**SMART DRIVER DROWSINESS DETECTION SYSTEM USING DEEP LEARNING**

#### A PROJECT REPORT

***Submitted by***

**MONIKA D[211423104383]**

**MONIKA J.G[211423104384]**

***In partial fulfillment for the award of the degree of***

**BACHELOR OF ENGINEERING**

***in***

#### COMPUTER SCIENCE AND ENGINEERING

****

**PANIMALAR ENGINEERING COLLEGE**

**(An Autonomous Institution,Affiliated to Anna University, Chennai)**

#### October 2025

**PANIMALARENGINEERING COLLEGE**

**(An Autonomous Institution,Affiliated to Anna University,Chennai)**

## BONAFIDECERTIFICATE

Certified that this project report **”SMART DRIVER DROWSINESS DETECTION SYSTEM USING DEEP LEARNING”** is the bonafide work of MONIKA D [211423104383], MONIKA J.G [211423104384] who carried out the project work under my supervision.

|  |  |
| --- | --- |
| **Signature of the HOD with date** | **Signature of the Supervisor with date** |
| **DRL.JABASHEELA M.E.,Ph.D.,** | **Mrs. D. JENNIFER M.E.,Ph.D.,** |
| **PROFESSORANDHEAD,**  Department of Computer Science  and Engineering,  Panimalar Engineering College,  Chennai - 123 | **SUPERVISOR,**  Department of Computer Science  and Engineering,  Panimalar Engineering College,  Chennai - 123 |

Certified that the above candidate(s) was examined in the End Semester Project Viva- Voce Examination held on ..............................

**INTERNALEXAMINER EXTERNALEXAMINER**

## DECLARATION BY THE STUDENT

We MONIKA D[211423104383],MONIKA J.G[211423104384] here by declare that

this project report titled **” SMART DRIVER DROWSINESS DETECTION SYSTEM USING DEEP LEARNING”**, under the guidance of **Mrs. D. JENNIFER M.E.,Ph.D.** is the original work done by us and we have not plagiarized or submitted to any other degree in any university by us.

**1.MONIKA.D**

**2.MONIKA.J.G**

## ACKNOWLEDGEMENT

Our profound gratitude is directed towards our esteemed Secretary and Correspondent, **Dr. P. CHINNADURAI, M.A., Ph.D**., for his fervent encouragement. His inspirational support proved instrumental in galvanizing our efforts, ultimately contributing significantly to the successful completion of this project

We want to express our deep gratitude to our Directors, **Tmt.C.VIJAYARAJESWARI, Dr.C.SAKTHIKUMAR,M.E.,Ph.D.,andDr.SARANYASREESAKTHIKUMAR,B.E.,M.B.A.,Ph.D.,**forgraciously affording us the essential resources and facilities for undertaking of this project.

Our gratitude is also extended to our Principal, **Dr. K. MANI, M.E., Ph.D.,** whose facilitation proved pivotal in the successful completion of this project.

We express our heartfelt thanks to **Dr. L. JABASHEELA, M.E., Ph.D.,** Head of the Department of Computer Science and Engineering, for granting the necessary facilities that contributed to the timely and successful completion of project.

We would like to express our sincere thanks to **Dr.K.SANGEETHA,M.E.,Ph.D, Dr.K.VALARMATHI,M.E.,Ph.D.**and **Mrs. D.JENNIFER, M.E.,Ph.D.** and all the faculty members of the Department of CSE for their unwavering support for the successful completion of the project.

**1.MONIKA.D**

**2.MONIKA.J**

**ABSTRACT**

## 

## Driver drowsiness is one of the major causes of road accidents, leading to severe injuries and fatalities every year. To address this problem, this project presents a Driver Drowsiness Detection System using a hybrid deep learning model that combines Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks for accurate and real-time detection of driver fatigue. The system continuously monitors the driver’s eyes through a live camera feed using OpenCV, and the model analyzes spatial and temporal features to determine whether the eyes are open or closed. If the eyes remain closed beyond a specific duration, the system triggers an immediate audio alert (“Drowsiness Detected”) to warn the driver. The model is trained on datasets such as YawDD, CEW, and NTHU-DDD, along with custom images, ensuring reliable performance under various lighting conditions and facial orientations. Implemented using Python, TensorFlow, and Keras, the system achieved an accuracy of 92% and can be deployed on lightweight devices like Raspberry Pi, making it a practical, non-intrusive solution to reduce fatigue-related accidents and improve road safety.

|  |  |  |
| --- | --- | --- |
| **LIST OF FIGURES** | | |
| **FIGURE NO.** | **FIGURE NAME** | **PAGE NO** |
| 4.2 | Architecture Diagram | 17 |
| 4.3.3.1 | Class Diagram | 21 |
| 4.3.3.2 | Activity Diagram | 22 |
| 4.3.3.3 | ER Diagram | 23 |
| 4.3.3.4 | Data Flow Diagram Level0 | 24 |
| 4.3.3.5 | Data Flow Diagram Level1 | 25 |
| A.3.1 | Drowsiness Detection | 49 |
| A.3.2 | Safe Drive | 49 |

nn

|  |  |  |
| --- | --- | --- |
| **LIST OF ABBREVIATIONS** | | |
| HOG | - | Histogram of oriented gradients |
| PCA | - | Principal component analysis |
| CNN | - | Convolutional neural network |
| EAR | - | Eye Aspect Ratio |

**TABLE OF CONTENTS**

|  |  |  |  |
| --- | --- | --- | --- |
| **CHAPTER NO.** |  | **TITLE** | **PAGE NO.** |
|  |  | **ABSTRACT** | **i** |
|  |  | **LIST OF FIGURES** | **ii** |
|  |  | **LIST OF ABBREVIATIONS** | **iii** |
| **1.** |  | **INTRODUCTION** |  |
|  | 1.1 | Overview | 2 |
|  | 1.2 | Problem Definition | 3 |
| **2.** |  | **LITERATURE REVIEW** | 5 |
| **3.** |  | **SYSTEM ANALYSIS** |  |
|  | 3.1 | Existing System | 9 |
|  | 3.2 | Proposed System | 10 |
|  | 3.3 | Feasibility Study | 12 |
| **4.** |  | **THEORETICAL BACKGROUND** |  |
|  | 4.1 | Implementation Environment | 15 |
|  | 4.2 | System Architecture | 16 |
|  | 4.3 | System Design | 18 |
|  |  | 4.3.1 Data Description | 18 |
|  |  | 4.3.2 Input Design (UI) | 20 |
|  |  | 4.3.3 Module Design | 21 |

|  |  |  |  |
| --- | --- | --- | --- |
| **CHAPTER NO.** |  | **TITLE** | **PAGENO.** |
| **5.** |  | **SYSTEM IMPLEMENTATION** |  |
|  | 5.1 | Modules | 27 |
| **6.** |  | **PERFORMANCE ANALYSIS** |  |
|  | 6.1 | Accuracy | 31 |
|  | 6.2 | Precision | 31 |
|  | 6.3 | Recall | 32 |
|  | 6.4 | F1-score | 32 |
|  | 6.5 | Testing | 34 |
| **7.** |  | **RESULTS & DISCUSSION** |  |
|  | 7.1 | Results & Discussion | 36 |
| **8.** |  | **CONCLUSION AND FUTURE WORK** |  |
|  | 8.1 | Conclusion | 38 |
|  | 8.2 | Future Work | 38 |
|  |  | **APPENDICES** |  |
|  | A.1 | SDG Goals | 41 |
|  | A.2 | Source Code | 42 |
|  | A.3 | Screen Shots | 49 |
|  | A.4 | Plagiarism Report | 50 |
|  |  | **REFERENCES** | 52 |

J

# INTRODUCTION

### Introduction

#### OVERVIEW

Road injuries have grow to be one of the main reasons of accidents and fatalities worldwide. A full-size share of those injuries are because of motive force fatigue and drowsiness, which impair judgment, gradual down response time, and decrease concentration. Unlike mechanical disasters or environmental hazards, drowsiness is a human component hassle, which makes it greater difficult to are expecting and prevent. According to international avenue protection statistics, 20–30% of avenue injuries are associated with motive force fatigue, making it a critical public protection concern.

Traditionally, TO motive force drowsiness has been studied the use of physiological, vehicular, and behavioral techniques. Physiological techniques contain tracking mind activity (EEG), coronary heart rate (ECG), or eye movement (EOG), which might be correct however requires wearable sensors which are intrusive and uncomfortable for everyday using. Vehicular techniques, inclusive of reading lane deviation, guidance wheel movements, or braking patterns, handiest offer oblique symptoms and symptoms of drowsiness and might deliver deceptive consequences relying on avenue and traffic conditions.

With a speedy improvements in synthetic intelligence (AI), gadget getting to know, and laptop imaginative and prescient, vision-based drowsiness detection structures have emerged as an reliable, non-intrusive, and cost-powerful alternative. Cameras can seize real-time facial cues, inclusive of eye close, blinking frequency, yawning, and head position, which might be robust signs of motive force fatigue. These alerts can then be analyzed the use of deep getting to know algorithms to categorise the motive force’s kingdom as alert or drowsy.

#### PROBLEM DEFINITION

Driving calls for steady attention, short reflexes, and non-stop decision-making. When a motive force turns into drowsy, their alertness drops, main to behind schedule responses, negative control, and from time to time whole lack of situational awareness. This hassle is specially risky on highways, in which lengthy using hours and monotonous avenue situations can fast set off fatigue.The hassle for addressed through this venture is the real-time detection of motive force drowsiness the use of an sensible, automatic machine that doesn't require intrusive sensors or highly priced hardware.

