DRIVER ALERTNESS DETECTION

A Project Report

Submitted by

DHUR WAHID QUADRI	**20191CSE0131**
DILIP KUMAR M	**20191CSE0132**
EARANNA RAJU C	**20191CSE0140**
GAGANA G	**20191CSE0153**
GALLA GIREESH	**20191CSE0158**

Under the guidance of

Dr. Sulaiman Syed Mohamed
Associate Professor

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY
IN
COMPUTER SCIENCE AND ENGINEERING
At

SCHOOL OF COMPUTER SCIENCE ENGINEERING

&

INFORMATION SCIENCE

PRESIDENCY UNIVERSITY

BENGALURU

JUNE 2023

PRESIDENCY UNIVERSITY

SCHOOL OF COMPUTER SCIENCE ENGINEERING
&
INFORMATION SCIENCE

CERTIFICATE

This is to certify that the Project report **"DRIVER ALERTNESS DETECTION"** being submitted by **DHUR WAHID QUADRI, DILIP KUMAR M, EARANNA RAJU C, GAGANA G, GALLA GIREESH** bearing roll number(s) **20191CSE0131, 20191CSE0132, 20191CSE0140, 20191CSE0153, 2019CSE0158** in partial fulfillment of the requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering is a Bonafide work carried out under my supervision.

27/1/2023

Dr. SULAIMAN SYED MOHAMED
Assistant Professor
School of CSE & IS
Presidency University

Pallavi R 27/05/23

Dr. PALLAVI R
Associate Professor & HOD
School of CSE & IS
Presidency University

Dr. C. KALAIARASAN
Associate Dean
School of CSE & IS
Presidency University

Dr. MD. SAMEERUDDIN KHAN
Dean
School of CSE & IS
Presidency University

PRESIDENCY UNIVERSITY

SCHOOL OF COMPUTER SCIENCE & ENGINEERING
&
INFORMATION SCIENCE

DECLARATION

We hereby declare that the work, which is being presented in the project report entitled **DRIVER ALERTNESS DETECTION** in partial fulfillment for the award of Degree of **Bachelor of Technology** in **Computer Science and Engineering**, is a record of our own investigations carried under the guidance of **Dr. SULAIMAN SYED MOHAMED, Assistant Professor, School of CSE & IS, Presidency University, Bengaluru.**

We have not submitted the matter presented in this report anywhere for the award of any other Degree.

Name	ID	Signature
Dhur Wahid Quadri	20191CSE0131	DhurWahidQuadri
Dilip Kumar M	20191CSE0132	Dilip kumar
Earanna Raju C	20191CSE0140	Earanna Raju
Gagana G	20191CSE0153	Gagana G
Galla Gireesh	20191CSE0158	Galla Gireesh

ABSTRACT

The need of good driver alertness detection systems is highlighted by the fact that driver weariness and drowsiness are significant causes of traffic accidents. The effort described in the following piece is centered on the creation and assessment of a driver alertness detecting system. The system uses a number of technologies and approaches to track and spot indicators of driver drowsiness improving traffic safety. The project starts with a thorough literature review to understand the methods and procedures currently employed in driver alertness detection systems. It examines the physiological and behavioral signs of driver weariness, including eye movements, facial expressions, and bodily functions including heart rate and skin conductance. A thorough framework is created based on this analysis, including numerous sensors and algorithms for precise alertness detection. The technology uses computer vision techniques to examine head movements, eye closure, and facial expressions in order to assess the driver's level of weariness. The essential data from the sensor data is extracted using feature extraction techniques, and classification models are trained to differentiate between alert and exhausted states. The implementation of real-time processing capabilities enables the driver to receive feedback right away and initiate the necessary actions, such as audio alarms and notifications. The newly designed driver alertness detection technology has the potential to reduce the dangers of driver drowsiness and increase road safety. The results of this project can be used as a starting point for more study and development in the area of driver alertness monitoring systems, promoting technological improvements and lowering the number of accidents caused by driver drowsiness.

ACKNOWLEDGEMENT

We express our sincere thanks to our respected dean **DR. MD. SAMEERUDDIN KHAN**, Dean, School of CSE & IS, Presidency University for getting us permission to undergo the project.

We record our heartfelt gratitude to our beloved Associate Dean **Dr. C. KALAIARASAN,** Professor **Dr. T K THIVAKARAN,** University Project-II In-charge, School of CSE & IS, Presidency University for rendering timely help for the successful completion of this project.

We would like to convey our gratitude and heartfelt thanks to the University Project II Co-ordinators **Mr. MRUTYUNJAYA MS, Mr. SANJEEV P K KAULGUD, Mr. RAMA KRISHNA K and Dr. MADHUSUDHAN MV**.

We are greatly indebted to our guide, **Dr. SULAIMAN SYED MOHAMED**, Associate Professor School of CSE & IS, Presidency University for his inspirational guidance, valuable suggestions and providing us a chance to express our technical capabilities in every respect for the completion of the project work.

We thank our family and friends for the strong support and inspiration they have provided us in bringing out this project.

DHUR WAHID QUADRI	**20191CSE0131**
DILIP KUMAR M	**20191CSE0132**
EARANNA RAJU C	**20191CSE0140**
GAGANA G	**20191CSE0153**
GALLA GIREESH	**20191CSE0158**

List of Figures

TABLE OF CONTENTS

CHAPTER-1
INTRODUCTION

Given the rising incidence of crashes involving fatigued and drowsy drivers in today's fast-paced world, road safety has emerged as a top priority. Statistics show that driver drowsiness is a major factor in many accidents around the world, many of which result in serious injuries or even fatalities. Advanced technologies have arisen to address this crucial issue, attempting to improve driver safety by spotting indicators of tiredness and warning the driver in real-time.

In order to monitor a driver's condition and issue prompt warnings to prevent accidents brought on by drowsiness, this project introduces a Driver Alertness Detection System. This system seeks to identify early indicators of driver tiredness or distraction, allowing necessary actions to be taken before it may result in potentially fatal accidents. It does this by utilizing cutting-edge technology and artificial intelligence algorithms.

1.1 Features

A number of crucial components are included in the Driver Alertness Detection System project to efficiently monitor driver alertness and improve traffic safety. This system's primary components include the following:

The device tracks the driver's eye movements and gaze patterns using cutting-edge eye-tracking technology. The system can identify indicators of drowsiness or inattention by examining elements such as blink rate, ocular closure duration, and eye movement patterns.

The project makes use of face recognition algorithms to assess the driver's facial expressions. It is capable of spotting changes in facial expressions that indicate exhaustion or drowsiness, including yawning or drooping eyelids. By keeping track of the driver's head motions, the system can spot sudden or pronounced head nods, which are frequent indications of tiredness. By allowing for prompt intervention, this feature aids in identifying early signs of weariness.

The project incorporates a real-time alert mechanism that sends the driver an instantaneous warning when indicators of distraction or drowsiness are found. The alerts can be sent to the driver as tactile, aural, or visual cues to make sure they are received in a timely manner and that they are aware of the need to take action. To continuously evaluate and understand data from numerous sensors, the system uses machine learning algorithms. The algorithms may adjust to the unique characteristics of each driver over time, improving the accuracy of sleepiness detection through the use of historical data and pattern recognition.

The Driver Alertness Detection System project seeks to offer a comprehensive solution for tracking driver alertness, preventing accidents, and eventually improving road safety by combining these characteristics.

1.2 Purpose and Scope.

The Driver Alertness Detection System project's goal is to create and put into practice a technological system that can identify and track a driver's state of alertness while operating a vehicle. By delivering real-time feedback to the driver and notifying them if their attention level dips below a specific threshold, the system seeks to increase road safety by lowering the likelihood of accidents brought on by fatigued driving. The Driver Alertness Detection System project's scope includes a number of facets connected to the creation and application of the system.

The following are the main aspects of the scope:

The project's scope involves defining the Driver Alertness Detection System's functional requirements. This entails determining the precise features and abilities the system ought to have, such as real-time monitoring of driver attentiveness, detection of drowsiness or weariness, and generation of relevant notifications for the driver. The task at hand entails choosing and incorporating the right sensors into the car in order to gather pertinent information for gauging driver awareness. Cameras, infrared sensors, steering sensors, and other devices that can track a driver's behavior and physiological signs can all be included in these sensors. To process the data gathered from the sensors, the project includes building algorithms and data processing methods. To evaluate the data and ascertain the degree of driver attentiveness, this entails the development of machine learning models or other computational techniques. The system should be able to recognize tiredness or a decline in alertness with accuracy. Designing a reliable feedback system to notify the driver when their level of attentiveness drops below a predetermined threshold is included in the scope. Depending on the system's architecture, this can entail visual clues, audio messages, or haptic feedback. The existing systems of the vehicle must be integrated with the Driver Alertness Detection System. To enable seamless interaction and coordination with other vehicle operations, this includes establishing communication and integration with the instrument cluster, infotainment system, or central control unit.

The project's scope includes creating a simple user interface that the driver can utilize to communicate with the alertness detecting system.

1.3 Objectives.

The purpose of a driver alertness detection system is to enhance road safety by monitoring and identifying signs of driver fatigue, drowsiness, or inattention. It aims to prevent accidents caused by drivers who are excessively tired, sleepy, or distracted. By detecting early warning signs and providing timely alerts, the system helps drivers to maintain their alertness and take appropriate actions to ensure safe driving. The ultimate purpose of a driver alertness detection system is to reduce the occurrence of fatigue-related accidents and improve overall road safety for both the driver and other road users.

The primary objective is to identify signs of driver fatigue or drowsiness at an early stage. This allows for timely intervention to prevent accidents causedby impaired driving due to sleepiness or exhaustion.

Continuously monitor the driver's behavior and physiologicalsignals to detect any changes indicative of reduced alertness. This includes monitoring eye movements, facial expressions, head position, and even vital signs like heart rate and skin conductance.

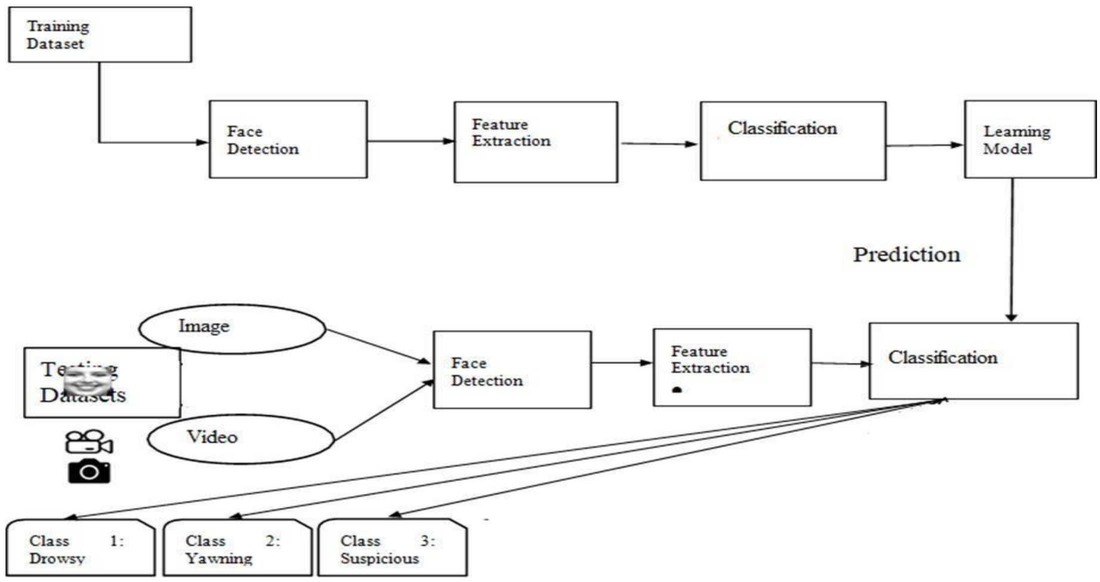


Fig 1 Block diagram of Driver Alertness System

CHAPTER 02
LITERATURE SURVEY

This According to a National Highway Traffic Safety Administration survey, 56,000 car accidents in the United States in 1996 were thought to be caused by sleep deprivation. According to a2007 survey, weariness was the primary cause of 18% of accidents. Up to 20% of serious traffic accidents in Britain were brought on by weariness. In a similar vein, a report conducted by the Road and Traffic Authority indicates that 20% of traffic accidents in 2007were caused by fatigue. Drowsy driving accidents were avoided and managed while the carwas out of control. A vehicle's alcohol detector can also be used to catch intoxicated drivers.Using the driver's eye blink is the word employed here to indicate that the driver is tired.

