APPENDIX 1

DRIVER DROWSINESS DETECTION USING RASPBERRY PI

A PROJECT REPORT

Submitted by

VISHWA POOJITHA J V K (910619106070)

SARUMATHY M (9106191060)

VAISHALY R (9106191060)

in partial fulfilment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

K.L.N. COLLEGE OF ENGINEERING, POTTAPALAYAM

ANNA UNIVERSITY::CHENNAI 600 025

MAY 2023

APPENDIX 2

ANNA UNIVERSITY :: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report “DRIVER DROWSINESS DETECTION USING RASPBERRY PI” is the bonafide work of

“ VISHWA POOJITHA J V K(910619106070),

SARUMATHY M(9106191060),

VAISHALY R(910619106065)”

carried out the project work under my supervision.

SIGNATURE	**SIGNATURE**
Dr. V. KejalakshmiM.E.,Ph.D.	**Mr.D.ANAND., M.E.,**
HEAD OFTHEDEPARTMENT	**SUPERVISOR**
	ASSISTANTPROFESSOR
Department of Electronics and	Department of Electronics and
Communication Engineering,	Communication Engineering,
K.L.N. College of Engineering,	K.L.N. College of Engineering,
Pottapalayam,	Pottapalayam,
Sivagangai –630612	Sivagangai – 630612

Submitted for the project Vice-Voce examination held on ....

INTERNALEXAMINER	EXTERNALEXAMINER

ACKNOWLEDGEMENT

First and foremost we thank and praise the almighty god for his guidance and protection throughout the course of our study. It is worth to quote that "Behind every successful efforts there lays contribution from numerous sources irrespective of their magnitude". At this juncture we take this opportunity to express our sincere and whole hearted thanks to all.

We extend our gratitude to our college President, **Er.K.N.K.KARTHIK,** K.L.N. College of Engineering for making us march towards the glory of success.

We extend our sincere thanks to our college Secretary and Correspondent **Dr. K.N.K.GANESH,** K.L.N. College of Engineering, for the motivation throughout the academic works. We express our thanks to our principal **Dr. A.V.RAMPRASAD,** for his kind cooperation and then keen interest by which our project ends in no vain.

We would like to express our gratefulness to our beloved Head of the Department, **Dr.V.KEJALAKSHMI M.E., Ph.D.,** and Project Co-ordinator **Mr. KUMARESH AP(Sr.Gr)/ECE** for her invaluable guidance and encouragement for completing this project work.

We express our deep gratitude to our project guide Mr**.D.ANAND M.E,** Assistant Professor, Department of Electronics and Communication Engineering, who greatly supported us, and helped us to complete this project successfully.

Last but not least, we also thank our parents, all staffs in our department and friends who have given a lot of motivation, valuable suggestions and helped us to complete the project in time.

Finally, we thank each and every one who helped us to do this project.

ABSTRACT

Drowsy driving is one of the major causes behind fatal road accidents. In recent years driver fatigue is one of the major causes of vehicle accidents in the world. This certainly highlights the fact that across the world the total numbers of road traffic deaths are very high due to driver’s drowsiness.A direct way of measuring driver fatigue is measuring the state of the driver i.e. drowsiness. So it is very important to detect the drowsiness of the driver to save life and property. The aim of the project is to develop an alarm system for a driver while they are sleepy and not focusing on the road.

This system is a real time system which captures image continuously and measures the state of the eye according to the specified algorithm and gives warning if required. Real Time Driver Drowsiness System using Image Processing to capture drivers eye state using computer vision. In this paper, we present to detect driver drowsiness using Open CV, raspberry pi and image processing. The major aim of this project is to detect eyes and yawn of the driver. Ocular measure to detect driver eye condition and possible vision based on eye closure is well suited for real world driving conditions, since it can detect the eyes open/ closed state using a camera.

APPENDIX 3

TABLE OF CONTENTS

LIST OF TABLE(S)

LIST OF FIGURES

LIST OF ABBREVIATIONS

S NO	ACRONYM	ABBREVIATION
1	**IoT**	Internet of Things
2	**FPGA**	Field Programmable Gate Arrays
3	**NICHD**	The National Institute of Child Health and Human Development SBCU Special Baby Care Units
4	**VR**	Virtual Reality
5	**AR**	Augmented Reality
6	**GPU**	Graphical Processing Unit
7	**SpO2**	Oxygen Saturation
8	**IoT-BBMS**	Internet of Things Based Baby Monitoring System for Smart Cradle
9	**NodeMCU**	Node Microcontroller Unit
10	**MQTT**	Message Queuing Telemetry Transport
11	**WIFI**	Wireless Fidelity
12	**GSM**	Global System for Mobile Communications
13	**DC**	Direct Current
14	**SMEs**	Small and Medium Enterprises
15	**LCD**	Liquid Crystal Display
17	**RTOS**	Real Time Operating system
18	**ROM**	Read Only Memory
19	**HW/SW**	Hardware/Software
20	**ASIP**	Application Specific Instruction Set Processor
21	**IO**	Input-Output

22	**A-D**	Analog to Digital
23	**D-A**	Digital to Analog
24	**PIR Sensor**	Passive InfraRed Sensor
25	**MIC**	Microphone
26	**UWB**	Ultra Wide Band
27	**AI**	Artificial Intelligence
28	**CPU**	Central Processing Unit
29	**RAM**	Random Access Memory
30	**GPIO**	General Purpose Input/Output
31	**TV**	Television
32	**HDMI**	High Definition Multimedia Interface
33	**MIPI**	Mobile Industry Processor Interface
34	**DSI**	Display Serial Interface
35	**SDIO**	Secure Digital Input Output
36	**ARM**	Acorn RISC Machine
37	**RISC**	Reduced-instruction-set Computing
38	**PXE**	Preboot Execution Environment
39	**SD Card**	Card Secure Digital Card
40	**IR**	InfraRed
41	**MP**	Megapixel
42	**CCTV**	Closed-Circuit Television
43	**ADC**	Analog to Digital Converter
44	**DC**	Direct Current
45	**PCB**	Printed Circuit Board
46	**VCC**	Voltage Common Collector
47	**GND**	Ground
48	**DO**	Digital Value Output
49	**AO**	Analog Value Output
50	**DB**	Decibel
51	**PWM**	Pulse width modulation

52	**TTL**	Transistor–transistor logic

CHAPTER 1

INTRODUCTION

1.1 SYSTEM:

A system is an arrangement in which all its unit assemblies work together according to a set of rules. It can also be defined as a way of working, organizing or doing one or many tasks according to a fixed plan. For example, a watch is a time displaying system. Its components follow a set of rules to show time. If one of its parts fails, the watch will stop working. So we can say, in a system, all its subcomponents depend on each other.

1.2 EMBEDDED SYSTEM:

As its name suggests, Embedded means something that is attached to another thing. An embedded system can be thought of as a computer hardware system having software embedded in it. An embedded system can be an independent system or it can be a part of a large system. An embedded system is a microcontroller or microprocessor based system which is designed to perform a specific task. For example, a fire alarm is an embedded system; it will sense only smoke.

1.3 COMPONENTS OFEMBEDDED SYSTEM:

An embedded system has three components:

- It has hardware.

- It has applicationsoftware.

- It has a Real Time Operating system (RTOS) that supervises the application software and provides a mechanism to let the processor run a process as per scheduling by following a plan to control the latencies. RTOS defines the way the system works. It sets the rules during the execution of the application program. A small scale embedded system may not have RTOS.

So we can define an embedded system as a Microcontroller based, software driven, reliable, real-time control system.

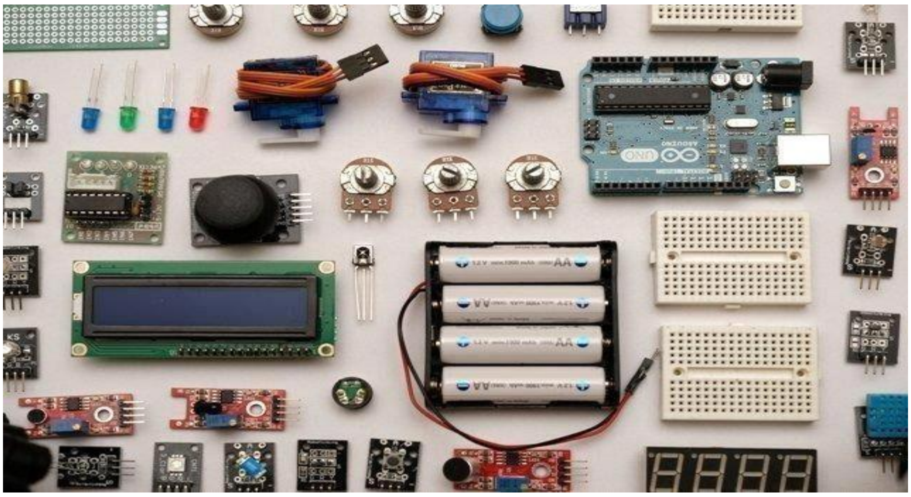


Figure 1.3 Components of Embedded system

1.4 HISTORY OF EMBEDDED OPERATING SYSTEMS

The first modern, real-time embedded computing system was the Apollo Guidance Computer, developed in the 1960s by Dr. Charles Stark Draper at the Massachusetts Institute of Technology for the Apollo Program. The Apollo Guidance Computer was designed to collect data automatically and provide mission-critical calculations for the Apollo Command Module and Lunar Module.

In 1971, Intel released the first commercially available microprocessor unit -- the Intel 4004 -- an early microprocessor that still required support chips and external memory; in 1978 the National Engineering Manufacturers Association released a standard for programmable microcontrollers, improving the embedded system design; and by the early 1980s, memory, input and output system components had been integrated into the same chip as the processor, forming a microcontroller.

The microcontroller-based embedded system would go on to be incorporated into every aspect of consumers' daily lives, from credit card readers and cell phones, to traffic lights and thermostats.

.

1.4 CHARACTERISTICS OF AN EMBEDDEDSYSTEM:

- **Single-functioned** – An embedded system usually performs a specialized operation and does the same repeatedly. For example: A pager always functions as apager.
- **Tightly constrained** − All computing systems have constraints on design metrics, but those on an embedded system can be especially tight. Design metrics is a measure of an implementation's features such as its cost, size, power, and performance. It must be of a size to fit on a single chip, must perform fast enough to process data in real time and consume minimum power to extend batterylife.
- **Reactive and Real time** − Many embedded systems must continually react to changes in the system's environment and must compute certain results in real time without any delay. Consider an example of a car cruise controller; it continually monitors and reacts to speed and brake sensors. It must compute acceleration or de-accelerations repeatedly within a limited time; a delayed computation can result in failure to control the car.
- **Microprocessors based** − It must be microprocessor or microcontroller based.
- **Memory**− It must have a memory, as its software usually embeds in ROM. It does not need any secondary memories in the compute.

