A PROJECT REPORT

ON

"ACCIDENT ALERT SYSTEM"

Submitted to

UNIVERSITY OF MUMBAI

In Partial Fulfilment of the Requirement for the Award of

BACHELOR'S DEGREE IN
COMPUTER ENGINEERING

BY

Ansari Mohd Aamir Mohd Sharif Yasmin	**ROLL NUMBER 17CO38**
Nathani Aamir Haneef Nasima	**ROLL NUMBER 17CO39**
Khan Nazim Matiullah Sanjida	**ROLL NUMBER 17CO31**
Ansari Faisal	**ROLL NUMBER 17DCO62**

UNDER THE GUIDANCE OF
PROF.KHAN MUBASHIR

DEPARTMENT OF COMPUTER ENGINEERING
Anjuman-I-Islam's Kalsekar Technical Campus
SCHOOL OF ENGINEERING & TECHNOLOGY
Plot No. 2 3, Sector - 16, Near Thana Naka,
Khandagaon, New Panvel - 410206

2020-2021

AFFILIATED TO

UNIVERSITY OF MUMBAI

A PROJECT II REPORT
ON

"ACCIDENT ALERT SYSTEM"

Submitted to
UNIVERSITY OF MUMBAI

In Partial Fulfilment of the Requirement for the Award of

BACHELOR'S DEGREE IN COMPUTER ENGINEERING

BY

Ansari Mohd Aamir Mohd Sharif Yasmin	**ROLL NUMBER 17CO38**
Nathani Aamir Haneef Nasima	**ROLL NUMBER 17CO39**
Khan Nazim Matiullah Sanjida	**ROLL NUMBER 17CO31**
Ansari Faisal	**ROLL NUMBER 17DCO62**

UNDER THE GUIDANCE OF
PROF. KHAN MUBASHIR

DEPARTMENT OF COMPUTER ENGINEERING
Anjuman-I-Islam's Kalsekar Technical Campus
SCHOOL OF ENGINEERING & TECHNOLOGY
Plot No. 2 3, Sector - 16, Near Thana Naka,
Khandagaon, New Panvel - 410206

2020-2021
AFFILIATED TO

UNIVERSITY OF MUMBAI

Anjuman-i-Islam's Kalsekar Technical Campus

Department of Computer Engineering
SCHOOL OF ENGINEERING & TECHNOLOGY
Plot No. 2 3, Sector - 16, Near Thana Naka,
Khandagaon, New Panvel - 410206

CERTIFICATE

This is certify that the project entitled

"ACCIDENT ALERT SYSTEM"

submitted by

Ansari Mohd Aamir Mohd Sharif Yasmin	**ROLL NUMBER 17CO38**
Nathani Aamir Haneef Nasima	**ROLL NUMBER 17CO39**
Khan Nazim Matiullah Sanjida	**ROLL NUMBER 17CO31**
Ansari Faisal	**ROLL NUMBER 17DCO62**

is a record of bonafide work carried out by them, in the partial fulfilment of the requirement for the award of Degree of Bachelor of Engineering (Computer Engineering) at ***Anjuman-I-Islam's Kalsekar Technical Campus, Navi Mumbai*** under the University of MUMBAI. This work is done during year 2020-2021, under our guidance.

Date:25/05/2021

(Prof. Mubashir Khan)
Project Supervisor

(Prof. Kalpana R.Bodke)
Project Coordinator

(Prof. Tabrez Khan)
HOD, Computer Department

DR. ABDUL RAZAK HONNUTAGI
Director

External Examiner

Acknowledgements

I would like to take the opportunity to express my sincere thanks to my guide **Prof. Khan Mubashir**, Assistant Professor, Department of Computer Engineering, AIKTC, School of Engineering, Panvel for his invaluable support and guidance throughout my project research work. Without his kind guidance & support this was not possible.

I am grateful to him/her for his timely feedback which helped me track and schedule the process effectively. His/her time, ideas and encouragement that he gave is help me to complete my project efficiently.

We would like to express deepest appreciation towards **DR. ABDUL RAZAK HONNUTAGI**, Director, AIKTC, Navi Mumbai, **Prof. TABREZ KHAN**, Head of Department of Computer Engineering and **Prof. KALPANA R.BODKE**, Project Coordinator whose invaluable guidance supported us in completing this project.

At last we must express our sincere heartfelt gratitude to all the staff members of Computer Engineering Department who helped me directly or indirectly during this course of work.

ANSARI MOHD AAMIR
NATHANI AAMIR HANEEF
KHAN NAZIM MATIULLAH
ANSARI FAISAL

Project I Approval for Bachelor of Engineering

This project entitled ***Äccident Alert System¨*** by ***Ansari Mohd Aamir (17CO38), Nathani Aamir (17CO39), Khan Nazim (17CO31), Ansari Faisal (17DCO62)*** is approved for the degree of ***Bachelor of Engineering in Department of Computer Engineering.***

Examiners

1. .....
2. .....

Supervisors

1. .....
2. .....

Chairman

.....

Declaration

I declare that this written submission represents my ideas in my own words and where others ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Student Name: Ansari Mohd Aamir
Roll Number:17CO38

Student Name: Nathani Aamir
Roll Number:17CO39

Student Name: Khan Nazim
Roll Number:17CO31

Student Name: Ansari Faisal
Roll Number: 17DCO62

ABSTRACT

— Nowadays, Driver drowsiness is one of the major cause for most of the accidents in the world. Detecting the driver eye tiredness is the easiest way for measuring the drowsiness of driver. The existing systems in the literature, are providing slightly less accurate results due to low clarity in images and videos, which may result due to variations in the camera positions.

In order to solve this problem, a driver drowsiness detection system is proposed in this paper, which makes use of eye blink counts for detecting the drowsiness. Specifically, the proposed framework, continuously analyzes the eye movement of the driver and alerts the driver by activating the vibrator when he/she is drowsy. When the eyes are detected closed for too long time, a vibrator signal is generated to warn the driver. The experimental results of the proposed system, which is implemented on Open CV and Raspberry Pi environment with a single camera view, illustrate the good performance of the system in terms of accurate drowsiness detection results and thereby reduces the road accidents.

Keywords: Drowsiness, Fatigue Detection, Raspberry Pi, Image Processing, Eye Detection, EAR

Contents

List of Figures

List of Tables

Chapter 1

Introduction

According to Ndtv Research 377 people die every day, equivalent to a jumbo jet crashing every day in India due to Road Accidents. Out of which around 40 percent of the Road Accident happens due to drivers dozing off the wheel as a study by the Central Road Research Institute (CRRI) states. Our focus in this project is to minimize that above mentioned 40 percent Accidents by our System.Driver drowsiness is a serious hazard and major concern, which is identified as a direct or contributing cause in most of the road accidents. Since drowsiness can seriously slow down the reaction time and subsequently decreases drivers awareness and judgment. The development of a driver monitoring system capable of producing warning to the driver upon detecting signs of drowsiness can prevent road accidents and thus save lives. From another prospective, image processing gained popularity in computer science engineering, selected fields which has impacted in multi dimensional way. If image processing technique are used for drowsiness detection, it can simultaneously reduce road accidents promise scheme which detect driver drowsiness with help of image processing such as eye blink count. In this study, we present a low cost and fully automatic solution for handling the drowsy driver detection problem. Our system uses a standard webcam and detects the pattern of long duration eye-lid closures. The eye blink duration is the time spent while upper and lower eye-lids are connected. The pattern indicates a potential drowsiness prior to the driver falling asleep and then alerts the driver by alarm.

1.1 Purpose

Our purpose of this project is to implement the system by capturing the live images (video) of the eyes and fed them in our system which will then process the video and convert it into frames and process it accordingly.On the whole, by sensing the eye blinks and it's duration we can decide whether the driver is drowsy or not and can sound alarm accordingly.Our proposed method detects visual changes in eye locations using the proposed horizontal symme- try feature of the eyes. Our new method detects eye blinks via a standard webcam in real-time.

1.2 Project Scope

Driver drowsiness pose a major threat to highway safety, and the problem is particularly severe for commercial motor vehicle operators. Twenty-four hour operations, high annual mileage, exposure to challenging environmental conditions, and demanding work schedules all contribute to this serious safety issue. Monitoring the driver's state of drowsiness and vigilance and providing feedback on their condition so that they can take appropriate action is one crucial step in a series of preventive measures necessary to address this problem. Currently there is not adjustment in zoom or direction of the camera during operation. Future work may be to automatically zoom in on the eyes once they are localized.

1.3 Project Objectives

a. The objective of this project is to build System which will be combination of Hardware Software .

b. This System will detect the doziness of driver by camera and trigger an Alarm System inside the vehicle .

c. This Alarm System will wake up the driver and minimize the chances of Accident.

Chapter 2

Literature Survey

2.1 Paper Title 1

Noninvasive biological sensor system for detection of drunk driving.

Discription: The Proposed system is capable of monitoring the biological condition of a driver and issuing warnings during instances of drowsiness have recently been studied. Moreover, many researchers have reported that biological signals, such as brain waves, pulsation waves, and heart rate, are different between people who have and have not consumed alcohol. Currently, in this paper they are developing a noninvasive system to detect individuals driving under the influence of alcohol by measuring biological signals.

2.1.1 Advantages of Paper

a. Successfully distinguishes between normal and intoxicating state of a person.

2.1.2 Disadvantages of Paper

a. High installation Cost for constructing a seat incorporating an air-pack sensors.

2.1.3 How to overcome the problems mentioned in Paper

a. We will directly detect eyes from Camera thus reducing Cost.

b. OpenCV library will overcome the problem of low accuracy.

2.2 Paper Title 2

Visual Analysis of Eye State and Head Pose for Driver Alertness Monitoring

Discription: This paper presents visual analysis of eye state and head pose (HP) for continuous monitoring of alertness of a vehicle driver. Most existing approaches to visual detection of nonalert driving patterns rely either on eye closure or head nodding angles to determine the driver drowsiness or distraction level. The proposed scheme uses visual features such as eye index (EI), pupil activity (PA), and HP to extract critical information on nonalertness of a vehicle driver.Experimental results show that the proposed scheme offers high classification accuracy with acceptably low errors and false alarms for people of various ethnicity and gender in real road driving conditions.

2.2.1 Advantages of Paper

a. It uses visual features such as eye index (EI), pupil activity (PA), and HP to extract critical information on nonalertness of a vehicle driver.

b. Cost Efficient

2.2.2 Disadvantages of Paper

a. If the pupil is red then it fails to detect the eye of the driver

b. Low Accuracy

2.2.3 How to overcome the problems mentioned in Paper

a. We will Implement TensorFlow Models for enhancing Accuracy and Time

2.3 Paper Title 3

Design and Implementation of a Driving Assistance System in a Car-like Robot When Fatigue in the User is Detected

Discription: In this article it is presented a driving assistance system when drowsiness is detected in the driver, the system is tested by a car like robot that is wirelessly controlled by a computer in which it is connected a joystick that consists in a steering wheel and pedals, and a computational interface developed in Visual Studio 2010 which emulates an automobile panel, with these elements the user is able to drive the robot. Through an artificial vision system the driver's head orientation is monitored for determining if he/she is in a drowsiness state; if so, the robot control turn into automatic and the robot pull over to the right side of the way.

