Drowsiness Detection System

Capstone Project Report

END SEMESTER EVALUATION

Submitted by:

101903774 Sarthak

101903758 Arashpardeep Singh

101953001 Preeti Rani

102083018 Taniya

BE Fourth Year, COE

CPG No: 236

Under the Mentorship of

Dr. Nitigya Sambyal

Assistant Professor

THAPAR INSTITUTE OF ENGINEERING & TECHNOLOGY (Deemed to be University)

Computer Science and Engineering Department
Thapar Institute of Engineering and Technology, Patiala

DEC 2022

ABSTRACT

The majority of car accidents that occur worldwide involve drowsy driving. Driver drowsiness detection is the most crucial strategy for reducing these instances, enabling us to avoid many road accidents. Our project aims to develop a Drowsiness Detection System that sounds an alarm when a driver's eyes are closed for a predetermined amount of time. Deep learning is employed in this study to propose a new frame that categorizes the driver's eye status, i.e., open or closed. The suggested device emits a beep when a driver is recognized as drowsy after the sleepiness measure reaches a certain saturation point. The proposed system works well irrespective of the driver wearing spectacles and under bad lighting conditions.

DECLARATION

We hereby declare that the design principles and working prototype model of the project entitled Drowsiness Detection System is an authentic record of our own work carried out in the Computer Science and Engineering Department, TIET, Patiala, under the guidance of Dr. Nitigya Sambyal and during 6th & 7th semester (2022).

Date: 19 December, 2022

Roll No.	Name	Signature
101903774	Sarthak	Sarthak
101903758	Arashpardeep Singh	Arashpardeep
101953001	Preeti Rani	Preeti Rani
102083018	Taniya	Taniya

Counter Signed By:
Faculty Mentor: Nitigya Sambyal.

Dr. NITIGYA SAMBYAL

Designation

CSED,

TIET, Patiala

ACKNOWLEDGEMENT

We would like to express our thanks to our mentor Dr. Nitigya Sambyal. She has been of great help in our venture, and an indispensable resource of technical knowledge. She is truly a fantastic mentor to have.

We are also thankful to Dr. Shalini Batra, Head, Computer Science and Engineering Department, entire faculty and staff of Computer Science and Engineering Department, and also our friends who devoted their valuable time and helped us in all possible ways towards successful completion of this project. We thank all those who have contributed either directly or indirectly towards this project.

Lastly, we would also like to thank our families for their unyielding love and encouragement. They always wanted the best for us and we admire their determination and sacrifice.

Date: 19 December, 2022

Roll No.	Name	Signature
101903774	Sarthak	Sarthak
101903758	Arashpardeep Singh	Arashpardeep
101953001	Preeti Rani	Preeti Rani
102083018	Taniya	Taniya

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CNN	Convolutional Neural Network
ML-NLP	Machine learning - natural language processing
NHTSA	National Highway Traffic Safety Administration
D2CNN-FLD	Drowsiness Detection based on Convolutional Neural Network and Facial Landmark Detection
SS-CNN	Specific Designed Shallow Convolutional Neural Network
OBDII	On-Board Diagnostic II
HTML	Hypertext Mark-up Language
CSS	Cascading Style Sheets
YOLO	You Only Look Once
PERCLCOS	Percentage of Eye Closure
LSTM	Long short-term memory
GPU	Graphics processing unit
UI	User Interface
SVM	Support Vector Machine
IBM	International Business Machines

INTRODUCTION

1.1 Project Overview

The proposed project detects the drowsiness level of the driver while driving the vehicle and raises an alarm to alerts him in case the system finds him drowsy. Such an automated system will help in reducing vehicle accidents. The proposed drowsiness detection system is based on convolution neural networks which will analyse the facial features of the driver to determine his level of drowsiness.

The key issues are real-time processing capacity and the algorithm's robustness to variations in the human face. Using traditional image processing and custom computer vision techniques, we can effectively handle the variances in the face. However, the usual technique also fell short in addressing various differences, such as position variation, lighting condition, intra-class variance, and face expression. Deep Learning is an alternative to traditional approaches that performs better by automatically extracting and learning the key attributes. In order to manage the drowsiness detection system for drivers in real time, we are employing several convolution neural networks. The method for doing this is based on facial expression indicators, which are extensively used to define drowsiness.

In this project, we will train various CNN over a given data set. The model is trained using a data set that has many images of people with various features such as closing and opening eyes, with or without glasses, males or females, in the dark or in the light, and many more factors so that when any person is using this system in any condition, if he/she feels drowsy, the model determines how sleepy the driver is and warns them by blaring an alarm. Thereafter performance comparison will be done to choose the model that performs the best. The suggested project will then make use of the chosen model.

Thereafter, the model will use a web camera to capture the driver's facial features. We also give time parameters as for how much time the eyes get closed, for example, if a person just blinks their eyes, then also eyes get closed for a while, and if in that case also the alarm beeps, then it would be irritating for a driver. So, we also have taken into account that for how many seconds the eyes are closed, if it does not open after a certain short period then the alarm beeps and alerts the driver.

1.2 Need Analysis

The most significant causes of car accidents are drowsiness and fatigue. These road accidents due to the drowsiness of the driver are increasing tremendously every year. To avoid such accidents, an intelligent system is required that alerts the driver by raising an alarm whenever he is sleepy while driving.

Vehicle accidents occur mostly due to a single moment of negligence, thus a system that monitors the drowsiness of the driver continuously in real time is necessary to minimize these accidents.

Autonomous driving vehicles are becoming more and more prominent, but they still have many efficiency and safety challenges to solve. Hybrid automobiles are currently in vogue, and fast travel and safety are entirely dependent on human abilities. So, anybody may experience the ramifications of driving when fatigued. Short night-time sleeps or an unaltered physical state are typically to blame for this.

According to NHTSA records, 846 fatalities involving sleepy drivers occurred in 2014. Over the previous 10 years, these deaths have basically stayed the same. Between 2005 and 2009, there were around 83,000 accidents annually that were thought to be caused by fatigued driving. For these reasons, danger alert systems can rouse sleepy drivers and help prevent mishaps. Therefore, there is a critical need for drowsiness detection systems in automobiles to prevent deaths that result from driver drowsiness.

An automated system based on continuous monitoring of the facial features of the driver will help in detecting drowsiness and hence will help in reducing vehicle accidents.

1.3 Research Gap

1. The models proposed in existing literature do not perform well if the driver is wearing spectacles. This is mainly due to obstruction in viewing facial features accurately.
2. The existing models fail to detect drowsiness in poor lighting conditions.
3. The methods used for drowsiness detection have low accuracy.

1.4 Problem Definition and Scope

There are many serious car accidents in India. A mechanism is required to address the problem of a traffic collision. Our project attempts to create software that uses a user's face and eyes to gauge their level of sleepiness.

1.5 Assumptions and Constraints

The following tables present the assumptions and constraints that are imposed upon the "Drowsiness Detection System":

TABLE 1.1: Assumptions

S. No.	Assumptions
1.	The drivers must see the main information and understand the primary functionality first. An example is adjusting the on/off control apart from any other groupings.
2.	The interface should include built-in cues that describe what it is and how it operates.
3.	There are consequences for every action a user makes with the product. These must be communicated to users through useful feedback. Such as, switching the system on requires feedback to let the driver know that the monitor is working. An alert sound when the driver closes their eyes beyond the allowed threshold.
4.	To enable quick and simple recovery, a reset button for software failure must be built into the system architecture.
5.	If a user forgets his password, he should be able to change it.

TABLE 1.2: Constraints

S. No.	Constraints
1.	The users access the online drowsiness detection system from any device with internet browsing capabilities and an Internet connection.
2.	The user's email address must be unique to register. The same email address cannot be used more than once during registration since this information is required to be unique.

1.6 Approved Objectives

The project's primary objectives are:

1. To study and explore the existing techniques for Drowsiness Detection.
2. To design and implement a deep learning system for detecting drowsiness.
3. To evaluate the performance of the suggested model and compare it with the approaches already in use.
4. To develop a web-based application using the proposed model.

1.7 Methodology

1. *Data Collection:* We have selected a data set that has many images that consist of features like eye status (open or closed), with or without glasses, bright or dim light, etc.

2. *Pre-processing of data:* The data will be cleaned and various pre-processing techniques will be applied to make it suitable for machine training.

3. *Model Training:* A Deep Learning based model will be proposed for drowsiness detection. The model will be analysed on various performance parameters.

4. *Model Testing:* Testing will be done in parallel to have a minimum chance of error at every step of implementation. To check the accuracy of the trained model, various error techniques like error using root mean square or mean square, etc. will be used.

5. *Comparative Analysis:* We compare our model in different scenarios, such as taking different batch-size, learning rates, etc. each time, and using various pre-trained models. The model which gives us maximum accuracy with the appropriate conditions would be chosen as our final model.

6. *Web Application:* A web application with an easy-to-use UI can be handled both verbally and by directly interacting with the device screen. The app runs completely offline on the mobile device and runs the ML-NLP models on the back -end.

7. *Web Testing:* Testing will be done in parallel to have a minimum chance of error at every step of implementation. Both the Black Box and White Box Testing techniques will be used to have maximum accuracy.

1.8 Project Outcomes and Deliverables

This capstone project will result in a Drowsiness Detection System made using Deep Learning and finally deployed in a web App. This system will help to reduce road accidents caused due to drowsiness of the drivers as our app will alarm the driver when it detects the driver is sleepy beyond a safe limit and thus assisting in lowering the risk of accidents caused by drowsy drivers.

• Key features of the system would be:

1. It will continuously record the movements and detect the drowsiness of the driver.
2. Drowsiness measurement parameter would be a drowsiness score which will
3. If the drowsiness score is above the par limit, it will warn the driver by beeping the alarm.
4. The developed system will even perform better when in the dark/dim light or the driver wearing spectacles or lenses.
5. The model would provide great help in preventing road accidents caused due to drowsiness of the drivers.

1.9 Novelty of Work

The novelty of the project is that this also works well on people with glasses and also in the lights. In the normal drowsiness detection system, first of all, they are mostly made using normal machine learning but we are using deep learning as this project is done in deep learning in a very less number. Now in normal machine learning-based detection systems, they do not work properly when the glasses are on the eyes as sometimes, they recognized the reflections on the eyes as the eyes its themselves his is probably not the case with our system. It manages to differentiate between the eyes and the reflections on the glasses. At the same time, our project also works well when the lights of the room are not sufficiently bright. In that, case also it detects the eyes properly. The major impact of this is that if any person installs our system in their cars, even if they drive in the nights when the lighting is not sufficient, even at that time if the driver feels drowsy, the system will detect it efficiently and save a person from the major accidents, which is generally not the case with normal machine learning drowsiness detection system.

2.1 Literature Survey

2.1.1 Theory Associated with Problem Area

Self-driving is a major concern all over the world. India is a big country, and the population of India is very large than other countries. In India, many people die and get injured due to road accidents caused due to drowsiness of the driver, which can be a severe issue. Recent studies say that 40% of people in India are killed during road accidents. Many theories and technologies are being developed, and some are currently being used. This will save people during road accidents, and some of them are helpful, which will observe the driver's actions while driving the car; if they seem fatigued, it will alert the driver before any mishap happen and save people's lives.