- **Connected**– It must have connected.

- **HW-SW systems** − Software is used for more features and flexibility. Hardware is used for performance and security.

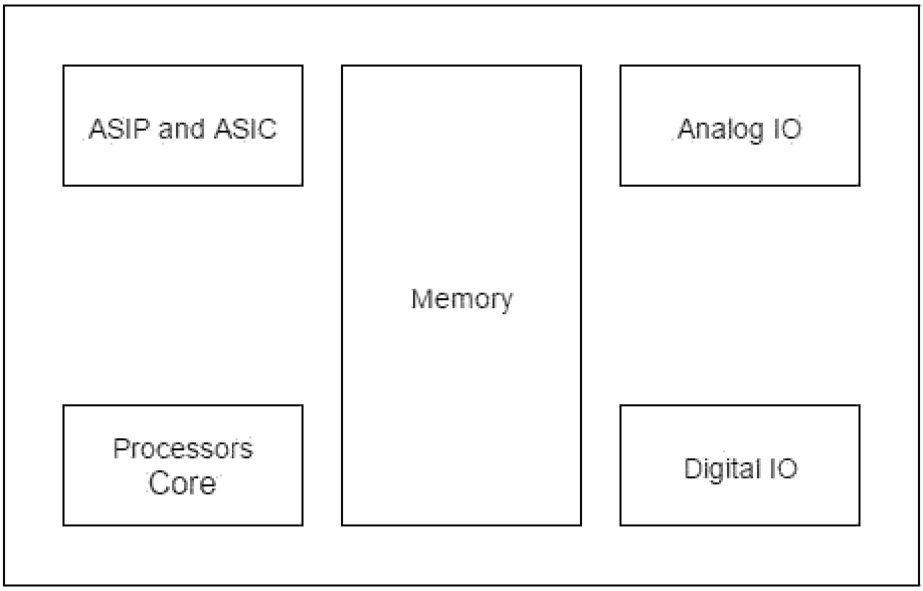


Figure 1.5 Block diagram of Embedded system

Advantages:

- EasilyCustomizable
- Low powerconsumption
- Lowcost
- Enhanced performance

Disadvantages:

- High development effort
- Larger time to market

1.6 BASIC STRUCTURE OF AN EMBEDDEDSYSTEM:

The following illustration shows the basic structure of an embedded system:

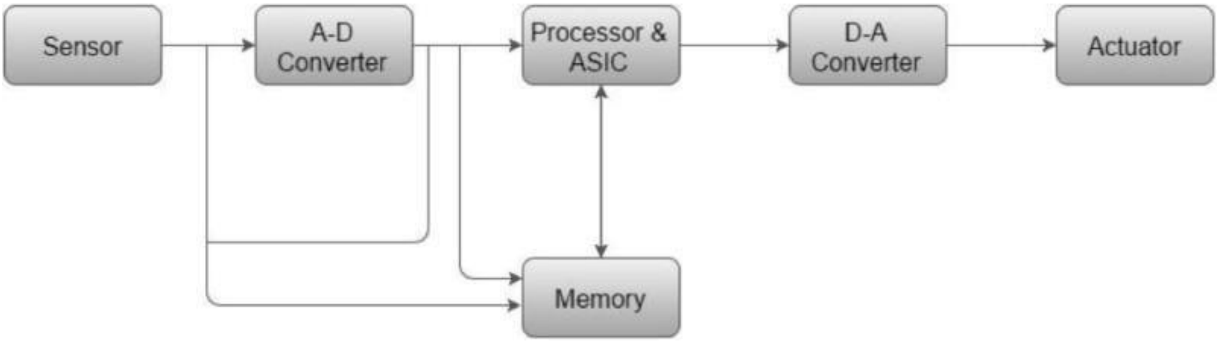


Figure1.6 Basic Structure of Embedded System

- **Sensor−** It measures the physical quantity and converts it to an electrical signal which can be read by an observer or by any electronic instrument like an A2D converter. A sensor stores the measured quantity to the memory.

- **A-D Converter −** An analog-to-digital converter converts the analog signal sent by the sensor into a digitalsignal.

- **Processor & ASICs −** Processors process the data to measure the output and store it to the memory**.**

- **D-A Converter −** A digital-to-analog converter converts the digital data fed by the processor to analogdata

- **Actuator −** An actuator compares the output given by the D-A Converter to the actual (expected) output stored in it and stores the approved output.

1.7 WORKING OF EMBEDDEDSYSTEM:

Embedded systems are managed by microcontrollers or digital signal processors (DSP), application-specific integrated circuits (ASIC), field-programmable gate arrays (FPGA), GPU technology, and gate arrays. These processing systems are integrated with components dedicated to handling electric and/or mechanical interfacing. Embedded systems programming instructions, referred to as firmware, are stored in read-only memory or flash memory chips, running with limited computer hardware resources. Embedded systems connect with the outside world through peripherals, linking input and output devices.

1.8 FUTURE TRENDS IN EMBEDDEDSYSTEMS:

The industry for embedded systems is expected to continue growing rapidly, driven by the continued development of Artificial Intelligence (AI), Virtual Reality (VR) and Augmented Reality (AR), machine learning , deep learning, and the Internet of Things(IoT). The cognitive embedded system will be at the heart of such trends as: reduced energy consumption, improved security for embedded devices, cloud connectivity and mesh networking, deep learning applications, and visualization tools with real time data. According to a 2018 report published by QYResearch, the global market for the embedded systems industry was valued at $68. 9 billion in 2017 and is expected to rise to $105.7 billion by the end of 2025.

1.9 APPLICATION OF EMBEDDEDSYSTEM:

1. Telecommunications

- Telephoneswitches
- Cellphones
- Routers

2. Consumer electronics

- MP3 Players
- Televisionsets
- Digitalcameras

3. Household appliances

- Microwaveovens
- Washingmachines
- Dishwashers

4. Transportation systems

- Avionics
- Automobiles
- Electricvehicles,
- Hybridvehicles

5. Medical equipments

- MRI
- CTScan

CHAPTER 2

LITERATURE SURVEY

REVIEW OF FACE RECOGNITION METHODS

Face recognition methods divided into categories

- Knowledge-based methods
- Feature-invariant methods
- Template matching methods
- Appearance-based methods

2.1 KNOWLEDGE-BASED METHODS:

Knowledge-based methods are encoding our knowledge of human faces. These are rule-based methods. They try to capture our knowledge of faces, and translate them into a set of rules. It’s easy to guess some simple rules. For example, a face usually has two symmetric eyes, and the eye area is darker than the cheeks. Facial features could be the distance between eyes or the color intensity difference between the eye area and the lower zone. The big problem with these methods is the difficulty in building an appropriate set of rules. There could be many false positives if the rules were too general. On the other hand, there could be many false negatives if the rules were too detailed. A solution is to build hierarchical knowledge-based methods to overcome these problems. These methods show themselves efficient with simple inputs. But, what happens if a man is wearing glasses? There are other features that can deal with that problem. For example, there are algorithms that detect face-like textures or the color of human skin.

DRAWBACKS:

- difficulty in building an appropriate set of rules
- false positives if the rules were too general
- false negatives if the rules were too detailed
- hierarchical knowledge-based methods used for this but it detect face based
- on textures or the color of human skin

2.2 FEATURE-INVARIANT METHODS:

Feature-invariant methods that try to find invariant features of a face despite its angle or position. Facial recognition utilizes distinctive features of the faceincluding: distinct micro elements like: Mouth, Nose, Eye, Cheekbones, Chin, Lips, Forehead, Ears, Upper outlines of the eye sockets, the areas surrounding the cheekbones, the sides of the mouth, and the location of the nose and eyes. The distance between the eyes, the length of the nose and the angle of the jaw.

DRAWBACK:

- Facial expression

2.3 TEMPLATE MATCHING METHODS:

These algorithms compare input images with stored patterns of faces or features. Template matching methods try to define a face as a function. One can try to find a standard template of all the faces. Different features can be defined independently. For example, a face can be divided into eyes, face contour, nose and mouth. Also a face model can be built by edges. But these methods are limited to faces that are frontal. A face can also be represented as a shape. Other templates use the relation between face.

regions in terms of brightness and darkness. These standard patterns are compared to the input images to detect faces. This approach is simple to implement, but it’s insufficient for face detection. It cannot achieve good results with variations in pose, scale and shape.

DRAWBACKS:

- Limited to faces that are frontal.
- A face can also be represented as a shape.
- Other templates use the relation between face regions in terms of brightness and darkness.
- This approach is simple to implement, but it‟s insufficient for face detection.
- It cannot achieve good results with variations in pose, scale and shape.

2.4 APPEARANCE-BASED METHODS:

A template matching method whose pattern database is learnt from a set of training images. In general, appearance-based methods rely on techniques from statistical analysis and machine learning to find the relevant characteristics of face images. Principal Component Analysis (PCA) is well-organized method for face recognition. It is one of the most usable methods for a face image. It is used to reduce the dimensionality of the image and also holds some of the variations in the image data. It is projecting face image data into a feature space that covers the significant variations among known facial images. Those significant features are known as “Eigen faces”, because they are the eigenvectors or Principal Component of the set of faces. That is not necessary to correspond to the features such as eyes, ears, and noses. The projection operation characterizes an individual face by a weighted sum of the Eigen faces features. So to recognize a particular face, it is necessary only to compare these weights to those individuals.

