# CHAPTER 1

# INTRODUCTION

In an era where automotive technology is rapidly evolving, ensuring road safety remains a paramount concern. One of the critical factors contributing to road accidents is driver drowsiness, which can lead to impaired reaction times and compromised decision-making abilities. According to the National Highway Traffic Safety Administration (NHTSA), drowsy driving is responsible for thousands of fatalities and injuries each year. Drowsy driving is a prevalent and dangerous issue, contributing to a significant number of accidents and fatalities on our roads. Addressing this issue is imperative, and modern advancements in artificial intelligence, particularly deep learning, have emerged as a promising methodology for driver drowsiness detection.

.

Driver Drowsiness Detection Systems represent a significant advancement in road safety technology. By offering real-time monitoring and timely warnings, DDD systems have the potential to prevent countless accidents and save lives. Continued research and development focused on accuracy, user acceptance, and ethical considerations are crucial for maximizing the impact of DDD systems. As DDD technology matures and becomes widely adopted, a future with significantly reduced drowsy driving accidents becomes a realistic possibility.

We will explore the burgeoning field of artificial intelligence-based solutions designed to detect signs of drowsiness in real-time, providing drivers with timely warnings and potentially preventing accidents. This report will not only introduce the fundamental concepts of deep learning but also highlight its relevance in mitigating the critical issue of driver drowsiness. As we venture deeper into this topic, it becomes evident that the integration of deep learning into driver drowsiness detection systems has the potential to save lives and reduce accidents on our roads.

By providing an overview of the latest advancements, challenges, and opportunities in this field, this seminar report aims to contribute to the ongoing efforts to make our roads safer through cutting-edge technology and innovative approaches. Currently, drivers rely on self-awareness to identify fatigue. However, recognizing the early signs of drowsiness can be challenging, especially when sleep deprivation accumulates. Traditional methods like stopping for coffee or opening windows provide temporary relief but do not address the underlying issue of sleepiness. DDD systems hold immense potential for revolutionizing road safety by combating drowsy driving. As the technology matures, tackles challenges, and gains wider acceptance, we can look forward to a future with significantly fewer accidents caused by fatigued drivers.

In the subsequent sections, we will explore the historical context and significance of driver drowsiness, delve into the theoretical foundations of deep learning, examine the various data sources and sensors used in drowsiness detection, and assess the performance of cutting-edge deep learning models in real-world scenarios. The motivation behind this report is to shed light on the crucial role that deep learning plays in enhancing road safety by addressing the critical issue of driver drowsiness. By understanding the principles and methodologies involved, we hope to contribute to the collective effort to create safer and more efficient transportation systems.

# CHAPTER 2

# LITERATURE REVIEW

1. **A Driver Drowsiness Detection by applying Deep learning Techniques to sequence of Images. Magan, E.; Sesmero, M.P.; Alonso-Weber, J.M.; Sanchis.**

   In this paper, two different implementations for a driver drowsiness detection system are proposed, where deep learning plays an important role. These systems use images of the driver to identify fatigue symptoms, but instead of predicting whether a driver is tired or not from a single image, in this work, a full sequence of 60 s is used to determine whether the driver is tired or not over the last minute. The first solution proposed uses a model based on deep learning for the estimation of the drowsiness level of the driver, using a combination of a convolutional neural network with a recurrent neural network. The second solution uses fuzzy logic for calculating the fatigue but needs to apply artificial intelligence and deep learning techniques to preprocess the data before using the fuzzy inference system.

2. **Real Time Machine Learning Based Driver Drowsiness Detection using Visual Features. Yaman Alba Davi, Mohammed awad.**

   In Over the past decade, the drowsiness detection field has experienced significant enhancements, due to technological advancements in IoT, sensor miniaturization, and artificial intelligence. This paper has presented a detailed and up-to-date review of the driver drowsiness detection systems that have been implemented in the last ten years. It has described the four main approaches followed in designing DDD systems and categorized them based on the type of drowsiness indicative parameters employed. The paper has provided a detailed description of all the presented systems, in terms of the used features, implemented AI algorithms, and datasets used, as well as the resulting system accuracy, sensitivity, and precision. Furthermore, the review has highlighted the current challenges in the DDD field, discussed the practicality of each DDD system, and discussed the current trends and future directions that aim to utilize affordable, easy-to-use, and practical methods to improve accuracy and reliability.

We conclude by emphasizing that DDD technology has enormous market potential. Many car manufacturers, such as Toyota and Nissan, have recently installed or upgraded driver assistance devices in their products. The artificial intelligence and deep learning fields are developing tremendously. Soon, the DDD systems will most likely evolve, enabling the formation of smart cities.

3. **Real-Time Driver Drowsiness Detection Techniques using Deep Learning. Md. Tanvir Ahammed Dipu1, Syeda Sumbul Hoassain, Yeasir Arafat, Fatama Binta Rafiq.**

The paper described an improved drowsiness detection system based on CNN. The key goal is to ensure that a lightweight approach is applied in integrated devices while maintaining and achieving high efficiency. In the trained model, we only use 250 low-light images. The main improvement done in the future is adding more low-light photos to work well in low-light conditions. Further enhancement of the dataset is required using a yawning dataset as we could not use those annotations for detecting drowsiness. Eyelid closing has been a much more reliable predictor of drowsiness. Many of the systems has built ought to rely on eyelid closing for driver drowsiness detection even though the other behavior is also a predictor like faster blinking time, sneezing, a slow movement of the eyelid, repeated blinking, set eyes, and sagging pose

# CHAPTER 3

# ANALYSIS OF PROBLEM

## 3.1 Current Scenario

Driver drowsiness is a critical safety concern that contributes to a significant percentage of road accidents worldwide. Drowsiness impairs a driver's ability to stay alert and focused, resulting in slower reaction times and poor decision-making. Identifying drowsy drivers in real-time is challenging because the signs of drowsiness—such as prolonged eye blinks, head nodding etc can be subtle and difficult to monitor without assistance.

## 3.2 Solution over the problem:

- Preventive Approach: An automated drowsiness detection system can provide real-time alerts to prevent accidents.
- Reduction in Accident Rates: Early detection and intervention can significantly reduce the risk of drowsy driving-related accidents.
- Urgency: Developing a reliable drowsiness detection system is critical to enhance road safety.
- Potential Impact: A successful system can save lives and improve the well-being of drivers and other road users.

# CHAPTER 4

# DESIGN PROCESS

The design process for a Driver Drowsiness Detection system can be broken down intoseveral steps:

**4.1 Data Collection**: Gather a diverse dataset of images or videos containing drivers in various states of alertness (e.g., alert, drowsy, asleep). Annotate the dataset to label each image or video frame with the corresponding state (e.g., drowsy or not drowsy).

**4.2 Data Preprocessing**: In this step, the collected data is preprocessed to extract relevant features that can be used for classification. Resize and normalize the images to a consistent format. Augment the dataset with techniques like rotation, cropping, and brightness adjustments to increase diversity. Split the data into training, validation, and testing sets.

**4.3 Model Selection**: In this step, the appropriate machine learning or deep learning model is selected. Deep learning models such as Convolutional Neural Networks (CNNs) or Recurrent Neural Networks (RNNs) are often used as they can learn complex patterns from the raw data. Modify the output layer to have two classes: drowsy and not drowsy.

**4.4 Training the Model**: Train the chosen model using the preprocessed data and extracted features. Train the model using the training dataset. Implement data augmentation, dropout, and batch normalization to prevent overfitting. This involves adjusting the parameters of the model to minimize the error between the predicted and actual labels.

