**KWAME NKRUMAH UNIVERSITY OF SCIENCE AND TECHNOLOGY**

DEPARTMENT OF COMPUTER SCIENCE

## DRIVER DROWSINESS DETECTION SYSTEM USING COMPUTER VISION



BY

**GODMAN OWUSU OBENG-DENTEH** (8560621)

**ALBERT ARYEH** (8536821)

A Final Year Project Work Submitted to the Department of Computer Science,

Kwame Nkrumah University of Science and Technology,

In Partial Fulfilment of the Requirements for the Award of

The Degree of **Bachelor of Science in Computer Science**

*SUPERVISOR*: PROF. JAMES BEN HAYFRON-ACQUAH

# Dedication

We dedicate this work to our family and loved ones, whose unwavering support, encouragement, and prayers have been our source of strength throughout this journey.

# Acknowledgement

We wish to express our deepest gratitude to our supervisor, **Prof. James Ben Hayfron-Acquah**, for his invaluable guidance, constructive feedback, and encouragement during the course of this research.

We are also grateful to our lecturers, colleagues, and friends who provided insight and assistance throughout the project. Finally, we thank our family for their patience, love, and support which enabled us to complete this thesis

successfully.

# Abstract

Driver drowsiness is a leading cause of road accidents worldwide, contributing to thousands of fatalities annually. This study presents a real-time drowsiness detection system using computer vision and machine learning. The system employs facial landmark detection, Eye Aspect Ratio (EAR), and Mouth Aspect Ratio (MAR) to monitor eye closures and yawning. A continual learning approach using Scikit-learn’s SGDClassifier adapts to user-specific behavior, with a calibration phase ensuring personalized detection without storing facial data. Experiments showed reliable detection of drowsiness and inattentiveness, though challenges remain with sunglasses, extreme head poses, and lighting variations. The system is effective for real-time monitoring, with potential to enhance road safety.

# List of Figures

3.1 Data Flow Diagram of the Driver Drowsiness Detection System 24

5.1 User Looking Directly into Frame (Stable Condition) . . . . . 37

5.2 User Looking Outside the Frame . . . . . . . . . . . . . . . . 38

5.3 Extreme Head Positions . . . . . . . . . . . . . . . . . . . . . 39

5.4 Simulated Drowsy State . . . . . . . . . . . . . . . . . . . . . 40

# List of Tables

5.1 Summary of Results . . . . . . . . . . . . . . . . . . . . . . . 41

## Contents

[**Dedication** **i**](#_Toc42237)

[**Acknowledgement** **ii**](#_Toc42238)

[**Abstract** **iii**](#_Toc42239)

[**List of Figures** **iv**](#_Toc42240)

[**List of Tables** **v**](#_Toc42241)

**Table of Contents ix**

1. **Introduction 1**
   1. Background of the Study . . . . . . . . . . . . . . . . . . . . . 1
   2. Problem Statement . . . . . . . . . . . . . . . . . . . . . . . . 2
   3. Objectives . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4
   4. Scope of the Project . . . . . . . . . . . . . . . . . . . . . . . 5
   5. Significance of the Study . . . . . . . . . . . . . . . . . . . . . 7
   6. Beneficiaries of the Project . . . . . . . . . . . . . . . . . . . . 9
   7. Structure of the Report . . . . . . . . . . . . . . . . . . . . . . 11
2. **Literature Review 13**

|  |  |
| --- | --- |
| 2.1 Introduction . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 13 |
| 2.2 Drowsiness and Road Safety . . . . . . . . . . . . . . . . . . . | 13 |
| 2.2.1 Face and Eye Detection . . . . . . . . . . . . . . . . . | 14 |
| 2.2.2 Facial Landmark Detection . . . . . . . . . . . . . . . | 14 |
| 2.2.3 Eye Aspect Ratio (EAR) . . . . . . . . . . . . . . . . . | 15 |
| 2.2.4 Mouth Aspect Ratio (MAR) . . . . . . . . . . . . . . . | 15 |
| 2.2.5 Head Pose Estimation . . . . . . . . . . . . . . . . . . | 15 |
| 2.3 Machine Learning Models in Drowsiness Detection . . . . . . . | 15 |
| 2.4 Limitations of Existing Approaches . . . . . . . . . . . . . . . | 16 |
| 2.5 Continual Learning in Driver Monitoring . . . . . . . . . . . . | 16 |
| 2.5.1 Concept of Continual Learning . . . . . . . . . . . . . | 16 |
| 2.5.2 Use of SGDClassifier . . . . . . . . . . . . . . . . . . . | 17 |
| 2.6 Summary . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 17 |
| **3 Methodology** | **18** |
| 3.1 System Overview . . . . . . . . . . . . . . . . . . . . . . . . . | 18 |
| 3.2 Feature Extraction . . . . . . . . . . . . . . . . . . . . . . . . | 19 |
| 3.3 Calibration Mode . . . . . . . . . . . . . . . . . . . . . . . . . | 20 |
| 3.4 Continual Learning . . . . . . . . . . . . . . . . . . . . . . . . | 22 |
| 3.5 Data Flow Diagram . . . . . . . . . . . . . . . . . . . . . . . . | 23 |
| **4 System Design and Implementation** | **25** |
| 4.1 Technologies Used . . . . . . . . . . . . . . . . . . . . . . . . . | 25 |
| 4.2 Folder Structure . . . . . . . . . . . . . . . . . . . . . . . . . . | 27 |
| 4.3 Running the System . . . . . . . . . . . . . . . . . . . . . . . | 30 |
| **5 Results and Discussion** | **33** |

|  |  |
| --- | --- |
| 5.1 Introduction . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 33 |
| 5.2 Experimental Setup . . . . . . . . . . . . . . . . . . . . . . . . | 34 |
| 5.2.1 Hardware Setup . . . . . . . . . . . . . . . . . . . . . . | 34 |
| 5.2.2 Software Setup . . . . . . . . . . . . . . . . . . . . . . | 34 |
| 5.2.3 Participants . . . . . . . . . . . . . . . . . . . . . . . . | 35 |
| 5.2.4 Testing Procedure . . . . . . . . . . . . . . . . . . . . . | 35 |
| 5.3 Results from User Tests . . . . . . . . . . . . . . . . . . . . . | 36 |
| 5.3.1 User Looking Directly into Frame (Stable Condition) . | 36 |
| 5.3.2 User Looking Outside the Frame . . . . . . . . . . . . | 36 |
| 5.3.3 Extreme Head Positions . . . . . . . . . . . . . . . . . | 37 |
| 5.3.4 Simulated Drowsy State . . . . . . . . . . . . . . . . . | 38 |
| 5.3.5 Summary of Results . . . . . . . . . . . . . . . . . . . | 39 |
| 5.4 Discussion of Calibration’s Role . . . . . . . . . . . . . . . . . | 39 |
| 5.5 Challenges and Limitations . . . . . . . . . . . . . . . . . . . | 42 |
| 5.5.1 Extreme Head Positions . . . . . . . . . . . . . . . . . | 42 |
| 5.5.2 Lighting Conditions . . . . . . . . . . . . . . . . . . . . | 42 |
| 5.5.3 Eyeglasses and Sunglasses . . . . . . . . . . . . . . . . | 43 |
| 5.5.4 False Positives and Negatives . . . . . . . . . . . . . . | 43 |
| 5.5.5 Hardware Dependence . . . . . . . . . . . . . . . . . . | 43 |
| 5.5.6 Real-World Application Constraints . . . . . . . . . . . | 44 |
| 5.6 Overall Discussion . . . . . . . . . . . . . . . . . . . . . . . . | 44 |
| 5.7 Conclusion of Results . . . . . . . . . . . . . . . . . . . . . . . | 46 |
| **6 Conclusion and Recommendations** | **47** |
| 6.1 Conclusion . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 47 |
| 6.2 Recommendations . . . . . . . . . . . . . . . . . . . . . . . . . | 48 |

6.3 Final Remark . . . . . . . . . . . . . . . . . . . . . . . . . . . 50

**References 51**

**Appendix 53**

**Chapter 1**

**Introduction**

## 1.1 Background of the Study

Fatigue is increasingly recognized as one of the most significant contributors to road traffic accidents, particularly among commercial, shift, and longdistance drivers. Prolonged periods of driving without adequate rest can lead to a reduction in alertness, slower reaction times, and impaired decisionmaking, all of which are critical factors in maintaining road safety. Studies by organizations such as the World Health Organization (World Health Organization (WHO) 2015) and the National Highway Traffic Safety Administration (National Highway Traffic Safety Administration (NHTSA) 2017) have shown that drowsy driving is as dangerous as drunk driving, yet it often remains undetected or underreported due to the absence of clear indicators and standardized detection protocols.

Traditional driver monitoring systems have attempted to mitigate this issue using indirect vehicle-based indicators such as lane deviation, erratic steering behavior, or sudden braking patterns. While these systems provide useful signals under controlled conditions, they suffer from critical limitations when applied in real-world environments. Factors such as road quality, vehicle type, driver experience, and even weather conditions can affect the reliability of these external indicators. Furthermore, such systems often fail to account for individual differences in driving style or behavior, leading to high rates of false positives or missed detections.

With the rapid advancement of computer vision, facial analysis, and realtime machine learning, there is now an opportunity to design systems that observe the driver directly rather than relying solely on vehicle behavior. By monitoring visual cues such as eye closure rate, blinking patterns, and yawning frequency through a standard camera, a more direct and accurate estimation of a driver’s drowsiness level can be achieved. Integrating these techniques with adaptive learning models further enhances the ability of such systems to personalize their thresholds and responses based on the unique facial characteristics and behavioral patterns of individual drivers.