These incidents happened because the driver was drowsy and could not control the car whenhe or she awoke. The driver's infrared sensor, which was attached to the frame of his glasses,used the pace at which his eyes closed to detect sleepiness. The vehicle's speed is reduced ifthe driver is drowsy, and the obstacle sensor is used to detect any nearby vehicles to preventcollisions. If there are no nearby vehicles on the left side of the road, the vehicle moves to the left side of the road by auto steering and controlling, and is then parked with advance warning. Over the past few years, many researchers worked on these devices and few approaches have been reported. One of the proposed techniques is to keep an eye on how thecar is moving to spot the driver's tiredness. However, this approach has drawbacks becausethe findings depend on the kind of vehicle and the state of the road. Processing the driver's electrocardiogram (ECG) readings is another technique. This method also has disadvantagesbecause ECG probes must always be attached to the body of the driver. The motorist wouldbe bothered by that. Few studies have attempted to measure the driver's eye blink rate as a measure of their level of weariness. Many studies have developed approaches based on a combination of projection and the geometry characteristics of the iris and pupil for the successful identification of eye blink rate. The fact that the iris and pupil are darker than theskin and white area of the eye is used by T.D. Orazio and Z. Zhang.

An algorithm based on the cascade AdaBoost classifier was proposed by **Y.Lei. T. Hong**, using a specified threshold, transforms a grayscale image of an eye into a binary image. Thestate of the eye is then determined using the binary image's number of black and white pixels. The iris and eye openness were both detected using the Hough Transform in **Ms. Devi's** method. Some researchers base their conclusions about the condition of an eye on the projection of the image. The picture of the two eyes is projected vertically according to **Z. Liu**. To measure the distance between eyebrows and eyelids and to identify the condition ofan eye, one uses the horizontal projection image of the eye. The condition of an eye is calculated from the horizontal projection of the image of a face. On the "Support Vector Machine" (SVM) classifier, certain works are also based. To determine the condition of theeye, the SVM classifier is employed. F Smach extracted ocular characteristics using an SVMclassifier and Gabor filter. The developers of the aforementioned methods employed a few circumstances that made identifying the eye state challenging. The device, which includes an eye blink sensor to measure the driver's blink rate and an adaptive speed controller built with stepper motors to provide accurate throttle valve location, recognizes the driver's signsof exhaustion, vehicle’s speed.

Technology advancements give us some hope that we can, at least partially, avoid these. In this study, accidents are measured and controlled using IR and alcohol sensors. To capture video footage of the driver, it uses remotely placed charge-coupled devices cameras with active infrared illuminators. To determine the driver's level of weariness, a variety of visualcues that generally indicate a person's level of awareness were extracted in real-time and systematically merged. The visual clues used to describe head motions, eyelid, gaze, and facial expressions. To characterize human exhaustion and calculate fatigue based on visual cues, a probabilistic model was created. An accurate description of weariness is obtained bythe simultaneous use of visual clues and their methodical arrangement. This system was validated using human test subjects from various ethnic backgrounds, both with and withoutglasses, and under various lighting circumstances. It was discovered to be fairly trustworthyand accurate in describing weariness.

The Supervised Descent Method (SDM), which tracks some face landmarks to extract a cropof regions of interest (ROI) (the driver's ear region), is used by Seshadri's computer vision- based method to assess whether a driver is holding a cell phone close to his or her ears. TheROIs are used to extract features, and classifiers that have already been trained are used to identify phone usage. Near real-time processing is possible for the system. Yang's method involves sending high pitched beeps through the car's audio system, using Bluetooth to connect devices, and using phone based software to record and interpret sound signals. The location of the cell phone is determined by the sounds, and we can then determine when thedriver is on the phone. This method allows for hands-free operation, however it is brand andoperating system dependent, and the driver must continuously enable the software.

Watkins also makes a concept that describes the actions of a texting-distracted driver. The method involves pushing and releasing any key while typing on a cell phone that has been set up to record everything. Through a study of these records, distractions can be confirmed.Six volunteers took part in experiments to determine the various patterns of typing frequency displayed in each condition when using a cell phone as a passenger and a driver. The averagefrequency of two key presses per second for the driver to respond to text messages is too low. In 99% of cases, this restriction finds the driver, however it only functions when data is processed offline.

The system might be used with additional devices, such as the GPS. **Martin** describes a technique for monitoring and analyzing head motion in the driver. It makes use of the driver's optical path. A preset set of pertinent movements from the automobile environment are compared to the movements that were detected. The device tracks how long the motorist keeps looking in one direction. The distraction is noticed if the amount of time spent exceedsa set time limit. The system's precision in a real-world setting is 86%. **Doshi** and **Trivedi's** method determines the degree of driving distraction caused by looking at things outside thecar. Two computer vision systems are combined. Using a salience map, the first one determines the driver's field of view (from the inside of the car), and the second one determines movement (from the outside), since it is the motions outside that must catch thedriver's attention.

There have been several studies on tiredness detection and intoxicated driving conducted during the past ten years. Studies have been undertaken employing features based on a driver's visual characteristics, physiological characteristics, and driving behaviors, each with their own advantages for tiredness identification and by using sensors for drunk driving detection.

Anirban Dasgupta proposed a robust real-time embedded platform to monitor the loss of attention of the driver during day and night driving conditions. The percentage of eye closure was used to indicate the alertness level. Face detected using HAAR -like features, the eye state was classified as open or closed using support vector machines. **Boon-Giin Lee** et al.,proposed a method to monitor driver safety for fatigue using two distinct methods: Eye movement monitoring and Bio- signal processing. The monitoring system was designed onan Android-based smartphone, where it receives sensory data via wireless sensor network and further processed the data to indicate the current driving aptitude of the driver. The sensors used were a video sensor to capture the driver image and a bio-signal sensor to gather the driver Photoplethysmography (PPG) signal.

CHAPTER-03

REQUIREMENT ANALYSIS

Analysis is the process of decomposing a complex topic or substance into smaller parts to gain a deeper understanding. Engineers and analysts examine requirements, structures, mechanisms, and system dimensions. Analysis is an exploratory activity that marks the beginning of the project lifecycle. During the Analysis Phase, high-level project deliverables are broken down into detailed business requirements. This phase also involves establishing the overall direction of the project through the creation of project strategy documents.

The feasibility study is conducted during the system analysis phase, where a business proposal with a general project plan and cost estimates is put forth. The purpose of the feasibility study is to ensure that the proposed system does not burden the company. It is essential to have some understanding of the major system requirements for conducting the feasibility analysis. The three key considerations in feasibility analysis are economic feasibility, technical feasibility, and social feasibility.

Economic Feasibility study assesses the economic impact of the system on the organization. The company's available funds for research and development are limited, and expenditures must be justified. The developed system should stay within the budget, which was achieved by utilizing freely available technologies and only purchasing customized products.

The technical feasibility study examines the technical requirements of the system. It is important that the system does not place excessive demands on the available technical resources. A developed system should have modest requirements to minimize the impact on the client, ensuring minimal or no changes are required for implementation. The social feasibility study focuses on the level of acceptance of the system by users.

This includes user training to ensure efficient system usage. The system should not be perceivedas a threat but rather as a necessity. User acceptance depends on effective methods of educating and familiarizing users with the system. User confidence should be raised so that constructive criticism can be provided since users are the final beneficiaries of the system. Functional requirements define the functions of a system and its components, specifying inputs, behavior, and outputs. They encompass calculations, technical details, data manipulation, processing, and other specific functionalities. The functional requirements inthis system include:

- Setting up the virtual keyboard at system startup.
- Resetting the column and row intervals after each selection.
- Accurately detecting blinks.
- Accurately selecting text.

Non-functional requirements are essential guidelines employed to evaluate the performance and behavior of a system, focusing on its overall qualities and attributes rather than specific functionalities or behaviors. They articulate the desired characteristics or traits that a software system should possess in terms of its performance, usability, reliability, security, and other relevant aspects. In contrast to functional requirements that delineate what the system should accomplish, non-functional requirements establish the expected level of performance or behavior exhibited by the system.

3.1 Hardware Requirement

Hardware requirements pertain to the specifications and capabilities of the tangible elements essential for sustaining and executing a software system or application. These prerequisites delineate the minimum or suggested hardware configurations, encompassing details such as the processor's type and speed, the amount of memory (RAM), storage capacity, graphics capabilities, network connectivity, and other factors pertinent to hardware.

- Camera: iBall super-view c8.0 camera
- Speaker: Any desktop/laptop speaker
- Processor: 32bit(X-86)/64bit(X-64) Processors
- Hard Disk: 0GB to 80GB
- RAM: 4GB to 8 GB

3.2 Software Requirement

Software Requirement is a detailed explanation of the operation of the software system that is being created, often known as a requirement specification. The SRS comprises both functional as well as non-functional criteria along with an overview of how the software works. Software requirements is a branch of software engineering that focuses on gathering, analyzing, specifying, and validating software needs.

- Operating System: Windows 7+ and above
- Software : OpenCV library
- Language : Python 3.5.0
- Packages : pygame , pyInstaller , twilio and python dlib.

3.1.1 Python

Fig 2 – Python

Python is a fantastic and easy-to-learn language. It is enjoyable and versatile enough for both little and big undertakings. Your development time will be significantly reduced using Python, and Python is generally considerably faster to write than other languages. Python serves as a general-purpose, high-level, interpretive dynamic language for programming which places a strong emphasis on code readability. In contrast to Java or C++, Python's syntax enables programmers to create code in a simpler manner. Programming is simple and enjoyable with the language that creator Guido Van Rossum created in 1991.

3.1.2OpenCV

Fig 3 – OpenCV

A library of programming functions with a focus on computer vision in real time is called OpenCV (Open Source Computer Vision). Itseez (Intel later bought) originally backed the innovation after it had been developed by Intel, followed by Willow Garage. The freely available BSD license for the library renders it multi-platform as well as free to use.

3.2.3Dlib

Fig 4 – Dlib

Dlib is a well-known C++ library that enjoys significant popularity in the realm of open- source software. Its primary focus lies in facilitating machine learning and computer visiontasks, rendering it an invaluable resource for developers engaged in these areas. A diverse array of functions and tools are provided by dlib, empowering developers to create applications that encompass a wide range of tasks such as face recognition, object detection, image processing, and more.

3.2.4twilio

Fig 5 – twilio

Twilio is a cloud-based communication platform that offers developers a comprehensive range of APIs and services. These powerful tools enable developers to seamlessly integrate messaging, voice into their applications. With Twilio, businesses and developers can enhance their software systems by incorporating a wide array of communication features, including SMS messaging, voice calls, and video conferences

CHAPTER – 04
SYSTEM ANALYSIS

4.1 EXISTING METHOD

The physiological approach involves the measurement of bodily signs and physiological parameters to assess driver drowsiness. This approach relies on sensors that are directly connected to the driver's body. It measures physiological indicators such as heart rate, heart rate variability, EEG signals, EOG signals, and blood pressure. Algorithms analyze these signals to identify patterns and changes indicative of driver fatigue or drowsiness. While this approach provides a direct assessment of the driver's physiological state, it requires intrusive sensors to collect the necessary data.

On the other hand, vehicle-based methods analyze the behavior and patterns of the vehicle itself to detect signs of driver fatigue or drowsiness. Sensors and data inputs from the vehicle, such as steering wheel motion, accelerator and brake patterns, vehicle speed, and lane departure, are continuously monitored to assess the driver's state. The underlying assumption is that when a driver is fatigued or drowsy, their driving behavior and patterns deviate from normal. Changes or fluctuations in these parameters can be seen as evidence of driver impairment. Vehicle-based methods are non-intrusive and can be implemented using existing sensors in modern vehicles.

Behavior-based methods focus on analyzing the driver's behavior and actions to detect signs of drowsiness. This approach often involves computer vision techniques to track facial expressions, eye movements, and head positions. It may also analyze voice patterns and speech characteristics. By observing behavioral cues and changes in these parameters, algorithms can infer the driver's level of drowsiness. Behavior-based methods are non-intrusive and typically utilize technologies like webcams to capture visual input, which is then analyzed using machine learning algorithms such as Support Vector Machines (SVM) to identify drowsiness characteristics.

To summarize, the physiological approach directly measures physiological indicators, vehicle-based methods analyze the vehicle's behavior, and behavior-based methods observe the driver's behavioral cues. Each approach has its own advantages and considerations, with vehicle-based methods offering non-intrusive monitoring and behavior-based methods leveraging computer vision techniques for analysis.

4.2 PROPOSED METHOD

The described system is a driver drowsiness detection system that employs a combination of a PC, the OpenCV image processing library, and the QT editor. Its primary objective is to quickly detect sleepiness and process relevant data. To determine if the driver's eyes are open or closed in real time, the system relies on a Logitech camera. This camera is selected for its reliability and lack of external connections, ensuring system stability and minimizing the chance of malfunctions.

Incorporating IoT capabilities, the system provides notification services. By utilizing eye ball detection, the system can warn the driver when drowsiness is detected, thereby preventing accidents caused by drowsy driving. While the system can be applied to any object, its main purpose is to predict facial landmarks and facial form. This enables accurate tracking and analysis of the driver's face for signs of drowsiness.

The proposed design continuously records the driver's face using the camera, allowing the system to identify levels of hypo vigilance. Sleepiness is indicated by the motion of closing one's eyes. An increase in the number of eye blinks compared to the driver's usual blinking pattern is considered a sign of fatigue. Additionally, microsleep episodes lasting 3 to 4 seconds are recognized as reliable indicators of exhaustion. The OpenCV image processing library is utilized to implement closed eye motion detection. By analyzing the video feed from the camera, the

system accurately detects when the driver's eyes are closed and determines their level of drowsiness. To alert the driver of their drowsiness level, a buzzer is integrated into the system. When significant sleepiness or signs of microsleep are detected, the buzzer notifies the driver, prompting them to take necessary actions to prevent accidents. The system also includes technology to recognize other driving behaviors, such as distracted driving and mobile phone use. By using the camera to detect distractions or mobile phone usage, the system can warn the driver to pay attention to the road, encouraging safer driving habits.

