DRIVER DROWSINESS DETECTION USING OPENCV AND KERAS

SUBMITTED BY-

- TEAM 148 -

SHREASI SEN	*20BCE2738*
SAMRIDHI HISARIA	*20BCE0792*
SRILEKHA BHATTACHARJEE	*20BCE2727*
RAUNIT GARG	*20BCE2089*

1. INTRODUCTION

1.1 Overview

In order to preserve their safety, the car security system can recognise the driver's face and provide notifications anytime the driver becomes tired and blinks his eyelids.

These days, one of the key causes or contributing factors to traffic accidents is sleepiness. Here, a sleepiness alert system and vehicle safety system are being developed in order to prevent similar incidents. We're developing this system utilizing Artificial Intelligence (AI) technologies. Initially, the driver's image is taken and recognised by face recognition software. Once the driver is inside the car and has started driving, an alert or alarm will sound to let him know if he feels sleepy so he can get up, take a break, and then resume driving.

1.2 Purpose

One of the major causes of road accidents lately has been due to drowsiness, which may be caused by various reasons such as intake of alcohol, stress, exhaustion etc. This project aims to provide a solution wherein real-time sleep detection is done using facial features mainly closing of the eyelids, if more than 1.6 seconds an alert message along with an alarm noise is initiated to warn the driver to stay awake or pullover and take medical assistance. The data is logged into servers for doctors to examine as well. The model is obtained with Pretrained Shape Predictor Dlib Model -shape_predictor and Python imutils dlib cv2 numpy. The model can be used in medical studies to predict when critical impairment will occur. TheModel accuracy will be greater with behavioral (eye and head) indicators. Driving time data also can be used further to study more reasons and improve accuracy to reduce road accidents. This model can later also be proposed to airlines to make sure that the pilot stays alert during the flight.

2. LITERATURE SURVEY

2.1 Existing problem

There are several existing approaches and methods to solve the driver drowsiness problem. Here are a few commonly used ones:

Eye Monitoring: This approach focuses on tracking the driver's eye movements and detecting signs of drowsiness or fatigue. Techniques such as eye-tracking cameras or infrared sensors can be used to monitor the driver's eye closure, blink rate, and gaze patterns. If the system detects signs of drowsiness, it can alert the driver through visual or auditory cues.

Facial Analysis: Facial analysis techniques involve using cameras or sensors to analyze the driver's facial expressions and detect signs of drowsiness. This can include monitoring for drooping eyelids, yawning, or changes in facial muscle activity. Machine learning algorithms can be trained to recognize these patterns and trigger alerts when drowsiness is detected.

EEG-based Systems: Electroencephalography (EEG) measures the electrical activity of the brain and can be used to detect drowsiness. EEG sensors placed on the driver's scalp can monitor brainwave patterns associated with drowsiness or fatigue. Machine learning algorithms can analyze these patterns in real-time and issue warnings when necessary.

Steering Behavior Analysis: This approach involves monitoring the driver's steering behavior to detect signs of drowsiness. Changes in steering patterns, such as drifting or erratic steering, can indicate drowsiness. Sensors in the steering wheel or vehicle can track these movements and trigger alerts if drowsiness is suspected.

Physiological Monitoring: Physiological monitoring systems can measure various bodily parameters, such as heart rate, respiration rate, and skin conductance, to detect signs of drowsiness. Changes in these parameters can indicate increased fatigue levels. Machine learning algorithms can analyze these physiological signals and provide timely warnings.

Machine Learning Models: Machine learning algorithms can be trained to recognize patterns of drowsiness using data from various sensors and monitoring systems. These models can learn to identify specific features or combinations of features associated with drowsiness and issue alerts accordingly.

It's worth noting that many modern driver drowsiness detection systems combine multiple approaches for increased accuracy and reliability. These systems are typically integrated into vehicles or wearable devices to provide real-time monitoring and alerts to ensure driver safety.

2.2 Proposed solution

The method proposed by us is using opencv and keras which requires the following steps:

Data Collection: Gather a dataset of images or video footage that contains both drowsy and alert driver samples. The dataset should include images or frames where the driver's eyes are open and closed, as well as various lighting conditions and angles.