This project aims to leverage these technological developments by implementing a vision-based driver drowsiness detection system that incorporates both traditional facial landmark monitoring and modern continual learning algorithms. Through this approach, the system can achieve real-time detection accuracy, personalization, and adaptability, even in resource-constrained environments. Such a system has the potential to significantly reduce fatiguerelated accidents and improve road safety, especially in developing regions where access to sophisticated driver-assist technologies remains limited.

## 1.2 Problem Statement

Drowsiness while driving poses a significant threat to road safety, contributing to a large number of traffic accidents and fatalities worldwide. Despite growing awareness, many existing driver monitoring systems remain fundamentally limited in their approach and effectiveness. Traditional methods, such as steering pattern analysis, lane-keeping monitoring, and vehicle-based sensors, rely heavily on indirect indicators of fatigue. These methods often fail in scenarios involving poor road conditions, unusual driver behavior, or inconsistent driving environments - situations that are especially common in many developing countries.

To address these limitations, researchers have increasingly turned to visionbased techniques that monitor the driver’s face using a camera to detect signs of fatigue such as frequent blinking, eye closure, and yawning. However, many of these systems rely on fixed thresholds for key features like Eye Aspect Ratio (EAR) and Mouth Aspect Ratio (MAR), which are not universally accurate across individuals. Facial structure, behavioral tendencies, and even camera angles can significantly affect the reliability of such static threshold-based systems.

Moreover, a critical shortfall in many of these solutions is their lack of adaptability. Most current drowsiness detection systems are rigid - they do not learn from the user’s behavior over time, nor do they adjust their parameters based on live data. This results in high false-positive or false-negative rates when used across different drivers or environments. The absence of real-time learning makes these systems less practical for long-term deployment and reduces user trust and effectiveness in real-world settings.

Additionally, deployment challenges such as computational overhead, lack of hardware flexibility, and system complexity further limit adoption. Especially in low-resource or embedded environments, there is a need for a lightweight, adaptive solution that can continuously learn and improve its accuracy without relying on cloud-based systems or expensive hardware.

In summary, there exists a clear gap in the development of an effective, real-time, personalized driver drowsiness detection system that can adapt to individual users while remaining computationally efficient and easy to deploy.

## 1.3 Objectives

* **Design a facial-landmark-based driver drowsiness detection system**: The primary goal is to build a system that can detect drowsiness using facial landmarks, such as the eyes and mouth. By leveraging tools like Dlib and OpenCV, the system will extract precise facial features and track changes associated with fatigue, such as prolonged eye closure or yawning. This visual-based approach ensures a non-intrusive and accurate method for assessing driver alertness.
* **Implement real-time monitoring of eye and mouth movements using EAR and MAR**: The system will continuously calculate the Eye Aspect Ratio (EAR) and Mouth Aspect Ratio (MAR) from live webcam input. These ratios help quantify eye closure and yawning behaviors, which are strong indicators of drowsiness. Real-time computation ensures that the system can provide immediate feedback or alerts to the driver when signs of fatigue are detected.
* **Introduce a calibration mode for user-specific threshold tuning**: Since facial structures and behaviors vary from person to person,

a one-size-fits-all model is not ideal. This objective focuses on building a calibration phase where the user is asked to remain alert and simulate drowsy behavior. This data is used to personalize the system by computing custom EAR and MAR thresholds or training a lightweight model. This helps reduce false positives and improves the system’s

overall reliability for different users.

* **Integrate continual learning with partial fit to adapt the model on-the-fly**: Instead of relying on a static model, the system will use a machine learning algorithm (e.g., SGDClassifier) that supports continual learning via partial fit. This means the model can be updated in real time using new EAR and MAR data without retraining from scratch. This objective allows the system to adapt to the user’s changing behaviors over time (e.g., due to tiredness patterns, lighting conditions, or camera position), improving long-term accuracy and performance.

## 1.4 Scope of the Project

This project focuses on developing a desktop-based driver drowsiness detection system that operates in real time using facial landmarks extracted from webcam video input. The core idea is to utilize computer vision and machine learning to non-invasively monitor driver alertness based on visual cues such as eye closure and yawning frequency.

**Platform**: The solution is designed as a Windows desktop application, built with Python and compatible libraries such as OpenCV, Dlib, Scikit-learn, and Pygame. It runs locally on a computer with no requirement for internet connectivity, making it suitable for low-resource settings.

**Input Source**: A standard webcam serves as the sole input device. No additional sensors (e.g., IR sensors, accelerometers, or steering wheel monitors) are used. This keeps the setup simple, affordable, and accessible for general use.

**Feature Extraction**: The system uses Dlib’s 68-point facial landmark predictor to extract relevant regions of interest (eyes and mouth). From these, it calculates:

* Eye Aspect Ratio (EAR) to detect eye closure or blinking patterns.
* Mouth Aspect Ratio (MAR) to monitor yawning frequency.

**Real-time Monitoring and Feedback**: The application continuously processes the webcam feed to detect signs of drowsiness and provide instant feedback through visual display and an audible alarm when the driver appears drowsy or inattentive.

**Calibration Mode**: A dedicated calibration phase allows the system to learn the individual user’s facial characteristics while awake and simulate drowsy behavior. This enables user-specific threshold tuning or model training for improved accuracy.

**Machine Learning and Continual Learning**: A lightweight model (SGDClassifier) is trained during calibration and continuously updated with new data using partial fit. This continual learning loop enables adaptation over time to user-specific behavior without requiring full retraining.

**Output**: The system displays annotated video in a GUI with real-time EAR, MAR values, and status (Awake / Drowsy / No Face). It also triggers an audible alarm (using Pygame) when drowsiness is detected.

**Limitations and Exclusions**:

* The system does not integrate with car hardware such as the steering wheel, brakes, or ignition systems.
* It is not deployed as a mobile app or embedded in a vehicle’s onboard system.
* It does not use external sensors like EEG, heart rate monitors, or steering behavior trackers.
* The project is a prototype and is not certified for commercial or critical automotive applications.
* Autonomous driving functionalities (e.g., controlling the vehicle) are not part of this project.

## 1.5 Significance of the Study

This project addresses a critical need in the domain of road safety, specifically the detection and prevention of driver drowsiness, which remains a major contributor to vehicular accidents globally. The significance of this work lies in its emphasis on accessibility, adaptability, and affordability, particularly tailored for developing countries where high-end safety systems are often unavailable due to cost or infrastructure constraints.

**Key Contributions**:

* **Low-Cost and Non-Invasive Monitoring**: The system relies solely on a webcam and open-source libraries, avoiding expensive hardware or proprietary equipment. This makes it highly accessible for lowbudget implementations, including public transport vehicles, taxis, and commercial fleets in developing regions.
* **Real-Time Drowsiness Detection Using Facial Landmarks**: By analyzing real-time EAR and MAR values, the system provides a responsive and immediate assessment of driver alertness. Unlike systems that rely on post-incident data (e.g., crash sensors), this system acts proactively by issuing audible warnings before a critical event occurs.
* **User-Specific Adaptability via Calibration**: One of the most innovative aspects of this study is its calibration mode, which allows the system to learn baseline values from each unique user. This helps reduce false positives and negatives that commonly occur in fixed-threshold systems and ensures better detection accuracy across different face shapes, skin tones, and behaviors.
* **Continual Learning for Long-Term Accuracy**: Through the use of online learning techniques (partial fit from SGDClassifier), the system continuously updates its model during use. This mimics human-like learning and makes it possible to adapt to changes over time, such

as fatigue patterns, lighting conditions, or even facial changes (e.g., glasses, beards, aging).

* **Potential for Integration into Larger Systems**: The architecture of the project is modular and lightweight, making it suitable for future integration into vehicle infotainment systems, dashboard cameras, or broader IoT-based fleet monitoring solutions. It lays the foundation for personalized driver monitoring systems in commercial applications.
* **Contribution to Research and Education**: Academically, this project contributes to the growing body of work in human-centered AI and adaptive safety systems. It serves as a practical example of applying machine learning in a real-time, resource-constrained environment relevant for future student researchers, software engineers, and automotive designers.

## 1.6 Beneficiaries of the Project

This project is designed to benefit multiple stakeholders, both in the short and long term. The following groups are expected to gain from the implementation or adaptation of the proposed Driver Drowsiness Detection System:

* **Commercial Drivers and Transport Operators**: Drivers of longhaul trucks, buses, and taxis who are especially prone to fatigue due to extended working hours can use this system to receive early warnings and prevent drowsiness-related incidents.
* **Fleet Management Companies**: Companies managing fleets can integrate this system into their operations to improve driver safety, reduce accident-related costs, and ensure compliance with transportation safety regulations.
* **Developing Countries and Low-Income Communities**: With its low hardware requirements and open-source software design, this system is ideal for cost-sensitive markets that lack access to advanced vehicle safety systems.
* **Automotive Startups and Safety Tech Developers**: The project

provides a modular, scalable foundation for integrating real-time behavioral monitoring into commercial vehicles or advanced driver-assistance systems (ADAS).

* **Researchers and Students**: This system demonstrates a practical application of facial landmark detection, continual learning, and realtime computer vision. It can serve as a learning resource for future academic research in AI-driven safety systems, human-computer interaction, and computer vision.
* **Government and Regulatory Bodies**: As road safety becomes a

public concern, policymakers and transportation authorities may consider this system a model for affordable, scalable safety technologies in national transportation planning.