**4.5 Model Evaluation**: Evaluate the performance of the trained model using a separate testing dataset. The evaluation can be done using metrics such as accuracy, precision, recall, and F1 score.

**4.6 Deployment**: Integrate the model into a real-time video processing pipeline. Utilize technologies like OpenCV for video capture and processing. Apply the model to each of the frame of the video to detect drowsiness.

**4.7 Continual Improvement**: Continually improve the recognition system by collecting new data, fine-tuning the model, and updating the interface based on user feedback.

## 4.8 FLOW DIAGRAM

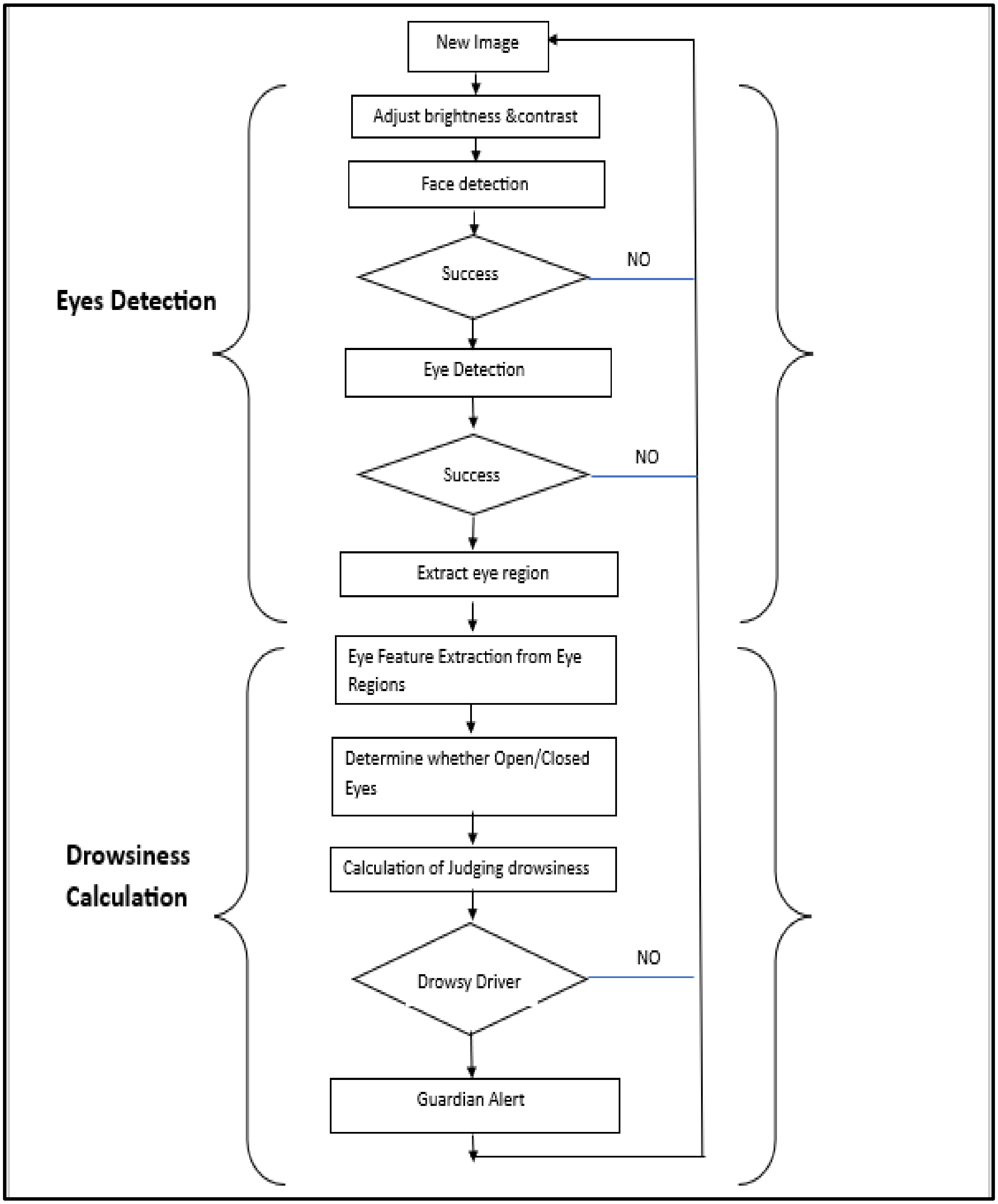


**Figure 4.1**: Flow diagram of Driver Drowsiness Detection System.

## 4.9 Haar Cascade Classifier

- **Face Detection**: The first step is to locate the driver's face in the video frame. A pre-trained Haar cascade classifier for faces is used to achieve this.
- **Eye Detection**: Once the face is identified, another Haar cascade classifier specifically trained for eyes is used to find the driver's eyes within the face region.
- Haar cascade is a machine learning-based approach for object detection. It is commonly used for detecting faces in images and videos.
- The concept behind Haar cascade is to train a classifier using positive and negative samples of the object to be detected. The classifier then uses a set of Haar-like features to identify the object in new images or frames of a video. Haar Cascade is a powerful tool in driver drowsiness detection as it allows for real-time monitoring of the driver's facial expressions and can trigger alerts or interventions when drowsiness is detected.

## 4.10 Transfer Learning

- Transfer learning is a powerful technique in deep learning that allows us to leverage pre-trained models on a new task. In driver drowsiness detection, it can significantly reduce training time and improve accuracy compared to training a model from scratch.
- Building a deep learning model for driver drowsiness detection requires a massive amount of labeled data containing images or videos of drivers in various states (drowsy, alert, etc.). This data collection and labeling process can be expensive and time-consuming.

- **Transfer Learning to the Rescue**:

1. **Pre-trained Model**: The core idea is to utilize a pre-trained deep learning model on a vast image classification dataset like ImageNet. These models, like VGG16, ResNet50, or InceptionV3, have already learned powerful feature extraction capabilities from a massive amount of generic image data (objects, animals, scenes).
2. **Feature Extraction**: We extract features from the pre-trained model's intermediate layers. These layers capture generic visual concepts that can be beneficial even for a different task like drowsiness detection.

3. **Fine-tuning**: We remove the final classification layers of the pre-trained model and add new layers specific to drowsiness detection. These new layers learn to classify the extracted features into drowsiness states (drowsy, alert).
4. **Evaluation**: Evaluate the model's performance on a separate test dataset to assess its accuracy in real-world scenarios.
5. **Reduced Training Time**: By leveraging pre-trained weights, the model requires significantly less training data and time compared to training from scratch. This makes it more practical for real-world scenarios.
6. **Improved Accuracy**: The pre-trained model already has strong feature extraction capabilities, which can be beneficial for learning drowsiness-specific features in the new layers

Overall, transfer learning provides a compelling approach for driver drowsiness detection. It overcomes challenges of data collection and training time, leveraging the power of pre-trained models to achieve high accuracy and develop robust driver drowsiness detection systems.