2.3.1 Advantages of Paper

a. Drowsiness is detected using Driver's Head Orientation through Atrificial vision system

2.3.2 Disadvantages of Paper

a. Low Accuracy

b. Slow Processing

2.3.3 How to overcome the problems mentioned in Paper

a. We will Implement TensorFlow Models for enhancing Accuracy and Time

2.4 Technical Review

2.4.1 Advantages of Technology

a. The main benefit of OpenCV is its vast access to algorithms, extensive use, and algorithmic efficiency. OpenCV provides algorithmic efficiency mainly to process real-time programs. Moreover, it has been designed in a way that allows it to take advantage of hardware acceleration and multi-core systems to deploy

b. HOG is reminiscent of edge orientation histogram, SIFT descriptor and shape context. They are computed on a dense grid of cells that overlap local contrast histogram normalization's of image gradient orientations to improve the detector performance. So that, this feature set performs very well for other shape based object classes (i.e. face detection) because of the distribution of local intensity gradients, even not precising any knowledge of the corresponding gradient .

2.4.2 Reasons to use this Technology

a. OpenCV is the huge open-source library for the computer vision, machine learning, and image processing and now it plays a major role in real-time operation which is very important in today's systems. By using it, one can process images and videos to identify objects, faces, or even handwriting of a human. In our System we have used OpenCV to process video frames.

b. Histograms of Oriented Gradients are generally used in computer vision, pattern recognition and image processing to detect and recognize visual objects (i.e. faces). We propose to use HOG descriptors because we need a robust feature set to discriminate and find faces under difficult illumination backgrounds, wide range of poses, etc, by using feature sets that overcome the existing ones for face detection. It's also one of the fastest algorithm in detecting faces.

Chapter 3

Project Planning

3.1 Members and Capabilities

Table 3.1: Table of Capabilities

SR. No	Name of Member	Capabilities
1	Ansari Mohd Aamir	Programming,Hardware
2	Nathani Aamir	Hardware,Report Making
3	Khan Nazim	Literature Review,Diagrams
4	Ansari Faisal	Literature Review

Work Breakdown Structure

a. All of the members are equally important in developing the project.

b. We work on a different part of the project based on one's capability

c. Firstly we came up with documentation, And based on the documentation we set our goal and created a blueprint.

d. We then started going hands-on with the project to develop it according to the flow as decided earlier.

3.2 Roles and Responsibilities

Table 3.2: Table of Responsibilities

SR. No	Name of Member	Role	Responsibilities
1	Ansari Mohd Aamir	Team Leader	Coding and PPT
2	Nathani Aamir	System Embedding	Hardware Work and Report
3	Khan Nazim	Project Managing	Diagram And Publication
4	Ansari Faisal	Project Study	Literature Review

3.3 Assumptions and Constraints

a. User of this system are persons who need to drive long especially at night.

b. User of this app should have hearing power.

c. System detects drowsiness in Realtime.

d. Alarm triggers for 5 secs when drowsiness detected.

3.4 Project Management Approach

a. Planning of project

b. Defining the scope of the project.

c. Estimation of time and It's management.

d. Properly assigning tasks to members.

e. Reporting the progress of project with the guide.

3.5 Ground Rules for the Project

a. Properly planning and gathering relevant information is very important.

b. Developing a Blueprint of the project and work accordingly

c. All the members should report to the guide whenever required.

d. Setting up small goals every week.

e. Achieving the small goal within that span of time.

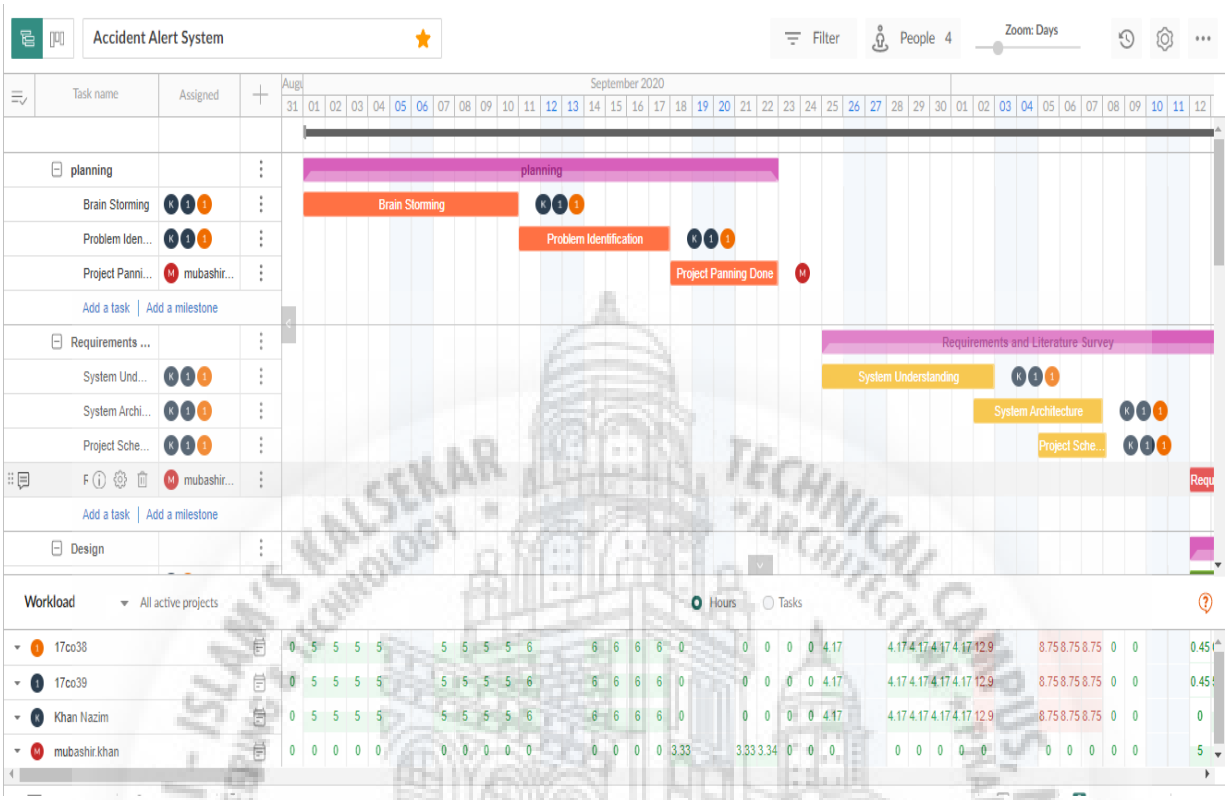
f. Keeping tracks of the progress towards project.