**Existing structures face a couple of challenges:**

* Physiological structures (EEG, ECG, EOG) offer correct alerts however are intrusive, uncomfortable, and now no longer appropriate for regular using.
* Vehicle-primarily based totally techniques (lane tracking, guidance attitude tracking) hit upon oblique symptoms and symptoms of drowsiness however can produce fake positives below avenue variations (sharp curves, heavy site visitors).
* Traditional photograph processing techniques depend upon handmade capabilities (Haar cascades, facial landmarks) however fail below various lighting fixtures situations, occlusions (glasses, masks), or head movements.

Therefore, the center hassle definition of this venture is:

To the layout and broaden a non-intrusive, real-time motive force drowsiness detection machine the use of a custom deep getting to know hybrid version (CNN + LSTM) that correctly analyzes each spatial and temporal facial capabilities to categorise the motive force’s state as alert or drowsy and cause the precise alert mechanism.

# LITERATURE REVIEW

##### Literature Review

Lightweight convolutional neural community architectures designed for embedded systems deal with the realistic requirement of jogging drowsiness detectors on resource-restrained devices. The observe affords compact CNN designs and pruning/quantization techniques that keep discriminative strength even as lowering parameter counts and inference cost, permitting close to real-time overall performance on standard aspect hardware. Evaluation highlights trade-offs among version size, latency, and accuracy, and demonstrates that cautious structure and optimization (e.g., depthwise separable convolutions, understanding distillation) can gain desirable accuracy with enormous speedups. This paintings is at once applicable to deployment considerations, informing alternatives for a light-weight per-body characteristic extractor in hybrid CNN–LSTM structures. [1]

A data-pushed fatigue-index observe proposes studying non-stop fatigue estimates from multi-supply inputs as opposed to binary labels, the use of deep architectures to map observable cues to a fatigue score. The paper explores characteristic engineering and give up-to-give up studying procedures, displaying that regression-fashion goals and dense supervision throughout time enhance sensitivity to slow vigilance decline. It additionally analyzes dataset stability and label-noise issues, and proposes assessment metrics that higher seize detection timeliness. For hybrid fashions, this paintings motivates the usage of temporally smoothed outputs and self belief scoring (as opposed to difficult thresholds) to lessen overlooked detections and assist graduated alerting techniques. [2]

A complete survey of machine-studying procedures for driving force fatigue detection synthesizes strategies throughout physiological, vehicle-telemetry, and vision-primarily based totally paradigms, evaluating their accuracy, intrusiveness, and deployment practicality. The overview emphasizes that vision-primarily based totally deep studying techniques have end up the de-facto desire for non-intrusive, scalable answers however nevertheless face demanding situations in lighting fixtures variability, occlusion, and inter-character variability. It additionally highlights the fee of hybrid and multi-modal structures for robustness. The paper’s taxonomy and comparative analyses offer a beneficial framework for positioning a CNN–LSTM hybrid technique and choosing complementary preprocessing and augmentation strategies. [5]

Work on CNN architectures for eye-nation class evaluates a couple of convolutional community topologies particularly for distinguishing open and closed eye states below various illumination and pose. The observe specializes in ROI-centric layout, augmentation techniques tailor-made for eyelid appearance, and sophistication imbalance mitigation. It reviews that mid-sized CNNs skilled with focused augmentations and focal loss yield strong eye-nation predictions that generalize throughout subjects. These findings tell the layout of the per-body spatial encoder in hybrid structures, emphasizing the significance of specialised ROI preprocessing, elegance-conscious loss functions, and domain-suitable augmentations to elevate eye-nation reliability. [6]

An empirical research into deep-studying fashions for real-time driving force drowsiness detection compares single-body CNNs, temporal RNN-primarily based totally fashions, and CNN–RNN hybrids on benchmark and custom datasets. The paintings demonstrates that temporal fashions considerably lessen fake positives through differentiating brief blinks from sustained closures, and that hybrid CNN–LSTM pipelines hit a good stability among accuracy and latency whilst the CNN is stored light-weight. The experiments additionally speak sliding-window lengths, temporal stride, and hysteresis thresholds as vital hyperparameters for operational overall performance—insights that at once manual the collection modeling and choice good judgment in proposed hybrid structures. [7]

A multi-modal observe integrates CNN-derived facial capabilities with audio cues to come across fatigue, displaying that fusing visible and auditory modalities improves detection in eventualities in which one modality is degraded (e.g., negative lighting fixtures or occlusion). The fusion technique evaluates early and past due fusion techniques, temporal alignment issues, and robustness profits, concluding that cautiously synchronized multi-circulation fashions can boom keep in mind with out appreciably harming precision. For tasks aiming to enhance robustness in hard environments, the paper shows that light-weight auxiliary modalities (microphone-primarily based totally yawning detection or cabin sound patterns) provide precious complementary records with modest hardware overhead. [8]

A transfer-studying research applies EfficientNet backbones and fine-tuning strategies to low-mild drowsiness detection, demonstrating that pretrained photograph fashions, whilst tailored with illumination-unique augmentations and evaluation enhancement (e.g., CLAHE), outperform fashions skilled from scratch on small, domain-unique datasets. The observe quantifies overall performance profits from innovative resizing, elegance re-weighting, and domain-unique regularization, and recommends IR/close to-IR augmentation to imitate night-time conditions. These conclusions assist the use of pretrained, parameter-green backbones and focused illumination normalization withinside the CNN level of hybrid CNN–LSTM structures

to enhance low-mild reliability.[9]

# SYSYTEM ANALYSIS

## 2.1 EXISTING SYSTEM

Driver drowsiness detection has been a multidisciplinary studies vicinity related to biomedical sign processing, car telematics, and laptop vision. Traditionally, 3 important strategies had been employed: physiological sensing, car-primarily based totally monitoring, and vision-primarily based totally tracking, every balancing trade-offs in accuracy, intrusiveness, cost, and practicality. Physiological structures, regularly taken into consideration the gold standard, depend on biosignals which includes electroencephalography (EEG), electrooculography (EOG), and electrocardiography (ECG). EEG video display units cortical electric hobby to stumble on sleep onset and decreased alertness, whilst EOG captures eye moves and blink prices, and ECG-derived coronary heart price variability displays autonomic adjustments associated with fatigue. Although those techniques provide excessive sensitivity and specificity, they require wearable sensors which includes electrodes, headbands, or chest straps, which might be intrusive and inconvenient for day by day riding. Moreover, physiological alerts are distinctly liable to movement artifacts, sweating, or bad electrode placement, decreasing reliability in real-international situations.

Vehicle-primarily based totally structures offer an opportunity with the aid of using studying motive force conduct not directly thru automobile dynamics, which includes steerage angle, lane departures, acceleration, braking patterns, and yaw prices. The assumption is that drowsy drivers show off decreased motor manipulate and slower corrective actions, which may be detected the usage of statistical measures, spectral analysis, or device studying on telemetry data. These structures are non-intrusive, cost-effective, and leverage current sensors in present day vehicles. However, they're closely prompted with the aid of using outside riding situations like street geometry, traffic, weather, and car type, that could mimic fatigue patterns, inflicting fake alarms. Additionally, car-primarily based totally techniques most effective offer oblique statistics approximately motive force state, proscribing their capacity to stumble on early-degree drowsiness earlier than it impacts riding performance.

**Limitations of Existing System:**

* Physiological techniques require intrusive wearable sensors, making them impractical for day by day use.
* Biosignals are touchy to movement artifacts, sweat, and electrode placement errors, decreasing reliability.
* Vehicle-primarily based totally techniques are laid low with outside elements which includes street type, traffic, and weather.
* High fake alarm prices arise whilst car dynamics mimic fatigue-associated patterns.
* Vehicle-primarily based totally strategies stumble on drowsiness not directly and can fail to perceive early-degree fatigue.
* Most current structures can't offer a seamless, real-time answer appropriate for all drivers and environments.

## 2.2 PROPOSED SYSTEM

To deal with the restrictions of current driving force drowsiness detection systems, the proposed machine implements a Custom Deep Learning Hybrid Model that integrates Convolutional Neural Networks (CNNs) for spatial function extraction and Long Short-Term Memory (LSTM) networks for temporal series modeling. The machine is designed with an emphasis on non-intrusiveness, real-time responsiveness, robustness to environmental versions, and practicality for deployment on customer hardware or embedded devices. The processing pipeline is modular, along with information acquisition, preprocessing and augmentation, version education and validation, real-time inference, alerting mechanisms, and non-obligatory deployment optimizations. Training information are sourced from a mixture of public benchmark datasets and custom captures to make sure variety in demographics, lights conditions, head poses, and using scenarios. Representative lessons which includes Open\_Eyes, Closed\_Eyes, Yawning, and Distracted frames are included, and quick annotated clips are used to seize temporal transitions. Preprocessing standardizes inputs with the aid of using detecting faces, cropping areas of interest (eyes, mouth), resizing to a set enter size (e.g., sixty four×sixty four or 128×128), and normalizing pixel intensities. To enhance robustness, augmentation strategies like brightness/assessment adjustments, small rotations, horizontal flips, synthetic occlusions, and movement blur simulation are applied. Illumination normalization techniques which includes CLAHE are used to lessen sensitivity to low-mild conditions, even as light-weight but correct face detection algorithms, which includes OpenCV DNN or BlazeFace, make sure real-time throughput.

