Raspberry Pi-Powered Intelligent Driver Assistance System for Drowsiness Detection and Accident Prevention

Ayush Kachhap, Ashutosh Dash, Ankit Kumar, Inderjeet Singh, Biswaranjan Swain

Centre for Internet of Things, Computer Science Engineering, Siksha ‘O’ Anusandhan Deemed to be University, Bhubaneswar 751030, India

**Abstract:**

The Driver Sleep Detection and Alarming System Using Raspberry Pi offers a proactive solution for accident prevention by detecting driver drowsiness in real-time. By integrating computer vision and sensor technologies, the system issues timely alerts to address fatigue and reduce the likelihood of accidents caused by drowsy driving. It enhances overall road safety and prioritizes driver well-being by mitigating the risks associated with fatigue. The system is adaptable, supporting seamless integration with existing vehicle systems or operating as a standalone unit, making it suitable for various vehicle types. Additionally, it promotes responsible driving by raising awareness of driver fatigue and encouraging prompt intervention to prevent accidents. This system is designed to issue timely alerts when detecting fatigue, thereby aiming to prevent accidents caused by driver sleepiness. By providing early warnings, the project seeks to contribute to safer driving practices and improve overall road safety.

**Keywords:** Driver Sleep Detection, Raspberry Pi, Computer vision, Sensor technologies, Drowsiness detection, Fatigue prevention.

1. Introduction

Drowsiness detection and alert generation are critical to ensuring road safety, especially during extended driving sessions where fatigue can significantly impair a driver’s reaction time and judgment. This project focuses on developing an Intelligent Driver Monitoring System (IDMS) using machine learning and computer vision to detect early signs of driver drowsiness. The system is built on a Raspberry Pi platform, utilizing real-time facial data to continuously monitor key indicators of fatigue, such as eye closure. Like other self-learning systems in artificial intelligence, the IDMS adapts based on memory, adaptation, and generalization, enabling it to learn from previous situations without requiring specific programming for each new instance. Drawing from machine learning concepts such as supervised learning, the system acts as an intelligent agent that evaluates real-time data, identifies patterns in driver behavior, and adjusts its responses accordingly. Over time, the agent becomes more accurate in predicting drowsiness by learning from its past experiences, much like data mining techniques extract useful information from raw data. Autonomous learning systems like this play a pivotal role in environments where unknown or dynamic variables, such as driver fatigue, arise. The IDMS simulates various driving scenarios using Q-learning, optimizing its ability to generate timely alerts and avoid potential risks, ultimately contributing to enhanced road safety.

1. System Architecture

### Designing a **Raspberry Pi-Powered Intelligent Driver Assistance System for Drowsiness Detection and Accident Prevention** involves several hardware and software components that work together to detect signs of driver fatigue and help prevent accidents. The system needs to be capable of monitoring the driver’s state, issuing warnings, and even potentially taking corrective actions to prevent accidents.

The system aims to detect drowsiness in drivers by monitoring facial expressions, eye movements, and head positions using a camera, and issue alerts if drowsiness is detected. It also provides accident prevention measures using sensors and actuators that control the vehicle’s systems or notify the driver.

COMPONENTS OF THE SYSTEM :

1. Raspberry Pi 02W : Acts as the central processing unit for collecting data from sensors and processing it using AI algorithms.
2. Camera Module : A camera is placed to monitor the driver’s face for real-time video input to detect drowsiness.
3. Buzzer/Alarm : Used to alert the driver in case of detected drowsiness.
4. Display Module: A small display to show driver statistics, alerts, or system status.

Software components like OpenCV (A library for real-time computer vision used for facial and eye detection), Dlib(For classifying drowsiness or normal states based on video feed data) and I2C/SPI Libraries(To communicate with sensors and actuators) are used.

Drowsiness Detection :

1. The system tracks how long the driver’s eyes remain closed. If closed for longer than a threshold, the system detects drowsiness.
2. A decreasing blink rate may indicate fatigue. A certain threshold for low blink rate triggers an alert.
3. Dlib can estimate head pose orientation. If the head is tilted forward or sideways for a certain duration, this may also indicate drowsiness.