Preprocessing: Preprocess the collected data to enhance the features necessary for drowsiness detection. Common preprocessing steps include resizing the images, converting to grayscale, and normalizing pixel values.

Eye Detection and Tracking: Use OpenCV's Haar cascades or deep learning-based object detection models (such as Dlib) to detect and track the driver's eyes within each frame. Extract the region of interest (ROI) containing the eyes.

Feature Extraction: Extract meaningful features from the eye ROIs to represent their current state. This can include measurements such as eye aspect ratio (EAR), which is calculated based on the ratio of eye landmarks, or other relevant features that capture eye openness and blinks.

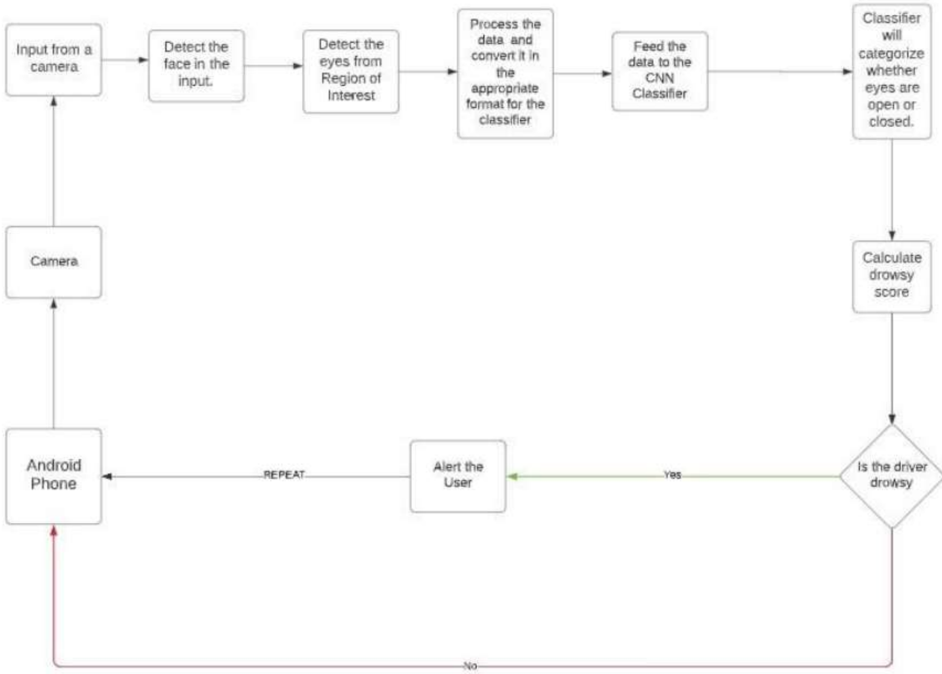
Training a Drowsiness Classification Model: Use Keras or another deep learning framework to build a classification model. Train the model using the preprocessed data and corresponding labels indicating drowsy or alert states. You can use a convolutional neural network (CNN) architecture to learn discriminative features from the eye ROIs and classify them as drowsy or alert.

Model Evaluation: Evaluate the trained model's performance using separate validation and test datasets. Measure metrics such as accuracy, precision, recall, and F1-score to assess the model's effectiveness in detecting drowsiness.

Real-time Drowsiness Detection: Apply the trained model to new video frames or live camera feed. Use the eye detection and tracking techniques to extract the eye ROIs, preprocess them, and feed them to the trained model for drowsiness classification. If the model predicts drowsiness, issue an alert or take appropriate action to ensure the driver's safety.

3. THEORETICAL ANALYSIS

3.1 Block diagram



3.2 Hardware / Software designing

Designing a driver drowsiness detection project involves integrating both hardware and software components. OpenCV and Keras are popular libraries used for computer vision and deep learning, respectively.

Here's a high-level overview of how you can approach this project:

Hardware Components:

Camera: Use a camera to capture the driver's face and eye movements. You can either use a USB webcam or integrate a camera module, such as Raspberry Pi Camera, into your hardware setup.

