**AUTOMATIC ACCELERATION CONTROL SYSTEM BASED ON**

**DRIVER FATIGUE**

A picture containing text

Description automatically generatedText

Description automatically generated

# A PROJECT REPORT

***Submitted by***

**VIGNESH. D**

**(421118104095)**

**SUBASH SURYA. E**

**(421118104084)**

***in partial fulfilment for the award of the degree***

***of***

**BACHELOR OF ENGINEERING**

**in**

**COMPUTER SCIENCE AND ENGINEERING**

**IFET COLLEGE OF ENGINEERING, VILLUPURAM – 605 108**

**(An Autonomous Institution)**

**ANNA UNIVERSITY: CHENNAI 600 025**

# APRIL 2022

**BONAFIDE CERTIFICATE**

Certified that this project report “**AUTOMATIC ACCELERATION CONTROL SYSTEM BASED ON DRIVER FATIGUE** “is the bonafide work of “VIGNESH.D**(421118104095), SUBASH SURYA.E (421118104084)**” who carried out the project work under my supervision.

# SIGNATURE SIGNATURE

**Mrs. P. KANIMOZHI (Ph.D.),** **Ms.P. DIVYA,**

**HEAD OF THE DEPARTMENT, SUPERVISOR,**

ASSOCIATE PROFESSOR, ASSISTANT PROFESSOR,

Department of Computer Science and Department of Computer Science and

Engineering, Engineering,

IFET College of Engineering, IFET College of Engineering,

Villupuram – 605 108. Villupuram – 605 108.

Submitted for the project viva voce examination held on……………………

# INTERNAL EXAMINER EXTERNAL EXAMINER

**ACKNOWLEDGEMENT**

I thank the almighty, for the blessings that have been showered upon me to bring forth the success of the project. I would like to express my sincere gratitude to our Chairman **Mr.K.V.Raja**, and our Secretary **Mr.K.Shivram Alva** for providing us with an excellent infrastructure and necessary resources to carry out this project and I extend my gratitude to our principal **Dr.G.Mahendran**, for his constant support to our work.

I also take this opportunity to express my sincere thanks to our Vice Principal and Dean Academics **Dr.S.Matilda** and our Head Placement and Corporate Affairs **Prof.S.Viswanathan**, who has provided all the needful help in executing the project successfully.

I wish to express my thanks to our Head of the Department **Dr.P.Kanimozhi**, Associate Professor for her persistent encouragement and support to complete this project. I express my heartfelt gratitude to our guide **Ms.P. Divya,** Assistant Professor, Department of Computer Science and Engineering for her priceless guidance and motivation which helped us to bring this project to a perfect shape.

And I thank our Faculty Advisor **Mr. Parthiban,** Assistant Professor, Department of Computer Science and Engineering who encouraged us in each and every step of this project to complete it successfully.

I express my deep sense of thanks to all faculty members and lab technicians in my department for their cooperation and interest shown at every stage of our endeavor in making a project work success.

Last but not the least, I whole heartedly thanks to our lovely parents and friends for their moral support in tough times and their constructive criticism which made me to succeed in our work.

**ABSTRACT**

Car accident is the major cause of death in which around 1.3 million people die every year. Majority of these accidents are caused because of distraction or the drowsiness of driver. Construction of high-speed highway roads had diminished the margin of error for the driver. The countless number of people drives for long distance every day and night on the highway. Lack of sleep may lead to an accident. Drowsiness and Fatigue of drivers are amongst the significant causes of road accidents. Every year, they increase the amounts of deaths and fatalities injuries globally. To prevent such accidents, we propose a system which alerts the driver if the driver feels drowsy. Facial landmarks detection is used with help of image processing of images of the face captured using the camera, for detection of drowsiness. In this project, a module for Advanced Driver Assistance System is presented to reduce the number of accidents due to drivers’ fatigue and hence increase the transportation safety; this system deals with automatic driver drowsiness detection based on visual information and Artificial Intelligence. This project proposes an algorithm to locate, track, and analyze both the drivers face and eyes to measure EAR (Eye Aspect Ratio), a scientifically supported measure of drowsiness associated with slow eye closure.  Thus, this project helps to save life of people as well as maintain road safety using AI.

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **CHAPTER NO** | **TITLE** | **PAGE NO.** |
|  | **ABSTRACT** | iii |
|  | **LIST OF FIGURES** | viii |
|  | **LIST OF SYMBOLS AND ABBREVIATIONS** | ix |
| **1** | **INTRODUCTION** | 1 |
|  | 1.1 GENERAL INTRODUCTION | 1 |
|  | 1.2 TECHNOLOGIES USED | 2 |
|  | 1.2.1 Embedded system | 2 |
|  | 1.2.1.1 Structure of embedded systems | 3 |
|  | 1.2.1.2 Benefits of using embedded systems | 4 |
|  | 1.3 Objectives of the Project | 4 |
|  | 1.4 Scope of the Project | 5 |
| **2** | **LITERATURE SURVEY** | 6 |
|  | 2.1 LITERATURE SURVEY | 6 |
|  | 2.1.1 Driving Behavior Analysis Guidelines for Intelligent Transportation Systems | 6 |
|  | 2.1.2 Blockchain-Enabled Certificate-Based Authentication for Vehicle Accident Detection and Notification in Intelligent Transportation Systems | 7 |
|  | 2.1.3 Machine Learning for Severity Classification of Accidents Involving Powered Two Wheelers | 8 |
|  | 2.1.4 Rear-End Collision Avoidance-Based on Multi-Channel Detection | 8 |
|  | 2.1.5 Collision Detection by Networked Sensors | 9 |
|  | 2.1.6 Deep Spatio-Temporal Representation for Detection of Road Accidents Using Stacked Autoencoder | 10 |
|  | 2.1.7 A Pythagorean-Type Fuzzy Deep Denoising Autoencoder for Industrial Accident Early Warning | 11 |
|  | 2.1.8 Inevitable Collision States for Motorcycle-to-Car Collision Scenarios | 12 |
|  | 2.1.9 Haptic Feedback to Assist Bus Drivers for Pedestrian Safety at Low Speed | 13 |
|  | 2.1.10 Frailty Models for the Estimation of Spatiotemporally Maximum Congested Impact Information on Freeway Accidents | 13 |
| **3** | **EXISTING SYSTEM** | 15 |
|  | 3.1 EXISTING SYSTEM | 15 |
|  | 3.2 DISADVANTAGES OF EXISTING SYSTEM | 16 |
| **4** | **PROPOSED SYSTEM** | 17 |
|  | 4.1 PROPOSED SYSTEM | 17 |
|  | 4.2 ADVANTAGES OF PROPOSED SYSTEM | 17 |
|  | 4.3 SYSTEM ARCHITECTURE | 18 |
|  | 4.4 WORKING | 18 |
| **5** | **SYSTEM ANALYSIS** | 20 |
|  | 5.1 MODULE DESCRIPTION | 20 |
|  | 5.1.1 Processing Module | 20 |
|  | 5.1.2 Eye Aspect Ratio (EAR) | 23 |
|  | 5.1.3 Motor Controlling Module | 24 |
| **6** | **HARDWARE AND SOFTWARE DESCRIPTION** | 27 |
|  | 6.1 HARDWARE DESCRIPTION | 27 |
|  | 6.1.1 Raspberry Pi | 27 |
|  | 6.1.2 Arduino | 28 |
|  | 6.1.3 Power Supply | 29 |
|  | 6.1.4 AVR Microcontroller | 30 |
|  | 6.1.5 DC Motor | 32 |
|  | 6.1.6 Relay | 33 |
|  | 6.1.7 Buzzer | 35 |
|  | 6.1.8 Camera | 37 |
|  | 6.2 SOFTWARE REQUIREMENTS | 39 |
|  | 6.2.1 Python | 39 |
| **7** | **RESULTS AND DISCUSSION** | 40 |
|  | 7.1 IMPLEMENTATION | 40 |
|  | 7.2 RESULTS OBTAINED | 40 |
| **8** | **CONCLUSION AND FUTURE WORK** | 44 |
|  | 8.1 CONCLUSION | 44 |
|  | 8.2 FUTURE WORK | 44 |
|  | APPENDIX-I | 45 |
|  | APPENDIX-II | 56 |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **FIGURE NO** | **NAME OF THE FIGURE** | **PAGE NO.** |
| 1.1 | Structure of embedded systems | 4 |
| 4.1 | Proposed System Architecture | 18 |
| 5.1 | Eye Detection | 24 |
| 5.2 | Flemings Left Hand Rule | 26 |
| 6.1 | Raspberry Pi | 27 |
| 6.2 | Power Supply | 30 |
| 6.3 | Microcontroller ATmega328 | 31 |
| 6.4 | DC Motor | 32 |
| 6.5 | Relay | 33 |
| 6.6 | Buzzer | 36 |
| 6.6 | USB Camera | 38 |
| 7.1 | Complete Hardware Kit | 40 |
| 7.2 | Running the code | 41 |
| 7.3 | Drowsiness Detection | 42 |
| 7.4 | Hardware Kit with DC Motor Running | 42 |
| 7.5 | DC Motor Off and Buzzer On | 43 |

**LIST OF SYMBOLS AND ABBREVIATIONS**

|  |  |
| --- | --- |
| IoT | Internet of Things |
| LSTM | Long Short-Term Memory |
| DSP | Digital signal processors |
| ASIC | Application-specific integrated circuits |
| FPGA | Field-programmable gate arrays |
| EAR | Eye Aspect Ratio |

**CHAPTER 1**

**INTRODUCTION**

* 1. **GENERAL INTRODUCTION:**

Drowsy driving is a major contributor to motor vehicle collisions. According to the National Highway Traffic Safety Administration (NHTSA), in 2017 drowsy driving led to at least 91,000 crashes, resulting in roughly 50,000 injuries and 800 deaths. This data likely underestimates the impact of drowsy driving because it’s often impossible to definitively determine whether drowsy driving caused an accident, especially after fatal crashes. In light of this, other studies calculate that drowsy driving causes up to 6,000 deadly crashes every year. Researchers estimate that around 21% of fatal car crashes involve a person driving while drowsy.

Drowsy driving significantly increases the risk of car accidents. Microsleeps are when a person [dozes off for just a few seconds](https://pubmed.ncbi.nlm.nih.gov/23008180/) and when they occur while driving, it’s easy for the car to run off the road or collide with another vehicle. The damage from these crashes increases when they occur at high speeds.