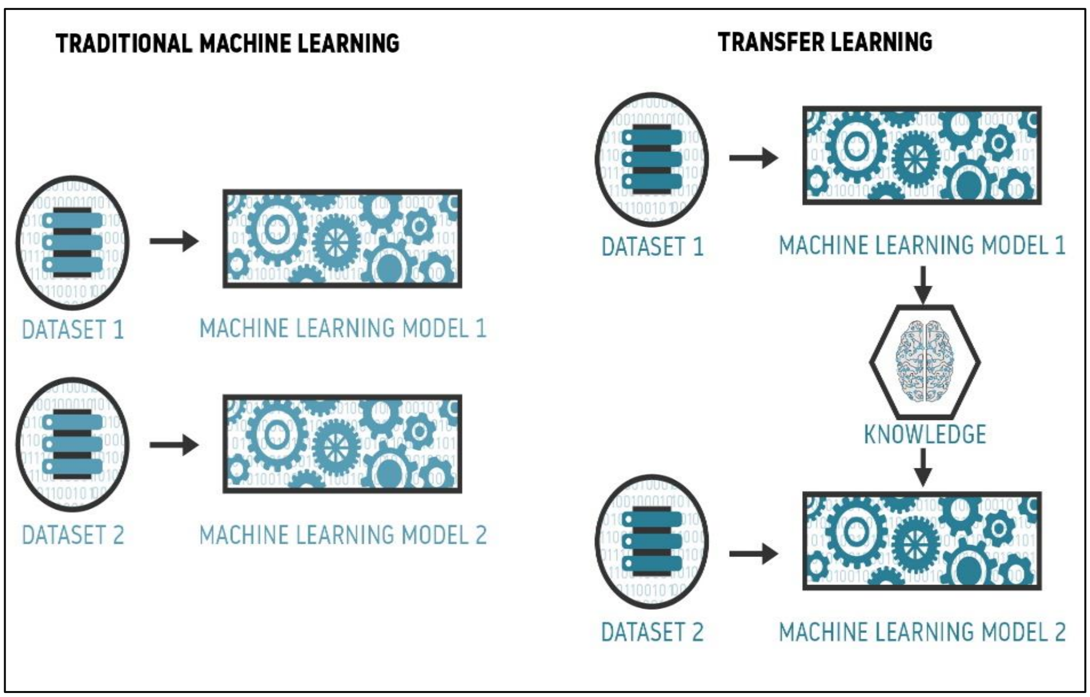


**Figure 4.2**: Transfer Learning

**1. Start Webcam**: User first have to start the webcam of the system.

**2. Continuous Monitor**: The camera will continuously monitor the face of driver.

**3. Capture Frame:** The system will then extract the frames of frontal face.

**4. Extract Features:** Here the Convolutional Neural Network algorithm extract the features of the Eyes.

**5. Match Feature:** This is the step where captured image is mapped with image stored in dataset.

**6. Display Result:** Finally, Result is displayed.

# CHAPTER 5

# IMPLEMENTATION

Here’s a breakdown of implementing driver drowsiness detection system that involves several steps, including capturing an image of the face, preprocessing the image to remove noise and enhance features, extracting relevant features from the image, and using a machine learning algorithm to classify based on the extracted features. Here's a high-level overview of how this process could be implemented:

**Step 1:** Capture Image

One common way to capture an image in driver drowsiness detection is to use a camera, such as a webcam or a smartphone camera. In driver drowsiness detection systems, image capture is a crucial first step for gathering information about the driver's state.

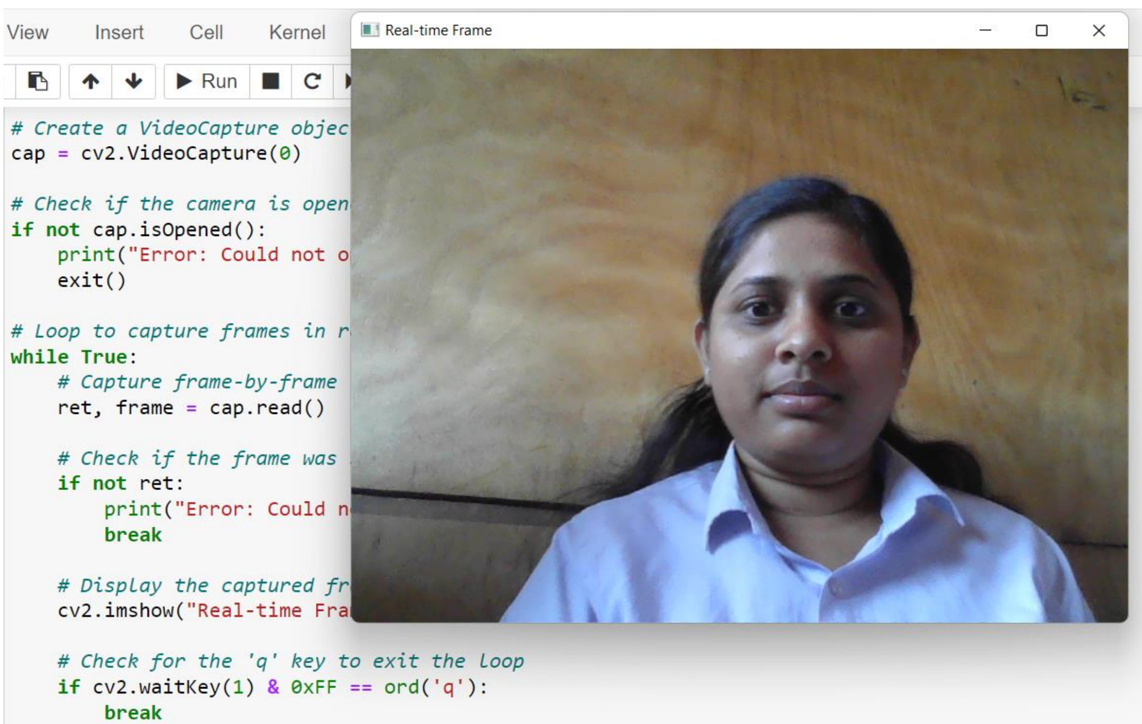


**Figure 5.1:** System Captured Image

**Step 2:** Continuous Monitor

The camera is typically positioned within the car, strategically mounted on the dashboard or A-pillar (upright pillar beside the windshield) to capture the driver's face from a frontal view. This angle allows for optimal visibility of the eyes, mouth, and head pose. The camera continuously captures frames (individual images) at a set frame rate. This creates a stream of images that provide real-time information about the driver's state.

