection. When the alcohol content exceeds the level of sobriety, the display will read alcohol concentration. After that, the speed of motor would gradually decrease. Following that is a system that alerts the driver when it detects a change in facial behaviours, indicating that the driver is drowsy. In addition, for the drowsiness system, entails the process of camera installations that enable day and night detection as well the alert notification. This system will detect closed and open eye and mouth movements. Thus, the facial landmark shape detector and Haar cascaded clarified algorithms are applied. Furthermore, the vehicle tracking system is designed to trace the location of the vehicle and transmit information through message to the heirs contact in the case of triggered the limit switch. Thus, the $GPGGA String is one of algorithm that has applied, it is to extract coordinates from the GPS receiver and sends to GSM in terms of coordinates as latitude and longitude. At the end, the developed system can be implemented in older models of vehicles to decrease the rate of accidents and fatalities in Malaysia, as well improve safety of the driver.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

EAR	Eye Aspect Ratio
MAR	Mouth Aspect Ratio
OpenCV	Open-source Computer Vision

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Background

Seat belt is a safety belt for vehicles that has been designed to secure the driver and passenger from unexpected accidents in such a sudden stop situation. Besides that, seat belts as a safety feature can reduce the risk of serious injury for the driver or passenger in accidents. According to Vocabulary.com Dictionary, seat belts are a safety belt used in a car or plane to hold you in your seat in case of an accident and a belt attaching you to some object as a restraint to prevent you from getting hurt. As stated in Merriam-Webster, a seat belt is an arrangement of straps designed to hold a person steady in a seat as in an airplane or automobile. Because it can prevent them from being ejected during a crash by reducing the force of the impact on the ejected part of the head and body. Thus, wearing a seat belt will properly help them to keep it held in place.

In several research, people not wearing seat belts will be thrown into various parts of the interior of the vehicle or thrown completely out of the vehicle causing serious injury and death. According to [1] seat belt use will decrease fatal and non-fatal accidents in front and rear seat passengers by 60% and 44%, respectively, according to research. This article it has been discovered that drivers who do not wear seat belts are 8.3 times more likely to suffer a fatal injury and 5.2 times more likely to suffer a moderate injury than drivers who do wear seat belts. Moreover, the seat belt is made of a webbed fabric that is strong and flexible to allow a small movement when worn properly, but for it to protect a person in a crash it needs to be a tight fit. Therefore, [2] the use of seat belts was deemed one of the most effective ways in Malaysia and consequently the mandatory seat belt law was enforced in the early seventies.

In our projects we are focusing on safety in seat belt control system usage for the driver of the vehicle. Besides that, vehicles nowadays have a reminder system for those not wearing seat belts. As a result, these vehicle monitoring systems will be created to alert the driver and third party and develop drowsiness detection systems that can function in various scenarios, as opposed to cars that do not have this system. The system is to reduce the road accidents that cause damage to property as well as life.

1.2 Problem Statement

In Malaysia, road accidents are one of the leading causes of death and injury. Even if the vehicles are in great condition, human errors can often result in fatal results. Driver not wearing a belt, drowsiness, driving in the state of alcohol influence is the main cause of road accidents in Malaysia and as elsewhere in the world [3]. In addition, all automobiles have a seat belt system, but only the most recent models include a seat belt Alert System to ensure the driver wears the seat belt while driving. Because the old car does not have such a complex system, some drivers take it easy by not wearing a seat belt, compromising their safety while driving. Furthermore, drivers lose their control of the vehicle when they are feeling sleepy or high alcohol consuming, which may cause of a car collision. Therefore, drivers who are sleepy are also contributing to traffic accidents. Thus, this is more common when the drivers are driving alone. When the heiress has no information where the accident occurred, the scenario becomes even more difficult.

1.3 Objective

The objectives for these projects are to develop the control system for seat belt usability in a different situation with a system that has been established to a fully monitoring system. This can be achieved through the following objectives:

i. To develop a prototype of the monitoring system that can detect a seat belt and alcohol level for the driver.

ii. To develop a real-time eye-detecting system to detect the drowsy driver.

iii. To develop a real-time GPS system for car tracking.

1.4 Research Scope

The scope of study in this project is focused on the implementation of seat belt control systems and this project required a defined scope of work and a well thought strategy to be completed to meet the objective.

i. To provide a reliable monitoring system for the project when the driver is drunk or not fastened the seat belt with the system in the car which connected to the relay and car battery that can open connection of the battery spark plug.

ii. Provide a value of data collection to eye and mouth identification with to be used for multiple humans. It also notifies the user when the drowsiness value is reached at threshold value.

iii. To provide a monitoring and tracking vehicle system that can be accessed from multiple users.

Integration of selected approaches and coding development of three interconnected components that is seat belt and alcohol detection, drowsiness, and vehicular tracking system.

1.5 Thesis Organization

This thesis is divided into three sections. The goals and scope of the general study with software and electronics design are described in Chapter 1 together with the developing system of drowsiness detector system, the problems statement of issues, the objective project to be completed, and the goals and scope of the general study with software and electronics design.

Chapter 2 explains the study's goal, outlines the project's background, and contains a literature review to demonstrate the need for this analysis. An explanation of the preparation drowsiness detector system, research of the face landmark systems, a comparison of methods for constructing the face detector and the software to be utilised in the system's development. The approaches that have been examined and recommended in this study will be explained in

Chapter 3. Further investigation of the performance evaluation for the creation of a monitoring and control system drowsiness system, which can detect the fatigue behaviour of the driver. This system processes in real-time video, it can alert the driver when in the system is trigger.

The findings and discussion are presented in the following section of Chapter 4. Every goal that must be achieved has a conclusion. Chapter 5 describes the results of the experiments that were discussed and completed.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

The prior research has provided insight into way of the work will be implemented. The simplest approach to avoid accidents will be to detecting the drowsiness and notify the driver. This technology operates through analysing the driver's eye closed distance and yawn opened distance with warning the driver by triggering the audio alert. Facial landmark for clarifying attentiveness were generated with real-time videoing the driver's face which used a camera. The technology will classify initial phases of drowsiness and length the durations during drowsiness ranging up to 3 seconds by monitoring the driver's eyes.

There have been several major components included as in seatbelt control system under this project: drowsiness detector. The system has been designed on a Raspberry Pi 3 model B with a 5MP night vision Pi camera and audio speaker, which makes the device both accessible and transportable. The system detects weariness well, also delivers a notification to regulate the condition and prompt the driver to rest up.

2.2 Related Work

In the precursor techniques, the monitoring system of drowsiness can only be performed in a variety of ways, such as remotely monitoring pulse rate or face expression. [4], the conventional AdaBoost method is combined with a Violas cascade algorithm utilizing Haar-Like characteristics to locate eyes. that is a real-time process [5]. Since using the Haar Cascade algorithm, the Haar-like bounding rectangular frames were stabilized over through the targeted eye region, which will be used in the drowsiness classification module. Furthermore, Haar features are images elements which are already commonly used in object tracking implementations. When a closed eye is sensed, the data will only be delivered. Next, the real-time process of [6] a rectangular principal feature called Haar-Like is utilized as being an internal node for a cascaded classifier. This feature will be applying to filter into one specific of image.

Besides that, [6] Viola cascaded algorithm uses the AdaBoost in the way that mixes a series of AdaBoost classifiers as a filter chain. If each of these filters in the acceptance region of the image shows the eyes close fails, this area is immediately classified as eyes open. When a filter accepts an area of image the eyes close, then the area enters the next filter in the chain. When the area of the image passes, so all the chain filters successfully, it is classified the eyes is closed. As a result, OpenCV is being used as an approach for detecting Haar features' eyes, an integral image for efficient image processing, feature extraction using AdaBoost, and a cascaded classifier for expeditious sensing.

Furthermore, [4] image Binarization is the initial stage is locating the eyes in a photograph, as it converts the scale image into a binary. Image segmentation, for example, converts a grayscale image to a binary image. As a result, parameters are determined depending on the lighting circumstances and the driver's facial structure. Binarization quite efficient and being use in a variety of document image Binarization processes. For image processing, the previous system mostly used a serial approach. As a result, the picture processing time is long, which means it takes longer to generate the [7].

Besides that, the Hough Transform algorithm is a standard image analysis technique to detecting features of a certain shape in digitalized images, such as lines or circles. The radius and canter of the pupil and iris are determined using the circular Hough transform. Thus, using the circle Hough Transform in OpenCV, the driver's current eye state may be determined. Furthermore, OpenCV is an accessible resource computer vision library that is being used to develop real-time techniques and algorithms. It primarily focuses on image processing, video capture, and analyzation, which includes features like face and object tracking.

Then, [4] the eyes detector method involves real-time processing of the input image stream to determine the driver's level of drowsiness and the analysis based on calculating the exact value of eyes aspect ratio (EAR) during eyes is closed and open. Therefore, when driver feels drowsy then EAR value drops to approximately zero. To attain the best exactness and predictability in identifying all types of human expressions using varied devices EAR (eye aspect ratio) and forms, this method requires a lot of time to train the facial dataset. [8] The eyes detection model was built with numerous facial features, and the rate of real application was near to 85%, except the driver's eyes were 50% closed, which could mislead the program. As a result, the system includes a data collection of 100 percent closed eyes shots to fully visualize the image of a fatigued driver who has lost attentiveness caused to drowsiness.

Lastly, the Raspberry Pi program is develop using python language which is the programming algorithm for drowsiness detection [4][5] and OpenCV Linux version is installed to the Raspberry-pi. The Raspberry-pi specifications used for this article were ARM11 controller orientated compact capacity open-source CPU with 512 MB RAM and 700 MHz speed of processing as it facilitates linking of varied low and high level devices such as digital camera and GPIO [4]. In [5] Raspberry Pi is 40 times faster than Arduino, it was chosen above Arduino and Galileo. It can multitask like a computer and is the ideal option if a project requires us to work together. The Raspberry Pi can indeed, be configured it into data server with SSH (Secure Shell) accessibility and files shared through an FTP website.

2.3 Drowsiness Detector System Component

2.3.1 Hardware Apparatus

2.3.1.1 Raspberry Pi 3 model B

Figure 2.1 and Figure 2.2 shown the Raspberry Pi 3 Model B and a 5MP night vision camera with accessibility to two infrared LED module, respectively in to developing the system of image processing. In addition, the processor frequency has been increased from 900MHz to 1.2GHz. In supply power, Raspberry Pi power input has also been converted to 5.1Vdc/2.5A. Therefore, the Pi's power supply has been improved from 2A to 2.5A, enabling that to run much more powerful hardware through USB ports. The Pi 3 includes four USB ports packed in for connecting to a mouse, keyboard, or something else featuring a USB connector. A USB hub are being installed to add extra connections, and a powered hub is advised to avoid overtaxing in on voltage regulator. Aside from that, the SD card is used for data storage and booting; no hard disc or SSD is present. The Raspberry Pi 3 Model B includes ARMv8 with 1GB RAM provides double the ram and a much faster processor that is 10 times better. It also provides wireless LAN and Bluetooth connection, making it a great option for strong networked designs.

Figure 2.1 Raspberry Pi Model B

2.3.1.2 Pi Camera Night Vision

Furthermore, 5MP Night Vision Camera is included the flex cable and the camera focus ring can be adjusted to desired of the position of objects. It is using external 1W high power at 850nano meter infrared photography lighting with 3.3V external power supplies. The camera has a maximum range of vision of 80 degrees. A night vision Pi camera been selected since it allows for picture capturing in both bright daytime and dimly illuminated environments. A modest Infrared light could be used to lighten the surroundings without disturbing the driver to improve video capturing in full fading light, as would be necessary at night and the sensor best PIX is 1080p.

Figure 2.2 5MP Night Vision Camera

2.3.1.3 Stereo speaker

Moreover, a speaker Figure 2.3 comes with a USB A type jack for power, which can be connected to any USB port on the Raspberry Pi board and provides 5V dc and a maximum current of 500mA. The Raspberry Pi Board's Audio barrel can also connect a standard 3.5mm audio jack.

