FATIGUE SURVEILLANCE AND SHIELD

A PROJECT REPORT

Submitted by

PAVITHRA M (170701150)

THYAGARAJEN T (170701245)

in partial fulfilment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE

CHENNAI 600 025

APRIL 2021

ANNA UNIVERSITY, CHENNAI

BONAFIDE CERTIFICATE

Certified that this Report titled "**FATIGUE SURVEILLANCE AND SHIELD"** is the bonafide work of **"PAVITHRA M (170701150)** and **THYAGARAJEN T (170701245)"** who carried out the work under my supervision. Certified further that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE	**SIGNATURE**
Dr.R.G.Sakthivelan M.E., Ph.D., Professor and Head Department of Computer Science and Engineering Rajalakshmi Engineering College Thandalam-602 105	Dr.S.Prithi, M.Tech., PhD., AssistantProfessor(SG), Department of Computer Science and Engineering Rajalakshmi Engineering College Thandalam-602 105

Submitted to Project Viva-Voice Examination held on___________________

InternalExaminer External Examiner

ACKNOWLEDGEMENT

First, we thank the almighty god for the successful completion of the project. Our sincere thanks to our chairman **Mr.S.Meganathan, B.E., F.I.E.,** for his sincere endeavour in educating us in his premier institution. We would like to express our deep gratitude to our beloved Chairperson **Dr.ThangamMeganathan, Ph.D.,** for her enthusiastic motivation which inspired us a lot in completing this project and Vice Chairman **Mr.Abhay Shankar Meganathan, B.E., M.S.,** for providing us with the requisite infrastructure.

We also express our sincere gratitude to our college Principal, **Dr.S.N.Murugesan, M.E., Ph.D.,**who helped us in providing the required facilities for completing the project. We would like to thank and express our gratitude to **Dr.R.G.Sakthivelan, Ph.D.,** Head of the Department of Computer Science and Engineeringand our project guide **Dr.Prithi.S, Ph.D.,** for her encouragement and guiding us throughout theproject.

We would like to thank our project coordinator **Mr.S.Vinod Kumar M.Tech.,** for his encouragement and guidance towards successful completion of this project. We also extend our sincere thanks to all faculty members and supporting staffs for their direct and indirect involvement in successful completion of the project. We express our gratitude to our parents, friends and well-wishers for their encouragement and support

PAVITHRA M 170701150

THYAGARAJENT 170701245

ABSTRACT

Travelling has become our daily essential like oxygen, water and food. Vehicles evolved along with human needs and desires, so as the risk and danger. It is natural for a human to get dozed off while driving for a long duration, especially during the night. The drowsiness or fatigue of drivers is one of the main reasons behind road accidents. A continued surveillance with the required preventive/ emergency measures through an efficient model will be provided in this project. This project comprises of two phases. In phase one, with the help of camera module, a Computer Vision model can be used for surveillance and detection of the fatigue state. The proposed system incorporates Digital library. The Face Landmark Detection algorithm offered by Digital library is an implementation of the Ensemble of Regression Trees (ERT) and HOG (Histogram of Oriented Gradients) descriptor for object detection. The EAR (Eye Aspect Ratio) value is determined followed by the threshold frame count to endorse the fatigue state. On the detection of drowsiness or fatigue an alert is displayed on the screen along with an alert alarm. Simultaneously, the achieved value is uploaded in real time to Firebase (cloud database storage). In phase two, a simulated car with Raspberry Pi as its base control unit is established to take inputs from the firebase. For every specified threshold value, the actions of the car are subjected to certain changes. As a preventive measure the motor slows down the vehicle's wheels as the fatigue threshold increases and finally the car is stopped. As an emergency measure the car is integrated with GSM and GPS module which in turn sends a alert message and location of the vehicle to a nearest hospital and one's respective emergency contact.

KEYWORDS: Digital library, Histogram of Oriented Gradients, Eye Aspect Ratio, Ensemble of Regression Trees, GSM, GPS

LIST OF CONTENTS

CHAPTER NO	TITLE	PAGE NO
	ABSTRACT	**iv**
	LIST OF FIGURES	**vii**
1	**INTRODUCTION**	**1**
	1.1 GENERAL	1
	1.2 OBJECTIVE	2
	1.3 MACHINE LEARNING	2
	1.4 INTERNET OF THING	3
	1.5 ELECTRIC VEHICLE	3
	1.6 HISTOGRAM OF ORIENTED GRADIENTS	3
2	**LITERATURE SURVEY**	**4**
	2.1 DRAWBACKS IN EXISTING SYSTEM	12
	2.2 PROPOSED SYSTEM	12
	2.2.1 ADVANTAGES OF PROPOSED SYSTEM	12
3	**SYSTEM DESIGN**	**13**
	3.1 OVERVIEW	13
	3.1.1 ARCHITECTURE DIAGRAM	13
	3.1.2 DEVELOPMENT ENVIRONMENT	14
	3.1.3 DESIGN OF THE ENTIRE SYSTEM	19
	3.1.3.1 USE CASE DIAGRAM	19
	3.1.3.2 ACTIVITY DIAGRAM	21
	3.1.3.3 CLASS DIAGRAM	22
	3.1.3.4 COLLABORATION DIAGRAM	23
4	**PROJECT DESCRIPTION**	**24**
	4.1 GENERAL	24
	4.2 METHODOLOGIES	25
	4.2.1 MODULES	25
	4.2.2 MODULE DESCRIPTION	25

5	**IMPLEMENTATION AND RESULT DISCUSSION**	**27**
6	**SYSTEM TESTING AND MAINTENANCE**	**31**
7	**CONCLUSION AND FUTURE ENHANCEMENTS**	**35**
	7.1 CONCLUSION	35
	7.2 FUTURE ENHANCEMENTS	35
	APPENDIX	**36**
	REFERENCES	**44**

LIST OF FIGURES

FIGNO	TITLE	PAGE NO
3.1.1	ARCHITECTURE DIAGRAM	13
3.1.2.1	RASPBERRY PI	15
3.1.2.2	PIN DIAGRAM	16
3.1.2.3	DC MOTOR	16
3.1.2.4	SIM900A GSM	17
3.1.2.5	NEO-6M GPS	18
3.1.2.6	L298N	18
3.1.3.1	USE CASE DIAGRAM	20
3.1.3.2	ACTIVITY DIAGRAM	21
3.1.3.3	CLASS DIAGRAM	22
3.1.3.4	COLLABORATION DIAGRAM	23
5.1.	CAMERA ADMISSION	27
5.2	SERIAL MONITOR	28
5.3	FATIGUE ALERT	29
5.4	EMERGENCY ALERT MESSAGE	30
6.1.	FATIGUE MONITORING INTERFACE	31
6.2	CLOUD REPOSITORY	32
6.3	GLOBAL POSITIONING SYSTEM FRAME	33
6.4	GSM FRAME	34
6.5	SIMULATED VEHICLE MODEL	34

CHAPTER 1

INTRODUCTION

1.1 GENERAL

The evolution and growth of the automobile industry at the present time period is impeccable. Increase in vehicle automation and self-sustainability are witnessed every day. Despite all the innovations, the road safety and lives of the people on board a vehicle are still intriguing. Every year thousands of people lose their lives due to road accidents. There are many causes for the accident, for example over speeding, drunk and drive, but still the foremost factor especially on rural roads, is the driver's fatigue and monotony. According to the data collected at 2017 from the World Health Organization, an approximate number of 1.25 million deaths has been recorded worldwide, i.e., approximately for every 25 seconds an individual out there will experience a road crash.

The current evolved vehicle generation does not have the proper system to prevent the accidents caused by drowsiness. A proper Fatigue monitoring system should be implemented in order to prevent road crashes and save lives. A model with just detecting the fatigue and alerting the driver alone will not be sufficient enough to prevent road crashes from happening.

A shield system that simultaneously synchronizes with the fatigue monitoring phase to save lives and avoid road accidents must be incorporated. These phases altogether will form an efficient barrier to road accidents. Continuous surveillance with the help of Computer Vision and implementing the changes with respect to the observed changes will enhance vehicle safety and ensures to prevent road crashes.

1.2 OBJECTIVE

Fatigue surveillance and shield system lays driver protection and road safety as its foundation. This model aims to receive facial landmarks of an individual as input, where it is processed to generate a result that will be indicating the state of fatigue. The further proceedings of the model depend on that state and executes the necessary actions to achieve its goal. If an individual is observed to be drowsy or inattentive while driving, the drowsiness is detected through camera and the alert system will be triggered mentioning that the driver is drowsy. This result will be updated in cloud storage for latter. If there is a condition prevails where the driver remains to be drowsy even after the alert alarm and the indication message goes on, the momentum of the vehicle will be reduced step by step with the help of the value attained from the cloud for every threshold frame till the car halts and being prevented from crash. Once the car is halted, a caution text and the location of the fatigued individual will be delivered to their respective emergency contact.

The main objectives of this project are:

- To implement an efficient model for fatigue monitoring.
- To prevent road accidents and save lives.
- To innovate a solution/shield model for aftermath of detection.

1.3 MACHINE LEARNING

Machine learning is a form of AI that enables a system to learn from data rather than through explicit programming. However. Machine learning is not a simple process. As the algorithms ingest training data, ist is then possible to produce more precise models based on the data. A machine learning model is the output generated when you train your machine-learning algorithm with data. After training, when you provide a model with an input, you will be given an output. For example, a predictive algorithm will create a predictive model. Then, when you provide the predictive model with data, you will receive prediction based on the data that trained the model. Machine learning enables models to train on data sets before being deployed. Some machine- learning models are online and continuous. This iterative process of online models leads to an improvement in the types of associations made between data elements. Machine-learning techniques are required to improve the accuracy of predictive models. Depending on the nature of the business problem addressed, there are different approaches based on the type and volume of the data.

1.4 INTERNET OF THINGS

The Internet of Things (IoT) refers to a system of interrelated, internet-connected objects that are able to collect and transfer data over a wireless network without human intervention. A thing in the internet of things can be a person with a heart monitor implant, a farm animal with a biochip transponder, an automobile that has built-in sensors to alert the driver when tire pressure is low or any other natural or man-made object that can be assigned an Internet Protocol (IP) address and is able to transfer data over a network.

1.5 ELECTRIC VEHICLE

An electric vehicle (EV) is a vehicle that uses one or more electric motors or traction motors for propulsion. An electric vehicle may be powered through a collector system by electricity from off-vehicle sources, or may be self-contained with a battery, solar panels, fuel cells or an electric generator to convert fuel to electricity.

1.6 HISTOGRAM OF ORIENTED GRADIENTS

HOG, or Histogram of Oriented Gradients, is a feature descriptor that is often used to extract features from image data. The HOG descriptor focuses on the structure or the shape of an object. Now you might ask, how is this different from the edge features we extract for images? In the case of edge features, we only identify if the pixel is an edge or not. HOG is able to provide the edge direction as well. This is done by extracting the gradient and orientation.