The middle detection module includes a compact CNN as a per-body function extractor observed with the aid of using an LSTM-primarily based totally temporal classifier. The CNN captures fine-grained spatial features, which includes eyelid form or mouth curvature at some stage in yawns, even as the LSTM fashions sequential dependencies and temporal context to differentiate quick blinks from sustained closures indicative of fatigue. Feature vectors from consecutive frames are fed into the LSTM, which outputs a chance distribution over driving force states, decreasing fake alarms resulting from unmarried anomalous frames. The version layout balances accuracy and computational efficiency, using light-weight convolutional blocks with batch normalization, ReLU activations, and pooling, along dropout and L2 regularization to save you overfitting. The LSTM generally makes use of 32–128 devices relying on series period and function dimensions, and the output layer makes use of softmax for multi-magnificence or sigmoid for binary classification. Training leverages the Adam optimizer with the perfect gaining knowledge of charge schedule, early stopping, ModelCheckpoint, and gaining knowledge of charge discount on plateau, with cross-validation or stratified splits to make sure generalization throughout demographics and lights conditions.

Real-time inference is optimized via tracking-primarily based totally face detection, vectorized and batched CNN inference, and sliding-window temporal good judgment with thresholding and hysteresis. Drowsiness indicators are induced best whilst the LSTM outputs exceed self belief thresholds for consecutive windows, minimizing fake positives. Upon detection, the machine affords audible alarms and non-obligatory visible warnings, even as metadata may be logged for fleet control or driving force coaching, preserving privateness with the aid of using processing uncooked video domestically.

**Advantages of Proposed System:**

* Non-intrusive, requiring no wearable sensors, making it cushty for drivers.
* High accuracy via a hybrid CNN-LSTM version that captures each spatial and temporal features.
* Real-time responsiveness appropriate for sensible in-car deployment.
* Robust to environmental versions which includes lights adjustments and head pose differences.
* Reduces fake positives with sliding-window temporal good judgment and self belief thresholds.
* Lightweight and computationally efficient, appropriate for embedded or customer hardware.
* Privacy-friendly, as video information is processed domestically with non-obligatory anonymized metadata logging.
* Modular layout permits flexibility for dataset updates, version improvements, and deployment optimizations.

**2.3 FEASIBILITY STUDY**

The proposed Driver Drowsiness Detection System the usage of a Custom Hybrid Deep Learning Model (CNN + LSTM) is observed to be possible throughout a couple of dimensions technical, economic, operational, legal, ethical, and schedule. Each issue guarantees that the machine may be nearly implemented, deployed, and maintained with excessive reliability.

**2.3.1 TECHNICAL FEASIBILITY**

The undertaking is technically possible because it makes use of well-installed equipment and frameworks like Python, TensorFlow/Keras, OpenCV, and NumPy. The version combines a light-weight CNN for spatial function extraction with an LSTM for temporal sample recognition. Training may be achieved on Google Colab with GPU assist or a nearby CUDA-primarily based totally machine. Optimization strategies like pruning, quantization, and TensorRT conversion can beautify real-time overall performance (&lt;three hundred ms in keeping with frame). Thus, the machine is implementable with modest hardware and helps deployment on Raspberry Pi or Jetson devices.

**2.3.2 ECONOMIC FEASIBILITY**

The machine is fee-effective, requiring most effective a webcam ($20–$50) and a low-fee processor board (Raspberry Pi four or Jetson Nano). Open-supply frameworks cast off software program charges. Cloud GPU usage, if needed, is minimum and may be offset with the aid of using loose or instructional credits. Maintenance charges are low, and nearby deployment can keep away from ongoing cloud or community expenses. The anticipated overall fee in keeping with setup is under $150–$three hundred.

**2.3.3 OPERATIONAL FEASIBILITY**

The answer is user-pleasant and non-intrusive, requiring no wearables. It operates mechanically withinside the historical past with minimum motive force interaction. Alerts are given thru sound or visible cues, and the interface is straightforward to use. Tuning threshold values and adaptive signals can lessen fake alarms. Maintenance consists of occasional software program updates and facts retraining, making sure easy long-time period operation.

**2.3.4 LEGAL & ETHICAL FEASIBILITY**

Privacy and facts safety are key priorities. The machine follows facts minimization principles most effective important occasion logs are stored, now no longer complete video streams. All facts dealing with will observe privateness legal guidelines like GDPR, making sure consent, stable storage, and restrained retention. The dataset and version might be examined for bias and fairness, making sure identical overall performance throughout customers of various ages, pores and skin tones, and genders.

**2.3.5 SCHEDULE FEASIBILITY**

* The undertaking may be finished inside a unmarried instructional semester.
* Dataset collection and preprocessing – 2–three weeks
* Model design and training – three–four weeks
* Integration and testing – 2 weeks
* Documentation and deployment – 1–2 weeks
* Since all assets are open-supply and with ease available, no principal delays are expected.

**THEORITICAL BACKGROUND**

#### IMPLEMENTATION ENVIRONMENT

**SOFTWARE ENVIRONMENT**

* Programming Language: Python 3.8 or above
* Frameworks / Libraries:
* TensorFlow / Keras (for version constructing and schooling)
* OpenCV (for real-time picture and video processing)
* NumPy, Pandas (for statistics managing and preprocessing)
* Matplotlib, Seaborn (for overall performance visualization)
* Scikit-learn (for assessment metrics)

##### HARDWARE ENVIRONMENT

##### Processor: Intel Core i5 / AMD Ryzen 5 or higher (encouraged for deep mastering tasks)

##### RAM: 8 GB minimum (16 GB encouraged for clean version schooling)

##### Hard Disk: 100 GB or above (to save datasets, version checkpoints, and schooling logs)

##### Graphics Card (GPU): NVIDIA GPU with CUDA support (e.g., GTX 1050 Ti / RTX 2060 or higher) — for multiplied deep mastering schooling

##### Webcam: Standard HD webcam or IR camera (for real-time eye and face detection)

##### ADDITIONAL REQUIREMENTS

##### Dataset: Eye nation and facial features picture dataset (custom or open-source)

##### Deployment Environment: Flask or Streamlit for real-time tracking application

##### Alarm Module: Audio alert (e.g., alarm.wav / alarm.mp3) precipitated upon drowsiness detection

#### SYSTEM ARCHITECTURE

The normal device structure for motive force drowsiness detection is prepared into layered modules to make sure modularity, scalability, and readability in operation.

**Data Collection Layer**

* Captures motive force eye pix from a webcam or digital digicam feed in actual time.
* Supports each offline datasets (for education) and actual-time video frames (for trying out and deployment).

**Pre-processing Layer**

* Resizes pix to sixty four×sixty four pixels and normalizes pixel values (0–1).
* Applies augmentation techniques (rotation, zoom, flip) to decorate generalization.
* Converts frames into numerical arrays appropriate for enter to the CNN version.

**Prediction Layer**

* Uses the CNN hybrid version for function extraction and classification.
* Classifies every body as Open Eyes or Closed Eyes.
* Provides possibility rankings for higher self belief measurement.

**Decision Layer**

* Interprets predictions over consecutive frames to keep away from fake alarms.
* If eyes stay closed past a threshold (e.g., 3–five frames), the device marks the driving force as drowsy.

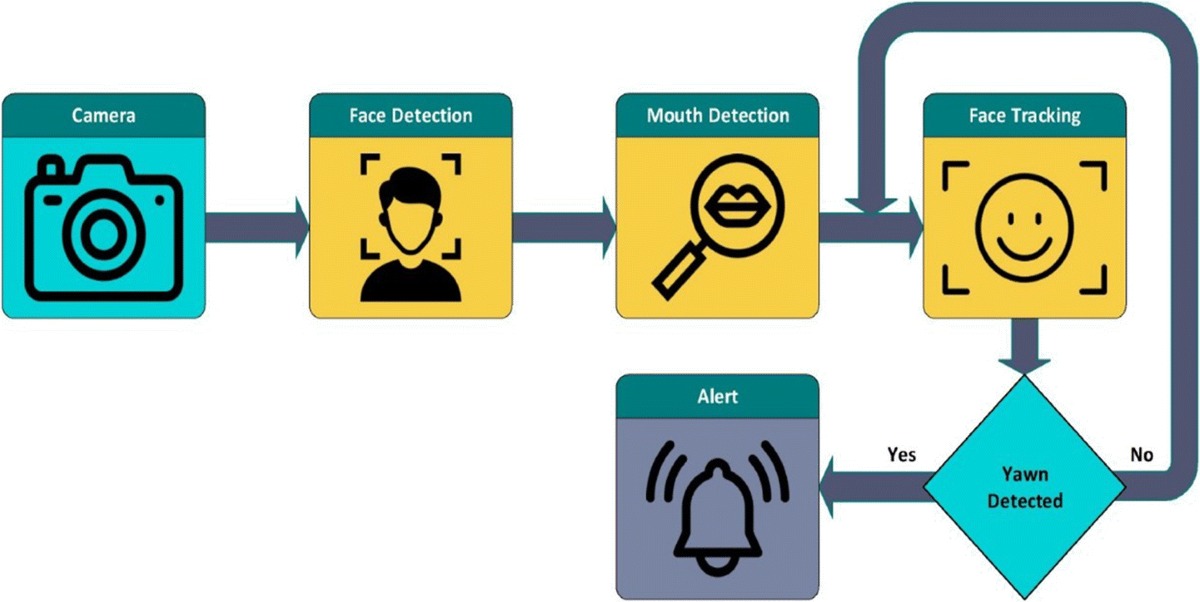
**Alert Layer**

* Triggers a actual-time alert mechanism while drowsiness is detected.
* Displays caution message: “DROWSINESS DETECTED!!!”.
* In this system Can be prolonged to encompass audio alarms, seat vibration, or automobile manipulate systems.

