JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Real-Time Driver Distraction Detection with Alarm System

Rekha D[1], Chaithra B[2], Deeksha N[3], Ashritha S[4], Poojitha K[5]

[5] ***Assistant Professor*** *Department of Information Science and Engineering, BGS College of Engineering*

[1 2 3 4] *Department of Information Science and Engineering, BGS College of Engineering and Technology, Bangalore, Karnataka, India*

Abstract : Driver distraction is a major contributor to road accidents, particularly due to mobile phone use and fatigue. Ensuring driver attentiveness is therefore critical for road safety. This project presents a Real-Time Driver Distraction Detection and Alert System that uses computer vision to monitor facial expressions, eye movements, blinking patterns, and head orientation from a live camera feed. The system identifies signs of drowsiness, phone usage, or looking away from the road and provides instant audio-visual alerts to help the driver regain focus and prevent accidents. Designed to be low-cost and non-intrusive, the system can be easily installed in existing vehicles without significant modifications. It also supports the development of intelligent transportation systems, with potential future enhancements including IoT integration, adaptive alert mechanisms, and partial automation to further improve road safety.

Keywords--{Yolo8; Mediapipe, Heuristics, Pygame, Driver monitoring, Driver distraction, Real-time detection, YOLOv8, MediaPipe, Heuristic analysis, Eye aspect ratio, Head pose estimation, Alert system, Vehicle safety}

I. Introduction

Road accidents continue to pose a serious global concern, causing millions of fatalities and substantial economic and social impacts each year. Among the various contributing factors, driver distraction has become one of the most frequent and hazardous, often stemming from mobile phone use, fatigue, or lapses in attention. Research shows that even short periods of inattention can greatly raise the risk of accidents, showing the need to watch driver behavior for road safety.

Ensuring driver attentiveness plays a crucial role in maintaining road safety. Although conventional safety tools such as seat belts, airbags, and traffic regulations help lessen the severity of accidents, they often cannot prevent them from happening in the first place. With the rapid progress in computer vision and intelligent transportation technologies, it is now possible to adopt preventive systems that can anticipate and reduce such risks.

This project develops a Real-Time Driver Distraction Detection and Alert System that uses a live camera to monitor the driver's face and head orientation. Through machine learning and image-processing techniques, the system can recognize early signs of inattention, including long eye closures, excessive blinking, looking away from the roadway, or operating a mobile phone.

Whenever these behaviors are detected, the system generates immediate sound and visual warnings to prompt the driver to regain focus. Designed to be low-cost, user-friendly, and non-intrusive, it can be easily incorporated into existing vehicles. In addition to enhancing individual safety, the system contributes to building smarter transportation networks aimed at preventing accidents and promoting safer driving practices.

2. Literature Survey

The proposed system [1] uses machine learning techniques to detect driver drowsiness in real time through continuous facial analysis. It focuses on identifying signs of fatigue and distraction with high accuracy by monitoring metrics such as the Percentage of Eye Closure (PERCLOS), Eye Aspect Ratio (EAR), and blink rate. Using dlib and OpenCV for facial landmark detection, the system runs on a Raspberry Pi with a Pi camera and provides audio-visual alerts when fatigue is detected. It is designed to be affordable and non-intrusive, but some challenges remain — including reduced accuracy when drivers wear masks or glasses, poor performance in low-light conditions, and limited processing power of the Raspberry Pi. The testing was also restricted to only 60 participants. Future improvements may involve infrared lighting, integration with vehicle telemetry, hybrid models, and large-scale real-world testing to increase reliability.

The study [2] presents a real-time facial analytics system based on event cameras (neuromorphic vision sensors), which detect brightness changes asynchronously. These cameras are ideal for edge computing because they offer high speed, low delay, and low energy use. The system applies a two-stage deep learning pipeline: first, a modified YOLOv3-tiny with RCNN for quick face and eye detection, and second, a Multi-Task Network for estimating head pose, eye gaze, and facial occlusions. Using Leaky Time Surfaces and ROI-based triggering, it processes only meaningful changes in the data stream. However, because real-world event-based datasets are scarce, the use of synthetic data for training reduces accuracy on real datasets like BIWI. The system struggles with low-motion, low-light, and occluded faces. To overcome this, combining event cameras with RGB or depth sensors is suggested.