CHAPTER 2

LITERATURE SURVEY

A Real-time Driving Drowsiness Detection Algorithm with Individual Differences Consideration [1]

In this paper, the algorithm detects the individual differences of a driver. A deep cascaded convolutional neural network is constructed to detect the face region and also where the eye aspect ratio concept is introduced to evaluate the drowsiness of the individual in the present frame.

Advantages:

- It doesn't struggle to identify faces of numerous individuals.
- Usage of deep cascaded CNN avoids the problem poor accuracy by artificial feature extraction.

Disadvantages:

- Its performance is dragged down when the face is subjected to very dim light ambience.

Facial features monitoring for real time drowsiness detection [2]

In this paper, the author proposes efficient model that incorporates three phases for real time facial monitoring namely, proposes an efficient facial features detection using Viola Jones, the eye tracking and yawning detection.

Advantages:

- The illumination of the skin part and considering the chromatic components alone helps in rejecting most of the non-facial image background.

- Implementation of binary linear Support Vector Machine classifies continues frames into fatigue and non-fatigue states, thus making the system highly efficient.

Disadvantages:

- The system is unable to predict drowsiness while the spectacles is worn by an individual.

Enhanced Drowsiness Detection Using Deep Learning [3]

This paper presents deep learning based drowsiness detection for brain-computer interface (BCI) using functional near-infrared spectroscopy (fNIRS) where the activities of the brain are measured with continuous- wave fNIRS system. Deep neural network (DNN) are implemented to classify the drowsy and alert states from the observed brain activities.

Advantages:

- The inclusion of the used CNN architecture attains an accuracy of 99.3% making the system reliable.
- The proposed approach not just detects but is also efficient in accessing the frame location for a passive BCI.

Disadvantages:

- The use of long-time windows, makes the study and grinds down the detection module less applicable for real-time scenarios

Comprehensive Drowsiness Level Detection Model Combining Multimodal Information [4]

This paper presents the drowsiness detection model which has the capacity of sensing the entire range of stages of drowsiness. The approach behind this model makes use of sitting posture-related index value that can indicate weak drowsiness.

Advantages:

- This system doesn’t struggle to comprehend and differentiate various drowsiness state.
- The posture information improves the accuracy of weak drowsiness detection and helps to cover all stages of drowsiness.

Disadvantages:

- Despite differentiating the level of drowsiness, the model just revolves around the detection leaving behind the safety and alerting measures.

Real-time Driver Drowsiness Detection for Android Application Using Deep Neural Networks Techniques [5]

In this paper, the author proposes a novel approach towards real-time drowsiness detection which is based on deep learning and can be executed on an android applications with high accuracy.

Advantages:

- Having a precise structure this model can be easily integrated into advanced driver assistance system and mobile application.

Disadvantages:

- Being an android application module, the model can't detect the distraction if the driver is rotating his head / lurking sideways and yawning of the driver.
- Current applications cannot be used in embedded systems due to their limited calculation and storage space.

A Driver Face Monitoring System for Fatigue and Distraction Detection [6]

The Author proposes a real-time system that can detect driver fatigue and distractions using machine vision approaches where face template matching and horizontal projection of top-half face image are utilised to obtain hypo vigilance symptoms from the face.

Advantages:

- Implementation of adaptive hypo vigilance monitoring modules for standard inputs.
- A very short training phase makes this model robust and efficiently detect individuals with different face and eyelid behavior.

Disadvantages:

- Despite several step approach to detect, the face tracking method is inaccurate and very computationally complex.

Real time driver drowsiness detection using facial features [7]

In this paper, the author proposes a system named DriCare that modulates to detect the driver's fatigue status such as yawning, blinking, and duration of eye closure, which introduces a new face-algorithm to improve tracking accuracy.

Advantages:

- Improvised KCF algorithm is a key factor.
- The non-contact method is cost efficient that does not require Computer Vision technology or sophisticate camera which allows the model usage in more cars.

Disadvantages:

- This model when subjected to dim light conditions, the detection rate drops and it may tend to lose accuracy.

Small Faces Attention face detector [8]

This paper proposes a new scale-invariant face detector, which is called as Small Faces Attention (SFA) face detector. This model is built over the first ever multi-branch face detection architecture and adopts multi-scale training and testing to make the model robust towards various scale.

Advantages:

- Promising detection performances on challenging detection benchmarks like WIDER FACE and FDDB data sets.
- Having multi-branch detection architecture, this model pays more attention to faces with small scale.

Disadvantages:

- Despite its accuracy, the model is not applicable Real Time

A Method of driver's eyes closure and yawning detection for drowsiness analysis using infrared camera [9]

In this paper, the author proposes a new module to detect an individual's eyes closure and yawning for drowsiness analysis using infrared camera. This module incorporates four steps, namely, face detection, eye detection and eyes closure and yawning detection.

Advantages:

- Enhancing the accuracy of the model by incorporating Augment Drowsiness monitoring during dim light conditions.

- This method can detect eye closure or yawning even in low light ambience.

Disadvantages:

- Errors occurred when a face is occluded such as hand and the method did not perform effectively when capture angle of camera is varied in width range.

A Novel Design of Autonomous Cars using IoT and Visual Features [10]

This paper proposes a system that utilizes mathematical models like neural networks and image processing techniques to sense the environment and it had been implemented as three major components, namely, curved road detection, road sign and signal detection and obstacle detection.

Advantages:

- This model is well trained and is efficient on obstacle and road sign detection.
- The proposed framework can work efficiently in breaking down the sign at both sunlight and night enlightenments.

Disadvantages:

- Its proficiency could be additionally upgraded on the off chance that it begins learning by itself and maintain a strategic distance from pointless figurines of the areas which are as of now known or commonplace.

Development of Arduino Glove-Based Autonomous Car [11]

This paper presents a low-cost human-computer Interaction device represented by Arduino glove to control RC car. Arduino glove includes gyro sensor to identify the angle of the hand while the fingers' movements are identified by the flex sensor.

Advantages:

- It is a low-cost human-computer.

Disadvantages:

- Flex sensor can be replaced with potentiometer to reduce cost as well as efficiency.

Development of an Autonomous RC-car [12]

In this paper we tackle the development of a robotic-car with hardware control, lane detection, mapping, localization and path planning capabilities. IT also introduce a generic map analysis system to increase the efficiency of certain paths on the track.

Advantages:

- It is completely independent and robust.

Disadvantages:

- The model is kinematic which self restricts to take physical properties into account.

2.1 DRAWBACKS IN EXISTING SYSTEM

- Most of the algorithm struggles to detect eyes in dim light conditions and while wearing glasses.
- Recovery phase to prevent accident has not prevalent.
- The data in the cloud can be tampered.
- The retrieval speed of the application is comparatively low.

2.2 PROPOSED SYSTEM

Fatigue Surveillance and Shield system focuses on capturing individual’s faces as inputs and processes them to attain a result which is simultaneously evaluated to determine whether the driver is in conscious or unconscious, i.e. Drowsy state. When an individual is observed to be drowsy, the system architecture is designed in a way that the open computer vision records each frame and triggers the alert message and alarm to indicate the drowsy state of the respective individual and as a road crash preventive measure, the vehicle reduces the speed gradually with the help of the frame count data collected using OpenCV. If the driver does not gains back one’s consciousness , the motion of the vehicle comes to rest while an emergency text and the location of the vehicle is delivered to their respective emergency contact suspecting that the driver might in recoverable danger.

2.2.1 ADVANTAGES OF PROPOSED SYSTEM

- Improvised quick facial feature recognition.
- Higher accuracy of detecting the eyes while wearing spectacles
- Innovation of shield system to prevent road crashes.
- Recovery phase to establish communication during causalities.

CHAPTER 3

SYSTEM DESIGN

3.1 OVERVIEW

System Design deals with the various UML [Unified Modelling Language] diagrams for the implementation of projects. Design is a meaningful engineering representation of a thing that is to be built. Software design is a process through which the requirements are translated into representation of the software. Design is the place where quality is rendered in software engineering. Design is the means to accurately translate customer requirements into finished products.

3.1.1 ARCHITECTURE DIAGRAM

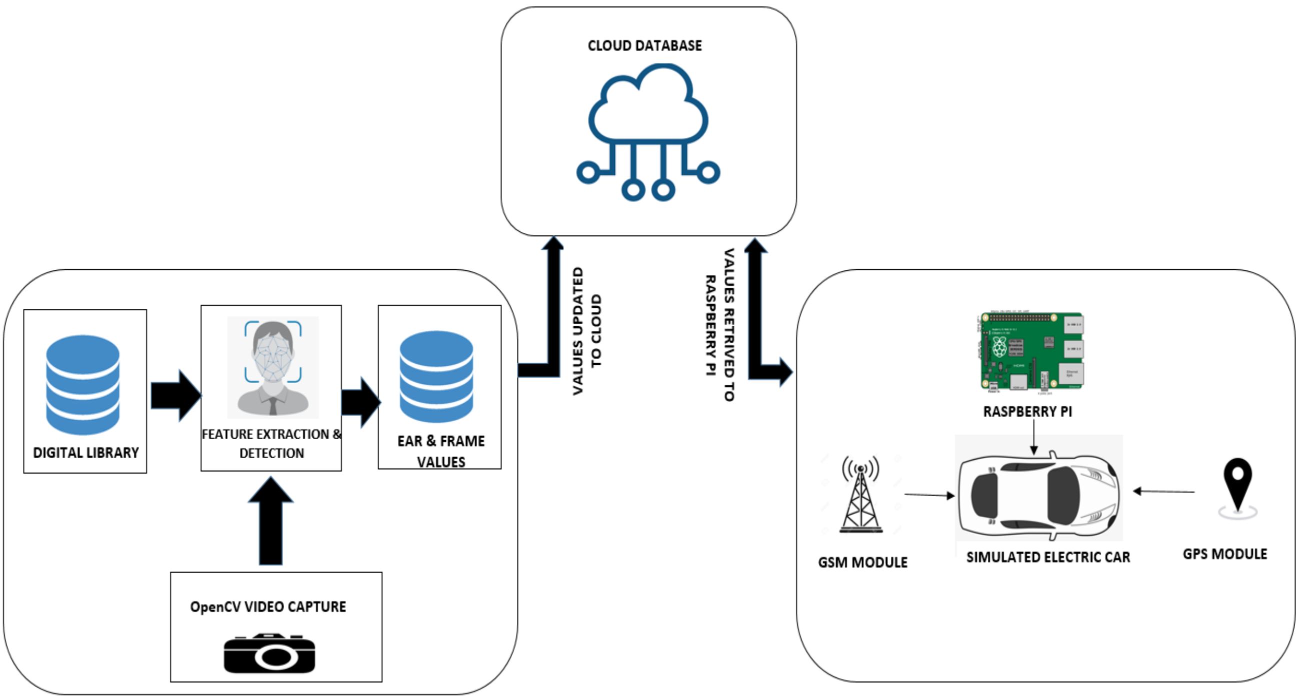


FIG 3.1.1 ARCHITECTURE DIAGRAM

In FIG 3.1.1, the architecture diagram channelizes from the extraction of image frames and processing it with the Dlib to plot the facial features and detect the eyes.