2.1.2 Existing Systems and Solutions

Extensive work has been done and many methods, machine learning, and deep learning algorithms are applied to determine whether the driver is sleepy or not. The biggest challenge is to save the driver's life during a road accident. There are many methods applied like respiration, heart rate, pulse rate, etc. But these devices were uncomfortable for the drivers. These methods give less accuracy. The present system struggles to operate in low light, and its accuracy suffers as a result of its inability to recognize the driver's face and eyes in conditions of darkness or low light. So, the blinking eyes and closing eyes method is a promising method of monitoring alertness. which will detect the driver's eye during driving the car or any vehicle and the eye-opening and blink rate give a more accurate result than the other methods

2.1.3 Research Findings for Existing Literature

TABLE 2.1: Literature Survey

S. No.	Roll Number	Name	Citation	Tools/ Technology	Findings
1.	101953001	Preeti Rani	R. Jabbar et al. [1]	D2CNN-FLD	Developed D2CNN-FLD model, which has a size of 75 KB and can be efficiently blended into the instrumental panels for the next origination of automobiles. Even under poor lighting situations, there is still possibility for performance enhancement and better face element identification.
2.			Y. Suresh et al. [3]	CNN, Haar cascades algorithm	A stacked deep CNN was developed and used in the learning phase to extract features. A CNN classifier's SoftMax layer was used to categorize drowsiness. For improved performance and accuracy, genetic algorithms and transfer learning can be used.
3.			S. Dalal et al. [6]	CNN, Haar cascades algorithm	The framework imitated the driver's eye bands and makes a low recurring sound when the car driver begins to nod off. The details of the driver and their history of sleepiness were stored in an IBM db2 information base. Constraints

					were face can't be recognized in low lightcircumstances and isn't valuable when driver is wearing shades.
4.			J. Flores-Monroy et al. [10]	Viola & Jones face detector, SS-CNN	SS-CNN was used to process the facial area that had been extracted using the viola Jones technique. Information whose eyes were closed or opened was given by SS-CNN. The alarm was set off when "eye closed" appeared for more than four consecutive frames.
5.	101903774	Sarthak	A. Manikandan and M. Sujith [4]	MobileNetV2, Raspberry Pi 4B, Pi Camera and Buzzer	A multi-featured system strengthening vehicle security and safety was put into place. Drowsiness detection and permitted engine activation were two of the system's key functions.
6.			Mohammad Shahverdy et al. [9]	CNN, OBDII adapter, throttle position sensor	Proposed method analysed driver behaviour based on vehicle signals during driving. Driving signals were converted into images using recurrence plot and then CNN was used to categorise the photos into five categories of driving behaviours, including normal, aggressive, distracted, sleepy, and alcoholic driving.
7.	102083018	Taniya	K. B. R. Teja and T. K. Kumar [2]	Viola Jones, YOLO	The system worked well under different lighting conditions.

				algorithms, PERCLOS	However, system neglected to detect the drowsiness when head was turned downwards to right or left.
8.			S. Sathasivam et al. [7]	Eye Aspect Ratio (EAR) technique, Pi camera, Raspberry Pi 4 and GPS module	The framework distinguished eyes area from pictures and computed the worth of Eye Aspect Ratio (EAR). Drivers who wear spectacles might make the framework neglect to perceive their eyes accurately and have blunders in recognizing their eyes.
9.	101903758	Arashpardeep	Faraji, F. et al. [5]	YOLOv3 CNN, LSTM	Proposed a technique based on a combination of LSTM and CNN system for accurate drowsiness detection. A comparison between two cases, a hybrid system of CNN and LSTM and using a single CNN with two strategies was made. Experiment showed the robust performance of the hybrid system.
10.			A. -U. -I. Rafid et al. [8]	CNN, CenterFace algorithm, Haar-Classifier	Both users who use glasses and those without are better served by this strategy. In less than two seconds in real time, the suggested model can identify sleepiness. Performance can be improved even when there is low light and the complexity of

					computation can be decreased in future.
11.			M. Ramzan et al. [3]	PERCLOS, SVM, Haar Cascade Classifier	SVM is the most commonly used classification which gives better accuracy in various situations but is not suitable for large datasets.

2.1.4 Problem Identified

When the distance between the webcam and the face is too great, sleepiness detection technology has difficulty. The face won't be correctly identified from the image by the system. As a result, it fails to recognize the eyes and provides through. Our system generates problems if the webcam detects more than one face.

2.1.5 Survey of Tools and Technologies Used

The majority of studies have employed a deep-learning strategy to identify drowsy drivers. Faces in the frames captured from the live video are detected using the Haar cascade approach which is an object detection algorithm. The technique uses Viola and Jones's edge or line detecting features proposed in the research paper "Rapid Object Detection using a Boosted Cascade of Simple Features". Several deep learning CNN models like D2CNN-FLD, SS CNN, MobileNetv2, and YOLOv3 CNN, recurrent neural networks (RNN) like LTSM, and machine learning (ML) approach SVM and other models like PERCLOS, eye aspected ratio (EAR) are used to classify the identified faces in the frames as drowsy or non-drowsy.

The application is accomplished using the Raspberry Pi 3 module, Raspberry Pi camera, Buzzer, throttle position sensor, OBDII adapter, and GPS module.

2.2 Software Requirement Specification

2.2.1 Introduction

2.2.1.1 Purpose

The main goal of this document is to give a thorough explanation of our software on the Drowsiness Detection System & its parameters. It will demonstrate the function, features, and terminals of the software and its other functioning. It also defines how our users view the final product.

2.2.1.2 Intended Audience and Reading Suggestions

The target audience for this publication is the individuals involved in developing and evaluating Drowsiness Detection System, including the development team, mentor, and project evaluation panel. SRS is divided into four sections:

- Introduction, which includes the generic data about the project like purpose and project scope.
- Overall description that focuses on the product perspective and product features.
- External Interface Requirements address software, hardware, and user interfaces.
- Other Non-functional Requirements address things performance, safety, and security requirements.

2.2.1.3 Project Scope

The goal is to design a drowsiness detection system for drivers. In this system, when the driver starts monitoring, the system begins recording the driver's behavior. The system employs a CNN model to determine a person's level of fatigue depending on whether or not their eyes are open or closed. OpenCV is employed for capturing the pictures from the web camera and giving those pictures to the CNN model, that will further categorize them as 'Open' or 'Closed. For closed eyes, a predetermined score is used. If the rate of closed eyelids surpasses the saturation point, the application recognizes the sleepiness of the driver and beeps an alarm to alert him before anything dangerous happens.

The software must be able to perform the following operations:

1. *Take images as input from a camera*: It must be able to capture images of the facial characteristics of the driver.

2. *Detect eyes in the image and classify them as open or close*: It must be able to detect eyes in the image captured, and on every captured image, a classifier should be applied at the back-end that will classify whether the eyes of the driver were open or close.
3. *Calculate score to check whether the driver is drowsy*: The score is mainly a number that will be used to calculate how long the person has kept his eyes closed. Therefore, if both eyes are closed, the score must continue to increase, and if the eyes are open, the score must be dropped.
4. *Beep alarm*: If the system finds the driver drowsy, it must be able to beep an alarm until the score decreases.

2.2.2 Overall Description

2.2.2.1 Product Perspective

The project is based on a system that detects the drowsiness of a driver and alerts him by beeping an alarm. The most significant causes of car accidents are drowsiness and fatigue. These road accidents due to the drowsiness of the driver are increasing tremendously every year. According to NHTSA records, 846 fatalities involving sleepy drivers occurred in 2014. Over the previous 10 years, these deaths have basically stayed the same. Between 2005 and 2009, there were around 83,000 accidents annually that were thought to be caused by fatigued driving. For these reasons, risk alert systems can awaken the drowsy driver and can able to avoid accidents. Thus, there is a vast need for drowsiness detection systems in vehicles to stop the fatalities that take place due to the drowsiness of the driver.

To avoid such accidents, an intelligent system is required to alert the driver by raising an alarm whenever he is sleepy. So, our team has designed a system that can save people from these accidents that happen due to drowsiness. The proposed drowsiness detection system is based on CNN which will analyze the facial features of the driver to determine his level of drowsiness.

The major issues are the hardiness of the mechanism towards the diversity of the face of a human and processing capability in real-time. We can handle facial variations with conventional methods of image processing and hard-coded machine-learning algorithms. But some variations like face expression, light conditions, variations within the class, and pose variation are some of the extra challenges the conventional method failed to solve. A substitute solution to conventional methods is Deep Learning which gives better results by automatically extracting and learning the significant features. Thus, we are using different CNN for handling the drowsiness detection system for drivers in real-time. The strategy for that to be used is based on the facial behavior measures which are completely taken advantage of to illustrate drowsiness.

In this project, we will train various convolution neural networks over a given data-set. The model is trained using a data set that has many images of people with various features such as closing and opening eyes, with or without glasses, males or females, in the dark or in the light, and many more factors so that when any person is using this system in any condition, if he/she feels drowsy, the model detects the degree of drowsiness and alerts the driver by beeping an alarm.

Thereafter, a performance comparison will be done to find the model with the best results. The selected model will then be used for the proposed project. Thereafter, the model will use a web camera to capture the driver's facial features. We also give time parameters as for how much time the eyes get closed, as if for example, if a person just blinks their eyes, then also eyes get closed for a while, if in that case also the alarm beeps, then it would be irritating for a driver. So, we also have taken into account that for how many seconds the eyes are closed, if it does not open after a certain short period then the alarm beeps and alerts the driver.

2.2.2.2 Product Features

1. It will continuously record the movements and detect the drowsiness of the driver.
2. Drowsiness measurement parameter would be a drowsiness score which will depend on the movement and closeness of the eye in a certain time interval.
3. If the drowsiness score is above the par limit, it will warn the driver by beeping the alarm.
4. The proposed system will even perform better when in the dark/dim light or the driver wearing spectacles or lenses.
5. The proposed system would provide great help in preventing road accidents caused due to drowsiness of the drivers.

2.2.3 External Interface Requirements

2.2.3.1 User Interfaces

TABLE 2.2: Function 1

Function 1	**Login/Signup**
Input	Name, email address, and password of the user
Processing	Verify the provided information, then enter the data in the database.
Output	Redirect to user's home page

TABLE 2.3: Function 2

Function 2	**Continuous image capturing from the video frame**
Input	It takes input by capturing continuous images from the video frame using a webcam.
Processing	Images are captured continuously from the video frame using a webcam and then processed.
Output	The continuously captured images that are processed, pass to the model.

TABLE 2.4: Function 3

Function 3	**Detection of eye motions and calculating a score**
Input	Continuously captured images of the driver and the data-set that includes images of closed and open eyes so to train the model.
Processing	The processing is done on the eyes of the driver and a score is calculated on the basis of the closeness of the eyes. If the eyes are closed, the score increases continuously and vice-versa.
Output	A score is displayed on the screen all the time that corresponds to how long and how many eyes are closed.

TABLE 2.5: Function 4

Function 4	**Beeping an alarm**
Input	The captured images and the score that is calculated above.
Processing	The score is monitored continuously. When the score increases from 30 due to closing of eyes for specified time, it initiates an alarm.
Output	An alarm beeps to alert the driver.