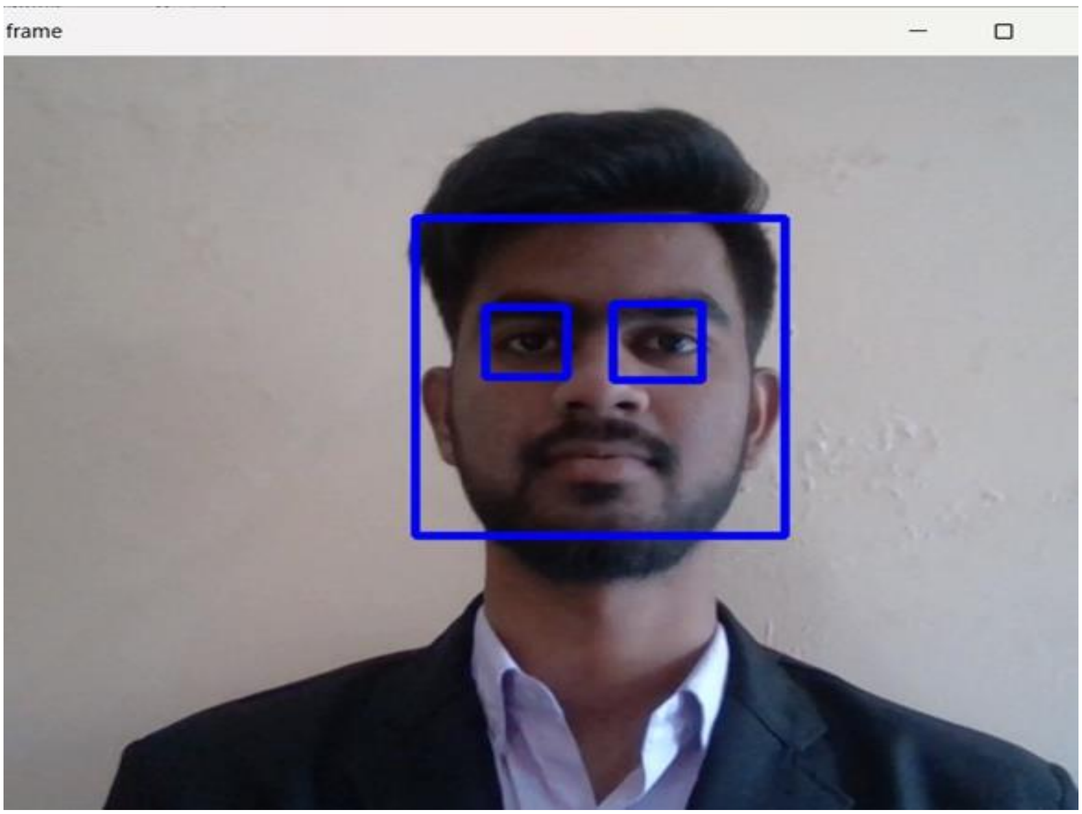


**Figure 5.2:** Continuous Monitor

**Step 3:** Extract Features

By effectively extracting relevant features from captured images, driver drowsiness detection systems can transform complex visual data into a format suitable for further analysis by machine learning algorithms or rule-based systems for drowsiness classification.