The Eigen Object Recognizer class applies PCA on each image, the results of which will be an array of Eigen values. To perform PCA several steps are undertaken:

Stage 1: Subtract the Mean of the data from each variable (our adjusted data)
Stage 2: Calculate and form a covariance Matrix

Stage 3: Calculate Eigenvectors and Eigenvalues from the covariance Matrix
Stage 4: Chose a Feature Vector (a fancy name for a matrix of vectors)

Stage 5: Multiply the transposed Feature Vectors by the transposed adjusted data

DRAWBACKS:

- Different head pose
- Different alignment
- Different facial expression

Distribution based Methods – LDA Algorithm

LDA also known as Fisher‟s Discriminant Analysis, is another dimensionality reduction technique. It is an example of a class specific method i.e. LDA maximizes the between – class scattering matrix measure while minimizes the within – class scatter matrix measure, which make it more reliable for classification. Lih-Heng Chan proposed a framework of facial biometric was designed based on two subspace methods i.e., Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA). First, PCA is used for dimension reduction, where original face images are projected into lowerdimensional face representations. Second, LDA was proposed to provide a solution of better discriminant. Both PCA and LDA features were presented to Euclidean distance measurement which is conveniently used as a benchmark. LDA-based methods outperform PCA for both face identification and verification. Fisher faces are one the most successfully widely used method for face recognition. It is based on appearance method. In 1930 Fisher developed linear/fisher discriminant analysis for face recognition which shows successful result in face recognition process. The disadvantage of LDA is that within the class the scatter matrix is always single, since the number of pixels in images is larger than the number of images so it can increase detection of error rate if there is a variation in pose and lighting condition within same images. So to overcome this problem many algorithms has been proposed. Because the fisher faces technique uses the advantage of within-class information so it minimizes the variation within class, so the problem with variations in the same images such as lighting variations can be overcome

DRAWBACKS:

- The face to classify must be in the DB .
- Can’t work well with high dimension.

CHAPTER -3

DRIVER DROWSINESS DETECTION USING RASPBERRY PI

3.1 INTRODUCTION:

Several surveys on road accidents says that around 30 percent of accidents are caused by fatigue of the driver. When driver drives for more than normal period for human then excessive fatigue is caused and also results in tiredness which drives the driver to sleepy condition or loss of consciousness. Drowsiness is a state in which the person is experiencing a decreased level of consciousness due to lack of energy. The word drowsy means the sensation of falling asleep. Driver exhaustion due to long working hours, lack of sleep and medical conditions is a significant cause of frequency and nodding the head.

3.2 OBJECTIVE:

Driver exhaustion may be characterized by behavioural reaction such as blinking eyes and yawning. Using those methods an intelligence system can be developed which would alert the driver in case drowsy condition and prevent accidents. Advantages and dis dvantages corresponding to each and every system is explained. Depending on advantages and disadvantages the most suitable method is chosen and proposed. Then the approach for entire system development is explained using a flow chart which includes capturing the image in real time continuously, then dividing it into frames. Then each frames are analyzed to find face first. If a face is detected then then next task is to locate the eyes. After the positive result of detecting eye the amount of closure of eye is determined and compared with the reference values for the drowsy state eye. If drowsy condition is found out then driver is alarmed else repeatedly the loop of finding face and detecting drowsy condition is carried out.

The specific objectives of the project include:

- In OpenCV, the face detection of the driver's image is carried out first followed by eye detection. The eye detection technique detects the open state of eyes only.
- HAAR cascade algorithm counts the number of times the status of opened eyes and yawn in each frame and calculates the criteria for detection of drowsiness.
- If the criteria are satisfied, then it seems the driver is said to be drowsy.
- Then the driver drowsiness system connected with display and buzzer perform actions to alert and correct the driver abnormal behavior.

3.3 OPENCV:

A comprehensive set of both classic and state-of-the-art computer vision and machine learning algorithms. These algorithms can be used to detect and recognize fa ces, identify OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products. Being a BSD-licensed 14 product, OpenCV makes it easy for businesses to utilize and modify the code. The library has more than 2500 optimized algorithms, which includes objects, classify human actions in videos, track camera movements, track moving objects, extract 3D models of objects, produce 3D point clouds from stereo cameras, stitch images together to produce a high resolution image of an entire scene, find similar images from an image database, remove red eyes from images taken using flash, follow eye movements, recognize scenery and establish markers to overlay it with augmented reality, etc. OpenCV has more than 47 thousand people of user community and estimated number of downloads exceeding 18 million. The library is used extensively in companies, research groups and by governmental bodies. Along with well-established companies like Google, Yahoo, Microsoft, Intel, IBM, Sony, Honda, Toyota that employ the library, there are many startups such as Applied Minds, VideoSurf, and Zeitera, that make extensive use of OpenCV. OpenCV's deployed uses span the range from stitching streetview images together, detecting intrusions in surveillance video in Israel, monitoring mine equipment in China, helping robots navigate and pick up objects at Willow Garage, detection of swimming pool drowning accidents in Europe, running interactive art in Spain and New York, checking runways for debris in Turkey, inspecting labels on products in factories around the world on to rapid face detection in Japan.

It has C++, Python, Java and MATLAB interfaces and supports Windows, Linux, Android and Mac OS. OpenCV leans mostly towards real-time vision applications and takes advantage of MMX and SSE instructions when available. A full-featured CUDAand OpenCL interfaces are being actively developed right now. There are over 500 algorithms and about 10 times as many functions that compose or support those algorithms. OpenCV is written natively in C++ and has a templated interface that works seamlessly with STL containers.

3.4 ALGORITHM:

HAAR Cascade:

A widely popular subject with a huge range of applications. Modern day Smart phones and Laptops come with in-built face detection softwares, which can authenticate the identity of the user. There are numerous apps that can capture, detect and process a face in real time, can identify the age and the gender of the user, and also can apply some really cool filters. The list is not limited to these mobile apps, as Face Detection also has a wide range of applications in Surveillance, Security and Biometrics as well. But the origin of its Success stories dates back to 2001, when Viola and Jones proposed the first ever detection framework for Real Time Face Detection in Video Footage. But how exactly does facial recognition work to classify faces, considering the large number of features as input and the striking similarities between humans?

Enter Haar classifiers, classifiers that were used in the first real-time face detector. A Haar classifier, or a Haar cascade classifier, is a machine learning object detection program that identifies objects in an image and video A detailed description of Haar classifiers can be seen in Paul Viola and Michael Jones's paper "Rapid Object Detection using a Boosted Cascade of Simple Features".

3.3 Facial recognition on an iPhone X

So what is Haar Cascade? It is an Object Detection Algorithm used to identify faces in an image or a real time video. The algorithm uses edge or line detection features proposed by Viola and Jones in their research paper "Rapid Object Detection using a Boosted Cascade of Simple Features" published in 2001. The algorithm is given a lot of positive images consisting of faces, and a lot of negative images not consisting of any face to train on them.

FEATURES:

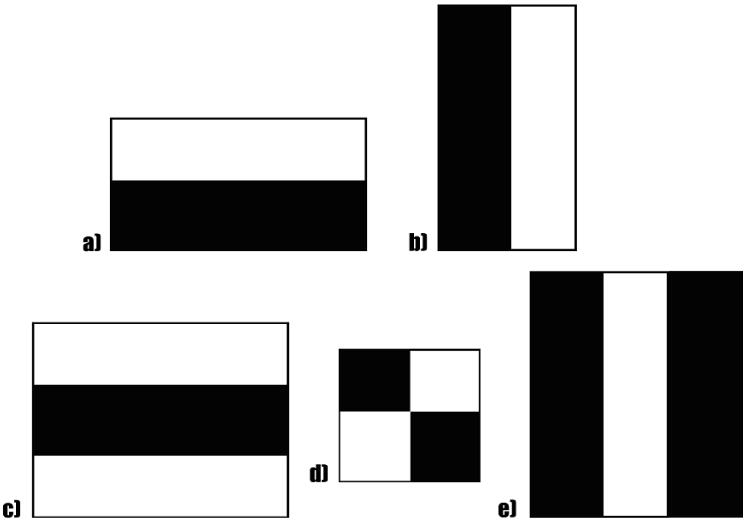
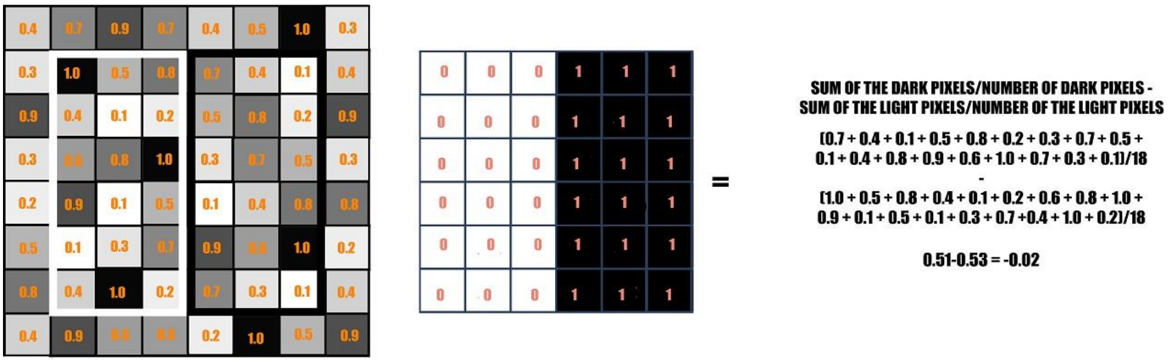


Figure 3.3 a) 7

The first contribution to the research was the introduction of the haar features shown above. This feature on the image makes it easy to find out the edges or the lines in the image, or to pick areas where there is a sudden change in the intensities of the pixels.



0.4	0.7	0.9	0.7	0.4	0.5	1.0	0.3
0.3	1.0	0.5	0.8	0.7	0.4	0.1	0.4
0.9	0.4	0.1	0.2	0.5	0.8	0.2	0.9
0.3	0.6	0.8	1.0	0.3	0.7	0.5	0.3
0.2	0.9	0.1	0.5	0.1	0.4	0.8	0.8
0.5	0.1	0.3	0.7	0.9	0.6	1.0	0.2
0.8	0.4	1.0	0.2	0.7	0.3	0.1	0.4
0.4	0.9	0.8	0.8	0.2	1.0	0.5	0.9

0	0	0	1	1	1
0	0	0	1	1	1
0	0	0	1	1	1
0	0	0	1	1	1
0	0	0	1	1	1
0	0	0	1	1	1

=

SUM OF THE DARK PIXELS/NUMBER OF DARK PIXELS -
SUM OF THE LIGHT PIXELS/NUMBER OF THE LIGHT PIXELS

(0.7 + 0.4 + 0.1 + 0.5 + 0.8 + 0.2 + 0.3 + 0.7 + 0.5 +
0.1 + 0.4 + 0.8 + 0.9 + 0.6 + 1.0 + 0.7 + 0.3 + 0.1)/18
-
(1.0 + 0.5 + 0.8 + 0.4 + 0.1 + 0.2 + 0.6 + 0.8 + 1.0 +
0.9 + 0.1 + 0.5 + 0.1 + 0.3 + 0.7 +0.4 + 1.0 + 0.2)/18

0.51-0.53 = -0.02

Figure 3.3 b) Binary calculation

Fig. The rectangle on the left is a sample representation of an image with pixel values 0.0 to 1.0. The rectangle at the center is a haar kernel which has all the light pixels on the left and all the dark pixels on the right.