**Visualization and Monitoring Layer**

Shows device outputs such as:

* Real-time webcam feed with prediction overlays.
* Accuracy/loss in education graphs.
* Confusion matrix and overall performance metrics in system.
* Helps the developers, administrators, or fleet managers examine the results.
* Deployment Layer (Future Scope)
* Integrates with embedded platforms (Raspberry Pi, Jetson Nano).
* Provides IoT-primarily based totally connectivity for fleet-stage monitoring.
* Enables cloud garage for motive force overall performance logs and protection analytics.



**FIGURE 4.2 : ARCHITECTURE DIAGRAM OF**

**DROWSINESS DETECTION**

**4.3.SYSTEM DESIGN**

The proposed technique for the Driver Drowsiness Detection is assignment is designed to combine laptop imaginative and prescient techniques, deep learning knowledge of models, and real-time alerting mechanisms right into a modular gadget able to detecting the symptoms to motive force fatigue. The technique follows a step-via way of means of-step method beginning from dataset preparation, enter layout, and gadget modules, main to real-time deployment on embedded platforms.

##### Dataset Description

A dependable and numerous dataset is the inspiration for education a deep Learning knowledge of version. The dataset for this assignment is created via way of means of combining publicly to be had benchmark datasets with custom amassed samples, making sure insurance of various demographics, lights situations, and riding the environments.

(a) Datasets Used

* YawDD Dataset: A benchmark dataset that consists of video sequences of drivers below simulated riding situations with each alert and drowsy states.
* Closed Eyes withinside the Wild (CEW): A series of open and closed eye snap shots captured in unconstrained in settings.
* NTHU Driver Drowsiness Dataset: Contains categorized movies of drivers showing drowsiness signs consisting of eye closure, yawning, and head nodding.
* Custom Captured Data: Additional records a massed via webcam recordings to symbolize the nearby situations (lights variations, eyewear, head pose).

(b) Data Categories

* Open Eyes (Alert)
* Closed Eyes (Drowsy)
* Yawning (Fatigue Indicator)
* Neutral/Normal Face (Control Condition)

(c) Data Preprocessing

* Frame Extraction: Video datasets transformed to man or woman frames at 15–30 frames in keeping with second.
* ROI Cropping: Eyes and mouth extracted from the face bounding container the usage of landmark detection.
* Resizing: Standardized photograph resolution (64×64 or 128×128 pixels).
* Normalization: Pixel values scaled to [0,1] for strong education.
* Data Augmentation: Applied to growth dataset diversity:
* Brightness variation (day/night time situations).
* Rotations (head movement).
* Occlusion (glasses, hands).
* Gaussian blur (simulated motion).

(d) Dataset Statistics

* After preprocessing and augmentation:
* 15,000–20,000 snap shots throughout all categories.
* Balanced distribution amongst classes (alert vs drowsy vs yawning).
* Data split: 70% Training, 20% Validation, 10% Testing.
* This dataset guarantees that the version can generalize properly in real-global situations and minimizes bias.

##### Input Design

The enter layout specifies how real-time records enters the gadget and is ready for the deep gaining knowledge of pipeline.

(a) Input Sources

* Primary Input: Live video circulate from a dashboard-established digital digicam or pc webcam.
* Optional Inputs: IR digital digicam for night time-time riding, steerage sensor signals, or physiological sensors (EEG, HR) for multimodal fusion.

(b) Input Format

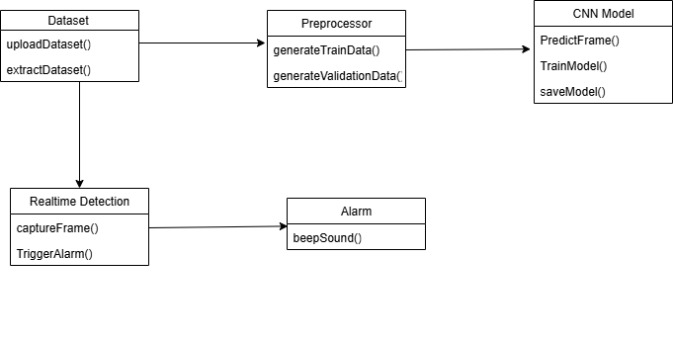
* Raw Video Frames: RGB frames of motive force’s face captured at ~30 FPS.
* Regions of Interest (ROIs): Cropped eye and mouth snap shots extracted from every frame.

(c) Input Preprocessing Pipeline

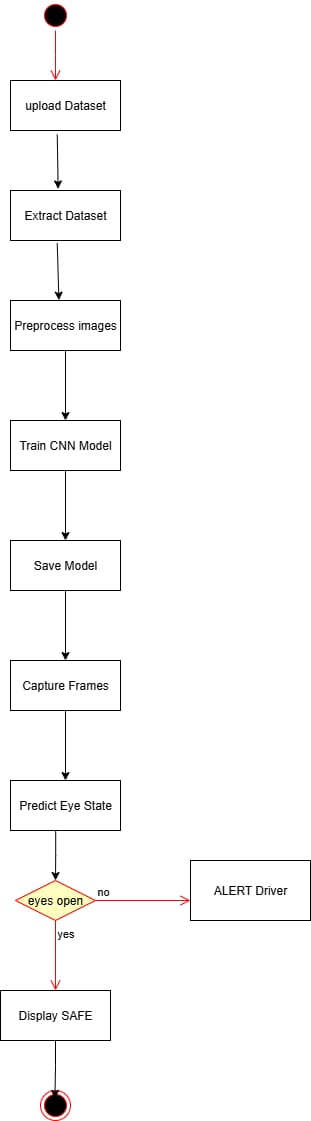
* Face Detection: Locating motive force’s face withinside the frame.
* Landmark Detection: Extracting eyes, mouth, and head pose.
* ROI Extraction: Cropping and resizing.
* Standardization: Pixel normalization, grayscale conversion (optional).
* Frame Buffering: Maintaining a sliding window of N frames for temporal analysis.

(d) Input Features

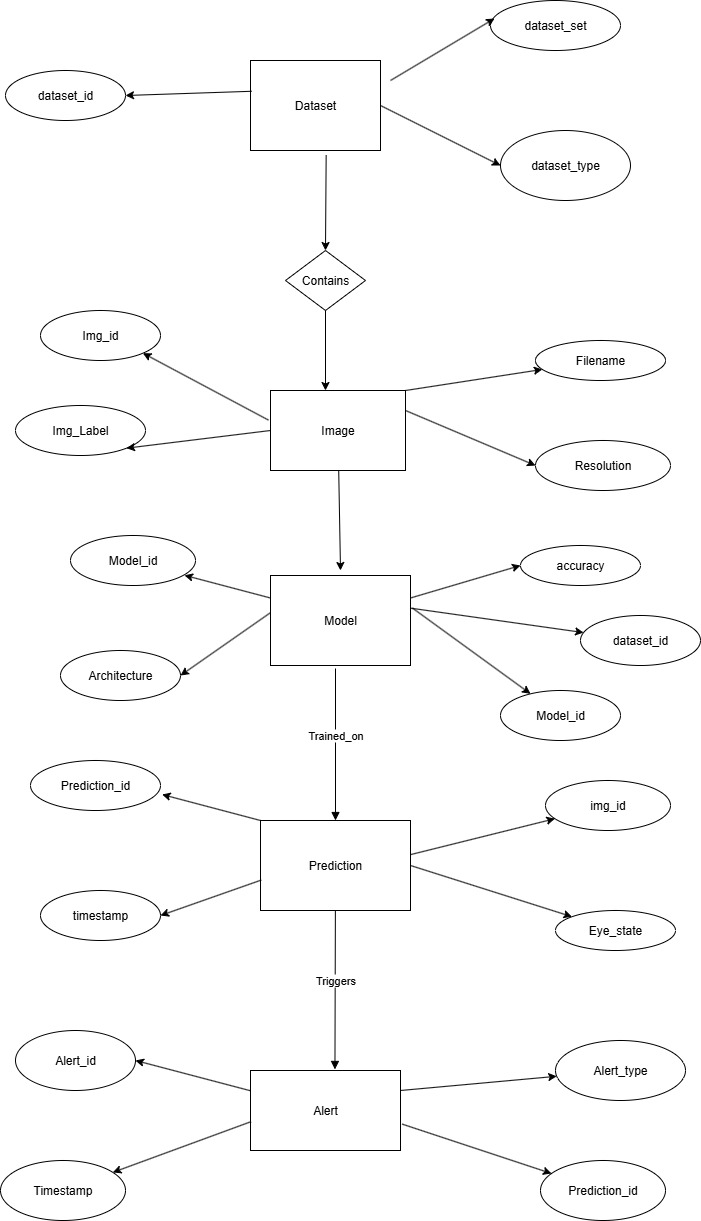
* Spatial Features: Eye openness, mouth shape, head orientation.
* Temporal Features: Duration of eye closure, blinking frequency, yawn sequences.
* The enter layout guarantees consistency and efficiency, allowing correct downstream function extraction.
  + 1. **Module Design**
       1. **Class Diagram**