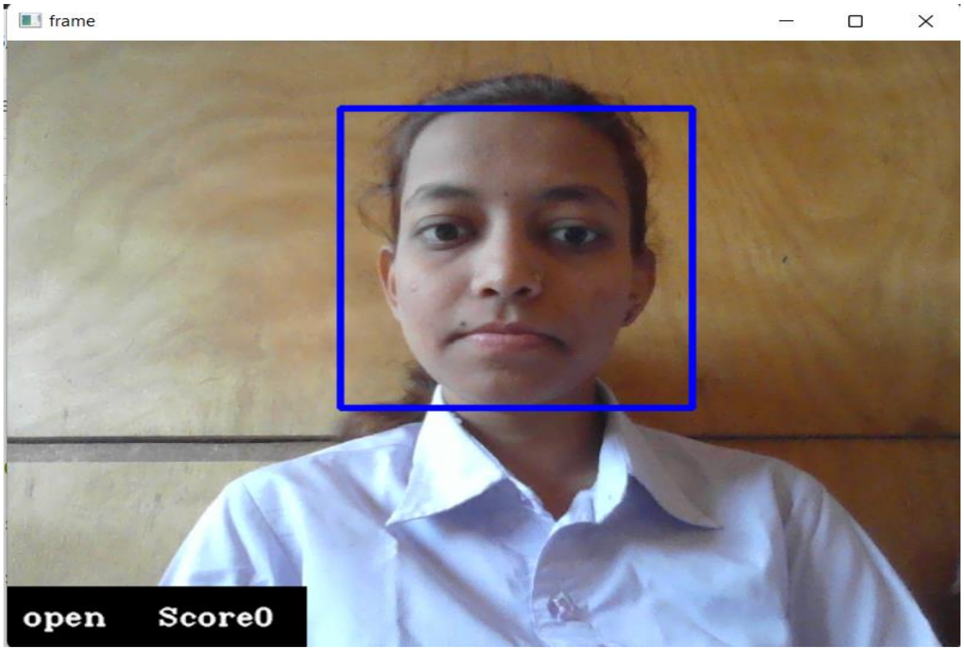


**Figure 5.3:** System Extract Open Eye Features

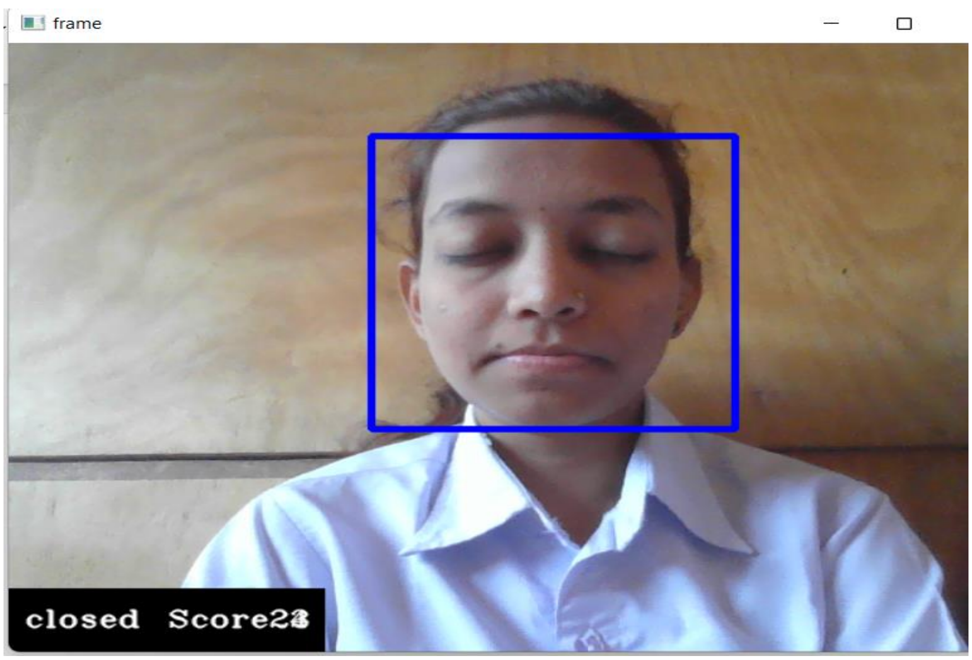


**Figure 5.4:** System Extract Closed Eye Features

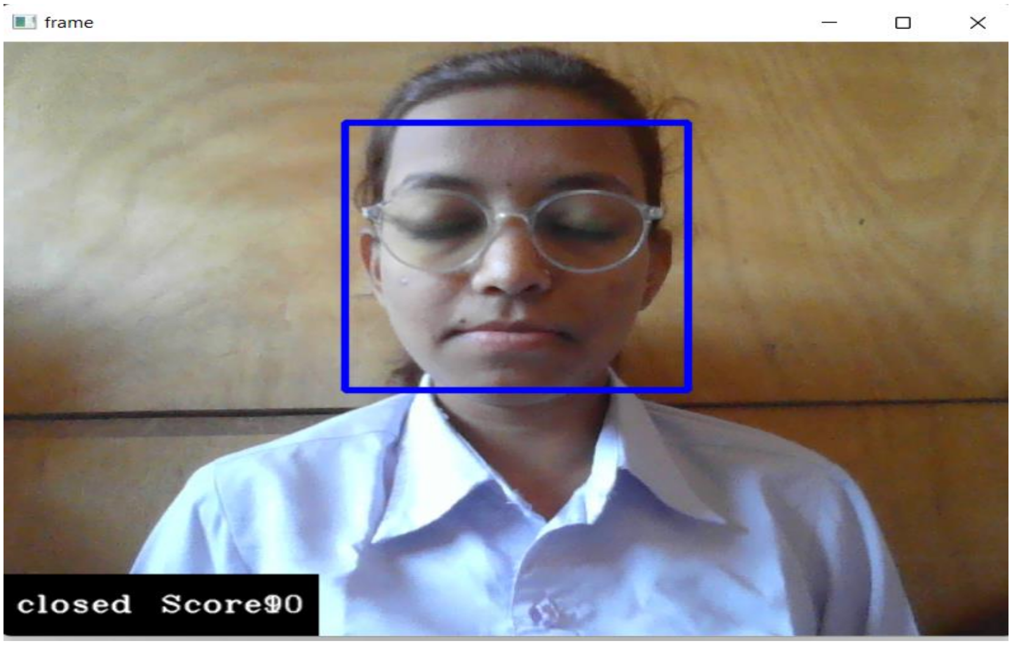


**Figure 5.5**: System Extract Features with Specs.

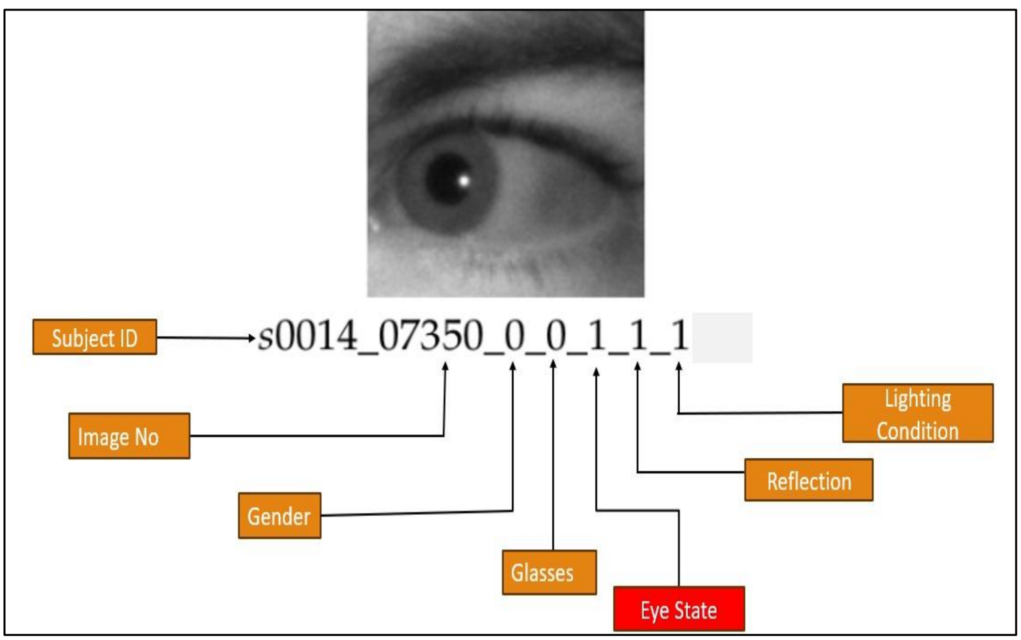


**Figure 5.6**: Image Labelling Annotations in Dataset

Once trained, the model can receive new features extracted from the live camera feed and classify the driver's state (drowsy or alert) in real-time. Machine learning models can adapt to variations in facial features or lighting conditions that might affect traditional rule-based systems. The model can reveal which features are most significant for drowsiness classification, aiding in future feature selection and system improvements. It's worth noting that this is a high-level overview, and each step could involve many different techniques and algorithms. The model's performance depends on the quality and diversity of the training data. Biases in the data can lead to inaccurate drowsiness detection in certain demographics.

# CHAPTER 6

# TESTING

A driver drowsiness detection system is a safety system designed to monitor the alertness level of a driver and issue warnings or take action if the driver appears to be drowsy. These systems typically use visual inputs such as a camera to monitor the driver's face and eyes for signs of drowsiness, including the frequency and duration of eye closure. Testing the system for both open and closed eyes involve verifying its ability to accurately detect the state of the driver's eyes and respond appropriately.

Here is how the theory of testing the project for both open and closed eyes can be elaborated:

**1. Objective:**

- **Detection Accuracy**: Ensure the system accurately detects whether the driver’s eyes are open or closed.
- **Response Time**: Verify that the system responds within an appropriate time frame to changes in the driver's eye state.

**2. Data Collection**:

- **Sample Data**: Gather a diverse set of images or video recordings of drivers with various eye states (open, closed, partially closed) and from different angles and lighting conditions.
- **Ground Truth**: Label each data point (e.g., frame) with the correct state of the driver’s eyes.

**3. Scenario Testing**:

- **Real-world Scenarios**: Test the system in various driving conditions (day/night, different weather, road types) to observe its effectiveness in practical scenarios.
- **Drowsiness Simulation**: Simulate driver drowsiness using test subjects or controlled scenarios to validate the system’s ability to detect drowsiness.

**4. Edge Cases and Variability**:

- **Edge Cases**: Test the system's performance under edge cases such as sunglasses, reflections, or obscured vision.
- **Variability**: Account for different eye shapes, sizes, and facial features among individuals.

**5. Consistency and Robustness**:

- **Consistency**: Ensure the system provides consistent results across different environments and drivers.
- **Robustness**: Evaluate the system's ability to handle changes in lighting, facial expressions, or head movement.

**6. Feedback and Improvement**:

- **User Feedback**: Gather feedback from drivers and stakeholders to assess the system's impact and user experience.
- **Iterative Improvement**: Continuously improve the system based on testing results and feedback.

**7. Compliance and Standards**:

- **Regulatory Compliance**: Ensure the system meets industry standards and regulations related to safety and privacy.
- **Ethical Considerations**: Consider ethical implications, such as the use of personal data and the potential impact on driver behavior.

By following these steps, the drowsiness detection system can be thoroughly tested for its ability to accurately detect open and closed eyes and respond effectively to potential driver drowsiness.