Drowsy driving is dangerous even if a person doesn’t actually fall asleep. Research shows that [sleep deprivation](https://www.sleepfoundation.org/sleep-deprivation) leads to mental impairment that is [similar to drunkenness](https://pubmed.ncbi.nlm.nih.gov/9230429/) with 24 hours of sleep deprivation roughly equating to a blood alcohol content (BAC) of 0.10%.

This impairment makes a person less attentive to their surroundings and more easily distracted. It slows their reaction time, making it harder to avoid dangers in the roadway. Insufficient sleep is also tied to worsened decision-making, which can lead to risk-taking behind the wheel.

Over the long-term, good sleep is the best protection against drowsy driving. Focusing on sleep hygiene, which includes your habits and sleep setting, can enable better sleep every night. Examples of sleep hygiene include maintaining a stable sleep schedule, limiting the use of electronic devices before bed, and making sure that your bedroom is quiet, dark, and conducive to uninterrupted rest. In addition to sleep hygiene improvements, you should talk with a doctor if you have persistent or severe problems with falling or staying asleep or if you regularly have daytime sleepiness. Working with your doctor can identify the optimal approach to enhancing your sleep, which may involve testing to determine if you are affected by an underlying sleep disorder.

* 1. **TECHNOLOGIES USED:**

**1.2.1 EMBEDDED SYSTEM**

An embedded system is a microprocessor-based computer hardware system with software that is designed to perform a dedicated function, either as an independent system or as a part of a large system. At the core is an integrated circuit designed to carry out computation for real-time operations.

Complexities range from a single microcontroller to a suite of processors with connected peripherals and networks; from no user interface to complex graphical user interfaces. The complexity of an embedded system varies significantly depending on the task for which it is designed.

Embedded system applications range from digital watches and microwaves to hybrid vehicles and avionics. As much as 98 percent of all microprocessors manufactured are used in embedded systems.

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.

**1.2.1.1 Structure of embedded systems**

The basic structure of an embedded system includes the following components:

Sensor: The sensor measures and converts the physical quantity to an electrical signal, which can then be read by an embedded systems engineer or any electronic instrument. 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 digital signal.

**Processor & ASICs:** Processors assess the data to measure the output and store it to the memory.

**D-A Converter:** A digital-to-analog converter changes the digital data fed by the processor to analog data

**Actuator:** An actuator compares the output given by the D-A Converter to the actual output stored and stores the approved output.

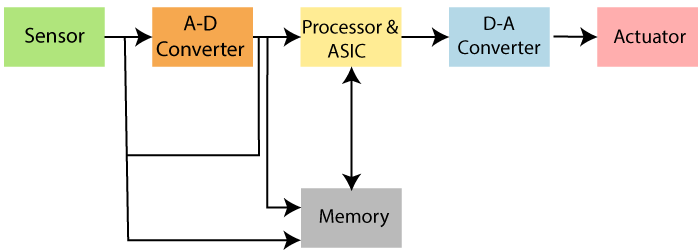


Figure 1.1 Structure of embedded systems

**1.2.1.2 Benefits of using embedded systems**

* Small size and faster to load
* More specific to one task
* Easy to manage
* Low cost
* Spend less resources
* This operating system is dedicated to one device so performance is good and use less resources like memory and micro-processors.

**1.3 OBJECTIVES OF THE PROJECT:**

* The objective of the project is to detect the drowsy drivers using EAR
* To alert the driver when any abnormal condition occurs such as long closure of eyes
* To save the life of the driver as well as the passengers
* To save property
* To promote an accident-free road and safe transport system

**1.4 SCOPE OF THE PROJECT:**

* Can be implemented in vehicle as safety measure
* Can be implemented in all vehicles

**CHAPTER - 2**

**LITERATURE SURVEY**

**2.1 LITERATURE SURVEY:**

**2.1.1 Driving Behaviour Analysis Guidelines for Intelligent Transportation Systems**

The advent of in-vehicle networking systems as well as state-of-the-art sensors and communication technologies have facilitated the collection of large volume and almost real-time data on vehicles and drivers, thus opening up future possibilities. Processing and analysing this data provides unprecedented opportunities to offer remarkable insights and solutions for driving behaviour analysis (DBA). Characterizing driving behaviour plays a key role in a variety of research areas such as traffic safety, the development of automated vehicles, energy and fuel management, risk assessment, and driver identification and profiling. Advances in DBA-based driver inattention or drunk driver detection can help reduce fatal car crashes, and understanding the driving style (e.g., eco-friendly or aggressive) of drivers can contribute to fuel management and risk assessment of the drivers. These facts have led to a growing interest in addressing DBA challenges. This paper aims to present the state-of-the-art methodologies for DBA and provide a clear roadmap about the main current and future trends in DBA. To this end, we propose categorizing the current research on driving behaviour based on the types of data employed for the analysis, the ultimate goals of the analysis, and the techniques based on which the driving data are modelled. We provide an overview of different data resources and available datasets for DBA. Moreover, we discuss the application of DBA along with the key research challenges in this field and potential future directions.

**Pros:** This technique have great potential for alerting the drivers

**Cons:** This system follows very complex architecture

**2.1.2** **Frailty Models for the Estimation of Spatiotemporally Maximum Congested Impact Information on Freeway Accidents**

The objective of this paper is to develop models for the estimation of the temporal and spatial extent of congestion impact caused by accidents. Although there have been various approaches based on the deterministic queuing diagrams and kinematic wave (or shockwave) theory, only a few studies have been able to estimate the spatiotemporal congested region based on field data, such as ubiquitous loop detector data. Accordingly, this paper applies a previously developed procedure to capture the spatiotemporal accident impacts based on binary integer programming (BIP). The procedure provides a foundation for models of the following: 1) maximum spatial distance to the end of the congestion region affected by each accident and 2) maximum time affected by congestion resulting from each accident. Based on these models, the objective of this paper is to estimate two statistical models for providing maximum congested distance and time information due to freeway accidents. Since various observations from BIP were censored with respect to time and space, survival analysis—specifically, frailty models to account for unobserved heterogeneity—is applied to identify factors critical to spatiotemporal congestion impacts of freeway accidents.

**Pros:** Since various observations from BIP were censored with respect to time and space, survival analysis—specifically, frailty models to account for unobserved heterogeneity—is applied to identify factors critical to spatiotemporal congestion impacts of freeway accidents.

**Cons:** The system is inaccurate to a certain making it, not recommended for real time applications.

**2.1.3** **Near Real–Time Freeway Accident Detection**

In this paper, the problem of detecting accidents using speed sensors distributed spatially on a freeway is considered. Due to the significant impact of road accidents on health and development, early and accurate detection of accidents is crucial. To address this issue, a novel Bayesian quickest change detection formulation is introduced, which considers both average detection delay and false alarm rate. The optimum strategy is derived via dynamic programming and shown to compare a recursively computed statistic with a function of false alarm to identify accidents as they happen. Considering that post–accident conditions are typically not known, two methods are proposed that recursively estimate multiple unknown parameters during the accident detection process. Further, four aggregation methods are proposed to improve performance exploiting spatial correlations between sensors. Extensive evaluation results demonstrate improvement up to 65.2% and 87.2% in average detection delay and false alarm rate, respectively, against prior work.

**Pros:** Alerts the pedestrian effectively by signalling him/her with alerts or noises.

**Cons:** The project makes use of noise as alert the pedestrian of the car approaching, this may lead to unwanted noise pollution.

**2.1.4** **Inevitable Collision States for Motorcycle-to-Car Collision Scenarios**

This paper presents a method to identify inevitable collision states (ICS) specifically for a motorcycle when interacting with an opponent passenger car in typical traffic scenarios. Previous ICS methods were applied to passenger cars or generic vehicles; however, the peculiarities of motorcycles urge the definition of specific methods for these vehicles. The findings extend the applicability of previous algorithms to include all motorcycle-to-car collisions, irrespective of collision configurations. ICS identification can be adopted as a triggering criterion for more invasive safety systems such as motorcycle autonomous emergency braking (MAEB), which require a last-resort approach in their initial phases of development. In this regard, this paper also presents an evaluation of an idealized MAEB through experiments simulating real-world crashes in a computer-based virtual environment.

**Pros:** Produces very good accuracy compared to other state-of-the-art methods.**Cons:** MAEB through experiments simulating real-world crashes in a computer-based virtual environment.

**2.1.5** **Rear-End Collision Avoidance-Based on Multi-Channel Detection**

With multi-sensor-based collision avoidance systems (CASs) being adopted in today’s automobiles, a new method that enables collaborative decision-making with preceding vehicle detection under various external environments is needed. In this paper, spatial–temporal correlations of multi-channel signals that are collected by multiple sensors on the host vehicle are considered, and a multi-channel detection technique with a stochastic model is introduced for automobile collision avoidance. We propose an accurate and robust multi-channel, generalized likelihood ratio test (GLRT)-based detection and collaborative decision-making scheme, with a vehicle kinematic analysis for avoiding rear-end collisions. The results of simulations and physical experiments demonstrated that our detector expands the detection range with a high detection rate and that our proposed scheme obtains good performance under varying operating and environmental conditions.

**Pros:** The results of simulations and physical experiments demonstrated that the detector expands the detection range with a high detection rate.

**Cons:** The system is too expensive to be implemented for real time applications.

**2.1.6** **Blockchain-Enabled Certificate-Based Authentication for Vehicle Accident Detection and Notification in Intelligent Transportation Systems**

As the communications among the vehicles, the Road-Side Units (RSU) and the Edge Servers (ES) take place via wireless communication and the Internet, an adversary may take the opportunity to tamper with the data communicated among various entities in an Internet of Vehicles (IoV) environment. Therefore, it demands secure communication among the involved entities in an IoV-based Intelligent Transportation System (ITS) deployment. In this work, we design a new blockchain-enabled certificate-based authentication scheme for vehicle accident detection and notification in ITS, called BCASVADN. In BCAS-VADN, through the authentication process, each vehicle can securely notify accident-related transactions to its nearby Cluster Head (CH), if an accident is detected on roads either by its own or neighbour vehicle(s). The CH then securely sends the transactions received from the vehicles to its RSU and subsequently, these transactions are also received secretly by the ESs. The ES is responsible for preparing partial block containing transactions and Merkle tree root, and a digital signature on that information, and then forwarding to its associated Cloud Server (CS) in the Blockchain Centre (BC) for complete block creation, verification and addition of the block using the designed consensus process. Due to blockchain technology usage, it is shown that BCAS-VADN is not only secure against various potential attacks, but also maintains transparency, immutability and decentralization of the information. Furthermore, a comprehensive comparative analysis reveals that BCAS-VADN achieves better security and more functionality attributes, and has low communication and computational overheads as compared to other competitive authentication schemes in IoV. In addition, the practical demonstration using the blockchain technology has been also provided.