2.2.3.2 Hardware Interfaces

- Laptop (with web camera)
- Cloud GPU

2.2.3.3 Software Interfaces

- TensorFlow
- Open CV
- Kaggle
- Real-time Database
- Django
- Google charts

2.2.4 Other Non-functional Requirements

2.2.4.1 Performance Requirements

The allowed response times for system functioning are defined by performance criteria. The accomplishment of the application will be greatly influenced by the hardware and software components of the computer, even if the website is designed to need the least amount of system resources. The load time for user interface displays must be under a few seconds when considering the temporal relationships of the system. Within a short amount of time, the login details will be validated. The search function becomes more precise when query results are returned in a short amount of time.

2.2.4.2 Safety Requirements

The Drowsiness Detection System has different user levels. To access the various subsystems, a user log-in page will be used, which asks for a username, email address, and password. This provides several views and user-level functions that are accessible through the system. The security of the system's database is ensured by maintaining backups. In the event of an emergency, the system can be restored.

2.2.4.3 Security Requirements

- A user will register/log in with a valid email id only. Otherwise, the system gives an error.
- To store the user's data, we used the built-in database of MongoDB.

2.3 Risk Analysis

1. Application shuts down abruptly.
2. Camera switches off.

3.1 Investigative Techniques

1. **Data Exploration:** The most important thing needed for our project is a dataset. So, finding the right dataset that may fit our work is essential. So, in the initial stages of preparing our project, we are dedicated mostly to finding the right dataset that would go with our ideas for the project. For the same, we have searched various sites like Kaggle, YouTube, or some other google sites for some consecutive days. There are many datasets we find on the internet regarding drowsiness detection but the one we have chosen is the best suitable for our entire idea for the project. The data set also consists of images of eyes with glasses means we can also train the proposed system on the eyes with glasses, thus giving better results even in the case of glasses and also working fine in dim lighting conditions.

2. **Survey on Literature:** After the data set has been chosen, we have to study a lot for our project as we do not have much technical knowledge about how the project works, what are the different technologies we can use for it, and what are the existing methods. So, for this, we go through many of the research papers, analyze them all properly, and find a way to move on further to the next step to create our project.

3. **Comparative Analysis:** There are many technologies that exist today with which the drowsiness of a person can be detected but all methods do not pose the same accuracy. For better results, we need more accuracy, and to find out which method provides more accurate results, we must have to do a comparative analysis of the existing methods. Firstly, we do comparative analysis for some machine learning technologies, but they won't show many accurate results, so we shift towards deep learning. Now, in deep learning also, there are many Transfer Learning models, so we have searched for the best transfer learning models and found the best three out of all, and then we run all of them on the proposed system and check the accuracies in all of them. The one with the maximum accuracy, that is, InceptionV3 is chosen for the detection of drowsiness.

3.2 Proposed Solution

In this system user will log in/register then it opens the camera After the web camera is started it captures the image, and the classifier is loaded by the model that we have trained using neural networks for this project, which reads the video stream of the driver using the web camera. And then check the facial characteristics when the facial characteristics are detected then the eye detection is started it checks the eye open and close .and then detects drowsiness if the driver is drowsy it will give the alert alarm and alert the driver.

These are the following stages we used:

1. Login/Register
2. Camera
3. Face detection
4. Eye detection
5. Drowsiness detection
6. Alert system

1. *Login/register* – First of all users can register on the site and if he/she will be already registered then they will log in directly and start the application.
2. *Camera* – When the user starts the application. he/she will open the camera first and the camera capture the image and continuously captures the image.
3. *Face detection*-in the captured image the neural networks detect the facial characteristics of the driver from the video stream.
4. *Eye detection* – On every captured image, a classifier is applied at the back-end that will classify whether the eyes of the driver were open or closed. As the closing of eyes also happens during blinking of eyes, and during sneezing, so to clarify if the person is actually drowsy or fatigued, we kept a score in the system. As the system finds the eyes of the person closed, the score starts increasing.
5. *Drowsiness detection* – In this it will check the drowsiness of the driver if the score increases by a certain threshold it will show that the driver is in a drowsy state and if the score is not increased by a certain condition it will tell that driver is not drowsy.
6. *Alert alarm* – If the driver is drowsy it will the alert message or audio message to the driver to be awake if not then the sound not start.

3.3 Work Breakdown Structure

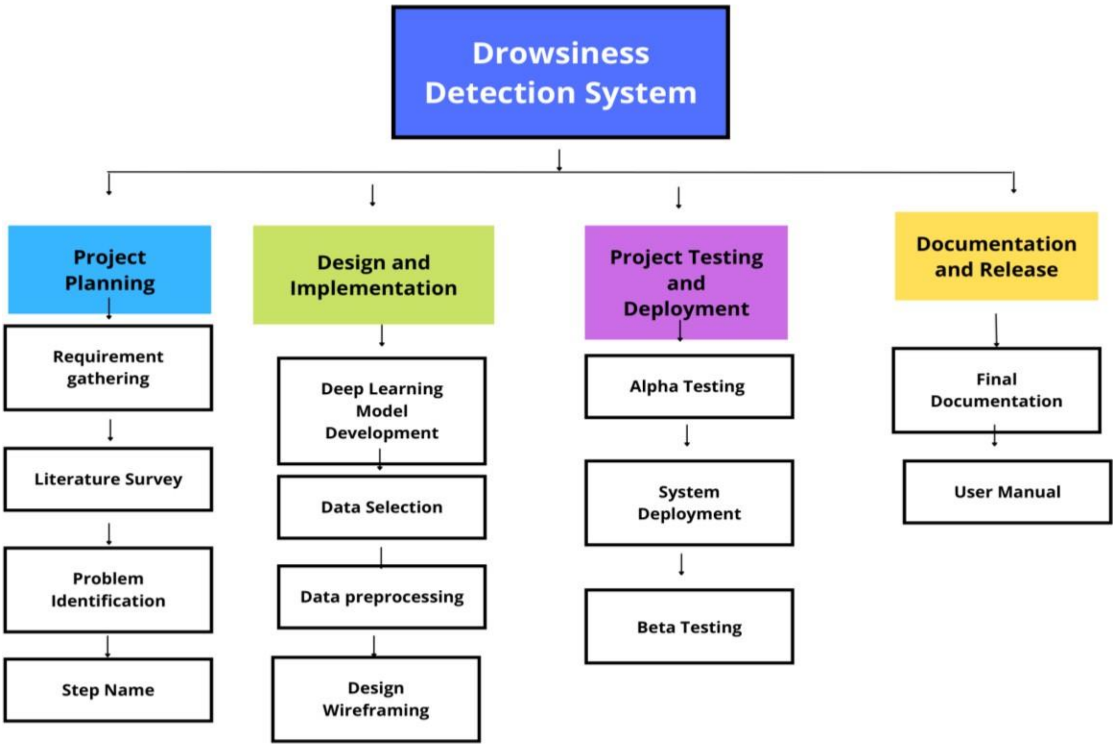


FIGURE 3.1: Work Breakdown Structure

3.4 Tools and Technology

1. Webcam
2. Python
3. HTML/Java-script/CSS
4. Django
5. Deep Learning

4.1 System Architecture

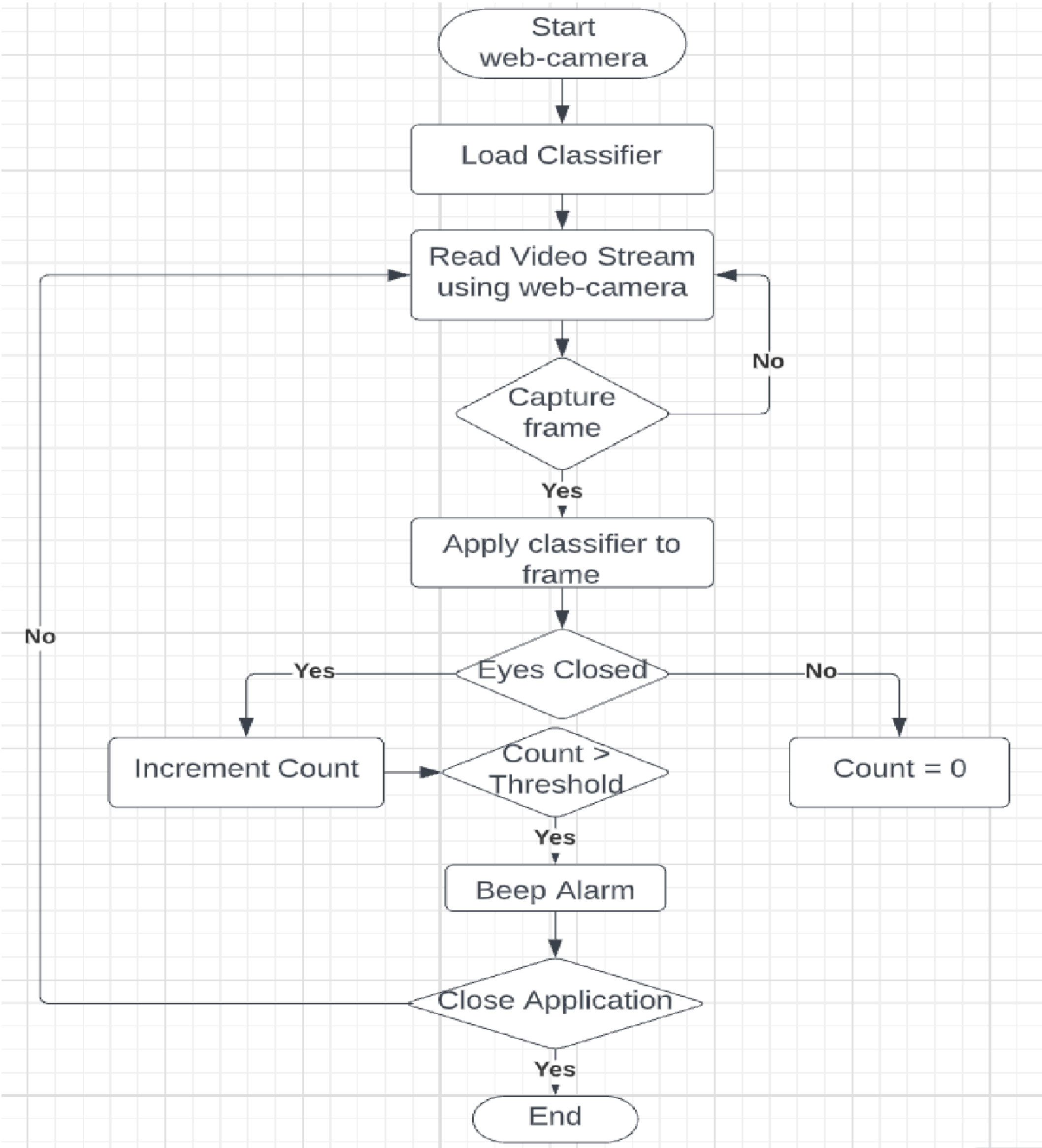


FIGURE 4.1: System Architecture

The architecture for detecting the drowsiness of the driver is shown in the above figure. When the web camera is activated, the model that we trained using neural networks for this project loads the classifier, which reads the video stream of the driver using the web camera. This will continuously capture images of the facial characteristics of the driver from the video stream.