Figure 2.3 Stereo speaker

2.3.2 Software Apparatus

2.3.2.1 Python

Raspbian is a Linux distribution based on Debian that has been customized for the Raspberry Pi. It includes a package of fundamental apps and tools for running the Raspberry Pi. Python is a programming language that may be used to construct computer programmers on the Raspberry Pi, while Thorny and PyCharm is an application that generates and runs Python scripts. Python was chosen as the primary language by the Raspberry Pi Foundation because of its strength, adaptability, and flexibility of use. The cycle of editing, testing, and debugging is quick. Moreover, a defect or incorrect input would never trigger a segmentation fault in a Python application, therefore debugging them is simple. When the interpreter detects an error, it throws an exception. The interpreter produces a summary table if the application fails to catch the error. Analysis of locally and globally variables, execution of random phrases, provided a chance, stepping through the code one line at a time, and so on are all possible with a source level debugging.

2.3.2.2 PyCharm

An Integrated Development Environment (IDE) that is assigned to computer programming, especially the Python programming language, which gives a comprehensive set of tools for Python programmers. It's closely linked to produce an efficient Python, website, and data science development environment. Furthermore, PyCharm includes smart script execution, coding inspections, on-the-fly error flagging and rapid solutions, as well as automatic code coding and sophisticated browsing features. PyCharm Python is utilized upon this laptop desktop as well before transferring the coding to hardware and executing on thorny Python.

2.3.2.3 Fritzing

A fully accessible project to develop CAD software for electronic component design, with the objective of assisting designers and producers which are ready to proceed well beyond testing stage and develop a more solid circuit. The circuit schematic for the entire system was developed using the Fritzing platform.

2.3.2.4 Xming and Putty

PuTTY is Project Xming's primary and integrating X terminal browser for Microsoft Windows, to access Raspberry Pi window.

2.4 Concept of system

The Image Real time and Drowsiness Detection approach involves recording the driver's face input images and then recognising facial landmarks, such as the eye and mouth aspects. The Pi camera and Raspberry Pi module are used to capture and process images. This is performed by the detecting of yawns and eye movement. The technology detects whether the driver is sleepy relying on the above that has mentioned the criteria.

The Haar Cascade, facial landmark shape predictor and eyes-mouth aspect ratio algorithm [Appendix A] are involved and it programmed in Python using OpenCV computer vision modules. The following libraries, such as NumPy, Dlib, OpenCV, and Pi camera, must be installed before the application may begin running. It's built for speed and efficiency, with a heavy emphasis on real-time use. It is an accessible video processing project planning phase of providing a software platform for image processing algorithms as well as a collection of libraries and applications. It includes I/O libraries for easily acquiring and manipulating video data from several camera inputs.

2.4.1 Haar Cascade Classifier

Haar Cascading is a computer vision technique that involves developing a classification from a huge number of positive and negative images. Paul Viola and Michael Jones are the developers of the program. The classifications developed for object detection are Haar feature-based cascade classifiers. The face recognition applies to the concept of detecting person's facial features in real-time video. The faces were interpreted using the Haar cascade classification. The Haar Cascade classifier consists of four main phases, which are Cascading classifiers, Haar features selections, Integral image construction and Adaboost training [9].

This classification uses a computer development approach to establish a cascade operation from the images to determine objects differently in various photos. The identification of faces and facial movements in images is also effective. The task is completed by presenting the classification with positive and negative images. The properties are then extracted from the image. Every feature is a distinct quantity obtained by reducing the number of pixels in the white rectangle from the total of pixels in the black rectangle. That it recognises the faces of various people in various settings. Due to integral pictures, the Haar-like feature of any scale may be determined in fixed time.

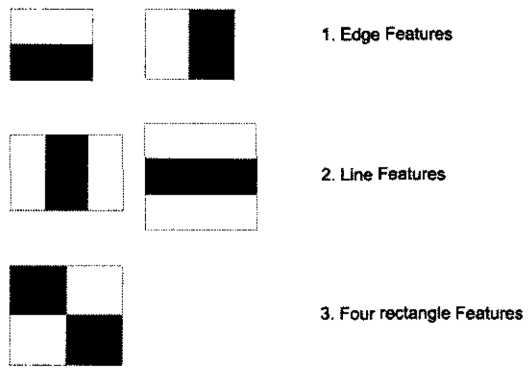


Figure 2.4 Haar rectangular features

As seen below, the Haar cascade is a classifier by applying embedded method cv2.imread(img path) to extract the input images, with the image directory as an input argument. Then, changing image to grayscale and then showed. Lastly, the Haar cascade classification loaded.

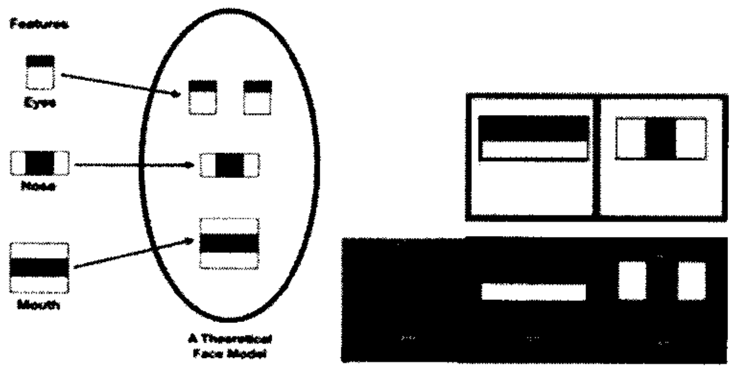


Figure 2.5 Haar like feature

Figure 2.5 is indicating the Haar-like characteristic. It comprises of two features: edge and line. The white bar in the gray-scale picture depicts the pixels closest towards the background light.

The Haar Classifier is a method for detecting objects. The characteristics is being retrieved from the images in applied to measure the objects and determine what those are. Equation (1) may be used to compute the Haar pixel intensity [10].

$$\text{Pixel value} = \left(\frac{\text{Sum of the Dark pixels}}{\text{Number of Dark pixel}}\right) - \left(\frac{\text{Sum of the Light pixels}}{\text{Number of Light pixels}}\right) \qquad (1)$$

The computation of such Haar characteristics is sped considerably by using integral images. Rather than determining at each pixel, it divides the screen into sub-rectangles and produces array sources for each of them. The Haar features are then determined using them.

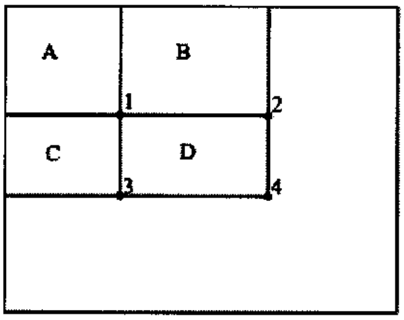


Figure 2.6 Illustration for how an integral image works, a rectangle that has been divided into many parts.

Figure 2.6 illustrates that to determine the sum of pixels inside this rectangle using four array values. The total of an pixels in rectangle A is the amount of the integral image at point 1. A+B is the value at spot 2, A+C is the amount at point 3, and A+B+C+D is the amount at point 4. 4+1-(2+3) is the total of the statistics stated [11].

The summation of the pixels above and to the left of *(x,y)* including is contained in the integral image at point *(x,y)*. The integral image is *ii (x,y)*, whereas the original image is *i (x,y)*. Utilizing reoccurrence that shown below.

$$ii(x,y) = \sum_{x' \le x, y' \le y} i(x', y'), \tag{2}$$

$$s(x,y) = s(x,y-1) + i(x,y) \tag{3}$$

$$ii(x,y) = ii(x-1,y) + s(x,y) \tag{4}$$

Figure 2.7 below indicate the Haar cascade classifier's flowchart. When the camera captures a picture, it transforms it to grayscale. The cascade classifier identifies the face; if the face is identified, the classifier analyses including each eye inside this human face; when both eyes are presented, the classifier validates the pixel size of the facial images. The picture will be extracted for facial images, along the image that are getting compared to a database of face samples [12][13]. The advantages of the Haar Cascade classifier it has a higher sensory precise and a lower positive ratio.

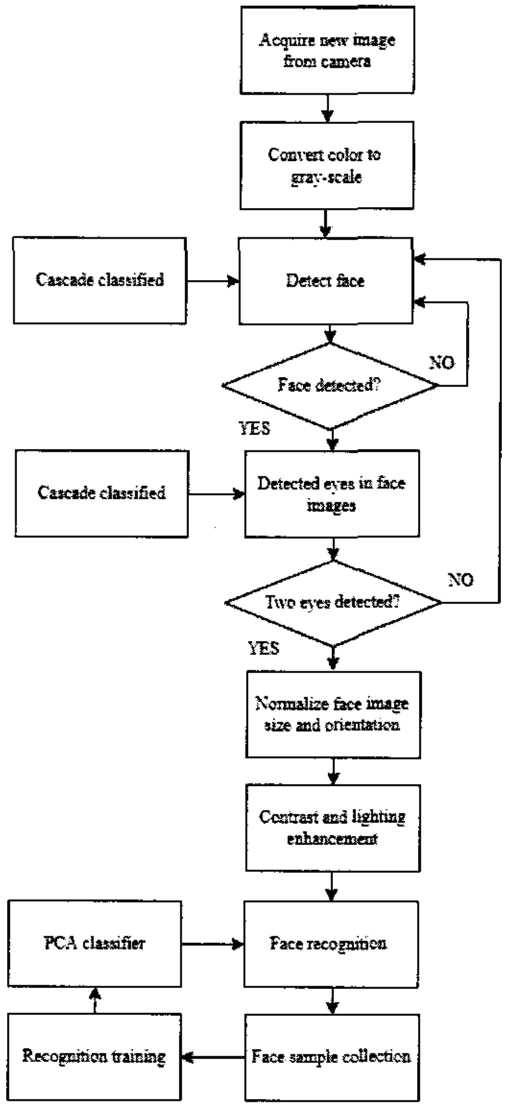


Figure 2.7 Haar Cascade flowchart

Moreover, viola and Jones implement the AdaBoost method since that improves the classifier outcomes of a basic learning, which means it is straightforward to compute. The AdaBoost training process generates a large amount of data, which can then be aggregated into a classifier. Gradient detection is often performed using a classifier built of minor characteristics on the facial [14].

Last but least, cascade Classifier as sequence of AdaBoost classifiers produce the filtering cascade depicted in Figure 2.8 The weight of AdaBoost result obtained determines the order of filtration in the cascade. Whenever an image fails to identify the first stage, the cascade removes them. The bigger the filters weights during first chain, the faster will it be to remove the non-face images region. As a result, a Cascade Classifier that effectively integrates multiple data [14].

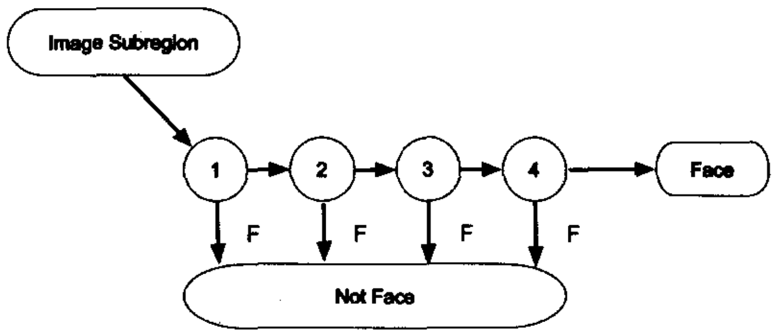


Figure 2.8 Cascade classifier illustration

2.4.2 Facial Landmark Predictor

Facial landmarks are often applied to identify and portray important facial features such as the eyes, brows, nose, mouth, and jawline. Face landmark detection is a subsection of the shape prediction aspect. A shape predictor aims to pinpoint significant focuses upon a shape based on a given image that identifies the item of significance. The objective of applying shape prediction algorithms to detect significant facial structures on the face in the framework of facial landmarks is to recognize specific facial structures on the face. As a result, detecting face landmarks is a two-step procedure:

Step 1: Identify the face in that picture.