In summary, this system aims to deliver efficient and reliable driver drowsiness detection by utilizing a combination of a PC, the OpenCV image processing library, and the QT editor. It continuously monitors the driver's face, detects sleepiness indicators, and alerts the driver to take necessary actions, ensuring safer driving experiences.

CHAPTER-05

SYSTEM DESIGN

These assistants are completely adaptable and offer a wide range of services. Each service provides various tasks for users to do, and by combining their data, they deliver a fully working virtual assistant. Here is a brief explanation of how the virtual assistant will operate as well. The initial phase of automated facial recognition comes first. If the user is detected, the process proceeds to the next step; if not, the predefined quaternary is loaded, and the user is also asked to respond to the remaining questions for the registration process. The prompt "User who don't seem to be detected want to register as new user" is then provided.

Once all of the questions have been satisfactorily answered, a facial sample photo is taken, the user is successfully registered, and the appliance then restarts. When a user is identified, the application connects to an InfoBase that has information specific to that person, and the assistant is ready to answer questions. The user has the option to start a discussion by asking a question or acting as they choose. As long-term data for voice recognition, the speech recognition program translates the user's speech into text format and stores that information in the user database. After that, the created text is sent to the chatbot program, also known as a dialogue manager. The knowledge database is then used to produce the appropriate response. Following the generation of the response, the text is translated into speech and the output is then delivered through speakers.

5.1 FACE DETECTION.

- **Face Recognition module**

Objective: It employs machine learning algorithms to scan images for facial recognition. Due to the intricacy of facial structures, there isn't a singular straightforward test to determine the presence of a face. Instead, numerous intricate patterns and characteristics need to be matched. The algorithms divide the task of facial identification into numerous manageable subtasks, each with a simple solution.

Input: The input can be an image, video stream, or captured using a web camera. The output is the identification or verification of the subject depicted in the image or video.

- **Face Detection: Haar Cascade Algorithm**

Haar Cascade Algorithm is an object detection technique utilized for recognizing faces in images or real-time videos. The algorithm incorporates edge or line detection features developed by Viola and Jones in their 2001 research paper titled "Rapid Object Detection using a Boosted Cascade of Simple Features." To train the algorithm, a large dataset of positive images containing faces and negative images without any faces is used.

Here's a breakdown of the steps involved:

Step 1: The first step involves importing the necessary libraries, such as cv2 and numpy, and utilizing the Cascade Classifier function from OpenCV. The function is used to specify the location of the XML file (haarcascade_frontalface_default.xml in this case) containing the pre-trained model for face detection.

Step 2: In the second step, the image is loaded and converted into grayscale. Normally, images are represented in RGB (Red, Green, Blue) format, but OpenCV stores them in BGR (Blue, Green, Red) format. To facilitate image recognition, the BGR channel is converted to grayscale, which simplifies the processing and reduces computational complexity. Grayscale images only contain a single channel representing varying shades of black and white.

Step 3: After converting the image to grayscale, the algorithm proceeds to locate the facial features using the face classifier loaded with haarcascade_frontalface_default.xml. The "detectMultiScale" function, an inbuilt feature of the classifier, is used for this purpose. It returns the coordinates and dimensions of the detected faces within the image.

Facial recognition systems can operate in two modes:

1. Verification or authentication of a facial image: In this mode, the input facial image is compared with a reference facial image of the user requesting authentication. It involves a one-to-one comparison.

2. Identification or facial recognition: This mode involves comparing the input facial image with multiple facial images within a dataset to find a matching user. It entails a one-to-many comparison.

By utilizing Local Binary Pattern (LBP) in combination with histograms, face images can be represented as simple data vectors. LBP, being a visual descriptor, is also applicable for face recognition tasks. The following steps provide a detailed explanation of this process.

- **Local binary pattern histograms (LBPH)**

The LBPH (Local Binary Pattern Histograms) Algorithm for Face Recognition involves the following steps:

Step 1: Parameter Collection

The LBPH Algorithm utilizes four parameters: Radius, Neighbors, Grid X, and Grid Y.

Step 2: Algorithm Training

Initially, the algorithm needs to be trained. This involves using a dataset containing facial images of the individuals to be recognized. Each image is assigned an ID (which can be a number or the person's name). This ID information is utilized by the algorithm to recognize input images and produce corresponding outputs.

Step 3: LBP Operation Application

The first computational step in LBPH is to create an intermediary image that effectively represents the original image by emphasizing the facial features. This is achieved using a sliding window concept, based on the radius and neighbors parameters.

Step 4: Histogram Extraction

Following the previous step, the generated image is divided into multiple grids using the Grid X and Grid Y parameters. The resulting image grid configuration can be observed in the accompanying image.

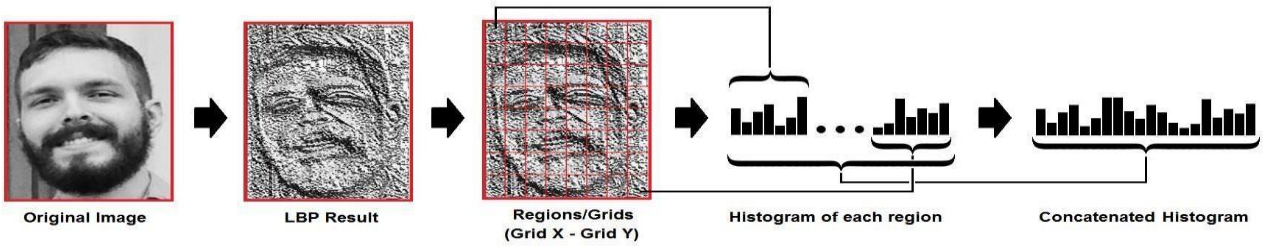


Fig 6 Histogram Extraction

Step 5: Performing the face recognition:

At this stage, the algorithm has undergone training. Each histogram produced serves as a representation for each image within the training dataset. When presented with a new input image, the process is repeated, resulting in the creation of a histogram that represents the image.

To compare histograms and calculate the dissimilarity between them, different approaches can be employed. Examples include measuring the Euclidean distance, utilizing the chi-square method, calculating the absolute value, and so on.

5.2 EYE DETECTION.

In our project, the condition of the patient is determined based on the state of their eyes. Once the face is detected using the facial landmark algorithm, a rectangular box is created around the eyes. This allows us to focus on the movement of the eyes, specifically whether they are open or closed. The initial step in this module is to locate the region of the eyes. The eye region, obtained from the detected face, is then utilized for eye tracking and blink detection. The localized region is depicted in the accompanying figure.

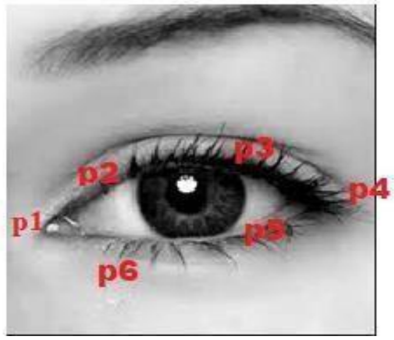


Fig 7 Eye Detection

Each eye is represented by six (x, y) coordinates, starting from the left corner of the person's eye and proceeding clockwise around the eye region. There exists a relationship between the width and height of these coordinates, denoted as p1, ..., p6, which are the 2D locations of facial landmarks. The figure above illustrates the six points used to represent the eyes. To identify the whites of the eyes within the extracted eye region, the time library is employed. If the white region of the eye disappears for a certain duration, it indicates a blink. Blink detection can be calculated using the Eye Aspect Ratio (EAR) method.

5.3 EYE ASPECT RATIO(EAR).

Once the eye is detected within the face, the subsequent task is to identify eye blinks. To achieve this, we employ the Eye Aspect Ratio (EAR) formula. The EAR offers a more elegant solution, as it involves a simple calculation based on the ratio of distances between facial landmarks associated with the eyes. The formula for calculating the EAR is provided below.

$$\mathbf{EAR} = \frac{\| p_2 - p_6 \| + \| p_3 - p_5 \|}{2 \, \| p_1 - p_4 \|}$$

The equation in this context compares the distances between vertical and horizontal eye landmarks. The numerator calculates the vertical distance, while the denominator measures the horizontal distance. The denominator is appropriately weighted, considering that there is only one set of horizontal points compared to two sets of vertical points. The eye aspect ratio remains relatively constant when the eye is open, but rapidly decreases to zero during a blink. By utilizing this straightforward equation, we can assess the ratio of eye landmark distances to determine if a person is blinking.

Please refer to the figure displayed below for illustration. In the top-left image, we observe a fully open eye, resulting in a large and relatively constant eye aspect ratio over time. However, when the person blinks (top-right image), the eye aspect ratio dramatically decreases, approaching zero. The bottom figure depicts a graph representing the eye aspect ratio over time in a video clip. It shows that the eye aspect ratio remains constant initially, then drops significantly close to zero, and subsequently increases again, indicating a single blink has occurred.

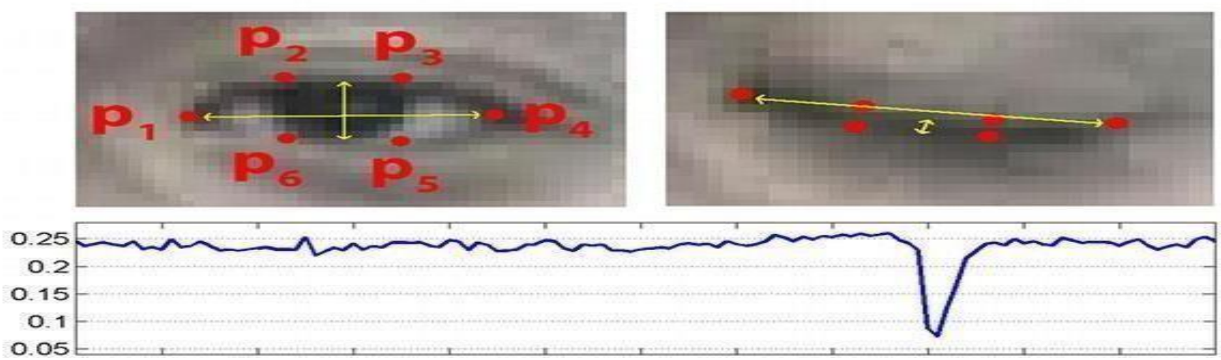


Fig 8 Eye Aspect Ratio

.

5.4 MODULE DESCRIPTION.

The system comprises of 3 phases:

1. **Capturing:**

   - The driver's image is captured using a Logitech camera, which is known for its clarity and affordability.
   - This camera records a video clip focusing on a single frame that captures the driver's eye blink.
   - The captured video is then divided into frames for analysis.

2. **Detection:**

   - In this phase, the driver's face is detected.
   - Facial landmark detection is utilized to locate the driver's face within a frame.
   - Only facial features and structures are detected, while objects like buildings, trees, and bodies are ignored.
   - In our approach, the state of the driver is determined based on the condition of their eyes.
   - The Eye Aspect Ratio (EAR) is calculated, which represents the ratio of the number of eye blinks to the width of the eye.

3. **Correction:**

   - The actual state of the eye is determined, whether it is closed, open, semi-closed, or semi-open.
   - If the eyes are determined to be closed or semi-closed and reach a specific threshold value, a warning message is generated.

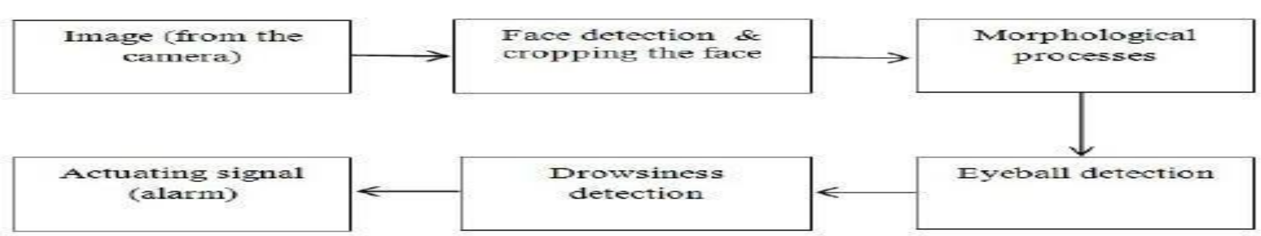


Fig 5.2.5 Block diagram of system module

If the system detects that the eyes are open, the process is repeated until closed eyes are detected.