## 1.7 Structure of the Report

This report is organized into several chapters that systematically guide the reader through the conceptualization, development, and evaluation of the Driver Drowsiness Detection System. The structure is as follows:

* **Chapter One: Introduction**: This chapter introduces the project by presenting the background, problem statement, objectives, scope, significance, beneficiaries, and timeline. It sets the foundation for understanding the rationale and expected impact of the system.
* **Chapter Two: Literature Review**: A comprehensive review of existing drowsiness detection technologies is conducted, including traditional and computer vision-based approaches. It highlights current research gaps, the limitations of existing systems, and how this project seeks to address those challenges.
* **Chapter Three: Methodology**: This chapter outlines the system requirements, architecture, and design methodology. It explains how facial landmarks are used, defines the features (EAR and MAR), and details the integration of continual learning using machine learning algorithms.
* **Chapter Four: System Implementation and Design**: A technical walkthrough of how the system was developed is provided in this chapter. It includes setup environments, software dependencies, code structure, calibration logic, model training, and real-time monitoring functionalities.
* **Chapter Five: Results and Discussion**: Here, the system is tested under various scenarios to assess its performance, accuracy, and reliability. Evaluation metrics and test results are analyzed, including strengths, limitations, and potential areas for refinement.
* **Chapter Six: Conclusion and Recommendations**: The final

chapter summarizes the project achievements, reflects on challenges encountered, and offers recommendations for future improvements and expansions. It also suggests potential applications and directions for further research.

* **References and Appendices**: This section contains all referenced academic and technical materials, along with additional content such as screenshots, code excerpts, test results, and user feedback forms, where applicable.

**Chapter 2**

**Literature Review**

## 2.1 Introduction

Driver drowsiness has emerged as a leading factor in road accidents worldwide. Studies indicate that a significant proportion of accidents are caused by drivers who are fatigued or inattentive (World Health Organization

(WHO) 2015, National Highway Traffic Safety Administration (NHTSA) 2017). These accidents not only result in fatalities but also in long-term injuries, property damage, and economic losses. Consequently, researchers have explored ways to develop automated drowsiness detection systems using computer vision and machine learning.

This chapter reviews existing work in driver drowsiness detection, with emphasis on face and eye detection, facial landmarks, feature extraction, and classification. It also highlights the limitations of existing approaches and identifies research gaps that informed the present study.

## 2.2 Drowsiness and Road Safety

Drowsy driving is often compared to drunk driving due to its effects on cognitive functions, reaction times, and decision-making abilities (Philip et al.

2005). According to the National Highway Traffic Safety Administration

(National Highway Traffic Safety Administration (NHTSA) 2017), more than 90,000 crashes annually in the United States are directly linked to drowsy driving. Similarly, the World Health Organization (World Health Organization (WHO) 2015) emphasizes that fatigue remains a critical risk factor in traffic safety worldwide.

### 2.2.1 Face and Eye Detection

The detection of facial regions is the first step in many vision-based driver monitoring systems. Early methods relied on simple object detection algorithms such as Haar Cascades and Histogram of Oriented Gradients (HOG). Haar Cascades, introduced by Viola & Jones (2001), became popular due to their real-time capability, though they are sensitive to lighting and occlusion. HOG-based methods (Dalal & Triggs 2005) offered improved robustness but were computationally intensive.

With the growth of deep learning, convolutional neural networks (CNNs) have been employed for face and eye detection (Zhang et al. 2016), but their integration into low-power systems such as in-vehicle monitoring units remains a challenge.

### 2.2.2 Facial Landmark Detection

Facial landmark detection is a crucial step in extracting discriminative features for drowsiness detection. The most widely used tool is Dlib’s 68-point facial landmark detector, developed by King (2009). This model can identify key facial regions such as the eyes, nose, and mouth. While effective under controlled conditions, landmark detection may degrade under poor illumination, extreme poses, or occlusions (Kazemi & Sullivan 2014).

### 2.2.3 Eye Aspect Ratio (EAR)

Soukupova´ & C´echˇ (2016) introduced the Eye Aspect Ratio (EAR), which computes the ratio of distances between specific eye landmarks. A persistently low EAR indicates eye closure and has been used as a reliable metric for blink detection and drowsiness monitoring. EAR is computationally efficient and robust to moderate head movements.

### 2.2.4 Mouth Aspect Ratio (MAR)

Similarly, the Mouth Aspect Ratio (MAR) is used to track yawning. Studies

(Abtahi et al. 2014) have shown that prolonged yawning combined with low EAR values is a strong indicator of driver fatigue.

### 2.2.5 Head Pose Estimation

Head pose has also been used to detect distraction and fatigue. Methods based on facial landmarks (Murphy-Chutorian & Trivedi 2009) estimate whether the driver is facing forward, downward, or sideways. Significant deviations from forward gaze can indicate inattention.

## 2.3 Machine Learning Models in Drowsiness Detection

Various classifiers have been applied to combine EAR, MAR, and head pose into reliable predictions. Support Vector Machines (Cortes & Vapnik 1995) are effective for binary classification but can be slow with large datasets. Random Forests (Breiman 2001) are robust to noise and overfitting. k-Nearest Neighbors (Cover & Hart 1967) is simple but computationally expensive for real-time tasks. Neural networks, though powerful, often require more training data and computational resources (Zhang et al. 2016).

## 2.4 Limitations of Existing Approaches

Despite their progress, existing systems face challenges:

1. **Lighting Variations**: Performance drops in low light or high glare

conditions.

1. **Occlusions**: Sunglasses or masks can obscure critical facial regions.
2. **Head Movements**: Extreme angles reduce the accuracy of landmark

detection.

1. **Generalization**: Systems trained on limited datasets may not adapt well across diverse populations (Abtahi et al. 2014).

These limitations highlight the need for adaptive methods capable of learning user-specific baselines.

## 2.5 Continual Learning in Driver Monitoring

### 2.5.1 Concept of Continual Learning

Continual learning allows models to update dynamically as new data arrives without retraining from scratch (Parisi et al. 2019). In driver monitoring, this means the system can calibrate to individual drivers’ natural eye and mouth patterns, reducing false alarms.

### 2.5.2 Use of SGDClassifier

Scikit-learn’s SGDClassifier, introduced by Pedregosa et al. (2011), is particularly suited for continual learning since it supports partial fitting. By incrementally updating weights, it allows the system to refine its performance based on real-time driver behavior.

## 2.6 Summary

This chapter reviewed the foundations of drowsiness detection, including facial detection, landmark identification, and feature extraction techniques such as EAR and MAR. It also explored machine learning models and their limitations, emphasizing the importance of robustness under real-world driving conditions. Finally, continual learning was identified as a promising direction for adaptive and user-specific driver monitoring systems, laying the groundwork for the system developed in this study.

**Chapter 3**

**Methodology**

## 3.1 System Overview

The proposed Driver Drowsiness Detection System is designed as a real-time monitoring application that integrates computer vision, machine learning, and audio feedback mechanisms to detect early signs of driver fatigue. The system leverages Python as the primary programming language due to its extensive support for scientific computing, machine learning, and computer vision libraries. The architecture follows a modular design to ensure scalability, maintainability, and ease of integration with potential in-vehicle systems.

At the core of the system, **OpenCV** (Open Source Computer Vision Library) is employed for continuous video capture from a connected webcam or in-vehicle camera module. It facilitates efficient frame-by-frame image processing, enabling real-time performance even on standard hardware.

For facial feature extraction, **Dlib** is used due to its robust facial landmark detection model, which provides 68 key points on a detected face. These landmarks are critical for computing the Eye Aspect Ratio (EAR) and other fatigue-related facial metrics such as blink frequency and eye closure duration. This precise mapping allows the system to detect subtle changes in the driver’s eye state that may indicate drowsiness.

To handle classification and adapt to new driving conditions or user-specific facial features, the system incorporates **Scikit-learn’s SGDClassifier** for continual learning. This ensures that the detection model can be incrementally updated with new data without requiring a full retraining cycle, improving accuracy over time and accommodating different drivers.

The alert mechanism is implemented using **Pygame**, which plays an audio warning when drowsiness is detected. The use of an audible alert ensures that the driver receives immediate feedback without requiring them to visually check a display, thus prioritizing safety.

The system processes the video stream in real-time, extracting features from each frame, classifying them as ”alert” or ”drowsy,” and triggering an appropriate response. This integration of computer vision, machine learning, and real-time feedback aims to reduce fatigue-related accidents by providing timely and reliable drowsiness detection.

## 3.2 Feature Extraction

In the proposed Driver Drowsiness Detection System, feature extraction focuses on quantifiable facial metrics that are closely associated with fatigue indicators. Two primary features are used: the **Eye Aspect Ratio (EAR)** and the **Mouth Aspect Ratio (MAR)**. These metrics are derived from

precise facial landmarks detected by the Dlib library’s pre-trained shape predictor model, which maps 68 reference points on the face.

**Eye Aspect Ratio (EAR)**: The EAR is a robust metric for detecting eye closure and blinking patterns. It is computed by measuring the ratio between the distances of specific vertical and horizontal eye landmarks. When a driver’s eyes are open, the EAR value remains relatively constant; however, during a blink or prolonged eye closure, the EAR drops sharply. By monitoring these fluctuations over time, the system can detect micro-sleeps and extended eye closures, which are strong indicators of drowsiness. The EAR is calculated for both eyes and averaged to minimize the effect of head tilt or partial occlusion.

**Mouth Aspect Ratio (MAR)**: The MAR is used to detect yawning, which is another physiological sign of fatigue. It measures the ratio of vertical mouth opening to the horizontal mouth width using specific mouth landmarks. A yawn results in a sudden and significant increase in MAR, typically sustained for a longer duration than casual speech. By establishing a threshold value, the system can differentiate between normal mouth movements (e.g., talking) and genuine yawns.

Both EAR and MAR values are computed for every frame of the video stream, and these temporal sequences of measurements are passed to the machine learning model for classification. The combination of EAR and MAR enables the system to detect both eye-related and mouth-related drowsiness symptoms, improving detection accuracy and reducing false positives.

## 3.3 Calibration Mode

The Calibration Mode serves as an essential initial setup phase for tailoring the Driver Drowsiness Detection System to an individual driver’s unique facial characteristics. During this mode, the system actively collects Eye Aspect Ratio (EAR) and Mouth Aspect Ratio (MAR) data under both alert and drowsy conditions. This personalized data acquisition helps reduce interperson variability caused by differences in facial geometry, eye size, mouth structure, and natural blinking or yawning patterns.