Step 2: Analyse the face for significant facial structures.

The built-in Haar cascades in OpenCV may be used to detect faces. Then, identifying significant facial features in the facial landmark. Face landmark detectors come in a range of shapes and sizes, but the aim is to locate and categorise the following facial landmarks: mouth, right and left eyebrows, right and left eyes, nose, and jaw.

This approach begins with a database of annotated facial landmarks on an image as a training dataset. These photos have been carefully annotated, with the (x, y) coordinates of areas around each face structure specified. This sequence of linear prediction is constructed using this training data to predict the facial landmark locations precisely from the image intensity.

The 68 (x, y) coordinates which correspond to facial structures on the face are estimated using the pre-trained facial landmark detection within its dlib package. On the figure below shown the indicators of the 68 coordinates.

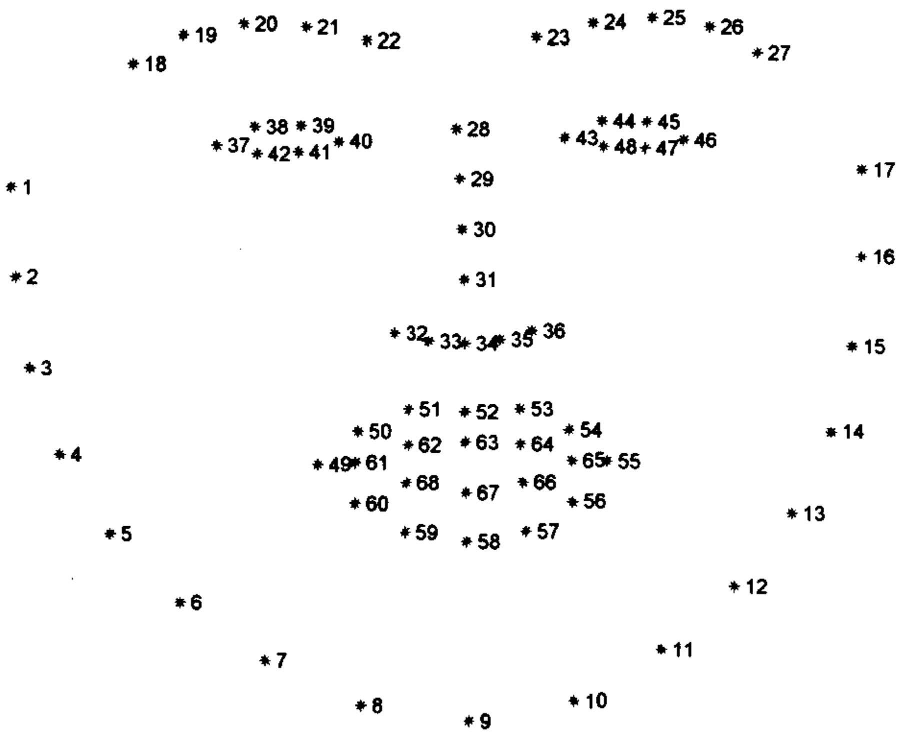


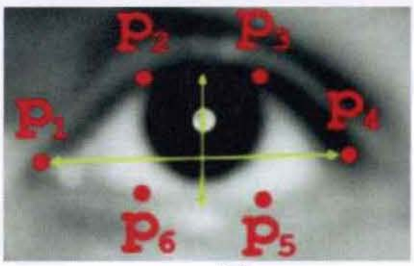
Figure 2.9 Shape predictor 68 faical landmark

Therefore, the high percentage of present activities simply include mappings for a limited portion of the total number of images. In certain circumstances, the precision of offered remarks is inadequate, due to human fatigue. Then, every resource has a varied number of landmarks in its prediction model.

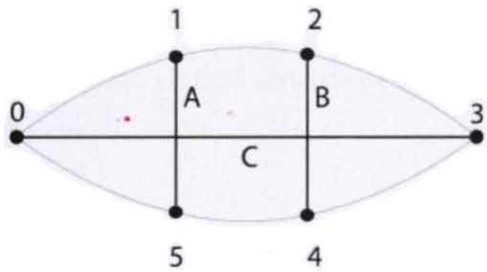
2.4.3 Eyes Aspect Ratio (EAR) and Mouth Aspect Ratio (MAR)

The areas of the eyes are indicated on the eye mapping. After that, we could use sufficient thresholding to restructure the eye mapping images to black and white. The eyes are intended to be white in this current photo, whereas the remaining is black. However, numerous reprocessing stages are necessary, involving degradation, dilatation, and the identification of the largest linked elements, such as eyes. Furthermore, in the last stage, specific parametric properties of the eyes are used to eliminate incorrect detections [15].

The ocular landmarks in the face will be determined to acquire the EAR, as shown in Figure 2.10. Dlib's 68 facial landmark model would be used to find these landmarks, which is a pre-processing method which can be applied internally with Python.



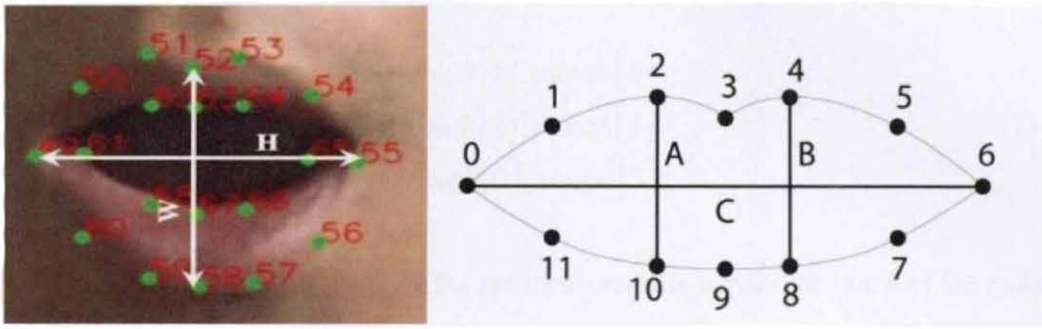
(a)



(b)

Figure 2.10 (a) Eye landmark (b) Eye region of reference coordinate

Identifying the point of the mouth and lips generally one upcoming phase in yawning detection. In order to provide it, the mouth portion of the face would be segmented. This approach takes use of the lips landmark, similarly with eyes detect [16].



(a) (b)

Figure 2.11 (a) Lip landmark (b) Mouth region of reference coordinate

Moreover, once the points have been obtained, the EAR can be calculated using the Equation (5) below. The p2, p3, p5 and p6 are used to measure the height whereas p1 and p4 are used to measure width of the eyes. Based on the Figure 2.10, the equation based on eye region with reference coordinate points.

$$\begin{aligned} Ae &= dis(\ eye[1], eye[5]) \\ Be &= dis(\ eye[2], eye[4]) \\ Ce &= dis(\ eye[0], eye[3]) \end{aligned} \tag{5}$$

At the Equation (5), A_e and B_e are measured the vertical distance of the eye, while C_e calculates the horizontal dimensions of the eye.

$$EAR = \frac{(\ Ae + Be)}{(\ 2 \times Ce)} \tag{6}$$

Based on the Figure 2.11, the equation based on mouth region with reference coordinate points.

$$\begin{aligned} Am &= dis(\,mouth[1], mouth[5]) \\ Bm &= dis(\,mouth[2], mouth[4]) \\ Cm &= dis(\,mouth[0], mouth[3]) \end{aligned} \tag{7}$$

At the Equation (7), A_m and B_m are measured the vertical distance of the mouth, while C_m calculates the horizontal dimensions of the mouth.

$$MAR = \frac{(Am + Bm)}{(2 \times Cm)} \tag{8}$$

CHAPTER 3

METHODOLOGY

3.1 Introduction

The development of the drowsiness detection system is discussed in this chapter, together with a simulation of how the system's procedure will work. The discussion is containing the performance evaluation for the creation of a monitoring of seat belt control system and drowsiness detection system, which can detect the fatigue behaviour of the driver and the process of system in real-time video and the system is trigger.

3.2 Process of seat belt control system

Figure 3.1 below shown the process of work for the whole seat belt control system. This technology is intended to provide a driving safety system that, which it is includes the interlink between sub-system at every system.

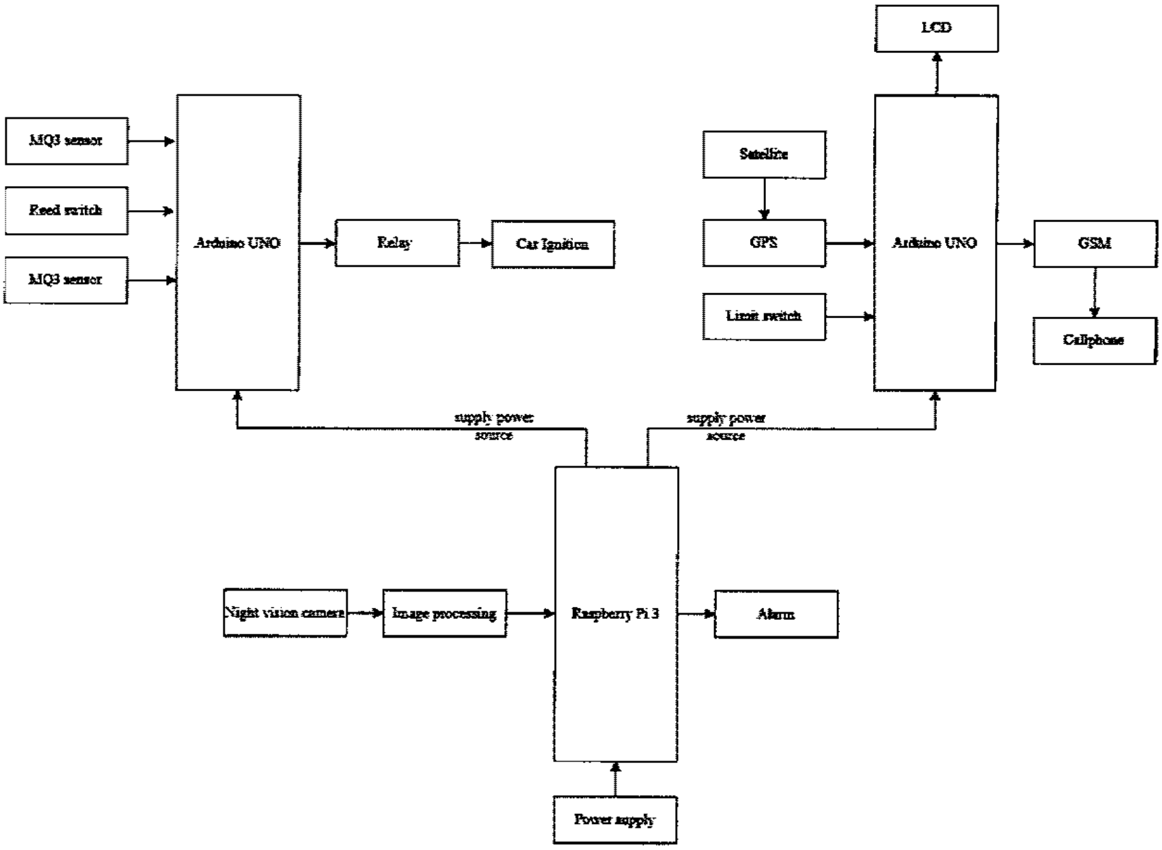


Figure 3.1 Block diagram of seat belt control system

There are three sub-system in this block diagram:

a. Drowsiness detection system

A main system, which is consume about 260mA of current at 5V from power supply. Then, Raspberry Pi microcontroller, will provide power source through other system which is seat belt and alcohol detection system as well vehicle tracking system of Arduino UNO microcontroller at 5V. Next, night vision camera with 3.3V external power activate indicate to utilize the image capture of drowsiness of the driver, which controlled by Raspberry pi itself. Afterward, the microcontroller activates the speaker, once it is trigger the fatigue behaviour of the driver.

b. Seat belt and alcohol detection system

The alcohol sensor uses a MQ3 sensor, which are using Metal Oxide Semiconductor (MOS) categories of sensor and senses alcohol by detecting the changes in resistance of the sensing material whenever presented to alcohol. Alcohol quantities can be measured by using it in a conventional voltage divider network. The seat belt detector is then activated by magnetic reed switches mounted inside the seat belt buckle. Plus, there are two common types: normally open and normally closed. A 5V source was required to generate the magnetic reed sensor. The operation is carried out by the seat belt is buckled and unbuckled, it will receive a high and low output. If the seat belt is unbuckled, an alert will sound for 10 seconds, and if the seatbelt is unbuckled, the motor will begin to decelerate. For the MQ3 alcohol sensor, a 5V power source was connected from the Raspberry Pi to the Arduino UNO. The MQ3 Sensor was then linked to an oled screen display through an Arduino UNO as the system's base connection. The display will indicate the alcohol concentration, and a condition have been put when the alcohol The display will indicate the alcohol concentration, and a condition will be set if the alcohol concentration exceeds the driver's level of sobriety. Afterward, the vehicle's speed would gradually decrease.

c. Vehicle tracking system

This system is design includes the Arduino UNO board microcontroller, GPS and GSM module, and LCD. The programming specification for the Arduino IDE software includes a Google map. A 5V DC from the battery voltage is required to power the GSM and GPS shielding. The operational voltage of the GPS receiver is 2.7 to 3.6 volts, while the SIM900A GSM module is 3.4 to 4.5 volts. To control the operation, an Arduino is used.