While the EAR value is calculated for each frames the values and frame counts are passed to the cloud database which in turn acts as the input for the raspberry pi that acts as the vehicle's control unit. GSM and GPS are the key components of the recovery phase.

3.1.2 DEVELOPMENT ENVIRONMENT

HARDWARE REQUIREMENTS:

The hardware requirements may serve as the basis for a contract for the implementation of the system and should, therefore, be a complete and consistent specification of the whole system. They are used by the software engineers as the starting point for the system design. It shows what the systems do and not how it should be implemented.

- Processor :Intel core duo or higher
- RAM :4GB(Minimum)
- Hard disk : (50 GB of free space)
- Monitor :15" color with VGI card support
- Processorspeed : Min 500 MHZ
- Camera :Any USB camera
- Operating system :Any OS

RASPBERRY PI:

FIG 3.1.2.1 RASPBERRY PI

In FIG 3.1.2.1, the simulated car is controlled by the raspberry pi. A low cost Linux and ARM based computer on a small circuit board sponsored by the charitable Raspberry Pi Foundation in UK. Raspbian is the Debian-based Linux OS that is provided with the device. The Raspberry Pi Model B was released February 2016 with a 1.2 GHz 64-bit quad core ARM Cortex-A53 processor, on board 802.11n Wi-Fi, Bluetooth and USB boot capabilities. While operating at 700 MHz by default, the first generation Raspberry Pi provided a real-world performance roughly equivalent to 0.041 GFLOPS. he graphical capabilities of the Raspberry Pi are roughly equivalent to the performance of the Xbox of 2001. The Raspberry Pi hardware has evolved through several versions that feature variations in the type of the central processing unit, amount of memory capacity, networking support, and peripheral-device support. After many upgrades the Raspberry Pi 4, with a quad-core ARM Cortex-A72 processor, is described as having three times the performance of a Raspberry Pi 3.

PIN DIAGRAM:

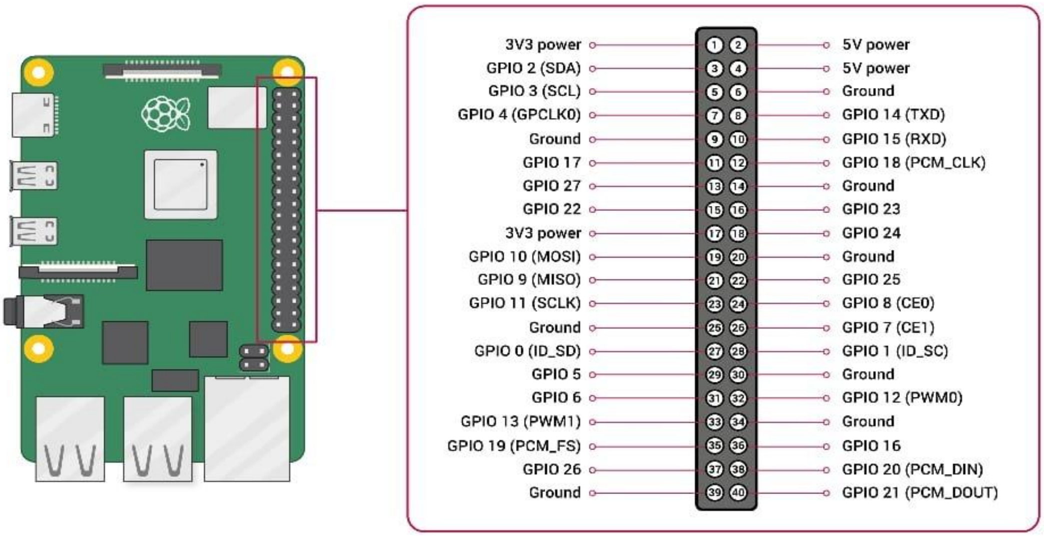


FIG 3.1.2.2 PIN DIAGRAM

In FIG 3.1.2.2, the pin configuration of the Raspberry pi is well portrayed clearly. A 40-pin GPIO header is found on all current Raspberry Pi boards.

DC MOTOR:

FIG 3.1.2.3 DC MOTOR

In FIG 3.1.2.3, a DC motor is any of a class of rotary electrical motors that converts direct current electrical energy into mechanical energy. The most common types rely on the forces produced by magnetic fields. Nearly all types of DC motors have some internal mechanism, either electromechanical or electronic, to periodically change the direction of current in part of the motor. DC motors were the first form of motor widely used, as they could be powered from existing direct-current lighting power distribution systems. A DC motor's speed can be controlled over a wide range, using either a variable supply voltage or by changing the strength of current in its field winding.

SIM900A GSM:

FIG 3.1.2.4 SIM900A GSM

In FIG 3.1.2.4, the SIM900A is a readily available GSM/GPRS module, used in many mobile phones and PDA. The module can also be used for developing IOT (Internet of Things) and Embedded Applications. SIM900A is a dual-band GSM/GPRS engine that works on frequencies EGSM 900MHz and DCS 1800MHz. SIM900A features GPRS multi-slot class 10/ class 8 (optional) and supports the GPRS coding schemes CS-1, CS-2, CS-3 and CS-4.

NEO-6M GPS:

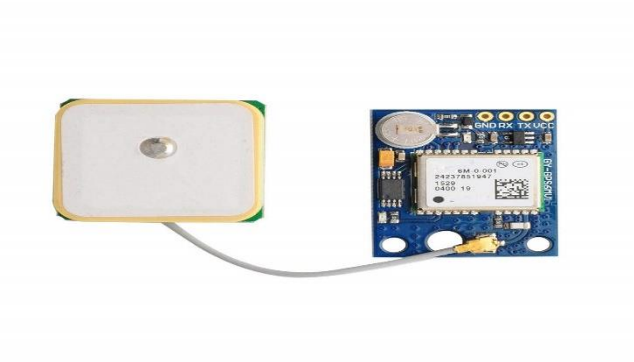


FIG 3.1.2.5 NEW-6M GPS

In FIG 3.1.2.5, the NEO-6M GPS engine on these modules is quite a good one, and it also has high sensitivity for indoor applications. Furthermore, there’s one MS621FE-compatible rechargeable battery for backup and EEPROM for storing configuration settings. The module works well with a DC input in the 3.3- to 5-V range.

L298N:

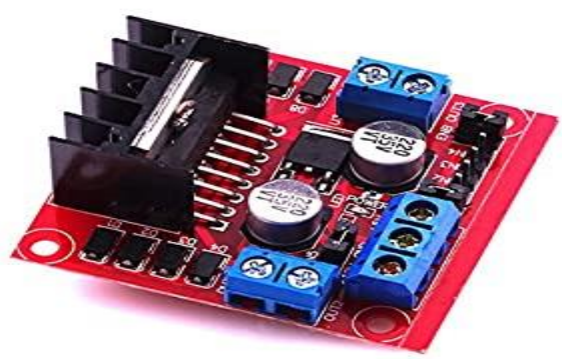


FIG 3.1.2.6 L2989

In FIG 3.1.2.6, this L298N Motor Driver Module is a high power motor driver module for driving DC and Stepper Motors. This module consists of an L298 motor driver IC and a 78M05 5V regulator. L298N Module can control up to 4 DC motors, or 2 DC motors with directional and speed control.

SOFTWARE REQUIREMENTS

The software requirements are the specification of the system. It should include both a definition and a specification of requirements. It is a set of what the system should do rather than how it should do. The software requirements provide a basis for creating the software requirements specification. It is useful in estimating cost, planning team activities, performing tasks and trackingthe team's progress throughout the development activity.

- OS : Ubuntu 16.04 LTS(64bit)
- cURL : latest version
- Docker : version 17.06.2-ce or greater
- Docker Compose : version 1.14.0 or greater
- Golang : version 1.11x
- NPM : version 5.x
- Python : version 2.7

3.1.3 DESIGN OF THE ENTIRE SYSTEM

3.1.3.1 USE CASE DIAGRAM

There are three actors in the driver, cloud and simulated car. The face of the driver is detected. The EAR value is calculated and uploaded to the cloud. The simulated car gets the value from the cloud and reduces the speed when drowsy**.**

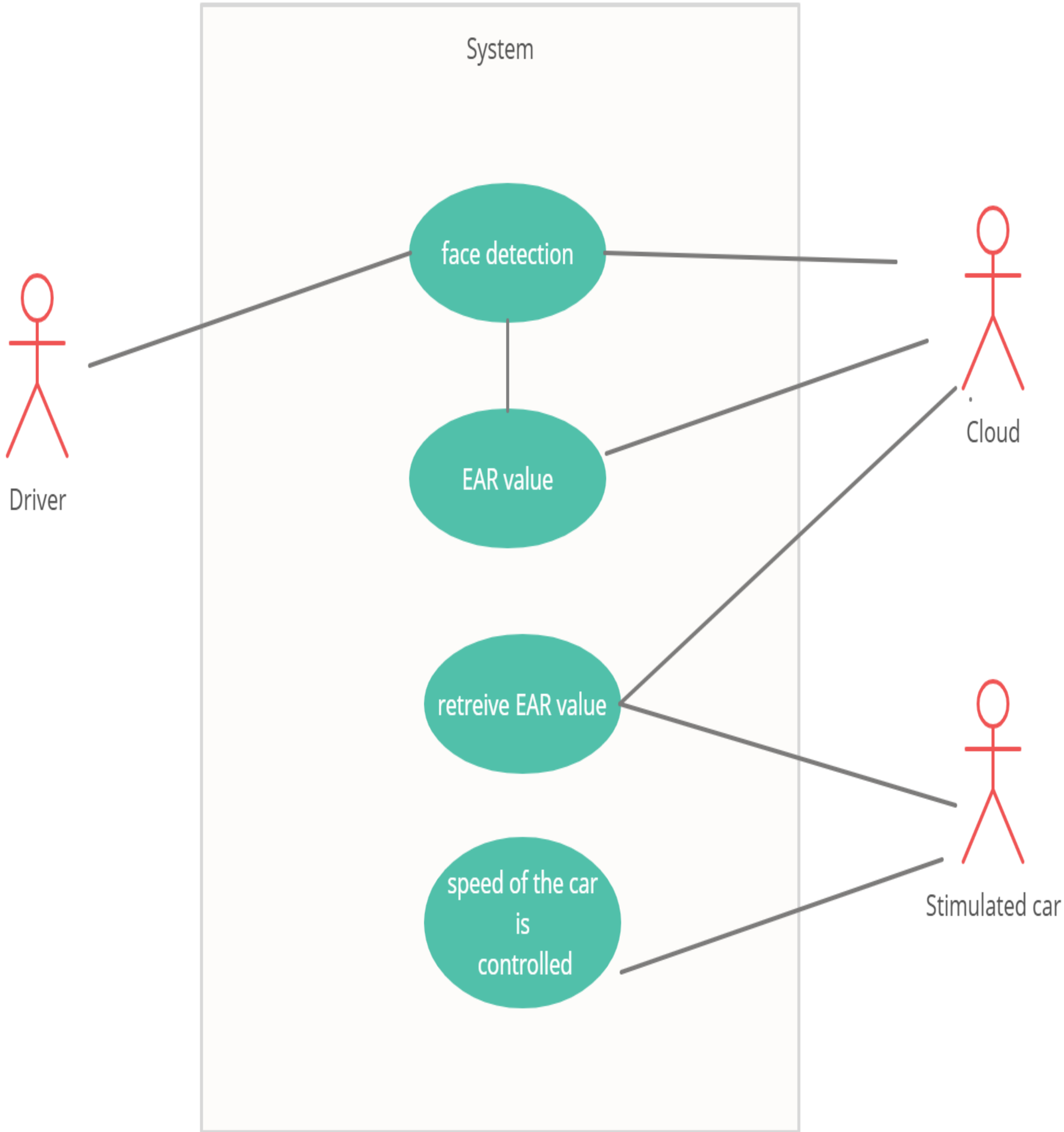


FIG 3.1.3.1 USE CASE DIAGRAM

3.1.3.2 ACTIVITY DIAGRAM

The flow of activity starts from the facial detection. The face is detected and the EAR value is calculated which is uploaded into cloud. With the help of raspberry pi, the simulated car gets the EAR value and if the frames exceed the threshold value, the speed of the car is reduced and the location is sent to the emergency contact.