**Pros:** The practical demonstration using the blockchain technology has been also provided, giving a very good recommendation for real time application.

**Cons:** The system is very slow in comparison to the other state-of- the art methods.

**2.1.7** **Collision Detection by Networked Sensors**

The advances in sensor technologies enable real-time collection of high-fidelity spatiotemporal data on transportation networks of major cities. We consider a set of speed sensors that are spatially distributed along a street and can communicate via an exogenously determined network. In this paper, we address the problem of detecting in real-time collisions the at occur within a certain distance from each sensor. The speed sensors observe the average speed value of the cars at regular time intervals and adopt a threshold-based approach to generate local predictions. Each sensor exchanges its local predictions with its neighbours and aggregates the local predictions it receives using a weighted majority aggregation rule to generate a final prediction. Since collisions are eventually reported (e.g., by a police officer or by crowd-sourced information), we assume that the information about the real occurrence of a collision is eventually given to the sensors. We propose an online learning rule that exploits this feedback to adapt the weights that each sensor gives to different local predictions. In the realizable case, i.e., when there exist unknown weights that would allow the sensors to distinguish between collisions and normal traffic behaviours, we determine an upper bound for the worst-case misdetection and false alarm probabilities of our scheme. We evaluate our scheme with traffic datasets collected from the segment of the 405 freeway that passes through Los Angeles County and the results show the efficacy of the proposed approach.

**Pros:** On evaluating the scheme with traffic datasets collected from the segment of the 405 freeway that passes through Los Angeles County and the results show the efficacy of the proposed approach.

**Cons:** The system makes use of the sensors to detect cars making it expensive and tiresome.**2.1.8** **Haptic Feedback to Assist Bus Drivers for Pedestrian Safety at Low Speed**

Buses and coaches are massive Passenger Transportation Systems (PTS), because they represent more than half of land PTS in the European Union. Despite of that, bus accident figures are lower than other means of transport, but its size and weight increase the severity of accidents in which buses are involved, even at low speed. In urban scenarios, turnings and manoeuvres around bus stops are the main causes of accidents, mostly due to low visibility, blind spots or driver’s distractions. Therefore, there is an increasing interest in developing driving assistance systems to avoid these situations, among others. However, even though there are some solutions on the market, they are not meant to work in urban areas at low speed and with the sole purpose of preventing collisions with pedestrians. In this sense, the paper proposes an active safety system for buses in manoeuvres at low speed. The safety system consists of haptic feedback devices together with collision avoidance and risk evaluation systems based on detected people nearby the bus. The performance of the active safety system has been validated in a simulated urban scenario. Our results show that driver’s reaction time is reduced and time to collision increased due to the proposed low-speed active safety system. In particular, it is shown that there is a reduction in the number of high-risk cases and collisions, which implies a considerable improvement in safety terms. In addition to this, a brief discussion about current regulations for innovative safety systems on a real vehicle is carried out.

**Pros:** The results show that driver’s reaction time is reduced and time to collision increased due to the proposed low-speed active safety system.

**Cons:** This project is very poor in accuracy, making it to be not useful in the case of real time applications.**2.1.9** **Machine Learning for Severity Classification of Accidents Involving Powered Two Wheelers**

Road traffic safety is one of the major challenges for the future of smart cities and transportation networks. Despite several solutions exist to reduce the number of fatalities and severe accidents happening daily in our roads, this reduction is smaller than expected and new methods and intelligent systems are needed. The emergency Call is an initiative of the European Commission aimed at providing rapid assistance to motorists thanks to the implementation of a unique emergency number. In this work, we study the problem of classifying the severity of accidents involving Powered Two Wheelers, by exploiting machine learning systems based on features that could be reasonably collected at the moment of the accident. An extended study on the set of features allows to identify the most important factors that enable to distinguish accident severity. The system we develop achieves around 90% of precision and recall on a large, publicly available corpus, using only a set of eleven features.

**Pros:** Produces a precision of 90%, hence is recommended for general purpose.

**Cons:** The project is primarily focused on preventing accidents from occurring in case of two wheelers.

**2.1.10** **A Pythagorean-Type Fuzzy Deep Denoising Autoencoder for Industrial Accident Early Warning**

Early warning is crucial for preventing industrial accidents and mitigating damage, but current methods are often time-consuming, error-prone, and incompetent to deal with uncertainty. The paper presents a fuzzy deep neural network for early warning of industrial accidents, which equips the classical deep denoising autoencoder (DDAE) model with Pythagorean type fuzzy parameters in order to enhance the model’s representation ability and robustness. To efficiently train the fuzzy deep model, we propose a hybrid algorithm combining Hessian-free (HF) optimization and biogeography-based optimization (BBO) metaheuristic to balance global search and local search. Experiments on datasets from several industrial zones in China show that the proposed Pythagorean-type fuzzy DDAE (PFDDAE) can achieve much higher accuracy of accident risk classification than the classical DDAE and the fuzzy DDAE using regular fuzzy parameters, and the proposed hybrid learning algorithm exhibits significant performance advantage over some other learning algorithms in training PFDDAE. In particular, a test on the 2014 Kunshan aluminium dust explosion accident shows that the deep learning model would be very likely to prevent the accident if it was adopted in advance.

**Pros:** Studyshows that the deep learning model would be very likely to prevent the accident if it was adopted in advance.

**Cons:** There is a certain degree of assumptions being made in this project making it worthless.

**CHAPTER 3**

**EXISTING SYSTEM**

**3.1 EXISTING SYSTEM**

The advent of in-vehicle networking systems as well as state-of-the-art sensors and communication technologies have facilitated the collection of large volume and almost real-time data on vehicles and drivers, thus opening up future possibilities. Processing and analysing this data provide unprecedented opportunities to offer remarkable insights and solutions for driving behaviour analysis (DBA). Characterizing driving behaviour plays a key role in a variety of research areas such as traffic safety, the development of automated vehicles, energy and fuel management, risk assessment, and driver identification and profiling. Advances in DBA-based driver inattention or drunk driver detection can help reduce fatal car crashes, and understanding the driving style (e.g., eco-friendly or aggressive) of drivers can contribute to fuel management and risk assessment of the drivers. These facts have led to a growing interest in addressing DBA challenges. This paper aims to present the state-of-the-art methodologies for DBA and provide a clear roadmap about the main current and future trends in DBA. To this end, we propose categorizing the current research on driving behaviour based on the types of data employed for the analysis, the ultimate goals of the analysis, and the techniques based on which the driving data are modelled. We provide an overview of different data resources and available datasets for DBA. Moreover, we discuss the application of DBA along with the key research challenges in this field and potential future directions.

**3.2 DISADVANTAGES OF EXISTING SYSTEM**

* Follows very complex architecture.
* Various uncertain factors create a major impact

**CHAPTER - 4**

**PROPOSED SYSTEM**

**4.1 PROPOSED SYSTEM**

Car accident is the major cause of death in which around 1.3 million people die every year. Majority of these accidents are caused because of distraction or the drowsiness of driver. Construction of high-speed highway roads had diminished the margin of error for the driver. The countless number of people drives for long distance every day and night on the highway. Lack of sleep may lead to an accident. Drowsiness and Fatigue of drivers are amongst the significant causes of road accidents. Every year, they increase the amounts of deaths and fatalities injuries globally. To prevent such accidents, we propose a system which alerts the driver if the driver feels drowsy. Facial landmarks detection is used with help of image processing of images of the face captured using the camera, for detection of drowsiness. In this project, a module for Advanced Driver Assistance System is presented to reduce the number of accidents due to drivers’ fatigue and hence increase the transportation safety; this system deals with automatic driver drowsiness detection based on visual information and Artificial Intelligence. This project proposes an algorithm to locate, track, and analyze both the drivers face and eyes to measure EAR (Eye Aspect Ratio), a scientifically supported measure of drowsiness associated with slow eye closure. Thus, this project helps to save life of people as well as maintain road safety using AI.

**4.2 ADVANTANGES OF PROPOSED SYSTEM**

* To alert the driver when any abnormal condition occurs such as long closure of eyes or continuous yawn detection
* To save the life of the driver as well as the passengers
* To save property
* To promote an accident-free road and safe transport system
* Can be implemented in all vehicles

**4.3 SYSTEM ARCHITECTURE**

**Diagram

Description automatically generated**

Figure 4.1 Proposed System block diagram

**4.4 WORKING**

Raspberry Pi to control the whole system in this project. Initially the raspberry pi is turned on to initiate the whole process. Once the Pi is turned on it will access the camera to capture the driver’s face, then this captured image will be processed to locate, track, and analyze both the drivers face and eyes to measure EAR (Eye Aspect Ratio), a scientifically supported measure of drowsiness associated with slow eye closure. After processing the image, it will check for any abnormal activity, if any such activity is found then the buzzer will be turned on to alert the driver and automatically speed of the vehicle will be reduced by controlling the accelerometer. Using this we can reduce the number of accidents due to drivers’ fatigue and hence increase the transportation safety; this system deals with automatic driver drowsiness detection based on visual information and Artificial Intelligence. This project proposes an algorithm to locate, track, and analyze both the drivers face and eyes. Thus, this project helps to save life of people as well as maintain road safety using AI.

**CHAPTER – 5**

**SYSTEM ANALYSIS**

**5.1 MODULE DESCRIPTION:**

* Processing Module
* Eye Aspect Ratio
* Motor Controlling Module

**5.1.1 Processing Module**

The Raspberry Pi 3 Model B is the latest version of the $35 Raspberry Pi computer. The Pi isn't like your typical machine, in its cheapest form it doesn't have a case, and is simply a credit-card sized electronic board -- of the type you might find inside a PC or laptop but much smaller.

The Raspberry Pi device looks like a motherboard, with the mounted chips and ports exposed (something you'd expect to see only if you opened up your computer and looked at its internal boards), but it has all the components you need to connect input, output, and storage devices and start computing.