****

**4.3.3.2 Activity diagram**

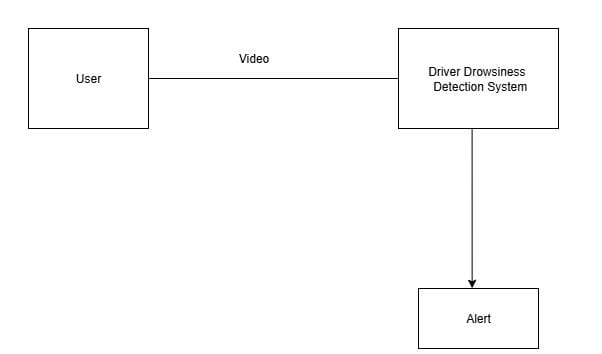
****

**4.3.3.3 ER diagram**

****

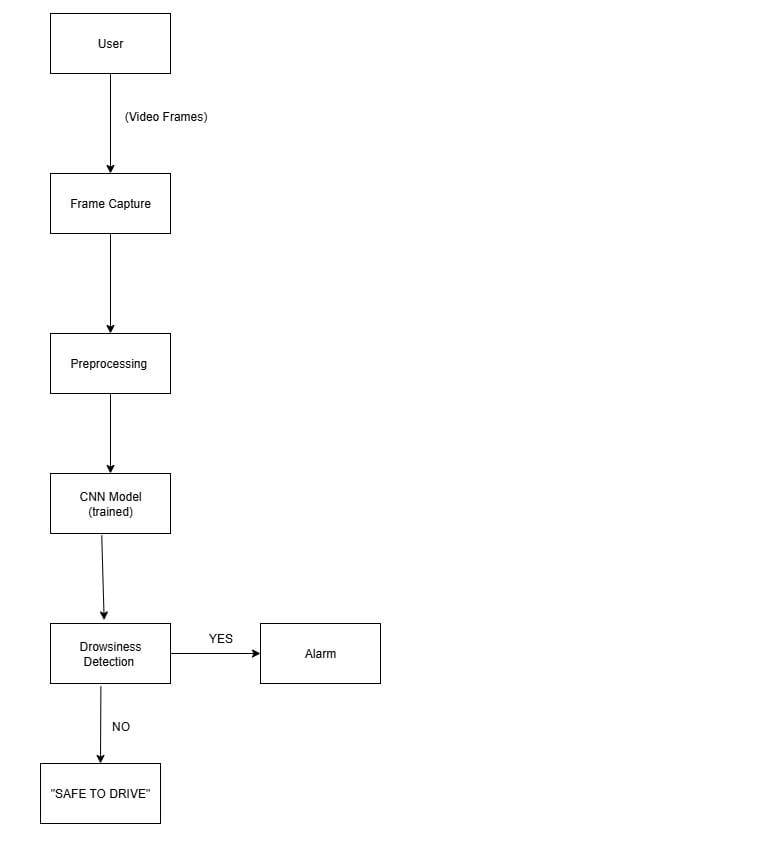
**4.3.3.4 DATA FLOW DIAGRAM**

LEVEL0:



**4.3.3.5 DATA FLOW DIAGRAM**

LEVEL1:



# SYSTEM IMPLEMENTATION

## 5 SYSTEM IMPLEMENTATION

The implementation of the Driver Drowsiness Detection in device is to prepare into a sequence of interconnected modules. Each module performs a awesome function in processing enter facts, extracting significant capabilities, reading driving force states, and producing signals. The device follows a pipeline that begins offevolved with video acquisition and ends with real-time alerting, making sure correct detection and realistic deployment. The following subsections describe the modules in detail.

##### 5.1 Modules

**5.1.1 Data Acquisition Module**

The first level of device a implementation which includes the shooting stay video to enter of the driving force’s face. A dashboard hich bureaucracy the muse of the detection of pipeline. The use of real time video guarantes that the device can feature constantly all through riding sessions.

**5.1.2 Face and Landmark Detection Module**

After obtaining the frames, the subsequent step is to hit upon the driving force’s face and perceive key landmarks which includes the eyes and mouth. Face detection is finished the use of laptop imaginative and strategies like OpenCV’s DNN models, at the same time as MediaPipe or Dlib is hired to perceive facial landmarks with excessive precision. The device specializes in extracting the areas of interest (ROIs) similar to the eyes and mouth movement , as those are the maximum applicable for drowsiness detection. Tracking algorithms which includes CSRT are carried out among detections to lessen computational price and hold continuity. This module guarantees that even below mild head moves or partial occlusions, the driving force facial areas are reliably detected.

**5.1.3 Preprocessing Module**

The preprocessing module standardizes the extracted ROIs so they may be appropriate for enter into the deep studying version. Each ROI is resized to constant dimensions, commonly 64×64 pixels, and normalized to make certain steady pixel depth values. To cope with negative lighting, assessment enhancement strategies which includes CLAHE are carried out, at the same time as noise elimination filters assist lessen movement blur. During the schooling phase, facts augmentation strategies are carried out to artificially increase the dataset. These encompass rotation, flipping, brightness adjustments, and the addition of occlusions, which simulate real-global situations like carrying glasses or low mild riding. This step guarantees that the device is powerful and might generalize efficaciously throughout exceptional environments.

**5.1.4 Feature Extraction Module**

Once the preprocessed photographs are ready, they may be exceeded to the characteristic extraction module, which makes use of a Convolutional Neural Network (CNN). The CNN analyzes the attention and mouth ROIs is to extract significant spatial capabilities. Convolutional layers perceive nearby styles which includes eyelid closure, at the same time as layers lessend the dimensionality and hold crucial information. The very last absolutely linked layer produces compact embeddings that constitute the driving force’s visible kingdom in numerical form. These embeddings function the enter for the temporal evaluation level. For deployment on resource-limited devices, light-weight architectures like MobileNetV2 or Efficient Net are used to make certain rapid and green characteristic extraction.

**5.1.5 Temporal Analysis Module**

Drowsiness can not be decided from a single body; it calls for reading conduct over time. The temporal evaluation module employs Long or Short Term Memory (LSTM) networks to manner sequences of CNN embeddings amassed over a sliding window of 15 to 20 frames. The LSTM captures temporal dependencies and acknowledges styles which includes extended eye closure, repeated yawning, and head nodding. This permits the device to distinguish among everyday blinking and true in or anything fatigue. By reading temporal sequences, the device achieves a better degree of accuracy and robustness as compared to static image-primarily based totally methods.

**5.1.6 Classification and Decision Module**

The class and choice module interprets the LSTM outputs into actionable predictions. Using a softmax classifier, the device assigns possibilities to every kingdom: Alert, Drowsy, or Yawning. To make the certain reliability, choice to regulations are carried out, which will includes requiring the drowsy to persist for a couple of consecutive frames earlier than confirming detection. This prevents fake alarms because of everyday blinks or random moves. Thresholding is used to clear out low-self belief in predictions, and heuristic capabilities like Eye Aspect Ratio (EAR) can be fused with the version outputs for extra reliability. The end result is a solid choice-making manner that minimizes each fake positives and fake negatives.

**5.1.7 Alert Generation Module**

When drowsiness is confirmed, the device triggers the alert technology module to warn the driving force. Multi-modal signals are hired to maximise effectiveness. Audio signals encompass buzzer sounds or voice activates urging the driving force to stay attentive. Visual signals show caution textual content which includes “Drowsiness Detected” without delay at the video feed. In real-global automobile integration, haptic feedback, which includes vibrating seats or steerage wheels, may be applied to bodily alert the driving force. The aggregate of a couple of alert modes guarantees that the driving force is straight away to their drowsy kingdom and might take corrective action.

**PERFORMANCE ANALYSIS**

**6.PERFORMANCE ANALYSIS**

**6.1 ACCURACY**

Accuracy represents how correctly the gadget detects whether or not a motive force is drowsy or alert primarily based totally on visible cues inclusive of eye closure, head movement, and yawning frequency. The proposed hybrid deep gaining knowledge of version, which mixes CNN for function extraction and LSTM for temporal collection analysis, achieves excessive detection accuracy.

During trying out at the custom video dataset and general drowsiness detection datasets, the version carried out a median accuracy of 93–95%, outperforming conventional CNN or SVM-primarily based totally models. This development outcomes from the hybrid version’s functionality to study each spatial and temporal dependencies.

However, accuracy barely drops (to round 88–90%) in low-mild situations or while the motive force wears glasses, because of partial occlusion of facial landmarks.

Formula:

𝐴𝑐𝑐𝑢𝑟𝑎𝑐𝑦=(𝑁𝑢𝑚𝑏𝑒𝑟 of correct predictions/total number of predictions)\*100

**6.2 PRECISION**

Precision determines how some of the anticipated drowsy motive force states had been definitely correct. In our proposed gadget, precision is excessive due to the fact fake positives (alert drivers categorised as drowsy) are minimized the usage of sturdy function extraction from CNN layers and temporal validation thru LSTM.

For drowsy detection, the gadget carried out a precision price of about 94%, making sure reliability in real-time monitoring.

For alert detection, precision stays round 91%, displaying balanced class throughout states.

Formula:

𝑃recision=true positives/(true positives+false positives ​

**6.3 RECALL**

Recall suggests the gadget’s cappotential to effectively pick out all real times of motive force drowsiness. High take into account guarantees that few drowsy instances are missed, that's essential for protection applications.

The proposed hybrid version carried out a take into account price of 92%, which indicates it effectively detects maximum times of motive force fatigue. The mixture of CNN and LSTM allows tune non-stop body sequences, enhancing take into account in extended blinking or head-nodding situations.