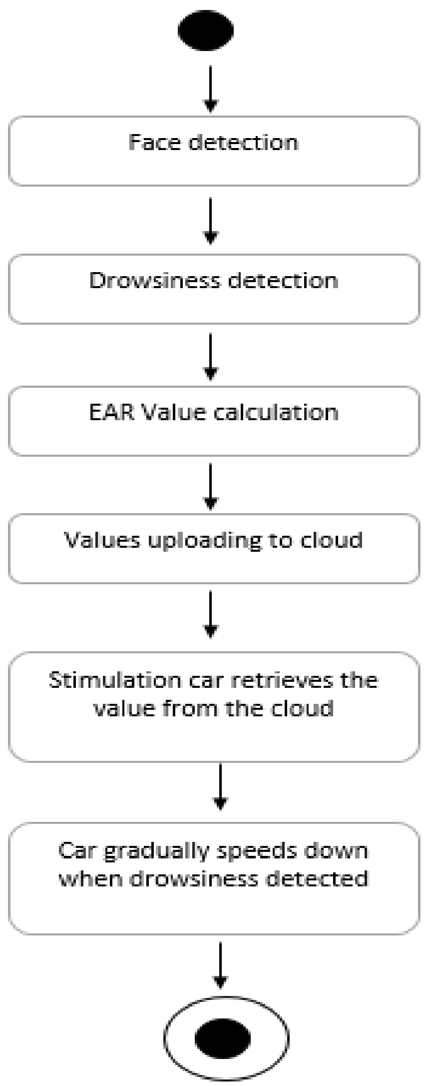


FIG 3.1.3.2 ACTIVITY DIAGRAM

3.1.3.3 CLASS DIAGRAM

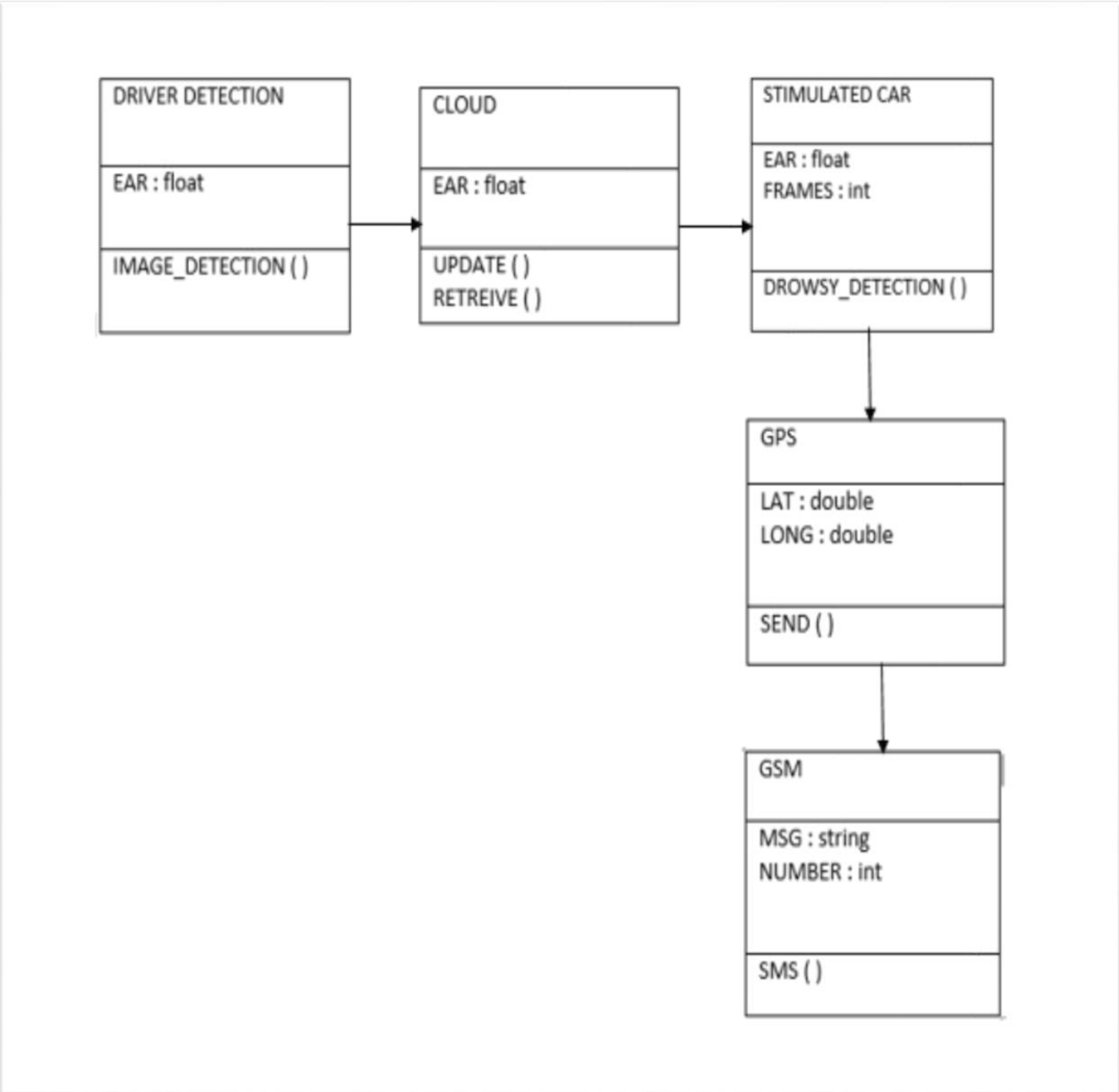


FIG 3.1.3.3 CLASS DIAGRAM

Driver Detection

When the face of the driver is detected, the EAR value is calculated.

Cloud

The EAR value is uploaded to the cloud and can also be retrieved.

Stimulated Car

The simulated car EAR value is retrieved and also compared with the number of frames.

GPS

The GPS coordinates longitude and longitude are needed for the location.

GSM

The location is sent to the emergency contact.

3.1.3.4 COLLABORATION DIAGRAM

This diagram depicts how components are wired together to form larger components or software systems. They are used to illustrate the structure of arbitrarily complex systems.

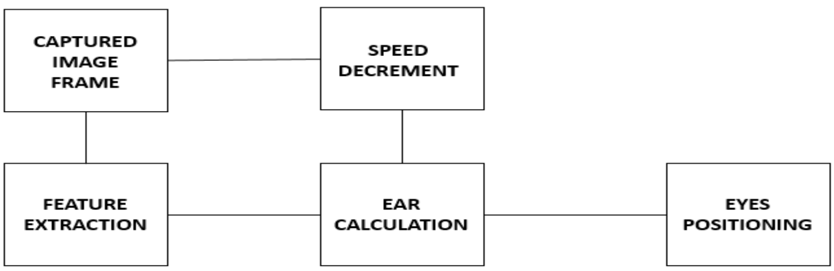


FIG 3.1.3.4 COLLABORATION DIAGRAM

CHAPTER 4

PROJECT DESCRIPTION

4.1 GENERAL

The Fatigue surveillance and shield system lays driver protection and road safety as its foundation. This model extracts facial landmarks with OpenCV of an individual as input, where it is processed to generate a result of facial landmarks. While the model is locating the eyes region, the respective EAR value will be calculated. With EAR value, if it is less than the threshold value, then that will be indicating the state of fatigue.

The further proceedings of the model depend on that state and executes the necessary actions to achieve its goal. If an individual is observed to be drowsy or inattentive while driving, the drowsiness is detected and the alert system will be triggered mentioning that the driver is drowsy.

This EAR and frame count value will be updated in cloud storage for latter. If there is a condition prevails where the driver remains to be drowsy even after the alert alarm and the indication message goes on, the momentum of the vehicle will be reduced gradually with the help of the EAR value attained from the cloud for every threshold frame till the car halts and being prevented from crash. Once the car is halted, using GSM and GPS a caution text and the location of the fatigued individual will be delivered to their respective emergency contact suspecting that the driver might in recoverable danger.

4.2 METHODOLOGIES

4.2.1 MODULES

- Image acquisition
- Face and eye detection
- Fatigue monitor
- Road crash recovery

4.2.2 MODULE DESCRIPTION

1.IMAGE ACQUISITION

This module is used for capturing image frames of an individual with the camera affixed. Using Computer Vision the camera is accessed and the face is captured live. These acquired frames are then converted to grey scale for further processing. Being the initial module, the image acquisition never stops or changes.

2. FACE AND EYE DETECTION

In this module the base of face detection, i.e., facial feature extraction of the individual is carried over. Dlib’s shape predictor is trained with 300W–IBUG training dataset. The 300W-ibug dataset consists of 11 columns and 1786 rows. It standardizes multiple alignment databases with 68 landmarks. The goal of 300W–IBUG is to train a shape predictor capable of localizing each individual facial structure. With the help of trained Dlib, it will be efficient to mark the facial feature landmarks and identify the eye region. The data points of the eye region will be extracted.

3. FATIGUE MONITOR

This module helps in monitoring the fatigue state of an individual. HOG (Histogram of Oriented Gradients) is a feature descriptor, generally used for object detection. The converted grey scale frame from the image acquisition module is then passed to HOG and the co-ordinates of the facial features are extracted from the 5 face landmarks defined in the dataset. The EAR is defined as the ratio of the vertical to the horizontal Euclidean distance of the landmarks in the eye. The EAR value determines whether is the eye is at open or closed state while continuously monitoring the closure angle of the eyes. The EAR (Eye Aspect Ratio) for the left and right eye is calculated with the attain coordinates. Once the EAR value drops below the threshold value, then it is considered as the fatigue state.