Raspberry Pi is a low-cost, basic computer that was originally intended to help spur interest in computing among school-aged children. The Raspberry Pi is contained on a single circuit board and features ports for:

* HDMI
* USB 2.0
* Composite video
* Analog audio
* Power
* SD Card

The computer runs entirely on open-source software and gives students the ability to mix and match software according to the work they wish to do.

The Raspberry Pi debuted in February 2012. The group behind the computer's development - the Raspberry Pi Foundation - started the project to make computing fun for students, while also creating interest in how computers work at a basic level. Unlike using an encased computer from a manufacturer, the Raspberry Pi shows the essential guts behind the plastic.

The quad-core Raspberry Pi 3 is both faster and more capable than its predecessor, the Raspberry Pi 2. For those interested in benchmarks, the Pi 3's CPU--the board's main processor--has roughly 50-60 percent better performance in 32-bit mode than that of the Pi 2, and is 10x faster than the original single-core Raspberry Pi (based on a multi-threaded CPU benchmark in SysBench). Compared to the original Pi, real-world applications will see a performance increase of between 2.5x--for single-threaded applications--and more than 20x--when video playback is accelerated by the chip's NEON engine.

Unlike its predecessor, the new board is capable of playing 1080p MP4 video at 60 frames per second (with a bit rate of about 5400Kbps), boosting the Pi's media centre credentials. That's not to say, however, that all video will playback this smoothly, with performance dependent on the source video, the player used and bitrate.

The Pi 3 also supports wireless internet out of the box, with built-in Wi-Fi and Bluetooth.

The latest board can also boot directly from a USB-attached hard drive or pen drive, as well as supporting booting from a network-attached file system, using PXE, which is useful for remotely updating a Pi and for sharing an operating system image between multiple machines.

The latest version of the Raspberry Pi's official OS has the Chromium browser, the open-source browser that Chrome is based on. Its performance is reasonable, as long as you don't open too many script-laden websites, and there are extensions that allow for smooth playback of video on YouTube and other sites. There are various options if you want to use the Pi 3 as a media centre but the most popular choices are the Kodi-based OSes OSMC or Libre Elec.

The Pi 3 has the added advantage of a slightly faster graphics processor, which the Raspberry Pi Foundation has said is able to play local H.264-encoded video recorded at 1920x1080 resolution and 60 frames per second. Another advantage is built-in support for Wi-Fi makes it easier to stream content to the Pi, while native Bluetooth simplifies the hooking up peripherals.

A wide range of vintage games will run on the Pi with the help of emulators like RetroPie, including some games from all of the systems listed above, although the more recent the system, the more likely it is that more demanding titles will struggle.

The Pi can run the official Raspbian OS, Ubuntu Mate, Snappy Ubuntu Core, the Kodi-based media centres OSMC and Libre Elec, the non-Linux based Risk OS (one for fans of 1990s Acorn computers). It can also run Windows 10 IoT Core, which is very different to the desktop version of Windows, as mentioned below.

However, these are just the officially recommended operating systems, and a large array of other weird and wonderful OSes also work on the Pi.

It's nothing like the full desktop version of Windows 10 that most people are familiar with. Instead, the Pi 3 runs Windows 10 IoT Core, a cutdown version of Windows 10 that doesn't boot into the graphical desktop and is designed to controlled via a command line interface on a remote computer. It can only run a single Fullscreen Universal Windows Platform app at a time, for example a kiosk app for a retail store, although other software can run in the background.

However, the Pi can act as a Windows 10 thin client, where Windows 10 is run on a server and streamed to the Pi and, with a powerful enough server, the experience can be virtually identical to running a Windows 10 machine.

The Pi 3 can run Windows desktop apps, although it requires you to buy the ExaGear Desktop software and spend some time setting it up.

Performance is also poor, with the tools needed to run Windows apps on the Pi sucking up so much processing power that you're basically restricted to running 20-year-old Windows apps and games, and simple modern text editors.

It can run Ubuntu with various desktops, with the Raspberry Pi Foundation highlighting Ubuntu Mate and Ubuntu Snappy Core as standouts.

**5.1.2 Eye Aspect Ratio (EAR)**

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

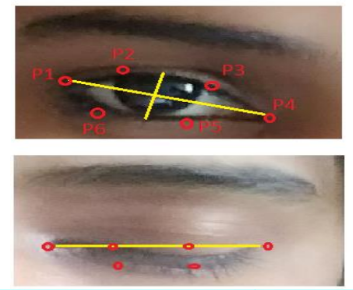
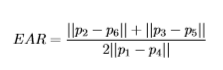


Figure 5.1: Eye Detection

The EAR is calculated for several frames of a video. A single blink is represented.



Where p1…, p6 are the two-dimensional landmark location, represented in Fig.2. The EAR is mostly stable when an eye is open and is getting close to zero while the eye is not in open state. If the person viewing the camera continuously, the Eye Aspect Ratio (EAR) is found to be normal and it reaches low value when he/she closing the eye for a longer time. When the lower value is reached, then drowsiness is detected.

**5.1.3 Motor Controlling Module**

An Electric DC motor is a machine which converts electric energy into mechanical energy. The working of DC motor is based on the principle that when a current-carrying conductor is placed in a magnetic field, it experiences a force.   
The direction of mechanical force is given by **Fleming’s Left-hand Rule** and its magnitude is given by **F = BIL Newton**.

There is no basic difference in the [construction of a DC generator](http://www.studyelectrical.com/2014/06/construction-dc-motor-dc-motor-construction.html) and a [DC motor](http://www.studyelectrical.com/2014/06/construction-dc-motor-dc-motor-construction.html). In fact, the same D.C. machine can be used interchangeably as a generator or as a motor. Like generators DC motors are also classified in to [shunt-wound](http://www.studyelectrical.com/2013/11/speed-control-of-dc-shunt-motors.html), [series-wound](http://www.studyelectrical.com/2013/11/speed-control-of-dc-series-motors.html),compound-wound.

DC motors are seldom used in ordinary applications because all electric supply companies furnish alternating current. However, for special applications such as in steel mills, mines and electric trains, it is advantageous to convert alternating current into direct current in order to use dc motors. The reason is that speed/torque characteristics of d.c. motors are much more superior to that of a.c. motors. Therefore, it is not surprising to note that for industrial drives, d.c. motors are as popular as 3-phase induction motors.

A machine that converts DC power into mechanical power is known as a DC motor.

Its operation is based on the principle that when a current carrying conductor is placed in a magnetic field, the conductor experiences a mechanical force.  As the drowsiness is detected, a command is sent to the hardware kit consisting of a DC motor running. The direction of this force is given by **Fleming’s left-hand rule** and magnitude is given by,

**F = BIL Newtons**

|  |
| --- |
| https://1.bp.blogspot.com/-UD2kOszb9m8/Wl9GjbtfOAI/AAAAAAAAA5g/0fH3Qf8URd85_rPPFjGozDYjktW_rNeWACLcBGAs/s1600/dcmfor.gif |
| Figure 5.2: Flemings Left Hand Rule |

SPECIFICATION:

Working voltage: 12

Maximum current:1.2

Power: 14.4

**CHAPTER – 6**

**HARDWARE AND SOFTWARE DESCRIPTION**

**6.1 HARDWARE REQUIREMENT**

The purpose of the Software Requirement Specification is to produce the specification of the analysis task and also to establish complete information about the requirement, behavior and also the other constraint like functional performance and so on.

**6.1.1 RASPBERRY PI**

The Raspberry Pi 3 Model B is the latest version of the $35 Raspberry Pi computer. The Pi isn't like your typical machine, in its cheapest form it doesn't have a case, and is simply a credit-card sized electronic board -- of the type you might find inside a PC or laptop but much smaller.

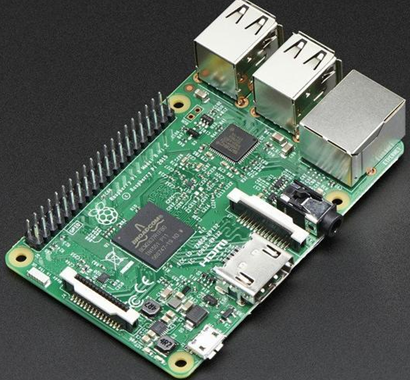


Figure 6.1: Raspberry Pi

The Raspberry Pi device looks like a motherboard, with the mounted chips and ports exposed (something you'd expect to see only if you opened up your computer and looked at its internal boards), but it has all the components you need to connect input, output, and storage devices and start computing.

The Raspberry Pi is believed to be an ideal learning tool, in that it is cheap to make, easy to replace and needs only a keyboard and a TV to run. These same strengths also make it an ideal product to jumpstart computing in the developing world.

**6.1.2 ARDUINO**

The Arduino software is easy-to-use for beginners, yet flexible enough for advanced users. It runs on Mac, Windows, and Linux. Teachers and students use it to build low-cost scientific instruments, to prove chemistry and physics principles, or to get started with programming and robotics. Designers and architects build interactive prototypes, musicians and artists use it for installations and to experiment with new musical instruments. Makers, of course, use it to build many of the projects exhibited at the Maker Faire, for example. Arduino is a key tool to learn new things. Anyone - children, hobbyists, artists, programmers - can start tinkering just following the step-by-step instructions of a kit, or sharing ideas online with other members of the Arduino community.

There are many other microcontrollers and microcontroller platforms available for physical computing. Parallax Basic Stamp, Net media’s BX-24, Phi gets, MIT's Handy board, and many others offer similar functionality. All of these tools take the messy details of microcontroller programming and wrap it up in an easy-to-use package. Arduino also simplifies the process of working with microcontrollers, but it offers some advantage for teachers, students, and interested amateurs over other systems:

* Inexpensive - Arduino boards are relatively inexpensive compared to other microcontroller platforms. The least expensive version of the Arduino module can be assembled by hand, and even the pre-assembled Arduino modules cost less than $50
* Cross-platform - The Arduino Software (IDE) runs on Windows, Macintosh OSX, and Linux operating systems. Most microcontroller systems are limited to Windows.
* Simple, clear programming environment - The Arduino Software (IDE) is easy-to-use for beginners, yet flexible enough for advanced users to take advantage of as well. For teachers, it's conveniently based on the Processing programming environment, so students learning to program in that environment will be familiar with how the Arduino IDE works.
* Open source and extensible software - The Arduino software is published as open-source tools, available for extension by experienced programmers. The language can be expanded through C++ libraries, and people wanting to understand the technical details can make the leap from Arduino to the AVR C programming language on which it's based. Similarly, you can add AVR-C code directly into your Arduino programs if you want to.
* Open source and extensible hardware - The plans of the Arduino boards are published under a Creative Commons license, so experienced circuit designers can make their own version of the module, extending it and improving it. Even relatively inexperienced users can build the breadboard version of the module in order to understand how it works and save money.

