PROJECT 4 - Individual Contribution Report
Drowsiness Detection in Drivers

Manjju Shree Devy Gendeti
mgendeti@asu.edu
1225464968

Topic: Monitoring and Analysing Driver and Driving conditions through video/sensory inputs

My TASK:

Detecting Drowsiness in Drivers through Real-time Video Analysis. The primary objective of this task is to develop a robust system capable of detecting and preventing drowsy driving by analyzing real-time video footage obtained from camera sensors focused on the driver's face.

Alignment with Guardian Angel:

The Guardian Angel initiative in Project 5 focuses on using advanced technology to enhance driving safety. Detecting driver drowsiness through real-time video analysis is a critical component, aligning perfectly with the initiative's goal of prioritizing road safety. This task proactively identifies drowsiness signs, preventing potential road hazards by alerting drivers promptly. With a strong emphasis on driver safety, the system intervenes to avert accidents, showcasing its commitment to proactive road safety measures. This initiative demonstrates the power of technology in prioritizing driver safety, contributing to a safer driving environment for all.

Specifications:

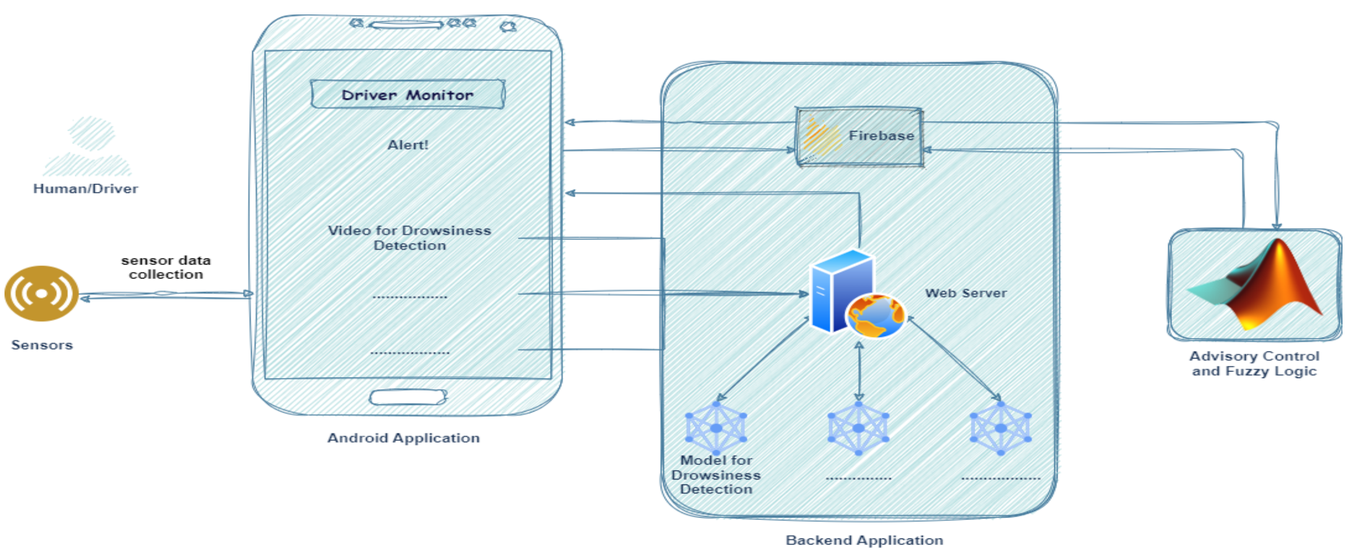
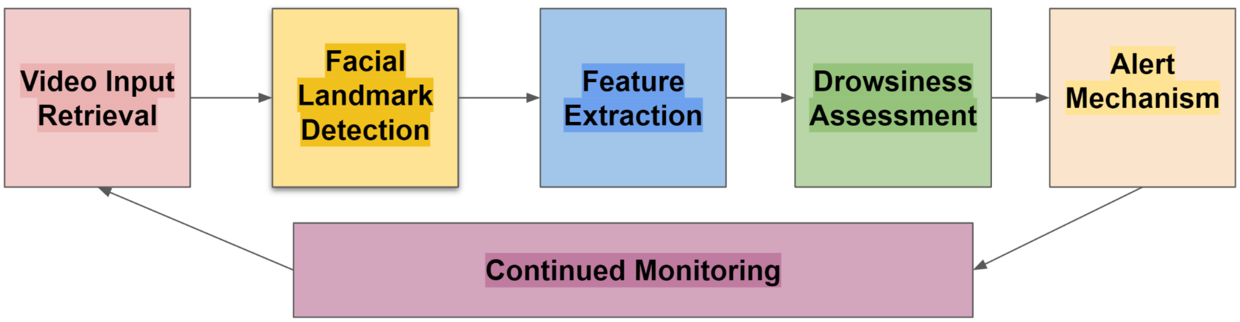