In the research [3], a real-time automotive safety system is introduced to detect both driver distraction and drowsiness. It runs efficiently on edge devices using a hybrid deep learning setup that merges MediaPipe for facial landmark extraction with lightweight CNNs like BlazeFace and MobileNet-V1/V2. It calculates safety indicators such as EAR, Eyes-Off-Road (EOR), and Head Pose Estimation (HPE) through the PnP algorithm. The system achieves up to 45 FPS performance and supports infrared cameras for night-time monitoring. Alerts are given via audio signals and seat vibrations to keep the driver alert. Despite these strengths, challenges include complex software dependencies, thermal issues, and sensitivity to camera calibration, along with a lack of large-scale validation and vehicle telemetry integration.

The approach [4], known as TUN-DAS, adopts time-series analysis and unsupervised learning to evaluate driving behavior without needing labeled data, making it both scalable and adaptive. It applies k-means clustering and a modified Dynamic Time Warping (DTW) algorithm to process driving signals such as glance behavior, acceleration, and speed. Using DTW Barycenter Averaging (DBA), it improves pattern interpretability. Built within a Driver-Vehicle-Environment (DVE) framework, it identifies both global and local anomalies while offering personalized driver insights. However, it mainly focuses on right-lane-change scenarios, lacks feature integration, and has limited dataset testing. Future research could expand it with mobile app validation, environmental factor analysis, and real-time deployment.

The study [5] introduces a hybrid computer vision and machine learning approach for driver drowsiness detection. It uses Viola-Jones for facial detection, SVM classifiers for drowsiness state recognition, and k-means clustering for feature grouping. The system includes yawning detection and Sobel operator-based eye analysis to track fatigue. To improve lighting robustness, it applies skin segmentation through the YCbCr color model combined with binary pattern recognition. The model achieves 100% accuracy in facial segmentation and 83.25% accuracy in emotion and gesture recognition, working without any wearable sensors. Yet, its performance decreases in low-light or long-distance conditions, and eye-state detection is less reliable than yawn detection. Future directions involve integrating EEG or ECG signals, infrared cameras, personalization features, and deep learning methods for better accuracy and adaptability.

The system [6] combines AI and IoT for a real-time, low-cost drowsiness detection system aimed at improving road safety. It employs MediaPipe Face Mesh for precise facial tracking and a CNN-based eye state classifier, running efficiently on a Raspberry Pi 4B with a NoIR camera, which supports dim-light operation. OpenCV handles real-time image processing, while IoT connectivity allows remote monitoring and alerts via seat vibrations and voice prompts. The system achieves a high AUROC score of 0.9788 and includes a speed control mechanism to help prevent accidents. However, it remains sensitive to camera position, lighting changes, and fixed EAR thresholds that limit personalization. Testing was also limited to 17 participants, and no multi-sensor fusion was used. Future work aims at adaptive thresholds, larger datasets, and extended real-world testing.

The research [7] applies Vision Transformers (ViT) for detecting driver drowsiness in real time. The model splits facial images into small patches and processes them through transformer encoders with multi-head self-attention to classify drivers as alert or drowsy. Using dlib for face detection, it is trained on datasets like NTHU-DDD and UTA-RLDD, reaching up to 99.4% accuracy. Implemented on a Raspberry Pi 4B with an IR camera, GSM/GPS modules, and an alert buzzer, it performs well even under partial occlusions. Still, its effectiveness drops when the face is mostly hidden and relies heavily on GPS or network stability. It also lacks temporal sequence modeling and diverse demographic testing. Future upgrades could integrate temporal analysis, multi-sensor fusion, and personalized driver profiles to boost adaptability.