CHAPTER 06

IMPLEMENTATION AND SOURCE CODE

There are several modules that are implemented and used in the project which are as follows:

6.1 Face Detection.

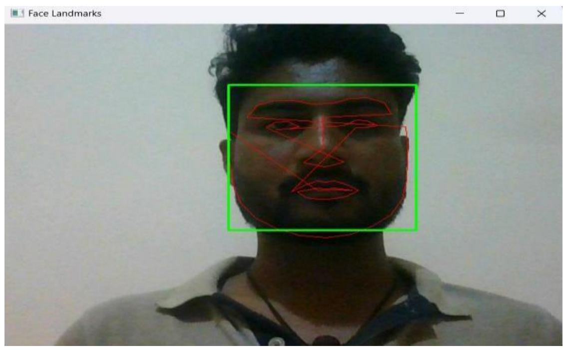


Fig 9 Face Detection Analysis

In this phase, a camera captures the picture of the user at a specific point in time. This picture is then used to train a module, which means that the module learns from this image to recognize and identify the user. The trained module is then stored in a database (DB) for future reference. To identify the user, the system utilizes the Haar Cascade algorithm. This algorithm is a machine learning-based approach used for object detection, particularly for detecting faces in images. It works by analyzing patterns and features of objects, in this case, faces, and can be trained to recognize specific patterns.

The Haar Cascade algorithm relies on a set of pre-trained models stored in XML files, known as Haar cascade classifiers. These classifiers contain the information necessary to detect specific objects, such as faces, based on their unique features. The algorithm uses these classifiers to identify and locate faces within an image.

Once the face is detected using the Haar Cascade algorithm, the system then compares the detected face with the trained models stored in the database. The trained models contain information about the individual's facial features, which were learned during the training phase using their specific image. By comparing the detected face with the trained models, the system can match the image to a specific individual. This matching process involves comparing the detected face's highlights, such as the position of eyes, nose, and mouth, with the corresponding features in the trained models. If the detected face closely matches the features in the trained model, the system can identify the individual associated with that model.

Overall, this process involves capturing an image, training a module with the image to recognize the user's features, utilizing the Haar Cascade algorithm to detect faces, and finally matching the detected face with the trained models stored in the database to identify the individual.

6.2 Eye Detection:

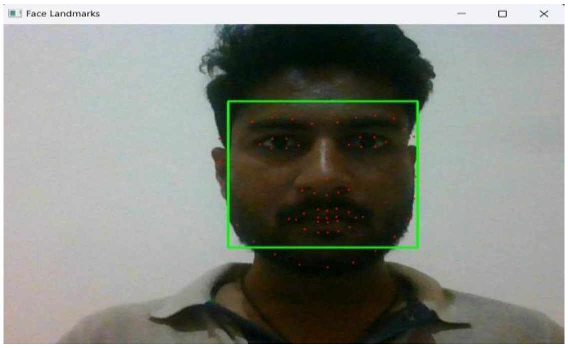


Fig 10 Eye Detection Analysis

The process you're describing involves utilizing a facial landmark algorithm to detect the face in an image or video frame. Once the face is detected, the algorithm identifies specific points on the face known as facial landmarks. These landmarks include key eyes, nose, mouth, and so on. To focus specifically on the eye movement, a rectangular box is created around the eye region using the detected facial landmarks. This box serves as a localized region of interest (ROI) that contains the eyes. By isolating the eye region, its easier to track and analyze eye movements.

Once the eye region is localized, it can be used for various eye-related tasks, such as eye tracking and blink detection. Eye tracking involves monitoring and recording the movement of the eyes over time. This information can be valuable in various applications, such as studying human behavior, developing user interfaces, or analyzing visual attention. Blink detection, on the other hand, focuses on identifying instances when the eyes are closed or blinked. By continuously monitoring the eye region, the algorithm can analyze changes in eye appearance and determine if a blink has occurred. Blink detection has applications in various fields, including driver drowsiness detection, attention monitoring, and human-computer interaction.

Overall, by localizing the eye region through facial landmark detection, it becomes possible to concentrate on eye movements, track eye gaze, and detect blinks. These capabilities have significant practical implications in fields such as computer vision, human-computer interaction, and behavioral analysis.

6.3 Eye Aspect ratio (EAR):

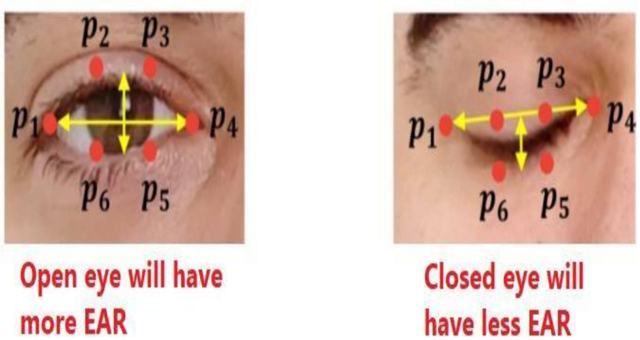


Fig 11 Eye Aspect Ratio Analysis

The EAR formula typically involves calculating the Euclidean distances between a set of landmarks. The specific landmarks used may vary depending on the system and the facial landmark detection algorithm being employed. However, a common set of landmarks used for eye blink detection includes the inner corner of the eye, the outer corner of the eye, and points on the top and bottom eyelid.

EAR = (||P2 - P6|| + ||P3 - P5||) / (2 * ||P1 - P4||)

Here, P1, P2, P3, P4, P5, and P6 represent the coordinates of the corresponding facial landmarks.

By monitoring the changes in the calculated EAR over time, you can detect eye blinks when the EAR falls below a certain threshold. A blink is typically identified when the EAR value decreases and then increases again within a short duration. Using the Eye Aspect Ratio formula provides an elegant and efficient solution for detecting eye blinks in facial recognition systems, making it a widely used technique in various applications such as drowsiness detection, attention monitoring, and human-computer interaction.

- **NumPy:** NumPy is a broadly useful cluster preparing bundle which gives a multidimensional exhibit item, and devices for working with these clusters. It's an essentialbundle for logical figuring with the python.
- **OpenCV:** This module sorts the image and analysis the video like face detection, license plate reading, optical character recognition, and so forth.
- **Cv2:** This python method loads an image from the specified file and read it. It consists of three flags which are as follows
- i.e., cv2.IMREAD_COLOR, cv2.IMREAD_GRAYSCALE, cv2. _UNHANGED
- **OS:** This module provides a way of using operating system dependent functionality. OS module allows us to interface with the underlying operating system that on python runs.

6.4 SOURCE CODE

```
from scipy.spatial import distance as dist
 from imutils.video import VideoStream
 from imutils import face_utils from threading
 import Thread
 import numpy as np
 import argparse
 import imutils
 import time
import dlib
import cv2
import os
import cv2
import numpy as np
import dlib from math
import hypot
from imutils import face_utils



face_landmark_path = 'shape_predictor_81_face_landmarks.dat'

K = [6.5308391993466671e+002, 0.0, 3.1950000000000000e+002,
    0.0, 6.5308391993466671e+002, 2.3950000000000000e+002,
    0.0, 0.0, 1.0]
D = [7.0834633684407095e-002, 6.9140193737175351e-002, 0.0, 0.0, 1.3073460323689292e+000]

cam_matrix = np.array(K).reshape(3, 3).astype(np.float32)
dist_coeffs = np.array(D).reshape(5, 1).astype(np.float32)

object_pts = np.float32([[6.825897, 6.760612, 4.402142],
                         [1.330353, 7.122144, 6.903745],
```

```
                    [-1.330353, 7.122144, 6.903745],
                    [-6.825897, 6.760612, 4.402142],
                    [5.311432, 5.485328, 3.987654],
                    [1.789930, 5.393625, 4.413414],
                    [-1.789930, 5.393625, 4.413414],
                    [-5.311432, 5.485328, 3.987654],
                    [2.005628, 1.409845, 6.165652],
                    [-2.005628, 1.409845, 6.165652],
                    [2.774015, -2.080775, 5.048531],
                    [-2.774015, -2.080775, 5.048531],
                    [0.000000, -3.116408, 6.097667],
                    [0.000000, -7.415691, 4.070434]])

reprojectsrc = np.float32([[10.0, 10.0, 10.0],
                           [10.0, 10.0, -10.0],
                           [10.0, -10.0, -10.0],
                           [10.0, -10.0, 10.0],
                           [-10.0, 10.0, 10.0],
                           [-10.0, 10.0, -10.0],
                           [-10.0, -10.0, -10.0],
                           [-10.0, -10.0, 10.0]])

line_pairs = [[0, 1], [1, 2], [2, 3], [3, 0],
              [4, 5], [5, 6], [6, 7], [7, 4],
              [0, 4], [1, 5], [2, 6], [3, 7]]



def get_head_pose(shape):     image_pts = np.float32([shape[17], shape[21], shape[22], shape[26], shape[36],
                            shape[39], shape[42], shape[45], shape[31], shape[35],                          shape[48], shape[54],
shape[57], shape[8]])

    _, rotation_vec, translation_vec = cv2.solvePnP(object_pts, image_pts, cam_matrix, dist_coeffs)
```

```
    reprojectdst, _ = cv2.projectPoints(reprojectsrc, rotation_vec, translation_vec, cam_matrix,
dist_coeffs)

    reprojectdst = tuple(map(tuple, reprojectdst.reshape(8, 2)))

    # calc euler angle
    rotation_mat, _ = cv2.Rodrigues(rotation_vec)     pose_mat = cv2.hconcat((rotation_mat, translation_vec))
    _, _, _, _, _, _, euler_angle = cv2.decomposeProjectionMatrix(pose_mat)

    return reprojectdst, euler_angle




def midpoint(p1 ,p2):
    return int((p1.x + p2.x)/2), int((p1.y + p2.y)/2)

font = cv2.FONT_HERSHEY_PLAIN

def eye_aspect_ratio(eye):
        A = dist.euclidean(eye[1], eye[5])B =
        dist.euclidean(eye[2], eye[4])

        C = dist.euclidean(eye[0], eye[3])ear = (A
        + B) / (2.0 * C)
        return ear

    def final_ear(shape):
        (lStart, lEnd) = face_utils.FACIAL_LANDMARKS_IDXS["left_eye"] (rStart,
        rEnd) = face_utils.FACIAL_LANDMARKS_IDXS["right_eye"]

        leftEye = shape[lStart:lEnd] rightEye
```

```
    = shape[rStart:rEnd]

    leftEAR = eye_aspect_ratio(leftEye) rightEAR
    = eye_aspect_ratio(rightEye)

    ear = (leftEAR + rightEAR) / 2.0return
    (ear, leftEye, rightEye)

def lip_distance(shape): top_lip =
    shape[50:53]
    top_lip = np.concatenate((top_lip, shape[61:64]))

    low_lip = shape[56:59]
    low_lip = np.concatenate((low_lip, shape[65:68]))

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

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


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

EYE_AR_THRESH = 0.3
EYE_AR_CONSEC_FRAMES = 10
YAWN_THRESH = 20
alarm_status = False
```

```
alarm_status2 = False

EYE_AR_CONSEC_FRAMES = 10
YAWN_THRESH = 20
alarm_status = False
alarm_status2 = False

saying = False COUNTER
= 0

print("-> Loading the predictor and detector...")detector
= dlib.get_frontal_face_detector()
detector1 = cv2.CascadeClassifier("haarcascade_frontalface_default.xml")    #Faster but lessaccurate
predictor = dlib.shape_predictor('shape_predictor_68_face_landmarks.dat')


print("-> Starting Video Stream")
vs = VideoStream(src=args["webcam"]).start()
#vs= VideoStream(usePiCamera=True).start()       //For Raspberry Pi
time.sleep(1.0)

while True:

    frame_1 = vs.read()
    new_frame_1 = np.zeros((500, 500, 3), np.uint8)
    gray = cv2.cvtColor(frame_1, cv2.COLOR_BGR2GRAY)faces =
    detector(gray)
    if len(faces) > 0:
            shape = predictor(frame_1, faces[0]) shape =
            face_utils.shape_to_np(shape)
            reprojectdst, euler_angle = get_head_pose(shape)for (x, y)
```

```
            in shape:
                cv2.circle(frame_1, (x, y), 1, (0, 0, 255), -1)

##              for start, end in line_pairs:
##                  cv2.line(frame_1, reprojectdst[start], reprojectdst[end], (0, 0, 255))

            # cv2.putText(frame_1, "X: " + "{:7.2f}".format(euler_angle[0, 0]), (20, 20),
cv2.FONT_HERSHEY_SIMPLEX,
            #           0.75, (0, 0, 0), thickness=2)
            # cv2.putText(frame_1, "Y: " + "{:7.2f}".format(euler_angle[1, 0]), (20, 50),
cv2.FONT_HERSHEY_SIMPLEX,
            #           0.75, (0, 0, 0), thickness=2)
            # cv2.putText(frame_1, "Z: " + "{:7.2f}".format(euler_angle[2, 0]), (20, 80),
cv2.FONT_HERSHEY_SIMPLEX,
            #           0.75, (0, 0, 0), thickness=2)# if
            euler_angle[0, 0] > 10 :
#print('Head bend is front detected')
#cv2.putText(frame_1, "Head bend is front detected Drowsiness",(20, 20),
cv2.FONT_HERSHEY_SIMPLEX,
            #           0.75, (0, 0, 255), thickness=2)# if
            euler_angle[1, 0] > 3 :
            #   print('Head bend is rigth detected ')
            #   cv2.putText(frame_1, "Head bend is right detected Drowsiness",(20, 20),
cv2.FONT_HERSHEY_SIMPLEX,
            #           0.75, (0, 0, 255), thickness=2)# if
            euler_angle[2, 0] > 10 :
            #   print('Head bend is left detected ')
            #   cv2.putText(frame_1, "Head bend is left detected Drowsiness",(20, 20),
cv2.FONT_HERSHEY_SIMPLEX,
            #           0.75, (0, 0, 255), thickness=2)#
        cv2.imshow("demo", frame_1)
        frame = vs.read()
```

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

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

    #rects = detector(gray, 0)
    rects = detector1.detectMultiScale(gray, scaleFactor=1.1, minNeighbors=5, minSize=(30,
30),flags=cv2.CASCADE_SCALE_IMAGE)

    #for rect in rects:
    for (x, y, w, h) in rects:
        rect = dlib.rectangle(int(x), int(y), int(x + w),int(y + h))

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

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

        distance = lip_distance(shape)

        leftEyeHull = cv2.convexHull(leftEye)
        rightEyeHull = cv2.convexHull(rightEye)
        cv2.drawContours(frame, [leftEyeHull], -1, (0, 255, 0), 1)
        cv2.drawContours(frame, [rightEyeHull], -1, (0, 255, 0), 1)

        lip = shape[48:60]
cv2.drawContours(frame, [lip],
-1, (0, 255, 0), 1)


if ear < EYE_AR_THRESH:
```

```
    COUNTER += 1

    if COUNTER >= EYE_AR_CONSEC_FRAMES:
        print('DROWSINESS ALERT!')