3.5 FACE DETECTION :

The proposed system will start by capturing the video frames one by one. OpenCV provides extensive support for processing live videos. The system will Sdetect the face in the frame image for each frame. This system uses Viola-Jones object detector which is a machine learning approach for visual object detection (Paul Viola, 2004 and Paul Viola, 2001). This is achieved by making use of the Haar algorithm for face detection. Haarcascade is a well-known robust feature-based algorithm that can detect the face image efficiently. With the use of uses of cascade of stages, Haar algorithm able to remove the candidates that are non face. And each stage consists of combination of different Haar features and each feature in turn is classified by a Haar feature classifier. The inbuilt OpenCV xml "haarcascade_frontalface_alt2.xml" file is used to search and detect the face in individual frames. This file contains a number of features of the face and constructed by using a number of positive and negative samples.

First load the cascade file then pass the acquired frame to an edge detection function, which detects all the possible objects of different sizes in the frame. Since the face of the driver occupies a large part of the image, instead of detecting objects of all possible sizes, specify the edge detector to detect only objects of a particular size i.e. for face region.

Next, the output the edge detector is stored and this output is compared with the cascade file to identify the face in the frame. The output of this module is a frame with face detected in it. Only disadvantage in Haar algorithm is that it cannot extrapolate and does not work appropriately when the face is not in front of the camera axis. Once the face detection function has detected the face of the driver, the eyes detection function tries to detect the driver's eyes.

3.6 EYE DETECTION:

Once the face detection function has detected the face of the driver, the eyes detection function tries to detect the automobile driver's eyes. After face detection find eye region by considering eyes are present only in upper part of the face and from top edge of the face, extract eyes Region of Interest (ROI) by cropping mouth and hair, we mark it the region of interest. By considering the region of interest it is possible to reduce the amount of processing required and also speeds up the processing for getting exact eyes.

After the region of interest is marked, the edge detection technique is applied only on the region of interest .Then search for eyes in ROI; Circular Hough Transformation is used here to find shape of eyes (Rhody Chester, 2005). The main advantage of the Hough transform technique is that it is liberal to gaps in feature boundary descriptions and is relatively unaffected by image noise, unlike edge detectors. The OpenCV function HoughCircles () is used to detect circles in an eye image. CHT ensure that at most two eyes found. With the eye detection technique we will only be able to detect the open state of eyes.

3.7 DROWISENESS DETECTION:

After getting eyes the algorithm then counts the number of open eyes form each frame and determines the drowsiness. If the criteria are satisfied, then the driver is said to be drowsy. The buzzer connected to the system performs actions to correct the driver abnormal behaviour. For this system, the eye and the face classifiers are required. The HARR Classifier Cascade files built-in there with the Open CV contains different classifiers for the face and eye detection. The inbuilt Open CV xml "haarcascade_frontalface_alt2.xml"and function "Houghcircles ()" is used to search and detect the face followed by individual frames. The face detection and open eye detection have been carried out on each frame of the driver's captured facial image.

The variable Eyes total is assigned to store the number of open eyes found in each frame. A variable will store the number of successive frames in which the eyes found to be closed with the values like 0, 1, 2, 3… etc. Initially, this variable is set to 0.When both the eyes are open, and then Drowsy count will be 0. Drowsy count will increase when Eyes total < 2. For an eye blink, Drowsy count value is raised by 1. If the eye blinks in more than 4 frames, i.e. variable count is greater than or equal to 4, then the condition for drowsiness is met and an alarm will be signalled at real time.

3.8 YAWN DETECTION:

Yawn detection is performed in two main steps:

1. In the first step we detect the yawn component in the face independent of the mouth location. This component is basically the hole in the mouth as the results of wide mouth opening.
2. In the second step we will be using the mouth location to verify the validity of the detected components. After skin segmentation, the largest hole located inside the face is selected as the candidate for a yawning mouth. This hole is actually related to a non-skin area inside the face that can be related to eyes, mouth or open mouth. It can be assumed that the open mouth will be the largest of the three in a yawning state. We will then use the information from the detected mouth to verify the detected yawning mouth. The verification criteria is the number of pixels located in the yawning mouth wih respect to the number of mouth pixels as well as the relative location of the open mouth with respect to the lips.

CHAPTER-4

EXISTING SYSTEM

Our proposed system consists of various modules like video acquisition, face detection, eye detection and drowsiness detection. In addition to these it consists of raspberry pi which is a single-board computer, a buzzer which acts as an alarm. First video acquisition is achieved by pi camera placed on raspberry pi and acquired video is converted into a series of image/frames. Next step is detecting the driver's face using OpenCV
(Haar algorithm) followed by eye detection using Region Of Interest (ROI). Once eyes have been detected the next step is to detect whether eyes are in a closed or open state. If eyes are detected to open in each frame, no action is taken. But if eyes are detected to be closed and it is above threshold value then it means that the driver is feeling drowsy and a sound alarm along with water sprinkler are triggered.

CHAPTER 5

PROPOSED SYSTEM

5.1 DESCRIPTION:

The primary focus is given to the faster drowsiness detection and processing of data. The number of frames in which the eyes are kept closed is monitored and then counted. If the number of frames exceeds a threshold value, then a warning message is generated on the display showing that the drowsiness is detected. The system should be capable of detecting drowsiness in spite of the skin colour and complexion of the driver, spectacles used by the driver and the darkness level inside the vehicle. All these objectives have been well satisfied by choosing the system using appropriate classifiers in OpenCV for eye closure detection.

In this algorithm, first a driver's image is acquired by the camera for processing. In OpenCV, the face detection of the driver's image is carried out first followed by eye detection. The eye detection technique detects the open state of eyes only. Then the algorithm counts the number of open eyes in each frame and calculates the criteria for detection of drowsiness. If the criteria are satisfied, then the driver is said to be drowsy. The display and buzzer connected to the system perform actions to correct the driver abnormal behavior.

For this system, the face and eye classifiers are required. The HARR Classifier Cascade files inbuilt on OpenCV include different classifiers for the face detection and the eyes detection. The inbuilt OpenCV xml "haarcascade_frontalface_alt2.xml" is used to search and detect the face in individual frames. The classifier "haarcascade_eye_tree_eyeglasses.xml" is used to detect eyes in the open state from the detected face. The system does not detect in the closed state of the eyes. The face detection and open eye detection have been carried out on each frame of the driver's facial image acquired from the camera. The variable Eyestotal is assigned to store the number of open eyes (0, 1 and 2) detected in each frame. The variable Drowsycount is assigned for storing the number of successive frames in which the eyes have been Drowsycount gets incremented when Eyestotal < 2. For an eye blink, Drowsycount value gets incremented to 1. If the eyeblink occurs for more than 4 frames, i.e. Drowsycount >=4, then the criterion for drowsiness is satisfied.

Face detection and recognition

The camera module is set where individuals go into the everyday schedule and video is taken inside a distance under 5 meters. A camera is utilized for taking video which contains different formats from which any of the housings can be utilized for confronting insistence and venturing the cooperation.

Picking a successful calculation for face affirmation or acknowledgment is fundamental in this proposed work. There are various face location calculations available in OpenCV, for example, Eigenfaces, Fisher appearances, and Neighborhood Binary Pattern Histograms. Considering the expectations for the ongoing affirmation a calculation which has been chosen is the Haar Cascade Algorithm for face recognition and affirmation. It is available in the OpenCV source library.

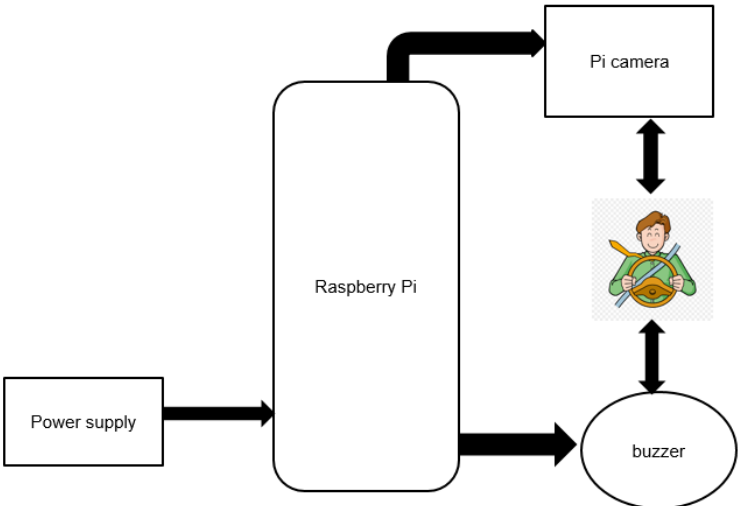
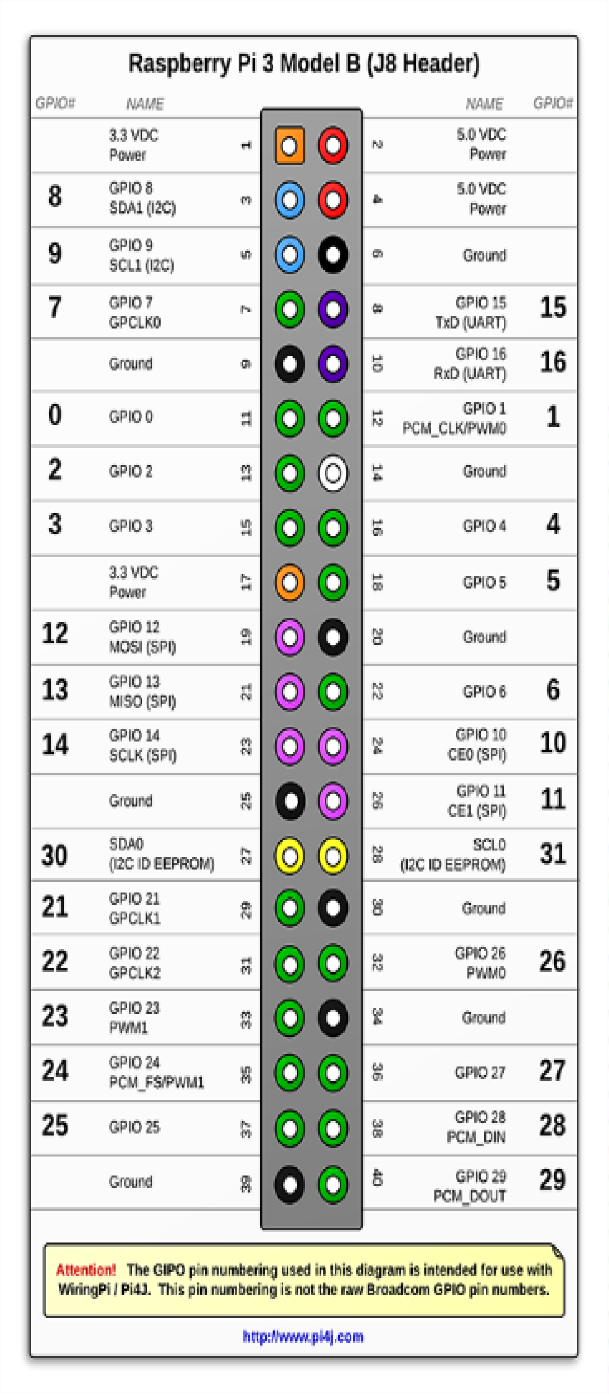