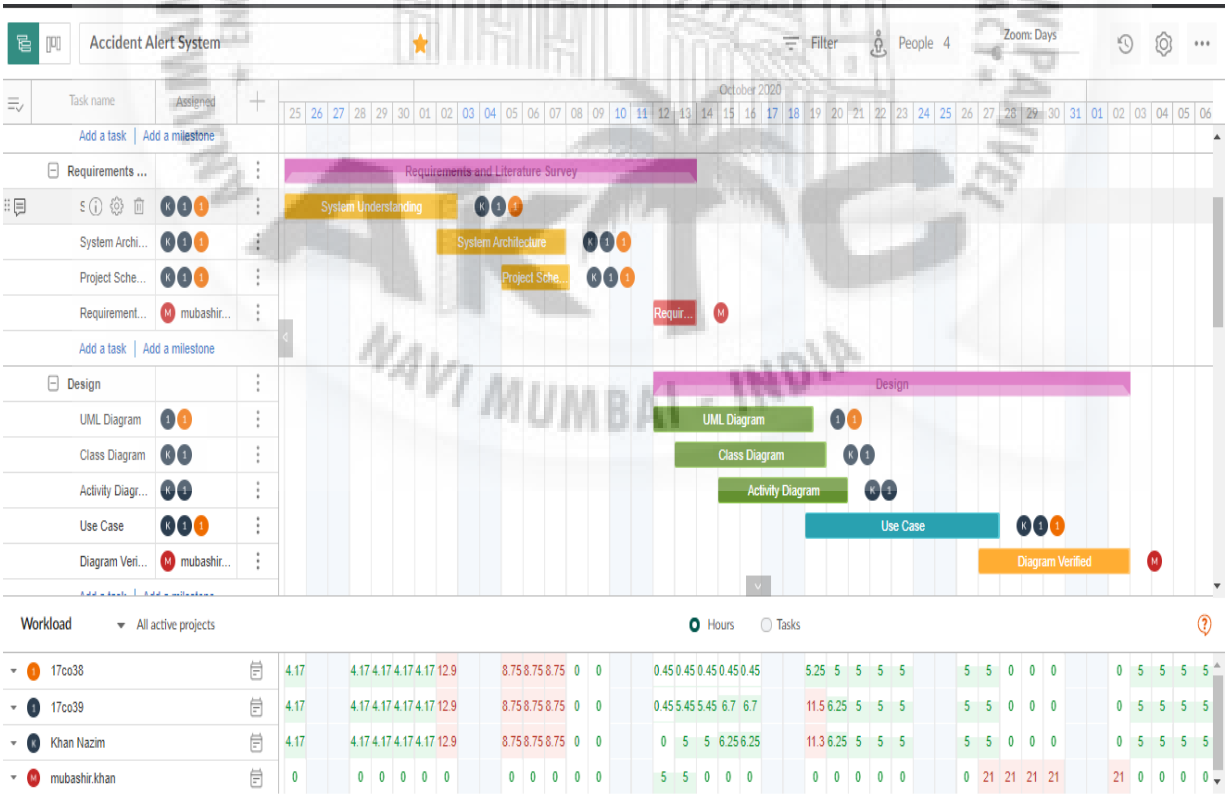
3.6 Project Budget

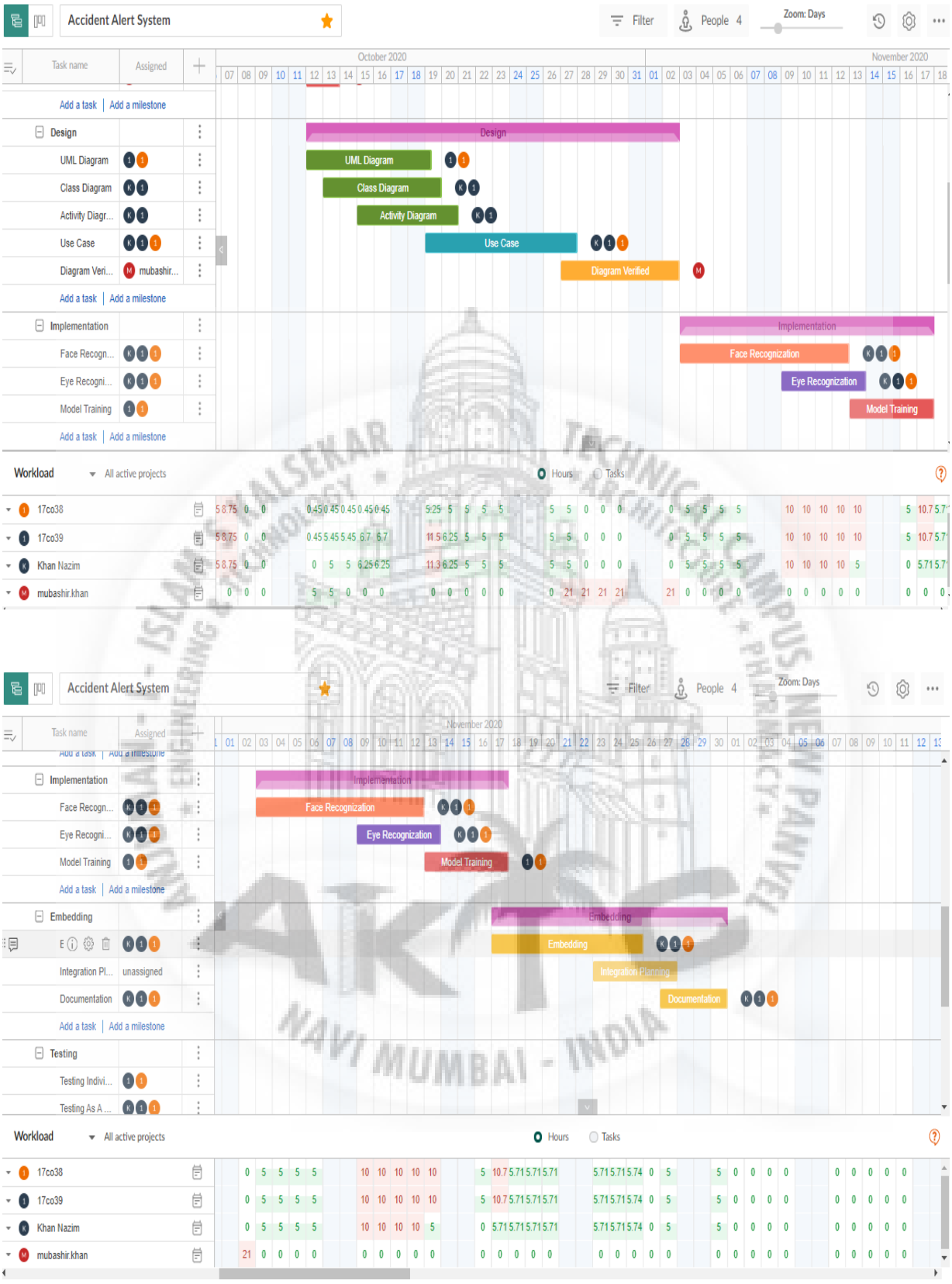
a. It's a light project with very less hardware requirements.

b. Cost of the project includes raspberry pi, camera, memory card and sound speaker.

3.7 Project Timeline






Accident Alert System
Filter
People 4
Zoom: Days
Task name
Assigned
October 2020
November 2020
Add a task | Add a milestone
Design
UML Diagram
Class Diagram
Activity Diagram
Use Case
Diagram Verified
Implementation
Face Recognization
Eye Recognization
Model Training
Embedding
Integration Planning
Documentation
Testing
Workload
All active projects
Hours
Tasks
17co38
17co39
Khan Nazim
mubashir.khan

Chapter 4

Software Requirements Specification

4.1 Description

Our System receives an input from a color video camera attached in front of the driver and processes the grabbed frames for the drowsiness detection. The detector system is composed of a video camera and software that regularly checks the eye of the driver to detect the eye blink duration. We start with the detection of the face using histogram of oriented gradients (HOG) based face detector (HOG-based) available in the dlib's library. Then we determine the facial landmark for the face region using dlib's facial landmark predictor. Extract the left and right eye coordinates from facial landmark to compute the eye aspect ratio (EAR). Eye aspect ratio is the average of euclidean distance of three pairs of distant eye co-ordinates (shown in figure 2). Once EAR is calculated it is compared with threshhold EAR which is predetermined, if it's less than threshhold counter is incremented. Once the counter reaches at maximum count which means the number of consecutive frames which consists of EAR greater than threshhold alarm is triggered.

4.2 Software Interfaces

4.2.1 Language

Python3 Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

4.2.2 Libraries

a. OpenCV: OpenCV is a huge open-source library for computer vision, machine learning, and image processing. OpenCV supports a wide variety of programming languages like Python, C++, Java, etc. It can process images and videos to identify objects, faces, or even the handwriting of a human. When it is integrated

with various libraries, such as Numpy which is a highly optimized library for numerical operations, then the number of weapons increases in your Arsenal i.e whatever operations one can do in Numpy can be combined with OpenCV. This OpenCV tutorial will help you learn the Image-processing from Basics to Advance, like operations on Images, Videos using a huge set of Opencv-programs and projects.

b. numpy: Numpy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays. It is the fundamental package for scientific computing with Python. Besides its obvious scientific uses, Numpy can also be used as an efficient multi-dimensional container of generic data.

c. Dlib: Dlib is a modern C++ toolkit containing machine learning algorithms and tools for creating complex software in C++ to solve real world problems. It is used in both industry and academia in a wide range of domains including robotics, embedded devices, mobile phones, and large high performance computing environments. Dlib's open source licensing allows you to use it in any application, free of charge.

d. Scipy: The SciPy library is one of the core packages that make up the SciPy stack. It provides many user-friendly and efficient numerical routines, such as routines for numerical integration, interpolation, optimization, linear algebra, and statistics.

e. pydub: pydub is a Python library to work with only . wav files. By using this library we can play, split, merge, edit our . wav audio files

4.2.3 Operating System

Raspian Raspbian is a free operating system based on Debian optimized for the Raspberry Pi hardware. An operating system is the set of basic programs and utilities that make your Raspberry Pi run. However, Raspbian provides more than a pure OS: it comes with over 35,000 packages, pre-compiled software bundled in a nice format for easy installation on your Raspberry Pi.

4.3 Hardware Interfaces

The proposed system is made up of the following primary components.

4.3.1 Raspberry Pi 4 Model B

The RPi 4 is the all new product in the Raspberry Pi series. It boasts ground-breaking processing and performance speed and a huge increase in memory and connectivity compared to the previous Raspberry Pi 3 Model B+. This product's key features includes a high-performance 64-bit quad-core processor, dual-display support at resolutions up to 4K via a pair of micro-HDMI ports, hardware video decode at up to 4Kp60, up to 4GB of RAM, dual-band 2.4/5.0 GHz wireless LAN, Bluetooth 5.0, Gigabit Ethernet, USB 3.0, and PoE capability. he 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 playing games.
In our system it performs a processing of the input video stream so to compute the level of fatigue of the driver. The analysis is based on calculating a number of frames of the Data stream where the driver eyes are closed. Video segments whose average eye state point exceeds the threshold value are detected as drowsy.

4.3.2 Speakers

It is an audio- or voice-generating device which can convert electromagnetic waves into sound. If the driver's drowsiness is detected, then a voltage is supplied as an alert to generate regular programmed voice sound

4.3.3 Memory Card

Memory card for the storage of Operating system for the Raspberry pi Model B and for the data and other storage.

4.3.4 Webcam

A camera is an optical instrument used to capture an image and video It is used for capturing the video in real time and monitors a stream for faces. In order to effectively capture the face, the webcam is placed onto the vehicle dashboard and is approximately 20cm away from the driver's face. At this distance, the webcam captures the most of the driver's face. The captured video is sent to the Raspberry Pi for further processing

4.4 Nonfunctional Requirements

4.4.1 Performance Requirements

a. Performance of overall system is very efficient and well optimize.

b. Right from capturing video frames till drowsiness detection everything happens in real time.

c. It takes 3-5 seconds for preprocessing at start.

4.4.2 Safety Requirements

a. Our system does not store any sort of data in any form.

b. All processing happens in real time.

c. In case of any updates in libraries used can lead to the failure in systems.

4.4.3 Security Requirements

All the libraries used are certified and standard.Also camera access is until the process is done completely.After that is released.

Chapter 5

System Design

5.1 System Requirements Definition

System requirement definitions specify [1] what the system should do, its functionality and its essential and desirable system properties. The techniques applied to elicit and collect information in order to create system specifications and requirement definitions involve consultations, interviews, requirements workshop with customers and end users. The objective of the requirements definition phase is to derive the two types of requirement:

5.1.1 Functional requirements

They define the basic functions that the system must provide and focus on the needs and goals of the end users.

5.1.2 System requirements (non-functional requirements)

These are non-functional system properties such as availability, performance and safety etc. They define functions of a system, services and operational constraints in detail.

a. Usability - System implementation is feasible using technologies that are accessible to the end-users.

b. Portability - The interfaces are compatible with every vehicle.

c. Performance Efficiency -System is able to perform well in a proper time constraint.

d. Time Efficiency - Time taken for the executing of system is less.

5.2 Algorithm

The different steps involved in algorithm for Raspberry-pi model are as follow.

1. Image capture-To read an image from the Raspberry pi camera; we first open the camera inside the Open CV using the function.
2. The next step is to apply facial landmark detection to localize each of the important regions of the face.
3. For each of the detected faces, we apply dlib's facial landmark detector and convert the result to a NumPy array.
4. The facial landmarks produced by dlib are an indexable list.