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

else:
    COUNTER = 0



if (distance > YAWN_THRESH): print('Yawn
    ALERT!') print("actual:"+str(distance))
    print("yawn:"+str(YAWN_THRESH))
    cv2.putText(frame, "Yawn Alert", (10, 30),
            cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)

else:
    cv2.putText(frame, "EAR: {:.2f}".format(ear), (300, 30),
            cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)
    cv2.putText(frame, "YAWN: {:.2f}".format(distance), (300, 60),
            cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)



cv2.imshow("Frame", frame)
key = cv2.waitKey(1) & 0xFF
if key == ord("q"):break

cv2.destroyAllWindows()vs.stop()
```

CHAPTER 07

TESTING

LAYERS IN TESTING A DRIVER ALERTNESS DETECTION

Testing is described as a method of assessing an entire system or its part(s) in order to determine whether it complies with the required specifications. Testing is the process of running a system to find any flaws, omissions, or gaps from the actual specifications.

A software engineer needs to comprehend the fundamental idea behind software testing before adopting techniques to create effective test cases. The client requirements should be able to be traced back to each test. There are different methods that are used for software testing are,

7.1 BLACK BOX TESTING.

Black-box testing is the process of testing an application without having any knowledge of its internal operations. The individual conducting the test lacks knowledge of the source code and is unaware of the system structure. A tester will often engage with the system's user interface duringa black-box test by providing inputs and evaluating outputs while not being aware of how or wherethe inputs are processed.

7.2 WHITE BOX TESTING

White-box testing is a thorough analysis of the code's internal structure and organization. Glass testing and open-box testing are various synonyms for white-box testing. A tester requires to be familiar with the internal operations of the code in order to carry out white-box testing on an application. The tester must examine the source code to identify the particular component or section of code that is acting improperly.

7.3 UNIT TESTING

The intended audience for this test is creators of voice-based technologies. To make sure the code is operating properly in solitude, they have to execute unit testing. Unit testing needs to be quick so it doesn't slow down your programming rate while you're doing it. The goal of unit testing is to verify that your source code and logic are sound, negating the necessity to visit the cloud (where the majority of voice applications' backends reside) or make calls to actual outside services. The unit testing tool you select must be one that's mocks-compatible and ideally local-running.

UNIT TEST CASES:

Sl # Test Case : -	UTC-1
Name of Test: -	Image or video capture
Items being tested: -	Input Image
Sample Input: -	Camera Stream
Expected output: -	Should Capture input image
Actual output: -	Image Captured Successful
Remarks: -	Pass.

Fig 12 Unit Test Case-1

Sl # Test Case : -	UTC-2
Name of Test: -	Eye Detection
Items being tested: -	Labelling
Sample Input: -	Image or video
Expected output: -	Eye Should detected as facial Landmark
Actual output: -	Facial Landmark is detected
Remarks: -	Test Passed

Fig 13 Unit Test Case-2

Sl # Test Case : -	UTC-3
Name of Test: -	Eye Blink Detection
Items being tested: -	Blinks
Sample Input: -	Image or video
Expected output: -	Blink State detected
Actual output: -	Same as Expected
Remarks: -	Test Passed

Fig 14 Unit Test Case-3

Sl # Test Case : -	UTC-4
Name of Test: -	Voice Intimation
Items being tested: -	Voice output for No blinks
Sample Input: -	Image or video
Expected output: -	Voice Output for different blinks
Actual output: -	Same as Expected
Remarks: -	Test Passed

Fig 15 Unit Test Case-4

7.4 INTEGRATED TESTING

Individual software components are merged and tested together during integration testing. This type of testing aims to detect problems that arise from the interaction between integrated units. Integration testing is supported by the utilization of test drivers and test subs. There are two techniques to conduct integration testing: Top-down and bottom-up integration testing are both used.

- **BOTTOM-UP INTEGRATION**

This testing starts with unit testing and progresses to tests of modules, which are assembled from units at progressively higher levels.

- **TOP DOWN INTEGRATION**

  In this testing, the most advanced modules are checked initially, followed by tests on simpler modules.Top-down testing is typically carried out after bottom-up testing in a thorough software creation environment. The procedure is completed with numerous testing of the entire program, ideally insituations created to closely resemble real-world circumstances.

Sl # Test Case : -	ITC-1
Name of Test: -	Image Capture and automated Facial Landmark Detection
Item being tested: -	Capture Image and automated Face and Eye Labelling
Sample Input: -	Video Stream
Expected output: -	Face and Eye Detection
Actual output: -	Functioned Properly
Remarks: -	Pass.

Fig 16 Integrated Test Case1

Sl # Test Case : -	ITC-2
Name of Test: -	Eye Blink Detection and Voice output
Item being tested: -	Input Image or Video and Appropriate voice output
Sample Input: -	Video Stream
Expected output: -	Voice output For No Blink or Drowsy State
Actual output: -	Functioned Properly
Remarks: -	Pass.

Fig 17 Integrated Test Case2

7.5 SYSTEM TESTING

The evaluation of software and hardware systems is conducted on a fully integrated system to determine its compliance with the specified requirements. System testing comes under the purview of black-box testing and, as such, shouldn't require any understanding concerning the inner workings of the logic or program. The following justifies why system testing is crucial:

System testing is the first stage in the software development life cycle where the application is thoroughly tested to ensure that it complies with the functional and technical requirements.

- The testing of the program takes place in a setting substantially similar to the one in which it will be used in production.
- System testing allows us to test, validate, and test both the business and the technology

Sl # Test Case : -	STC-1
Name of Test: -	System testing in various versions of OS
Item being tested: -	OS compatibility.
Sample Input: -	Execute the program in windows XP/ Windows-7/8
Expected output: -	Performance is better in windows-7
Actual output: -	Same as expected output, performance is better in windows-7
Remarks: -	Pass

Fig 18 System Test Case-1

CHAPTER 08

CONCLUSION AND FUTURE ENHANCEMENTS

8.1. CONCLUSION

"An innovative driver alertness detection system has been proposed for real-time fatigue detection. The method effectively identifies eye blinks and drowsiness by leveraging image processing algorithms to track the position of the eyes. This non-invasive approach offers a seamless way to detect drowsiness without causing annoyance or interference. A face recognition algorithm was employed to measure the blink rate accurately.

The algorithm demonstrated successful eye detection under medium and high illumination conditions, irrespective of gender or age. However, optimal detection required the camera to be positioned directly in front of the driver. To overcome potential issues caused by poor detection due to insufficient lighting, a night vision camera was implemented, ensuring reliable results even in low light conditions. A buzzer indicator was employed to alert the driver and ensure safe driving practices.

Drunk driving accidents remain a significant problem in modern times. This research offers an advanced solution that can be easily implemented in vehicles, incorporating multi-stage testing to prevent accidents caused by intoxicated driving. Such detectors hold great importance in the future, particularly in conjunction with IoT technology.

This project focuses on the hardware programming of an IoT device to serve as an alcohol detector and preventive measure. The system proposes a method to detect the usage of handheld cell phones while driving, employing a ring indicator. The system's output can be a warning to redirect the driver's attention solely to the vehicle and the road, or it can send a notification to a transport company or activate a buzzer. Alcohol sensors are used to detect alcohol consumption up to a predetermined threshold.

8.2 FUTURE ENHANCEMENT

The cell phone detection system can be enhanced further by integrating a hybrid solution with machine learning for movement detection. Additional features, such as horizontal movement, the area of connected components, and the dimensions of the detected region, can be extracted using Optical Flow. Increasing the frame per second processing and improving image resolution are also potential areas for improvement." detection system was proposed based on fatigue detection in real-time. The proposedmethod easily detects the eye blink and the drowsiness. Information about the eyes position was obtainedthrough image processing algorithms. Image processing offers a noninvasive approach to detect drowsiness without any annoyance and interference. An algorithm for performing face recognition wasused. It was found that with this algorithm, a good measurement of the blink rate was obtained.

CHAPTER 10

SNAPSHOTS

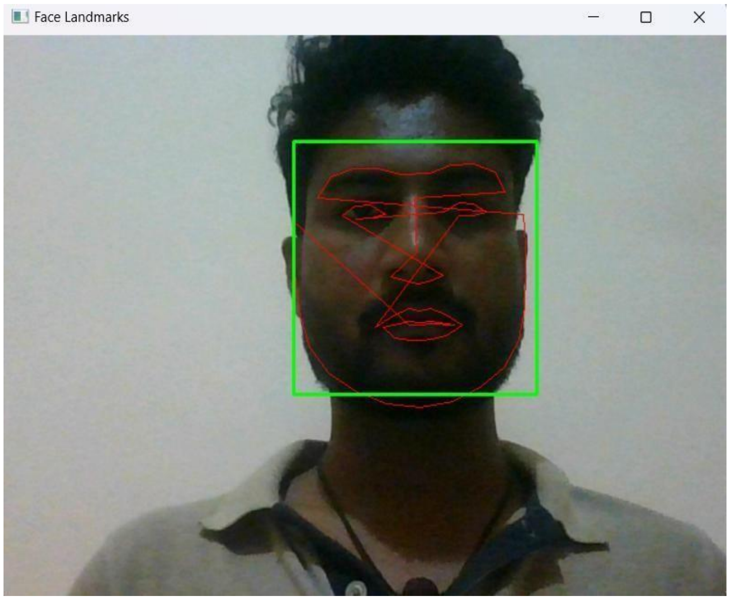


Fig 19 Face landmark and feature detection

Fig 20 Eye Detection

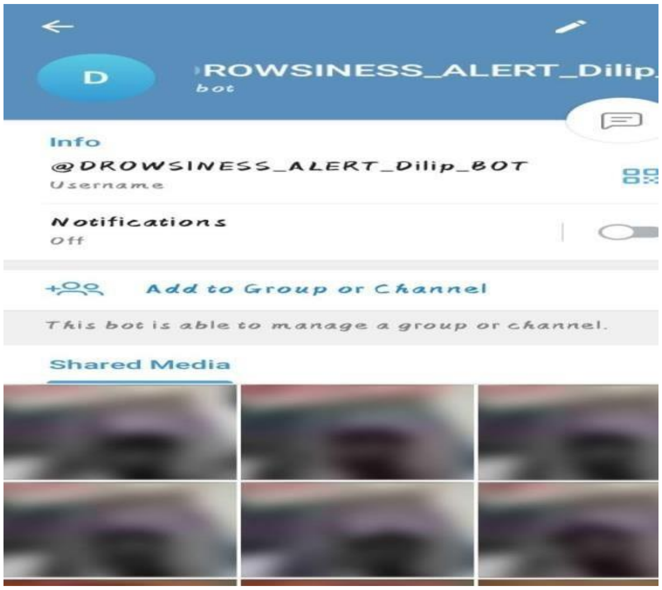


Fig 21 Telegram Bot

Fig 22 Message Alert with picture

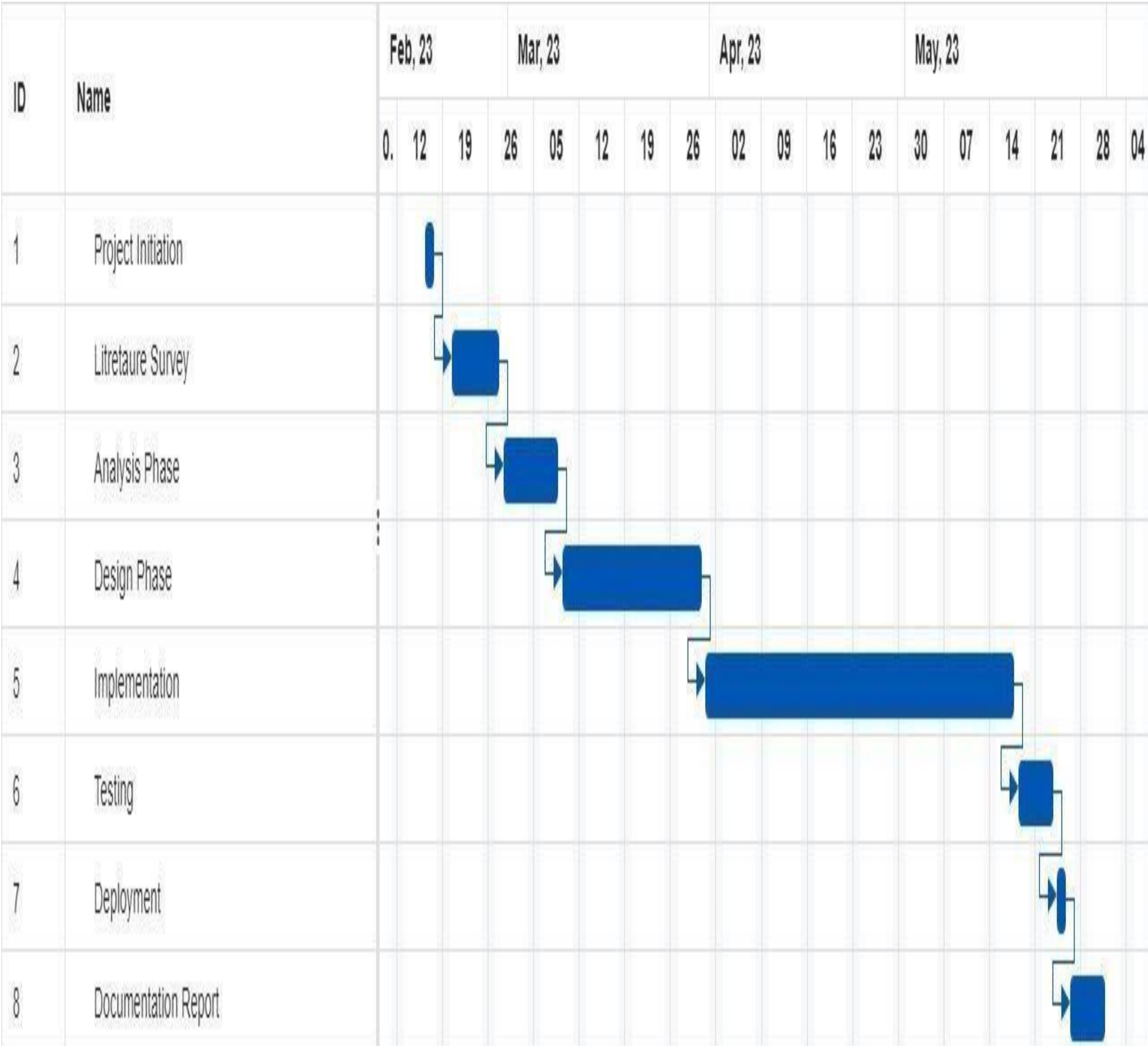


Fig 23 Timeline by Gantt Chart

CHAPTER 11

REFERENCES

[1] Data from NCRB, Government of India, available at http://ncrb.nic.in/StatPublications/ADSI/ADSI2014/ADSI2014.asp

[2] "Internet of Things Global Standards Initiative". ITU. Retrieved 26 June 2015. - http://www.itu.int/en/ITU-T/gsi/iot/Pages/default.aspx

[3] Z. Xiaorong et al, ―The Drunk Driving Automatic Detection System Based on Internet of Things‖, International Journal of Control and Automation.

[4] J. Dai, J. Teng, X. Bai, Z. Shen, and D. Xuan. "Mobile phone based drunk driving detection." In 2010 4th International Conference on Pervasive Computing Technologies for Healthcare, pp. 1-8. IEEE, 2010

[5] A. R. Varma, S. V. Arote, C. Bharti, and K. Singh. "Accident prevention using eye blinking and head movement." IJCA Proceedings on Emerging Trends in Computer Science and Information Technology-2012 (ETCSIT2012) etcsit1001 4 (2012)

[6] V. Savania, H. Agravata and D. Patela , ―Alcohol Detection and Accident Prevention of Vehicle‖, International Journal of Innovative and Emerging Research in Engineering, Volume 2, Issue 3, 2015, pp 55-59

[7] http://en.wikipedia.org/wiki/Global_Positioning_System.

[8] http://www.dadss.org/breath-based-technology.

[9] http://www.dadss.org/touch-based-technology.

[10] S sahabiswas, S sourav, ―Drunken driving detection and prevention models using Internet of things‖, Information Technology, Electronics and Mobile Communication Conference (IEMCON), 2016 IEEE 7th Annual, pp.1-16, IEEE,2016

[11] Open CV 2.4.11.0 documentation; Open CV Tutorials; objdetect module;Object Detection.

[12] Viola, Jones: Robust Real-time Object Detection, IJCV 2001.

[13] Open CV Computer Vision with Python; Tracking Faces with Haar Cascades

[14] S. Motorist, Driver Fatigue is an important cause of Road Crashes, Smart Motorist.

[15] Driver Fatigue and Road Accidents: A Literature Review and Position Paper

[16] Royal Society for the Prevention of Accidents, 2001.

[17] Road and Traffic Authority (RTA) Annual Report, Sidney, 2008.

[18] B. Praveenkumar, K Mahendrakan. "Prevention of Accident Due To Drowsy By Using Eye Blink." International Journal of Innovative

Research in Science, Engineering and Technology(IJIRSET)Volume 3, May 2014.