Figure 5.1 Block Diagram of Proposed System

5.2 CIRCUIT DIAGRAM:



Raspberry Pi 3 Model B (J8 Header)

GPIO#	NAME			NAME	GPIO#
	3.3 VDC Power	1	2	5.0 VDC Power	
8	GPIO 8 SDA1 (I2C)	3	4	5.0 VDC Power	
9	GPIO 9 SCL1 (I2C)	5	6	Ground	
7	GPIO 7 GPCLK0	7	8	GPIO 15 TxD (UART)	15
	Ground	9	10	GPIO 16 RxD (UART)	16
0	GPIO 0	11	12	GPIO 1 PCM_CLK/PWM0	1
2	GPIO 2	13	14	Ground	
3	GPIO 3	15	16	GPIO 4	4
	3.3 VDC Power	17	18	GPIO 5	5
12	GPIO 12 MOSI (SPI)	19	20	Ground	
13	GPIO 13 MISO (SPI)	21	22	GPIO 6	6
14	GPIO 14 SCLK (SPI)	23	24	GPIO 10 CE0 (SPI)	10
	Ground	25	26	GPIO 11 CE1 (SPI)	11
30	SDA0 (I2C ID EEPROM)	27	28	SCL0 (I2C ID EEPROM)	31
21	GPIO 21 GPCLK1	29	30	Ground	
22	GPIO 22 GPCLK2	31	32	GPIO 26 PWM0	26
23	GPIO 23 PWM1	33	34	Ground	
24	GPIO 24 PCM_FS/PWM1	35	36	GPIO 27	27
25	GPIO 25	37	38	GPIO 28 PCM_DIN	28
	Ground	39	40	GPIO 29 PCM_DOUT	29

Attention! The GIPO pin numbering used in this diagram is intended for use with WiringPi / Pi4J. This pin numbering is not the raw Broadcom GPIO pin numbers.