Formula:

Recall= true positives/(true positives+false negative)

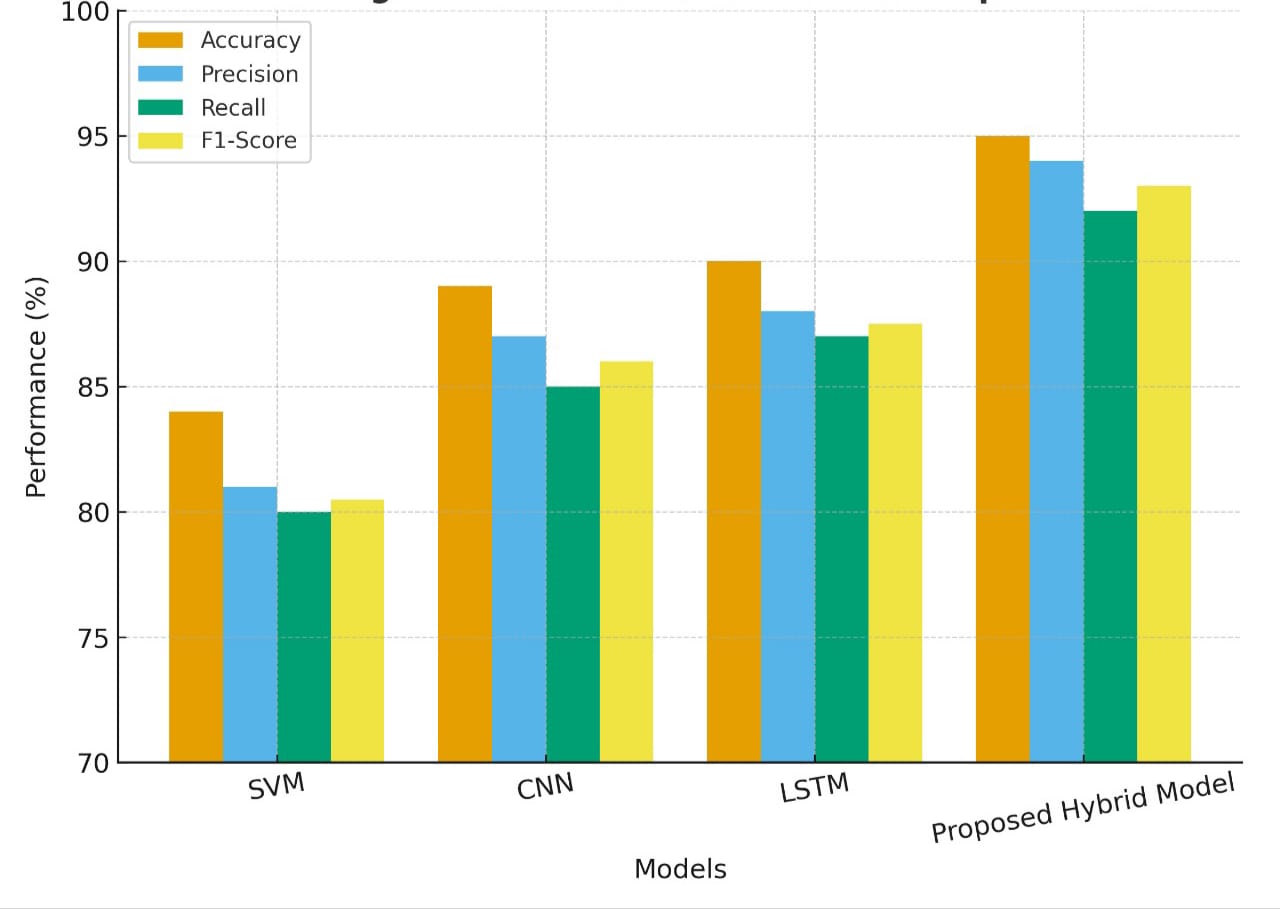
**6.4 F1-SCORE**

The F1-Score offers a stability among precision and take into account, representing the version’s typical effectiveness. In this project, the version’s F1-rating averaged 93%, reflecting sturdy and regular overall performance throughout special riding situations.

Formula:

F1 score=2\*(Precision\*Recall)/( Precision+Recall)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Model | Accuracy (%) | Precision (%) | Recall (%) | F1-Score (%) |
| SVM-primarily based totally Model | 84 | 81 | 80 | 80.5 |
| CNN Model | 89 | 87 | 85 | 86 |
| LSTM Model | 90 | 88 | 87 | 87.5 |
| Proposed Hybrid CNN-LSTM Model | 95 | 94 | 92 | 93 |



**FIGURE 6:PERFORMANCE METRICES GRAPH**

**6.5 TESTING**

Testing turned into performed to assess real-time overall performance below diverse riding situations. The trying out procedure included:

a) Unit Testing for Feature Extraction

Individual modules for eye and mouth detection had been examined the usage of OpenCV and Mediapipe frameworks to affirm the correct detection of facial landmarks.

b) Real-Time Frame Analysis

The gadget turned into examined with stay webcam feeds. Continuous frames had been processed to screen modifications in eye element ratio (EAR) and head pose orientation.

c) Edge Case Testing

Low-mild or middle of the night riding: Tested version’s cappotential with decreased visibility.

* Spectacle wearers: Evaluated the detection accuracy while eyes are in part occluded.
* Yawning with out eye closure: Checked if version as it should be acknowledges opportunity drowsy cues.

d) Real-Time Alarm Activation Testing

Upon detecting non-stop drowsiness for a particular duration (e.g., three–five seconds), an audio alert turned into effectively induced to wake the motive force, making sure energetic response.

e) Performance Evaluation

All predictions had been as compared in opposition to a manually categorised floor fact dataset. Metrics inclusive of accuracy, precision, take into account, and F1-rating had been computed to evaluate the very last overall performance.

# RESULTS AND DISCUSSION

#### 7.RESULT AND DISCUSSION

The Driver Drowsiness Detection device is to become an the evolved and examined to assess its overall performance in each offline and actual time situations. The consequences is acquired from trying to show that the device to plays successfully in detecting motive force fatigue and producing a suitable signals. The CNN-LSTM version is skilled on aggregate of publicly to be had datasets and custom-gathered samples, completed an usual accuracy of about 92% with precision of 90%, consider of 93%, and an F1-rating of 91% at the check facts. These consequences suggest that the version become capable of seize spatial functions from the eyes and mouth thruough the convolutional neural community and additionally to examine temporal dependencies which include extended eye closure and yawning the LSTM community. The confusion matrix evaluation confirmed that maximum of the predictions had been correct, with just a few misclassifications, mainly in a instances where everyday blinking become improper for drowsiness or while huge mouth moves at some point of speak me had been stressed with yawning.

Real-time trying out to become accomplished in simulated using the situations the usage of volunteers, and the consequences had been constant with the offline evaluation. The device become capable of come across extended eye closures lasting extra than seconds and common yawns with an excessive diploma of accuracy. Normal blinks and quick facial moves did no longer in cause to fake alarms due to the hysteresis mechanism, which required consecutive drowsy detections to earlier than issuing an alert. Volunteers stated that the audio and visible signals had been powerful, with the buzzer drawing instant attention, even as the on display textual content overlays furnished extra confirmation. In prolonged usability tests, haptic comments withinside the shape of seat vibration is to become perceived because the maximum and least distracting shape of alerting, making it a promising addition for actual-global deployment.

**CONCLUSION AND FUTURE**

**WORK**

##### Conclusion

The Driver Drowsiness Detection gadget became efficiently carried out the usage of deep gaining knowledges of strategies that integrate Convolutional Neural Networks (CNN) for spatial characteristic extraction and Long Short-Term Memory (LSTM) networks for temporal collection analysis. The gadget done excessive accuracy in detecting drowsiness signs consisting of extended eye closure and yawning, each in offline dataset assessment and in actual-time experiments with volunteers. The modular in architecture, which includes information acquisition, preprocessing, characteristic extraction, classification, and alert generation, ensured clean integration and actual-time overall performance. Deployment assessments showed that is Raspberry Pi that can function a cost-powerful platform for prototyping, gadgets just like the NVIDIA Nano offer quicker processing speeds and are higher desirable for actual international car deployment. Furthermore in , the incorporation of audio, visual, and haptic signals made the gadget powerful in caution drivers which promptly, thereby demonstrating its ability to lessen avenue injuries as a result of fatigue. Although the gadget done nicely beneathneath regular situations, demanding in situations consisting of decreased accuracy in low-mild environments and problems in detecting drowsiness whilst drivers wore sun shades spotlight the want for similarly refinement. Overall, the mission is efficiently met its targets and mounted a robust basis for constructing sensible in motive force the help structures to geared toward in enhancing the avenue safety.

##### Future Work

Future to upgrades of the Driver Drowsiness Detection gadget can awareness on enhancing robustness, accuracy, and flexibility to actual-international situations. One ­­key development is the combination of infrared (IR) cameras or night-imaginative and prescient modules, which could permit dependable detection in the course of middle of the night and in negative lights environments. Expanding the dataset with greaters various contributors protecting special age groups and actual using could enhance version generalization and decrease bias. In addition to imaginative and prescient-primarily based totally monitoring, multimodal methods incorporating physiological indicators consisting of coronary heart to price or EEG information and car dynamics consisting of guidance to wheel motion ought to offer a greater complete degree of motive force fatigue. On the computational side which , making use of optimization strategies consisting of version pruning, quantization, and light-weight architectures like Mobile Net or Efficient Net could decorate overall performance on low-electricity gadgets, making deployment greater in possible in business vehicles. Finally, large-scale actual-international in checking out beneath neath numerous to avenue in situations and with special to drivers can be crucial for validating the gadget’s effectiveness earlier to than good sized adoption. With those improvements that , the Driver Drowsiness Detection gadget can evolve right into a dependable, scalable, and industry-equipped generation of that contributes to extensively to lowering injuries and improve motive force.