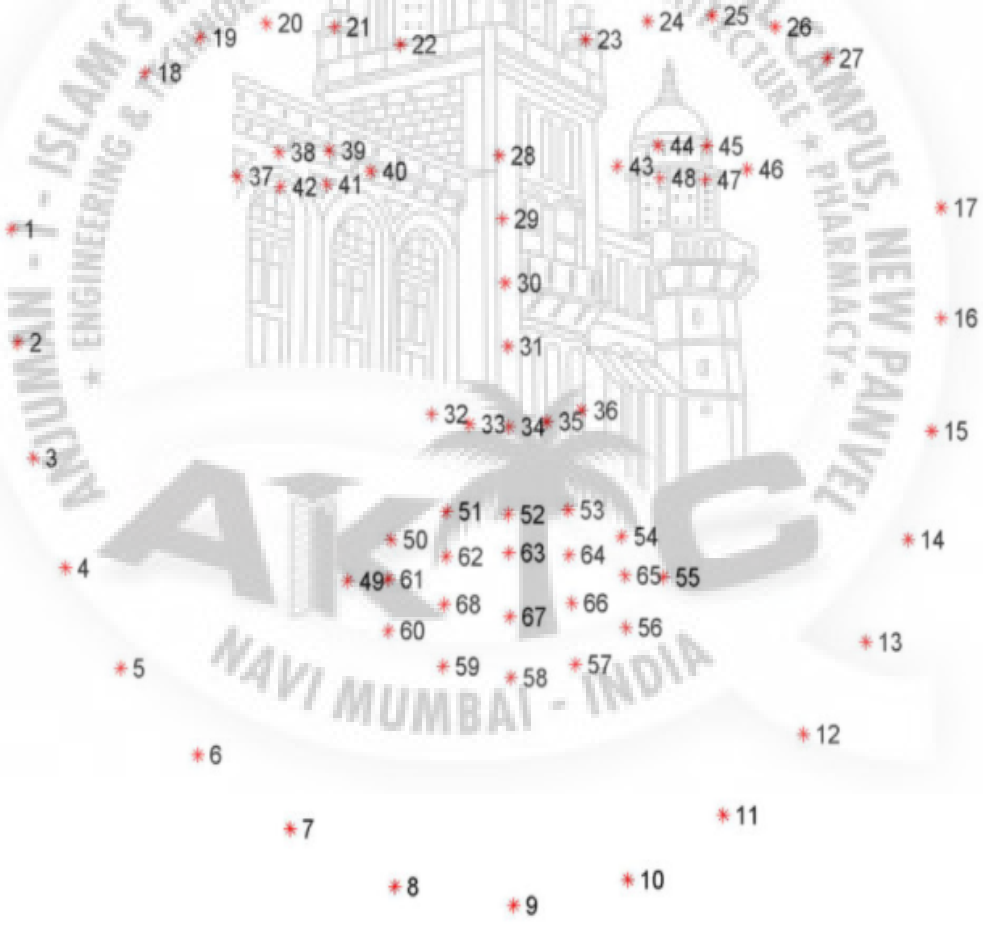


Figure 5.1: Face Landmarks

5. Using NumPy array slicing we can extract the (x, y)-coordinates of the left and right eye, respectively .

6. Given the (x, y)-coordinates for both eyes, we then compute their eye aspect ratios.
7. Define the eye aspect ratio function which is used to compute the ratio of distances between the vertical eye landmarks and the distances between the horizontal eye landmarks.
8. The return value of the eye aspect ratio will be approximately constant when the eye is open.
9. If the eye is closed, the eye aspect ratio will again remain approximately constant, but will be much smaller than the ratio when the eye is open.
10. Define the EYE AR THRESH. If the eye aspect ratio falls below this threshold, we'll start counting the number of frames the person has closed their eyes for.
11. Define CONSEC EYE AR THRESH. If the counter exceeds the above threshold ,we'll sound an alarm.

5.3 System Architecture Design

A system architecture is the conceptual model that defines the structure, behavior, and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures and behaviors of the system.

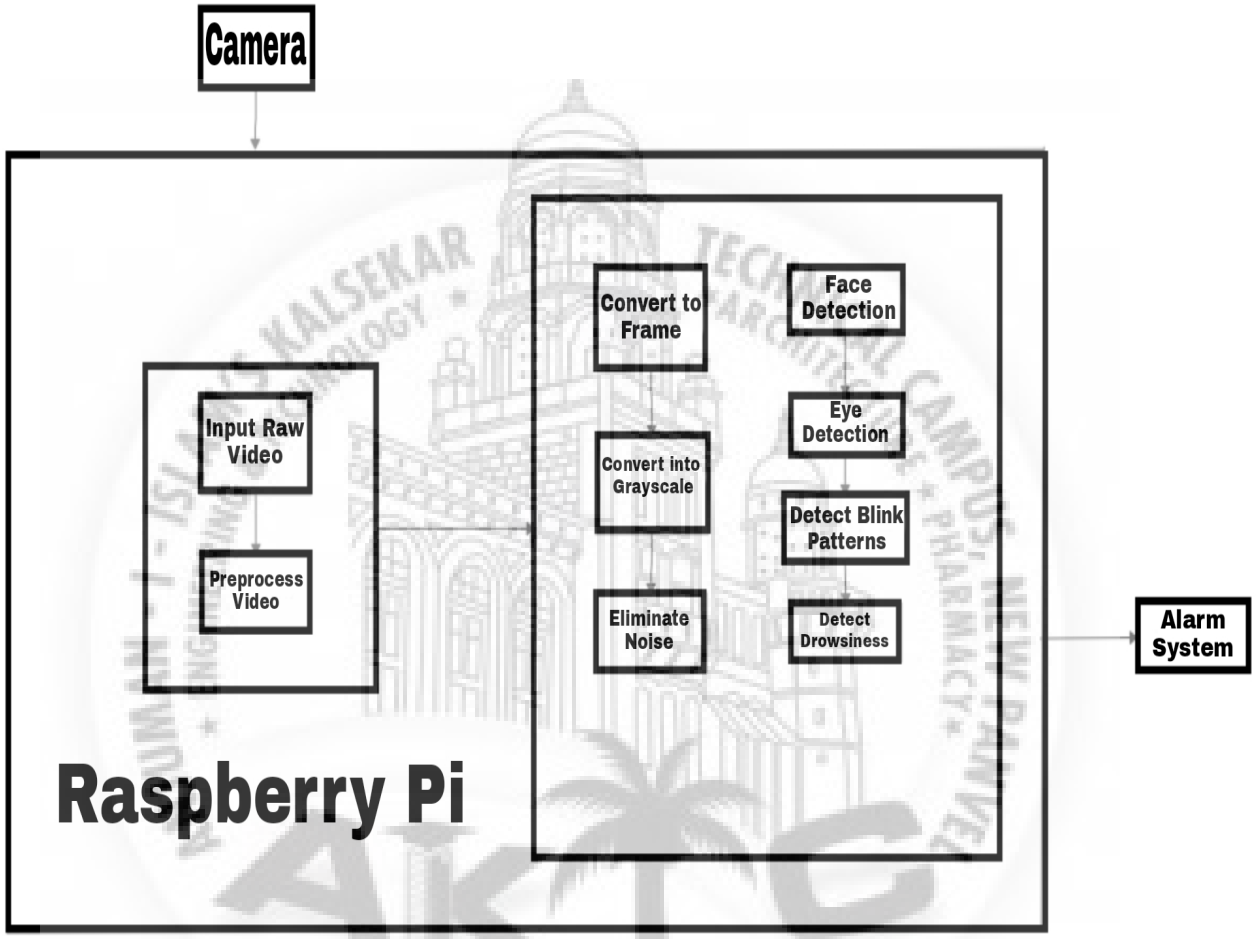


Figure 5.2: System Architecture

5.4 Design Details

5.4.1 UseCase

A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved.In our system User will interact with use cases like Capture Image,Audio Input,Save text,Retrieve Text,Audio output.

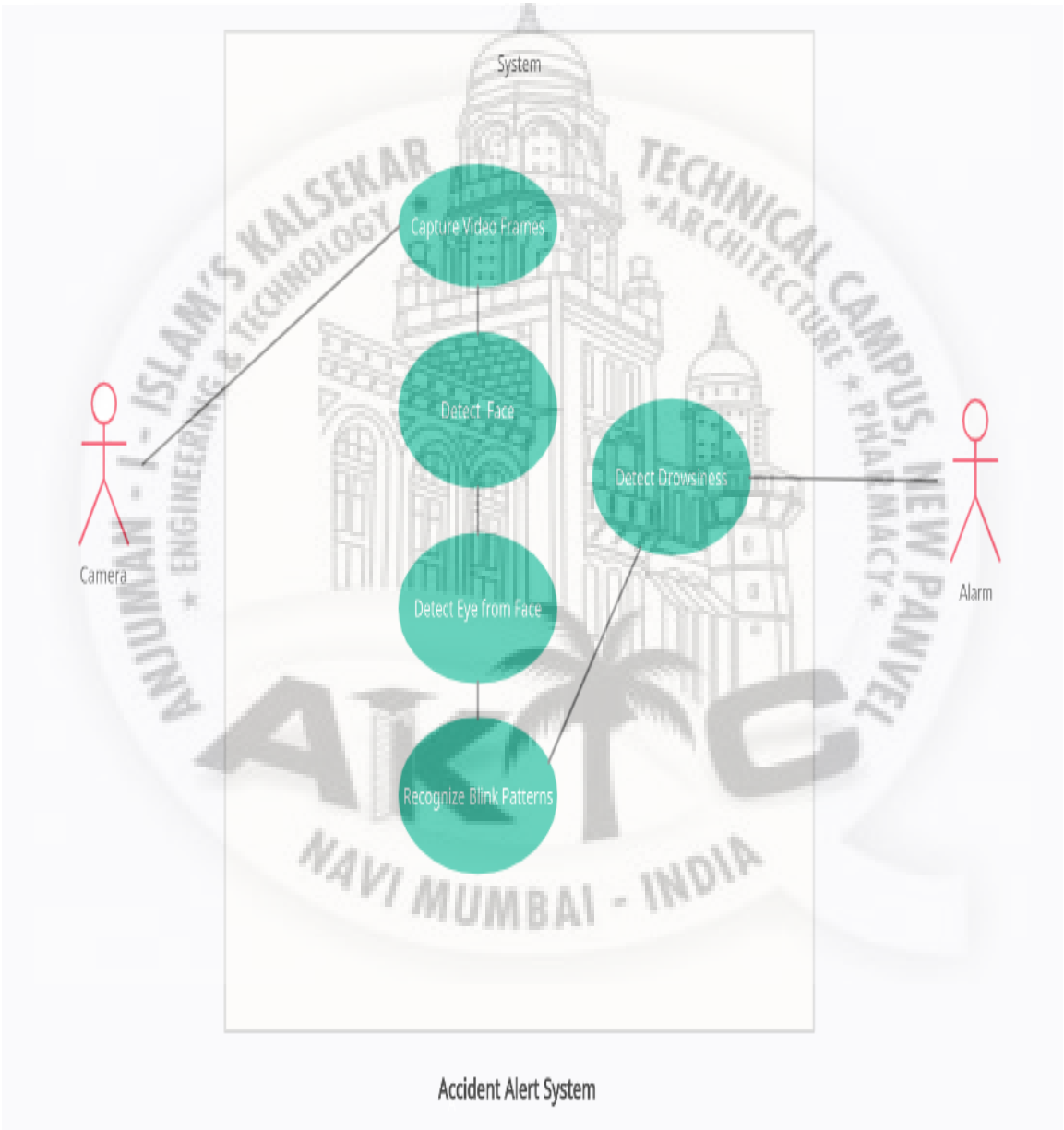


Figure 5.3: UseCase Diagram

5.4.2 Class Diagram

In software engineering, a class diagram in the Unified Modeling Language is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations, and the relationships among objects.Our System consist of three classes Camera, System and Alarm.