Every image that is taken has a classifier applied at the back end that determines whether the driver's eyes were open or closed. The score starts rising as soon as the system recognize that the user has closed eyes. If the score rises above a predetermined threshold, it indicates that the person is sleepy since their eyes are closed, and the system sounds an alert to wake them up. This alarm continuously rings until the driver does not wake up and the system recognizes the eyes of the driver to be open. When the system captures the open eyes, the score decreases and alarm stops ringing.

4.2 Design Level Diagrams

Component diagram

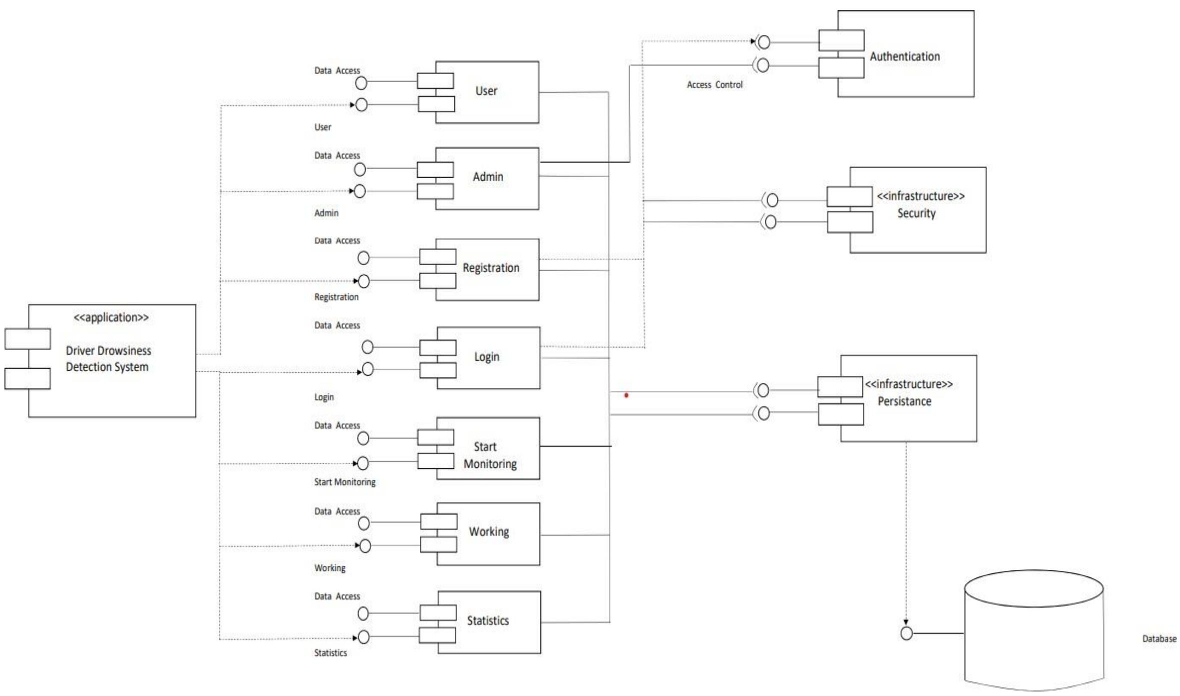


FIGURE 4.2: Component Diagram

The component diagram of the drowsiness detection system has seven components: admin, user, registration, login, start monitoring, working, and statistics. Login and registration components are required interfaces for authentication and security. Admin, user, registration, login, start monitoring, working, and statistics are provided interfaces for the driver drowsiness detection system and are the required interface for persistence. The persistence component relies on the database component.

Class Diagram

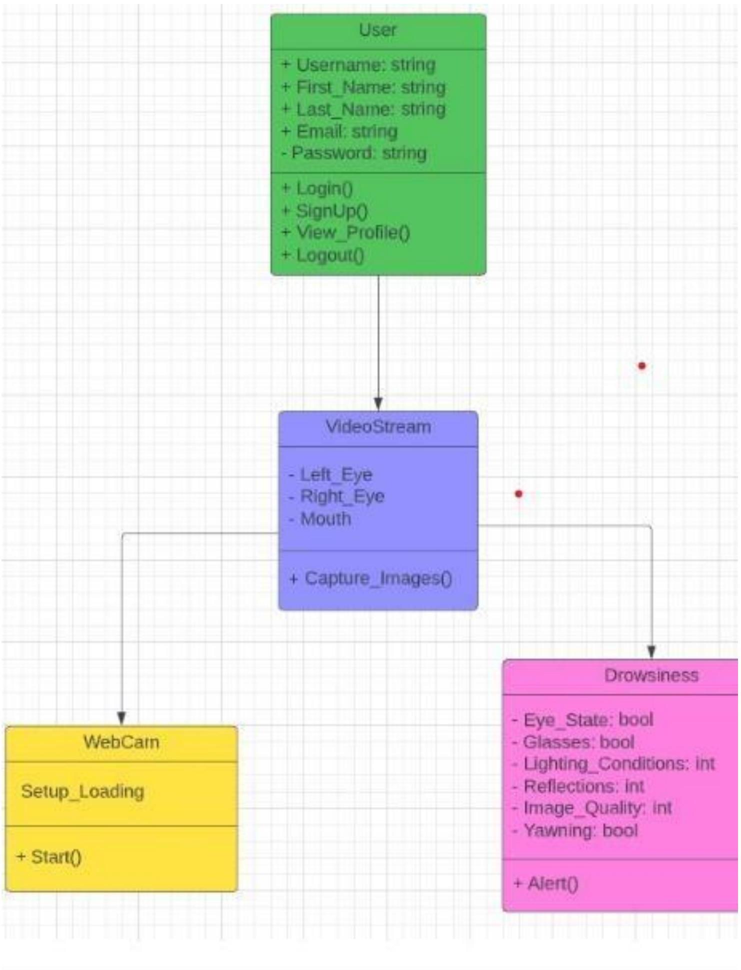


FIGURE 4.3: Class Diagram

A user class has an attribute of username, first name, last name, email, and password. A user can sign up with a new account, log in or log out to his account, or even view his or her profile.

Now the user can start video streaming. Video streaming consists of the attributes of the left and right eyes including the face and it can capture the images of the user in a real-time. Video streaming is including the web camera and drowsiness class. Webcam has an attribute of setup loading and it can start the streaming, and in the drowsiness class, the attributes included are eye status, glasses, lighting conditions, reflections, etc. Eye status checks if the eyes are closed or open, similarly glasses check if a person wears glasses or not, and so on. When a user feels drowsy, an alert signal is given by ringing an alarm so that the person wakes up.

4.3 User Interface Diagrams

Sequence Diagram

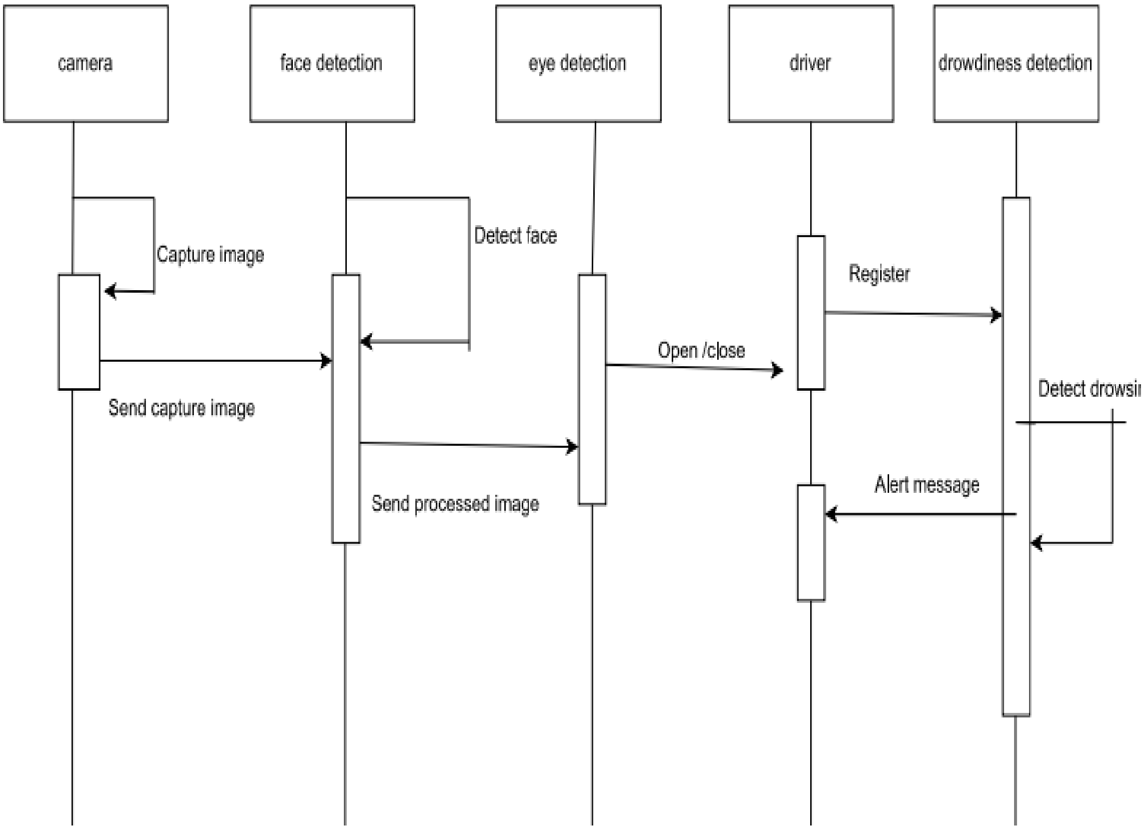


FIGURE 4.4: Sequence Diagram

1. Camera: While the vehicle is moving, it continuously captures the image of the driver and sends it to a storage device.
2. Facial Landmark Detector: This feature monitors the driver's facial changes and reports them.
3. Eye detection: It determines whether the eyes of the driver are open or closed.
4. Driver: The person who is driving the vehicle.
5. The Drowsiness Detection System: It is an image-capture system that analyses a series of photos using different Algorithms to state if the driver is sleepy or not.

IMPLEMENTATION AND EXPERIMENTAL RESULTS

5.1 Experimental Setup

We design a website the application necessarily runs on a mobile (e.g., an Android-based smartphone) mounted on the instrumental panel of the vehicle. It ideally needs to be easily carriage able to other mobiles of comparable size and computational capabilities (e.g., iOS-based).