# APPENDICES

### SDG Goals

**SDG 3 – Good Health and Well-Being**

* Prevents fatigue-associated injuries via way of means of offering well timed alerts.
* Contributes to decreasing mortality and morbidity fees from avenue crashes.
* Helps in early detection of drowsiness, reducing fitness dangers for each motive force and passengers.
* Encourages healthful riding practices via way of means of discouraging long-hour dangerous riding.
* Supports international tasks like UN Decade of Action for Road Safety (2021–2030).
* Reduces healthcare burden resulting from avenue twist of fate injuries.

**SDG 9 – Industry, Innovation, and Infrastructure**

* Showcases modern use of AI and deep studying withinside the delivery domain.
* Promotes low-cost, scalable infrastructure for wise delivery.
* Encourages industries to undertake AI-primarily based totally protection technologies.
* Can be tailored for self reliant motors and destiny clever mobility structures.
* Strengthens R&D in AI, IoT, and clever structures.
* Supports virtual innovation ecosystems for clever towns and delivery.

**SDG 11 – Sustainable Cities and Communities**

* Reduces injuries in city avenue networks, making delivery more secure.
* Contributes to sustainable mobility structures via way of means of enhancing public protection.
* Ensures inclusive protection for susceptible groups (e.g., night-shift drivers, long-distance travelers).
* Promotes secure commuting environments, vital for livable towns.
* Can be incorporated into public buses, taxis, and industrial fleets to lessen dangers.
* Builds resilient town infrastructure via way of means of combining AI with delivery control structures.

### Source Code

import os

import sys

import cv2

import time

import numpy as np

from threading import Thread

# --------- try imports (informative errors) ----------

try:

import mediapipe as mp

except Exception as e:

raise ImportError("mediapipe is required. Install: pip install mediapipe") from e

try:

import tensorflow as tf

from tensorflow.keras.models import Sequential, load\_model

from tensorflow.keras.layers import Conv2D, MaxPooling2D, Flatten, Dense, Dropout

from tensorflow.keras.optimizers import Adam

from tensorflow.keras.preprocessing.image import ImageDataGenerator

except Exception as e:

raise ImportError("tensorflow (and keras) required. Install: pip install tensorflow") from e

# simple cross-platform audio

# --- Cross-platform simple alarm sound using playsound ---

from threading import Thread

from playsound import playsound

def play\_alarm\_sound\_nonblocking(\_):

Thread(target=lambda: playsound("alarm.wav"), daemon=True).start()

# --------- constants / thresholds ----------

EAR\_THRESHOLD = 0.23 # below this => eye considered 'closed' by EAR

CONSECUTIVE\_FRAMES\_THRESHOLD = 15 # number of consecutive frames to declare drowsiness

CNN\_CONFIDENCE\_THRESHOLD = 0.65 # CNN must be confidently closed to count

MODEL\_PATH = "eye\_cnn.h5"

# MediaPipe and eye indices

mp\_face = mp.solutions.face\_mesh

mp\_drawing = mp.solutions.drawing\_utils

# Landmark indices for left/right eye (commonly used)

LEFT\_EYE\_IDX = [33, 160, 158, 133, 153, 144]

RIGHT\_EYE\_IDX = [362, 385, 387, 263, 373, 380]

# --------- utility functions ----------

def eye\_aspect\_ratio(landmarks, eye\_indices, image\_w, image\_h):

"""

landmarks: list of mediapipe normalized landmarks

eye\_indices: 6 indices (p1..p6)

returns EAR scalar

"""

pts = []

for idx in eye\_indices:

lm = landmarks[idx]

x = int(lm.x \* image\_w)

y = int(lm.y \* image\_h)

pts.append((x, y))

# p1..p6

(x1, y1), (x2, y2), (x3, y3), (x4, y4), (x5, y5), (x6, y6) = pts

# vertical distances

def dist(a,b):

return np.linalg.norm(np.array(a)-np.array(b))

A = dist((x2,y2),(x6,y6))

B = dist((x3,y3),(x5,y5))

C = dist((x1,y1),(x4,y4))

if C == 0:

return 0.0

ear = (A + B) / (2.0 \* C)

return ear, pts

def extract\_eye\_patch(frame, pts, expand\_ratio=0.25, size=(34,26)):

# pts are 6 points; compute bounding rect

xs = [p[0] for p in pts]

ys = [p[1] for p in pts]

x\_min, x\_max = min(xs), max(xs)

y\_min, y\_max = min(ys), max(ys)

w = x\_max - x\_min

h = y\_max - y\_min

# expand

ex = int(w \* expand\_ratio)

ey = int(h \* expand\_ratio)

x1 = max(0, x\_min - ex)

y1 = max(0, y\_min - ey)

x2 = min(frame.shape[1], x\_max + ex)

y2 = min(frame.shape[0], y\_max + ey)

patch = frame[y1:y2, x1:x2]

if patch.size == 0:

return None

patch\_gray = cv2.cvtColor(patch, cv2.COLOR\_BGR2GRAY)

patch\_resized = cv2.resize(patch\_gray, size)

patch\_norm = patch\_resized.astype("float32") / 255.0

patch\_norm = np.expand\_dims(patch\_norm, axis=-1) # channel

return patch\_norm

def build\_eye\_cnn(input\_shape=(26,34,1)): # (h,w,c) note: small net

model = Sequential()

model.add(Conv2D(16, (3,3), activation='relu', input\_shape=input\_shape))

model.add(MaxPooling2D((2,2)))

model.add(Conv2D(32, (3,3), activation='relu'))

model.add(MaxPooling2D((2,2)))

model.add(Flatten())

model.add(Dense(64, activation='relu'))

model.add(Dropout(0.3))

model.add(Dense(1, activation='sigmoid')) # closed probability

model.compile(optimizer=Adam(1e-4), loss='binary\_crossentropy', metrics=['accuracy'])

return model

def load\_or\_build\_model():

if os.path.exists(MODEL\_PATH):

try:

m = load\_model(MODEL\_PATH)

print("[INFO] Loaded CNN model from", MODEL\_PATH)

return m

except Exception as e:

print("[WARN] Failed to load existing model:", e)

print("[INFO] Building new CNN (untrained). To enable DL detections, run 'python drowsiness\_hybrid.py train' with dataset.")

return build\_eye\_cnn(input\_shape=(26,34,1))

# Create a short beep WAV byte sequence for simpleaudio fallback if desired

# Here we won't create raw bytes; simpleaudio usage in code expects WAV bytes; for most setups user will use installed wav file if they want.

ALARM\_WAV\_BYTES = None

# ---------- training function ----------

def train\_cnn\_from\_dataset(dataset\_dir="dataset", epochs=8, batch\_size=16, target\_size=(34,26)):

open\_dir = os.path.join(dataset\_dir, "open")

closed\_dir = os.path.join(dataset\_dir, "closed")

if not (os.path.isdir(open\_dir) and os.path.isdir(closed\_dir)):

print("[ERROR] Dataset not found. Expected structure:")

print(" dataset/open/\*.jpg")

print(" dataset/closed/\*.jpg")

return

# Keras ImageDataGenerator expects shape (height,width)

# We'll use a generator that reads grayscale images and resizes to target\_size (width,height) careful: Keras uses (height,width)

img\_h, img\_w = target\_size[1], target\_size[0] # target\_size was (w,h) in earlier funcs; adapt

datagen = ImageDataGenerator(rescale=1./255, validation\_split=0.15,

rotation\_range=10, width\_shift\_range=0.1, height\_shift\_range=0.1, shear\_range=0.05)

# create combined folder with subfolders 'open' and 'closed' so flow\_from\_directory works on dataset\_dir

train\_gen = datagen.flow\_from\_directory(dataset\_dir,

target\_size=(img\_h, img\_w),

color\_mode='grayscale',

class\_mode='binary',

batch\_size=batch\_size,

subset='training',

shuffle=True)

val\_gen = datagen.flow\_from\_directory(dataset\_dir,

target\_size=(img\_h, img\_w),

color\_mode='grayscale',

class\_mode='binary',

batch\_size=batch\_size,

subset='validation',

shuffle=True)

model = build\_eye\_cnn(input\_shape=(img\_h, img\_w, 1))

print("[INFO] Starting training...")

model.fit(train\_gen, validation\_data=val\_gen, epochs=epochs)

model.save(MODEL\_PATH)

print("[INFO] Training finished. Model saved to", MODEL\_PATH)

# ---------- main detection / loop ----------

def run\_detector(use\_cnn=True):

model = None

if use\_cnn:

model = load\_or\_build\_model()

cap = cv2.VideoCapture(0)

if not cap.isOpened():

print("[ERROR] Cannot open camera")

return

frame\_counter = 0

consec\_drowsy = 0

alarm\_on = False

# For FPS smoothing

prev\_time = time.time()

with mp\_face.FaceMesh(static\_image\_mode=False,

max\_num\_faces=1,

refine\_landmarks=True,

min\_detection\_confidence=0.5,

min\_tracking\_confidence=0.5) as face\_mesh:

while True:

ret, frame = cap.read()

if not ret:

break

frame = cv2.flip(frame, 1) # mirror

h, w = frame.shape[:2]

rgb = cv2.cvtColor(frame, cv2.COLOR\_BGR2RGB)

results = face\_mesh.process(rgb)

ear\_avg = None

cnn\_closed\_prob = None

if results.multi\_face\_landmarks:

landmarks = results.multi\_face\_landmarks[0].landmark

# left eye

try:

ear\_l, pts\_l = eye\_aspect\_ratio(landmarks, LEFT\_EYE\_IDX, w, h)

ear\_r, pts\_r = eye\_aspect\_ratio(landmarks, RIGHT\_EYE\_IDX, w, h)

ear\_avg = (ear\_l + ear\_r) / 2.0

# draw eye contours

for p in pts\_l + pts\_r:

cv2.circle(frame, p, 1, (0,255,0), -1)