4. ROAD CRASH RECOVERY

The attained EAR value will be simultaneously uploaded to the cloud database. From the cloud the values are taken as input by the raspberry Pi control unit. Based on the input EAR value being received, the motor reduces the speed/rpm of the car for every threshold frame value and finally the vehicle attains the rest. Once the car is in ceased position, with the help of GSM and GPS modules the location and alert message is sent to the emergency contact for recovery. This module innovates road safety by preventing the road crash by ceasing the vehicles motion.

CHAPTER 5

IMPLEMENTATION AND RESULT DETECTION

5.1 CAMERA ADMISSION

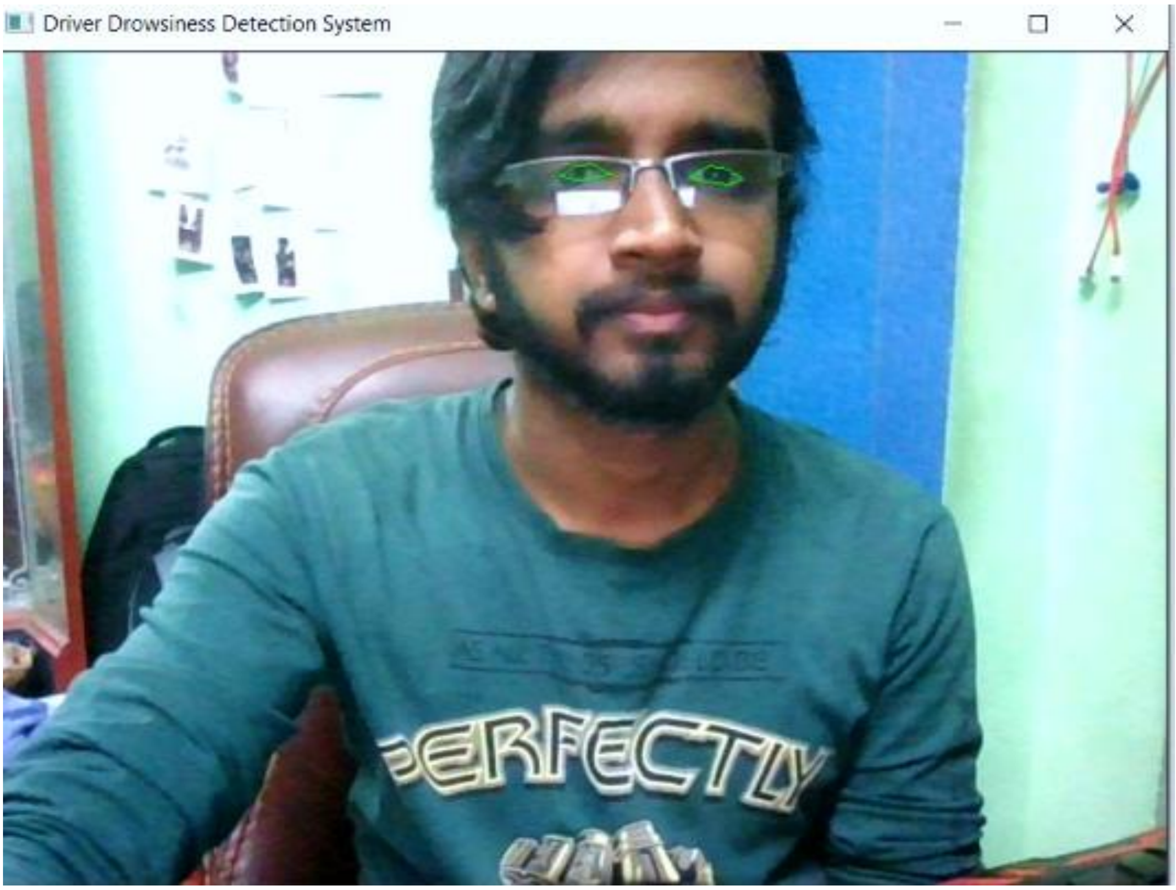


Fig 5.1. CAMERA ADMISSION

In FIG 5.1, once the program is executed, the OpenCV access the camera and starts the capturing. The face is captured frame by frame continuously without any halts. Thus the OpenCV acts as the entry point.

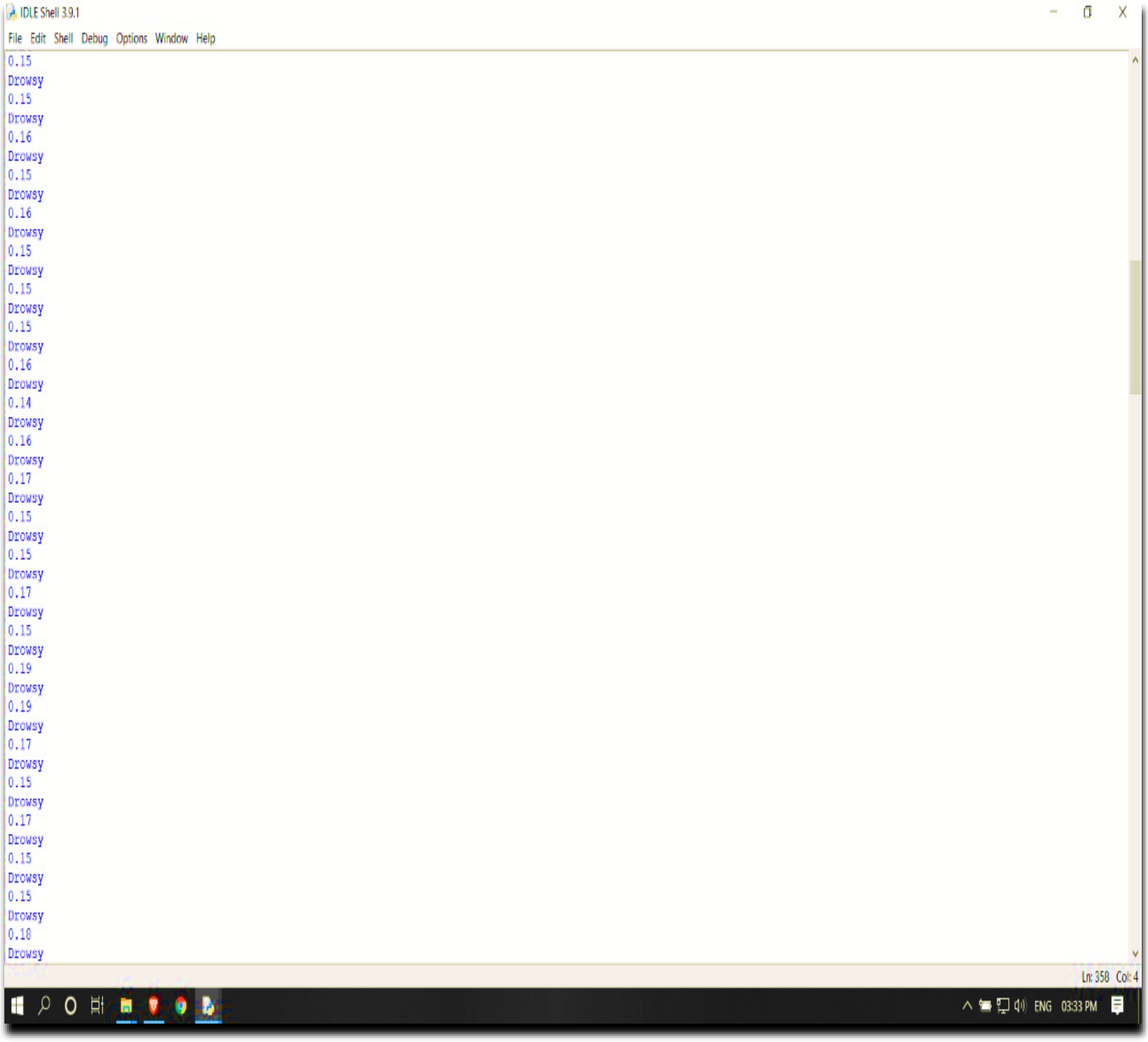


FIG 5.2 SERIAL MONITOR

In FIG 5.2, after the detection phase, with the use of Dlib and HOG face descriptor the facial landmarks for the eyes are marked and their respective EAR value is calculated. The distance between each eye coordinate is taken as input and the EAR is based over that values. In the serial monitor the EAR value calculated for each frame is printed as output. Once the value drops below the threshold EAR value, then drowsy will be displayed.