http://www.pi4j.com

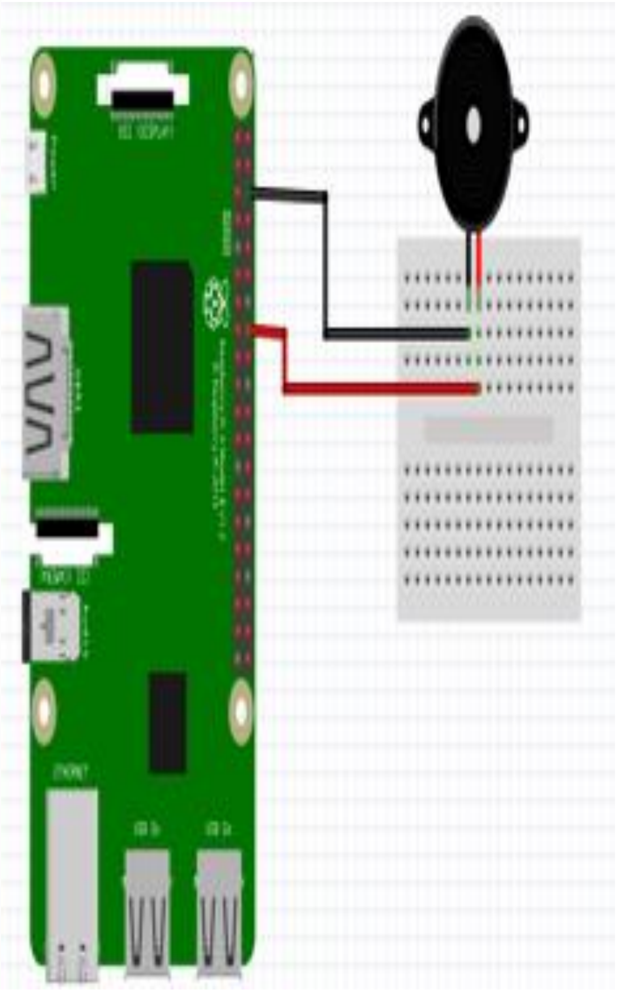


Figure 5.2 Circuit Diagram

5.3 PROJECT SETUP:

Figure 5.2 a) Pi camera

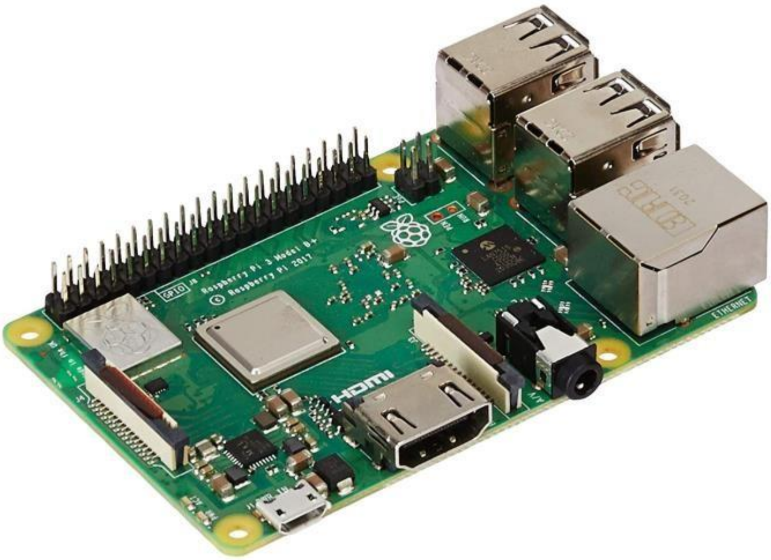


Figure 5.2 b) Raspberry pi 3 model B+

Figure 5.2 c)Buzzer

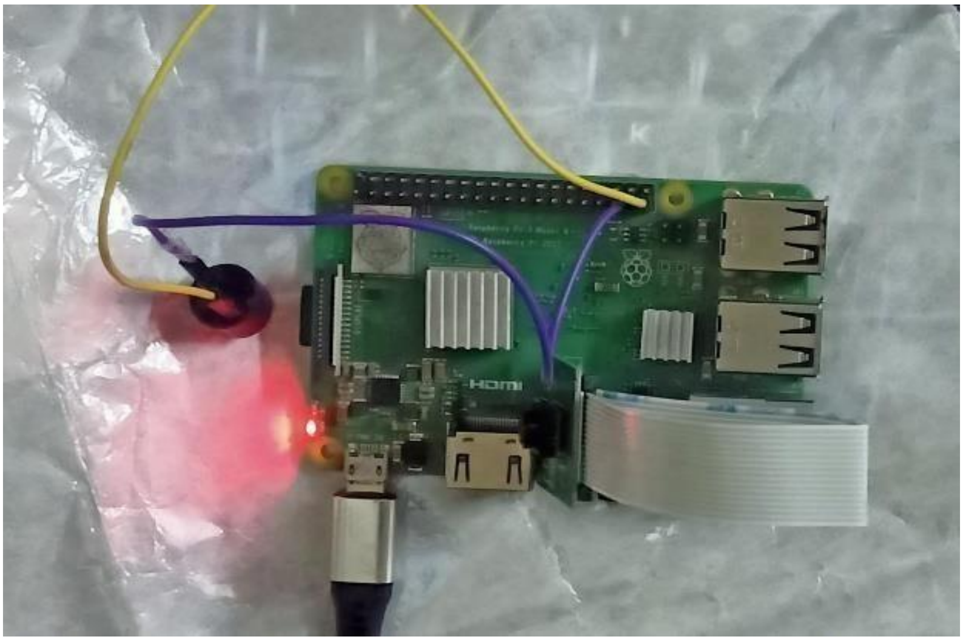


Figure 5.2 d) Raspberry pi connected with buzzer

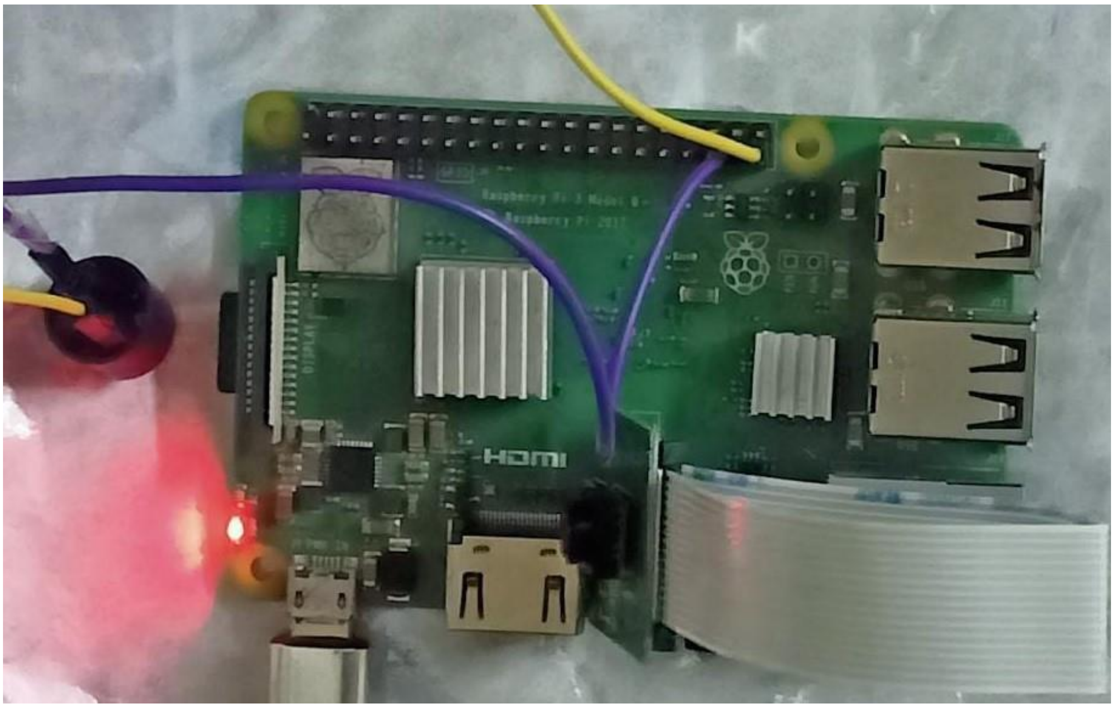
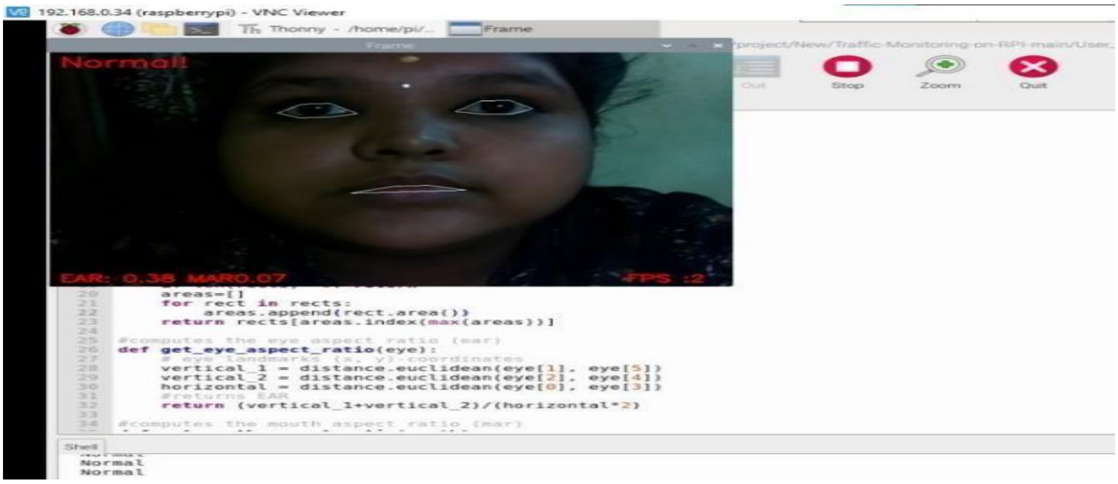


Figure 5.2 e) Whole setup

OUTPUT:



192.168.0.34 (raspberrypi) - VNC Viewer
Thonny - /home/pi/...
Frame
Frame
/project/New/Traffic-Monitoring-on-RPI-main/User_
Out
Stop
Zoom
Quit
Normal!
EAR: 0.38 MAR0.07
FPS :2
Shell

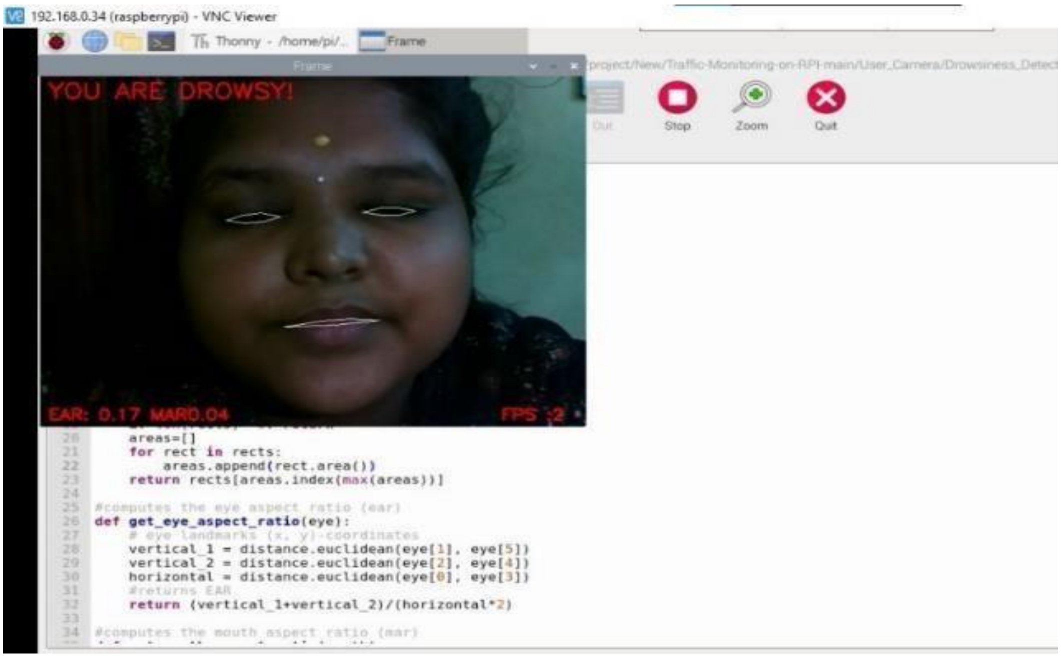
```
    areas=[]
    for rect in rects:
        areas.append(rect.area())
    return rects[areas.index(max(areas))]

#computes the eye aspect ratio (ear)
def get_eye_aspect_ratio(eye):
    # eye landmarks (x, y)-coordinates
    vertical_1 = distance.euclidean(eye[1], eye[5])
    vertical_2 = distance.euclidean(eye[2], eye[4])
    horizontal = distance.euclidean(eye[0], eye[3])
    #returns EAR
    return (vertical_1+vertical_2)/(horizontal*2)

#computes the mouth aspect ratio (mar)
```

```
Normal
Normal
```

Figure 5.2 f) Face recognition



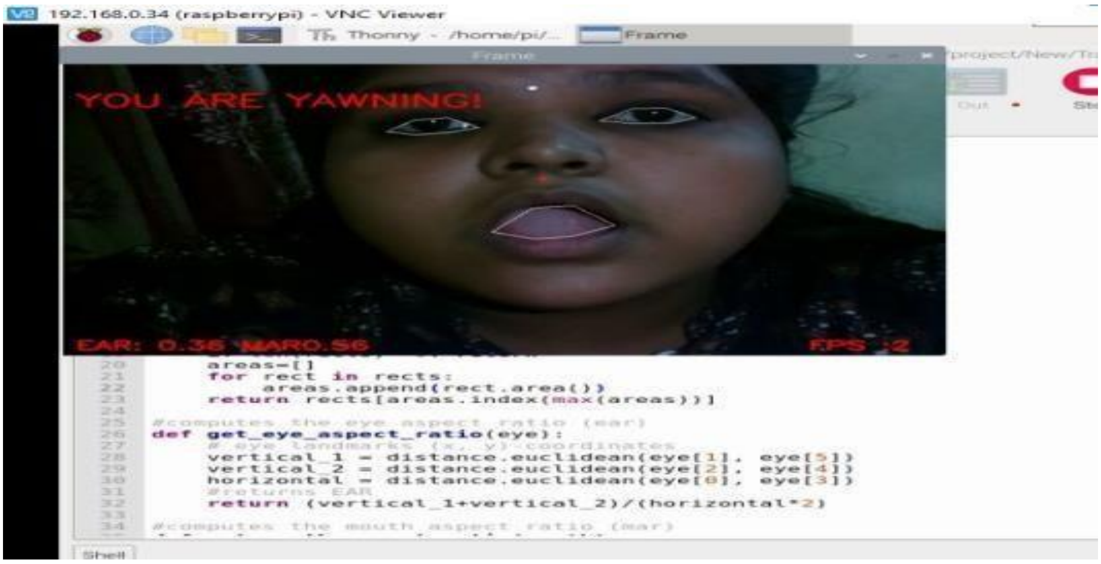
192.168.0.34 (raspberrypi) - VNC Viewer
Thonny - /home/pi/...
Frame
Frame
project/New/Traffic-Monitoring-on-RPI-main/User_Camera/Drowsiness_Detect
Out
Stop
Zoom
Quit
YOU ARE DROWSY!
EAR: 0.17 MARO.04
FPS :2

```
    areas=[]
    for rect in rects:
        areas.append(rect.area())
    return rects[areas.index(max(areas))]

#computes the eye aspect ratio (ear)
def get_eye_aspect_ratio(eye):
    # eye landmarks (x, y)-coordinates
    vertical_1 = distance.euclidean(eye[1], eye[5])
    vertical_2 = distance.euclidean(eye[2], eye[4])
    horizontal = distance.euclidean(eye[0], eye[3])
    #returns EAR
    return (vertical_1+vertical_2)/(horizontal*2)

#computes the mouth aspect ratio (mar)
```

Figure 5.2 g) Eye detection



```
    areas=[]
    for rect in rects:
        areas.append(rect.area())
    return rects[areas.index(max(areas))]

#computes the eye aspect ratio (ear)
def get_eye_aspect_ratio(eye):
    # eye landmarks (x, y)-coordinates
    vertical_1 = distance.euclidean(eye[1], eye[5])
    vertical_2 = distance.euclidean(eye[2], eye[4])
    horizontal = distance.euclidean(eye[0], eye[3])
    #returns EAR
    return (vertical_1+vertical_2)/(horizontal*2)

#computes the mouth aspect ratio (mar)
```

Figure 5.3 h) Yawn detection

CHAPTER 6

HARDWARE DESCRIPTION

6.1 RASPBERRY PI:

Raspberry Pi is the name of a series of single-board computers made by the Raspberry Pi Foundation, a UK charity that aims to educate people in computing and create easier access to computing education.

The Raspberry Pi launched in 2012, and there have been several iterations and variations released since then. The original Pi had a single-core 700MHz CPU and just 256MB RAM, and the latest model has a quad-core 1.4GHz CPU with 1GB RAM. The main price point for Raspberry Pi has always been $35 and all models have been $35 or less, including the Pi Zero, which costs just $5.

All over the world, people use Raspberry Pi to learn programming skills, build hardware projects, do home automation, and even use them in industrial applications.

The Raspberry Pi is a low cost, credit-card sized computer that plugs into a computer monitor or TV, and uses a standard keyboard and mouse. It is a capable little device that enables people of all ages to explore computing, and to learn how to program in languages like Scratch and Python. It’s capable of doing everything you’d expect a desktop computer to do, from browsing the internet and playing high-definition video, to making spreadsheets, word-processing, and play in games.

