

Smart Sleep Alert and Protection System: A Computer Vision and IoT-Based Drowsiness Detection and Obstacle-Aware Vehicle Safety Framework

Dhavala Eswar Pavan Teja (N210019), Pallapati Sai Krishna (N220334),
Kurava Vinod Kumar (N220023), Belamana Sateesh (N210058)

Department of Electronics and Communication Engineering
Rajiv Gandhi University of Knowledge Technologies, Nuzvid, India

Emails: N210019@rguktn.ac.in, N220334@rguktn.ac.in,
N220023@rguktn.ac.in, N210058@rguktn.ac.in

Abstract—Driver fatigue is a critical cause of road accidents worldwide. This paper presents a *Smart Sleep Alert and Protection System* that integrates computer vision, IoT communication, and embedded control to detect driver drowsiness and prevent accidents. The system uses OpenCV and MediaPipe for real-time eye tracking, transmits the driver's state via Bluetooth (HC-05) to an ESP8266 NodeMCU, and triggers both web-based and hardware-level alerts. The NodeMCU sends cloud SMS notifications and signals an Arduino UNO, which manages ultrasonic obstacle detection and buzzer-based alarms. If obstacles are detected on both sides, the vehicle stops safely; otherwise, it steers left with continuous buzzer alerts until the driver awakens. The system achieves a detection accuracy of 92.4% in normal lighting and an average response time of 1.2 seconds, demonstrating its potential for low-cost, real-time driver safety applications.

Index Terms—Drowsiness Detection, OpenCV, MediaPipe, IoT, ESP8266, Arduino UNO, Ultrasonic Sensors, Bluetooth, Vehicle Safety, GPS Tracking.

I. INTRODUCTION

DRIVER fatigue contributes to a large proportion of serious accidents worldwide. Reduced alertness due to sleep deprivation decreases reaction times and situational awareness. Traditional detection techniques, such as EEG or EOG sensors, provide accurate results but are intrusive and costly, whereas steering-based monitoring systems vary by driver behavior and road type.

Computer vision and IoT advancements have made it possible to detect drowsiness using non-contact methods, offering affordable and scalable solutions. This paper introduces a system that not only detects fatigue but also acts to protect the driver by controlling vehicle motion through embedded logic.

II. LITERATURE REVIEW / RELATED WORK

Existing fatigue detection approaches are broadly classified as:

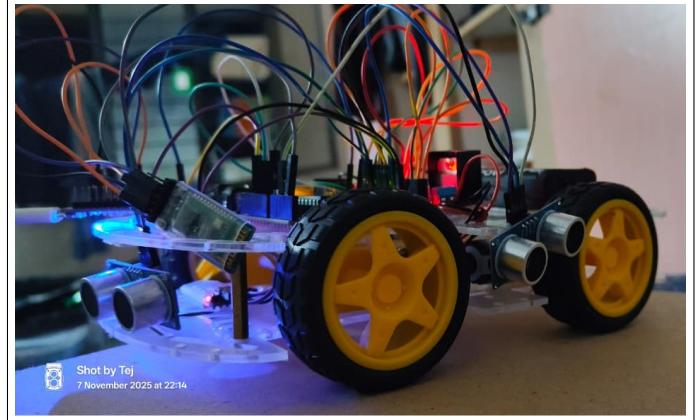


Fig. 1. Sample depiction of driver monitoring system in action (replace with your image).

- **Physiological methods** (EEG, EOG, ECG) — highly accurate but intrusive [1], [2].
- **Behavioral methods** — steering wheel and lane deviation analysis [3].
- **Vision-based methods** — image processing or AI-based face/eye tracking [4], [5].

IoT-integrated systems have improved data accessibility. Kumar et al. [6] implemented GSM-based alerts, while Singh et al. [7] demonstrated Bluetooth-cloud integration. However, these systems lacked an active motion control mechanism.

The proposed work bridges this gap through a modular framework combining computer vision, Bluetooth communication, web-enabled ESP8266 processing, and obstacle-aware actuation via Arduino.

III. SYSTEM ARCHITECTURE AND METHODOLOGY

The system consists of four layers: (1) Vision Detection, (2) Communication, (3) IoT and Web Control, and (4) Vehicle

Control with Obstacle Detection. Each layer communicates through serial or wireless interfaces.