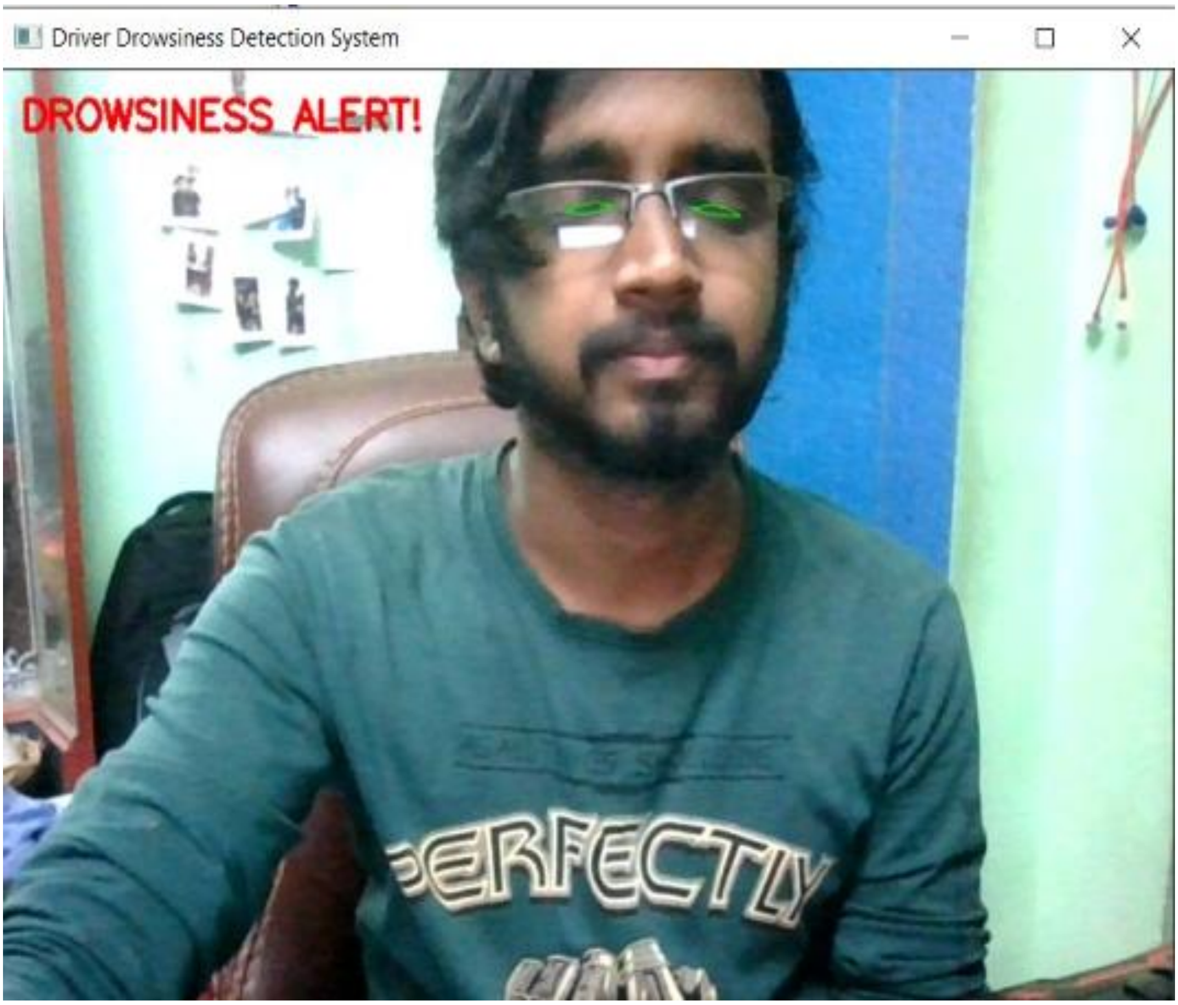


FIG 5.3 FATIGUE ALERT

In FIG 5.3, after successful detection of the eye closure, the EAR value drops below the threshold value and the fatigue alert message is triggered on the Computer Vision window. Along with the alert message, an alarm sound is flashed to bring the driver back to conscious. This process continues relentlessly in a loop without any hindrance.

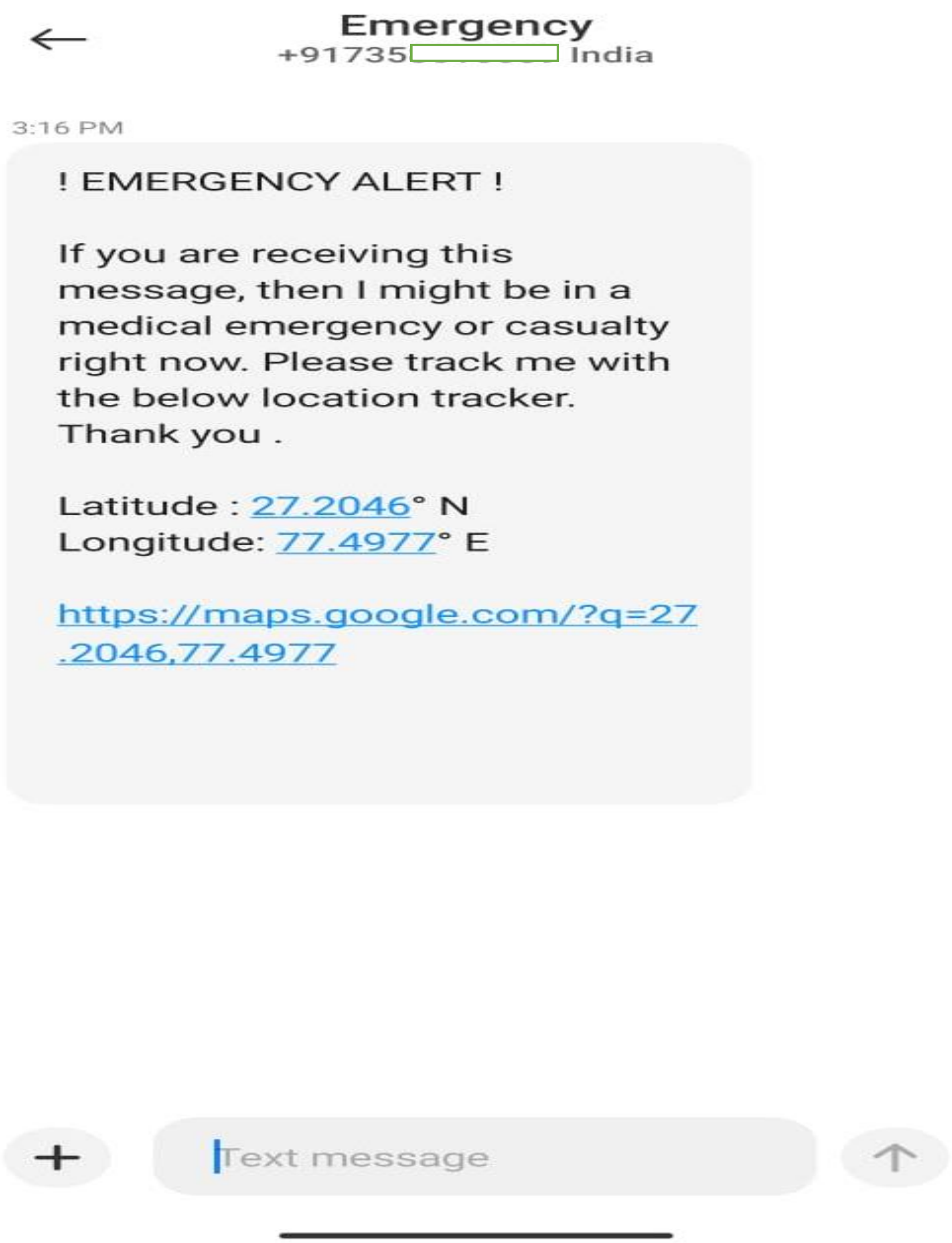


FIG 5.4 EMERGENCY ALERT MESSAGE

In FIG 5.4, the emergency text along with the location received by the emergency contact is displayed. Once the vehicle comes to complete rest position, the alert message indicating need for help along with the location to track down the individual will be delivered without any interruption.

CHAPTER 6

SYSTEM TESTING AND MAINTENANCE

6.1 GENERAL

Thepurposeoftestingistodiscovererrors.Testingistheprocessoftrying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, subassemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement

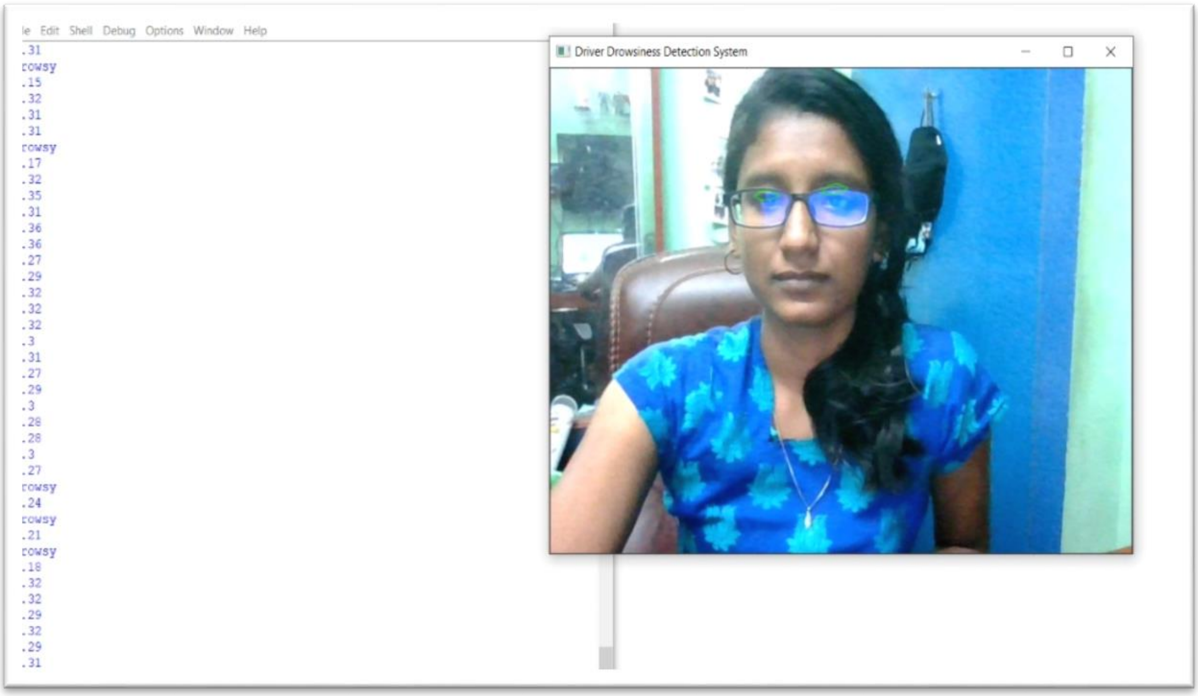


FIG 6.1 FATIGUE MONITORING INTERFACE

In FIG 6.1, the fatigue monitoring interface is activated while the facial feature, i.e., the eyes of the individual is highlighted and EAR values are displayed simultaneously.