**6.1.3 Power Supply**

The power supply circuit consists of step-down transformer which is 230v step down to 12v.In this circuit 4diodes are used to form bridge rectifier which delivers pulsating dc voltage & then fed to capacitor filter the output voltage from rectifier is fed to filter to eliminate any A.C components present even after rectification.

The filtered DC voltage is given to regulator to produce 12v constant DC voltage. 230V AC power is converted into 12V AC (12V RMS value wherein the peak value is around 17V), but the required power is 5V DC; for this purpose, 17V AC power must be primarily converted into DC power then it can be stepped down to the 5V DC. AC power can be converted into DC using one of the power electronic converters called as Rectifier. There are different types of rectifiers, such as half-wave rectifier, full-wave rectifier and bridge rectifier. Due to the advantages of the bridge rectifier over the half and full wave rectifier is frequently used for converting AC to DC.

Diagram, schematic

Description automatically generated

Figure 6.2 Power Supply

**6.1.4 AVR Microcontroller**

AVR is a family of microcontrollers developed by Atmel beginning in 1996. These are modified Harvard architecture 8-bit RISC single-chip microcontrollers. AVR was one of the first microcontroller families to use on-chip flash memory for program storage, as opposed to one-time programmable ROM, EPROM, or EEPROM used by other microcontrollers at the time.

The AVR architecture was conceived by two students at the Norwegian Institute of Technology (NTH), Alf-Egil Bogen and Vegard Wollan.

The original AVR MCU was developed at a local ASIC house in Trondheim, Norway, called Nordic VLSI at the time, now Nordic Semiconductor, where Bogen and Wollan were working as students. It was known as a μRISC (Micro RISC) and was available as silicon IP/building block from Nordic VLSI. When the technology was sold to Atmel from Nordic VLSI, the internal architecture was further developed by Bogen and Wollan at Atmel Norway, a subsidiary of Atmel. The designers worked closely with compiler writers at IAR Systems to ensure that the AVR instruction set provided efficient compilation of high-level languages

A picture containing timeline

Description automatically generated

Figure 6.3 Microcontroller ATmega328

Atmel says that the name AVR is not an acronym and does not stand for anything in particular. The creators of the AVR give no definitive answer as to what the term "AVR" stands for However, it is commonly accepted that AVR stands for Alf and Vegard's RISC processor, note that the use of "AVR" in this article generally refers to the 8-bit RISC line of Atmel AVR Microcontrollers.

Among the first of the AVR line was the AT90S8515, which in a 40-pin DIP package has the same pin out as an 8051 microcontroller, including the external multiplexed address and data bus.

**6.1.5 DC Motor**

An DC motor is a machine which converts electric energy into mechanical energy.

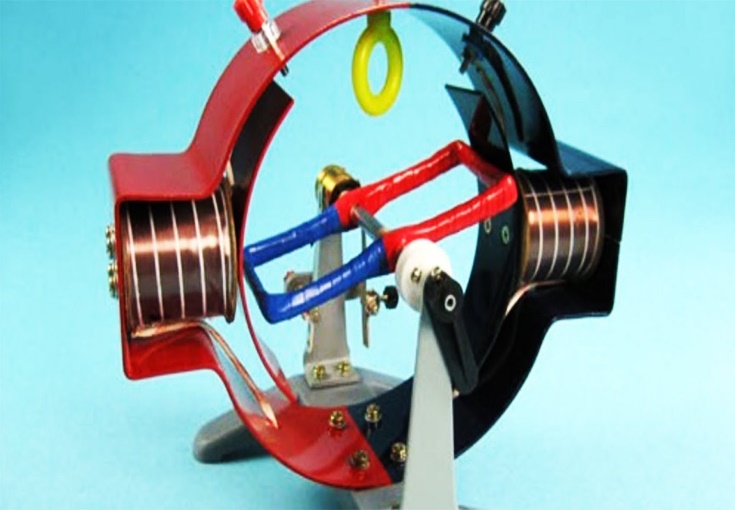


Figure 6.4 DC Motor

There is no basic difference in the [construction of a DC generator](http://www.studyelectrical.com/2014/06/construction-dc-motor-dc-motor-construction.html) and a [DC motor](http://www.studyelectrical.com/2014/06/construction-dc-motor-dc-motor-construction.html). In fact, the same d.c. machine can be used interchangeably as a generator or as a motor. DC motors are seldom used in ordinary applications because all electric supply companies furnish alternating current. However, for special applications such as in steel mills, mines and electric trains, it is advantageous to convert alternating current into direct current in order to use dc motors. The reason is that speed/torque characteristics of d.c. motors are much more superior to that of A.C. motors. Therefore, it is not surprising to note that for industrial drives, D.C. motors are as popular as 3-phase induction motors.

**6.1.6 Relay**

A relay is an [electrically](https://en.wikipedia.org/wiki/Electric) operated [switch](https://en.wikipedia.org/wiki/Switch). Many relays use an [electromagnet](https://en.wikipedia.org/wiki/Electromagnet) to mechanically operate a switch, but other operating principles are also used, such as [solid-state relays](https://en.wikipedia.org/wiki/Solid-state_relay). Relays are used where it is necessary to control a circuit by a separate low-power signal, or where several circuits must be controlled by one signal. The first relays were used in long distance [telegraph](https://en.wikipedia.org/wiki/Electrical_telegraph) circuits as amplifiers: they repeated the signal coming in from one circuit and re-transmitted it on another circuit. Relays were used extensively in telephone exchanges and early computers to perform logical operations.

Text

Description automatically generated with medium confidence

Figure 6.5 Relay

A type of relay that can handle the high power required to directly control an electric motor or other loads is called a [contactor](https://en.wikipedia.org/wiki/Contactor). [Solid-state relays](https://en.wikipedia.org/wiki/Solid-state_relay) control power circuits with no [moving parts](https://en.wikipedia.org/wiki/Moving_parts), instead using a semiconductor device to perform switching. Relays with calibrated operating characteristics and sometimes multiple operating coils are used to protect electrical circuits from overload or faults; in modern electric power systems these functions are performed by digital instruments still called "[protective relays](https://en.wikipedia.org/wiki/Protective_relay)".

Magnetic latching relays require one pulse of coil power to move their contacts in one direction, and another, redirected pulse to move them back. Repeated pulses from the same input have no effect. Magnetic latching relays are useful in applications where interrupted power should not be able to transition the contacts.

Magnetic latching relays can have either single or dual coils. On a single coil device, the relay will operate in one direction when power is applied with one polarity, and will reset when the polarity is reversed. On a dual coil device, when polarized voltage is applied to the reset coil the contacts will transition. AC controlled magnetic latch relays have single coils that employ steering diodes to differentiate between operate and reset commands. A simple electromagnetic relay consists of a coil of wire wrapped around a [soft iron core](https://en.wikipedia.org/wiki/Magnetic_core), an iron yoke which provides a low [reluctance](https://en.wikipedia.org/wiki/Magnetic_reluctance) path for magnetic flux, a movable iron [armature](https://en.wikipedia.org/wiki/Armature_(electrical_engineering)), and one or more sets of contacts (there are two contacts in the relay pictured). The armature is hinged to the yoke and mechanically linked to one or more sets of moving contacts. The armature is held in place by a [spring](https://en.wikipedia.org/wiki/Spring_(device)) so that when the relay is de-energized there is an air gap in the magnetic circuit. In this condition, one of the two sets of contacts in the relay pictured is closed, and the other set is open. Other relays may have more or fewer sets of contacts depending on their function. The relay in the picture also has a wire connecting the armature to the yoke. This ensures continuity of the circuit between the moving contacts on the armature, and the circuit track on the [printed circuit board](https://en.wikipedia.org/wiki/Printed_circuit_board) (PCB) via the [yoke](https://en.wikipedia.org/wiki/Yoke), which is soldered to the PCB.

When an [electric current](https://en.wikipedia.org/wiki/Electric_current) is passed through the coil it generates a [magnetic field](https://en.wikipedia.org/wiki/Magnetic_field) that activates the armature, and the consequent movement of the movable contact(s) either makes or breaks (depending upon construction) a connection with a fixed contact. If the set of contacts was closed when the relay was de-energized, then the movement opens the contacts and breaks the connection, and vice versa if the contacts were open. When the current to the coil is switched off, the armature is returned by a force, approximately half as strong as the magnetic force, to its relaxed position. Usually, this force is provided by a spring, but gravity is also used commonly in industrial motor starters. Most relays are manufactured to operate quickly. In a low-voltage application this reduces noise; in a high voltage or current application it reduces [arcing](https://en.wikipedia.org/wiki/Relay#Undesired_arcing).

When the coil is energized with [direct current](https://en.wikipedia.org/wiki/Direct_current), a [diode](https://en.wikipedia.org/wiki/Diode) is often placed across the coil to dissipate the energy from the collapsing magnetic field at deactivation, which would otherwise generate a [voltage spike](https://en.wikipedia.org/wiki/Voltage_spike) dangerous to [semiconductor](https://en.wikipedia.org/wiki/Semiconductor) circuit components. Such diodes were not widely used before the application of [transistors](https://en.wikipedia.org/wiki/Transistor) as relay drivers, but soon became ubiquitous as early [germanium transistors](https://en.wikipedia.org/wiki/Bipolar_junction_transistor#Germanium_transistors) were easily destroyed by this surge. Some automotive relays include a diode inside the relay case.

If the relay is driving a large, or especially a [reactive](https://en.wikipedia.org/wiki/Electrical_reactance) load, there may be a similar problem of surge currents around the relay output contacts. In this case a [snubbed](https://en.wikipedia.org/wiki/Snubber) circuit (a capacitor and resistor in series) across the contacts may absorb the surge. Suitably rated capacitors and the associated resistor are sold as a single packaged component for this commonplace use.

If the coil is designed to be energized with [alternating current](https://en.wikipedia.org/wiki/Alternating_current) (AC), some method is used to split the flux into two out-of-phase components which add together, increasing the minimum pull on the armature during the AC cycle. Typically, this is done with a small copper "shading ring" crimped around a portion of the core that creates the delayed, out-of-phase component, which holds the contacts during the zero crossings of the control voltage.