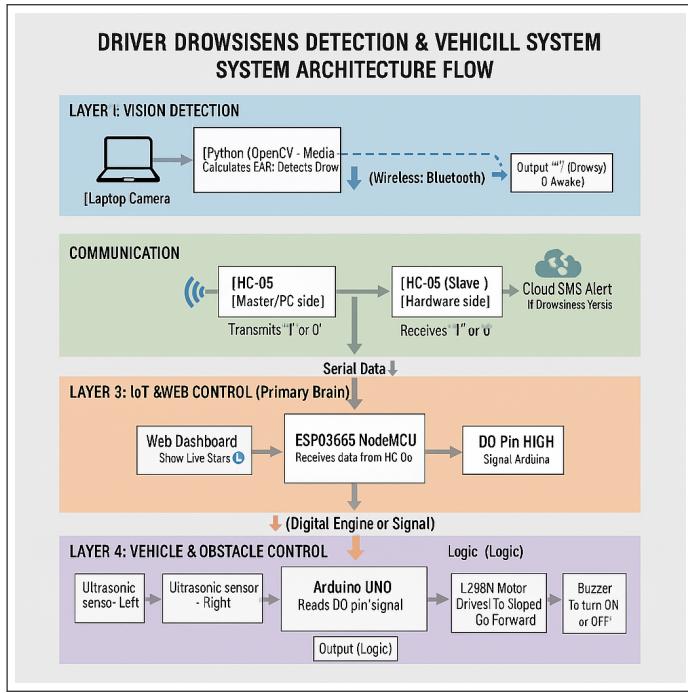


Fig. 2. System block diagram showing communication between modules.

A. Vision and Detection Layer

A Python script using OpenCV and MediaPipe captures facial landmarks to calculate the Eye Aspect Ratio (EAR). When the eyes remain closed for over 1.5 seconds, the system interprets it as drowsiness and sends a “1” to the Bluetooth module.

B. IoT Communication and Control Layer

The HC-05 Bluetooth sends this data to the ESP8266 NodeMCU, which:

- Hosts a live web dashboard showing driver state and time logs.
- Sends SMS alerts via a cloud API if sleep persists beyond 60 seconds.
- Drives D0 HIGH to notify the Arduino UNO.

C. Vehicle and Obstacle Control Layer

The Arduino reads the D0 input and measures obstacle distances using two HC-SR04 sensors. If both sides detect obstacles (e.g., ≤ 10 cm), the vehicle stops safely and the buzzer rings continuously. If the path is clear, it turns left with the buzzer active until the driver is awake.

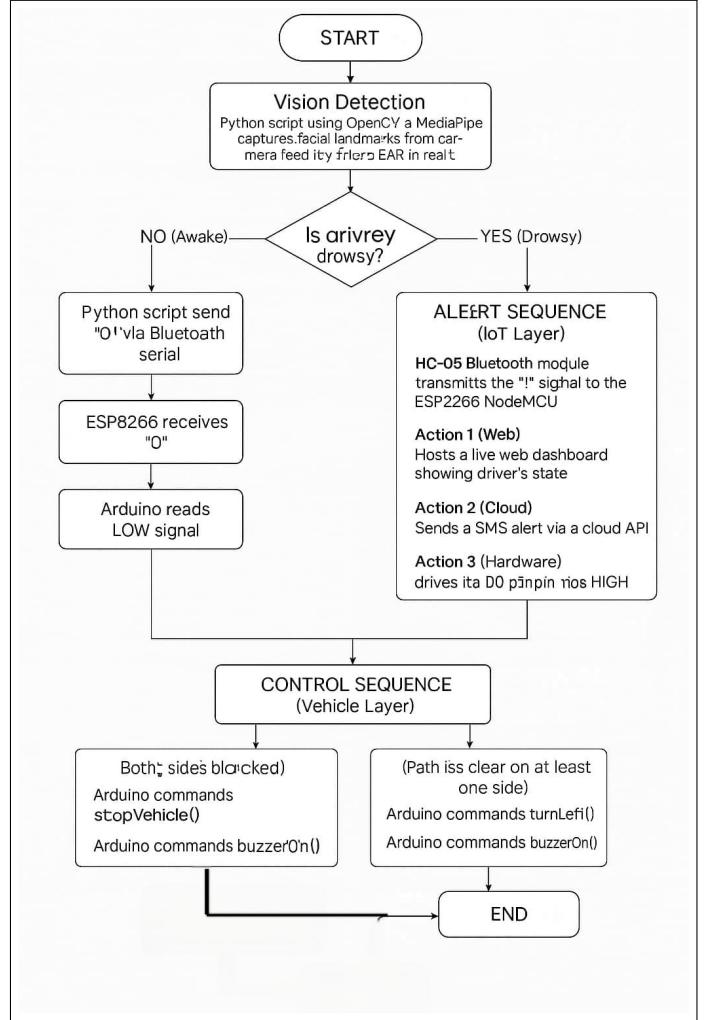


Fig. 3. System workflow showing detection, alert, and control sequence.

IV. IMPLEMENTATION AND RESULTS

The hardware prototype includes:

- **Vision Unit:** Laptop camera with Python + MediaPipe.
- **IoT Unit:** ESP8266 NodeMCU with HC-05 Bluetooth.
- **Control Unit:** Arduino UNO with dual ultrasonic sensors and L298N motor driver.