The Raspberry Pi is a very cheap computer that runs Linux, but it also provides a set of GPIO (general purpose input/output) pins that allow you to control electronic components for physical computing and explore the Internet of Things(IOT).

6.2 MODELS:

There have been three generations of Raspberry Pi: Pi 1, Pi 2, and Pi 3, and there has generally been a Model A and a Model B of most generations. Model A is a cheaper variant and tends to have reduced RAM and ports like USB and Ethernet. The Pi Zero is a spinoff of the original (Pi 1) generation, made even smaller and cheaper.

Model (release date) Pi 1 Model B(2012)

Pi 1 Model A(2013)

Pi 1 Model B+(2014)

Pi 1 Model A+(2014)

Pi 2 Model B (2015)

Pi Zero (2015)

Pi 3 Model B (2016) Pi Zero W (2017)

Pi 3 Model B+(2018)

Pi 3 Model A+(2019)

Pi 4 Model B(2019)

6.3 RASPBERRY PI 3 MODAL B+ :

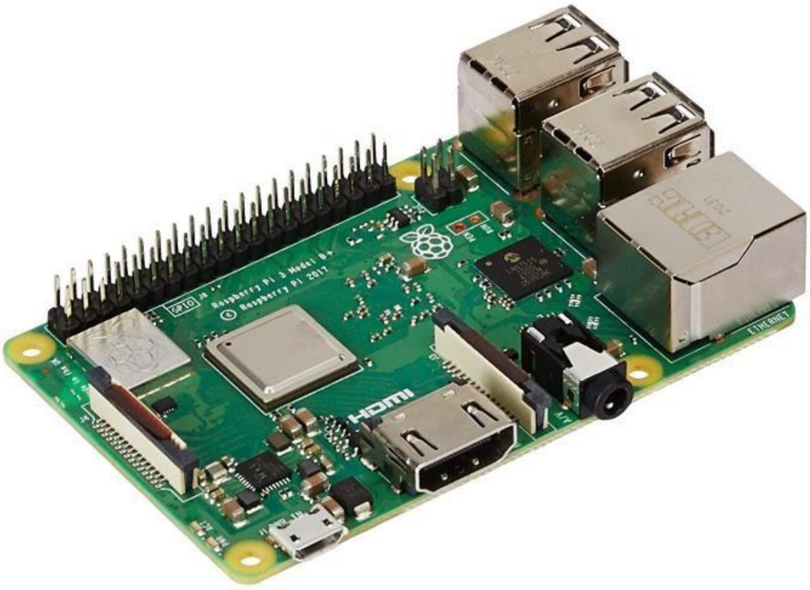


Figure 6.3 a) RASPBERRY PI 4 MODEL B

The Raspberry Pi 3 Model is the latest product in the famous range of electronic boards and has the same mechanical structure as its big sisters. But with its more powerful processor and optimised Ethernet connection, it’s the real star in the family. ith its quad core, 64-bit CPU clocked at 1.4 GHz, a throughput of 300 Mbps over USB 2.0, and the addition of support for Bluetooth/BLE and Wi-Fi, the new raspberry pi will become the ideal electronic platform for all your connected or IoT-based projects.

It will also delight designers as it complies with modularity standards. In a nutshell, you don’t need to test the wireless compliance of the Pi you can directly fit the board into your project, and save some precious time and money.

Technical specifications of the Raspberry Pi 3:

- Processor: Broadcom BCM2837B0, Cortex-A53 64-bit SoC @ 1.4 GHz
- Memory: 1 GB LPDDR2 SDRAM
- 2.4 GHz and 5 GHz IEEE 802.11.b/g/n/ac wireless LAN, Bluetooth 4.2, BLE
- Gigabit Ethernet USB 2.0 (300 Mbps)
- 4 USB 2.0 ports
- 40 GPIO pins

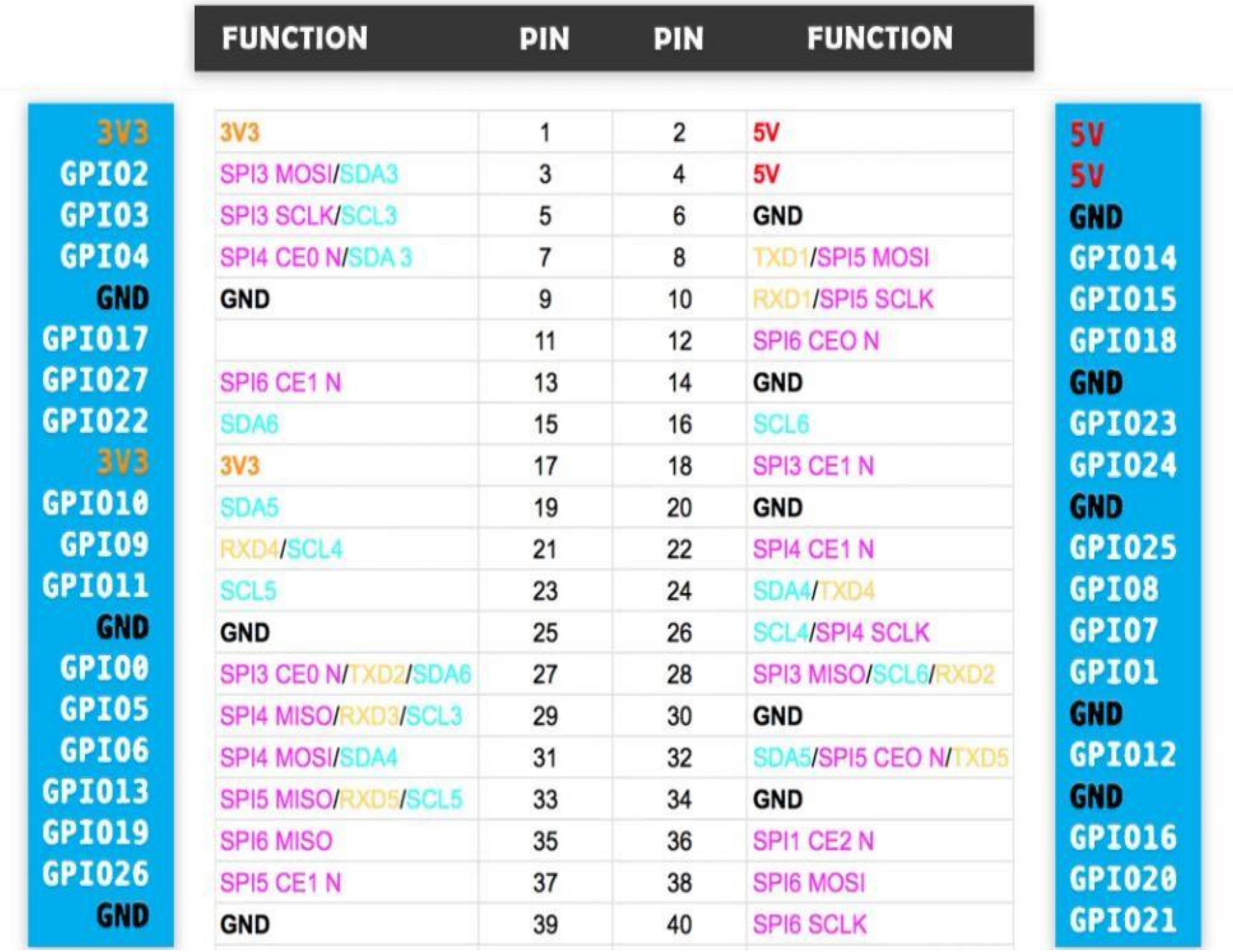
Video and audio:

- HDMI output
- MIDI DSI display port
- MIPI CSI camera port
- 4-pole stereo output and composite video port
- Multimedia: H.264, MPEG-4 decode (1080p30); H.264 encode (1080p30); OpenGL ES 1.1, 2.0 graphics
- Micro SD port

- 5V/2.5A DC power input via micro USB connector

Power supply:

- via the GPIO connector
- via Power over Ethernet (PoE) (requires a separate PoE HAT)
- Operating temp



	FUNCTION	PIN	PIN	FUNCTION	
3V3	3V3	1	2	5V	5V
GPIO2	SPI3 MOSI/SDA3	3	4	5V	5V
GPIO3	SPI3 SCLK/SCL3	5	6	GND	GND
GPIO4	SPI4 CE0 N/SDA 3	7	8	TXD1/SPI5 MOSI	GPIO14
GND	GND	9	10	RXD1/SPI5 SCLK	GPIO15
GPIO17		11	12	SPI6 CEO N	GPIO18
GPIO27	SPI6 CE1 N	13	14	GND	GND
GPIO22	SDA6	15	16	SCL6	GPIO23
3V3	3V3	17	18	SPI3 CE1 N	GPIO24
GPIO10	SDA5	19	20	GND	GND
GPIO9	RXD4/SCL4	21	22	SPI4 CE1 N	GPIO25
GPIO11	SCL5	23	24	SDA4/TXD4	GPIO8
GND	GND	25	26	SCL4/SPI4 SCLK	GPIO7
GPIO0	SPI3 CE0 N/TXD2/SDA6	27	28	SPI3 MISO/SCL6/RXD2	GPIO1
GPIO5	SPI4 MISO/RXD3/SCL3	29	30	GND	GND
GPIO6	SPI4 MOSI/SDA4	31	32	SDA5/SPI5 CEO N/TXD5	GPIO12
GPIO13	SPI5 MISO/RXD5/SCL5	33	34	GND	GND
GPIO19	SPI6 MISO	35	36	SPI1 CE2 N	GPIO16
GPIO26	SPI5 CE1 N	37	38	SPI6 MOSI	GPIO20
GND	GND	39	40	SPI6 SCLK	GPIO21

Figure 6.3b) Raspberry Pi 4 Pin Configuration

6.3.1 Board connectors:

Name	**Description**
Ethernet	Base T Ethernet Socket
USB	2.0 (Four sockets)
Audio Output	3.5mm Jack and HDMI
Video output	HDMI
Camera Connector	15-pin MIPI Camera Serial Interface (CSI-2)
Display Connector	Display Serial Interface (DSI) 15 way flat flex cable connector with two data lanes and a clock lane.
Memory Card Slot	Push/Pull Micro SDIO

6.3.2 Features:

- 1.4GHz 64-bit quad core ARMCortex-A-53CPU
- Gigabit Ethernet(over USB)
- Dual-band Wifi andBluetooth
- Power-over-Ethernet (PoE)pins
- Improved PXE and USBbooting
- Improved thermalmanagement

This is a big update to the existing Pi 4, including a re-spin of the BCM2837 CPU (running faster at 1.4GHz), new dual-band wireless connectivity (so you can connect to both 2.4- and 5-GHz WiFi networks), better thermal control, and more. As always, you'll need an up-to-date Raspbian SD card, but the new model is backwards-compatible with all previous models and runs the same operating system and programs.