### 6.1 Eye State Testing:

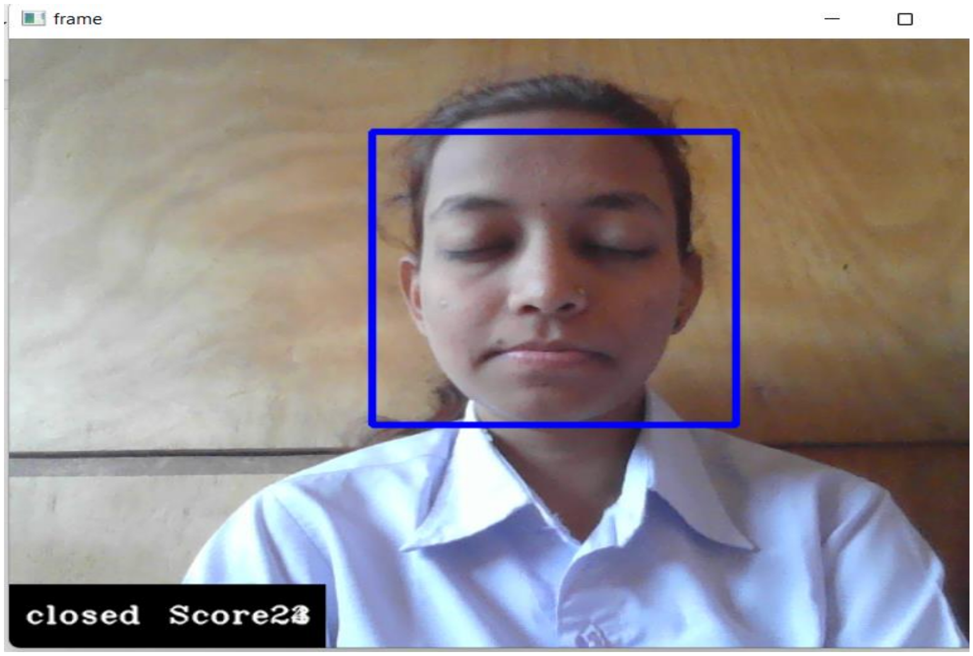


**Figure 6.1:** Closed Eye State Testing

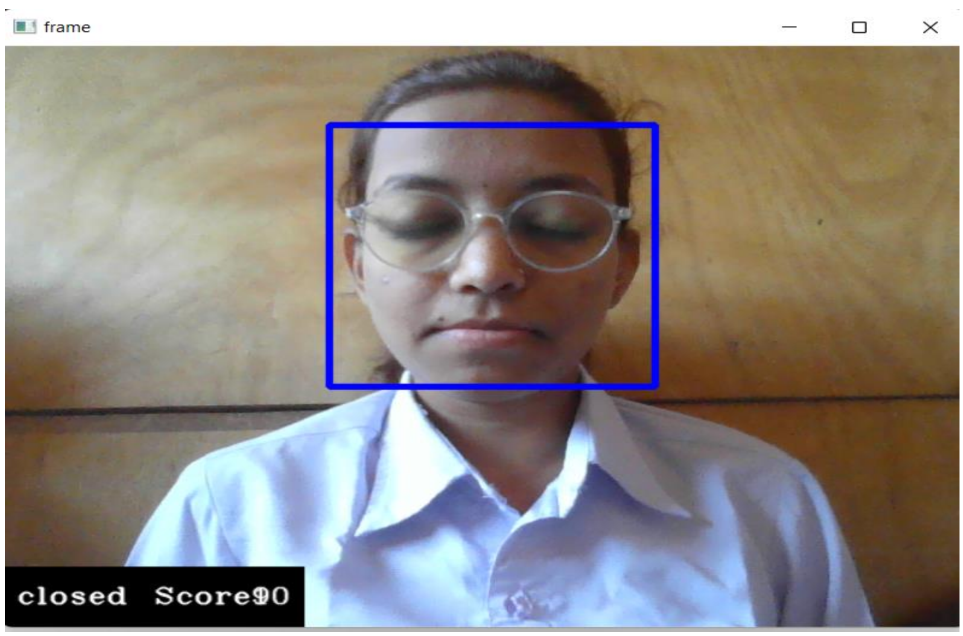


**Figure 6.2:** Closed Eye Testing with Specs

The system employs image processing and machine learning techniques to analyse video footage of the driver's face, specifically focusing on the eyes. When the driver's eyes are closed for an extended period or with high frequency, it indicates a potential state of drowsiness.

The core components of the system include:

- **Video Capture:** A camera is installed within the vehicle, positioned to capture clear video footage of the driver's face, including the eyes.
- **Image Processing:** The video frames are processed to detect and track the driver's eyes. This involves using facial recognition algorithms and eye detection methods to isolate the eye region.
- **Eye Closure Detection:** Once the eyes are detected, the system measures the duration and frequency of eye closures. If the eyes remain closed for a threshold time period (5 seconds), it is indicative of drowsiness.
- **Alarm Activation:** If drowsiness is detected, the Guardian Alarm is triggered, providing an audible warning to alert the driver.

By continuously monitoring the driver's eyes and providing timely alerts, this system helps prevent accidents caused by drowsiness, thus enhancing road safety.

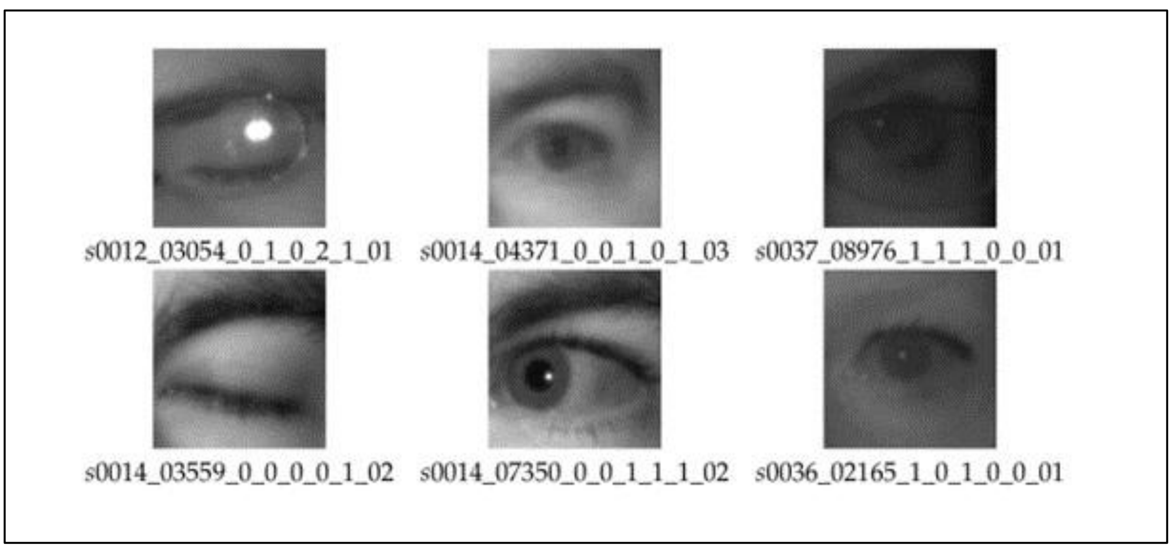


**Figure 6.3**: Different Eye States for Detection

# CHAPTER 7

# SOFTWARE & HARDWARE REQUIREMENTS

## 7.1 Hardware Required:

- Computer System

## 7.2 Software Required:

- Windows 10
- IDE: Jupyter Notebook or Visual Studio
- Libraries: NumPy, Pygame, OpenCV, TensorFlow

# CHAPTER 8

# RESULT

The final result of a project for a driver drowsiness detection system involves assessing the system's overall performance and effectiveness in detecting drowsiness based on the state of the driver's eyes (open and closed). This includes measuring the system's accuracy, reliability, and safety in various conditions. Here's an overview of the explanation of the final result:

**1. Detection Accuracy:**

- **Overall Accuracy**: Evaluate how well the system differentiates between open and closed eyes.
- **Precision and Recall**: Measure how precise the system is in detecting open and closed eyes and how well it captures all relevant instances.

**2. Response Time**:

- **Detection Speed**: Measure the system's speed in detecting changes in eye state and issuing appropriate alerts.
- **Latency**: Check the system's response time from detecting drowsiness to triggering an alert or action.

**3. Consistency:**

- **Stable Performance**: Assess whether the system consistently produces accurate results across various scenarios (e.g., different lighting, times of day, road types).
- **Test-Retest Reliability**: Ensure that the system provides consistent results upon repeated testing under the same conditions.

**4. Edge Cases Handling:**

- **Handling Variability**: Observe how well the system handles different facial features, eye shapes, and expressions.
- **Obstacles and Interference**: Assess the system's ability to work effectively despite obstacles such as sunglasses, hats, or head movements.

**5. User Experience:**

- **Ease of Use**: Determine whether the system is user-friendly for drivers and easy to integrate into vehicles.
- **Alert Quality**: Evaluate the quality and clarity of the system's alerts and their effectiveness in maintaining driver alertness.