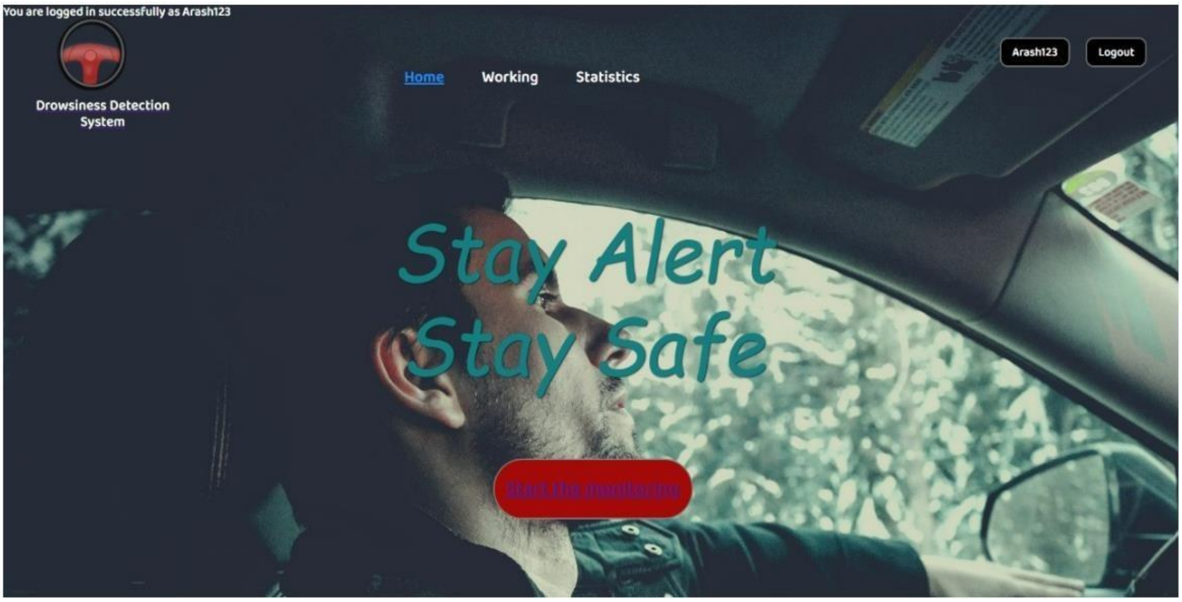


FIGURE 5.1: Home Page

The proposed system will begin by sequentially capturing video frames. For each frame, the system will recognize the face in the frame image. Once the face of the driver has been identified by the face detection function, the eyes detection function attempts to detect the eye of the driver. The system processes the input image stream in real-time to calculate the driver's degree of fatigue. The analysis is done by calculating a number of data stream frames with the driver's eyes closed. The result of the processing is sent to the alarm, which activates an alarmsignal when the drowsiness index exceeds a pre-specified parameter.

5.2 Experimental Analysis

5.2.1 Data

The dataset we have used is mrlEyes_2018_01 which consists of images of the eyes of 38 people. There are approximately 15000 images of the eye pupils of 38 people. The naming of every image of the data is based on several parameters like gender, eye state (open or closed), wearing glasses or not, light conditions, reflections, etc. Each image is named as subjectID_imageNo_gender_glasses_eyeState_reflections_lightingConditions_sensorType. In gender, if the person is male, the value is 0, and if the person is female, the value is 1. In glasses, if a person wears the glasses the value is 1 and it is 0 otherwise. In the eye state, the closed eyes gives the value 0 and open eyes gives the value 1. In reflections, if reflections are none, then the value is 0, if they are low, then the value is 1, and if they are high, the value is 2. Then training and test datasets are created by dividing the original dataset with both datasets having a folder of closed eyes and open eyes. There are a total of 80858 images in the training dataset of which 64687 are used for training and 16171 are used for validation. The test data consists of 4040 images in total.

5.2.2 Performance Parameters

In two steps, we improved the performance. To improve performance, we tweaked the code and switched it to GPU-based processing rather than CPU-based processing. Later, we increased the FPS for the Haar Cascade Algorithm in order to capture real-time video more smoothly and therefore improve its performance. To achieve more accuracy, we use a transfer learning model (Inception V3) which loads the network with a pre-trained version trained on innumerable images from the database called ImageNet. To hyper-tune the parameters, we use the batch size of 8, and no. of epochs equal to 16. We also add early stopping with the patience of 7 to stop the epochs early if the consecutive epochs do not contribute to increasing the accuracy. The system was then put to the test by live webcam video streaming by all four teammates. Then we put it to the test with a corner test case i.e., With Spectacles, testing in dim light, sneezing, eye blinking, etc.

5.3 Working of the project

5.3.1 Procedural Workflow

The web camera is primarily utilized to capture a static image or a video stream. The system receives the input in the form of this caught image or video stream. This module contains the following items: Image manipulation, Face recognition, and eye region recognition Determine the closeness of the eye. The resulted image or input video put into the application will be processed by the image processing module. It takes in data & transforms it into frames for processing it further. The technique of face detection is used to find the face in an image. The input video stream is transformed into frames, and those frames are used to detect the face. The Haar classifier is used in the Viola-Jones algorithm to determine the face in a provided image. The position of the eye of the driver is determined by applying the appropriate threshold. The edge detection of the eye's region is considered in this work. After finding the region of eye in the frame, the application determines if the eye is closed or it is open. If the driver's eye remains open, the system determines that he or she is not sleepy. If the eyes of the driver remain closed, the system detects that he or she is drowsy. The application constantly detects whether the driver is drowsy or not. If the driver feels drowsy, an alert signal is generated in the form of an alarm sound or a beep sound.

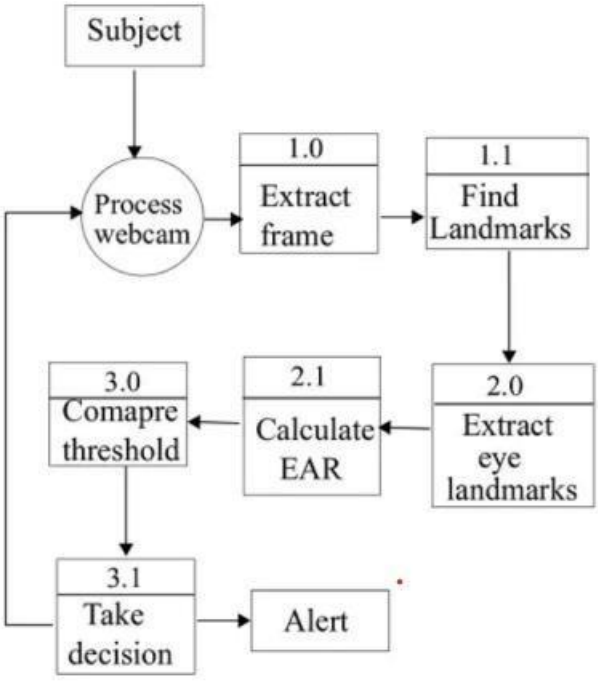


FIGURE 5.2: Workflow

5.3.2 Algorithmic Approaches Used

For drowsiness detection, firstly we take the dataset of many images of open and closed eyes. We divided the dataset into train and test datasets. Both of these datasets have folders of open and closed eyes. We take these images as input and reshape them using ImageDataGenerator by rescaling them by 255, setting the rotation range, shear range, zoom range, and similar other parameters to 0.2. The training data is divided into train and validation using validation_split which is set to 0.2. For test data, we put the target size to (80, 80), batch size to 8, and the class mode used is categorical. Then to train the model, we make a base model by using a transfer learning model, InceptionV3, and then flatten the base model, and add some more layers of relu and softmax activation function and a dropout layer of 0.5 to attain our final model. Then we train this final model. Then we add checkpoints by monitoring the value loss. The fine-tuning is done by using the parameters like learning rate (setting its patience to 3 by monitoring the value loss by using Reduce LR On Plateau), and an early stop with the patience of 7. Here patience means the model will stop early without completing all the epochs if the accuracy is not increasing or the validation loss is not decreasing. The model is then compiled using the Adam optimizer, loss as categorical cross-entropy, and accuracy used for the metrics. Then the model is fit into the fit generator with all the hyper-tuning parameters like the batch size as 8, early stopping, learning rate, setting epochs to 16, etc., and then the accuracy and loss of training, validation, and testing are calculated. Now, this model is used for our final output. We use Haar Cascade to detect the face and eyes, by converting the images to grayscale. And then if the eyes are closed the system alerts the user with an alarm while live streaming using the webcam.