Data from satellites is received using a 9600 baud GPS receiver. The data is transferred to the Arduino, which extracts GPS coordinates and sends them to GSM as latitude and longitude coordinates. The AT command is used by the GSM module to send the coordinates to the user or owner. A serial port is used by the AT Command to communicate. This enables users or owner to keep track of the vehicle's whereabouts on Google Maps. To displaying coordinates, LCD is also implemented. The limit switch indicates that aid is needed in an emergency. When an accident occurs, the pressure of the hit activates the limit switch. The latitude and longitude data will be collected by the GPS and sent to Arduino. The Arduino relayed the data to GSM and sent a message to the user's latitude and longitude, indicating that an accident had happened.

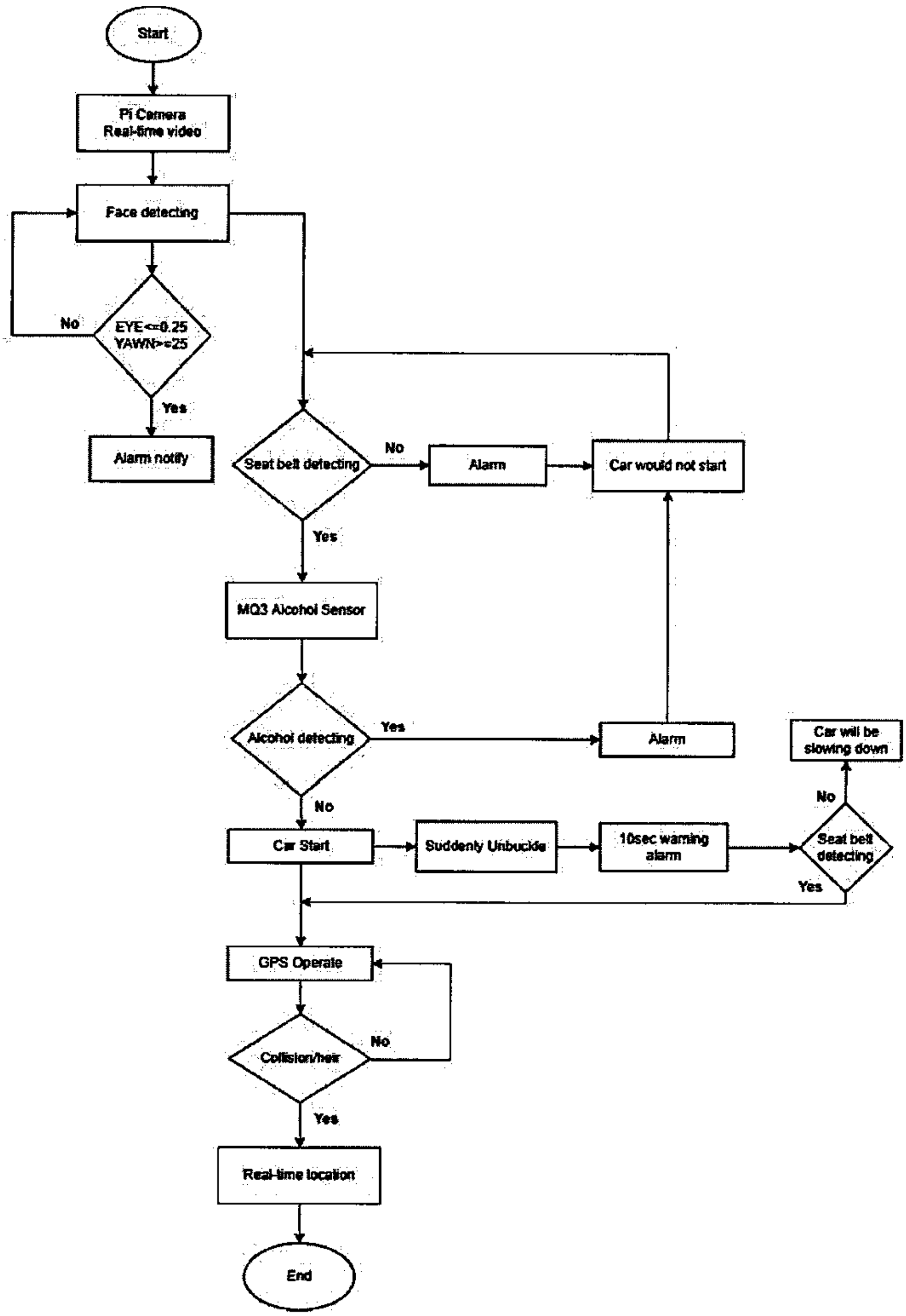


Figure 3.2 Flowchart seat belt control system

3.3 Process of drowsiness detection system

The Figure 3.3 below shown the process of work for the drowsiness detection system. Hence it included the interlink between component related in the drowsiness detection system are shown. This approach begins by facial feature extraction of a face images, which eliminates the image of the object from the background picture and allows for yet more precise detection of the placement of the eye and mouth. By utilizing the Pi camera's night vision, where it extracts grayscale frames, and a facial recognition library. As a result, it is suitable for real-time analysing the pattern of the eyes and lips. We can set a threshold for detecting eyes and lips that can range from 25 to 30 in MAR as well as 0.25 to 0.15 for EAR, allowing us to interpret and to get positive accuracy.

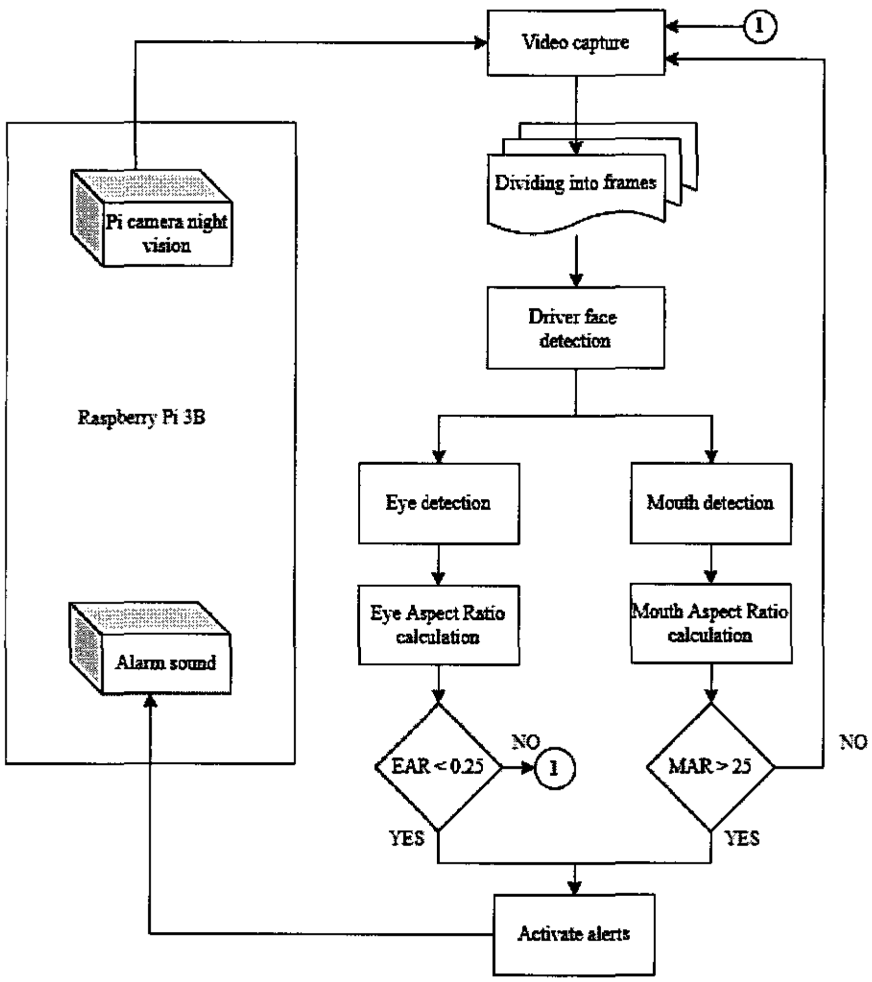


Figure 3.3 Process for drowsiness detection flowchart

3.4 System Architecture

This technology is intended to provide a sleepiness detection system that can detect driving weariness directly in the driver's face.

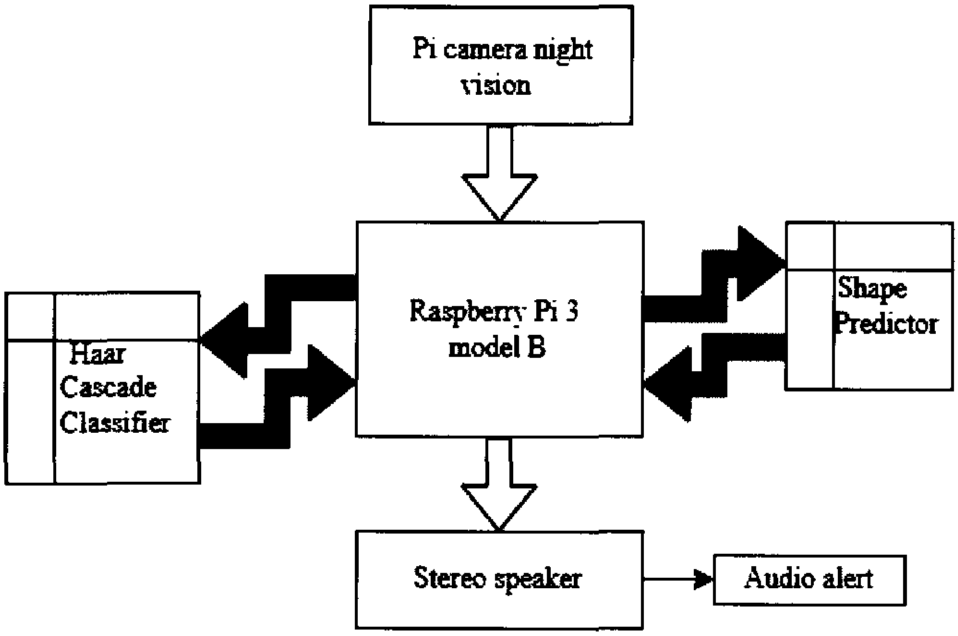


Figure 3.4 System Architecture of drowsiness detection system

There are three parts to this system architecture:

a. Raspberry Pi 3B

Image processor on the Raspberry Pi is assessed using a face landmark technique which is built on eye and lip detection algorithms. In a real-time system, the Microcontroller has a very high processor.

b. 5M Pi camera night vision

It was developed to acquire images in real time. It captures a real-time image on a regular to identify the driver drowsiness. The Raspberry Pi is then utilized to analyze the images capture for eye and lip detection.

c. Stereo speaker

It functioned as a parameter. When the system triggered the detection of drowsiness, an audio alert is generated.