**6.1.7 Buzzer**

A buzzer or beeper is an [audio](https://en.wikipedia.org/wiki/Sound) signalling device, which may be [mechanical](https://en.wikipedia.org/wiki/Machine), [electromechanical](https://en.wikipedia.org/wiki/Electromechanics), or [piezoelectric](https://en.wikipedia.org/wiki/Piezoelectricity). Typical uses of buzzers and beepers include [alarm devices](https://en.wikipedia.org/wiki/Alarm_devices), [timers](https://en.wikipedia.org/wiki/Timer), and confirmation of user input such as a mouse click or keystroke. The fig shows the connection of the buzzer in the circuit.

**Diagram, schematic

Description automatically generated**

Figure 6.6: Buzzer

Piezo buzzers are used for making beeps, tones and alerts. This one is petite but loud! Drive it with 3-30V peak-to-peak square wave. To use, connect one pin to ground (either one) and the other pin to a square wave out from a timer or microcontroller. For the loudest tones, stay around 4 KHz, but works quite well from 2 KHz to 10 KHz. For extra loudness, you can connect both pins to a microcontroller and swap which pin is high or low ('differential drive') for double the volume. The PS series are high-performance buzzers that employ unimorph piezoelectric elements and are designed for easy incorporation into various circuits.

***Features:***

* They feature extremely low power consumption in comparison to electromagnetic units.
* Because these buzzers are designed for external excitation, the same part can serve as both a musical tone oscillator and a buzzer.
* They can be used with automated inserters. Moisture-resistant models are also available.
* The lead wire type (PS1550L40N) with both-sided adhesive tape installed easily is prepared
* A buzzer or beeper is an audio signalling device, which may be mechanical, electromechanical, or piezoelectric. Typical uses of buzzers and beepers include alarm devices, timers and confirmation of user input such as a mouse click or keystroke. Buzzer is an integrated structure of electronic transducers, DC power supply, widely used in computers, printers, copiers, alarms, electronic toys, automotive electronic equipment, telephones, timers and other electronic products for sound devices. Active buzzer 5V Rated power can be directly connected to a continuous sound, this section dedicated sensor expansion module and the board in combination, can complete a simple circuit design.

***Specifications:***

* On-board passive buzzer
* On-board 8550 triode drive
* Can control with single-chip microcontroller IO directly
* Working voltage: 5V
* Board size: 22 (mm) x12 (mm)

**6.1.8 Camera**

A **webcam** is a [video camera](https://en.wikipedia.org/wiki/Video_camera) that feeds or [streams](https://en.wikipedia.org/wiki/Streaming_media) an image or video in real time to or through a [computer](https://en.wikipedia.org/wiki/Computer) to a [computer network](https://en.wikipedia.org/wiki/Computer_network), such as the [Internet](https://en.wikipedia.org/wiki/Internet). Webcams are typically small cameras that sit on a desk, attach to a user's monitor, or are built into the hardware. Webcams can be used during a video chat session involving two or more people, with conversations that include live audio and video. For example, [Apple](https://en.wikipedia.org/wiki/Apple_Inc.)'s [iSight](https://en.wikipedia.org/wiki/ISight" \o "ISight) camera, which is built into Apple laptops, iMacs and a number of [iPhones](https://en.wikipedia.org/wiki/IPhone), can be used for video chat sessions, using the [iChat](https://en.wikipedia.org/wiki/IChat) instant messaging program (now called [Messages](https://en.wikipedia.org/wiki/Messages_(Apple))). Webcam software enables users to record a video or stream the video on the Internet. As video streaming over the Internet requires a lot of [bandwidth](https://en.wikipedia.org/wiki/Bandwidth_(computing)), such streams usually use [compressed formats](https://en.wikipedia.org/wiki/Video_compression). The maximum resolution of a webcam is also lower than most handheld video cameras, as higher resolutions would be reduced during transmission. The lower resolution enables webcams to be relatively inexpensive compared to most video cameras, but the effect is adequate for video chat sessions.

The term "webcam" (a [clipped compound](https://en.wikipedia.org/wiki/Clipped_compound)) may also be used in its original sense of a [video camera](https://en.wikipedia.org/wiki/Video_camera) connected to the [Web](https://en.wikipedia.org/wiki/World_Wide_Web) continuously for an indefinite time, rather than for a particular session, generally supplying a view for anyone who visits its [web page](https://en.wikipedia.org/wiki/Web_page) over the Internet. Some of them, for example, those used as online [traffic cameras](https://en.wikipedia.org/wiki/Traffic_camera), are expensive, rugged [professional video cameras](https://en.wikipedia.org/wiki/Professional_video_camera).

Webcams typically include a lens, an [image sensor](https://en.wikipedia.org/wiki/Image_sensor), support electronics, and may also include one or even two [microphones](https://en.wikipedia.org/wiki/Microphone) for sound.



Figure 6.7 USB Camera

**6.2 SOFTWARE DESCRIPTION**

**6.2.1 PYTHON**

In this project python is used as a programming language for development.

In technical terms, Python is an object-oriented, high-level programming language with integrated dynamic semantics primarily for web and app development. It is extremely attractive in the field of Rapid Application Development because it offers dynamic typing and dynamic binding options.

Python is relatively simple, so it's easy to learn since it requires a unique syntax that focuses on readability. Developers can read and translate Python code much easier than other languages. In turn, this reduces the cost of program maintenance and development because it allows teams to work collaboratively without significant language and experience barriers.

Additionally, Python supports the use of modules and packages, which means that programs can be designed in a modular style and code can be reused across a variety of projects. Once a module or package is developed by an user, it can be scaled for use in other projects, and it's easy to import or export these modules.

One of the most promising benefits of Python is that both the standard library and the interpreter are available free of charge, in both binary and source form. There is no exclusivity either, as Python and all the necessary tools are available on all major platforms. Therefore, it is an enticing option for developers who don't want to worry about paying high development costs.

**CHAPTER – 7**

**RESULTS AND DISCUSSIONS**

**7.1 IMPLEMENTATION**

This chapter discusses about the practical results obtained while implementing the project in hardware.

**7.2 RESULTS OBTAINED**

To begin with, we can split our project into modules of implementation that is done.

The complete hardware kit of the project can be seen in the below figure

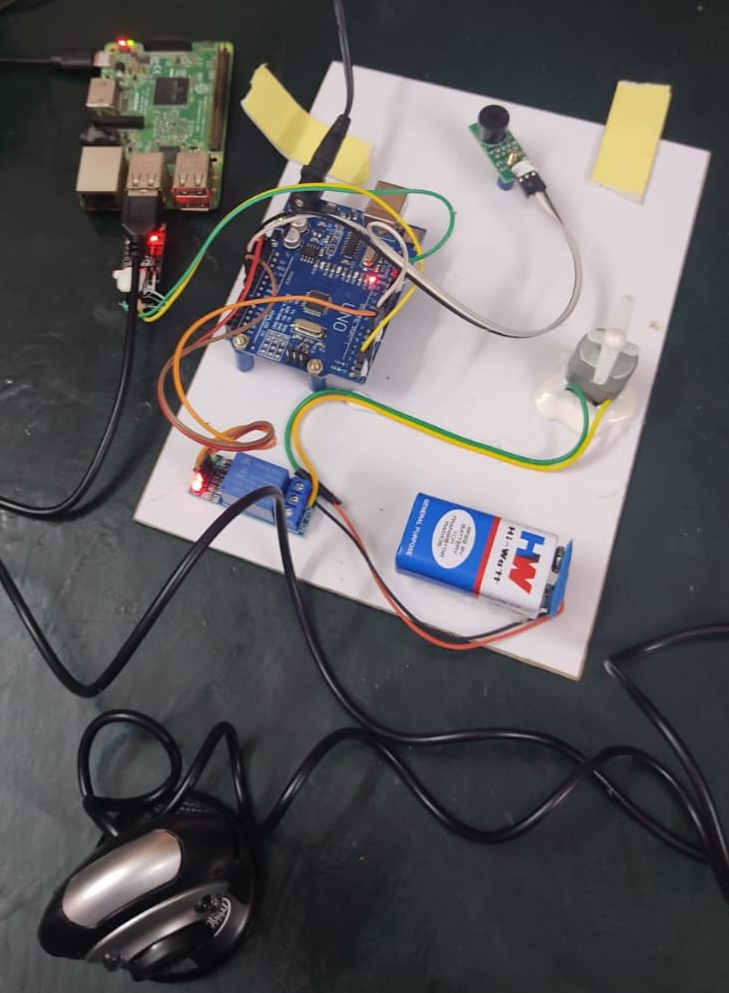


Figure 7.1: Complete Hardware Kit

The first step involved in this project is the processing of the face detection in the camera and finding the EAR to find out the drowsiness of the driver.

Raspberry pi is used as processing unit and camera module is used to predict the face in front of it and EAR calculations are done to predict and a threshold is set in the coding part to determine the drowsiness.

The code is run in the Raspberry pi which can be seen in the below figure

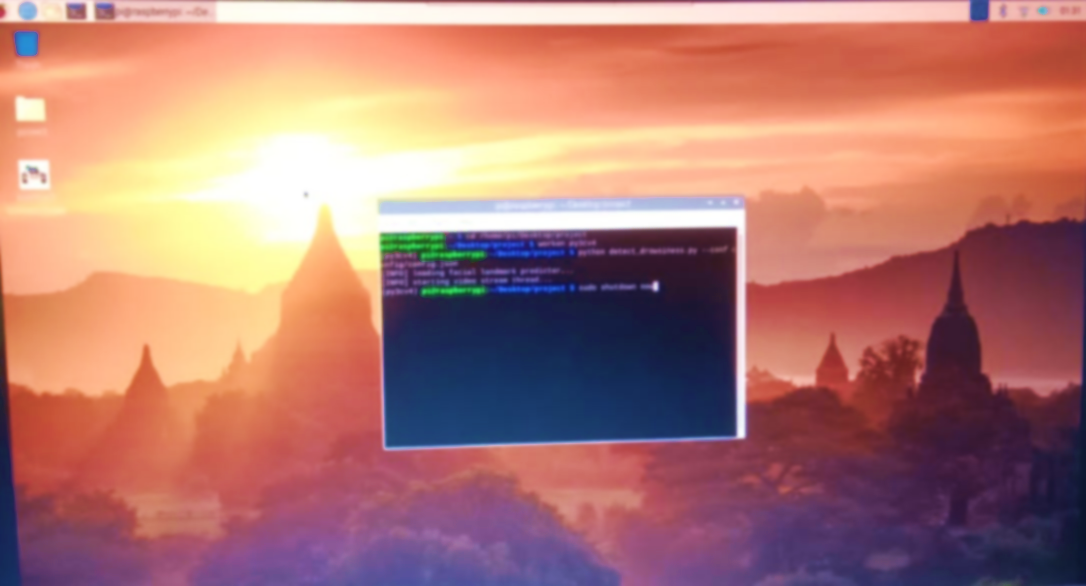


Figure 7.2 Running the code

The presence of face is determined using Haar cascade and as the EAR threshold is matched, drowsiness is predicted and it can be seen in the below figure.