Software Components:

Image Processing with OpenCV: Utilize OpenCV to perform real-time image processing on the captured video frames. You can apply various techniques, such as face detection and eye tracking, to extract relevant features from the driver's face.

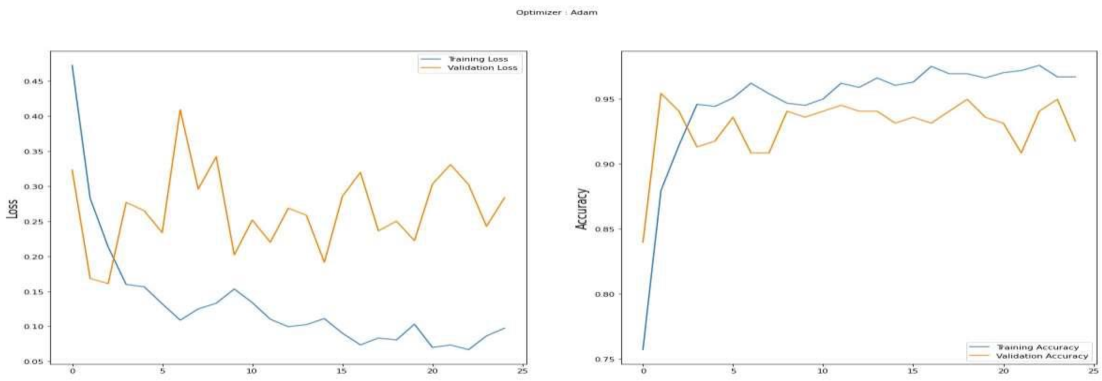
Feature Extraction: Use OpenCV to extract relevant features from the facial region, such as eye aspect ratio (EAR), to determine the level of drowsiness.

Training a Drowsiness Detection Model: Use Keras (with TensorFlow backend) to train a deep learning model for drowsiness detection. Prepare a dataset containing labeled images indicating drowsy and non-drowsy states. You can use the extracted features (e.g., EAR) as inputs to the model.

Model Integration: Integrate the trained model with the real-time video feed from the camera. Apply the model to the extracted features in each frame to predict the driver's drowsiness level.

Alert Mechanism: Based on the drowsiness prediction, implement an alert mechanism to warn the driver.