The calibration process is divided into two stages:

* **Alert Condition Data Collection**: The user is instructed to maintain an alert state by keeping their eyes open and refraining from exaggerated facial movements. EAR and MAR values are recorded continuously to establish baseline measurements for ”awake” behavior. These baselines are critical for setting individualized thresholds and minimizing false alarms.
* **Simulated Drowsiness Data Collection**: The user is guided to mimic drowsiness symptoms such as prolonged eye closure, frequent blinking, and yawning. This generates EAR and MAR samples representing the ”drowsy” class. Collecting these samples in a controlled environment ensures that the system learns accurate feature boundaries between alert and drowsy states.

Once sufficient labeled samples are collected for both conditions, they are used to fit an initial SGDClassifier from Scikit-learn. The SGDClassifier is particularly suited for this application due to its ability to handle incremental learning, allowing the system to be refined over time as more real-world data is collected. The initial model produced during Calibration Mode acts as the starting point for live detection and is subsequently updated during operation to improve adaptability and robustness.

## 3.4 Continual Learning

Continual learning is a key feature of the proposed Driver Drowsiness Detection System, enabling the model to adapt to the driver’s evolving facial behavior patterns over time without requiring complete retraining from scratch. This approach addresses one of the major limitations of static models, which often fail to maintain accuracy when exposed to new conditions such as changes in lighting, camera angle, facial hair growth, eyewear use, or natural variations in blinking and yawning frequency.

During live detection, the system continuously computes Eye Aspect Ratio (EAR) and Mouth Aspect Ratio (MAR) values for each frame. These measurements, along with the corresponding drowsiness predictions, are periodically reviewed. Selected instances - particularly those that are confidently classified - are stored in a temporary dataset. After a predefined interval or when enough samples are gathered, the model is incrementally updated using Scikit-learn’s partial fit method available in the SGDClassifier.

This incremental learning process ensures:

* **User-Specific Adaptation**: The model gradually learns the driver’s personal EAR and MAR ranges, making it more accurate for individual use cases.
* **Environment Adaptation**: The system becomes more robust to changes in lighting, camera position, and background.
* **Low Computational Cost**: Since only new data is used for updates, the process is computationally lightweight and suitable for real-time operation.
* **No Full Retraining Required**: Avoids the need to collect a large dataset or retrain the entire model, which would be time-consuming and resource-intensive.

By continuously refining the classification boundaries, continual learning extends the system’s long-term reliability and reduces performance degradation that typically occurs in static detection systems deployed in dynamic realworld environments.

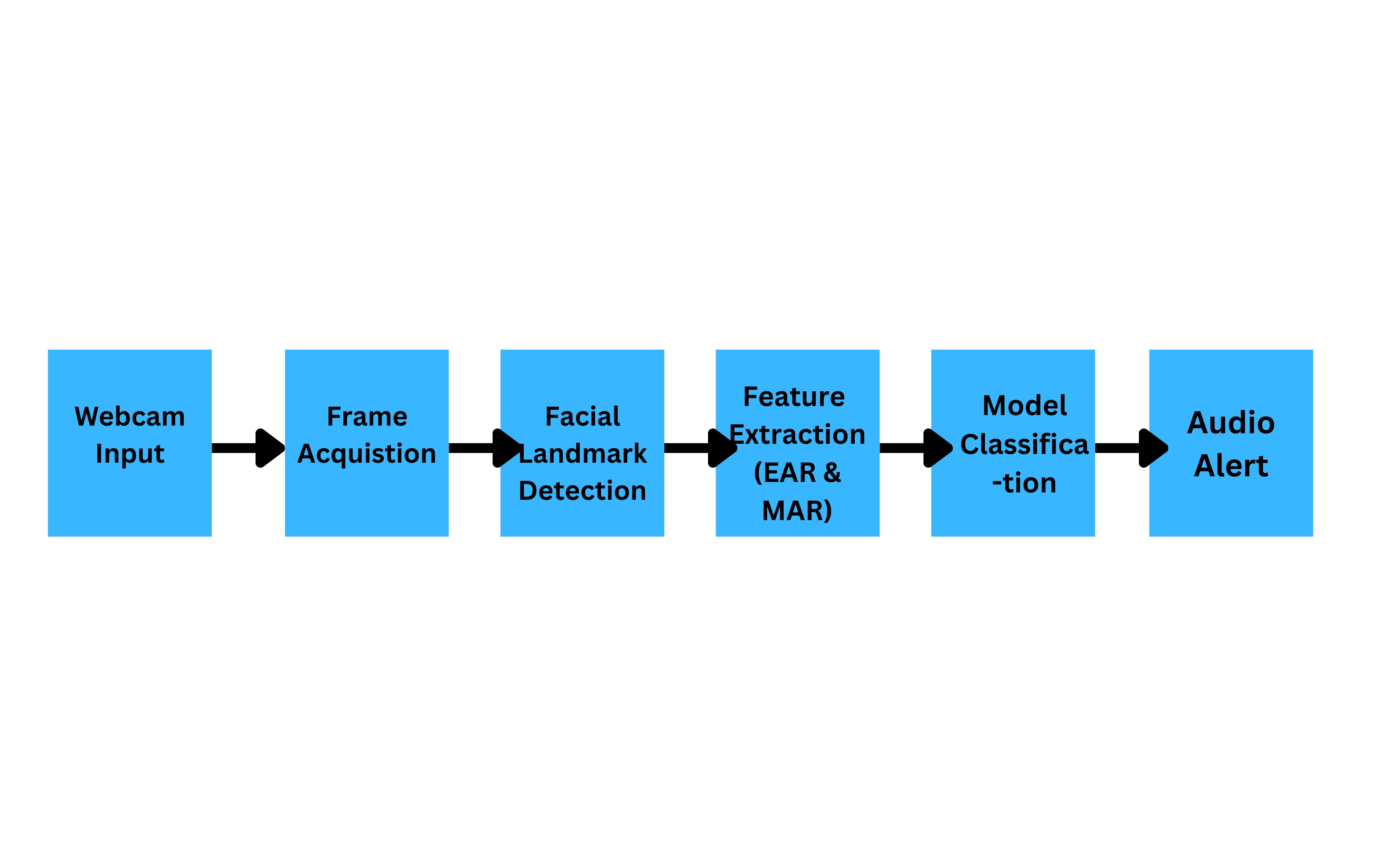
## 3.5 Data Flow Diagram

The data flow for the Driver Drowsiness Detection System illustrates the sequential processing stages from video input to alert generation. The diagram provides a high-level overview of the system’s operational pipeline:

* **Webcam Input**: A live video feed of the driver is captured in real time using a connected webcam or in-vehicle camera module.
* **Frame Acquisition**: Individual frames are extracted from the video stream for processing.
* **Facial Landmark Detection**: Using Dlib’s 68-point facial landmark predictor, the system identifies key regions of interest, including the eyes and mouth.
* **Feature Extraction (EAR and MAR)**: The Eye Aspect Ratio (EAR) and Mouth Aspect Ratio (MAR) are computed from the detected landmarks to quantify eye closure, blinking, and yawning be-

havior.

* **Classification**: Extracted EAR and MAR values are fed into the SGDClassifier, which determines whether the driver is in an ”Alert” or ”Drowsy” state.
* **Alert Mechanism**: If a drowsy state is detected, an audio alert is triggered via Pygame to prompt the driver to regain focus.



**Figure 3.1:** Data Flow Diagram of the Driver Drowsiness Detection System

**Chapter 4**

**System Design and Implementation**

## 4.1 Technologies Used

The development of the Driver Drowsiness Detection System incorporates a range of programming tools and libraries chosen for their efficiency, reliability, and compatibility with real-time computer vision tasks. The following technologies were employed:

* **Python 3.9**: Python was selected as the primary programming language due to its readability, extensive library support, and suitability for rapid prototyping in machine learning and computer vision applications. Version 3.9 was chosen to ensure compatibility with critical dependencies such as Dlib and OpenCV. Python’s rich ecosystem also allows seamless integration of data processing, model training, and alert mechanisms within a single environment.
* **OpenCV**: Open Source Computer Vision Library provides the essential tools for real-time image acquisition and processing. In this system, OpenCV is used to capture frames from the webcam, convert them to grayscale for faster processing, and perform image pre-processing before facial landmark detection. Its high-performance capabilities enable the system to operate with minimal latency, which is critical in safetyrelated applications like drowsiness detection.
* **Dlib with shape predictor 68 face landmarks.dat**: Dlib is utilized for its robust facial landmark detection, which relies on a pretrained model (shape predictor 68 face landmarks.dat) to identify 68 specific points on a detected face. These points map key facial features, including the eyes and mouth, enabling precise computation of the Eye Aspect Ratio (EAR) and Mouth Aspect Ratio (MAR). Dlib’s accuracy and resilience to slight variations in head pose make it well-suited for driver monitoring tasks.
* **Scikit-learn (SGDClassifier)**: The Scikit-learn library is employed to implement the classification component of the system. The SGDClassifier (Stochastic Gradient Descent Classifier) is particularly advantageous because it supports incremental learning through the partial fit method, enabling continual learning during live detection. This allows the model to adapt to each user’s unique facial metrics over time without requiring complete retraining.
* **NumPy 1.24.4**: NumPy is used for efficient numerical operations, including distance calculations between facial landmarks and feature scaling. Version 1.24.4 was specifically chosen to maintain compatibility with Dlib, as newer versions introduced changes that could cause dependency conflicts. NumPy’s optimized array handling ensures that feature extraction processes are computationally efficient, contributing to real-time performance.
* **Pygame**: Pygame is integrated into the system to handle audio output for alert notifications. When drowsiness is detected, Pygame plays a preloaded alarm sound to warn the driver. This approach ensures immediate and non-visual feedback, which is critical for road safety since the driver’s attention should remain on the road rather than onscreen alerts.