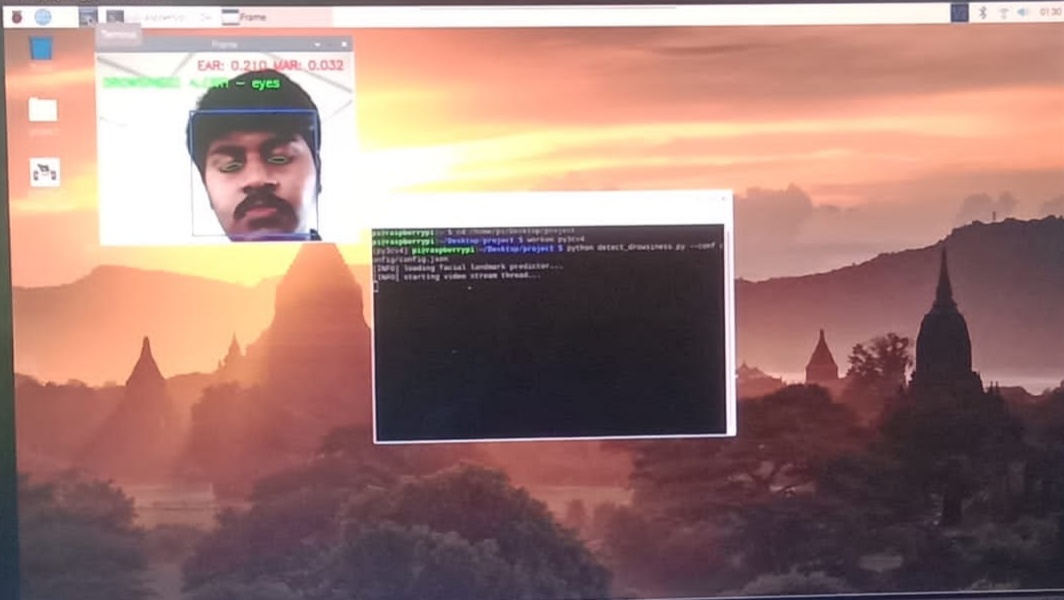


Figure 7.3 Drowsiness Detection

As the drowsiness is detected, a command is sent to the hardware kit consisting of a DC motor running which can be seen in the below figure.

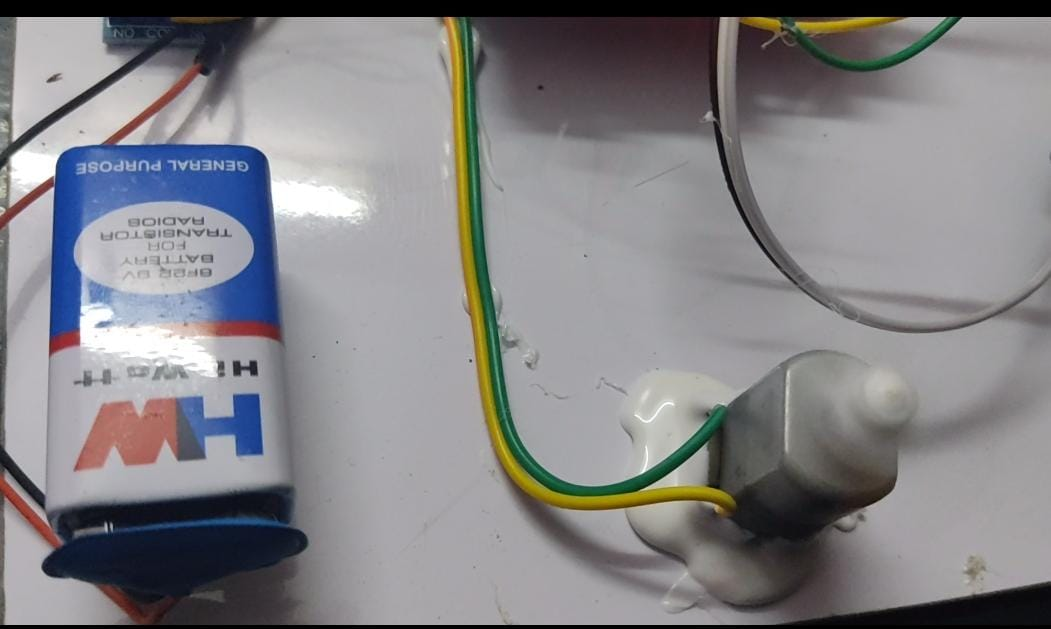


Figure 7.4 Hardware Kit with DC Motor Running

As the command from the raspberry pi comes, the DC motor is switched off and the buzzer is switched on which can be seen in the below figure

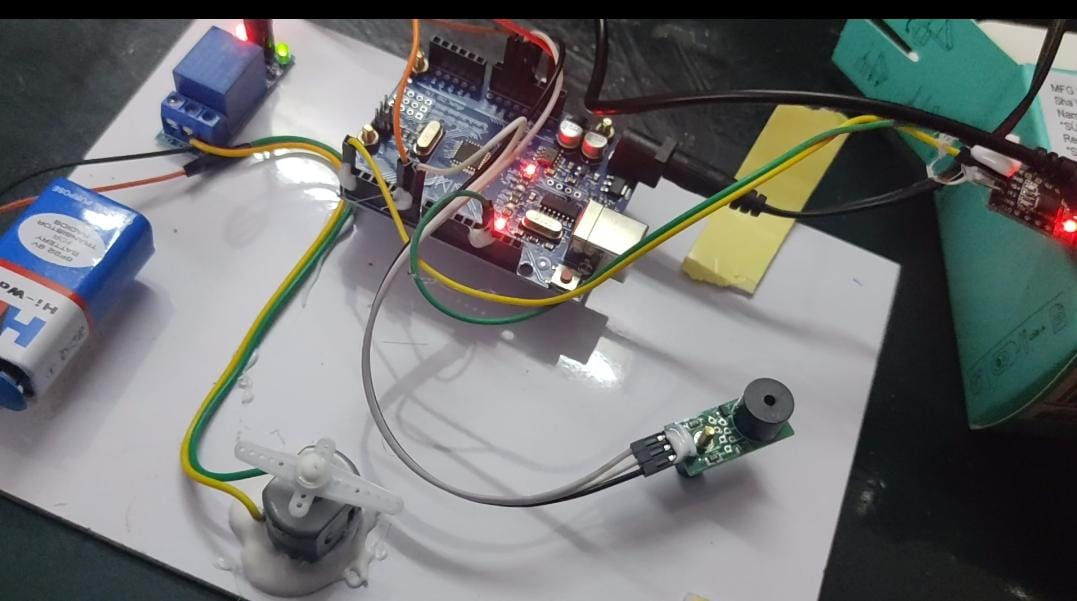


Figure 7.5 DC Motor Off and Buzzer On

Whenever the driver is felt with drowsiness, the motor is switch off and buzzer is switched on which in real time can be connected to the accelerometer of the car to safeguard the driver and the passengers.

**CHAPTER – 8**

**CONCLUSION & FUTURE WORK**

**8.1 CONCLUSION:**

In this Python project, we have built a drowsy driver alert system that can be implemented in numerous ways. We used Haar cascade to detect faces and eyes and then we used an EAR calculation to predict the status of driver whether he is drowsy or not. Automatically, the DC motor is switched off and the buzzer is switched on to alert the driver.

**8.2 FUTURE WORK:**

In the coming future, we review the application of the drowsiness detection in the automobile ﬁeld and it can promote for detecting the driver drowsiness with more accuracy. Thus, this project has an efficient scope in coming future where manual predicting can be converted to computerized production in a cheap way and save the life of the driver.