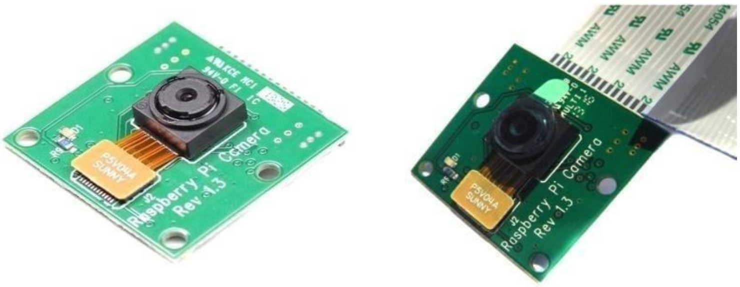
Specifications:

- Color: White + Green OR White +Blue
- Infrared Sensor with Control CircuitBoard
- The Sensitivity and Holding Time Can beAdjusted
- Working Voltage Range: DC 4.5V-20V
- Current Drain:<60uA
- Detection Range:<140°
- Voltage Output: High/Low level Signal: 3.3V TTLoutput
- Detection Distance: 3 to 7m (can beadjusted)
- Delay Time: 5 to 200s (Can be Adjusted, Default 5s +/-3%)
- Blockade time: 2.5s (Default)
- Work temperature:-20-+80°C
- Dimension: 3.2cm x 2.4cm x 1.8cm(Approx)

- Sensitive Setting: Turn to Right, Distance Increases (About 7M); Turn to Left, Distance Reduce (About3M)

- Time Setting: Turn to Right, Time Increases (About 200S); Turn to Left, Time Reduce (About5S).

6.4 Pi CAMERA



The system is currently in its initial stages. For the current user, a setup is created and optimized. The crucial step is for the driver's head location to be successful. [4] If the driver's head is found, it is then simple to process the image and determine the driver's current condition.

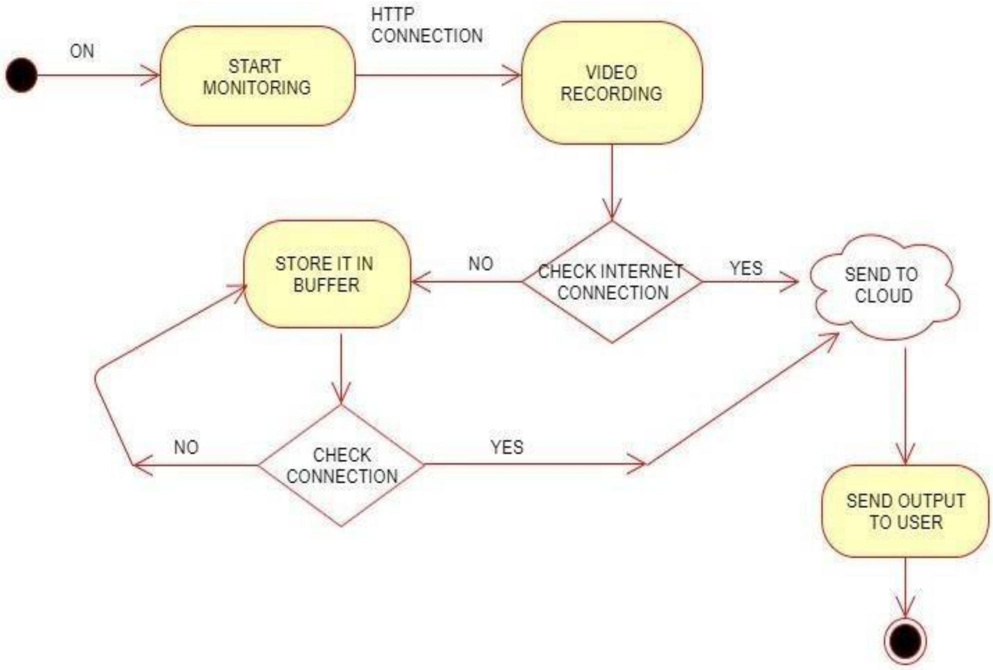


Figure 6.4 b) Pi Camera Flow Diagram

6.4.1 Camera components:

(1) The camera box, which holds and protects the sensitive film from all light except that entering through thelens;

(2) Film, on which the image is recorded, a light-sensitive strip usually wound on a spool, either manually or automatically, as successive pictures aretaken;

(3) The light control, consisting of an aperture or diaphragm and a shutter, both oftenadjustable;

(4) The lens , which focuses the light rays from the subject onto the film, creating the image, and which is usually adjustable by moving forward or back, changing the focus;and

(5) The viewing system, which may be separate from the lens system (usually above it) or may operate through it by means of amirror.

Note that there are actually two versions of this camera module. The simpler and cheaper one is shown in Fig - and it is the one we use in thisproject.

The 5MP Raspberry Pi 3 Model B Camera Module with Cable equips flexible cable for attaching with Raspberry Pi 3 Model B. The 5MP camera module is perfect for small Raspberry Pi projects which have very little space allowance just boot up the latest version of Raspbian

The high-definition 5MP camera delivers outstanding photos but can also shoot video, ideal for drones or a CCTV project. The lightweight camera module allows for it to be used in more practical roles, such as a hidden camera or even a camera for a Pi-phone, forexample.

This Raspberry Pi Camera Module is a custom designed add-on for Raspberry Pi. It attaches to Raspberry Pi by way of one of the two small sockets on the board upper surface. This interface uses the dedicated CSI interface, which was designed especially for interfacing to cameras. The CSI bus is capable of extremely high data rates, and it exclusively carries pixel data.

The board itself is tiny, at around 25mm x 23mm x 8mm. It also weighs just over 3g, making it perfect for mobile or other applications where size and weight are important. It connects to Raspberry Pi by way of a short flexible ribbon cable. The camera connects to the BCM2835 processor on the Pi via the CSI bus, a higher bandwidth link which carries pixel data from the camera back to the processor. This bus travels along the ribbon cable that attaches the camera board to thePi.

The sensor itself has a native resolution of 5 megapixels and has a fixed focus lens onboard. In terms of still images, the camera is capable of 2592 x 1944 pixel static images, and also supports 1080p30, 720p60 and 640x480p60/90 video.

No adapters required,this camera will plug directly into the Raspberry Pi 3 Model B camera port.

6.4.2 Features of raspberry pi 5mp camera module withcable:

- Fully Compatible with Both the Model A, Model B and Model B+ RaspberryPi
- 5MP Omnivision5647 CameraModule
- Still Picture Resolution: 2592 x1944
- Video: Supports 1080p @ 30fps, 720p @ 60fps and 640x480p 60/90

Recording

- 15-pin MIPI Camera Serial Interface - Plugs Directly into the Raspberry PiBoard
- Size: 20 x 25 x9mm
- Weight3g

6.5 BUZZER:

Active and passive buzzers are the two main categories of buzzer. When a voltage is applied across a passive buzzer, a tone is released. The generation of various tones also needs a particular signal. These are covered because active buzzers are much easier to use. While you can connect an active buzzer like an LED, you won't need to use a resistor to protect them because they are a little more durable.

CHAPTER 7

CONCLUSION AND FUTURE WORK

7.1 CONCLUSION:

In this system we have implemented the face recognition and driver's drowsiness, in toxication, or reckless behavior can be quickly identified by a system developed to monitor driver abnormalities. The driver's eye closure served as the basis for the development of a drowsiness detection system that can distinguish between normal eye blinking and drowsiness while also detecting drowsiness while driving. Accidents caused by drivers who are sleepy can be avoided with the proposed system. The system functions well even when drivers are wearing eyeglasses and in low light situations if the camera produces better results. Several in-house image processing algorithms are used to gather information about the position of the head , eyes and mouth. The monitoring system can determine whether the eyes are open or closed. There is a warning signal that is given when the eyes are closed for too long , when mouth should be yarned that time also give warning alert . Continuous eye closures are used by processing to determine the driver's level of alertness.

7.2 FUTURE WORK :

Drowsiness detection systems may be added to aircraft in the future to warn pilots when a pilot is getting sleepy.

•Drunk drivers can also use the alcoholsensor.

•Drowsiness detection systems can be implemented in schools and colleges in the future to alert staff members when astudent is showing signs of being sleepy ina class.

CHAPTER 8

REFRERENCE

[1]. Kouzani A, He F, Sammut K. Fractal face representation and recognition. IEEE Int Conf Syst Man Cybern1997;2.http://dx.doi.org/10.1109/ICSMC.1997.638231

[2]. Viola, P., & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on (Vol.1,pp.I-511).IEEE

[3]. Bagus G. Pratama, IgiArdiyanto, Teguh B. Adji, A Review on Driver Drowsiness Based on Image, Bio-Signal, and Driver Behavior, IEEE, July 2017.

[4]. E. Vural, M. Cetin, A. Ercil, G. Littlewort, M. Bartlett, and J. Movellan, "Drowsy driver detection through facial movement analysis," in International Workshop on Human-Computer Interaction, 2007, pp. 6- 18: Springer.

[5]. P.Thiffault and J. Bergeron, "Monotony of road environment and driver fatigue: a simulator study," Accident Analysis & Prevention, vol. 35, no. 3, pp. 381-391, 2003.

[6]. “Drowsy Driving,” National Highway Traffic Safety Administration (NHTSA), 01-Feb-2018.[Online].

[7].FACES ALGORITHM”,ARPN Journal of Engineering and Applied SciencesVOL.11,NO.13

[8].Rafael C. Gonzalez. 2009. Digital Image Processing. Pearson Education India.

[9]. Kyong Hee Lee, Whui Kim, Hyun Kyun Choi, Byung Tae Jan. A Study on Feature Extraction Methods Used to Estimate a Drivers Level of Drowsiness, IEEE, February 2019.

[10]. Fouzia, Roopalakshmi R, Jayantkumar A Rathod, Ashwitha S, Supriya K, Driver Drowsiness Detection System Based on Visual Features. , IEEE, April 2018.