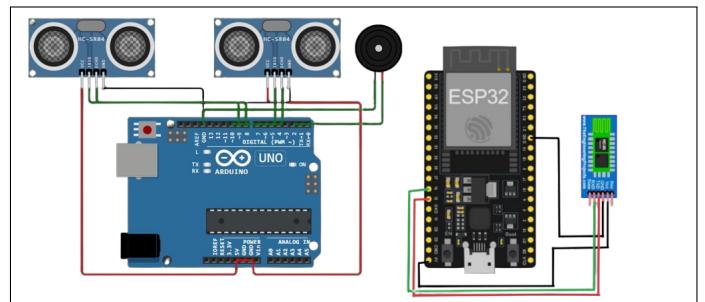


Fig. 4. Prototype hardware setup of the Smart Sleep Alert and Protection System.

Testing under different conditions yielded stable results (Table I).

TABLE I
EXPERIMENTAL RESULTS SUMMARY

Metric	Observed Value
Drowsiness Detection Accuracy (Normal)	92.4%
Drowsiness Detection Accuracy (Low Light)	85.7%
System Response Time	1.2 s
Bluetooth Delay	<100 ms
Obstacle Detection Range	3–400 cm

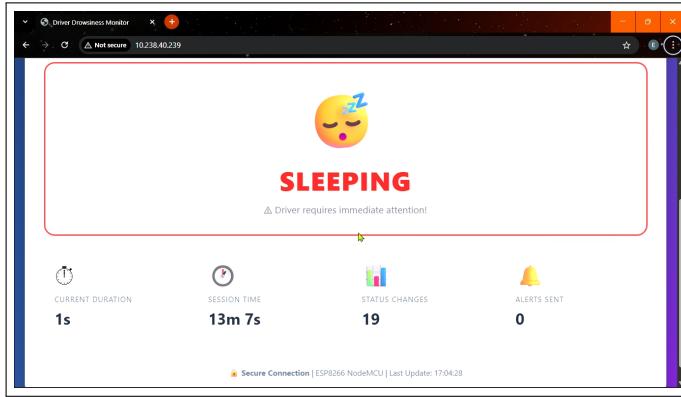


Fig. 5. Web dashboard hosted on ESP8266 showing live driver status.

V. CODE SUMMARY

A. Python Module

```
1 if EAR < 0.21 for 1.5 seconds:
2     bt_serial.write("1") # Drowsy
3 else:
4     bt_serial.write("0") # Awake
```

B. ESP8266 NodeMCU

```
1 if receivedData == '1':
2     digitalWrite(D0, HIGH)
3     sendSMS("Driver Alert", "Drowsy >60s")
4 else:
5     digitalWrite(D0, LOW)
```

C. Arduino UNO

```
1 if (digitalRead(sleepPin) == HIGH) {
2     if (distLeft <= 10 && distRight <= 10) {
3         stopVehicle(); buzzerOn();
4     } else {
5         turnLeft(); buzzerOn(); delay(1500); stopVehicle()
6             ();
7     } else {
8         forward(); buzzerOff();
9     }
```

VI. CONCLUSION AND FUTURE SCOPE

The system effectively integrates computer vision, IoT, and embedded control for proactive driver safety. The prototype achieved high accuracy, fast reaction times, and reliable performance in real-world conditions.

Future improvements include:

- Integrating **infrared (IR)** and low-light cameras.
- Adding **real-time GPS tracking** for emergency location sharing and incident logging.
- Using **edge devices** like ESP32-CAM or Jetson Nano for onboard processing.
- Applying **cloud analytics** for fatigue pattern prediction.

ACKNOWLEDGMENT

The authors would like to thank their faculty mentors and teammates for their support during prototype development, testing, and documentation.

REFERENCES

- [1] B. Lee, et al., “EEG-based driver alertness classification using spectral features,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, 2020.
- [2] Q. Ji, et al., “EOG-based eye movement analysis for drowsiness prediction,” *IEEE CVPR*, 2018.
- [3] L. M. Bergasa, et al., “Real-time driver monitoring based on steering/lane behavior,” *IEEE IV Symposium*, 2019.
- [4] S. Abtahi, et al., “Driver drowsiness detection using eye-blink patterns,” *Proc. ICPR*, 2018.
- [5] A. Patil, et al., “CNN-based eye closure detection for driver fatigue,” *IEEE Access*, 2021.
- [6] R. Kumar, et al., “IoT-enabled vehicle safety and GSM-based alerting,” *IEEE ICCE*, 2020.
- [7] P. Singh, et al., “Bluetooth and cloud-integrated real-time vehicle monitoring,” *IEEE ICAC*, 2021.