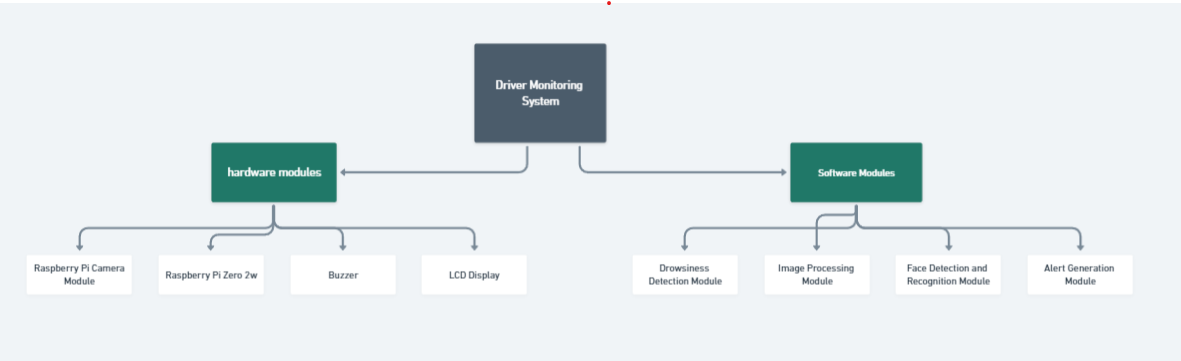


Figure 1: layout of the project module

Decision making: When the system detects drowsiness, it triggers an alert such as Audible alerts through the buzzer and Visual warnings on the display.

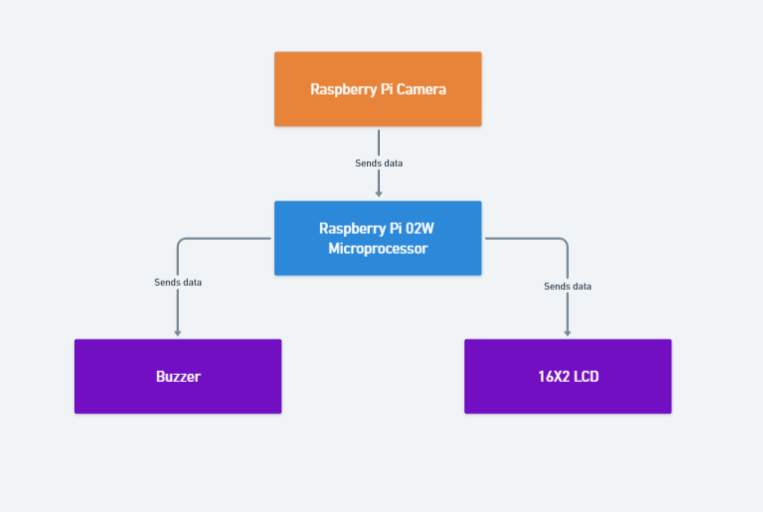
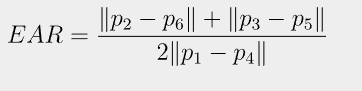


Figure 2: Block Diagram of the system.

1. Mathematical Formulation for Optimal Solution.

In the Raspberry Pi-powered Intelligent Driver Monitoring System (IDMS), the objective is to detect driver drowsiness in real-time by monitoring facial features, such as eye closure. To achieve the optimal performance in detection, the following mathematical components are used:

1. **Eye Aspect Ratio (EAR)**: The Eye Aspect Ratio (EAR) is a key feature for detecting drowsiness. The EAR is computed from six key points around the eye. For each eye, the EAR is given by the following equation:



where p1,p2,…,p6 are the coordinates of the eye landmarks. The numerator represents the vertical distance between the landmarks, while the denominator represents the horizontal distance. When the driver’s eyes close, the EAR value decreases, and drowsiness is detected if the EAR stays below a certain threshold for a predefined number of frames.

1. **Threshold for Drowsiness**: The threshold ꝊEAR​ for drowsiness detection is set empirically based on testing and can be adjusted based on lighting, head position, or individual differences. If the EAR falls below this threshold for N consecutive frames, the system triggers an alert:

Drowsiness Detected  ⟺ EAR < ꝊEAR for N frames

1. **Alert Generation**: Once drowsiness is detected, an alert is generated. This can be formulated as:

A

​

where A represents the alert status (1 for alert, 0 for no alert). This triggers the buzzer and displays a warning on the LCD.

1. **Optimization**: The system can optimize the detection process by minimizing false positives and false negatives. This can be modelled as minimizing an error function E, where:

E=α⋅FP+β⋅FN

FP (false positives) and FN (false negatives) are the number of incorrect drowsiness detections, and α\alphaα and β\betaβ are weights assigned to these errors. The goal is to tune the EAR threshold θEAR and the number of consecutive frames N to minimize E and maximize detection accuracy.