5.3.3 Project Deployment

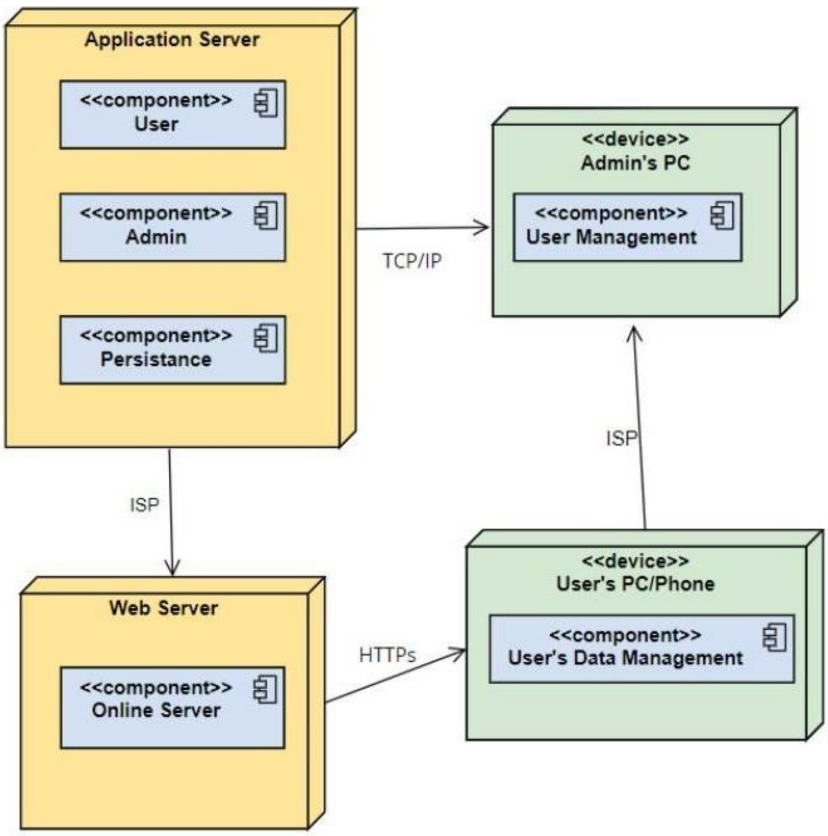


FIGURE 5.3: Deployment Diagram

5.3.4 System Screenshots

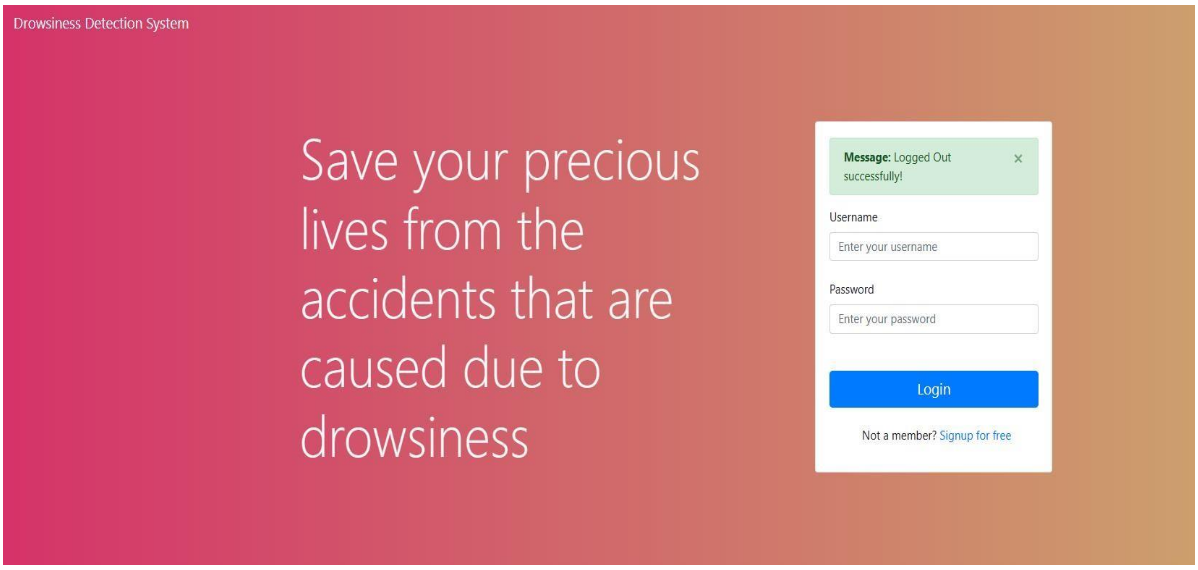


FIGURE 5.4: Login Page

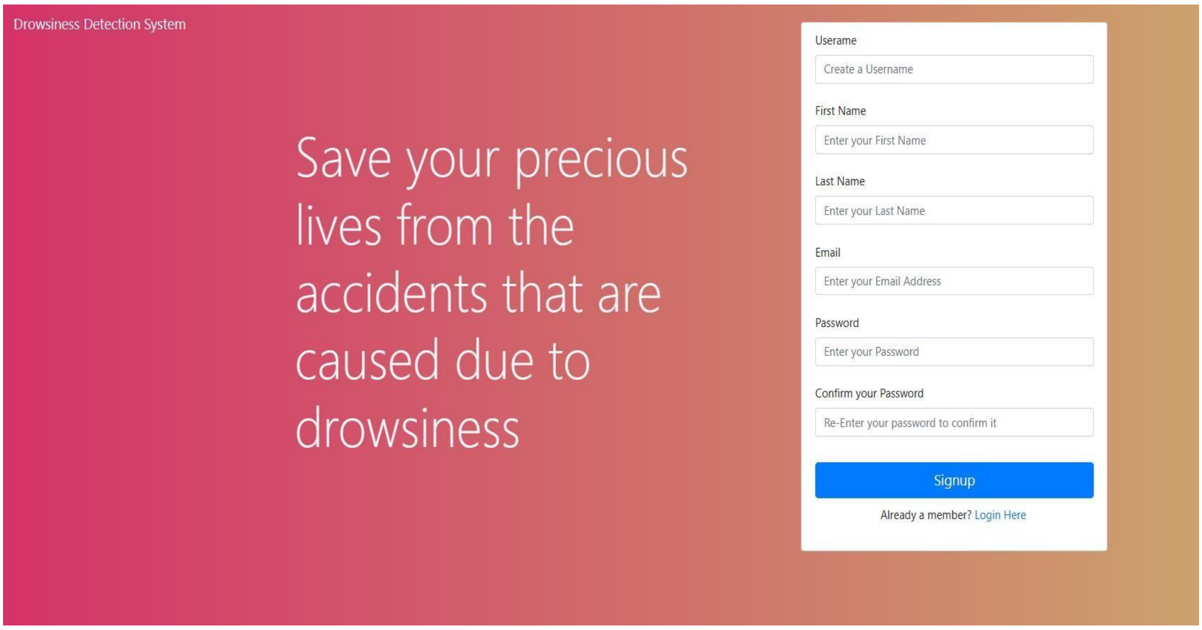


FIGURE 5.5: Signup Page

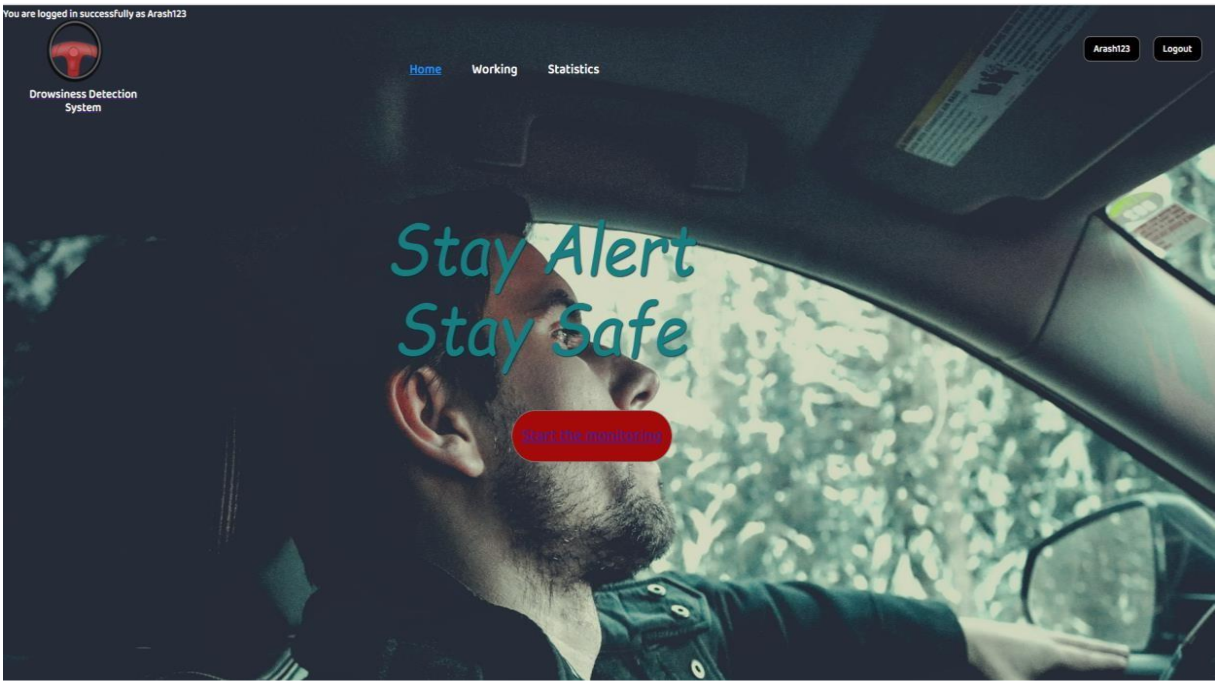


FIGURE 5.6: Home Page

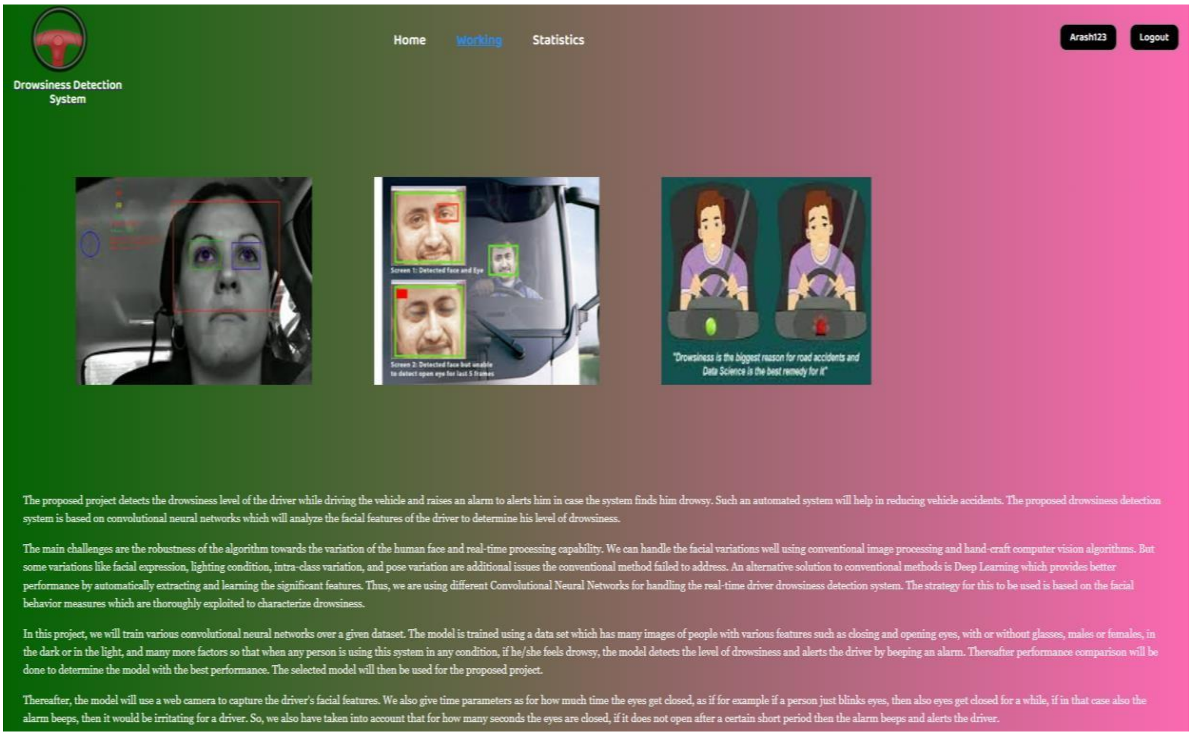


FIGURE 5.7: User Manual Page

FIGURE 5.8: User Manual Page

FIGURE 5.9: User Profile Page

5.4 Testing Process

5.4.1 Test Plan

The core of this project comprises three modules, face detection, eye detection, and detection of drowsiness using a webcam. The testing plan for the project was catered to ensure that all these modules were tested individually followed by integrated testing of all these modules to ensure that the final product works effectively and seamlessly. While the purpose of individual testing was to find errors and problems in each component and to correct them, integration testing is used to check how well these components perform when they are correlated with each other and perform all the corrections to ensure the smooth performance of the components in the final product.

5.4.2 Features to be tested

On-going with the workflow of the project there are 3 main features that have to be tested, Firstly, the face detection feature, which includes the algorithm that is responsible for detecting the face from the video streaming using a web camera. The second Feature to be tested is the Eye Detection Algorithm, which is all about detecting the eyes of the person in the video streaming using Haar Cascade. Finally, the last feature to be tested is the drowsiness detection algorithm which should raise the alarm if the person is founded to be drowsy.

5.4.3 Test Strategy

White Box Testing:

The list of modules is as follows –

Module 1: Face Detection: Using Haar Cascade.

Input: Input video feed from the webcam and the dataset.

Output: Output Video with the detected face of the person.

Module 2: Eye Detection: Detection of the eyes of a person using Haar Cascade.

Input: Input Video feed from the webcam and the dataset.

Output: Output Video which detects the eyes of the person.

Model 3: Drowsiness Detection

Input: Input Video streaming using a web camera detecting both face and eyes and the dataset.

Output: An alarm sound if a person is founded to be drowsy.

In Black Box Testing, we passed the model with the eyes of different people with various parameters to check if they are detected or not. The parameters used are eyes with spectacles or without them, in dim light or bright light, open or closed eyes, etc. The system has to detect whether the eyes are closed or open to find the drowsiness.

5.4.4 Test Techniques

5.4.4.1 Unit Testing

In this period of testing the emphasis was fundamentally on testing individual parts of the task like face discovery part, eyes discovery, and drowsiness detection part. These parts were tried independently utilizing limit case testing and so on.

5.4.4.2 Integration testing

The objective of this testing is to take the unit parts together in a program design and afterward attempt to see whether the parts are cooperating and delivering the ideal outcomes or not. It is parametric testing to check in the event that the parts have been incorporated appropriately and whether or not the shared association between the items is set up. To check whether the parts make obstruction every other working.