Finally, the framework [8], named D3-CDLM, merges traditional machine learning with deep learning to achieve high efficiency and accuracy. It extracts facial features using Histogram of Oriented Gradients (HOG), reduces them via Principal Component Analysis (PCA), and classifies them using algorithms such as SVM, Decision Tree, Random Forest, and XGBoost. Parallelly, a custom 30-layer CNN processes raw facial images, and the two outputs are fused to leverage both classical and deep learning strengths. This hybrid setup achieves up to 99.6% accuracy, handling lighting changes and head pose variations effectively. However, it depends on a single dataset, struggles with occluded faces, and requires high computational power. Future improvements could add multi-sensor inputs, real-world validation, and adaptive personalized models for detecting a wider range of risky driving behaviors.

Figure 1: Comparison of Accuracy, Precision, Specificity, and Score Across Different Papers

	accuracy	precision	specificity	f-score
paper-1	98.99%	98.62%	99.44%	99.03%
paper-2	91.04%	91.26%	89.75%	91.15%
paper-3	96.60%	95.90%	96.20%	96.22%
paper-4	97.12%	96.85%	96.40%	96.80%
paper-5	95.70%	94.80%	95.20%	95.00%
paper-6	97.80%	97.10%	96.90%	97.40%
paper-7	95.02%	94.11%	92.30%	96.10%
paper-8	91.11%	93.70%	94.90%	93.45%

Figure 2: Performance comparison of existing research papers on real-time driver distraction detection with an alarm system.

The table shows performance metrics of eight papers, where Paper-1 and Paper-6 achieve the best overall results with accuracy and F-scores near 99% and 97% respectively. Paper-3 and 4 also perform strongly with balanced metrics above 96%. Among the reviewed studies, Paper 5 and Paper 7 demonstrate moderate performance levels, whereas Paper 2 and Paper 8 achieve comparatively lower results, with accuracy around 91% and reduced specificity.

3. Related Works and Concepts

3.1 YOLOv8

YOLOv8 represents the most recent advancement in the You Only Look Once (YOLO) family of object detection algorithms. It is designed to deliver fast and precise object recognition in both images and video streams. Unlike earlier YOLO versions that relied on multiple processing stages, YOLOv8 performs end-to-end detection in a single pass, making it particularly suitable for real-time applications such as driver monitoring and distraction detection.

The model divides each input image into a grid and simultaneously predicts both the object class and its bounding box coordinates within each grid cell. It employs an improved backbone network for feature extraction and incorporates advanced detection heads, enabling the identification of objects of varying scales, including smaller or partially visible features like eyes, faces, or mobile phones.

What distinguishes YOLOv8 is its ability to maintain an excellent balance between speed, accuracy, and computational efficiency. Through refined architecture, optimized training strategies, and lightweight implementation, the model delivers strong performance not only on high-end systems but also on embedded devices with limited processing power, such as those used in vehicles. Owing to these attributes, YOLOv8 has emerged as one of the most effective solutions for real-time driver monitoring, ensuring rapid and accurate detection of distractions or inattention behind the wheel.

3.2 MediaPipe

MediaPipe is an open-source framework developed by Google that enables real-time, cross-platform machine learning pipelinesplatform machine learning pipelines. It is particularly popular for human perception tasks such as face, hand, and body tracking. One of its strongest features is its ability to map facial landmarks with very high accuracy and low latency, even when running on low-power devices. In driver monitoring systems, MediaPipe's Face Mesh solution is especially valuable. It can extract 468 three-dimensional facial landmarks from a regular RGB camera feed without requiring any special sensors. These landmarks include eye contours, mouth shape, and head orientation, which are critical indicators of driver states such as fatigue, drowsiness, or distraction.

Furthermore, MediaPipe's flexibility and real-time processing capabilities make it an essential tool for building intelligent driver assistance systems. It can handle varying lighting conditions, partial occlusions, and different facial orientations, which are common challenges in real-world driving scenarios. For example, even when the driver is wearing spectacles, turning slightly, or when sunlight partially covers the face, MediaPipe's robust landmark detection continues to track key facial features accurately. In addition, MediaPipe supports cross-platform compatibility, allowing it to run efficiently on operating systems like Windows, Linux, Android, and iOS. This versatility enables developers to easily integrate it into a wide range of devices from high-performance onboard computers to low-cost embedded systems without major code modifications.