**6. Compliance with Standards:**

- **Regulatory Compliance**: Verify that the system complies with industry standards and regulations for safety and privacy.
- **Ethical Standards**: Confirm that the system respects ethical standards regarding the use of biometric data and driver privacy.

**7. Impact and Recommendations:**

- **Effectiveness**: Determine the system's impact on reducing accidents and improving driver safety.
- **Feedback and Recommendations**: Gather feedback from drivers and industry experts to guide future improvements.

**8. Conclusions:**

- **Summary of Findings**: Summarize the system's performance, strengths, and areas for improvement based on testing and evaluation.
- **Final Thoughts**: Provide an overall assessment of the project's success and potential for future development.

By considering these factors, the final result of the driver drowsiness detection system should be a robust, reliable, and user-friendly solution that enhances road safety by effectively identifying and mitigating the risks associated with driver drowsiness.

## 8.1 Result if Eye State Open:

System is doing continuous eye tracking so, eye state might be found like open, closed or partially closed or open, if eye state found open then system continuously display Open Eyes (0) as output on screen to show eye state.

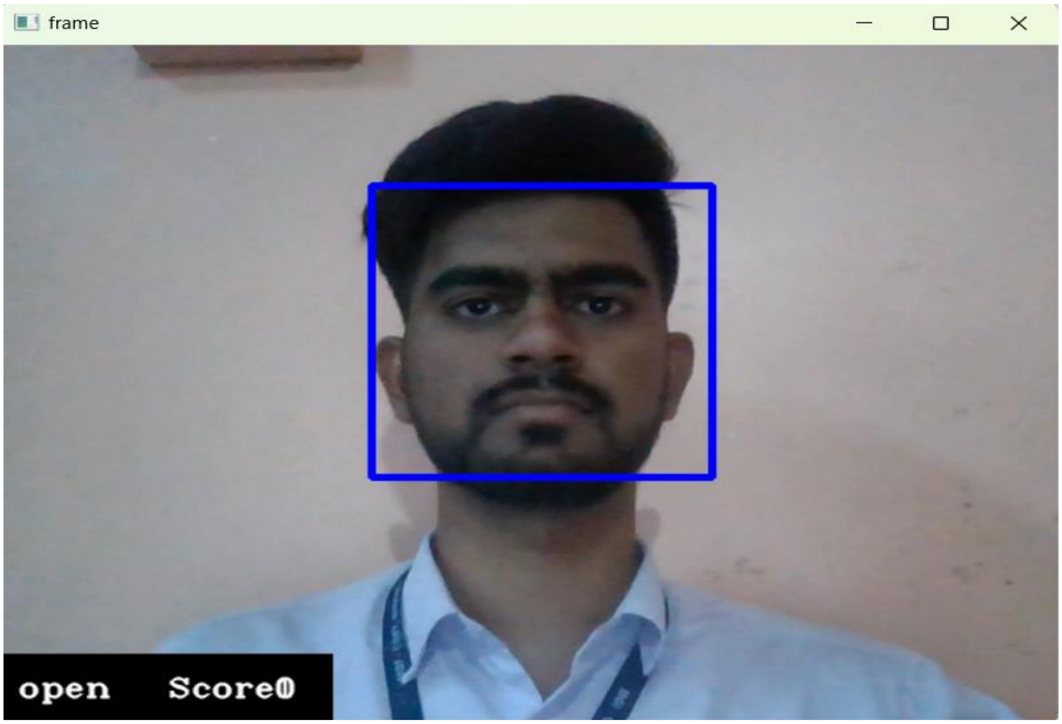


**Figure 8.1**: Open Eye State

## 8.2 Result if Eye State Closed:

Once system detect as eye state is fully closed then it will find score, if the closing time span of eyes is greater than 5 seconds or more then the alarm blow automatically to alerting the driver and once the eyes found opened again then the alarm will stop automatically

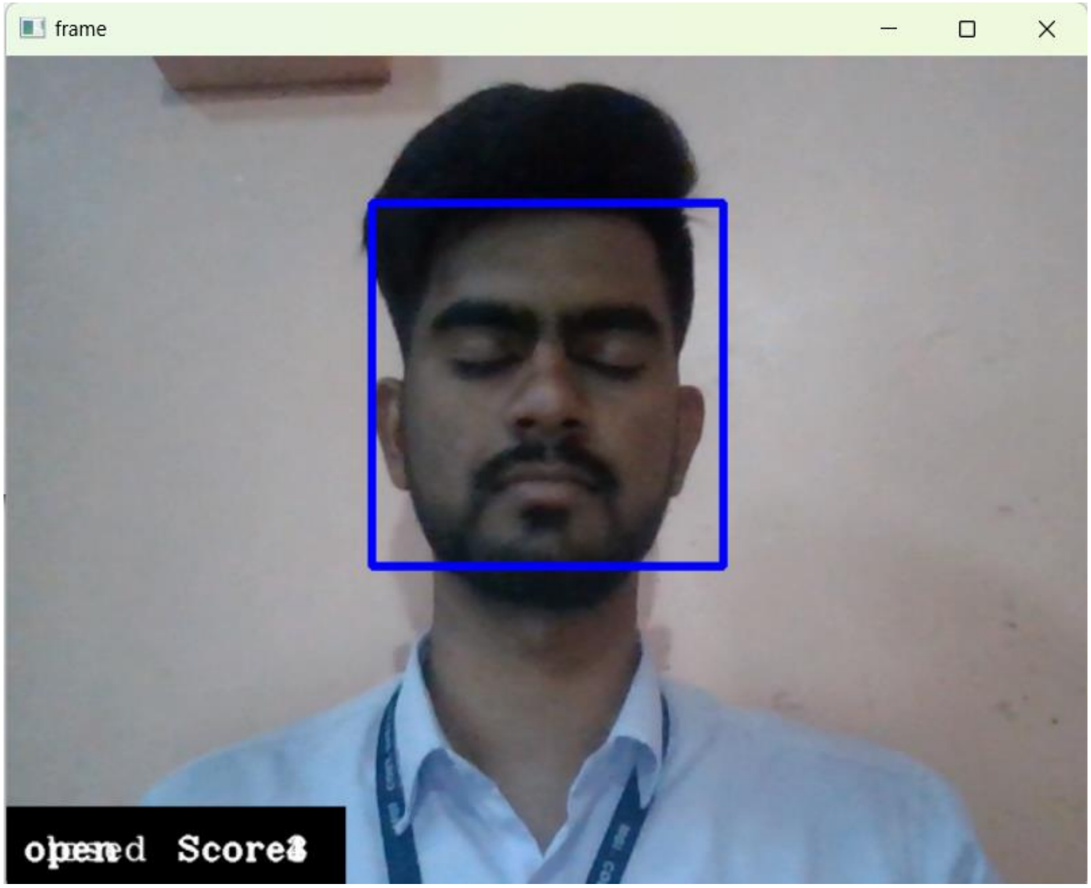


**Figure 8.2:** Closed Eye State

```
if prediction[0][0]>0.50:
    cv2.putText(frame,'closed',(10,height-20),fontFace=cv2.FONT_HERSHEY_COMPLEX_SMALL,fontScale=1,color=(255,255,255),
               thickness=1,lineType=cv2.LINE_AA)
    cv2.putText(frame,'Score'+str(Score),(100,height-20),fontFace=cv2.FONT_HERSHEY_COMPLEX_SMALL,fontScale=1,color=(255
               thickness=1,lineType=cv2.LINE_AA)
    Score=Score+1
    if(Score>5):                          #we can adjust this score depending on person to person
        try:
            sound.play()
        except:
            pass
```

**Figure 8.3:** Guardian Alarm

### 8.3 Result if Eyes are Partially Closed

Once system detect as eye state is partially closed then it will find score, if the partially closing time span of eyes is greater than 5 seconds or more then the alarm blow automatically to alerting the driver and once the eyes found fully opened again then the alarm will stop automatically.