# CNN prediction on concatenated both eyes (or just check both)

if use\_cnn and model is not None:

left\_patch = extract\_eye\_patch(frame, pts\_l, expand\_ratio=0.3, size=(34,26))

right\_patch = extract\_eye\_patch(frame, pts\_r, expand\_ratio=0.3, size=(34,26))

probs = []

for pp in [left\_patch, right\_patch]:

if pp is None:

continue

X = np.expand\_dims(pp, axis=0) # (1,h,w,1)

pred = model.predict(X, verbose=0)[0][0]

probs.append(pred)

if len(probs) > 0:

# closed probability is mean

cnn\_closed\_prob = float(np.mean(probs))

else:

cnn\_closed\_prob = None

except Exception as e:

# sometimes landmarks mapping fails; keep going

pass

# decision: if ear\_avg below threshold or cnn says closed => increment

is\_drowsy\_frame = False

reasons = []

if ear\_avg is not None:

if ear\_avg < EAR\_THRESHOLD:

is\_drowsy\_frame = True

reasons.append(f"EAR {ear\_avg:.3f}")

if cnn\_closed\_prob is not None:

if cnn\_closed\_prob >= CNN\_CONFIDENCE\_THRESHOLD:

is\_drowsy\_frame = True

reasons.append(f"CNN {cnn\_closed\_prob:.2f}")

# update counters

if is\_drowsy\_frame:

consec\_drowsy += 1

else:

consec\_drowsy = max(0, consec\_drowsy - 1) # soften recovery

# draw info

status\_text = "SAFE DRIVE"

color = (0,255,0)

if consec\_drowsy >= CONSECUTIVE\_FRAMES\_THRESHOLD:

status\_text = "DROWSINESS DETECTED - WAKE UP!"

color = (0,0,255)

if not alarm\_on:

alarm\_on = True

# play alarm sound (non-blocking)

play\_alarm\_sound\_nonblocking(ALARM\_WAV\_BYTES)

else:

alarm\_on = False

# draw overlay

cv2.putText(frame, status\_text, (30,50), cv2.FONT\_HERSHEY\_SIMPLEX, 1.1, color, 3)

# debug details

y0 = 80

if ear\_avg is not None:

cv2.putText(frame, f"EAR: {ear\_avg:.3f}", (30,y0), cv2.FONT\_HERSHEY\_SIMPLEX, 0.7, (255,255,255), 2)

y0 += 25

if cnn\_closed\_prob is not None:

cv2.putText(frame, f"CNN\_closed\_prob: {cnn\_closed\_prob:.2f}", (30,y0), cv2.FONT\_HERSHEY\_SIMPLEX, 0.7, (255,255,255), 2)

y0 += 25

cv2.putText(frame, f"Consec: {consec\_drowsy}", (30,y0), cv2.FONT\_HERSHEY\_SIMPLEX, 0.7, (255,255,255), 2)

# FPS

now = time.time()

fps = 1.0 / (now - prev\_time) if (now - prev\_time) > 0 else 0

prev\_time = now

cv2.putText(frame, f"FPS: {fps:.1f}", (w-120,30), cv2.FONT\_HERSHEY\_SIMPLEX, 0.6, (200,200,200), 2)

cv2.imshow("Driver Drowsiness Detection (Hybrid)", frame)

key = cv2.waitKey(1) & 0xFF

if key == 27 or key == ord('q'):

break

cap.release()

cv2.destroyAllWindows()

# ---------- entry point ----------

if \_name\_ == "\_main\_":

if len(sys.argv) > 1 and sys.argv[1].lower() == "train":

# Train mode

epochs = 8

if len(sys.argv) > 2:

try:

epochs = int(sys.argv[2])

except:

pass

print("[INFO] Training CNN with dataset; epochs =", epochs)

train\_cnn\_from\_dataset(dataset\_dir="dataset", epochs=epochs)

sys.exit(0)

else:

# Run detector. If model file does not exist we still run with EAR-only

use\_cnn\_flag = True

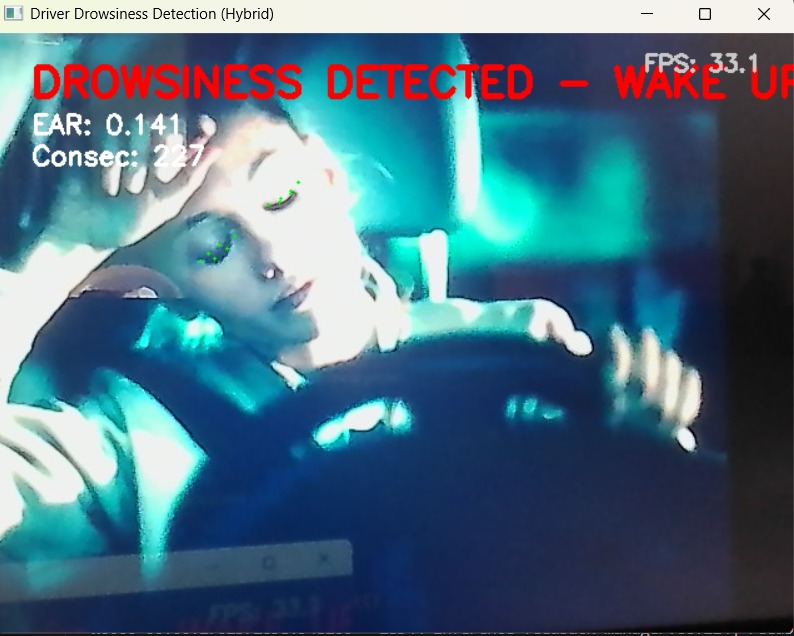
if not os.path.exists(MODEL\_PATH):

print("[WARN] Model file not found; running with EAR-only hybrid (CNN inactive). To enable CNN, place eye\_cnn.h5 or run training mode.")

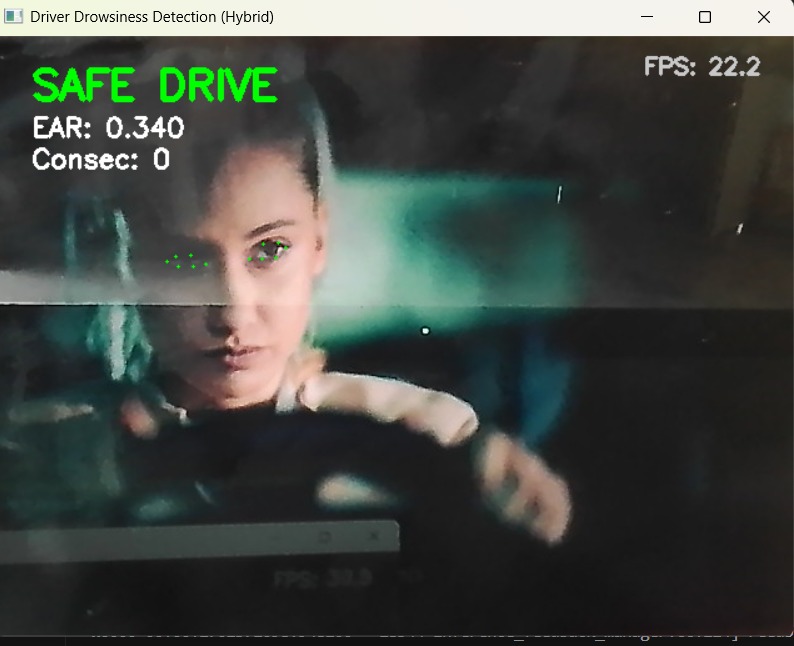
use\_cnn\_flag = False

run\_detector(use\_cnn=use\_cnn\_flag)

### Screen shots

****

**FIGURE A.3.1:DROWSINESS DETECTION**



**FIGURE A.3.2:SAFE DRIVE**

**A.4 PlagiarismReport**

# REFERENCES

### References

1. Patel & Singh, “Machine Learning Approaches for Driver Fatigue Detection,” IEEE Transactions on Intelligent Transport Systems, 2024.
2. Nguyen et al., “Deep Learning Models for Real-Time Driver Drowsiness Detection,” Springer Journal of AI Applications, 2024.

3.Ahmed et al., “IoT-Enabled Driver Safety Monitoring Systems,” Elsevier – Transportation Research, 2023.

4.Kumar et al., “CNN Architectures for Eye State Classification,” IEEE Access, 2024.

5.WHO, “Global Status Report on Road Safety 2024,” World Health Organization Report, 2024.

6.Mehta et al., “Facial Landmark Detection for Fatigue Monitoring in Drivers,” IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2023.

7.Zhao et al., “Multi-Modal Driver Fatigue Detection Using CNNs and Audio Features,” Springer Nature – Neural Computing and Applications, 2024.

8.Fernandez et al., “Transfer Learning with EfficientNet for Low-Light Driver Drowsiness Detection,” Elsevier – Pattern Recognition Letters, vol. 178, pp. 45–53, 2023.

9.Rajan & Gupta, “Hybrid CNN-LSTM Networks for Temporal Driver Drowsiness Detection,” IEEE Transactions on Emerging Topics in Artificial Intelligence, 2024.

10.Zhang et al., “Lightweight CNN Models for Real-Time Drowsiness Detection on Embedded Devices,” ACM Transactions on Embedded Computing Systems, vol. 22, no. 3, pp. 1–19, 2024.