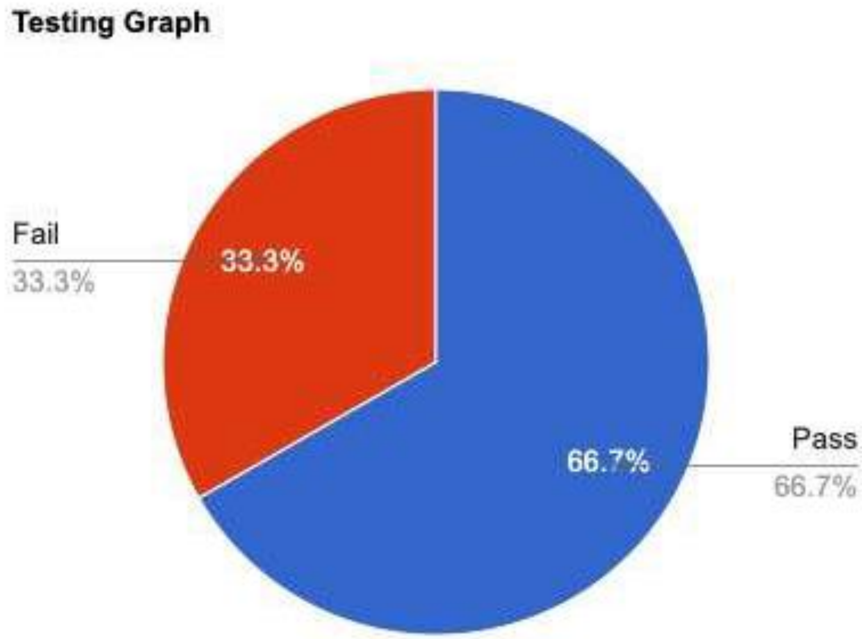
4. EXPERIMENTAL INVESTIGATIONS

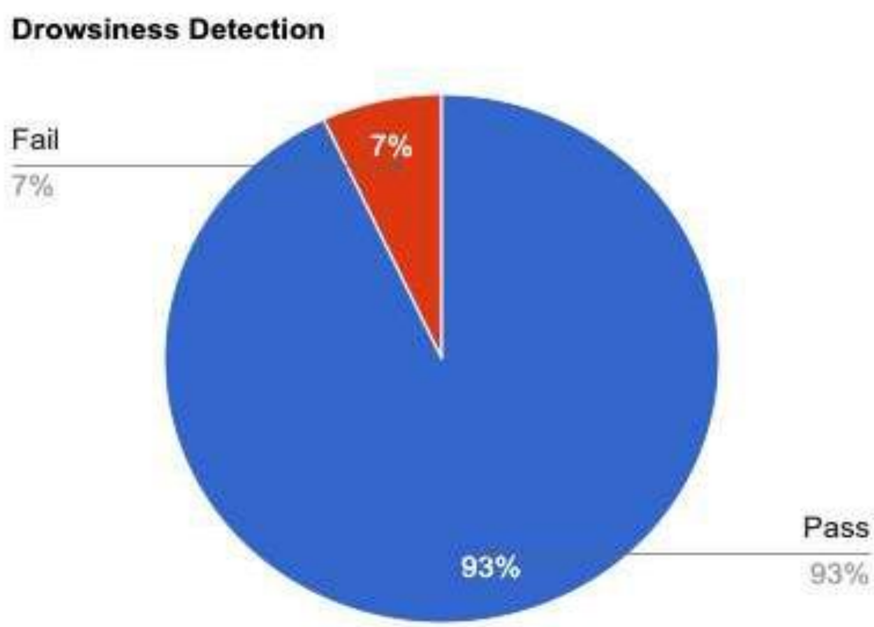


The above graphs show the training and validation accuracy as well as loss function after each epoch. As we can in these graphs the accuracy gradually increases after each epoch and the loss function gradually decreases. We can also see that the training and validation accuracy is very similar at each epoch. This means that the CNN model is neither overfitting nor underfitting.