3.5 Design Development

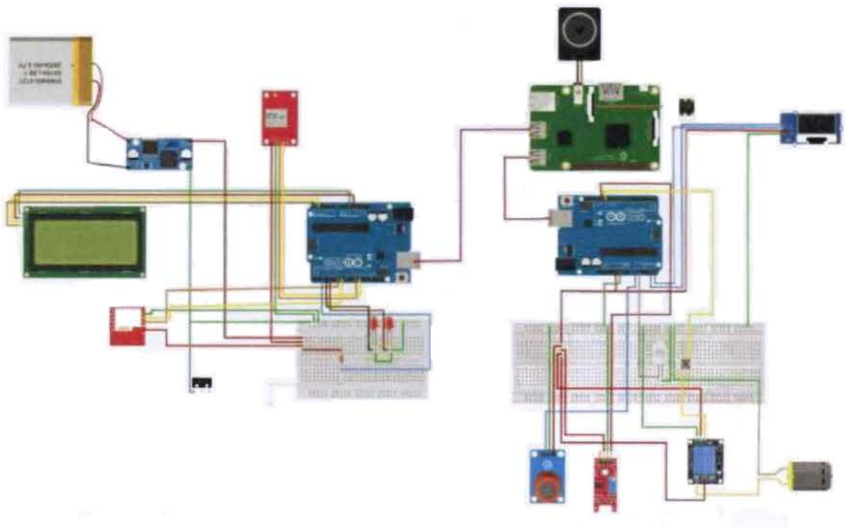


Figure 3.5 Seat belt control system circuit diagram

Figure 3.6 Prototype seat belt control system

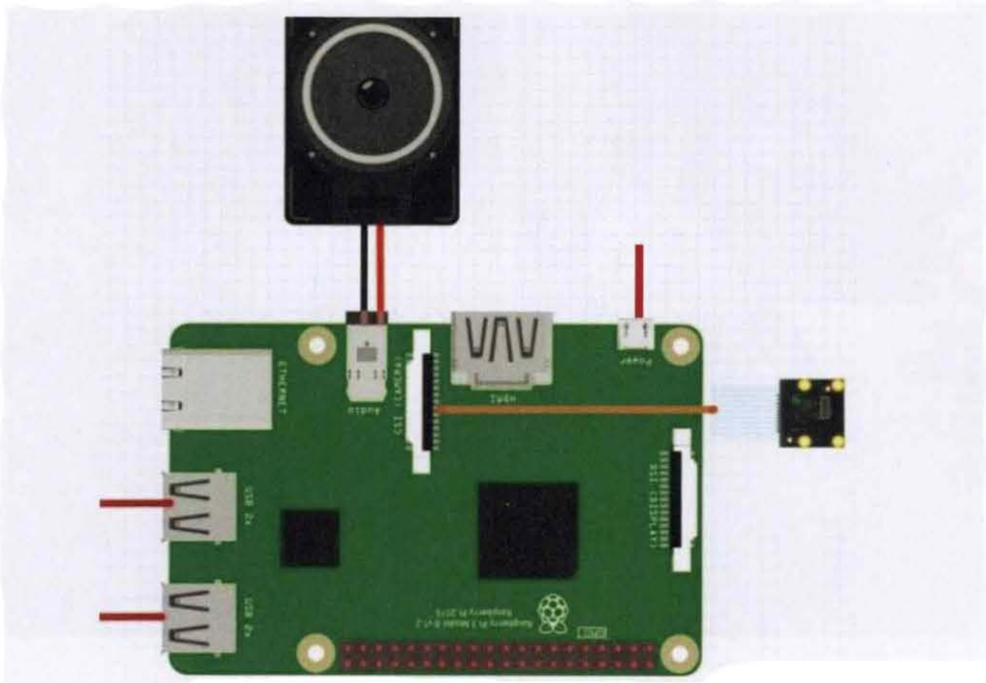


Figure 3.7 Drowsiness system circuit diagram

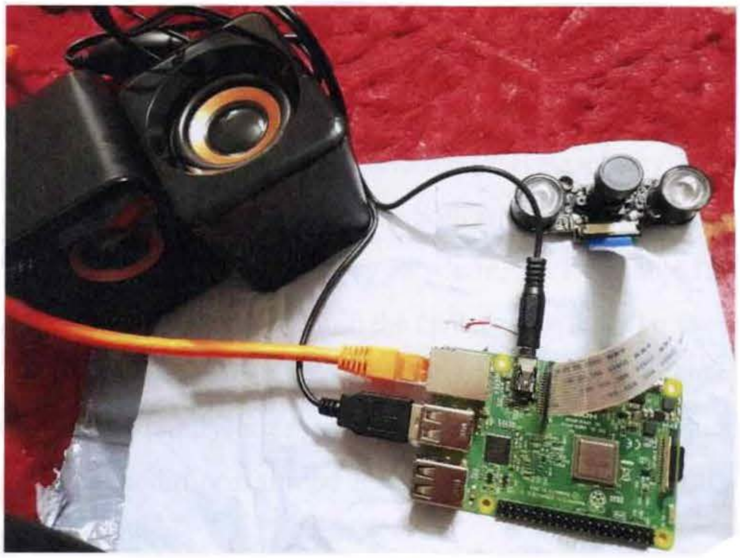
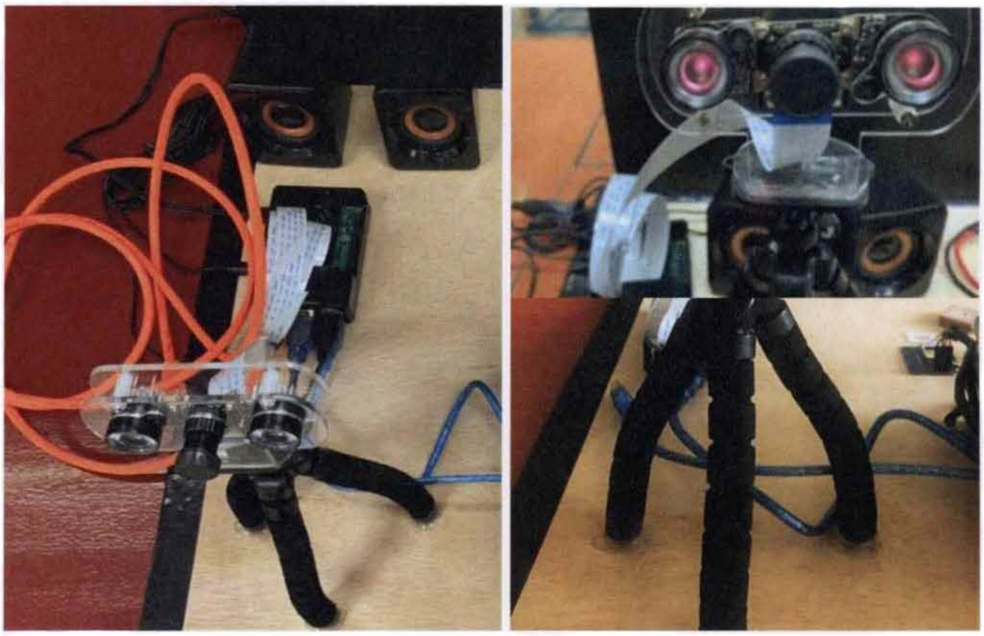


Figure 3.8 Drowsiness system before fabricating



(a) (b)

Figure 3.9 Prototype drowsiness system (a). Upper view (b). Front view

The PVC box is used to ensure the prototype locations in order. The wiring is connected according to the Fritzing schematic. At the bottom of the PVC box, a hole is drilled for the Arduino UNO cable to interface to the USB Raspberry Pi port, to deliver the power source for the purpose of powering up both Arduino UNO microcontroller. In addition, to hook up a Pi camera in Raspberry Pi where, the flex cable is included with the Pi camera and therefore should be put into the camera interface. To use it, simply pull up on the tabs. Then, slide the flex cable into the port until it is completely inserted. After that, press the tabs down slightly to secure the cable. Moreover, the stereo speaker is simply installed into the audio port on the Raspberry Pi. At last, the Raspberry Pi is powered by a 5V power source.

3.6 System Operation

3.6.1 Raspberry Pi configuration operation

Here is a method for configuring a Raspberry Pi so that it may connect toward a shell or a desktop using VNC via Wireless connections or Ethernet but without a monitor, keyboard, or mouse. Linking to the Raspberry Pi in this method is considered Headless method.

(a) (b)

Figure 3.10 (a) SD card plug into Raspberry pi
(b) Ethernet cable connected to LAN port Raspberry Pi and Laptop

At the first step is installing Raspbian on the SD card including at least 8 GB of available space using laptop. Then, without any extension, establish an unfilled file named SSH. After that, remove the SD card and insert these into the Raspberry Pi, then attach a 5V power source to the Microcontroller board. Next, plug ethernet cable connected to LAN port Raspberry Pi and laptop

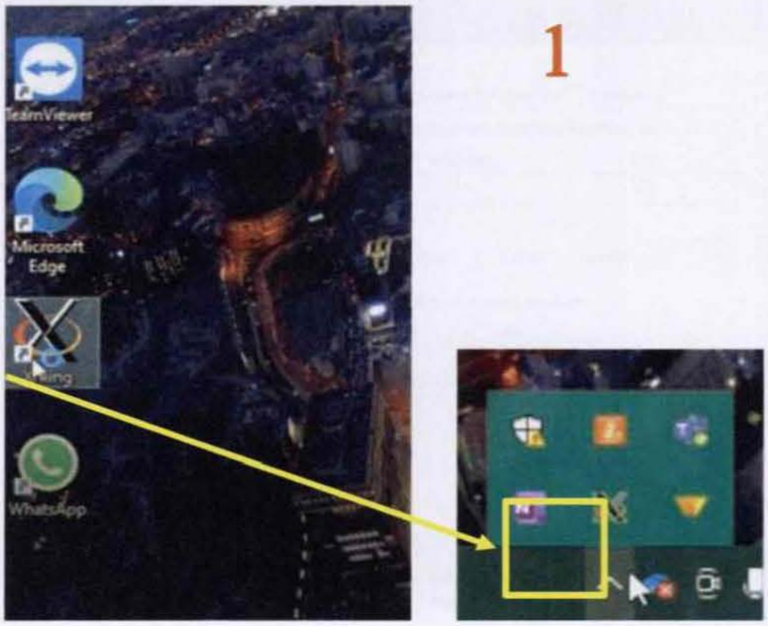


Figure 3.11 Enable Xming server software

To run Xming, select XLaunch from the desktop. Next, checking that Xming is executing; it will appear as a new running activity in the task bar. Which is "Xming Server:0.0" would appear when drag the cursor over the X icon on taskbar.