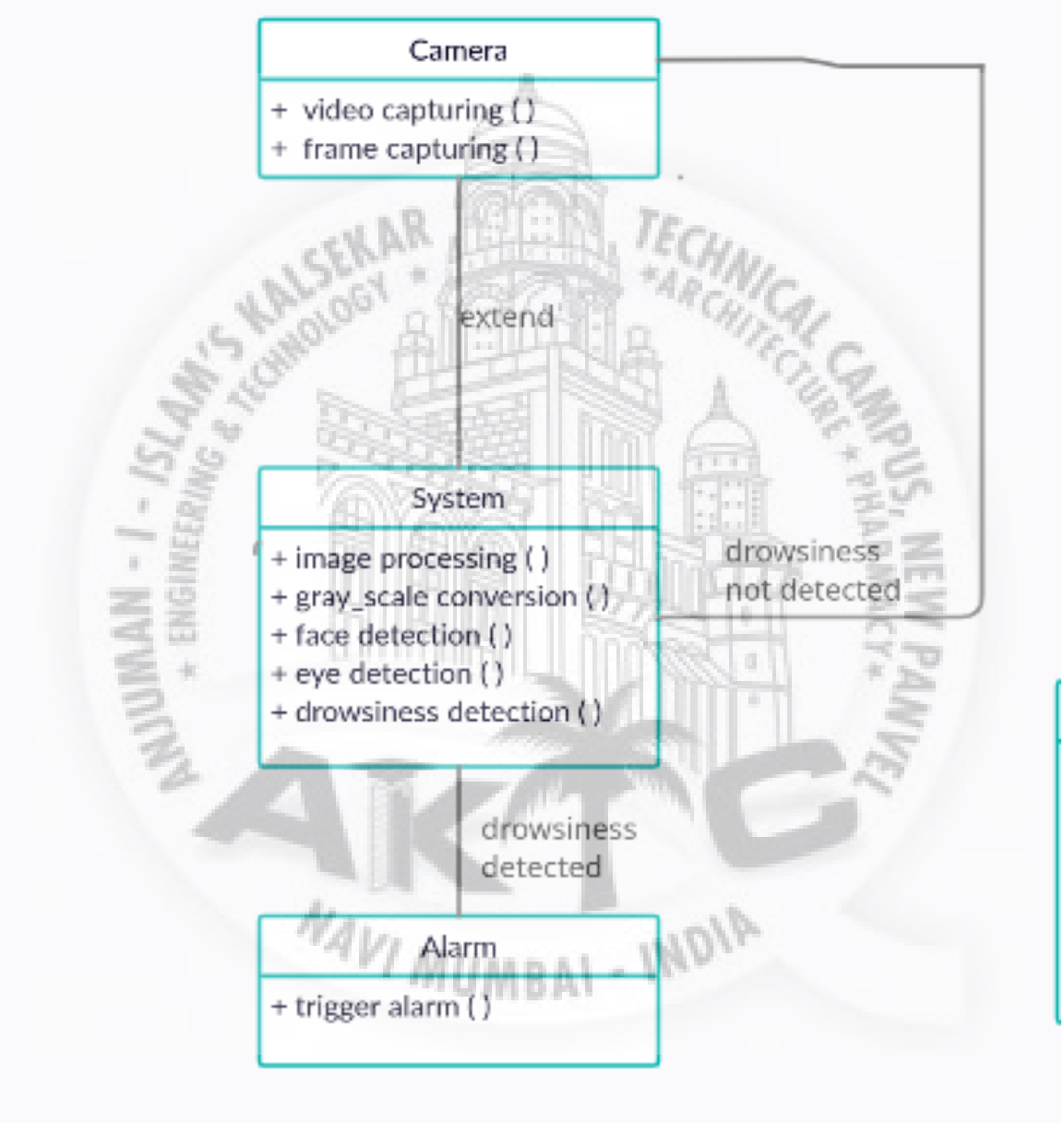


Figure 5.4: Class Diagram

5.4.3 Flow Diagram

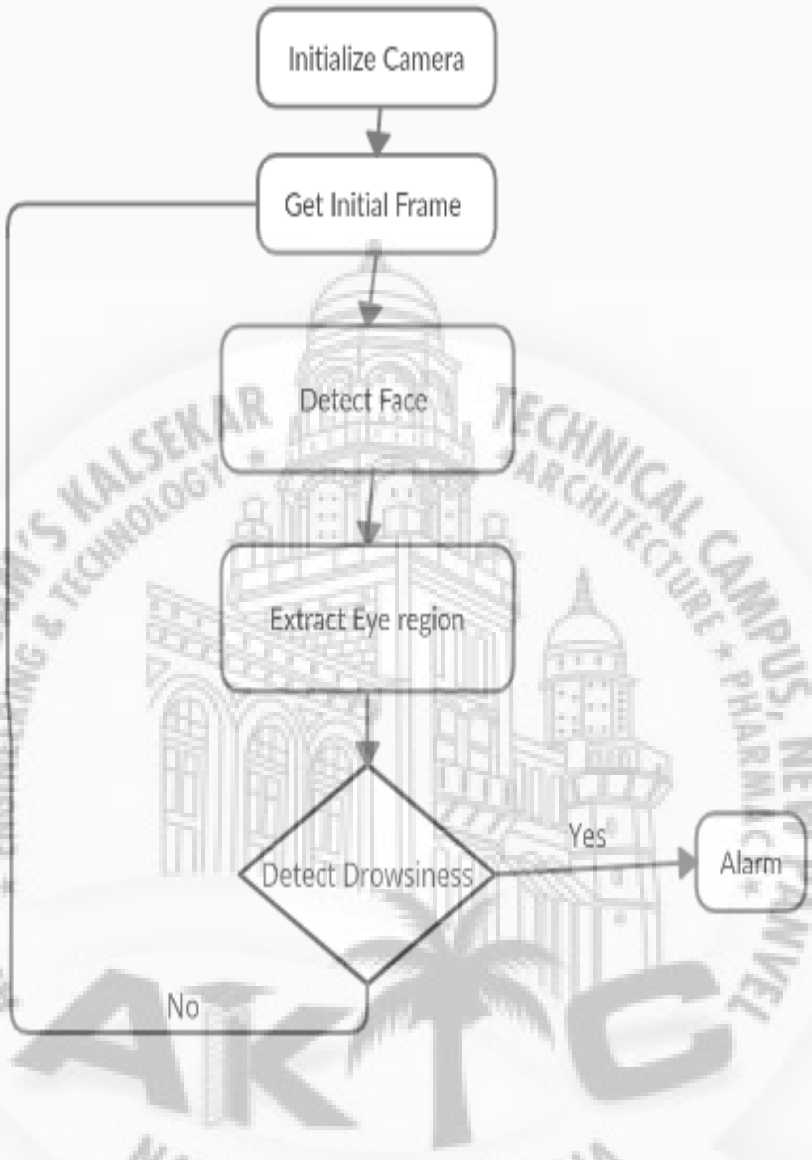


Figure 5.5: Flow Chart

Chapter 6

Implementation

6.1 Code

detectdrowsiness.py

```
#necessary imports
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
from pydub import AudioSegment
from pydub.playback import play
import argparse
import imutils
import time
import dlib
import cv2


# alarm function
def sound_alarm(path):
    print('playSound')
    song = AudioSegment.from_mp3("alarm.mp3")
    play(song)


# EAR function which calculates Eye Aspect Ratio and returns average
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



ap = argparse.ArgumentParser()
ap.add_argument("-w", "--webcam", type=int, default=0,
                help="index of webcam on system")
args = vars(ap.parse_args())
p = "shape_predictor_68_face_landmarks.dat"
a = "alarm.mp3"

```

```
42
43 #Declaration of EAR threshold and Counter threshold
44 EYE_AR_THRESH = 0.3
45 EYE_AR_CONSEC_FRAMES = 48
46
47 COUNTER = 0
48 ALARM_ON = False
49
50
51 # initialize dlib's face detector (HOG-based) and then create facial landmark
       predictor
52
53 print("[INFO] loading facial landmark predictor...")
54 detector = dlib.get_frontal_face_detector()
55 predictor = dlib.shape_predictor(p)
56
57
58 # grab indexes of the facial landmarks for the left and right eye
59 (lStart, lEnd) = face_utils.FACIAL_LANDMARKS_IDXS["left_eye"]
60 (rStart, rEnd) = face_utils.FACIAL_LANDMARKS_IDXS["right_eye"]
61
62
63 # start the video stream thread
64 print("[INFO] starting video stream thread...")
65 vs = VideoStream(src=args["webcam"]).start()
66 time.sleep(1.0)
67 # loop over frames
68 while True:
69     # grab the frame, resize it, and convert it to grayscale
70
71     frame = vs.read()
72     frame = imutils.resize(frame, width=450)
73     gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
74     # detect faces in the grayscale frame
75     rects = detector(gray, 0)
76
77     # loop over the detected faces
78     for rect in rects:
79         # determine the facial landmarks for the face region
80         shape = predictor(gray, rect)
81         shape = face_utils.shape_to_np(shape)
82         # extract the left and right eye coordinates, then use the coordinates
               to compute the eye aspect ratio for both eyes
83         leftEye = shape[lStart:lEnd]
84         rightEye = shape[rStart:rEnd]
85         leftEAR = eye_aspect_ratio(leftEye)
86         rightEAR = eye_aspect_ratio(rightEye)
87         # average EAR
88         ear = (leftEAR + rightEAR) / 2.0
89
90         # compute the convex hull for the left and right eye
91         leftEyeHull = cv2.convexHull(leftEye)
92         rightEyeHull = cv2.convexHull(rightEye)
93         cv2.drawContours(frame, [leftEyeHull], -1, (0, 255, 0), 1)
94         cv2.drawContours(frame, [rightEyeHull], -1, (0, 255, 0), 1)
95
96         # check if EAR crosses predetermined threshold
97         if ear < EYE_AR_THRESH:
98             COUNTER += 1
99         # if the eyes were closed for a sufficient number of
100            # then sound the alarm
```

```
            if COUNTER >= EYE_AR_CONSEC_FRAMES:
                # if the alarm is not on, turn it on
                if not ALARM_ON:
                    print('alarm function')
                    ALARM_ON = True



                    t = Thread(target=sound_alarm,
                               args=(a,))
                    t.deamon = True
                    t.start()
                # draw an alarm text on the frame
                cv2.putText(frame, "DROWSINESS ALERT!", (10, 30),
                            cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)
            # otherwise, set counter to 0 and alarm to false
        else:
            COUNTER = 0
            ALARM_ON = False

        # draw EAR on screen
        cv2.putText(frame, "EAR: {:.2f}".format(ear), (300, 30),
            cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)

    # show the frame
    cv2.imshow("Frame", frame)
    key = cv2.waitKey(1) & 0xFF

    # if the 'q' key was pressed, break from the loop
    if (key == ord("q")):
      break
# cleanup
cv2.destroyAllWindows()
vs.stop()
```

Chapter 7

System Testing

System testing is a level of software testing where a complete and integrated software is tested. The purpose of this test is to evaluate the system's compliance with the specified requirements.

7.1 Test Cases and Test Results

Test ID	Test Case Title	Test Condition	System Behavior	Expected Result
T01	Detecting of drowsiness	Straight face,Good Light, Open eyes	Drowsiness not detected	Non Drowsy
T02	Detecting of drowsiness	Straight face,Good Light, Closed eyes	Drowsiness detected	Drowsy

7.2 Sample of a Test Case

Title: Detect Drowsiness
Description: System should be able to successfully detect drowsiness.