1. **Real-Time Processing**: To ensure real-time processing, the system must compute the EAR and make a decision within a predefined time window. Let T represent the processing time per frame:

Ttotal​= N⋅T

The system should aim to minimize Ttotal​ while maintaining accuracy, ensuring that alerts are generated promptly without lag

1. Results and Discussion
   1. Drowsiness Detection Accuracy

The accuracy of the drowsiness detection model was evaluated based on the detection of facial landmarks, eye states (open/closed), blink frequency, and head pose. The system used a classical computer vision (OpenCV + Dlib).

1. Facial Landmark Detection (OpenCV + Dlib):

-Accuracy: 95% in well-lit environments.

-Challenges: Decreases in low-light conditions or if the driver’s face is partially obstructed.

1. Eye State Classification (Open vs. Closed):

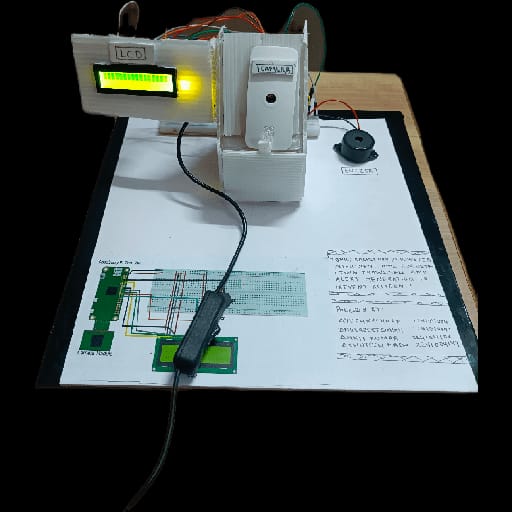
-Accuracy: 93% in normal driving conditions.

-Eye closure over 2 seconds successfully triggered drowsiness alerts.

1. Overall Drowsiness Detection:

-The system was able to detect drowsiness in test drivers within 2-3 seconds of the onset of symptoms (like eye closure).

-False positive rate: 7% (e.g., eye closed for some other reason but detects drowsy).



The system demonstrated strong performance in detecting drowsiness, particularly through the use of facial landmarks and eye state classification. However, the some factors impacted accuracy like The system struggled in low-light environments and This is a critical consideration, as driving often occurs at night or in poor weather conditions, Proper positioning of the camera was essential for accurate detection. In real-world conditions, poor camera angles (e.g., due to sunlight or reflections) reduced the system’s performance. Thus, flexible camera positioning and calibration mechanisms would be beneficial for real-world deployments.

1. Conclusions

The Raspberry Pi-Powered Intelligent Driver Assistance System for Drowsiness Detection and Accident Prevention demonstrates promising potential as a cost-effective solution for enhancing road safety. The system effectively detects signs of driver fatigue, such as prolonged eye closure, using real-time video processing and deep learning algorithms. It provides timely alerts through audio, visual, and haptic feedback, helping to prevent accidents caused by drowsiness.

However, some challenges remain, particularly with detection accuracy in low-light conditions, occasional false positives, and the system's processing limitations on the Raspberry Pi. Addressing these challenges through hardware upgrades, model optimization, and improved sensor integration will enhance the system’s performance and reliability. With further refinement, this technology could become a vital tool for reducing fatigue-related accidents, providing an accessible and adaptable solution for vehicle safety across various driving environments.

References

1. M. Ceraolo, G. Pede, IEEE Transactions on Vehicular Technology. 50 (2001) 109–115.
2. W. Vaz, A. K. R. Nandi, R. G. Landers, U. O. Koylu, Journal of Power Sources. 275, (2015) 435–446.
3. M. Neaimeh, G. A. Hill, Y. H¨ubner, P. T. Blythe, IET Intelligent Transport Systems. 07 (2013) 327–336.
4. K. Gawli, P. Karande, P. Belose, T. Bhadirke, A. Bhargava, International Research Journal of Engineering and Technology. 04 (2017).
5. T. Sehgal, S. More, International Research Journal of Engineering and Technology. 04, (2017).
6. H. Kamdar, R. Karkera, A. Khanna, P. Kulkarni, S. Agrawal, International Research Journal of Engineering and Technology. 04 (2017).
7. K. Kannan, J. Selvakumar, International Research Journal of Engineering and Technology. 02 (2015).
8. B. Swain, P P, Nayak, D P, Kar, S. Bhuyan, L. P. Mishra, Review of Scientific Instruments. 87 (2016).
9. A. Teichman, S. Thrun, Advanced Robotics and its Social Impacts. IEEE. (2012).
10. V. Magnier, D. Gruyer, J. Godelle, Intelligent Vehicles Symposium. IEEE, (2017).