[19] Stan, Ovidiu, Liviu Miclea, and Ana Centea. "Eye-Gaze Tracking Method Driven by Raspberry Pi Applicable in Automotive Traf¿c Safety. "Arti¿cial Intelligence, Modelling and Simulation (AIMS), 2014 2nd International Conference on IEEE, 2014.

[20] Friedrichs, Fabian, and Bin Yang. "Camera-based drowsiness reference for driver state classi¿cation under real driving conditions." Intelligent Vehicles Symposium (IV), 2010,IEEE. IEEE, 2010.

[21] "Eye detection and recognition in the fatigue warning system." Intelligent Networks and Intelligent Systems (ICINIS), 2010 3rd International Conference on IEEE, 2010.

[22] Turan, Garima, and Sheifali Gupta. "Road Accidents Prevention system using

Driver‘s Drowsiness Detection." International Journal of Advanced Research in Computer Engineering and Technology (IJARCET)

JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Driver Alertness Detection On Real Time Face And Eye Recognition

[1]Dr. Sulaiman Syed Mohammed, Dilip Kumar M[2], Gagana G[3], Galla Gireesh[4], EarannaRaju C[5], DhurWahidQuadri[6],

[1]Associate Professor, [23456]Student, Department of Computer Science and Engineering, Presidency University (Deemed to be University), Karnataka. Bengaluru, India.

Abstract: This paper is an overview of research that was done in the field of computer engineering to create a system for detecting driver drowsiness in order to avoid accidents caused by fatigued or sleepy drivers. A restricted implementation of the numerous strategies that are discussed in the thesis-es on the subject was addressed in the novel along with results and solutions. In the current state of traffic, the document addresses the many methods for detecting weariness and their effectiveness in averting accidents. The report also summarises the authors' observations in order to support further optimisation in the concerned area and produce the utility with greater effectiveness for a safer road.

Key words: Driver weariness; eye detection; yawn recognition; blink frequency; and fatigue while driving.

1.INTRODUCTION

Humans have always created tools and devised methods to make life easier and safer, whether for routine tasks like getting to and from work or for more intriguing ones like aviation. Our dependence on transportation increased tremendously as technology continued to progress. Our lives as we know them have been significantly impacted by it. Even our grandparents couldn't have imagined how quickly we can now travel. Today, practically everyone uses some kind of transportation every day. While some people are wealthy enough to buy a car, others go by public transportation. Regardless of their socioeconomic standing, drivers are nevertheless subject to certain laws and codes of behavior. One of them is continuing to be active and awake while driving.

Every year, thousands of fatalities are linked to this magnificent innovation because we neglected our obligations to promote safer travel. To most people, it can seem insignificant, yet adhering to traffic laws and regulations is crucial.

The most powerful object on the road is an automobile. When used carelessly, it can be harmful and occasionally endanger the lives of other road users as well. Not acknowledging when we are too sleepy to operate a vehicle is one form of irresponsibility. Numerous scholars have written research papers on driver neglect to track and stop a catastrophic effect from such carelessness. Not acknowledging when we are too sleepy to operate a vehicle is one form of irresponsibility. Plenty of researchers have written research papers on driver sleepiness detection systems in an effort to monitor and stop a disastrous outcome from such irresponsibility. However, sometimes the system's observations and points are not precise enough. Therefore, this review is being produced to provide information and a different viewpoint on the research conducted, in order to improve their implementations and further optimise the solution.

Our most recent numbers show that 148,707 persons perished in car-related incidents in India alone in 2015 [1]. At least 21% of these accidents were brought on by tired drivers making mistakes [1, 2, 3]. This number may even be lower given that, among the several factors that can cause an accident, weariness is sometimes severely underrated as a contributing factor. In developing nations like India, fatigue paired with poor infrastructure is a prescription for disaster.

Due to the nature of fatigue as a safety issue, no nation in the world has yet made significant progress in addressing it. Contrary to alcohol and drugs, which have obvious key signs and tests that are simple to obtain, fatigue is typically exceedingly difficult to measure or detect. The most effective ways to address this issue are likely to be raising awareness of incidents linked to driver drowsiness and encouraging them to admit it when necessary. Since the latter is impossible without the former due to the former's difficulty and higher cost, as well as due to the former's high financial benefit, long hours behind the wheel. The earnings associated with a job rise as the need for it rises, which encourages an increase in the number of individuals who take the position. In operating a vehicle at night, this is true. Money induces reckless behavior in drivers, such as night time driving despite exhaustion. This is primarily a result of the drivers themselves being unaware of the enormous risk involved with driving while fatigued. Although several nations have placed limits on how long a driver can operate a vehicle

2. LITERATURE SURVEY

A survey done by National Highway Traffic Safety Administration estimated that there were 56,000 sleep related road crashes in the U.S.A in 1996. Another survey done in 2007 says that 18% of accidents involved fatigue as the main factor. In Britain up to 20% of serious road accidents were caused due to fatigue. Similarly, survey done by the Road and Traffic Authority states that in the year 2007, fatigue contributed to 20% of accidents caused on road. Accidents due to drowsy was prevented and controlled when the vehicle is out of control. Also, the drunken driving is detected by using alcohol detector in the vehicle. The term used here for the identification that the driver is drowsy is by using eye blink of the driver.

These types of accidents occurred due to drowsy and driver could not be able to control the vehicle, when the driver wakes. The drowsiness was identified by the eye blink closure rate through infrared sensor worn by driver by means of spectacles frame. If the driver is in drowsy state, then the system will give buzzer signal and the speed of the vehicle was reduced and the obstacle sensor is used to sense the adjacent vehicle to avoid collision with that, and if there is no vehicle in left adjacent side then the vehicle move to the left side of the road by auto steering and controlling and vehicle was parked with prior indications. In the recent years, many researchers worked on these devices and few approaches have been reported .

One of the suggested methods is to monitor the movement of the vehicle to detect drowsiness of the driver. However this method has limitations as the results are influenced by the type of vehicle and the condition of road. Another method is to process the electrocardiogram (ECG) signals of driver. This approach also has limitations as ECG probes shall always be connected to the driver‘s body. That would disturb the driver. Few researches tried to assess the fatigue factor by monitoring the eye blink rate of the driver. Successful detection of eye blink rate has been the interest of many researchers proposed methods based on combination of projection and the geometry feature of iris and pupil. T.D Orazio and Z.Zhang use the fact that the iris and pupil are darker than skin and white part of the eye.

Y.Lei proposed an algorithm based on the cascade AdaBoost classifier. T.Hong, a gray level image of an eye is converted to a binary image, using a predetermined threshold. Then, based on the number of black and white pixels of this binary image, state of the eye is determined.

The algorithm presented by Ms. Devi used the Hough Transform to detect the iris and to determine openness of the eye. Some researchers are based on the projection of the image, to determine the state of an eye. Z. Liu the vertical projection of the image of both eyes is used. The horizontal projection image of an eye is used to determine the interval between eyebrows and eyelids and to recognize the state of an eye. The horizontal projection of the image of a face is calculated to determine state of an eye.

Some works also are based on "Support Vector Machine" (SVM) classifier. The SVM classifier is used to detect state of the eye. F Smach used SVM classifier and Gabor filter to extract eye characteristic. In the above methods, the authors used some conditions which make some difficulties in the eye state recognition. The system detects the fatigue symptoms of the driver which consists of an eye blink sensor for driver blink attainment and an adaptive speed controller designed using stepper motor for providing actual positioning of the throttle valve to adjust the speed of vehicle. Advanced technology offers some hope to avoid these up to some extent. This paper involves measure and control of accidents by using both alcohol sensor and IR sensor.

It uses remotely located charge-coupled-device cameras with active infrared illuminators to acquire video images of the driver. Various visual signs that typically characterize the level of alertness of a person were extracted in real-time and systematically combined to infer the fatigue level of the driver. The visual cues employed characterize eyelid, gaze, head movements and facial expressions. A probabilistic model was developed to model human fatigue and to estimate fatigue based on the visual cues. The simultaneous use of visual cues and their systematic combination earns an accurate fatigue characterization. This system was validated under real-life fatigue conditions with the human subjects of different ethnic backgrounds with or without glasses; and beneath different illumination conditions. It was found to be reasonably reliable, and accurate in fatigue characterization.

The computer vision based method of Seshadri determine if a driver is holding a cell phone close of his/her ears using the Supervised Descent Method (SDM) which it tracks some facial landmarks to extract a crop of regions of interest (ROI) (the driver's ear region). Features are extracted from the ROIs and the phone usage is detected using previously trained classifiers. The system can be processed in near real time. The approach of Yang send beeps of high frequency through the car sound equipment, network Bluetooth, and use software running on the phone for capturing and processing sound signals. The beeps are used to calculate the position where the cell phone is, and then we know when the driver (or another passenger in the car) is talking on it. The proposal achieved a classification accuracy of more than 90%. This approach works with hands-free usage, but it depends on the operating system and mobile phone brand, and the software has to be continually enabled by the driver.

3. PROPOSED METHODOLOGY