Assumption: System should be able to successfully detect drowsiness.

Test Steps:

1. Start the system.
2. Video capturing activity will be started.
3. Normal Eye movement
4. Closing eye more than partial

bfExpected Result: The System will capture video frames and detect drowsiness and trigger alarm if drowsiness found.

Actual Result: Drowsiness detected successfully followed by triggering Alarm

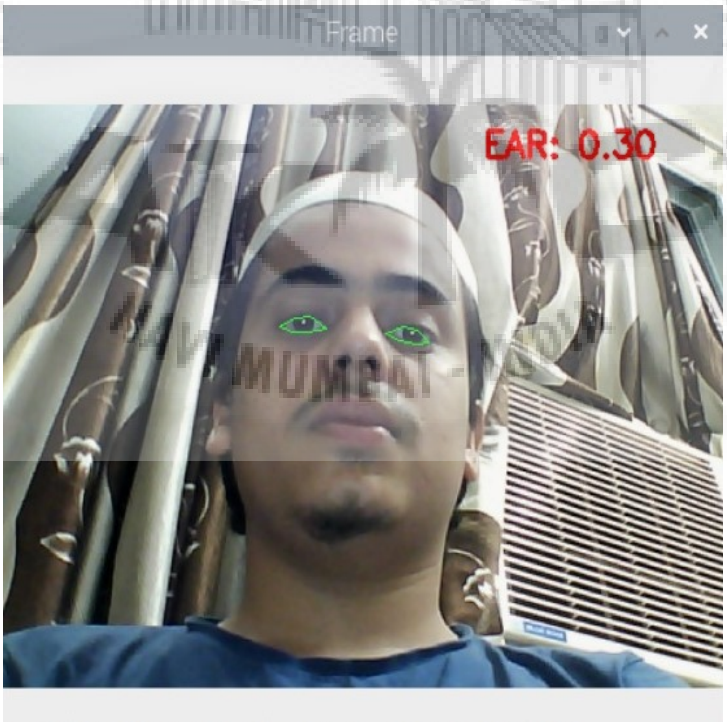


Figure 7.1: Eyes Detected from Face

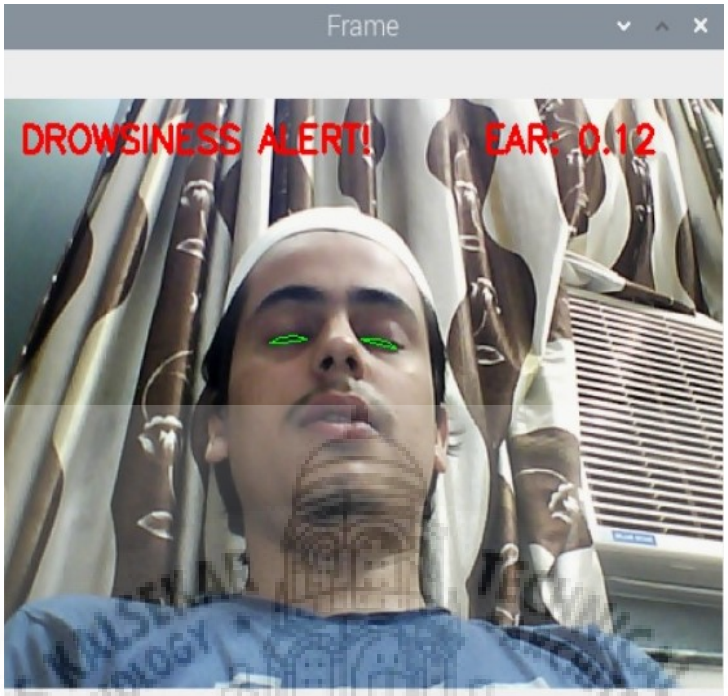


Figure 7.2: Drowsiness Detected

7.2.1 Software Quality Attributes

Availability : System should be fully functional and available to users once installed.
Efficiency-1 : The system should detect drowsiness correctly with an accuracy of more than 90 percent.
Efficiency-2 : The system should alert the user when drowsiness detected by playing Alarm ultimately preventing accident.

Chapter 8

Screenshots of Project

8.1 Hardware Things Of Project

8.1.1 Raspberry Pi

Raspberry pi is used as a processing device. The code is programmed in raspberry pi which controls every aspect of our system.

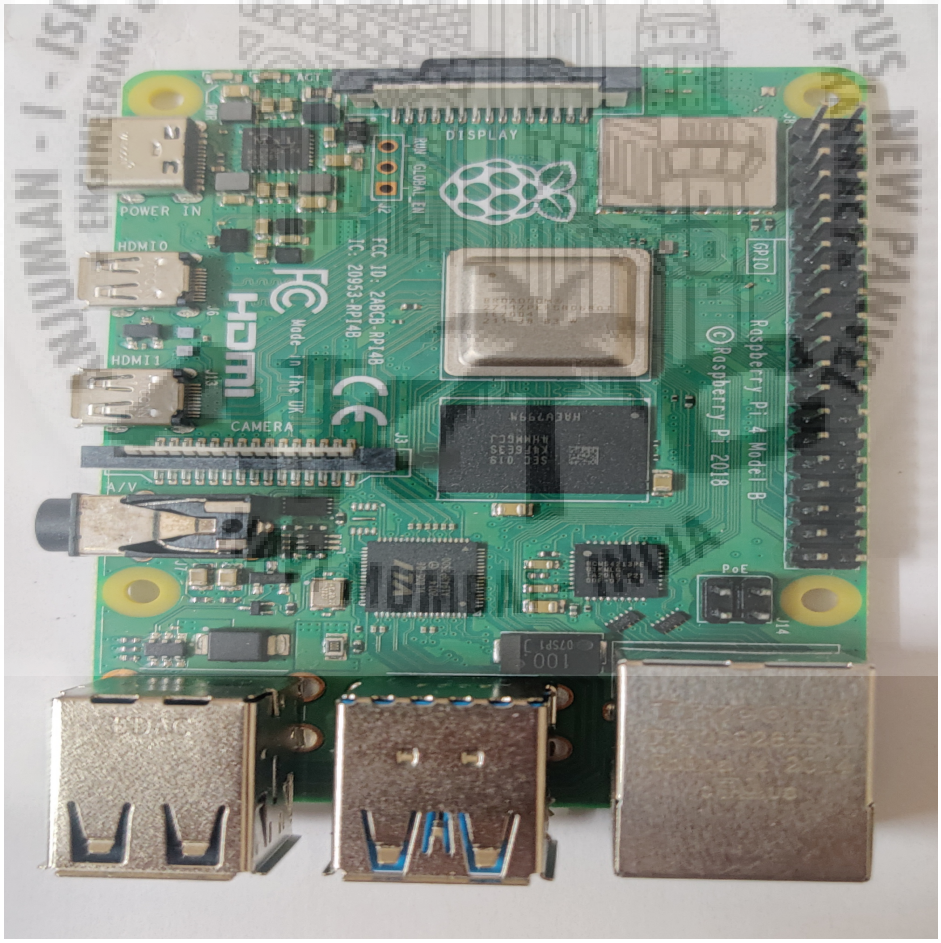


Figure 8.1: Raspberry Pi 4 B

8.1.2 Camera

Camera is used to capture live video (frames) of user for further processing.

Figure 8.2: Camera

8.1.3 Memory Card

Memory card is used for storage purposes. It acts as a HardDrive for raspberrypi and stores OS and other data.

Figure 8.3: 16Gb Memory Card

8.1.4 Speaker

Sound system is used to alert the user by alarming buzzing sound.

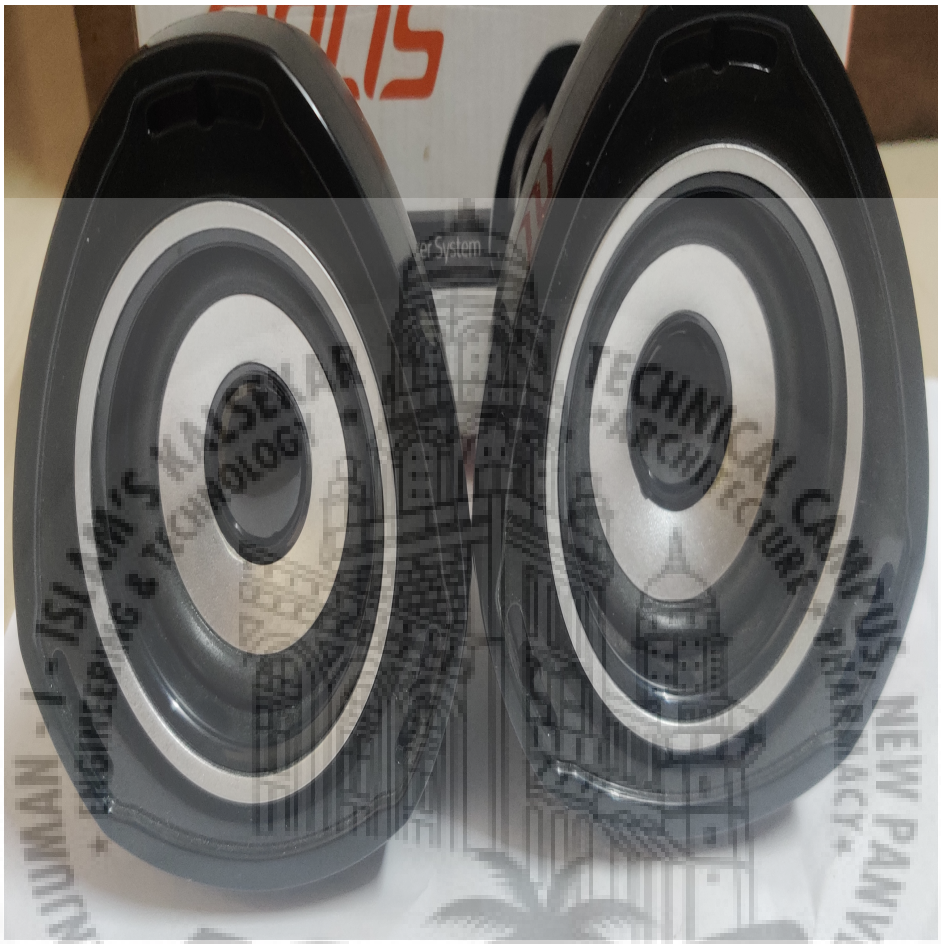


Figure 8.4: Speaker

8.2 Software Things Of Project

8.2.1 Vnc Viewer

Vnc Viewer is a desktop sharing system that allows you to remotely control another computer. In Our Project the Vnc is used to display the screen of Raspberry Pi System in our laptop as shown in figure.

Figure 8.5: Raspberry Pi Screen on Vnc Viewer

8.3 Output of The Project

8.3.1 Face Detection

Face and eye are detected and Eye Aspect Ratio is calculated and displayed on screen.