## 4.2 Folder Structure

The project is organized into a modular folder structure that ensures clarity, scalability, and ease of maintenance. This design philosophy separates source code, machine learning models, and resource files to prevent confusion and allow for straightforward updates or replacements of individual components without affecting the entire system.

* **baseline drowsiness.py - Main Execution Logic**: This Python script is the central entry point of the Driver Drowsiness Detection System. It orchestrates all major operations, including:
  1. **Initialization**: Loads required libraries, the pre-trained facial landmark model, and the saved SGDClassifier.
  2. **Video Capture**: Uses OpenCV to continuously acquire frames from the webcam.
  3. **Facial Landmark Detection**: Employs Dlib’s

shape predictor 68 face landmarks.dat to extract the 68 key facial points from each frame.

* 1. **Feature Extraction**: Computes the Eye Aspect Ratio (EAR) and Mouth Aspect Ratio (MAR) from the detected landmarks.
  2. **Classification**: Passes EAR and MAR values to the SGDClassifier to predict whether the driver is ”Alert” or ”Drowsy.”
  3. **Audio Alert Triggering**: Plays music.wav using Pygame if

drowsiness is detected.

* 1. **Continual Learning**: Periodically updates the classifier via the partial fit method to adapt to the user’s changing behavior pat-

terns.

By consolidating all core processes in a single file, this script simplifies execution (one command to run the system) while keeping the logic modular for potential migration into separate service layers in the fu-

ture.

* **music.wav - Alarm Audio Resource**: The alarm sound is stored as a .wav file due to its uncompressed format, which ensures minimal latency in playback - a critical factor in safety-critical applications like drowsiness detection. The audio is designed to be:
  1. **High Contrast**: Sharp and attention-grabbing to ensure quick driver response.
  2. **Loop-Capable**: Can be replayed until the system detects that the driver is alert again.
  3. **Lightweight**: Small in file size to enable quick loading into memory by Pygame.

Storing it as a separate resource allows easy customization; different alarm tones can be tested or localized for different regions without modifying the code.

* **models/shape predictor 68 face landmarks.dat - Pre-trained Facial Landmark Model**: This file contains Dlib’s widely used 68point facial landmark predictor, trained on iBUG 300-W datasets. It

is critical for:

* 1. **Eye Detection**: Identifying the 6 landmarks per eye required for EAR computation.
  2. **Mouth Detection**: Identifying the 20 landmarks around the mouth for MAR computation.
  3. **Pose Robustness**: Maintaining accurate landmark positioning even with slight head movements, partial occlusion (e.g., glasses), or varying lighting conditions.

Keeping this model in a models directory provides a clear structure for storing machine learning assets. This approach also supports potential future enhancements, such as adding alternative landmark models (e.g., a lightweight CNN for embedded devices).

## 4.3 Running the System

The Driver Drowsiness Detection System is designed to operate in a userfriendly and interactive manner while maintaining real-time performance. The execution process involves the following steps:

* **Launching the Application**: The user initiates the system by executing the baseline drowsiness.py script in the terminal or an integrated development environment (IDE) that supports Python execution. Upon launch, the system initializes all required modules, including OpenCV for video capture, Dlib for facial landmark detection, the pre-trained shape predictor 68 face landmarks.dat model, and the SGDClassifier (either the default saved model or a newly calibrated

version).

* **Calibration Mode (Optional but Recommended)**:
  + - The user can press the ‘c’ key to activate Calibration Mode.
    - During calibration, the system guides the user through two phases:

∗ **Alert State Recording**: The user is instructed to maintain an alert expression (eyes open, no yawning) while the system records EAR and MAR values for baseline measurements.

∗ **Simulated Drowsiness Recording**: The user simulates drowsy behaviors, such as yawning or prolonged eye closure, so the system can capture ”drowsy” EAR and MAR patterns.

* + - Once enough data is collected, the system fits an initial SGDClassifier tailored to the user’s facial metrics and stores it as default user adapter.pkl for future use.
* **Live Detection Mode**:
  + After calibration (or directly if skipped), the system enters live detection mode.
  + Frames are continuously captured from the webcam, facial landmarks are detected, and EAR/MAR values are computed in real

time.

* + The classifier predicts the driver’s state as either ”Alert” or ”Drowsy” based on current measurements.
* **Alert Mechanism**: If the model predicts a drowsy state, an alarm sound (music.wav) is immediately played via Pygame to prompt the driver to regain focus. The alert continues until the driver’s facial metrics return to an alert state.
* **Continual Learning During Operation**:
  + Throughout live operation, new EAR and MAR samples are periodically stored and used to incrementally update the model using

partial fit.

* + This ensures that the system adapts to changes in the user’s behavior over time, such as variations caused by lighting, fatigue patterns, or physical appearance changes.
* **Exiting the System**: The user can terminate the system at any time by pressing the ‘q’ key, which closes the webcam stream and releases system resources.

**Chapter 5**

**Results and Discussion**

## 5.1 Introduction

This chapter presents the results obtained from the implementation and testing of the Driver Drowsiness Detection System. The aim of this chapter is to evaluate how well the system performs in real-world conditions and to discuss its effectiveness in detecting signs of drowsiness among different users.

The results are organized into several sections. First, the system is tested without calibration to assess its baseline performance using default Eye Aspect Ratio (EAR) and Mouth Aspect Ratio (MAR) thresholds. This is followed by experiments conducted after user calibration, which allows the system to adapt to individual facial differences through an adaptive machine learning model. The performance of the continual learning mechanism, which enables the system to update itself over time, is also evaluated.

Additionally, the results consider the system’s performance under different conditions, such as night mode, face occlusion, and scenarios where no face is detected. Screenshots of the live system interface, along with tables and graphs of accuracy, false positives, and false negatives, are provided to illustrate the findings.

The discussion section then interprets these results, highlighting the strengths, limitations, and practical implications of the system for real-world driver safety applications.

## 5.2 Experimental Setup

### 5.2.1 Hardware Setup

* **Computer**: HP Pavilion Laptop, Intel Core i5, 8GB RAM, Windows

10

* **Camera**: Built-in HD webcam (30 fps)
* **Audio Output**: Laptop speakers used for alarm notification
* **Lighting Conditions**: Tests were conducted in both well-lit environments (daytime) and low-light environments (night mode enabled).

### 5.2.2 Software Setup

* **Programming Language**: Python 3.10
* **Libraries/Frameworks**:
  + OpenCV - real-time video capture and image processing
  + Dlib - facial landmark detection
  + Scikit-learn - machine learning (SGDClassifier for continual learning)
  + Pygame - alarm sound output
  + Imutils, NumPy, SciPy - utilities and mathematical operations
* **Operating System**: Windows 10 (64-bit)

### 5.2.3 Participants

* A total of 3 test subjects (3 males) aged between 20 and 30 years

participated in the experiments.

* Participants had different physical characteristics, including variations in eye size, facial structure, and the presence/absence of eyeglasses.
* Each participant was tested in both awake and drowsy conditions. The drowsy state was simulated by asking users to partially close their eyes, yawn, or tilt their head forward as if sleepy.

### 5.2.4 Testing Procedure

* Each participant first used the system without calibration to assess baseline detection.
* The calibration process was then initiated, during which the system collected eye and mouth data while users simulated awake and drowsy

states.

* After calibration, each participant repeated the test under identical conditions to compare results.
* The system was further tested with night mode enabled and in cases where no face was detected to evaluate robustness.
* Results were recorded in terms of accuracy, false positives, false negatives, and overall reliability. Screenshots of the detection interface were also captured for documentation.

## 5.3 Results from User Tests

Several tests were conducted with different participants under varying conditions to evaluate the performance of the Driver Drowsiness Detection System. In most cases, calibration was performed prior to testing so that the system could adapt to each user’s facial features. The following scenarios were examined:

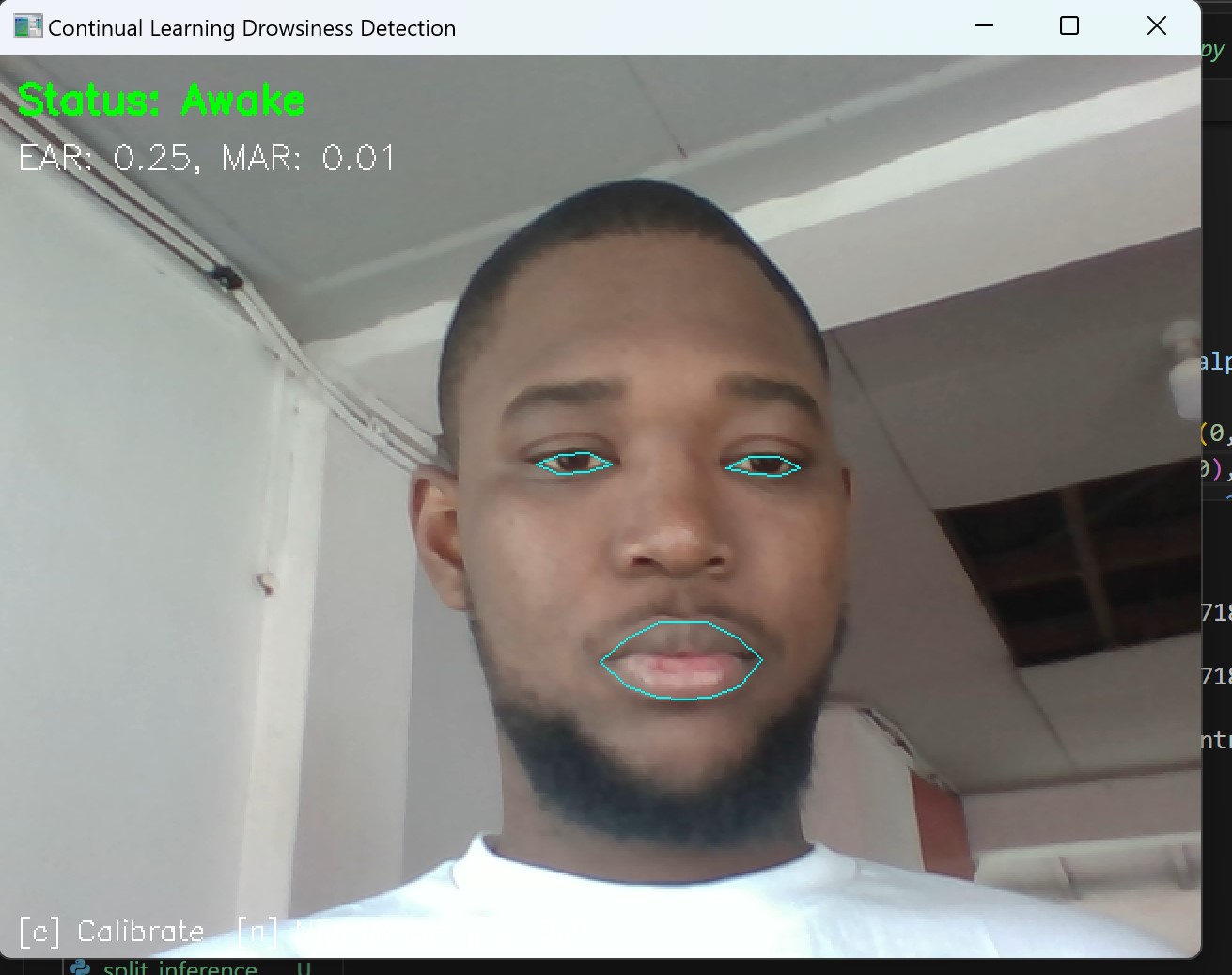
**5.3.1 User Looking Directly into Frame (Stable Con-**