5.4.4.3 Regression testing

This kind of testing is capable to guarantee that the item or program proceeds true to form particularly when another module is added to the current task. In our task, there were a few changes that we needed to so relapse testing was an extremely fundamental part.

5.4.5 Test Cases

Case 1: if the algorithm works well with open eyes.

Case 2: if the algorithm works well with closed eyes.

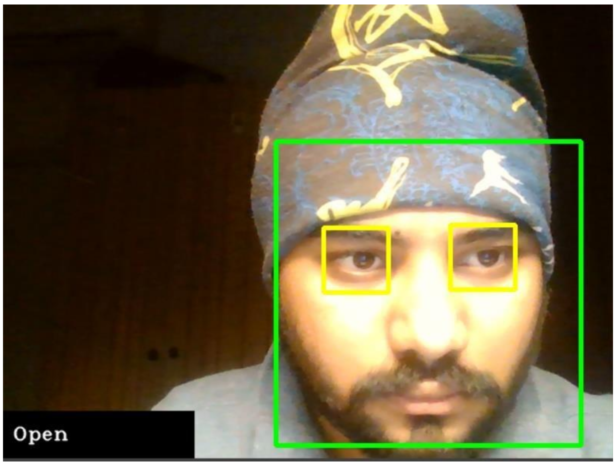
Case 3: if the algorithm works well with eyes with spectacles.

Case 4: if the algorithm works well with eyes without spectacles.

Case 5: if the algorithm works well with eyes in dim lights or dark.

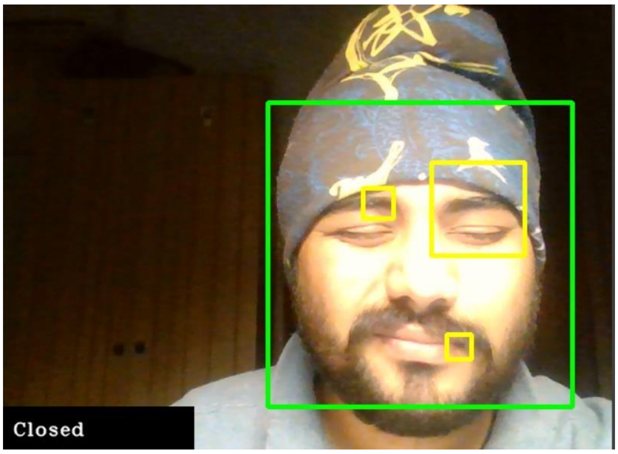
Case 6: if the algorithm works well with eyes in bright lights.

5.4.6 Test Results



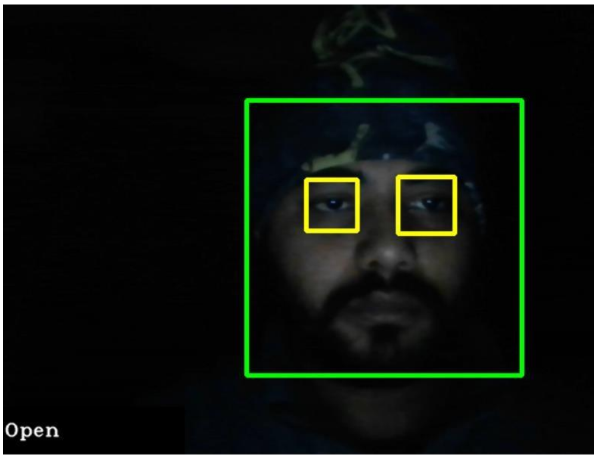
Test Case: Open Eyes in bright light

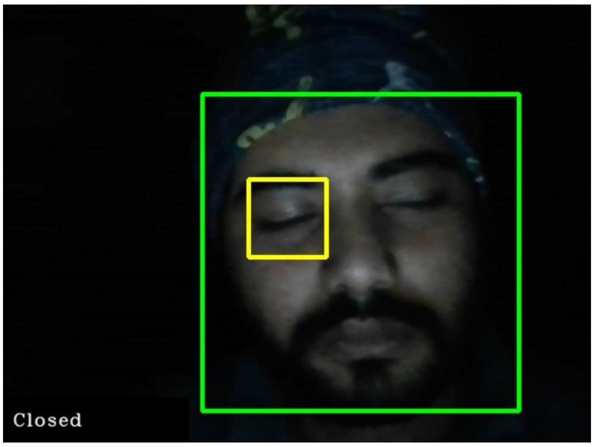
FIGURE 5.10: Open Eyes In bright light



Test Case: Closed Eyes in bright light

FIGURE 5.11: Closed Eyes In bright light





Test Case: Open Eyes in dim light

Test Case: Closed Eyes in dim light

FIGURE 5.12: Open Eyes In dim light

FIGURE 5.13: Closed Eyes In dim light

5.5 Results and Discussions

With Face and Eye Detection and drowsiness detection algorithm of Haar Cascade, we have been effectively ready to identify whether the person is drowsy or not. We have likewise been fruitful in expanding the quantity of edges of our video utilizing improvement methods given by the modules in the OpenCV library. Through the use of the Haar Cascade and Inception V3, we had the option to give the adequate result for the drowsiness of the person in the edge cases also like the person with spectacles or the eyes in dim lighting, etc. Drowsiness

Detection is carried out in utilizing the pre-prepared item acknowledgment models given by the OpenCV library.

Table 5.1: Comparison Summary

	InceptionV3	VGG16	ResNet50
Epochs	16	16	16
Batch Size	8	8	8
Training Accuracy	94.66%	90.03%	50.62%
Training Loss	13.68%	23.01%	69.31%
Validation Accuracy	93.15%	79.99%	50.62%
Validation Loss	17.46%	41.61%	69.31%
Test Accuracy	82.42%	72.18%	50.09%
Test Loss	55.82%	50.38%	69.32%

5.6 Inferences Drawn

Drowsiness Detection is a quickly developing field of work and a ton of examination is going on in this field. These innovations have far and wide applications in different regions. With upgrades in the exactness of the outcomes of these advances, the extent of these advances will additionally grow. By including innovation, we can all together have an effect on expenses and time used by the item. At first, the advancement of the item might take more expenses and different speculations yet it is especially productive and practical in the long haul. This is really very beneficial if used in vehicles to avoid accidents due to the drowsiness of the driver.

5.7 Validation of Objectives

Checking where the final product stands in accordance with the project objectives:

Table 5.2: Objective Table (To check if we were able to achieve them)

Objective	Successful/Unsuccessful
Face Detection	Successful
Eye Detection	Successful
Drowsiness Detection	Successful
GUI Application	Successful
Hardware Implementation	Future Scope

CONCLUSION AND FUTURE SCOPE

6.1 Conclusions

A novel method for detecting driver drowsiness based on the state of the eyes is proposed. This evaluates if the driver is drowsy or not and, if it is, notifies with an alarm. To find the face and eye region, the Viola-Jones detection method is applied. In the learning phase, a stacked deep convolutional neural network is built and used to extract features. CNN classifier is used to determine if the driver is drowsy or not. A buzzer sound has been added in, and the driver will be warned if drowsiness is detected. The suggested system accurately identifies the state of sleepy output when the model predicts it on a regular basis.

6.2 Environmental Benefits

Road accidents cause a significant number of deaths, and most of these accidents are the result of driver inattention and tiredness. Therefore, having a low-cost but effective driving assistance warning system may be a massive advantage in preventing the loss of life and property. Drivers of medium and heavy commercial vehicles frequently spend more than 12 hours straight behind the wheel, which increases their risk of mishap. The enhanced safety requirements would benefit vulnerable road users (pedestrians, cyclists, and motorcyclists) and non-vulnerable road users and vehicles. The proposed system will reduce the road accidents and bring down the loss of life and property.

6.3 Future Work plan

In the future, we are trying to improve the accuracy of the model using more improvised techniques. Also, we tried to do hardware implementation of our drowsiness detection system using Raspberry Pi and Zigbee and other IoT devices.

PROBLEM METRICS

7.1 Challenges Faced

a) Very slow training of the models due to large dataset and unavailability of the GPU.
b) Multiple detections of the eyes due to less accurate models.
c) Dealing with the backend of the server.
d) Crashing of the web camera or slow pop-up of the camera due to heavy streaming.
e) Programming of the task required troubleshooting and settling mistakes. This was overwhelmed by utilizing on the web gatherings where experienced engineers give significant direction.

7.2 Relevant Subjects

TABLE 7.1: Relevant Subjects

Subject Code	Subject Name	Description
UCS538	Data Science	This subject helps us in understanding the datasets in a better way and applying various operations to the datasets.
UML501	Machine Learning	This subject helps us in training and testing our model and using it is used in detecting the drowsiness of a person.
UCS503	Software Engineering	This subject helped in every stage of the project by providing a proper method to successfully and efficiently complete the project
UCS664	Natural Language Processing	This subject helped in understanding Neural Networks and how they can be used in drowsiness detection.

7.3 Interdisciplinary Knowledge Sharing

In this venture the information on the accompanying disciplines was used Image Processing, Machine Learning and Deep Learning. Colleagues zeroed in on propelling information specifically trains and further teamed up for consummation of the undertaking. Normal gatherings helped put out shared objectives, resolve difficulties and agree on basic choices for the undertaking (very much like in spry model for programming advancement).

7.4 Peer Assessment Matrix

TABLE 7.2: Peer Assessment Matrix

		Evaluation For			
		Sarthak	Arashpardeep Singh	Preeti Rani	Taniya
Evaluation By	Sarthak	5	4	4	4
	Arashpardeep Singh	4	5	4	4
	Preeti Rani	5	5	5	5
	Taniya	4	4	4	5

7.5 Role Playing and Work Schedule

The roles of individual team members are as follows:

1. Sarthak [Team Leader]:
   - Various Data-set of Drowsiness Detection System
   - Data pre-processing
   - System Architecture
   - Web deployment for the model
2. Arashpardeep:
   - Compare and Analyse the Drowsiness detection system.
   - Various pre-defined Transfer Learning models.
   - Class Diagram, E-R Diagram
   - Web app deployment for the model
3. Preeti Rani:
   - Literature Survey

- Research Work on various research papers.
- Component Diagram, State chart, Activity Diagram
- Front-end deployment

4. Taniya:

- Work Breakdown Structure
- Research Work on various research papers.
- Sequence Diagram, Use case diagram, Dfd diagram
- Front-end deployment