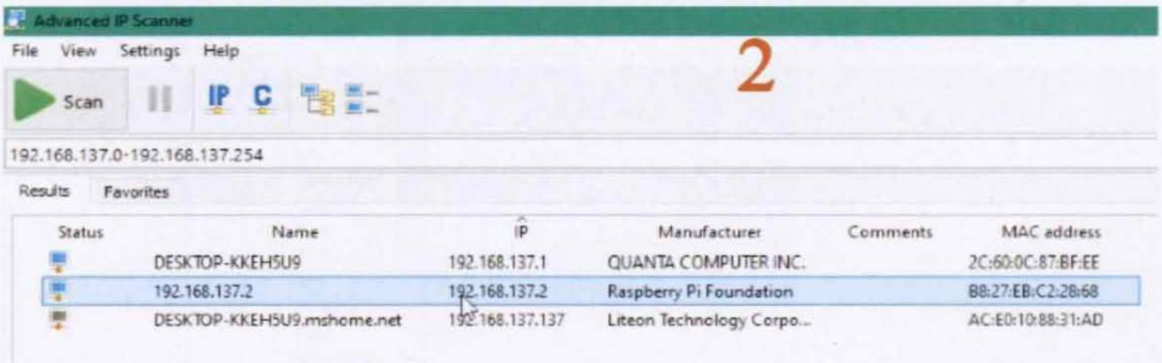
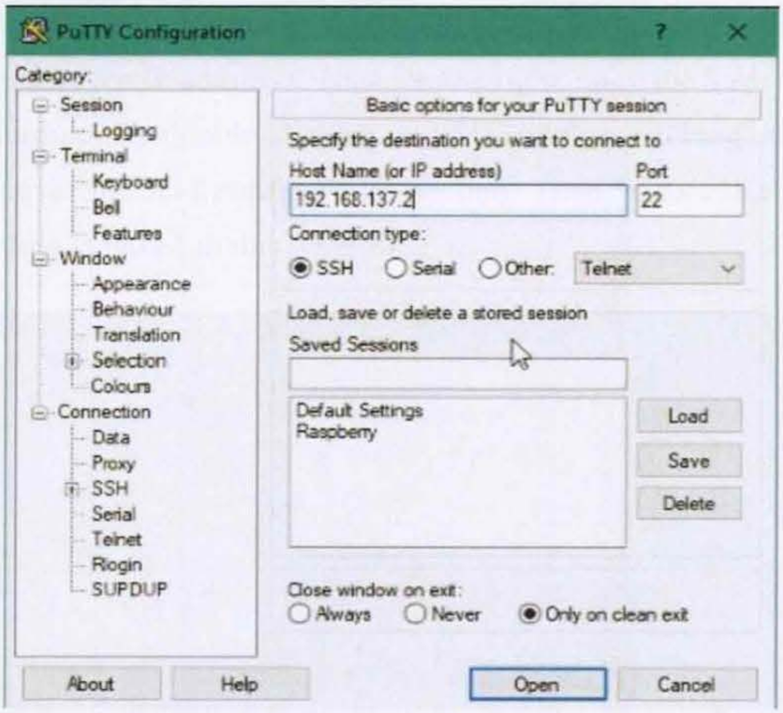
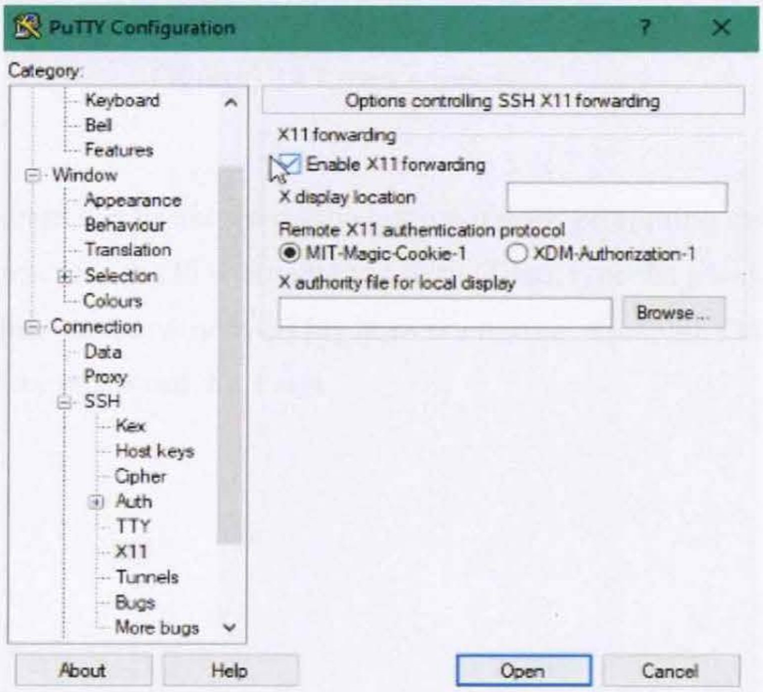


Figure 3.12 Scanning IP address on Advanced IP scanner software

On the top-left, click Scan. It displays the PC's and Pi's IP addresses. Noted down or coping by left click of the Pi's IP address. 3



(a)



(b)

Figure 3.13 (a)(b) Putty.exe server software

Open putty.exe. and insert or paste the IP address which has previously coping in the box under Host name (or IP address). Then, on the right, click the Save button. Plus, to set up X11 Forwarding. By double-clicking on SSH and then clicking on X11 on the left side, it can discover the X11 configuration section. Then, ensure the box labelled enable X11 Forwarding is ticked in this window.

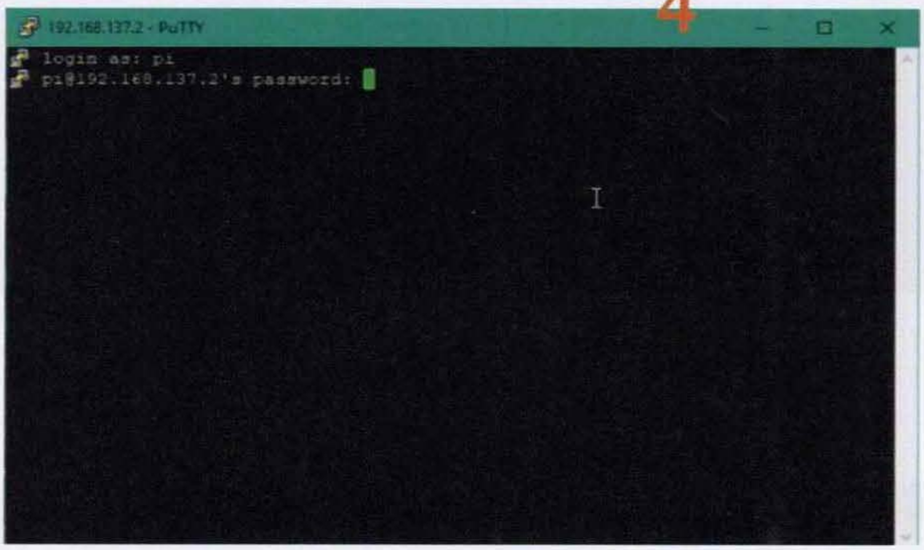


Figure 3.14 Login window

The linux prompt can be shown on the login window, prompting the user to provide their username. Use the Pi username to log in. Then, type the password in which there are no lines on the window. This is a very normal situation. Once completed inputting the password, hit Enter.

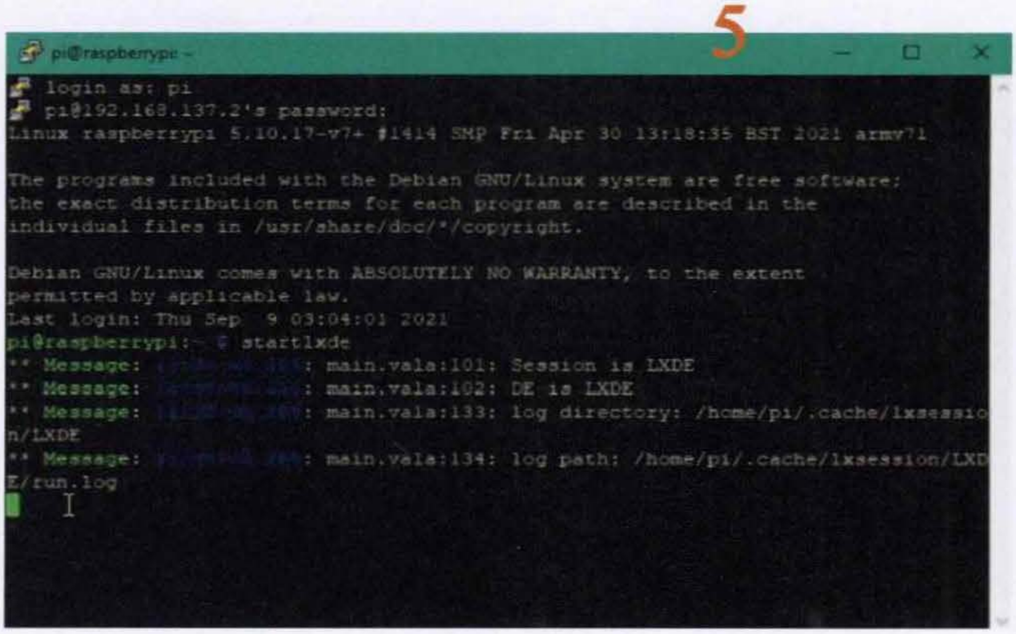


Figure 3.15 startlxde - the terminal session of LXDE

Figure 3.16 pcmanfm terminal session of LXDE - Raspberry Pi window

Figure 3.15 and 3.16 show the mentioned of startlxde. It is indicated of LXDE's terminal session which is started with the command startlxde. The Openbox, lxpanel, pcmanfm, dbus-launch, and lxsession are loaded by default.

3.6.2 Drowsiness system operation

As referred on the Figure 3.3 has shown the drowsiness system operation. There are step of drowsiness system as follow:

i. Step 1: video real-time capture

ii. Step 2: Facial landmark detecting

iii. Step 3: Eye and mouth detecting

iv. Step 4: Drowsiness detection

The Pi camera Module is a 5-megapixel camcorder with night vision that captures an ongoing video stream. Face detection is performed once the video recorded stream is transitioned it into a series of frames. Face landmark algorithm, which can identify facial structure, is used to analyse the facial recognition. Actually, detecting method works with OpenCv, a Python-based library package. OpenCv is a computer vision method which are used for real-time image interpretation. When facial characteristics are effectively recognised, the following step, such as detecting drowsiness drivers' eyes and lips, is achievable using facial landmark predictor algorithms. As a result, it would transform the framed format to grey scale, with six parameters tracing the identified eye and lip regions.

In addition, EAR and MAR have to be determined, which quantify the distance between vertical and horizontal eye and mouth landmark points using the Euclidean distance (ED) approach as shown in Equation (5). The eyelid and lip region distances are measured. Throughout synchronised eye blink and yawn, the EAR and MAR values of the left and right eyes, as well as the lips, are aggregated. Once both eyes are open and mouth is closed, the EAR and MAR predictor variables keep sustained, however numbers alter at randomized during eye blink and yawn. Furthermore, the audio speaker is enabled on, so the Raspberry Pi3 is activated to activate a speaker once the number of frames exceeds 30. The speakers are more efficient at waking up a drowsy driver.

3.7 System Implementation

The system has implemented by prototyping. Therefore, to be remarkable, this relating to development is not intended to use for an actual driving situation. Therefore, the desktop is utilized to show the camera's real-time video as shown in Figure 3.9.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

This paragraph emphasises depth on the results, interpretation, and analyzation. The developed drowsiness detection system's final prototype is seen in the Figure 3.9. Numerous phases of implementation were carried out in accordance with the overall system's execution.