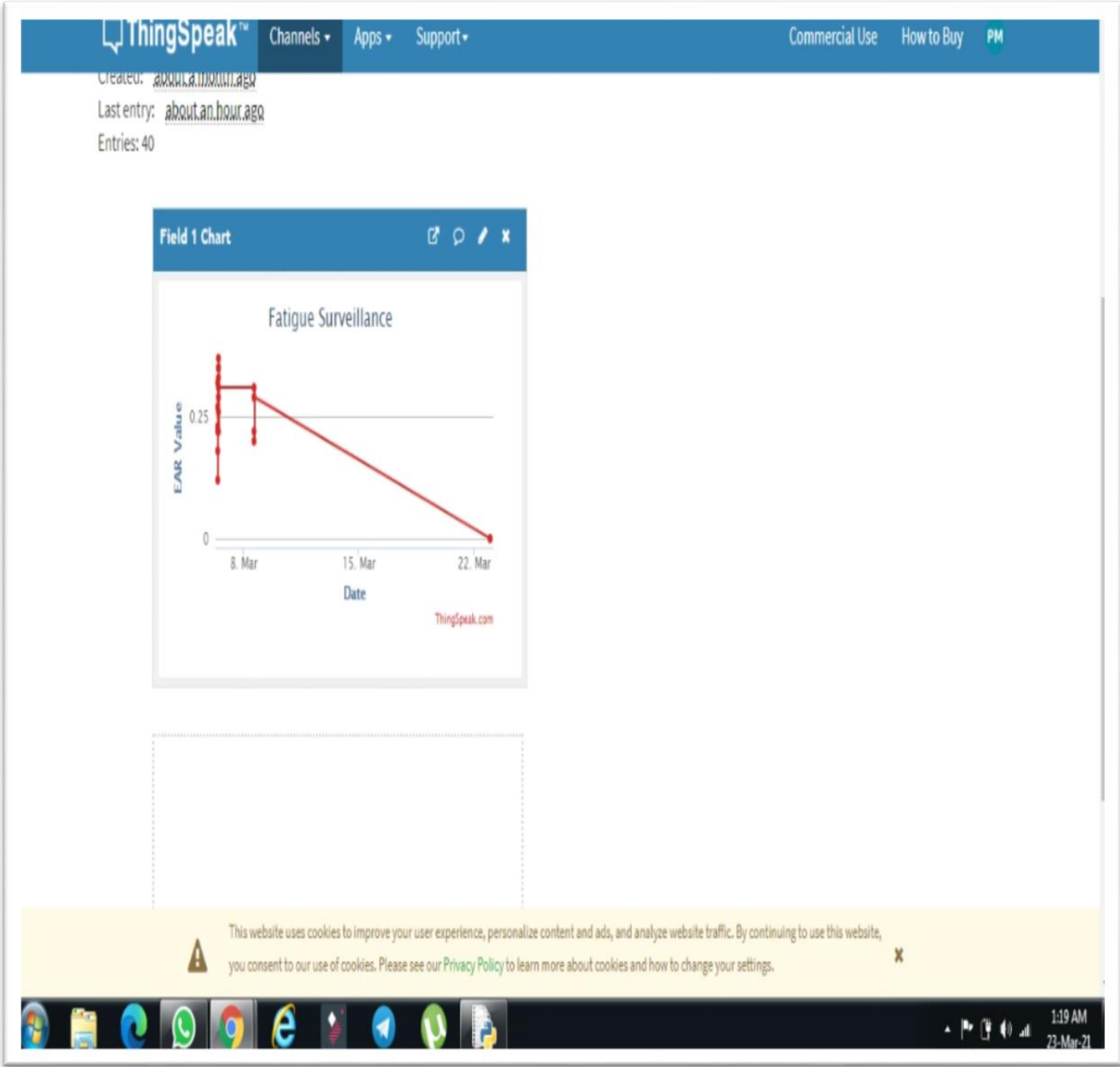
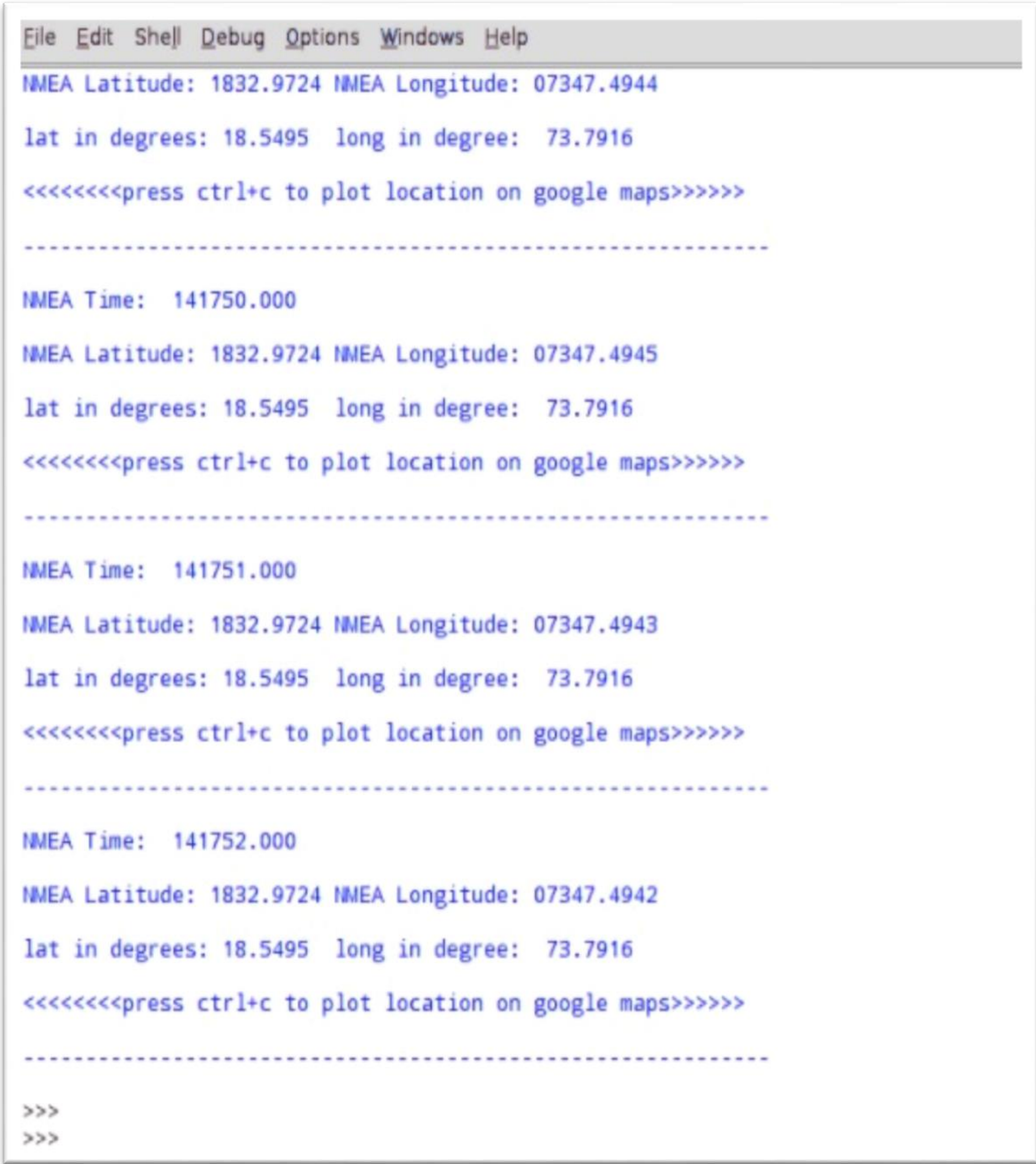


FIG 6.2 CLOUD REPOSITORY

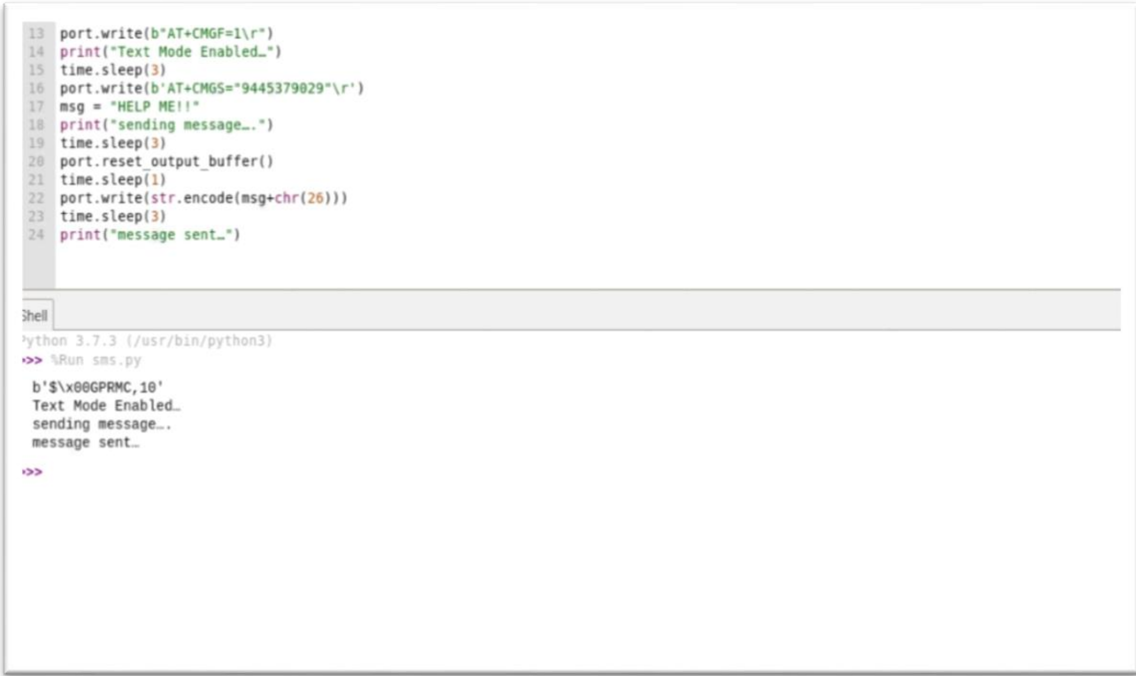
In FIG 6.2, the EAR value calculated while detecting are passed to the cloud repository and stored. The data are secured from being tampered. The graph represents the value's rise and fall during the fatigue monitoring phase.



```
File  Edit  Shell  Debug  Options  Windows  Help
NMEA Latitude: 1832.9724 NMEA Longitude: 07347.4944

lat in degrees: 18.5495  long in degree:  73.7916

<<<<<<<<press ctrl+c to plot location on google maps>>>>>>

----------------------------------------------------------

NMEA Time:  141750.000

NMEA Latitude: 1832.9724 NMEA Longitude: 07347.4945

lat in degrees: 18.5495  long in degree:  73.7916

<<<<<<<<press ctrl+c to plot location on google maps>>>>>>

----------------------------------------------------------

NMEA Time:  141751.000

NMEA Latitude: 1832.9724 NMEA Longitude: 07347.4943

lat in degrees: 18.5495  long in degree:  73.7916

<<<<<<<<press ctrl+c to plot location on google maps>>>>>>

----------------------------------------------------------

NMEA Time:  141752.000

NMEA Latitude: 1832.9724 NMEA Longitude: 07347.4942

lat in degrees: 18.5495  long in degree:  73.7916

<<<<<<<<press ctrl+c to plot location on google maps>>>>>>

----------------------------------------------------------

>>>
>>>
```

FIG 6.3 GLOBAL POSITIONING SYSTEM FRAME

In FIG 6.3, the testing of GPS code, while locating sharing and tracing coordinates is displayed. The coordinate of both latitude and longitude along with the time when the location is recorded is shared.



```
port.write(b"AT+CMGF=1\r")
print("Text Mode Enabled…")
time.sleep(3)
port.write(b'AT+CMGS="9445379029"\r')
msg = "HELP ME!!"
print("sending message….")
time.sleep(3)
port.reset_output_buffer()
time.sleep(1)
port.write(str.encode(msg+chr(26)))
time.sleep(3)
print("message sent…")
```

```
Shell
Python 3.7.3 (/usr/bin/python3)
>>> %Run sms.py

 b'$\x00GPRMC,10'
 Text Mode Enabled…
 sending message….
 message sent…

>>>
```

FIG 6.4 GSM FRAME

In FIG 6.3, the testing of GSM with local SIM and a confirmation code of sent message is displayed.