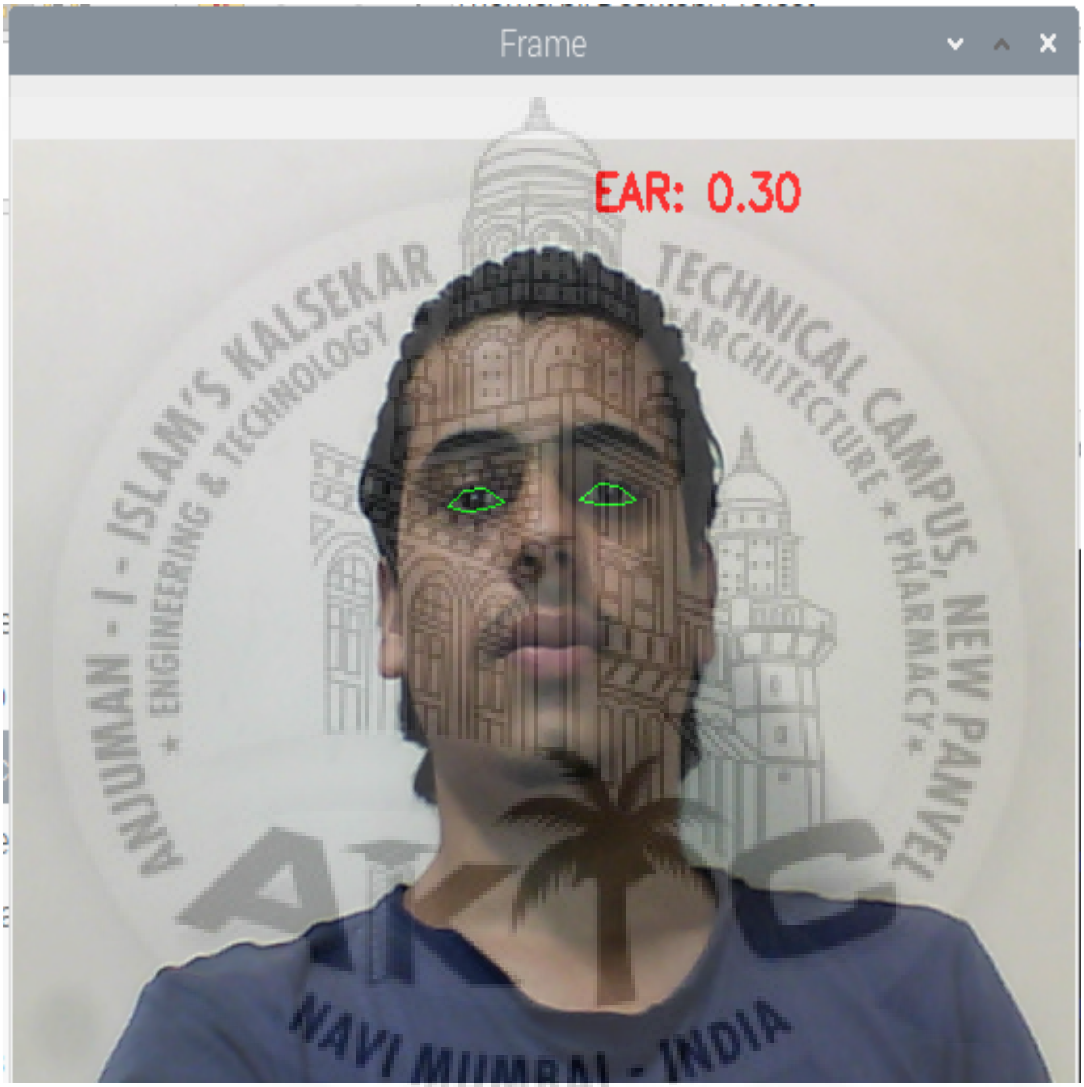


Figure 8.6: Face And Eyes Detected

8.3.2 Drowsiness Detection

Drowsiness is detected, displayed and system alerts the driver by making sound.

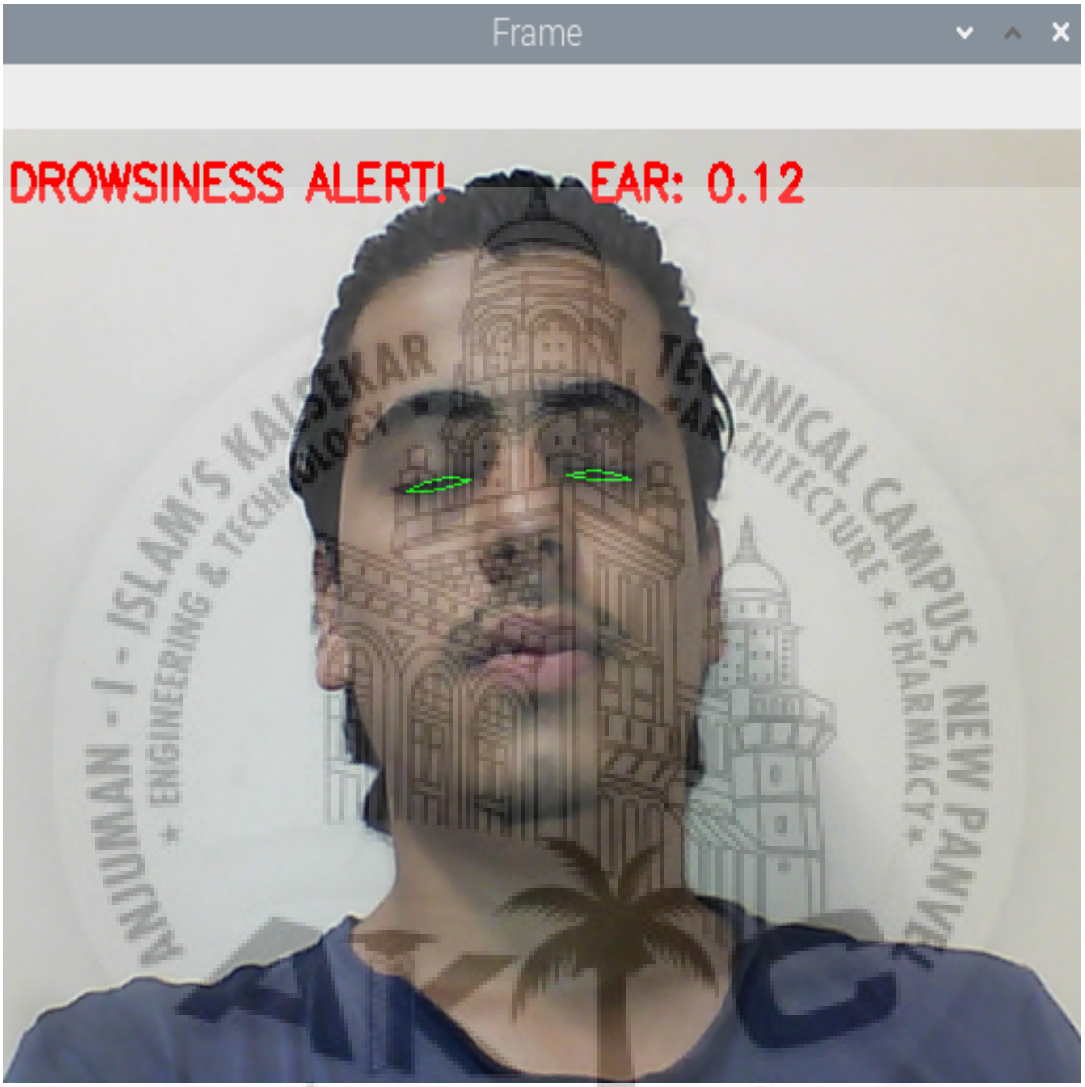


Figure 8.7: Drowsiness Detected

8.3.3 Face Detection in Specs

Sucessfully detects Face, eye and calculates Eye Aspect Ratio even while using Specs.

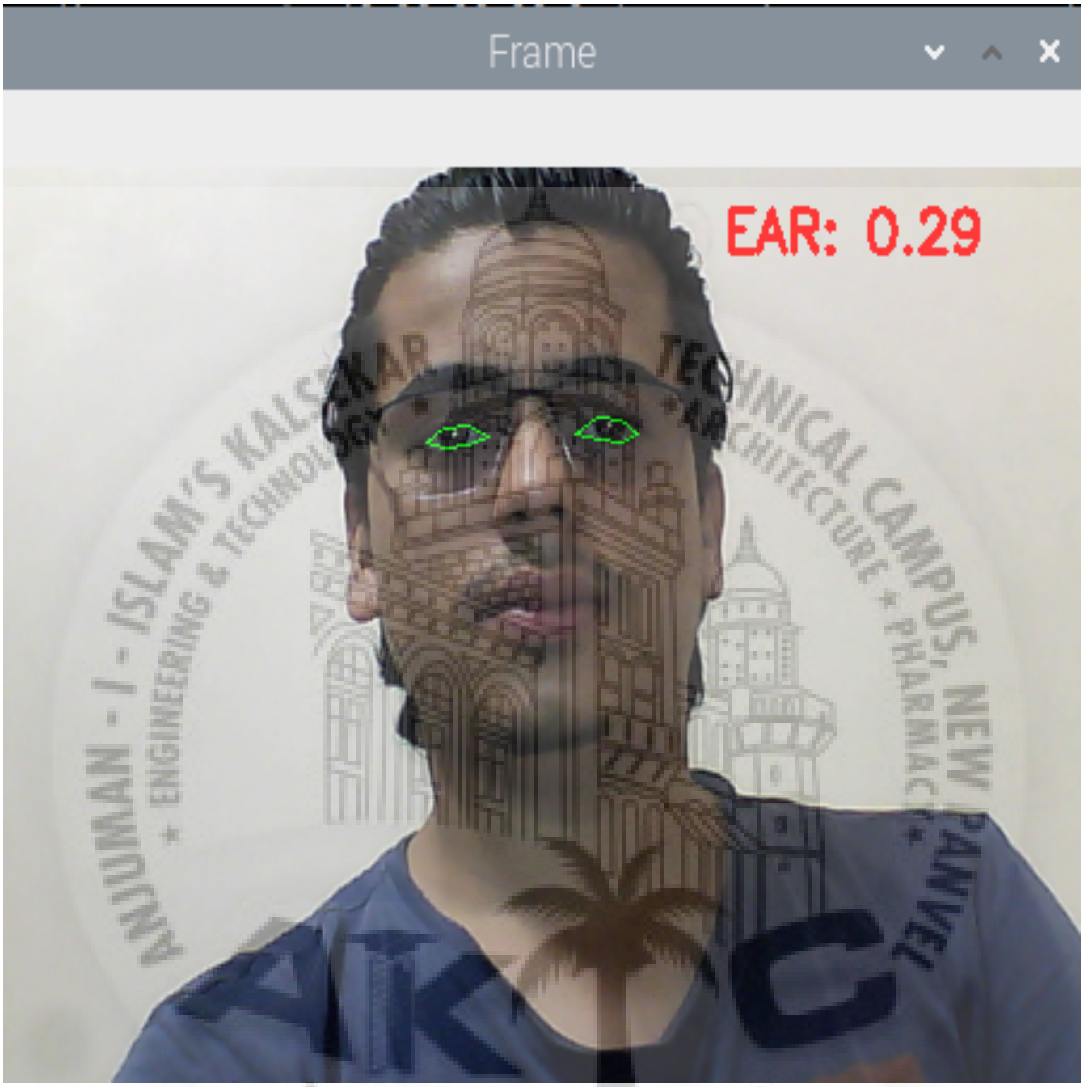


Figure 8.8: Face And Eyes Detected in Specs

8.3.4 Drowsiness Detection in Specs

Drowsiness is detected, displayed and system alerts the driver by making sound even while using specs.