### dition)

* When participants sat in front of the camera and looked directly into the frame, the system consistently detected their facial landmarks.
* The system classified them as ”Awake,” and no alarm was triggered.
* This scenario confirmed that the system could operate reliably during normal driving conditions.

### 5.3.2 User Looking Outside the Frame

* When participants turned their head away from the camera, the system lost track of the facial landmarks.
* The status changed to ”No Face Detected.” If the face remained outside the frame for longer than the threshold (30 frames), the alarm was



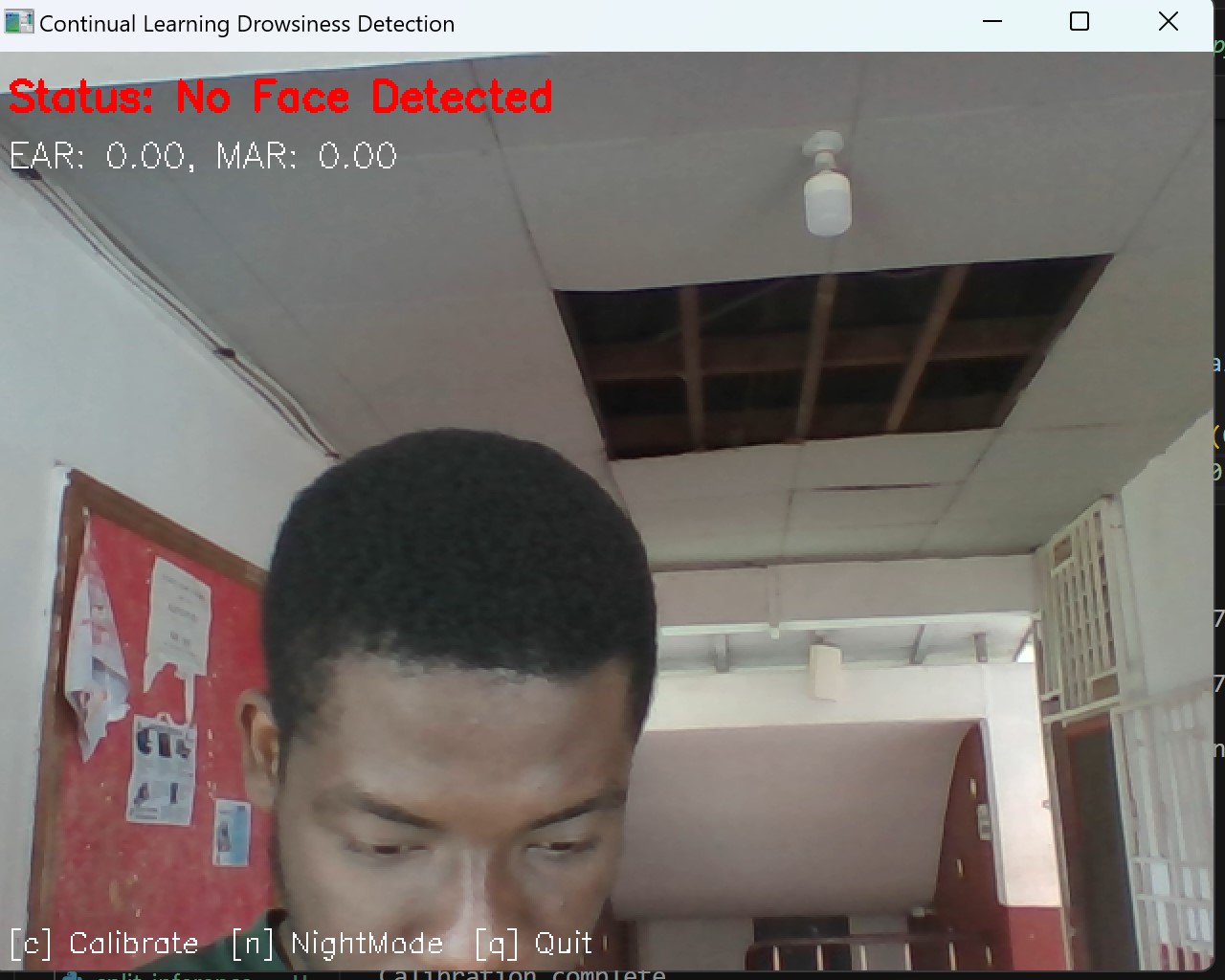
**Figure 5.1:** User Looking Directly into Frame (Stable Condition)

triggered.

* This mechanism proved effective in simulating driver distraction.

### 5.3.3 Extreme Head Positions

* When participants looked too far down or up, the system struggled to detect the eyes.
* In such cases, the status alternated between ”Awake” and ”No Face Detected,” occasionally triggering the alarm unnecessarily.
* Although calibration reduced some false positives, detection remained

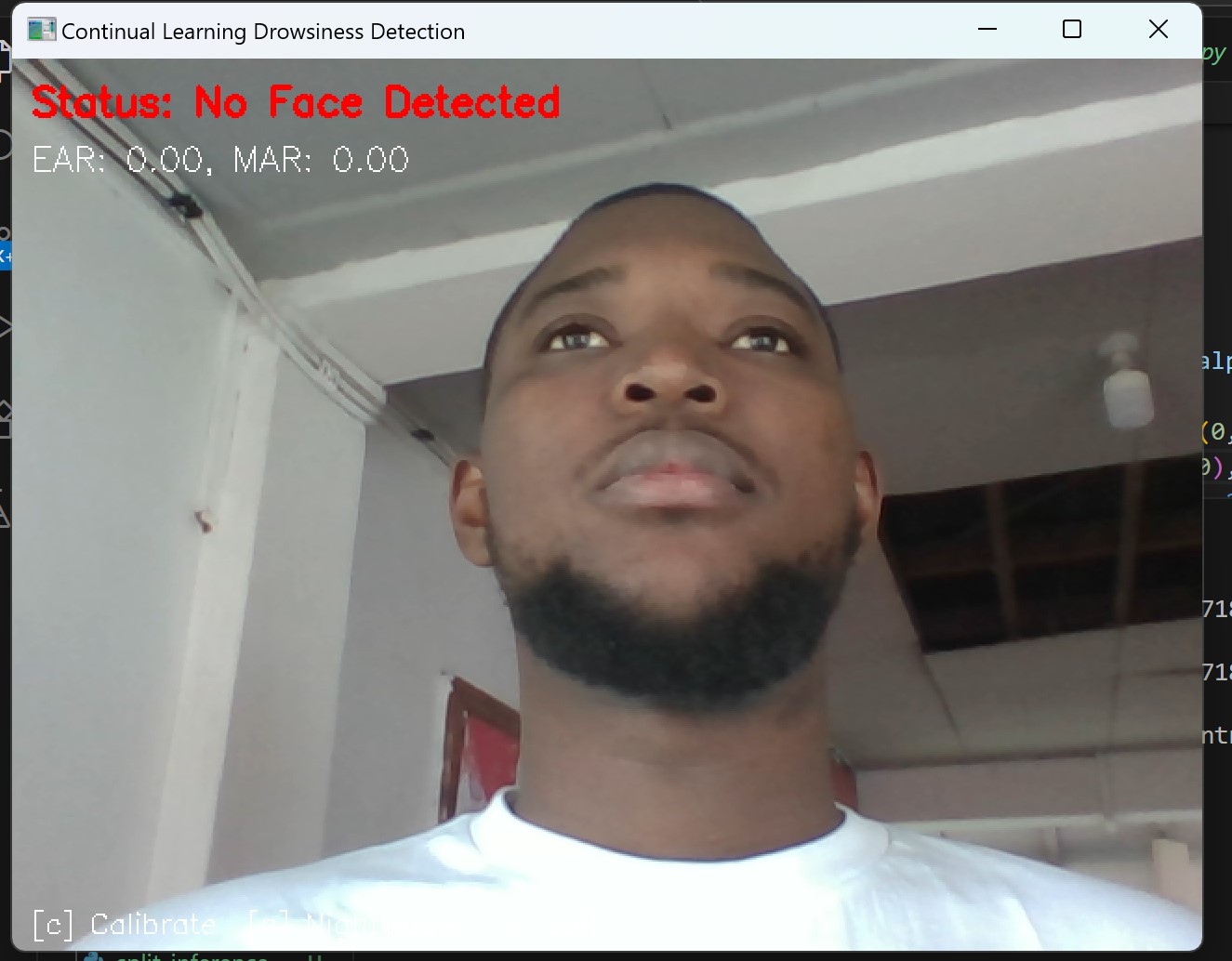


**Figure 5.2:** User Looking Outside the Frame

less reliable in these extreme positions.