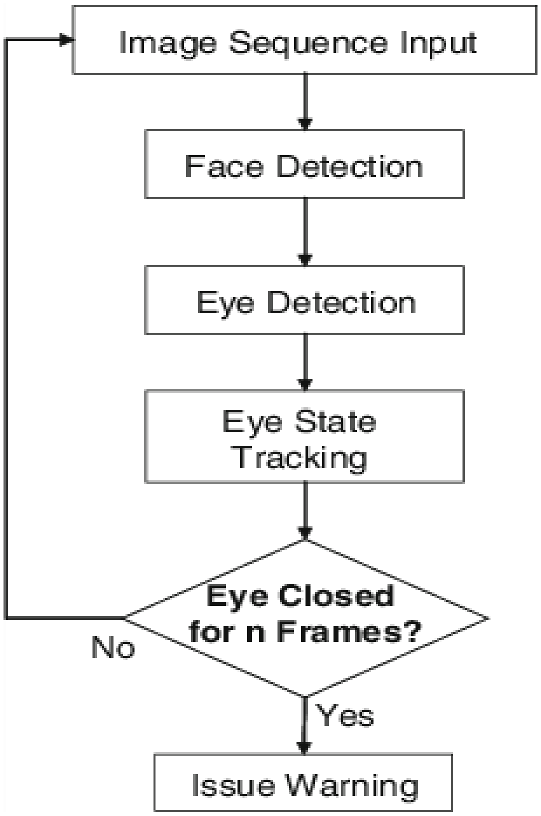
Testing for face detection-

Test CaseID	Activity	Inputs	Excepted Results	Actual Results	Status(pass/ fail)	Comments
TC-01	Video Capturing	Continuous stream of video	Face Detection using rectangular bounds	Face Detection	Pass	None
TC-02	Eye Detection	Continuous stream of video	Detection of eyes	Detection of eyes in a particular frame in a rectangular frame.	Pass	None
TC-03	Eye detection : when the driver is drowsy	Continuous stream of video	Detection of the frame to detect the eyes of the driver	The eyes of the drivers are closing	Pass	None
TC-04	Eye Detection: when the driver is drowsy	Continuous stream of video	Detection of the frame to detect the eyes of the drive	The eyes of the driver are not being detected	Fail	The system is not able to determine the condition of the driver.
TC-05	Alarm system	None	The alarm will alert the driver so that the driver regains alertness.	The alarm system alerts the driver	Pass	None
TC-06	Alarm System	None	The alarm will alert the driver so that the driver regains alertness.	The alarm system fails to alert the driver.	Fail	The alarm system failed to alert the driver.