4.2 Result and Discussion

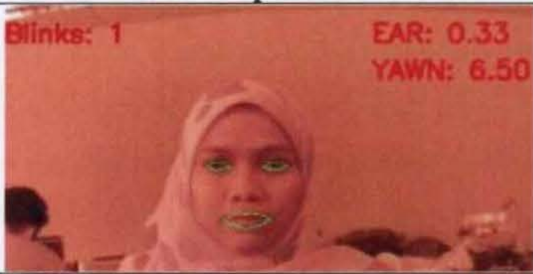
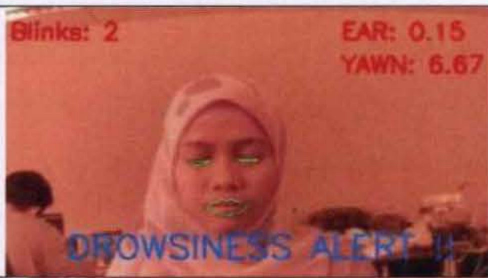
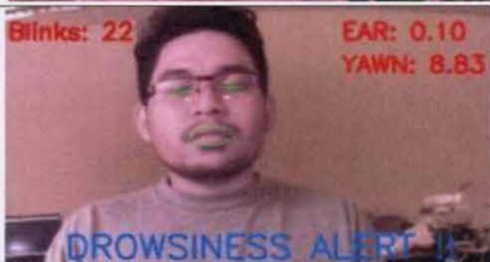
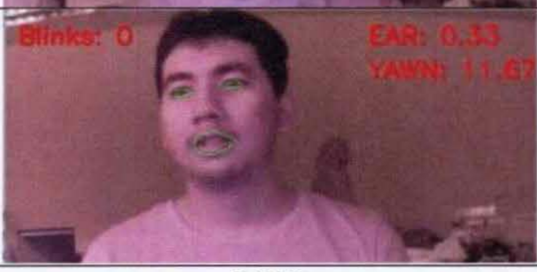
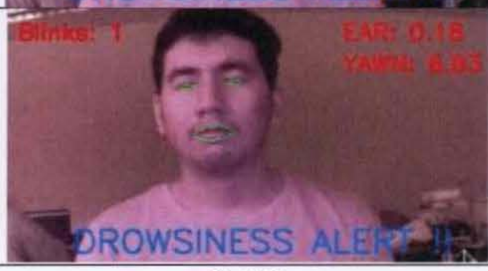
The intended outcomes of the Haar Cascade classifier approaches and the EAR and MAR equations are clarified based on numerous experiments conducted. The Drowsiness system setup and associated algorithm, as well as the image detection interfaces with Raspberry Pi. To be conclusive, this invention would not be utilised to be used in an actual driving circumstance. As a necessary consequence, the desktop is used to present real-time video from the camera. To engage in the tests, a total of 5 individuals were enrolled. All subjects were given a short explanation of the methods ahead towards the testing. They will seat 72 cm apart from the camera as shown on Figure 4.1. Tests have been used to verify the validity of the Haar Cascade classifier and the EAR, MAR techniques.

(a) (b)

Figure 4.1 (a) Position of the subject with camera (b) The distance between subject and camera.

The identification of eye and lip landmarks was used to verify the EAR equation and MAR. If the EAR number falls below a certain threshold, the driver might close their eyes, that depending to the algorithm. The MAR rate will rise if the driver opens their mouth. As a result, the EAR was calculated according to equation for each successive image frame, and an EAR with MAR threshold was formed inside the script. The parameter between vertical eye landmarks is computed by the numerator of this equation, whereas the distance between horizontal eye landmarks is computed by the denominator. The average of the EAR values for both eyes was shown on the window after the EAR measurements for both eyes were established. In addition, The EAR evaluation technique is equivalent to determining the MAR value. As a result, the window simultaneously displayed the average of the MAR value.

Table 4.1 The average value of EAR during eyes opened and closed

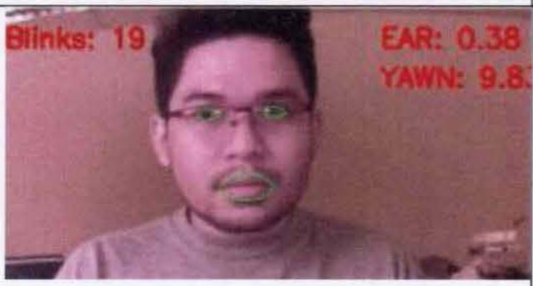
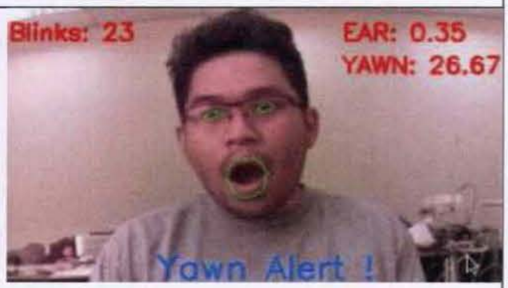
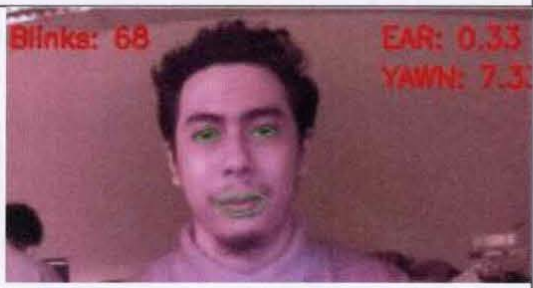
<table>
<tr><th>Subject</th><th>EAR during eyes open</th><th>EAR during eyes closed</th></tr>
<tr><td>1</td><td>
</td><td>
</td></tr>
<tr><td>2</td><td>
</td><td>
</td></tr>
<tr><td>3</td><td>
</td><td>
</td></tr>
<tr><td>4</td><td>
</td><td>
</td></tr>
<tr><td>5</td><td>
</td><td>
</td></tr>
<tr><td>Average EAR</td><td>0.352</td><td>0.158</td></tr>
</table>

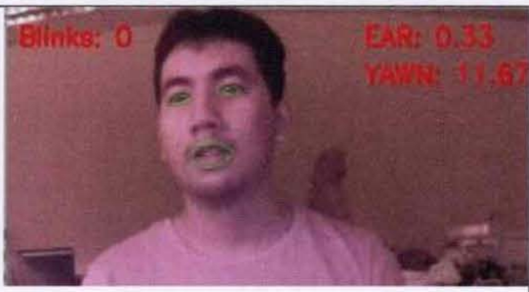
The estimated average value of EAR gained from the five participants is shown in Table 4.1. Once the eyes are closed and open, the average eye aspect ratio is 0.352 and 0.158, correspondingly. The configuration eye threshold has been assigned to 0.25 in the system, it is based on the optimal value of the beginning number for eyes closed, which has been evaluated for each subject. As a result, when the reading decreases below those thresholds, the output is triggered. These experimental results demonstrated that a zero mean EAR value could never be reached with the eyes closed. However, according on the EAR equation, it can be determined that when the EAR value drops, the driver is most likely in the condition of closing his eyes. As shown in Table 4.1, the decreased value of average EAR from 0.352 to 0.158 represents the condition of eyes closing. Plus, the detector was able to detect the eyes if wore glasses.

$$Eye\ Threshold\ (Te)\ ,\ Te\ =\ 0.158$$

$$Eye\ state = \begin{cases} Closed,\ EAR\ <\ Te \\ Opend,\ EAR\ \geq\ Te \end{cases} \tag{9}$$

Table 4.2 The average value of MAR during mouth opened and closed

Subject	MAR during not Yawn	MAR during Yawn
1		
2		
3		
4		

5		
Average Yawn	**8.232**	**27.5**

Furthermore, when the mouth is closed and opened sufficiently, the estimated average value of MAR is 8.23 and 27.5, respectively, as shown in Table 4.2. In the system, the configuration MAR threshold has been set to 2, it is based on the optimal value of the beginning number for mouth opened, which has been evaluated for each subject. As a consequence, the output is triggered when the readings surpass certain thresholds. Wherever the MAR value dramatically rises, it implies that the driver is most likely yawning, as seen in the data. Table 4.2 demonstrates that the average MAR value rises from 8.23 to 27.5, indicating yawning.

$$Yawn\ Threshold\ (\ Ty)\ ,\ Ty\ =\ 25$$

$$Mouth\ state = \begin{cases} Closed,\ MAR\ <\ Ty \\ Opend,\ MAR\ \geq\ Ty \end{cases} \tag{10}$$

CHAPTER 5

CONCLUSION AND RECOMMENDATION

5.1 Conclusion

In conclusion, the designed utilizing image processing that determine drivers' drowsiness and assigned on alerts as needed. Pixel per pixel, the footage of the driver is acquired and processed in real time. Applying Haar cascade classifiers as well as a facial landmark predictor, the system is enabled to generate the facial landmarks and from that, those requisite facial characteristics. The level of shift in form of certain face characteristics is used to determine drowsiness. Whenever the system detects drowsiness, it sends out locally alarms that awake the driver up, with the regularity changing depending on the amount of drowsiness detected. As a result, the method is both dependable and cost-effective. The identification of drowsiness results in appropriate alerts and warning for the driver, thereby sparing life from tragic accidents.

5.2 Recommendation

It can constantly be improved on the driver drowsiness detection system for industrial use. It may well be performed better reliable and useful by applying these recommendations, which is establishing a mechanism for remote notifications includes sending personalised Text messages to the driver's emergencies contacts as well as napping options for drivers. Bluetooth Low Energy is used to transmit the data.

REFERENCES

1. Høye, A. (2016). How would increasing seat belt use affect the number of killed or seriously injured light vehicle occupants? Accident Analysis and Prevention, Vol. 88, pg.175 – 186. https://doi.org/10.1016/j.aap.2015.12.022

2. Kulanthayan, S., Law, T. H., Raha, A. R., & Radin, U. R. S. (2004). Seat Belt Use Among Car Users in Malaysia. IATSS Research, Vol. 28 (1), pg.19 – 25. https://doi.org/10.1016/S0386-1112(14)60088-1

3. Kareem A. (2003). Review of global menace of road accidents with special reference to malaysia- a social perspective. The Malaysian journal of medical sciences: MJMS, Vol. 10(2), pg.31 – 39.

4. Anu K L, Dr. Yasha jyothi M shirur, Mr. Prasannakumar Y. (2018). Design and Implementation of Driver Assistance System (DAS) Using Raspberry Pi to Enhance Driver Safety, International Research Journal of Engineering and Technology, Vol. 05, Issue 04, pg.4315 – 4319.

5. Musale, T., & Pansambal, B.H. (2016). Real Time Driver Drowsiness Detection system using Image processing, International Journal for Research in Engineering Application & Management, Vol. 02, Issue 08, pg.34 – 38.

6. A. Singh, H, Herunde, F. Furtado. (2020). Modified Haar-Cascade Model for Face Detection Issues, International Journal of Research in Industrial Engineering, Vol. 9 (2), pg.143–171. DOI: 10.22105/riej.2020.226857.1129

7. Prakash N Ekhande, Sharad A Rumane, Mayur A Ahire (2015), Document Image Binarization Using Image Segmentation Algorithm in Parallel Environment, International Journal of Computer Science and Mobile Computing, Vol. 4, Issue. 1, pg.360 – 366

8. Ashish Naire, Umesh Patil, Vaijanath Patane, Prof. S.Y.Sawant (2017). Raspberry Pi Based Drowsiness Detection System, International Journal & Magazine of Engineering Technology, Management and Research, Vol. 4, Issue 5, pg.775 – 779

9. F.M. Javed Mehedi Shamrat, Anup Majumder, Probal Roy Antu, Saykot KumarBarmon, Itisha Nowrin, Rumesh Ranjan. (2022). Human Face Recognition Applying Haar Cascade Classifier. International Conference on Pervasive Computing and Social Networking, Vol. 317, pp 143-157. https://doi.org/10.1007/978-981-16-5640-8_12

10. C. B. Yuvaraj, M. Srikanth, V. S. Kumar, Y. V. S. Murthy and S. G. Koolagudi. (2017). An approach to maintain attendance using image processing techniques, 2017 Tenth International Conference on Contemporary Computing (IC3), pp. 1-3, doi: 10.1109/IC3.2017.8284353.

11. P. Viola and M. Jones. (2001). Rapid object detection using a boosted cascade of simple features, Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. pp. I-I, doi: 10.1109/CVPR.2001.990517

12. B.D. Parameshachari. (2021). Logistic sine map (LSM) based partial image encryption, in 2021 National Computing Colleges Conference (NCCC), IEEE, pp. 1–6, doi: 10.1109/NCCC49330.2021.9428854.