**APPENDIX-1**

**CODING**

**# USAGE**

**# Python detect\_drowsiness.py --conf config/config.json**

**# Import the necessary packages**

**from project.utils import Conf**

**from imutils.video import VideoStream**

**from imutils import face\_utils**

**from datetime import datetime**

**import numpy as np**

**import argparse**

**import imutils**

**import time**

**import dlib**

**import cv2**

**import time**

**import serial**

**ser = serial.Serial("/dev/ttyUSB0",9600,timeout=1)**

**counter=0**

**def euclidean\_dist(ptA, ptB):**

**# compute and return the euclidean distance between the two**

**# points**

**return np.linalg.norm(ptA - ptB)**

**def eye\_aspect\_ratio(eye):**

**# compute the euclidean distances between the two sets of**

**# vertical eye landmarks (x, y)-coordinates**

**a = euclidean\_dist(eye[1], eye[5])**

**b = euclidean\_dist(eye[2], eye[4])**

**# compute the euclidean distance between the horizontal**

**# eye landmark (x, y)-coordinates**

**c = euclidean\_dist(eye[0], eye[3])**

**# compute the eye aspect ratio**

**ear = (a + b) / (2.0 \* c)**

**# return the eye aspect ratio**

**return ear**

**def mouth\_aspect\_ratio(mouth):**

**# compute the euclidean distances between the three sets of**

**# vertical mouth landmarks (x, y)-coordinates**

**a = euclidean\_dist(mouth[1], mouth[7])**

**b = euclidean\_dist(mouth[2], mouth[6])**

**c = euclidean\_dist(mouth[3], mouth[5])**

**# compute the euclidean distance between the horizontal**

**# mouth landmark (x, y)-coordinates**

**d = euclidean\_dist(mouth[0], mouth[4])**

**# compute the mouth aspect ratio**

**mar = (a + b + c) / (2.0 \* d)**

**# return the mouth aspect ratio**

**return mar**

**# construct the argument parser and parse the arguments**

**ap = argparse.ArgumentParser()**

**ap.add\_argument("-c", "--conf", required=True,**

**help="Path to the input configuration file")**

**args = vars(ap.parse\_args())**

**# load the configuration file**

**conf = Conf(args["conf"])**

**# check to see if we are using GPIO/TrafficHat as an alarm**

**if conf["alarm"]:**

**from gpiozero import TrafficHat**

**th = TrafficHat()**

**print("[INFO] using TrafficHat alarm...")**

**# initialize the frame center coordinates**

**centerX = None**

**centerY = None**

**# initialize the blink counter, yawn counter, a boolean used to**

**# indicate if the alarm is going off, and start time**

**blinkCounter = 0**

**yawnCounter = 0**

**alarmOn = False**

**startTime = None**

**# load OpenCV's Haar cascade for face detection (which is faster than**

**# dlib's built-in HOG detector, but less accurate), then create the**

**# facial landmark predictor**

**print("[INFO] loading facial landmark predictor...")**

**detector = cv2.CascadeClassifier(conf["cascade\_path"])**

**predictor = dlib.shape\_predictor(conf["shape\_predictor\_path"])**

**# grab the indexes of the facial landmarks for the left, right eye,**

**# and inner part of the mouth respectively**

**(lStart, lEnd) = face\_utils.FACIAL\_LANDMARKS\_IDXS["left\_eye"]**

**(rStart, rEnd) = face\_utils.FACIAL\_LANDMARKS\_IDXS["right\_eye"]**

**(mStart,mEnd)=face\_utils.FACIAL\_LANDMARKS\_IDXS["inner\_mouth”]**

**# start the video stream thread**

**print("[INFO] starting video stream thread...")**

**vs = VideoStream(src=0).start()**

**#vs = VideoStream(usePiCamera=True).start()**

**time.sleep(2.0)**

**# loop over frames from the video stream**

**while True:**

**# grab the frame from the threaded video file stream, resize,**

**# flip horizontally, and convert to grayscale**

**frame = vs.read()**

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

**frame = cv2.flip(frame, 1)**

**gray = cv2.cvtColor(frame, cv2.COLOR\_BGR2GRAY)**

**# set the frame center coordinates**

**if centerX is None and centerY is None:**

**(centerX, centerY) = (frame.shape[1] // 2,**

**frame.shape[0] // 2)**

**# detect faces in the grayscale frame**

**rects = detector.detectMultiScale(gray, scaleFactor=1.1,**

**minNeighbors=5, minSize=(30, 30),**

**flags=cv2.CASCADE\_SCALE\_IMAGE)**

**# loop over the detected faces**

**for rect in rects:**

**# draw a bounding box surrounding the face**

**(x, y, w, h) = rect**

**cv2.rectangle(frame, (x, y), (x + w, y + h), (255, 0, 0), 2)**

**# check if the number of faces detected is greater than zero**

**if len(rects) > 0:**

**# sort the detected rectangles by their position relative to**

**# the center and grab the face that's closest to the center**

**centerRect = sorted(rects, key=lambda r: abs((**

**r[0] + (r[2] / 2)) - centerX) + abs((**

**r[1] + (r[3] / 2)) - centerY))[0]**

**# get the coordinates of the rectangle in the center and**

**# construct a dlib rectangle object from the Haar cascade**

**# bounding box**

**(x, y, w, h) = centerRect**

**rect = dlib.rectangle(int(x), int(y), int(x + w),**

**int(y + h))**

**# determine the facial landmarks for the face region, then**

**# convert the facial landmark (x, y)-coordinates to a NumPy**

**# array**

**shape = predictor(gray, rect)**

**shape = face\_utils.shape\_to\_np(shape)**

**# extract the left and right eye coordinates, then use the**

**# coordinates to compute the eye aspect ratio for both eyes**

**leftEye = shape[lStart:lEnd]**

**rightEye = shape[rStart:rEnd]**

**leftEAR = eye\_aspect\_ratio(leftEye)**

**rightEAR = eye\_aspect\_ratio(rightEye)**

**# average the eye aspect ratio together for both eyes**

**ear = (leftEAR + rightEAR) / 2.0**

**# compute the convex hull for the left and right eye, then**

**# visualize each of the eyes**

**leftEyeHull = cv2.convexHull(leftEye)**

**rightEyeHull = cv2.convexHull(rightEye)**

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

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

**# check to see if the eye aspect ratio is below the blink**

**# threshold**

**if ear < conf["EYE\_AR\_THRESH"]:**

**# increment the blink frame counter**

**blinkCounter += 1**

**# if the eyes were closed for a sufficient number of**

**# frames, then sound the alarm**

**if blinkCounter >= conf["EYE\_AR\_CONSEC\_FRAMES"]:**

**# if the alarm is not on, turn it on**

**if not alarmOn:**

**alarmOn = True**

**# check to see if the TrafficHat buzzer should**

**# be sounded and red light set to blink**

**if conf["alarm"]:**

**th.buzzer.blink(0.1, 0.1, 30,**

**background=True)**

**th.lights.red.blink(0.1, 0.1, 30,**

**background=True)**

**# draw an alarm on the frame**

**ser.write('a'.encode('utf-8'))**

**cv2.putText(frame, "DROWSINESS ALERT! - eyes", (10, 60),**

**cv2.FONT\_HERSHEY\_SIMPLEX, 0.7, (0, 255, 0), 2)**

**# otherwise, the eye aspect ratio is not below the blink**

**# threshold, so reset the counter and alarm**

**else:**

**blinkCounter = 0**

**alarmOn = False**

**# extract the inner mouth coordinates, then use the**

**# coordinates to compute the mouth aspect ratio for the mouth**

**mouth = shape[mStart:mEnd]**

**mar = mouth\_aspect\_ratio(mouth)**

**# compute the convex hull for the left and right eye, then**

**# visualize each of the eyes**

**mouthHull = cv2.convexHull(mouth)**

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

**# check to see if the mouth aspect ratio is above the yawn**

**# threshold**

**if mar > conf["MOUTH\_AR\_THRESH"]:**

**# increment the yawn frame counter and set the start time**

**yawnCounter += 1**

**startTime = datetime.now() if startTime == None else \**

**startTime**

**# check to see if yawn frame counter is greater than yawn**

**# frame threshold and if the difference between current**

**# time and start time is less than or equal to yawn**

**# threshold time (in which case the person is drowsy)**

**if yawnCounter >= conf["YAWN\_THRESH\_COUNT"] and \**

**(datetime.now() - startTime).seconds <= \**

**conf["YAWN\_THRESH\_TIME"]:**

**# if the alarm is not on, turn it on**

**if not alarmOn:**

**alarmOn = True**

**# check to see if the TrafficHat buzzer should**

**# be sounded and red light set to blink**

**if conf["alarm"]:**

**th.buzzer.blink(0.1, 0.1, 10,**

**background=True)**

**th.lights.red.blink(0.1, 0.1, 30,**

**background=True)**

**# draw an alarm on the frame**

**ser.write('a'.encode('utf-8'))**

**cv2.putText(frame, "DROWSINESS ALERT! - yawning",**

**(10, 85), cv2.FONT\_HERSHEY\_SIMPLEX, 0.7,**

**(0, 0, 255), 2)**

**# check to see if the start time is set**

**elif startTime != None:**

**# check if the difference between current time and start**

**# time is greater than yawn threshold time**

**if (datetime.now() - startTime).seconds > \**

**conf["YAWN\_THRESH\_TIME"]:**

**# reset yawn counter, alarm flag and start time**

**yawnCounter = 0**

**alarmOn = False**

**startTime = None**

**# draw the computed aspect ratios on the frame**

**cv2.putText(frame, "EAR: {:.3f} MAR: {:.3f}".format(**

**ear, mar), (175, 30), cv2.FONT\_HERSHEY\_SIMPLEX, 0.7,**

**(0, 0, 255), 2)**

**# if the 'display flag is set, then display the current frame**

**# to the screen and record if a user presses a key**

**if conf["display"]:**

**cv2.imshow("Frame", frame)**

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

**# if the `q` key was pressed, break from the loop**

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

**break**

**# check to see if we have any open windows, and if so, close them**

**if conf["display"]:**

**cv2.destroyAllWindows()**

**# release the video stream pointer**

**vs.stop()**

**APPENDIX-II**

**REFERENCES**

[1] Mozhgan Nasr Azadani and Azzedine Boukerche , Fellow, IEEE “Near Real–Time Freeway Accident Detection” Driving Behavior Analysis Guidelines for

Intelligent Transportation Systems” ,[2021]

[2] Anusha Vangala, Basudeb Bera, Sourav Saha, Student Member, Ashok Kumar Das, Senior Member, IEEE, Neeraj Kumar, Senior Member, IEEE, YoungHo Park, Member, IEEE,” Blockchain-Enabled Certificate-Based Authentication for Vehicle Accident Detection and Notification in Intelligent Transportation Systems”, [2020]

[3] Natalia Selini Hadjidimitriou , Marco Lippi , Mauro Dell’Amico, and Alexander Skiera,” Machine Learning for Severity Classification of Accidents Involving Powered Two Wheelers”[2019]

[4] Yu Xiang , Member, IEEE, Sida Huang, Min Li, Jin Li , Member, IEEE, and Wenyong Wang , Member, IEEE,” Rear-End Collision Avoidance-Based on Multi-Channel Detection ”,[2019]

[5] Luca Canzian, Ugur Demiryurek, Member, IEEE, and Mihaela van der Schaar, Fellow, IEEE,” Collision Detection by Networked Sensors ”,[2019]

[6] Dinesh Singh , Student Member, IEEE, and Chalavadi Krishna Mohan, Member, IEEE,” Deep Spatio-Temporal Representation for Detection of Road Accidents Using Stacked Autoencoder”,[2018]

[7] Yu-Jun Zheng, Member, IEEE, Sheng-Yong Chen, Senior Member, IEEE, Yu Xue, and Jin-Yun Xue,” A Pythagorean-Type Fuzzy Deep Denoising Autoencoder for Industrial Accident Early Warning”,[2017]

[8] Giovanni Savino, Federico Giovannini, Michael Fitzharris, and Marco Pierini,” Inevitable Collision States for Motorcycle-to-Car Collision Scenarios ”,[2016]

[9] Vicent Girbes, Leopoldo Armesto, Juan Dols and Josep Tornero,” Haptic Feedback to Assist Bus Drivers for Pedestrian Safety at Low Speed ”,[2015]

[10] Younshik Chung and Wilfred W. Recker,” Frailty Models for the Estimation of Spatiotemporally Maximum Congested Impact Information on Freeway Accidents”,[2015]