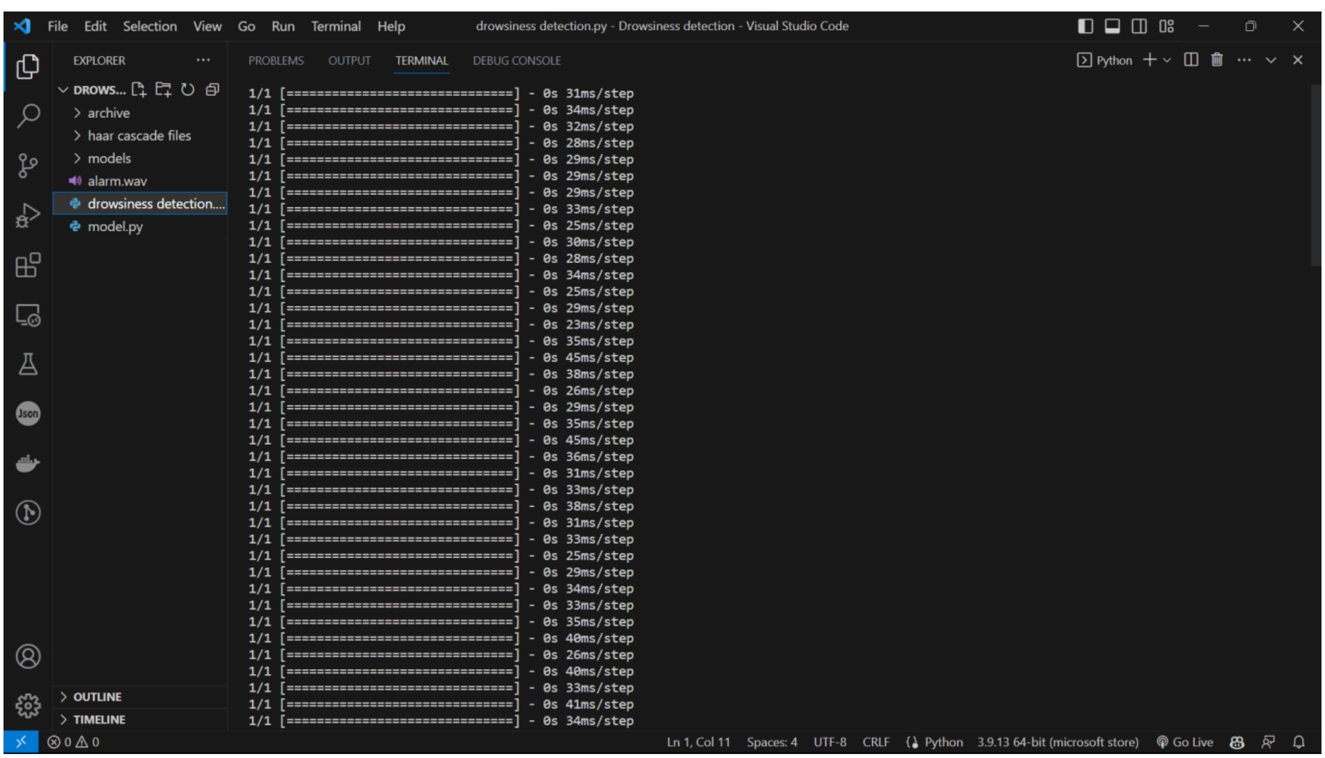

Testing Graph
Fail
33.3%
33.3%
Pass
66.7%
66.7%

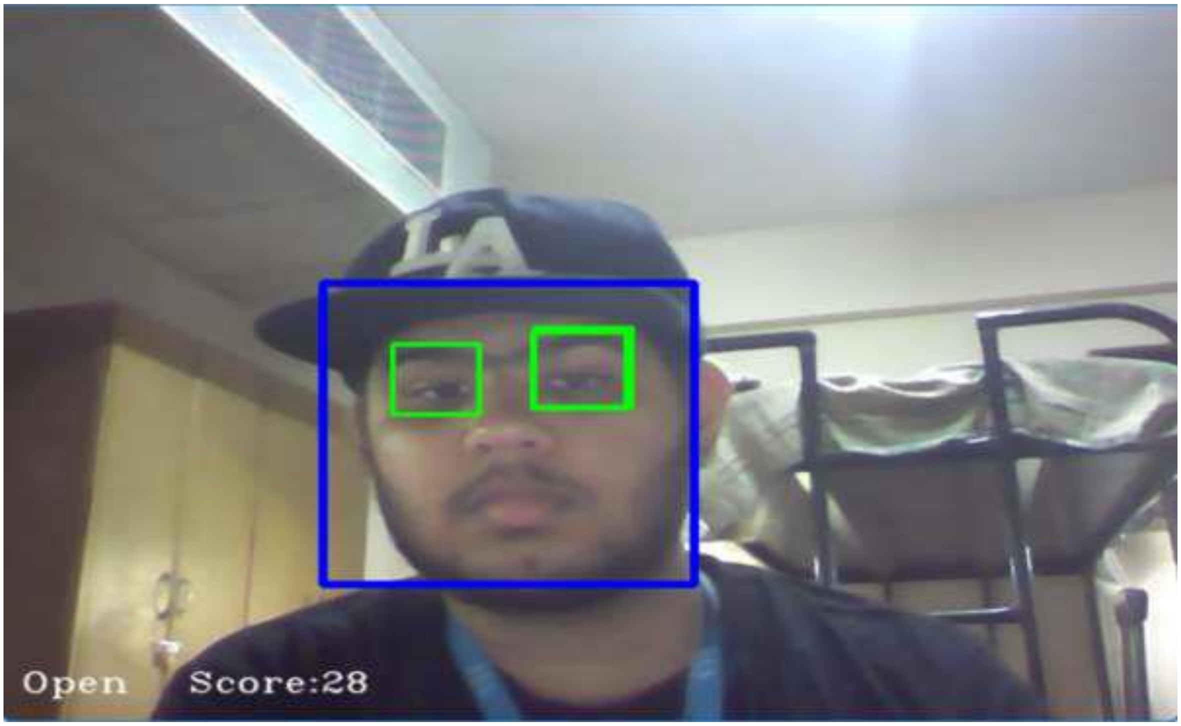

Drowsiness Detection
Fail
7%
7%
Pass
93%
93%

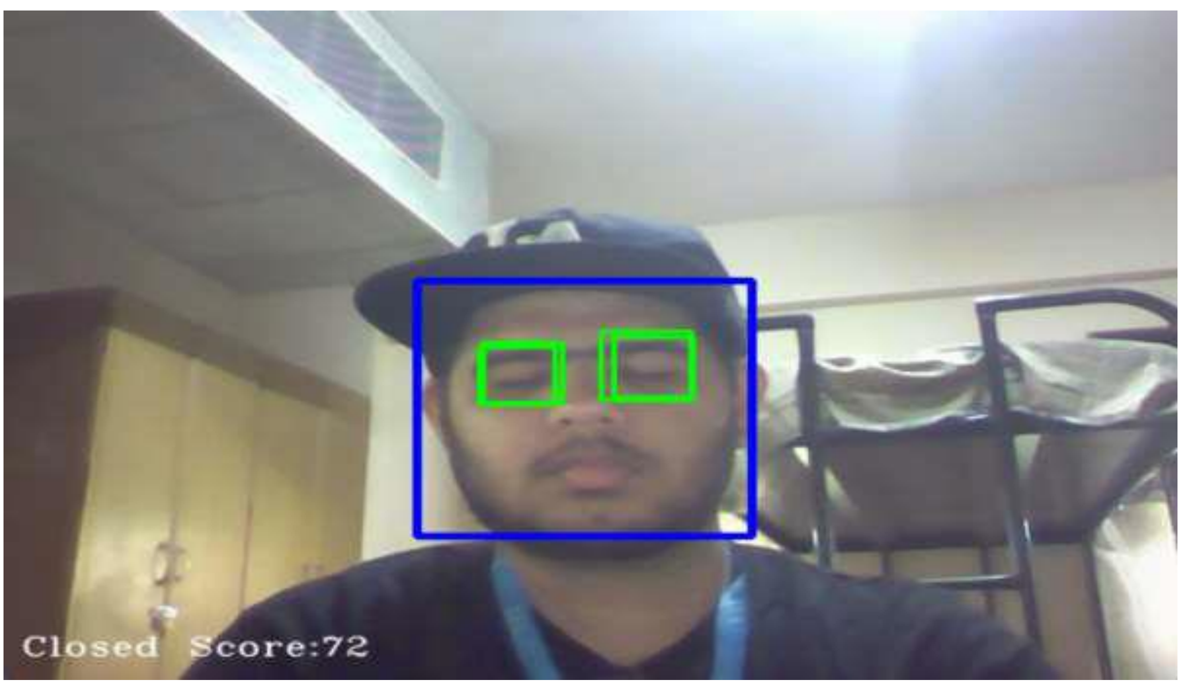
5. FLOWCHART



6. RESULT




Open Score:28


Closed Score:72

7. ADVANTAGES & DISADVANTAGES

Advantages of Drowsiness Detection:

Enhanced Safety: Drowsiness detection systems can significantly improve safety, particularly in transportation contexts like driving or operating heavy machinery. By detecting drowsiness in real-time, these systems can alert the individual or trigger automated safety measures, reducing the risk of accidents caused by drowsy driving.

Early Warning: Drowsiness detection systems can provide early warning signs of fatigue and drowsiness, allowing individuals to take preventive actions such as taking breaks, resting, or seeking assistance before the situation worsens. This can help mitigate the negative effects of fatigue and reduce the likelihood of accidents or errors.

Objective Assessment: Drowsiness detection systems provide an objective assessment of drowsiness levels, removing subjectivity and personal biases. This can be particularly useful in professions that involve safety-critical tasks, as individuals may not always accurately self-assess their own level of drowsiness.

Continuous Monitoring: These systems can continuously monitor drowsiness levels, providing ongoing feedback and alerts. This feature is especially valuable during long-duration activities or night shifts when the risk of drowsiness and fatigue is higher.

Disadvantages of Drowsiness Detection:

False Positives and False Negatives: Drowsiness detection systems may occasionally generate false positives, incorrectly flagging individuals as drowsy when they are not, or false negatives, failing to detect drowsiness in some individuals. This can lead to unnecessary interruptions or missed detections, respectively, impacting the system's reliability and user trust.

Variability in Drowsiness Symptoms: Drowsiness manifests differently in individuals, and symptoms can vary. Some people may exhibit physical signs of drowsiness, such as drooping eyelids, while others may show more subtle signs. Drowsiness detection systems relying on specific symptoms may not capture the full range of drowsiness indicators, reducing their effectiveness.

Hardware and Implementation Challenges: Drowsiness detection systems often require specialized hardware, such as cameras or sensors, to monitor physiological or behavioral cues indicative of drowsiness. Implementing and maintaining such systems may involve costs, technical challenges, and potential privacy concerns.