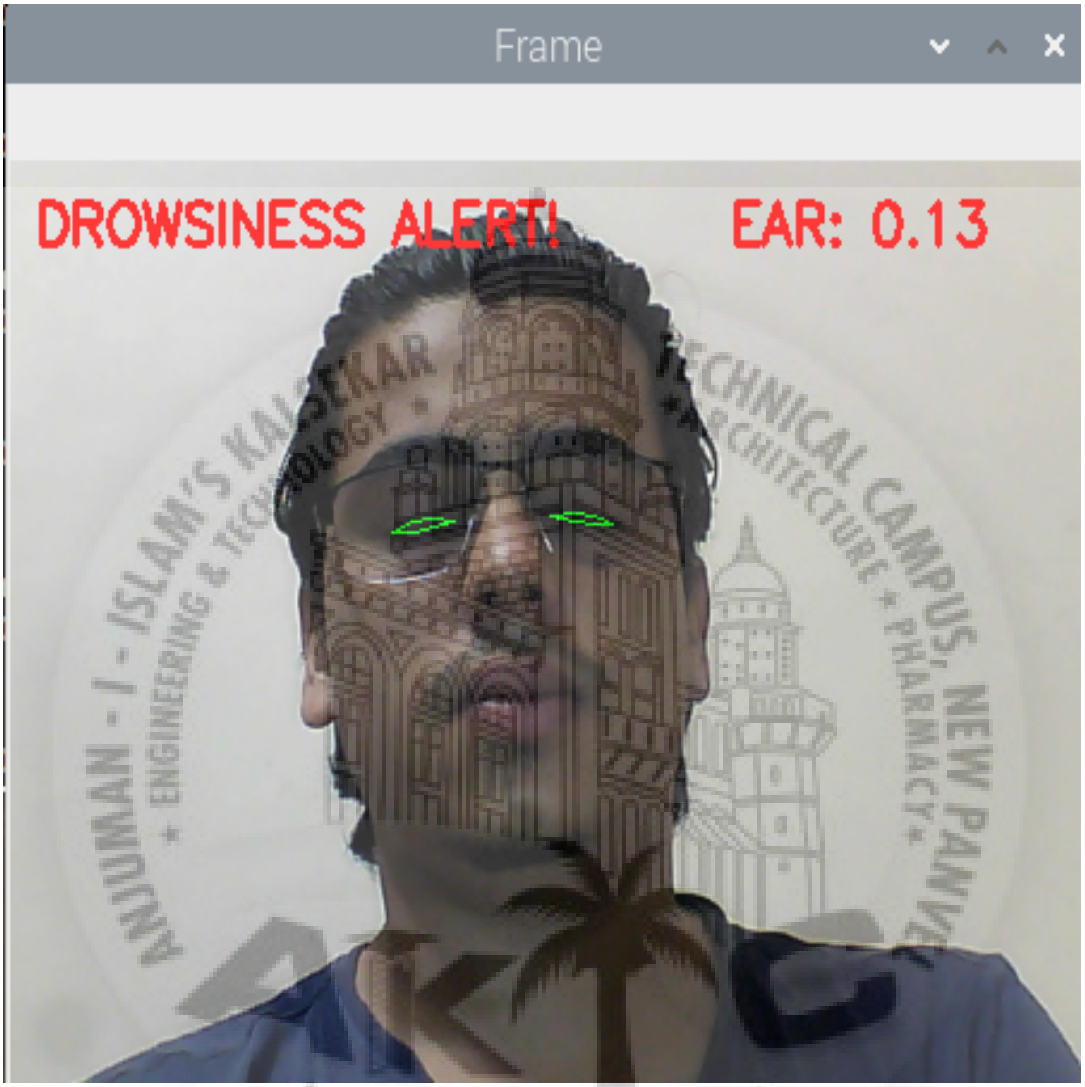


Figure 8.9: Drowsiness Detected in Specs

Chapter 9

Conclusion and Future Scope

9.1 Conclusion

This project introduces a drowsiness detection framework based on Dlib's HOG algorithm,that detects the eyes, and also counts the eye blink rate followed by drowsiness detection at real time.In the proposed system, the details about the eye status is obtained through image processing algorithms, which offer a non-invasive approach to detect drowsiness without any annoyance and interference.

9.2 Future Scope

- The model can be improved incrementally by using other parameters like blink rate, yawning, state of the car, etc.
- The system can be improved by adding mailing system.
- Currently there is not adjustment in zoom or direction of the camera during operation. Future work may be to automatically zoom in on the eyes once they are localized.
- Besides that, future works include studying different states to measure the driver attentiveness.

References

[1] *Noninvasive biological sensor system for detection of drunk driving in IEEE Transactions on information technology*; K Murata, E Fujita, S Kojima, S Maeda, Y Ogura, T Kamei, T Tsuji, S kaneko, M yoshizumi, and N Suzuki,journal Name, 2011.

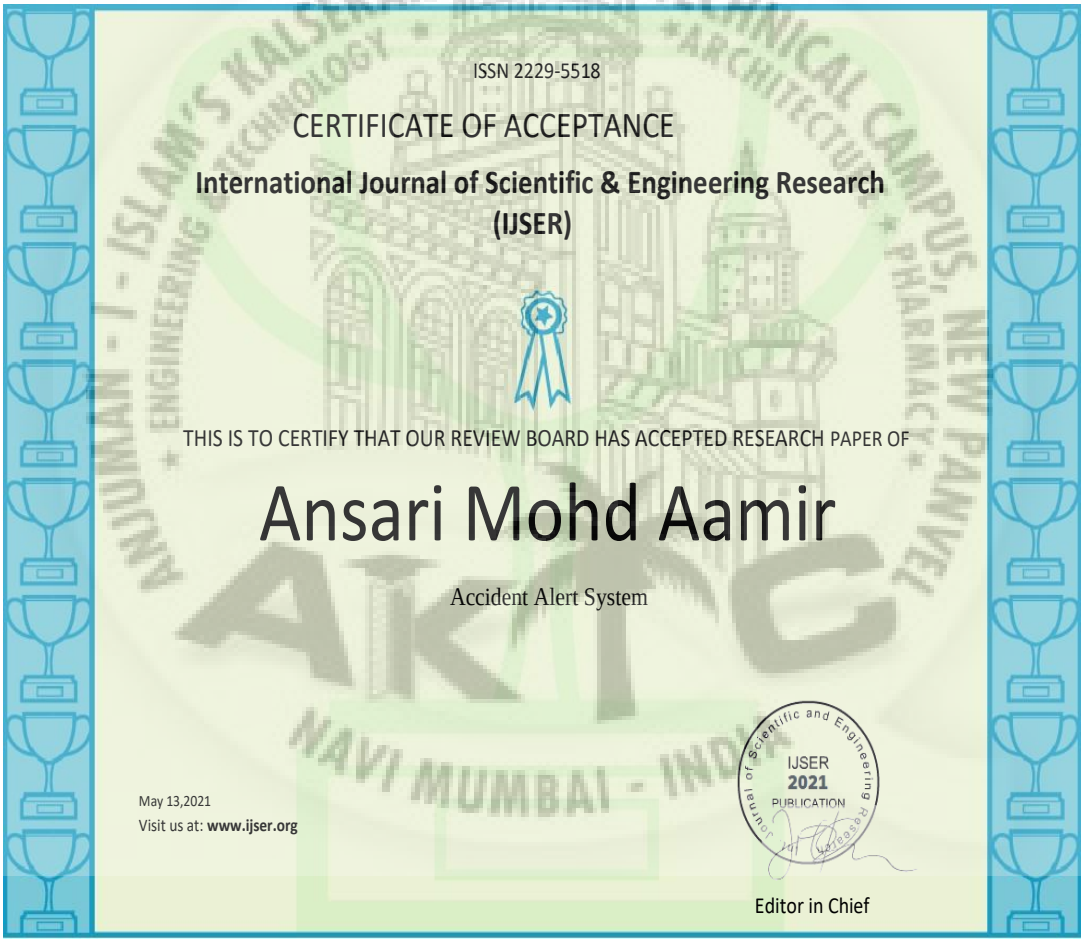
[2] *Design and Implementation of a Driving Assistance System in a Car-like Robot When Fatigue in the User is Detected*; J. Pilataxi, W. Vian and D. Chavez,2016.

[3] *Visual Analysis of Eye State and Head Pose for Driver Alertness Monitoring*; R Oyini Mbouna, Seong G. Kong, and M Chun, 2013.

[4] *Real-Time Driver Drowsiness Detection System Based on Visual Information*; Kunika Chhaganbhai Patel , Shafiullah Atiullah Khan , Vijaykumar Nandkumar Patil, 2018.

Achievements

Publication: We published our final year project on International Journal of Scientific Engineering Research(IJSER) and it is accepted. Below are the certificates of the published paper.



ISSN 2229-5518

CERTIFICATE OF ACCEPTANCE

International Journal of Scientific & Engineering Research (IJSER)

THIS IS TO CERTIFY THAT OUR REVIEW BOARD HAS ACCEPTED RESEARCH PAPER OF

Ansari Mohd Aamir

Accident Alert System

May 13,2021

Visit us at: **www.ijser.org**

IJSER 2021 PUBLICATION

Editor in Chief

ISSN 2229-5518

CERTIFICATE OF ACCEPTANCE

International Journal of Scientific & Engineering Research (IJSER)

THIS IS TO CERTIFY THAT OUR REVIEW BOARD HAS ACCEPTED RESEARCH PAPER OF

Nathani Aamir

Accident Alert System

May 13,2021
Visit us at: **www.ijser.org**

IJSER 2021 PUBLICATION

Editor in Chief

ISSN 2229-5518

CERTIFICATE OF ACCEPTANCE

International Journal of Scientific & Engineering Research (IJSER)

THIS IS TO CERTIFY THAT OUR REVIEW BOARD HAS ACCEPTED RESEARCH PAPER OF

Khan Nazim

Accident Alert System

May 13,2021
Visit us at: **www.ijser.org**

IJSER 2021 PUBLICATION

Editor in Chief

ISSN 2229-5518

CERTIFICATE OF ACCEPTANCE

International Journal of Scientific & Engineering Research (IJSER)

THIS IS TO CERTIFY THAT OUR REVIEW BOARD HAS ACCEPTED RESEARCH PAPER OF

Ansari Faisal

Accident Alert System

May 13,2021

Visit us at: **www.ijser.org**

IJSER 2021 PUBLICATION

Editor in Chief