Figure 1: Control Flow

After the app's launch, continuous video capturing is initiated to track the driver's actions. The collected data is forwarded to an AWS server where a drowsiness detection model is trained and stored for testing. Leveraging OpenCV's facial landmark detection model, trained on various image datasets of drivers' faces, proved superior to other models like CNN, CNN-LSTM, or RNNs. This algorithm, decision tree-based, processes real-time video feeds from the camera sensors, identifying and mapping 68 facial points, including eyes, eyebrows, nose, and mouth. Analyzing these key points enables the system to monitor specific facial movements indicating drowsiness, such as frequent blinking, prolonged eye closure, or yawning. Upon detecting these signs, the system triggers an auditory alert, notifying the driver and mitigating potential accidents due to drowsiness. Similar components are considered for other models, and AWS returns the output to the UI, which then forwards it to Firebase for processing of fuzzy logic and advisory control, determining whether an alert should be issued.

Design:

1. Component architecture diagram:



2. Overview of the design:

- Video Input Retrieval:

The process starts by capturing real-time video feed from dedicated camera sensors, concentrating on the driver's face, ensuring a continuous retrieval of video frames at a set rate.

- Facial Landmark Detection:

The video frames retrieved are analyzed through computer vision techniques like OpenCV facial landmark model, mapping key facial landmarks(like eyes & mouth) critical for evaluating drowsiness.

- Feature Extraction:

Using the detected landmarks, the system extracts crucial facial movements, analyzing parameters such as eye closures, blink frequency, and yawning actions to identify drowsiness indicators.

- Drowsiness Assessment:

The system assesses the driver's drowsiness level by analyzing the extracted features, deciding if the detected signs surpass a predefined threshold indicating drowsiness.

- Alert Mechanism:

When signs of drowsiness are detected, the system triggers alerts through audible warnings, visual cues, or notifications, ensuring the driver's attention is drawn to their drowsy state.

- Continued Monitoring:

The system continously monitors the driver's facial expressions and movements to assess drowsiness levels, ensuring ongoing vigilance.

3. Tech-stack:

OpenCV (Open Source Computer Vision Library), Python Programming Language, TensorFlow, Keras, Dlib, dlib's Facial Landmark Detector, and Webcams**.**

Testing strategies:

1. Unit Testing - Facial Landmark Detection Module: Validate landmark accuracy using sample images and ensure consistent detection across varying lighting conditions and facial orientations.

2. Integration Testing - Module Integration: Verified interactions among modules (landmark detection, feature extraction, drowsiness assessment) to ensure smooth data flow and comprehensive functionality.
3. Performance Testing - Real-time Processing Performance: Measure system speed and responsiveness in analyzing real-time video input to meet or surpass defined performance benchmarks.

Navigating Challenges:

Achieving consistent accuracy in detecting facial landmarks under different conditions was a big challenge. Expanded our dataset with diverse samples and used advanced preprocessing techniques to address this. Seamlessly integrating multiple modules demanded rigorous testing and refining how these components communicated. Lessons learned are the significance of continuous testing, adaptable methods, and constant improvement, ensuring our drowsiness detection system becomes more reliable and accurate over time.