User Acceptance and Intrusiveness: Some individuals may find drowsiness detection systems intrusive or uncomfortable, especially if they involve continuous monitoring or intrusive sensors. User acceptance and willingness to adopt such systems can vary, and it may be necessary to address privacy, trust, and user experience concerns for wider adoption.

8. APPLICATIONS

Drowsiness detection systems have various practical applications across different industries. Here are some common applications:

Transportation Safety: One of the primary applications of drowsiness detection is in transportation safety, particularly in the automotive industry. Drowsiness detection systems can be used in cars, trucks, buses, and trains to monitor driver drowsiness and alert them when fatigue levels are high, reducing the risk of accidents caused by drowsy driving.

Workplace Safety: Drowsiness detection systems find applications in industries where fatigue-related accidents can have severe consequences, such as manufacturing, construction, and mining. These systems can monitor employees' drowsiness levels to prevent accidents and provide timely alerts or intervention to ensure workplace safety.

Aviation: Drowsiness detection is crucial in the aviation industry to prevent accidents caused by tired pilots. Drowsiness detection systems can monitor pilots' fatigue levels based on their physiological indicators and alert them or recommend appropriate actions to maintain aviation safety.

Healthcare: In healthcare settings, drowsiness detection systems can help monitor the alertness levels of medical professionals, especially during long shifts or critical procedures. These systems can ensure that healthcare providers are alert and focused, reducing the likelihood of errors due to fatigue.

Elderly Care: Drowsiness detection systems can be used in elderly care facilities to monitor the sleep patterns and alertness levels of residents. This can help ensure the well-being of older adults by identifying potential sleep disorders or fatigue-related issues.

These are just a few examples of the diverse applications of drowsiness detection systems. As the technology advances, we can expect to see further integration of drowsiness detection in various domains to improve safety, performance, and overall well-being.

9. CONCLUSION

In recent years, driver drowsiness has been one of the major causes of road accidents and can lead to severe physical injuries, deaths and significant economic losses. Statistics indicate the need of a reliable driver drowsiness detection system which could alert the driver before a mishap happens. Researchers have attempted to determine driver drowsiness using the following measures: (1) vehicle-based measures; (2) behavioral measures and (3) physiological measures. A detailed review on these measures will provide insight on the present systems, issues associated with them and the enhancements that need to be done to make a robust system. The various ways through which drowsiness has been experimentally manipulated is also discussed. We conclude that by designing a hybrid drowsiness detection system that combines non-intrusive physiological measures with other measures one would accurately determine the drowsiness level of a driver. A number of road accidents might then be avoided if an alert is sent to a driver that is deemed drowsy.

10. FUTURE SCOPE

We have successfully implemented the project. We have made an android application which will help to detect whether the driver is drowsy or not, using OpenCV, Keras and CNN Algorithm. Using CNN algorithm, we can analyze the live video frame by frame and hence be able to determine whether the driver is tentative or not. The application is successfully able to detect whether the eyes if the user is open or closed while driving. The android application can sign up a new user and login an old user. Our future goals for the system will be to alert the user in a non-intrusive way. We can alert the user either by tightening the seat belt or increasing the temperature of the car.

11. BIBLIOGRAPHY

References of previous works or websites visited/books referred for analysis about the project, solution previous findings etc.

https://ieeexplore.ieee.org/abstract/document/8808931

https://ieeexplore.ieee.org/abstract/document/8405495

https://ieeexplore.ieee.org/abstract/document/6053857

https://ieeexplore.ieee.org/abstract/document/8261140

Appendix (Glossary)

It is used to track all the different variables, states and functional requirements that you describe in your document. To include the complete list of constants, state variables, inputs and outputs in a table. In the table, include the description of these items as well as related operations and requirements. a. User: the person who will be using the android application b. Camera: Image processing will be done using the captured video. c. Alarm: The alarm will ring at an interval of 1 minute and the front camera flash will flashlight so as to alert the user. d. Android Application: A non-intrusive monitoring system that will not distract the driver and ensures accuracy in detecting drowsiness.

A. Source Code

https://drive.google.com/drive/folders/108hYFNgl7Yw11XjRQH-n__zNaHiQpqKp?usp=drive_link