Authors of several study publications have offered a variety of approaches for successfully identifying driver weariness. Accurate face and eye detection for the purpose of detecting weariness can be achieved using the Python OpenCV package. This makes the system very simple to use, but it also makes the face detection process incredibly slow. This method can be made at least twenty times faster using various strategies, such as examining changes in successive frames to recognise face and eyes [6]. The entire drowsiness detection procedure can be made considerably more reliable [7] by using a method for human eye recognition that employs the Circular Hough Transform (CHT) for precise iris detection. The centre and radius of the iris are determined using CHT, which is crucial for determining the distance between the eyelids. Another technology employs video data to analyse the mouth and eyes for mouth monitoring and eye tracking to better gauge driver fatigue. Because only brightness can differentiate faces with varied complexions, the two components of YCbCr are Both luminance (EyeMapL) and chrominance (EyeMapC) can aid in better facial complexion detection because after luminance is removed from the Eye Map, it is much easier to distinguish between faces with various skin tones. To determine whether the eyes are open or closed, colour spaces like the HSV graph are employed. This information is then used to calculate the PERCLOS parameter, which is used to determine tiredness. Structural Similarity Measure SSIM), which performs better than any other standard metrics, can be used to detect eyes. By examining sleepiness levels, detection with this outcome provides information that aids in determining whether the alarm should be set.

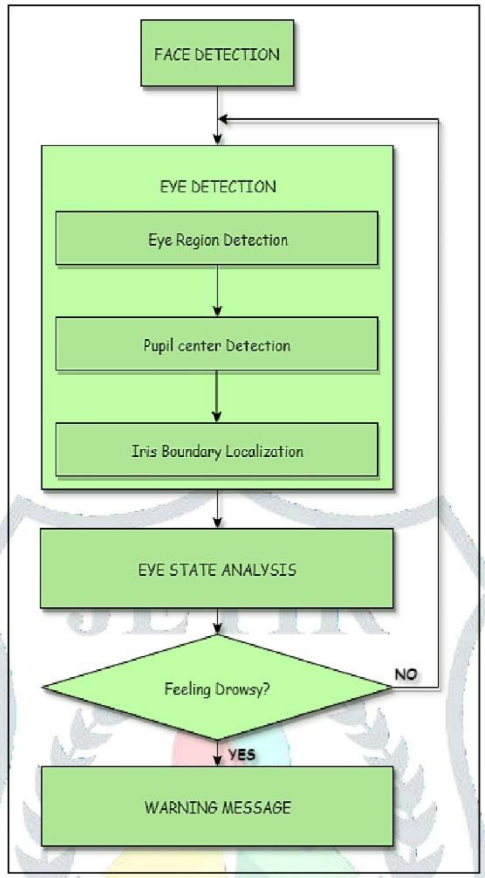


Figure 1: The mentioned system for alertness detection [16].

4. RESULTS

Modern technology must be used in non-intrusive, precise methods for a higher acceptance rate in order to offset the impacts of sleepiness. There is a tonne of research for putting such methods into practise. Many pieces of software, such as TensorFlow and OpenCV, are available that can assist in the recognition of faces and their many components, making the implementation of fatigue detection simpler. To determine drowsiness, there are methods like tracking the space between your eyes over time. Others include things like gaze detection and blinking rate. If a person yawns repeatedly, the next step is to watch their mouth for signs of weariness. Other methods involve tracking vehicle data to identify the driver's sleepiness. It involves erratically moving the steering wheel, sharp turns, abrupt acceleration or slowdown, lateral lane position, etc.

For the purpose of sleepiness detection, the following approaches and findings are explained in detail:

1) A technique based on eyelid movement
2) Analysis on eye state using the Circular Hough Transform (CHT)
3) Yawning and closing of eyes
4) Analysis of Open/Closed Eyes
5) A. Technique based on eyelid movement

Eyelid movement-based techniques can identify faces and eyes more quickly than conventional methods, which speeds up the entire fatigue detection procedure [9]. The process of detecting eyes and faces is twenty times faster when using motion information to trace the face in addition to mask matching and diamond searching techniques. This method focuses on the drivers' eyelid movements to assess their level of weariness. There can only be one of the driver's four eyelid states in two consecutive frames.

The four eyelid states are:

1) Totally closed condition
2) Fully opened condition
3) Partially opened condition
4) State that is partially closed.

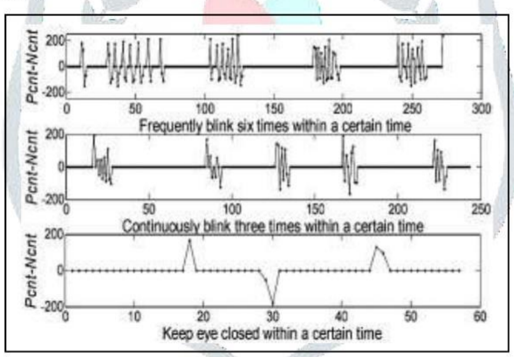
Sclera, iris, and eyelid are the three components that make up an eye. Each of these components may be easily differentiated when two consecutive frames are viewed and the temporal difference image is created. As a result, the latter two states of the eyelids can be distinguished by monitoring the transition of the eyelids from a closed to an opened state or from an opened to a closed state in accordance with the shift in the iris's colour gradient from dark to light (Ncnt) or from light to dark (Pcnt), respectively [6]. If a black peak appears during an eyelid movement before a brilliant peak, it means that the eye's condition altered, first opening from a closed position before closing. The formula below can be utilised to determine the present condition of the eyes.

The state of the eyelids serves as a basis for several metrics used to assess a person's level of sleepiness. These criteria are:

1) The length of time that the eyelids are closed is determined by the various peaks of the Pcnt and Ncnt curves. Following a comparison with a predetermined threshold, it is employed in the calculation.

2) Constant eye blinking groups: These are also determined by the various Pcnt and Ncnt curve peaks. The presence of more than two continuous groups may indicate that the driver is fatigued.

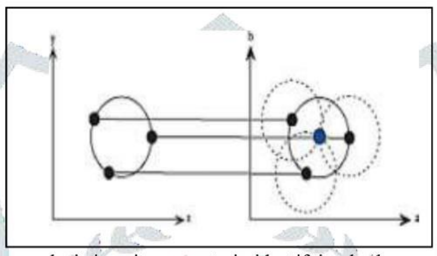
3) The frequency of eyelid closure is a crucial factor because it is one of the main ways a driver shows signs of fatigue. The motorist could try to flutter their eyes faster to awaken if they just started feeling sleepy. The driver may blink less frequently if they are exhausted for a longer amount of time.



Normal human blinking occurs every 6–10 seconds lasting 0.15–0.30 seconds, with a maximum of 2 blinks each moment. A person has a good probability of being drowsy if they disobey any of these rules.

Due to its simplicity and rapidity, this technique can be applied in real-world situations. However, due to the obvious flaws in the thesis itself, accuracy is not mentioned in the study and cannot be evaluated without thorough trials. The driver's mouth movement, which might lead to numerous irregularities, is not taken into consideration by this technique. The precision of the findings will decline with altered head positions. Additionally, any background movement of any kind may bring about erroneous positives. As this method uses a very good algorithm for eye detection but not for face or mouth detection, more research is required to improve drowsiness detection. Work needs to be done to address the aforementioned constraints in order to achieve the goal of a perfect driver sleepiness detection system.

B. Analysis of eye status using the Circular Hough Transform (CHT) The Circular Hough Transform (CHT), which can be used with any video-capturing equipment, including a driver-cam and even a webcam, was designed to do analysis of the status of eyes at various movements. This method suggests using the Circular Hough Transform to more precisely identify eyeballs [10]. In this process, the face is initially retrieved using face detection methods like the SVM algorithm [11], which Blake created and Bakir subsequently improved [12]. Second, the eyes are positioned specifically to prevent any mistake with other facial features that move. This is accomplished by using the gradient image to emphasise the edges. To detect above, lower, left, and right are now detected using horizontal and the eyes and the right edge of the face are now separated [7]. A fresh edge detection method that takes into account the shape of the eye was proposed by Alioua [7]. It recognises the eye by comparing the colour differences between the pupil, iris, and sclera. The iris is exceptionally brilliant and surrounds the extremely dark sclera. This aids in the precise identification of eyes. A threshold pixel value is chosen that is dark enough to be inside the sclera's pixel intensity boundary while yet retaining a significant contrast from the iris' pixel intensity. In order to determine whether or not the pixels to the left and right of the identified sclera pixels are iris-related, they are now tested to see if they are bright or not. The chosen pixel forms if all the requirements are satisfied with the sclera's right or left edge.



The suggested method for eye condition analysis is quite accurate in identifying both eyes and other eye structures. When used to sleepiness detection, this technique can reduce the incidence of false positives. However, because it relies on cameras, this method only works well in well-lit environments or with equipment that has a good low-light camera. Different head positions will further decrease the results' accuracy because the system is camera-based. Additionally, this method demands a lot of computer resources to operate due to its complexity. However, given sufficient lighting and processing capacity, it offers more accuracy than most traditional approaches for recognising eyes, as measured by the Kappa Statistic, Confusion Matrix, and Correct Classification Rate. According to the Confusion Matrix and Kappa statistics, respectively, and [7], the average accuracy is 99%. Although the findings show consistency in face detection, the aforementioned constraints make this algorithm ineffective for preventing accidents caused by fatigue. C. Combining yawning with eye closure In order to accurately determine the driver's level of sleepiness, two elements must be combined in this section. The accuracy in detecting the driver's condition of tiredness is lower because the aforementioned strategies only analyse the components separately. However, it is logical to infer that the driver's state is certain when several elements are combined [8]. By detecting and integrating two parameters, i.e., eye closure rate and yawning rate, in the repeated sequential frames, this technique raises the certainty that the driver is tired. This leads to a more accurate assessment of the drowsy state because yawning and increased eye closure are signs of weariness. A recognition system can use fusion in three different ways:

- Feature recognition system
- Stages 2 and 3 of the decision-making process After the decision-making process is complete, the state alert is either turned ON or off depending on the outcome of the detecting procedure [8]. Making decisions is a part of the process overall. Depending on the outcome of the detection process, the status alert is either turned ON or off after the decision-making process is complete [8]. Decision-making is a part of the entire process.

There are three detection categorization levels used by this technique;

1) Level 1 - ALERT: The eye blinks with the bare minimum frequency rate, and there is no sign of yawning. The eyes close for little more than a second, and the facial features are normal.

2) Level 2 - SEMI-DREAMY: There is some yawning and an increase in the frequency of eye blinked.

3) Level 3 - DROWSY: Closed eyes and rapid yawning are both noted. A high-alert area is this.

Utilizing a variety of parameters, such as eye closure detection and yawning, this method makes use of the subtle movements that the human body makes to detect fatigue. However, the method is difficult to implement due to its complexity, and because it is based on a camera, head rotation and poor lighting may further reduce accuracy. In addition, combining factors may result in the processing of redundant data from the combined input, resulting in even more inconsistent results. However, when compared to the other methods discussed in this paper, it may result in greater accuracy. Although M. Omidyeganeh does not specify the accuracy of the method in light of the various standards, a database containing 450 images of 27 distinct individuals reveals an accuracy of 98%. M. Omidyeganeh did not check the aforementioned limitations because the images used were of people looking straight into the camera. The highly accurate result indicates that, at least for people looking directly into the camera, this method is superior to the majority of techniques currently in use for detecting drowsiness. To have a framework with practically no vulnerability in tiredness location further work should be finished.

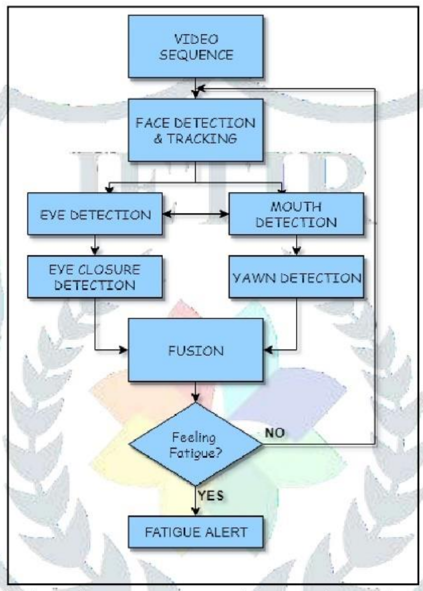


Figure 3: Driver alertness system using yawning and eye closure [8].

Because various skin tones have the same chrominance components and simply differ in brightness, they may represent varied complexity more accurately than RGB. Additionally, it is more difficult to distinguish faces using RGB since it does not support brightness to the same extent as YCbCr. Face analysis is now used to identify eyes, and Eye Map theory is used to localise the pupil centre and iris boundary. All of this is doable with a simple camera. Eye Maps of two different types—EyeMapC for chrominance and EyeMapL for luminosity—are combined to create a single eye map. EyeMapC is created using:

$$EyeMapC = \frac{1}{3}\{C_b^2 + (C)_r^2\left(\frac{C_b}{C_r}\right)\}$$

Chrominance Eye Map formula

Here, Cb and Cr represents the chrominance component of the $YCbCr$ curve. ($\tilde{C}r$) represents 255-Cr and all the different components are reduced to be in the range of [0,255]. The other Eye Map called EyeMapL is of luminance which when multiplied with EyeMapC produces the required Eye Map which is used to detect eyes. This process is not time-consuming as it requires some basic calculations [16]. After the eye region is detected Top Hat filter [17] is applied and then the intensity image is converted into a binary image using N. Otsu's method [18]. Now, the needed EyeMapL can be constructed using:

$$EyeMapL = \frac{Y(x,y) \oplus se(x,y)}{Y(x,y) \ominus se(x,y) + 1}$$

Luminance Eye Map formula

In which if the system detects the drowsiness will be sent through telegram bot so that the driver could analyse at what time he feels drowsiness while driving in the results and conclusion.for the future enhancement we could add a near by hotels beside the roads that he could find easier to get sleep if he feels the symptoms of drowsiness.

5. CONCLUSION

In this study, we investigated various methods that might be used to analyse eye-states in order to detect driver fatigue. All of the unique strategies for identifying eyeballs are briefly described in this study along with a step-by-step explanation of how they work. Based on the accuracy of the results and a real-world scenario, it also discusses the benefits and drawbacks of the various strategies. Since there is currently no dataset available for the various strategies, it is nearly hard to truly compare the outcomes of the various techniques for the real world. This paper discussed the significance of the findings reported by many writers and their relevance to the quest for the ideal sleepiness detection method.

This problem deserves a workable answer since it is a real one, not just one that is thought to exist. There are some methods available to detect driver drowsiness, as the paper has discussed, but each has its own shortcomings. Therefore, research must be conducted in order to develop a better solution for this problem that has a better implementation of a driver drowsiness detector with high accuracy.

In order to find a solution that will help to reduce and ultimately eradicate this problem, we intend to conduct more study in this area in the future. In order to improve road safety, a low-cost device that can detect sleepiness has been developed.

REFERENCES

[1] "TRAFFICACCIDENTS,"2015.[Online].Available:http://ncrb.gov.in/StatPublications/ADSI/ADSI2015/chapter1A%20traffic%20accidents.pdf

[2] M. Ruikar, "National statistics of road traffic accidents in India," J Orthop Traumatol Rehabil 2013, vol. 6, pp. 1–6, 2013. [Online].Available: http://www.jotr.in/text.asp?2013/6/1/1/118718

[3] T. A. Dingus, S. G. Klauer, V. Neale, A. Petersen, and Virginia Tech Transportation Institute, Co-Sponsors: National Highway Traffic Safety Administration and Virginia Department of Transportation, "The 100-car naturalistic driving study," 2006.

[4] Q. Ji, Z. Zhu, and P. Lan, "Real-Time Non-intrusive Monitoring and Pre- diction of Driver Fatigue," IEEE Trans. Vehicular Technology, vol. 53, no. 4, Jul.2004, pp. 1052–1068.

[5] S. Chokroverty, "Overview of sleep & sleep disorders," Indian J Med Res 131, pp. 126–140, 2010. [Online].Available: http://medind.nic.in/iby/t10/i2/ibyt10i2p126.pdf

[6] D. Liu, P. Sun, Y. Xiao, and Y. Yin, "Drowsiness Detection Based on Eyelid Movement," in 2010 Second International Workshop on Education Technology and Computer Science, Wuhan, and others, Ed., Wuhan, 2010, pp. 49–52.

[7] N. Alioua, A. Amine, M. Rziza, and D. Aboutajdine, "Eye state analysis using iris detection based on Circular Hough Transform," in 2011 International Conference on Multimedia Computing and Systems, Ouarzazate, 2011, pp. 1–5.

[8] M. Omidyeganeh, A. Javadtalab, and S. Shirmohammadi, and eye closure," in 2011 IEEE International Conference on Virtual Environments, Human-Computer Interfaces and Measurement Systems Proceedings, Ottawa, ON, 2011, pp. 1–6.

[9] D. Liu, P. Sun, and Y. Yin, "How to Speed-up an Embedded Face Detector?" 2009.

[10] R. Duda and P. E. Hart, "Use of the Hough transformation to detect lines and curves in picture," pp. 11–15, 1972.

Journal of Emerging Technologies and Innovative Research

An International Open Access Journal Peer-reviewed, Refereed Journal

www.jetir.org | editor@jetir.org An International Scholarly Indexed Journal

Certificate of Publication

The Board of

Journal of Emerging Technologies and Innovative Research (ISSN : 2349-5162)

Is hereby awarding this certificate to

Dr. Sulaiman Syed Mohammed

In recognition of the publication of the paper entitled

Driver Alertness Detection On Real Time Face And Eye Recognition

Published In JETIR (www.jetir.org) ISSN UGC Approved (Journal No: 63975) & 7.95 Impact Factor

Published in Volume 10 Issue 5 , May-2023 | Date of Publication: 2023-05-23

Parin P

EDITOR

EDITOR IN CHIEF

ISSN 2349-5162

JETIR2305947 **Research Paper Weblink http://www.jetir.org/view?paper=JETIR2305947** **Registration ID : 516647**

Journal of Emerging Technologies and Innovative Research
An International Open Access Journal Peer-reviewed, Refereed Journal
www.jetir.org | editor@jetir.org **An International Scholarly Indexed Journal**

Certificate of Publication

The Board of

Journal of Emerging Technologies and Innovative Research (ISSN : 2349-5162)

Is hereby awarding this certificate to

Dilip Kumar M

In recognition of the publication of the paper entitled

Driver Alertness Detection On Real Time Face And Eye Recognition

Published In JETIR (www.jetir.org) ISSN UGC Approved (Journal No: 63975) & 7.95 Impact Factor

Published in Volume 10 Issue 5 , May-2023 | Date of Publication: 2023-05-23

Pariu P
EDITOR

EDITOR IN CHIEF

ISSN 2349-5162

JETIR2305947 **Research Paper Weblink http://www.jetir.org/view?paper=JETIR2305947** **Registration ID : 516647**

An International Scholarly Open Access Journal, Peer-Reviewed, Refereed Journal Impact Factor Calculate by Google Scholar and Semantic Scholar | AI-Powered Research Tool, Multidisciplinary, Monthly, Multilanguage Journal Indexing in All Major Database & Metadata, Citation Generator

Journal of Emerging Technologies and Innovative Research

An International Open Access Journal Peer-reviewed, Refereed Journal

www.jetir.org | editor@jetir.org An International Scholarly Indexed Journal

Certificate of Publication

The Board of

Journal of Emerging Technologies and Innovative Research (ISSN : 2349-5162)

Is hereby awarding this certificate to

EarannaRaju C, DhurWahidQuadri

In recognition of the publication of the paper entitled

Driver Alertness Detection On Real Time Face And Eye Recognition

Published In JETIR (www.jetir.org) ISSN UGC Approved (Journal No: 63975) & 7.95 Impact Factor

Published in Volume 10 Issue 5 , May-2023 | Date of Publication: 2023-05-23

Parin P

EDITOR

EDITOR IN CHIEF

ISSN 2349-5162

JETIR2305947 Research Paper Weblink http://www.jetir.org/view?paper=JETIR2305947 Registration ID : 516647

Journal of Emerging Technologies and Innovative Research

An International Open Access Journal Peer-reviewed, Refereed Journal

www.jetir.org | editor@jetir.org An International Scholarly Indexed Journal

Certificate of Publication

The Board of

Journal of Emerging Technologies and Innovative Research (ISSN : 2349-5162)

Is hereby awarding this certificate to

Gagana G

In recognition of the publication of the paper entitled

Driver Alertness Detection On Real Time Face And Eye Recognition

Published In JETIR (www.jetir.org) ISSN UGC Approved (Journal No: 63975) & 7.95 Impact Factor

Published in Volume 10 Issue 5 , May-2023 | Date of Publication: 2023-05-23

Parin P

EDITOR

EDITOR IN CHIEF

ISSN 2349-5162 Journal of Emerging Technologies and Innovative Research

JETIR2305947 **Research Paper Weblink http://www.jetir.org/view?paper=JETIR2305947** **Registration ID : 516647**

An International Scholarly Open Access Journal, Peer-Reviewed, Refereed Journal Impact Factor Calculate by Google Scholar and Semantic Scholar | AI-Powered Research Tool, Multidisciplinary, Monthly, Multilanguage Journal Indexing in All Major Database & Metadata, Citation Generator

Journal of Emerging Technologies and Innovative Research
An International Open Access Journal Peer-reviewed, Refereed Journal
www.jetir.org | editor@jetir.org An International Scholarly Indexed Journal

Certificate of Publication

The Board of
Journal of Emerging Technologies and Innovative Research (ISSN : 2349-5162)
Is hereby awarding this certificate to

Galla Gireesh

In recognition of the publication of the paper entitled

Driver Alertness Detection On Real Time Face And Eye Recognition

Published In JETIR (www.jetir.org) ISSN UGC Approved (Journal No: 63975) & 7.95 Impact Factor

Published in Volume 10 Issue 5 , May-2023 | Date of Publication: 2023-05-23

Parin P
EDITOR

EDITOR IN CHIEF

ISSN 2349-5162

JETIR2305947 **Research Paper Weblink http://www.jetir.org/view?paper=JETIR2305947** **Registration ID : 516647**

JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue

JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)

An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Ref No : JETIR / Vol 10 / Issue 5 / 947

Confirmation Letter

To,
Dr. Sulaiman Syed Mohammed
Published in : Volume 10 | Issue 5 | 2023-05-23

Subject: Publication of paper at International Journal of Emerging Technologies and Innovative Research .

Dear Author,

With Greetings we are informing you that your paper has been successfully published in the International Journal of Emerging Technologies and Innovative Research (ISSN: 2349-5162). Following are the details regarding the published paper.

About JETIR	: An International Scholarly Open Access Journal, Peer-Reviewed, Refereed Journal Impact Factor Calculate by Google Scholar and Semantic Scholar | AI-Powered Research Tool, Multidisciplinary, Monthly, Multilanguage Journal Indexing in All Major Database & Metadata, Citation Generator, Impact Factor: 7.95, ISSN: 2349-5162
UGC Approval	: UGC and ISSN Approved - UGC Approved Journal No: 63975 | Link: https://www.ugc.ac.in/journallist/subjectwisejurnallist.aspx?tid=MjM0OTUxNjI=&&did=U2VhcmNoIGJ5IElTU04=
Registration ID	: JETIR 516647
Paper ID	: JETIR2305947
Title of Paper	: Driver Alertness Detection On Real Time Face And Eye Recognition
Impact Factor	: 7.95 (Calculate by Google Scholar)
DOI	:
Published in	: Volume 10 | Issue 5 | 2023-05-23
Publication Date	: 2023-05-23
Page No	: j6-j14
Published URL	: http://www.jetir.org/view?paper=JETIR2305947
Authors	: Dr. Sulaiman Syed Mohammed, Dilip Kumar M, Gagana G, Galla Gireesh, EarannaRaju C, DhurWahidQuadri

Thank you very much for publishing your article in JETIR. We would appreciate if you continue your support and keep sharing your knowledge by writing for our journal JETIR.

Indexing

Editor In Chief
International Journal of Emerging Technologies and Innovative Research
(ISSN: 2349-5162)

www.jetir.org | editor@jetir.org | Impact Factor: 7.95 (Calculate by Google Scholar)

No Dues Certificate

....Gagana G................(Student Name) bearing the Roll No. 20191CSE0153... has no dues with placement department and be considered for the evaluation and subsequent processing / announcement of results.

Signature: ....

Name of the Placement Officer / Manager: Pradeep. B.S (Admin Executive)

Date: 25/5/23...

No Dues Certificate

Dhur Wahid Quadri (Student Name) bearing the Roll No. 20191CSE013 has no dues with placement department and be considered for the evaluation and subsequent processing / announcement of results.

Signature: Pradeep

Name of the Placement Officer / Manager: Pradeep. B.S (Admin Executive)

Date: 25/5/23