Work Schedule

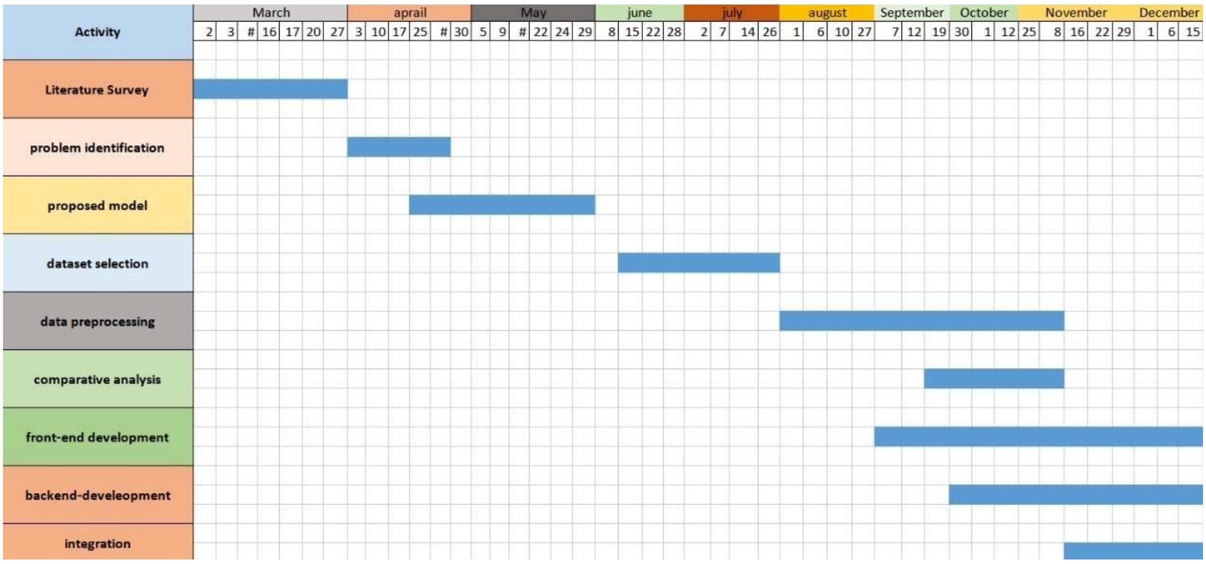


FIGURE 7.1: Work Schedule

7.6 Student Outcomes Description and Performance Indicators (A-K Mapping)

TABLE 7.3: Student Outcomes Description and Performance Indicators

S. No.	Description	Outcome
A1	Applying basic principles of science toward solving engineering problems.	Applying basic principles of science toward solving engineering problems.
A2	Applying engineering techniques for solving computing problems.	Used the concept of Neural networks for object detection.

B1	Identify the constraints, assumptions, and models for the problems.	Stated the constraints, assumptions, and models.
B2	Use the proper tools, methods, and approaches for gathering data.	Collected data using appropriate techniques.
B3	Analyze and interpret results with respect to assumptions, constraints, and theory.	Analyzed and interpreted the results with respect to assumptions, Constraints, and theory.
C1	Can understand scope and constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability.	Designed the system such that all the constraints were taken care of.
C2	Fulfil assigned responsibility in multidisciplinary teams.	Built the desired modules in separate teams.
D1	Can play different roles as a team player.	Played different roles as a team player.
E1	Identify engineering problems.	Clearly identified the engineering problems such as fog removal and object detection.
E2	Develop appropriate models to formulate solutions.	Formulated practical solutions for the desired problems
E3	Use analytical and computational methods to obtain solutions	Yes, we were able to obtain a solution by using experimental investigative techniques and analysis.
F1	Interact with peers and professional communities while demonstrating professionalism.	Showed professional responsibility while interacting with others.
F2	Able to assess a problem's ethical implications.	Yes, our problem identified was ethical as it helped those who need our utmost care and support.
G1	Produce different documents, such as lab or project reports, using the proper formats.	Produced the Required Documentation.
G2	Deliver well-organized and effective presentations.	Delivered effective oral presentations
H1	Aware of the environmental and societal impact of engineering solutions.	Yes, our project was developed by considering both aspects.

I1	Able to explore and utilize resources to enhance self-learning.	Yes, we were able to achieve it through team coordination and Interdisciplinary knowledge gathering.
I2	Recognize the importance of life-long learning.	Yes, it is our major learning from the capstone project.
J1	Comprehend the importance of contemporary issues.	Yes, we reviewed a variety of issues for our project development and it made us more aware of such issues.
K1	Use several programming languages to write code.	Wrote codes in different languages and platforms
K2	Use various data structures and algorithmic methods.	Applied different techniques studied in our course.
K3	Use software tools necessary for computer engineering domain	Applied and used the different necessary software tools

7.7 Brief Analytical Assessment

Q1. What sources of information did your group explore to arrive at the list of possible project problems?

Ans. From the start of the undertaking, we have the fundamental information on deep learning that were fundamental for this venture, which were Deep Learning and CNN. Thus, we began with investigating these subjects exhaustively to get a thought and outline of potential groups and difficulties we may confront by fostering this task. Furthermore, we likewise read writing review of undertakings that were utilizing devices and methods that were like our own venture.

Q2. what analytical, computational and/or experimental methods did your project use to obtain solution to the problems in the project?

Ans. The fundamental reason for our task is to detect drowsiness and alarm the driver, so there were a few phases in the undertaking where we needed to utilize different devices to acquire arrangements and push ahead. One of them was eye detection and level of drowsiness in them, and deploy the model to support and work on the effectiveness of the result. Another exploratory advance that we look by fostering this venture was to utilize a lot of continuous article identification and other calculations, like hyper tuning to see which one worked the best for us.

Q3. Did the project demand demonstration of knowledge of fundamentals, scientific and/or engineering principles? If yes, how did you apply answer?

Ans. In this capstone project, we have utilized standards of Deep Learning, CNN And Machine Learning. Aside from them, a lot of software engineering information, was likewise needed for the plan, design and documentation part of the venture. Basics of deep learning were utilized for picture handling and afterward, AI calculations were utilized to measure drowsiness in the eyes of the driver.

Q4. Does the project make you appreciate the need to solve problems in real life using engineering? And could the project development make you proficient with software development tools and environment?

Ans. Our venture tends to have a genuine issue utilizing design. Chipping away at this task, has caused us to see the value in the need to take care of true issues and has propelled us to take up new issues in different fields. We were acquainted with different new advances throughout this task, like Deep Learning, CNN, Neural Networks, Machine Learning and different Python libraries. Chipping away at these advances has exceptionally worked on our proficiencies with these innovations.

APPENDIX A: References

[1] R. Jabbar, M. Shenoy, M. Kharbeche, K. Al-Khalifa, M. Krichen, and K. Barkaoui, 2020, "Driver Drowsiness Detection Model Using Convolutional Neural Networks Techniques for Android Application," IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT), pp. 237-242.
https://ieeexplore.ieee.org/document/9089484

[2] K. B. R. Teja and T. K. Kumar,2021, "Real-Time Smart Drivers Drowsiness Detection Using DNN," 5th International Conference on Trends in Electronics and Informatics (ICOEI), 2021, pp. 1026-1030.
https://ieeexplore.ieee.org/document/9452938

[3] M. Ramzan, H. U. Khan, S. M. Awan, A. Ismail, M. Ilyas, and A. Mahmood, "A Survey on State-of-the-Art Drowsiness Detection Techniques," in IEEE Access, vol. 7, pp. 61904- 61919, 2019, DOI: 10.1109/ACCESS.2019.2914373.
https://ieeexplore.ieee.org/document/8704263

[4] Y. Suresh, R. Khandelwal, M. Nikitha, M. Fayaz, and V. Soudhri, 2021, "Driver Drowsiness Detection using Deep Learning," 2021 2nd International Conference on Smart Electronics and Communication (ICOSEC), pp. 1526-1531.
https://ieeexplore.ieee.org/abstract/document/9591957

[5] A.Manikandan and M. Sujith,2021, "A Novel System for Real-Time Drowsiness Warning and Engine Ignition Authorization using Face Recognition," 2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC), pp. 1658- 1663.
https://ieeexplore.ieee.org/document/9532599

[6] Faraji, F., Lotfi, F., Khorramdel, J., Najafi, A. and Ghaffari, A., 2022. Drowsiness Detection Based On Driver Temporal Behavior Using a New Developed Dataset.
https://arxiv.org/abs/2104.00125

[7] S. Dalal, S. Patidar, and U. Baburam, 2021, "Drowsiness Detection From Real-Time Video Capturing Using Convolution Neural Network and Low-Frequency Sound," 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT), pp. 1-5
https://ieeexplore.ieee.org/document/9579754

[8] S. Sathasivam, A. K. Mahamad, S. Saon, A. Sidek, M. M. Som and H. A. Ameen, 2020, "Drowsiness Detection System using Eye Aspect Ratio Technique," 2020 IEEE Student Conference on Research and Development (SCOReD), pp. 448-452.
https://ieeexplore.ieee.org/document/9251035

[9] A.-U. -I. Rapid, A. I. Chowdhury, A. R. Niloy and N. Sharmin,2021, "A Deep Learning Based Approach for Real-time Driver Drowsiness Detection," 2021 5th International Conference on Electrical Engineering and Information Communication Technology Page 16 of 16 (ICEEICT), pp. 1-5.
https://ieeexplore.ieee.org/abstract/document/9667944

[10] Mohammad Shahverdy, Mahmood Fathy, Reza Berangi, Mohammad Sabokrou, 2020, Driver behavior detection and classification using deep convolutional neural networks, Expert Systems with Applications, Volume 149,2020,113240, ISSN 0957-4174.
https://www.sciencedirect.com/science/article/pii/S095741742030066X

[11] J. Flores-Monroy, M. Nakano-Miyatake, G. Sanchez-Perez, and H. Perez-Meana, 2021, "Visual-based Real-Time Driver Drowsiness Detection System Using CNN," 2021 18th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE), pp. 1-5.
https://ieeexplore.ieee.org/abstract/document/9633082

APPENDIX B: Plagiarism Report

drowsiness detection system

ORIGINALITY REPORT

7%	3%	1%	6%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Thapar University, Patiala Student Paper	3%
2	ri.cmu.edu Internet Source	1%
3	www.ijmetmr.com Internet Source	<1%
4	Submitted to University of North Texas Student Paper	<1%
5	deepai.org Internet Source	<1%
6	Submitted to University of Greenwich Student Paper	<1%
7	Submitted to Fort Valley State Univeristy Student Paper	<1%
8	Submitted to International School of Monaco Student Paper	<1%
9	Submitted to Koc University Student Paper	<1%

10	Submitted to Middle East College of Information Technology Student Paper	<1 %
11	Submitted to Study Group Australia Student Paper	<1 %
12	bscw.fi-xifi.eu Internet Source	<1 %
13	Submitted to Universiti Malaysia Pahang Student Paper	<1 %
14	opensource.cit-ec.de Internet Source	<1 %
15	www.deltaelectronics.com.au Internet Source	<1 %
16	core.ac.uk Internet Source	<1 %
17	Submitted to Aligarh Muslim University, Aligarh Student Paper	<1 %
18	www.coursehero.com Internet Source	<1 %
19	Saravanaraj Sathasivam, Abd Kadir Mahamad, Sharifah Saon, Azmi Sidek, Mohamad Md Som, Hussein Ali Ameen. "Drowsiness Detection System using Eye Aspect Ratio	<1 %

Technique", 2020 IEEE Student Conference on Research and Development (SCOReD), 2020
Publication

20	Mohammad Shahverdy, Mahmood Fathy, Reza Berangi, Mohammad Sabokrou. "Driver behavior detection and classification using deep convolutional neural networks", Expert Systems with Applications, 2020 Publication	<1 %

Exclude quotes	Off	Exclude matches	< 4 words
Exclude bibliography	On		

Counter Signed By:

Faculty Mentor: Nitigya Sambyal.

Dr. NITIGYA SAMBYAL

Designation

CSED,

TIET, Patiala