Because it might the condition that driver try to take the rest due to fatigue so sometimes, they partially close their eyes while driving so their will be the chances of happening an accident so the system is also able to alert the driver in that condition.

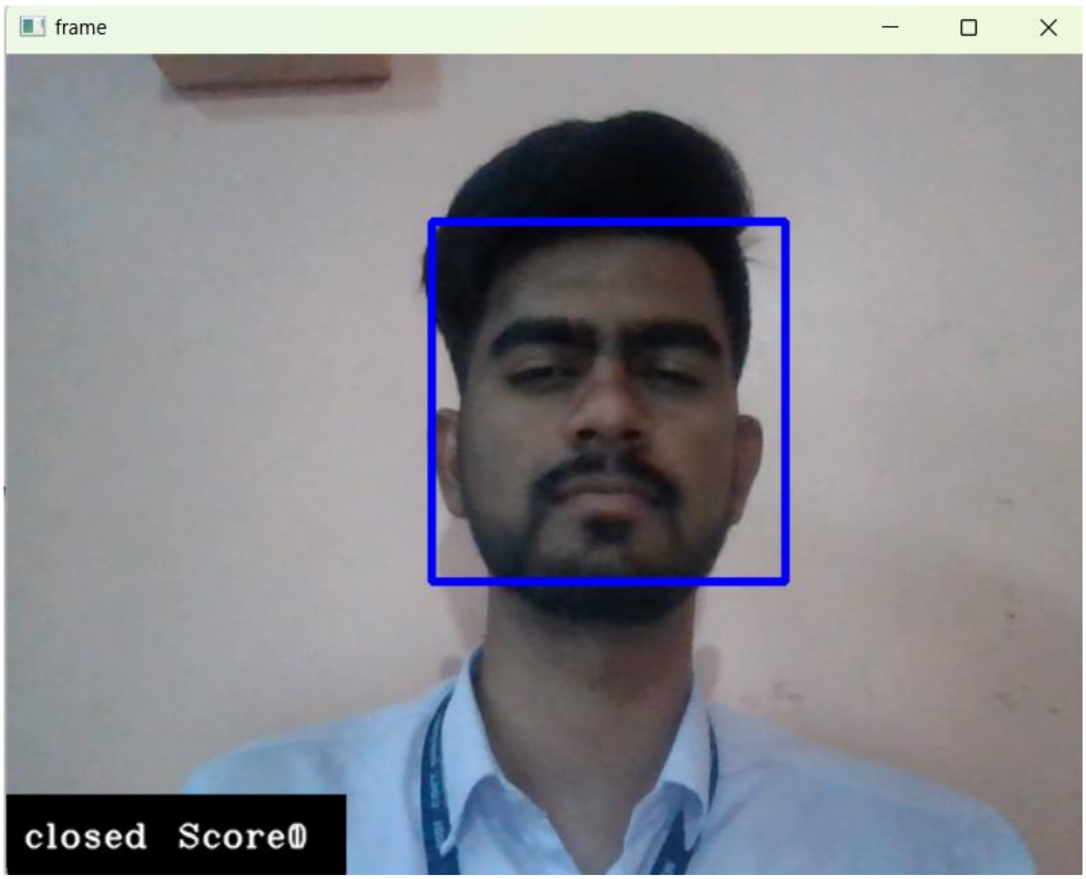


**Figure 8.3:** Partially closed Eye State

# CHAPTER 9

# CONCLUSION AND FUTURE SCOPE

This project has successfully demonstrated a prototype driver drowsiness detection system utilizing an ESP32 CAM for image capture, servos for pan-tilt camera control, and an Arduino Uno for basic processing. The system monitors eye closure using an eye blink sensor and provides a visual alert via an LED indicator.

- Developed a functional proof-of-concept for driver drowsiness detection.
- Demonstrated the feasibility of using an ESP32 CAM for continuous eye tracking.
- Implemented pan-tilt functionality to maintain the driver's face within the camera's view.

**Future Scope:**

The future scope of driver drowsiness detection using deep learning is vast, and there are many potential directions for further research and development.

1. **Understanding Driver Safety:** The seminar will delve into the critical aspect of driver safety by focusing on the detection and prevention of driver drowsiness, a leading cause of accidents on the road.

2. **Emerging Technology:** Attendees will gain insights into cutting-edge technology, including deep learning, which plays a pivotal role in real-time drowsiness detection.

3. **Cross-Industry Relevance:** The knowledge acquired in this seminar can be applied across various industries, from automotive to transportation logistics, ensuring widespread applicability.

4. **Road Safety Enhancement:** The primary goal is to contribute to improved road safety by preventing accidents caused by drowsy drivers, thereby saving live

5. **Data-Driven Approach:** Attendees will learn how data collection and analysis form the backbone of effective drowsiness detection systems, with a focus on real-world scenarios.

6. **Algorithmic Advancements:** The seminar explores the latest advancements in deep learning algorithms for better accuracy and robustness in detecting driver drowsiness.

7. **Algorithmic Advancements:** The seminar explores the latest advancements in deep learning algorithms for better accuracy and robustness in detecting driver drowsiness.

8. **Practical Implementation:** Practical guidance on deploying deep learning models in vehicles, highlighting the real-time nature of the application.

9. **Human-Machine Interaction:** Understanding the synergy between humans and intelligent systems is crucial, as it impacts both the driver and the vehicle.

10. **Cost-Effective Solutions:** Emphasis will be placed on solutions that are cost-effective and can be integrated into existing vehicle systems without major infrastructural changes.

11. **Research Opportunities:** The seminar will identify potential research areas and open questions in the field of driver drowsiness detection, encouraging further exploration.

# REFERENCES

[1] Magán, E.; Sesmero, M.P.; Alonso-Weber, J.M.; Sanchis, A. Driver Drowsiness Detection by Applying Deep Learning Techniques to Sequences of Images. Appl. Sci. 2022.

[2] Md. Tanvir Ahammed Dipu1, Syeda Sumbul Hossain2, Yeasir Arafat3, Fatama Binta Rafiq4 “Real-time Driver Drowsiness Detection using Deep Learning”.

[3] Yaman Albadawi, Mohammed Awad “Real-Time Machine Learning-Based Driver Drowsiness Detection Using Visual Features” April 2023.

[4] Oquab, M., Bottou, L., Laptev, I., Sivic, J.: Learning and Transferring Mid-Level Image Representations using Convolutional Neural Networks. IEEE Conf. Comput. Vis. Pattern Recognit. 1717–1724 (2014).

[5] Said, S., AlKork, S., Beyrouthy, T., Hassan, M., Abdellatif, O., Abdraboo, M.F. (2018). Realtime eye-tracking and detection-a driving assistance system. Advances in Science, Technology and Engineering Systems Journal,3(6):446-454

[6] Jabbar, R., Al-Khalifa, K., Kharbeche, M., Alhajyaseen, W., Jafari, M., Jiang, S. (2018). Real-time driver drowsiness detection for an android application using deep neural networks techniques. Procedia Computer Science,130:400-407