### 5.3.4 Simulated Drowsy State

* Participants were asked to simulate drowsiness by closing their eyes slowly or yawning.
* The system successfully detected these signs: a low EAR indicated closed eyes, while a high MAR indicated yawning.
* In both cases, the system classified the state as ”Drowsy” and triggered the alarm.



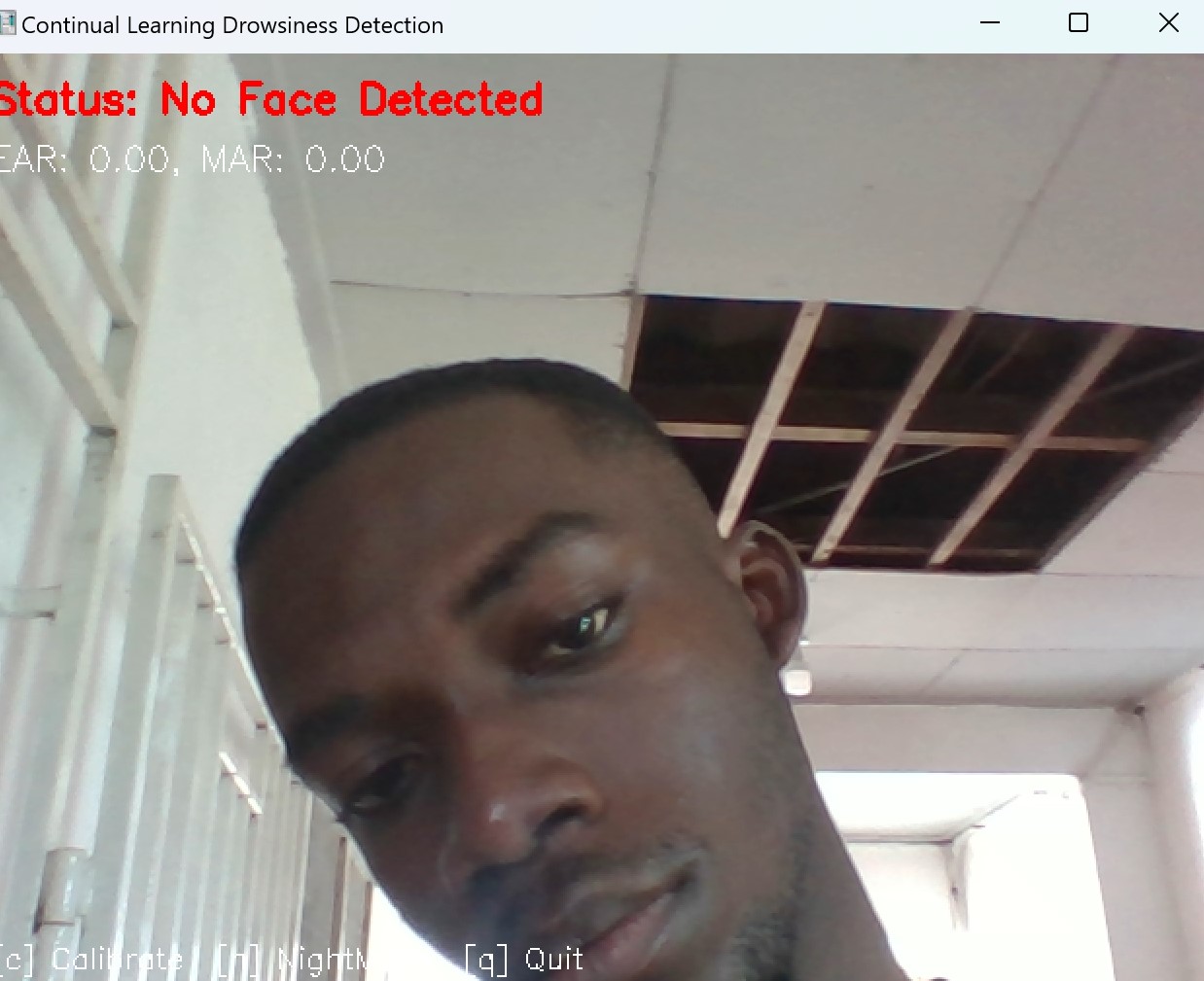
**Figure 5.3:** Extreme Head Positions

**5.3.5 Summary of Results**

The overall observations are summarized in the table below.

## 5.4 Discussion of Calibration’s Role

Calibration played a central role in the performance of the Driver Drowsiness Detection System. Since each participant had unique facial characteristics such as eye size, face shape, and mouth proportions, a fixed threshold for EAR and MAR was not sufficient for all users. By calibrating the system, the adaptive model (SGDClassifier) was trained to recognize what ”awake” and ”drowsy” looked like for each specific individual.



**Figure 5.4:** Simulated Drowsy State

The calibration process improved reliability in several ways:

* **Improved Accuracy Across Users**: After calibration, the system adapted to individual differences, leading to more accurate detection of both awake and drowsy states. This minimized the risk of false positives, where an awake user might be incorrectly classified as drowsy simply because their eyes appeared naturally smaller.
* **Reduction of False Alarms**: During initial trials, extreme head positions or brief eye movements occasionally triggered alarms. With calibration, the system became more tolerant of natural variations, re-

**Table 5.1:** Summary of Results

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Condition** | **Detection**  **Results** | **Alarm Trigger** | **Reliability** |
| User stable,  looking ahead | Awake (Cor-  rect) | No | High |
| User looking outside frame | No Face Detected →  Alarm | Yes | High |
| Extreme head  position | Mixed  (Awake /  No Face) | Sometimes | Medium |
| Simulated drowsiness | Drowsy (Cor-  rect) | Yes | High |

ducing unnecessary alarms. However, in extreme cases where the eyes were completely out of frame, calibration could not fully solve the issue, and the system defaulted to ”No Face Detected.”

* **Adaptability Through Continual Learning**: Beyond initial calibration, the system updated itself continuously during operation. This allowed it to refine its predictions over time and adjust to gradual changes, such as the user blinking more frequently as they became tired. This feature made the system more robust for prolonged use, where static thresholds would normally lose accuracy.
* **Practical Implications**: In a real driving environment, calibration ensures that the system personalizes itself to each driver, which is crucial for deployment in vehicles. Although calibration requires a short setup period, the benefits in terms of accuracy and reduced false alarms outweigh the inconvenience.

Overall, calibration significantly enhanced the system’s effectiveness, making it capable of distinguishing between genuine drowsiness and natural facial variations. While some limitations persisted - especially in cases of extreme head tilts or poor lighting - the calibration mechanism ensured that the system was reliable for normal usage scenarios.

## 5.5 Challenges and Limitations

### 5.5.1 Extreme Head Positions

* When users tilted their head too far upward or downward, the system often lost track of the eyes.
* This caused the status to alternate between ”Awake” and ”No Face Detected,” occasionally triggering false alarms.
* This shows that camera placement and driver posture strongly influence accuracy.

### 5.5.2 Lighting Conditions

* While the Night Mode overlay reduced screen glare, poor ambient lighting still affected facial landmark detection.
* In very low-light environments, the system sometimes failed to detect a face at all, resulting in false alarms.
* This limitation suggests the need for infrared (IR) or night-vision cameras in real vehicle deployment.

### 5.5.3 Eyeglasses and Sunglasses

* With plain transparent eyeglasses, the system was able to detect the eyes and track facial landmarks with minimal issues.
* However, with dark or reflective sunglasses, the eyes were obscured, preventing accurate EAR calculation.
* In such cases, the system displayed ”No Face Detected” or gave incorrect results, sometimes triggering the alarm unnecessarily.

### 5.5.4 False Positives and Negatives

* **False Positives**: Prolonged blinking or looking briefly downward (e.g., checking something on the lap) was sometimes misinterpreted as drowsiness.
* **False Negatives**: Subtle signs of fatigue, such as slower blinking without full closure, were occasionally missed.
* Although calibration and continual learning reduced these errors, they were not completely eliminated.

### 5.5.5 Hardware Dependence

* The built-in laptop webcam used in testing had limited resolution and field of view.
* This affected detection accuracy at extreme angles and under low light-

ing.

* A higher-quality, wide-angle camera could improve system performance.

### 5.5.6 Real-World Application Constraints

* All tests were conducted in a controlled environment with users seated in front of the webcam.
* In actual driving conditions, factors such as vehicle vibrations, changing sunlight, and sudden head movements could further challenge detection accuracy.
* Field testing in real vehicles is therefore necessary before the system can be deployed commercially.

## 5.6 Overall Discussion

The results of this study show that the Driver Drowsiness Detection System is capable of reliably identifying signs of drowsiness and distraction in controlled testing environments. By combining facial landmark analysis with adaptive calibration and continual learning, the system successfully distinguished between awake and drowsy states in most cases.

One of the key strengths of the system is its personalization through calibration. Users have unique facial features, and fixed thresholds for eye and mouth aspect ratios are often insufficient. By calibrating for each individual, the system adapted to natural variations and reduced false alarms. This improved accuracy, especially when distinguishing between small eyes that appear partially closed even when awake and genuine eye closure caused by drowsiness.

The system also proved effective in detecting driver distraction. When participants looked away from the frame, the software correctly reported ”No Face Detected” and triggered an alarm after the threshold period. This feature enhances safety, as distraction can be just as dangerous as drowsiness in real driving scenarios.