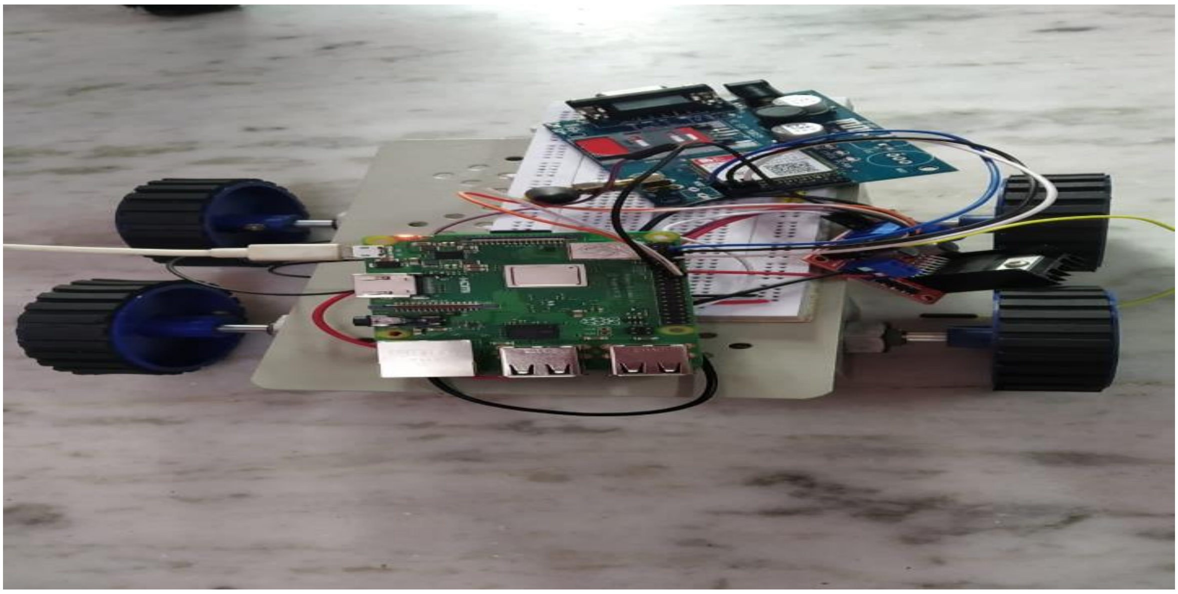


FIG 6.5 SIMULATED CAR MODEL

In FIG 6.5, the simulated car of a real time vehicles is at rest position to prevent crash, once after the fatigue monitor reaches the threshold EAR value.

CHAPTER 7

CONCLUSION AND FUTURE ENHANCEMENTS

7.1 CONCLUSION

The Fatigue surveillance and shield system performs and provides the necessary road safety by preventing accidents with the system’s enhanced capability of being agile to the result for every single second. Having the recovery phase, the model takes over the control on road and not just winds up with the detection and alert phase. Speed reduction and location sharing are the apex features that ensures Z grade safety in this practical world.

7.2 FUTURE ENHANCEMENT

The model can be further enhanced by incorporating AI in the vehicle’s driving mode. When detecting drowsiness, the accident can be further prevented with self-parking feature and wide alarm sound for causality recognition.

Different approaches for detecting drowsiness can be handled, such as body postures and linking smart watch with vehicle driving mode. Whenever an indication of cardiac arrest arises, the shield system can be activated.

APPENDIX 1

SOURCE CODE

FATMON.PY FILE

```
import cv2

importdlib

fromscipy.spatial import distance

importplaysound

import time

from threading import Thread


EYE_AR_THRESH = 0.26

EYE_AR_CONSEC_FRAMES = 38

COUNTER = 0

ALARM_ON = False


defsound_alarm(path):

playsound.playsound(path)


defcalculate_EAR(eye):

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

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

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

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

returnear_aspect_ratio


cap = cv2.VideoCapture(0)

hog_face_detector = dlib.get_frontal_face_detector()


dlib_facelandmark = dlib.shape_predictor(
"shape_predictor_68_face_landmarks.dat")


while True:

    , frame = cap.read()


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

faces = hog_face_detector(gray)

for face in faces:


face_landmarks = dlib_facelandmark(gray, face)

leftEye = []

rightEye = []
```

```
for n in range(36, 42):
        x = face_landmarks.part(n).x
        y = face_landmarks.part(n).y
leftEye.append((x, y))
next_point = n+1
if n == 41:
next_point = 36
x2 = face_landmarks.part(next_point).x
        y2 = face_landmarks.part(next_point).y
cv2.line(frame, (x, y), (x2, y2), (0, 255, 0), 1)

for n in range(42, 48):
        x = face_landmarks.part(n).x
        y = face_landmarks.part(n).y
rightEye.append((x, y))
next_point = n+1
if n == 47:
next_point = 42
x2 = face_landmarks.part(next_point).x
        y2 = face_landmarks.part(next_point).y
cv2.line(frame, (x, y), (x2, y2), (0, 255, 0), 1)

left_ear = calculate_EAR(leftEye)
```

```
right_ear = calculate_EAR(rightEye)

        EAR = (left_ear+right_ear)/2

        EAR = round(EAR, 2)

if EAR < EYE_AR_THRESH:
            COUNTER += 1
if COUNTER >= EYE_AR_CONSEC_FRAMES:
if not ALARM_ON:
                    ALARM_ON = True
                    t = Thread(target=sound_alarm,
args=('alarm_sound.wav',))
t.daemon = True
t.start()

cv2.putText(frame, "DROWSINESS ALERT!", (10, 30),
                            cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)

print("Drowsy")
else:
            COUNTER = 0
            ALARM_ON = False
```

```
print(EAR)


cv2.imshow("Driver Drowsiness Detection System", frame)
key = cv2.waitKey(1)
if key == 27:
break
cap.release()
cv2.destroyAllWindows()
```

GSM.PY FILE

```
import serial
importos, time
importRPi.GPIO as GPIO
GPIO.setmode(GPIO.BOARD)
port = serial.Serial("/dev/ttyS0", baudrate=9600, timeout=1)
port.write(b'AT\r')
rcv = port.read(10)
print(rcv)
time.sleep(1)


port.write(b"AT+CMGF=1\r")
print("Text Mode Enabled…")
time.sleep(3)
```

port.write(b’AT+CMGS=”9166873301″\r’)

msg = “! EMERGENCY ALERT !

If you are receiving this message, then I might be in a medical emergency or casualty right now. Please track me with the below location tracker. Thank you.

print(“sending message….”)

time.sleep(3)

port.reset_output_buffer()

time.sleep(1)

port.write(str.encode(msg+chr(26)))

time.sleep(3)

print(“message sent…”)

GPS.PY FILE

import serial

from time import sleep

importwebbrowser

import sys

defGPS_Info():

globalNMEA_buff

globallat_in_degrees

globallong_in_degrees

nmea_time = []

nmea_latitude = []

```
nmea_longitude = []

nmea_time = NMEA_buff[0]

nmea_latitude =NMEA_buff[1]

nmea_longitude = NMEA_buff[3]

print("NMEA Time: ", nmea_time,'\n')

print ("NMEA Latitude:", nmea_latitude,"NMEA Longitude:", nmea_longitude,'\n')

lat = float(nmea_latitude)

longi = float(nmea_longitude)

lat_in_degrees = convert_to_degrees(lat)

long_in_degrees = convert_to_degrees(longi)

defconvert_to_degrees(raw_value):

decimal_value = raw_value/100.00

degrees = int(decimal_value)

mm_mmmm = (decimal_value - int(decimal_value))/0.6

position = degrees + mm_mmmm

position = "%.4f" %(position)

return position

gpgga_info = "$GPGGA,"ser = serial.Serial ("/dev/ttyS0")

GPGGA_buffer = 0

NMEA_buff = 0

lat_in_degrees = 0

long_in_degrees = 0

try:
```

```
while True:

received_data = (str)(ser.readline())

GPGGA_data_available = received_data.find(gpgga_info)

if (GPGGA_data_available>0):

GPGGA_buffer = received_data.split("$GPGGA,",1)[1]

NMEA_buff = (GPGGA_buffer.split(','))

GPS_Info()

print("lat in degrees:", lat_in_degrees," long in degree: ", long_in_degrees, '\n')

map_link = 'http://maps.google.com/?q=' + lat_in_degrees + ',' + long_in_degrees

print("<<<<<<<<press ctrl+c to plot location on google maps>>>>>>\n")

print("------------------------------------------------------------\n")

exceptKeyboardInterrupt:

webbrowser.open(map_link)

sys.exit(0)
```

REFERENCES

[1]Feng You Xiaolong li, Yunbo Gong Haiwei Wang, and HongyiL, A Real-time Driving Drowsiness Detection Algorithm With Individual Differences Consideration, IEEE ACCESS. 2019.2958667.

[2]Manu.B.N, Facial features monitoring for real time drowsiness detection, International conference on Innovation in Information Technology ,IIT 2016.

[3]M.AsidTanveer, M.Jawad Khan, M.Jahangir Qureshi, NomanNaseer, and KeumShik Hong, Enhanced Drowsiness Detection Using Deep Learning, IEE ACCESS.2019.2942838

[4]MikaSunagawa , Shin-ichiShikii, WataruNakai, Makoto Mochizuki, Koichi Kusukame, and Hiroki Kitajima, Comprehensive Drowsiness Level Detection Model Combining Multimodal Information, IEEE Sensors journal, vol 20, NO 7, April 1,2020.

[5]RatebJabbar, Khalifa Al-Khalifa, Mohamed Kharbeche, WaelAlhajyaseen, Mohsen Jafari, Shan Jiang, Real-time Driver Drowsiness Detection for Android Application Using Deep Neural Networks Techniques, 9th International Conference on Ambient Systems, Networks, and Technologies (ANT 2018).

[6]Mohamad-HoseynSigari, MahmoodFathy, Mohsen Soryani, A Driver Face Monitoring System for Fatigue and Distraction Detection, International Journal of Vehicular Technology Volume 2013, Article ID 263983.

[7]Wanghuadeng ,ruoxuewu, Real time driver drowsiness detection using facial features, ACCESS.2019.2936663.

[8]Shi Luo, Xiongfei Li, Rui Zhu, Xiaolizhang, Small Faces Attention face detector, IEEE ACCESS.2019.2955757.

[9]Wisaroot Tipprasert, Theekapun Charoenpong, Chamaporn Chianrabutra, Chamaiporn Sukjamsri*, A Method of Driver’s Eyes Closure and Yawning

Detection for Drowsiness Analysis by Infrared Camera, International Symposiumon Instrumentation, Control, Artificial Intelligence, and Robotics (ICA-SYMP) , 2019.

[10] B Padmaja, P V Narasimha Rao , M MadhuBala, E Krishna Rao Patro, A Novel Design of Autonomous Cars using IoT and Visual Features , IEEE Xplore Part Number:CFP18OZV-ART; ISBN:978-1-5386-1442-6.

[11] Ruslan T, YerdenK.a, and Md.Hazrat Ali, Development of Arduino Glove Based Autonomous Car, 21st International Conference on System Theory, Control and Computing (ICSTCC), 2017.

[12] Daniel ClaesBAstinBroecker. Development of an Autonomous Rc-car, Conference: International Conference on Intelligent Robotics and Applications, 2013.