13. M.K. Kumar, B.D. Parameshachari, S. Prabu, S. Liberata Ullo. (2020). Comparative analysis to identify efficient technique for interfacing BCI system, IOP Conference Series: Materials Science and Engineering, IOP Publishing, Vol. 925 (1), https://doi.org/10.1088/1757-899x/925/1/012062

14. Sanjaya, W. S. Mada & Anggraeni, Dyah & Zakaria, Kiki & Juwardi, Atip & Munawwaroh, Madinatul. (2017). The design of face recognition and tracking for human-robot interaction. 315-320. 10.1109/ICITISEE.2017.8285519

15. Rein-Lien Hsu, M. Abdel-Mottaleb and A. K. Jain. (2002). Face detection in color images, in IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 24 (5), pp. 696-706, doi: 10.1109/34.1000242.

16. Anil Kumar Biswal, Debabrata Singh, Binod Kumar Pattanayak, Debabrata Samanta, Ming-Hour Yang. (2021). IoT-Based Smart Alert System for Drowsy Driver Detection, Wireless Communications and Mobile Computing, Vol. 2021, Article ID 6627217, 13 pages, https://doi.org/10.1155/2021/6627217

APPENDICES

Appendix A: ***Drowsiness detector system Algorithm***

```
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 pygame

import argparse

import imutils

import time

import dlib

import cv2


def sound_alarm():

    pygame.mixer.init()

    pygame.mixer.music.load("alarm.WAV")

    pygame.mixer.music.play()
```

```
def sound_alarm1():

    pygame.mixer.init()

    pygame.mixer.music.load("yawnalarm.wav")

    pygame.mixer.music.play()



def eye_aspect_ratio(eye):

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

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

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



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

    return ear



def final_ear(shape):

    (lStart, lEnd) =
face_utils.FACIAL_LANDMARKS_IDXS["left_eye"]

    (rStart, rEnd) =
face_utils.FACIAL_LANDMARKS_IDXS["right_eye"]
```

```
    leftEye = shape[lStart:lEnd]

    rightEye = shape[rStart:rEnd]


    leftEAR = eye_aspect_ratio(leftEye)

    rightEAR = eye_aspect_ratio(rightEye)


    ear = (leftEAR + rightEAR) / 2.0

    return (ear, leftEye, rightEye)


def lip_distance(shape):

    top_lip = shape[50:53]

    top_lip = np.concatenate((top_lip, shape[61:64]))


    low_lip = shape[56:59]

    low_lip = np.concatenate((low_lip, shape[65:68]))


    top_mean = np.mean(top_lip, axis=0)

    low_mean = np.mean(low_lip, axis=0)
```

```
    distance = abs(top_mean[1] - low_mean[1])

    return distance


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())


EYE_AR_THRESH = 0.25

EYE_AR_CONSEC_FRAMES = 30

YAWN_THRESH = 30

ALARM_ON = False

COUNTER = 0

TOTAL = 0
```

```
print("** Loading the predictor and detector...")

# detector = dlib.get_frontal_face_detector()

detector =
cv2.CascadeClassifier("haarcascade_frontalface_default.xml")

# Faster but less accurate

predictor =
dlib.shape_predictor('shape_predictor_68_face_landmarks.dat')



print("*** Starting Video Stream")

vs = VideoStream(src=args["webcam"]).start()

# vs= VideoStream(usePiCamera=True).start()   //For Raspberry Pi

time.sleep(1.0)



while True:



    frame = vs.read()

    frame = imutils.resize(frame, width=450)

    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
```

```
    # rects = detector(gray, 0)
    rects = detector.detectMultiScale(gray, scaleFactor=1.4,
                                       minNeighbors=5, minSize=(30, 30),
flags=cv2.CASCADE_SCALE_IMAGE)
    # for rect in rects:
    for (x, y, w, h) in rects:
        rect = dlib.rectangle(int(x), int(y), int(x + w),
int (y + h))

        shape = predictor(gray, rect)
        shape = face_utils.shape_to_np(shape)

        eye = final_ear(shape)
        ear = eye[0]
        leftEye = eye[1]
        rightEye = eye[2]
```

```
        distance = lip_distance(shape)



        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)



        lip = shape [48:60]

        cv2.drawContours(frame, [lip], -1, (0, 255, 0), 1)



        if ear < EYE_AR_THRESH:

            COUNTER += 1



            if COUNTER >= EYE_AR_CONSEC_FRAMES:

                TOTAL += 1

                COUNTER = 0
```

```
                cv2.putText(frame, "DROWSINESS ALERT !!", (60,
230),
                            cv2.FONT_HERSHEY_SIMPLEX, 1, (209,
80, 0, 255), 2)
                if not ALARM_ON:
                    ALARM_ON = True
                    t =Thread(target=sound_alarm)
                    t.setDaemon(True)
                    t.start()
        else:
            COUNTER = 0
            ALARM_ON = False

        if (distance > YAWN_THRESH):
                cv2.putText(frame, "Yawn Alert !", (140, 250),
                            cv2.FONT_HERSHEY_SIMPLEX, 1, (209,
80, 0, 255), 2)
```

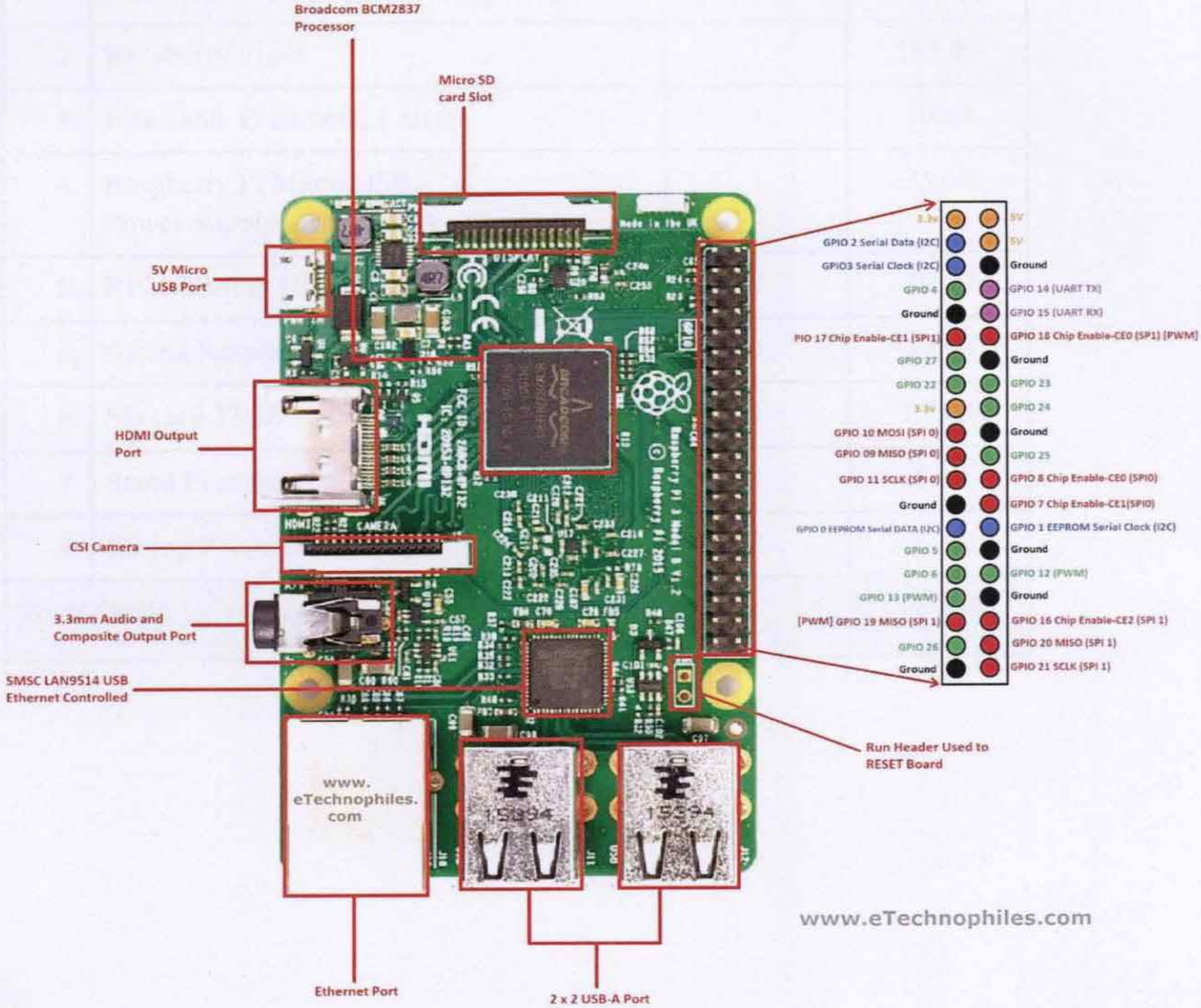
```
                if not ALARM_ON:

                    ALARM_ON = True

                    t = Thread(target=sound_alarm1)

                    t.start()

        else:

            ALARM_ON = False



        cv2.putText(frame, "Blinks: {}".format(TOTAL), (10, 30),

                cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)

        cv2.putText(frame, "EAR: {:.2f}".format(ear), (300, 30),

                cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)

        cv2.putText(frame, "YAWN: {:.2f}".format(distance),
(300, 60),

                    cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255),
2)



    cv2.imshow("Frame", frame)

    key = cv2.waitKey(1) & 0xFF
```

```
    if key == ord("q"):

        break


cv2.destroyAllWindows()

vs.stop()
```

Appendix B: ***Datasheet, Costing, Gantt chart***

Raspberry Pi 3B	
Establish date	29/2/2016
SoC	BCM2837
CPU	Quad Cortex A53 @ 1.2GHz
Instruction set	ARMv8-A
GPU	400MHz Video Core IV
RAM	1GB SDRAM
Storage	Micro-SD
Ethernet	10 / 100
Wireless	802.11n / Bluetooth 4.0
Video Input	Camera interface, used Pi camera or Pi NoIR camera
Video Output	HDMI / Composite
Audio Output	HDMI / Headphone
GPIO	40
Power source	5V with Mirco USB or GPIO header

Raspberry Pi 3 GPIO Pinout and Pin diagram



Costing of drowsiness detection system

No.	Component	Quantity	Cost (RM)
1.	5MP Night Vision Camera Raspberry Pi	1	58.50
2.	Raspberry Pi 3B	1	155.00
3.	Flex cable Pi camera, 1 meter	1	10.90
4.	Raspberry Pi Micro USB Power Supply - 5.1V, 2.5A	1	35.00
5.	RPi HDMI to HDMI Cable	1	20.00
6.	Casing Raspberry Pi 3B	1	10.00
8.	SD card 32GB	1	29.00
7.	Stand Pi camera	1	6.99
9.	Casing Pi camera	1	42.20
Total Cost			***367.59***

Gantt Chart

TASK	SEATBELT CONTROL SYSTEM																									
	MARCH 21			APRIL 21				MAY 21				OCT 21				NOV 21				DEC 21				JAN 22		
	1	2	3	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3
Project briefing	X	X																								
SDP title assigned	X	X																								
Planning																										
Literature review			X	X	X	X	X	X	X	X	X															
Research on problem statement of project			X	X	X	X	X	X	X	X	X															
Clarify the objective and specify the main target of project				X	X	X																				
Specific the scope and concept of the project				X	X	X																				
Clarify hardware and software to propose						X	X	X	X	X	X	X	X	X	X	X	X	X	X							
Researching on programming algorithm						X	X	X	X	X	X	X	X	X	X	X	X	X	X							
Design																										
Design on flowchart							X	X	X	X	X									X	X	X	X			
Concept and design details							X	X	X	X	X									X	X	X	X			
Design on circuit diagram							X	X	X	X	X									X	X	X	X			
Developing																										
Programming structure									X	X	X	X	X	X	X											
Configure Raspberry Pi microcontroller												X	X	X	X											
Fabricating																										
Component preparation															X	X	X	X	X	X						
Connection for system part																			X	X	X	X	X			
Testing on the system installation																						X	X	X	X	X
Integration system and testing																						X	X	X	X	X