However, the study also highlighted several limitations. Extreme head positions and poor lighting reduced detection accuracy, and dark sunglasses prevented the system from detecting the eyes altogether. These challenges reflect the real-world constraints of vision-based systems, which often rely on clear visibility of the eyes and face. Furthermore, false positives (e.g., prolonged blinking interpreted as drowsiness) and false negatives (e.g., subtle fatigue signs missed) remain issues that need further refinement.

Overall, the system demonstrates strong potential as a low-cost and adaptive solution for drowsiness detection. While it performed well in controlled tests, further development is required for deployment in real vehicles, where environmental factors such as vibrations, variable lighting, and driver movement introduce additional complexity. Integration with more advanced sensors, such as infrared cameras or physiological monitoring devices, could improve reliability in future versions.

## 5.7 Conclusion of Results

The testing of the Driver Drowsiness Detection System confirmed its effectiveness in detecting both drowsiness and distraction among users. With calibration enabled, the system adapted to individual differences in facial features and delivered reliable results in stable conditions. The alarm function responded appropriately when users acted drowsy or when no face was detected for a prolonged period, demonstrating the system’s potential as a preventive safety tool.

However, challenges such as extreme head positions, poor lighting, and the use of dark sunglasses revealed the system’s current limitations. Despite these issues, the overall findings indicate that the system provides a solid foundation for driver monitoring applications. With further improvements in hardware and detection algorithms, the system can be adapted for realworld use in vehicles.

**Chapter 6**

**Conclusion and Recommendations**

## 6.1 Conclusion

This project set out to design and implement a real-time Driver Drowsiness Detection System using computer vision and machine learning. The system analyzes facial landmarks - specifically the Eye Aspect Ratio (EAR) and Mouth Aspect Ratio (MAR) - to detect early signs of drowsiness such as prolonged eye closure or yawning. By integrating continual learning through an adaptive classifier, the system adjusts to each user’s facial baseline, improving accuracy across different individuals.

The results presented in Chapter Five demonstrated that the system can reliably detect awake, distracted, and drowsy states in controlled environments. Calibration proved essential in reducing false positives and tailoring the detection model to individual users. The system was also effective in identifying distraction when no face was detected, triggering alarms after the set threshold.

Nevertheless, limitations such as extreme head positions, poor lighting, and the use of dark sunglasses reduced accuracy. Despite these challenges, the project successfully met its objectives, showing that computer vision-based monitoring is a viable and low-cost solution for enhancing road safety. The system provides a strong foundation for further development and eventual integration into real driving scenarios.

## 6.2 Recommendations

While the current system is effective in controlled testing, further improvements are needed for real-world deployment. The following recommendations are proposed:

1. **Mobile and Embedded Deployment**:
   * + Develop lightweight versions of the system for smartphones, tablets, or Raspberry Pi devices so it can be integrated into cars without requiring a laptop.
     + A mobile app version would make the solution more accessible and

scalable.

1. **Enhanced Lighting Support**:
   * + Incorporate infrared (IR) cameras or low-light image sensors to improve detection at night or in poor lighting conditions.
     + This would solve one of the key limitations identified during test-

ing.

1. **Improved Robustness Against Sunglasses and Head Movement**:
   * Use advanced deep learning models (e.g., convolutional neural net-

works or attention-based face detectors) to better handle occlusions caused by sunglasses or extreme head angles.

* + Alternatively, integrate multiple cameras with wider angles for more consistent face tracking.

1. **Hybrid Models with Additional Sensors**:
   * Combine computer vision with other indicators of drowsiness, such as steering wheel sensors, lane-departure monitoring, or wearable devices that track heart rate and eye blinks.
   * A hybrid model would reduce false positives and increase reliability in real driving conditions.
2. **Field Testing in Real Vehicles**:
   * Conduct trials in moving vehicles under different weather, lighting, and road conditions.
   * This would validate the system’s effectiveness in real-world driving and provide critical feedback for refinement.
3. **User Interface and Usability Enhancements**:
   * Provide customizable alarm tones, vibration feedback (for mobile deployment), and a driver-friendly dashboard.
   * Enable cloud-based storage of drowsiness events for fleet monitoring in transportation companies.

## 6.3 Final Remark

In conclusion, this project demonstrates that real-time computer vision can be applied effectively to monitor driver alertness. With further improvements in robustness and integration with other technologies, such systems can play a critical role in reducing road accidents caused by fatigue and distraction, ultimately contributing to safer transportation systems.

## References

Abtahi, S., Omidyeganeh, M., Shirmohammadi, S. & Hariri, B. (2014), ‘Yawdd: A yawning detection dataset’, *Proceedings of the ACM Multimedia Systems Conference (MMSys)* pp. 24–28.

Breiman, L. (2001), ‘Random forests’, *Machine Learning* **45**(1), 5–32.

Cortes, C. & Vapnik, V. (1995), ‘Support-vector networks’, *Machine Learning* **20**(3), 273–297.

Cover, T. & Hart, P. (1967), ‘Nearest neighbor pattern classification’, *IEEE Transactions on Information Theory* **13**(1), 21–27.

Dalal, N. & Triggs, B. (2005), ‘Histograms of oriented gradients for human detection’, *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)* pp. 886–893.

Kazemi, V. & Sullivan, J. (2014), ‘One millisecond face alignment with an ensemble of regression trees’, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* pp. 1867–1874.

King, D. E. (2009), ‘Dlib-ml: A machine learning toolkit’, *Journal of Machine Learning Research* **10**, 1755–1758.

Murphy-Chutorian, E. & Trivedi, M. M. (2009), ‘Head pose estimation in computer vision: A survey’, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **31**(4), 607–626.

National Highway Traffic Safety Administration (NHTSA) (2017), Drowsy driving 2015, Technical report, U.S. Department of Transportation.

**URL:** *https://crashstats.nhtsa.dot.gov*

Parisi, G. I., Kemker, R., Part, J. L., Kanan, C. & Wermter, S. (2019), ‘Continual lifelong learning with neural networks: A review’, *Neural Networks* **113**, 54–71.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O. et al. (2011), ‘Scikit-learn: Machine learning in python’, *Journal of Machine Learning Research* **12**, 2825–2830.

Philip, P., Sagaspe, P., Taillard, J., Valtat, C., Moore, N., Akerstedt, T., Charles, A. & Bioulac, B. (2005), ‘Fatigue, sleep restriction, and driving performance’, *Accident Analysis Prevention* **37**(3), 473–478.

Soukupova´, T. & C´ech, J. (2016), ‘Real-time eye blink detection using facialˇ landmarks’, *Proceedings of the 21st Computer Vision Winter Workshop (CVWW)* pp. 1–8.

Viola, P. & Jones, M. (2001), ‘Rapid object detection using a boosted cascade of simple features’, *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)* **1**, 511–518.

World Health Organization (WHO) (2015), Global status report on road safety 2015, Technical report, WHO Press.

**URL:** *https://www.who.int/publications/i/item/9789241565066*

Zhang, K., Zhang, Z., Li, Z. & Qiao, Y. (2016), ‘Joint face detection and alignment using multitask cascaded convolutional networks’, *IEEE Signal*

*Processing Letters* **23**(10), 1499–1503.

**Appendix**

## Code Snippets

### Face and Eye Detection using Dlib

|  |
| --- |
| import cv2 import dlib  # Load face detector and landmark predictor detector = dlib.get\_frontal\_face\_detector() predictor = dlib.shape\_predictor("models/ shape\_predictor\_68\_face\_landmarks.dat")  # Start video capture cap = cv2.VideoCapture(0)  while True:  ret, frame = cap.read() gray = cv2.cvtColor(frame, cv2.COLOR\_BGR2GRAY) faces = detector(gray)  for face in faces:  landmarks = predictor(gray, face) for n in range(36, 48): # Eye landmarks x, y = landmarks.part(n).x, landmarks.part(n).y cv2.circle(frame, (x, y), 2, (0, 255, 0), -1)  cv2.imshow("Frame", frame) |
| if cv2.waitKey(1) & 0xFF == ord(’q’):  break  cap.release() cv2.destroyAllWindows() |

### Eye Aspect Ratio (EAR) Calculation

|  |
| --- |
| from scipy.spatial import distance as dist  def eye\_aspect\_ratio(eye):  # Compute distances between the vertical eye landmarks   1. = dist.euclidean(eye[1], eye[5]) 2. = dist.euclidean(eye[2], eye[4])   # Compute distance between horizontal landmarks   1. = dist.euclidean(eye[0], eye[3])   # EAR formula ear = (A + B) / (2.0 \* C)  return ear |

### Drowsiness Detection Logic

|  |
| --- |
| EAR\_THRESHOLD = 0.25  FRAME\_THRESHOLD = 20 COUNTER = 0  for face in faces:  landmarks = predictor(gray, face) leftEye = [(landmarks.part(n).x, landmarks.part(n).y) for n in range(36, 42)] |
| rightEye = [(landmarks.part(n).x, landmarks.part(n).y) for n in range(42, 48)]  leftEAR = eye\_aspect\_ratio(leftEye) rightEAR = eye\_aspect\_ratio(rightEye) ear = (leftEAR + rightEAR) / 2.0  if ear < EAR\_THRESHOLD: COUNTER += 1 if COUNTER >= FRAME\_THRESHOLD:  print("Drowsiness Alert!")  else:  COUNTER = 0 |

### Mouth Aspect Ratio (MAR) for Yawning

|  |  |
| --- | --- |
| def mouth\_aspect\_ratio(mouth):  A = dist.euclidean(mouth[2], mouth[10]) | # vertical |
| B = dist.euclidean(mouth[4], mouth[8]) | # vertical |
| C = dist.euclidean(mouth[0], mouth[6])  mar = (A + B) / (2.0 \* C)  return mar | # horizontal |