Another major advantage is its low-latency performance. In driver monitoring systems, immediate feedback is crucial to prevent potential accidents. MediaPipe offers real-time frame processing, enabling the system to quickly detect and respond to driver behaviors. This allows instant recognition of signs such as drowsiness, yawning, or looking away from the road, followed by the activation of alerts through connected interfaces like Pygame.

A key advantage of MediaPipe lies in its flexibility and scalability, allowing developers to integrate it easily with deep learning models or rule-based methods for improved detection reliability. For example, while MediaPipe efficiently extracts facial landmarks, models like YOLOv8 can be used to locate the face region, and heuristic algorithms can then analyze behaviors such as prolonged eye closure or downward head movement to identify distraction or fatigue.

Overall, MediaPipe's combination of speed, accuracy, adaptability, and integration capability makes it a highly suitable framework for real-time driver monitoring. It not only ensures consistent and precise tracking of facial features but also strengthens the overall performance, reliability, and safety of intelligent transportation systems.

3.3 Heuristics

Heuristics are rule-based techniques that provide quick and efficient decision-making without relying on large datasets or complex computations. In driver monitoring systems, heuristics are especially valuable because they enable real-time fatigue and distraction detection on resource-constrained devices such as embedded systems. Instead of heavy machine learning pipelines, these methods depend on geometric relationships between facial landmarks, making them lightweight yet practical.This ratio measures how open or closed the eyes are by calculating distances between six key landmarks around the eye: two at the corners and four along the eyelids. The EAR is defined as:

$$EAR = \frac{||p_2 - p_6|| + ||p_3 - p_5||}{2 \times ||p_1 - p_4||}$$

Here, $||p_i - p_j||$ represents the Euclidean distance between two landmark points. When the eyes are fully open, the vertical distances between key eye landmarks are relatively large, resulting in a higher Eye Aspect Ratio (EAR). As the eyes start to close, these distances decrease, causing the EAR value to drop. When the EAR remains below a certain threshold—commonly around 0.25—for several consecutive frames, the system interprets it as prolonged eye closure, indicating driver drowsiness. This method also effectively differentiates short blinks from longer closures associated with fatigue.

In addition to monitoring eye closure, head pose estimation serves as another important heuristic for identifying driver distraction. By calculating the orientation of the driver's head, the system can infer whether the driver is looking away from the road. For instance, a consistent downward head tilt might suggest mobile phone use, whereas sideward movements could indicate loss of attention. Specific angle thresholds are defined to determine when the driver is no longer focused on driving.

The main advantage of using heuristic-based methods lies in their speed and computational efficiency. They can operate in real time with minimal hardware requirements, making them ideal for embedded automotive systems such as those running on Raspberry Pi or similar platforms. However, these methods also have limitations—variations in facial structure, lighting conditions, and camera positioning can reduce detection accuracy, and fixed threshold values may not generalize well across all drivers, affecting system adaptability and reliability.

3.4 Pygame

Pygame is a Python library mainly used for building interactive applications like games and simulations. In the real-time driver distraction detection project, Pygame plays a key role in managing both visual and audio alerts to keep the driver attentive. When the system detects signs of drowsiness or distraction—for instance, if the driver's eyes remain closed for several seconds or their head turns away from the road—Pygame triggers warnings such as beep sounds or on-screen messages. These alerts are designed to immediately capture the driver's attention and help prevent accidents.

Pygame also enables continuous interaction between the detection modules and the user interface. By running an event loop, it constantly updates the system's response based on inputs from YOLOv8, MediaPipe, and heuristic algorithms, ensuring smooth and real-time coordination between detection and alert generation.

Another advantage of Pygame is its lightweight and efficient design, making it suitable for embedded systems like a Raspberry Pi or in-vehicle processors. It can easily manage simple graphical displays—such as “Driver Drowsy” or “Stay Alert”—alongside the live video feed without affecting system performance.

Overall, Pygame serves as an effective communication bridge between the detection system and the driver, ensuring that alerts are clear, immediate, and easy to understand, thereby enhancing the usability and effectiveness of the driver monitoring system.

4. Methodology

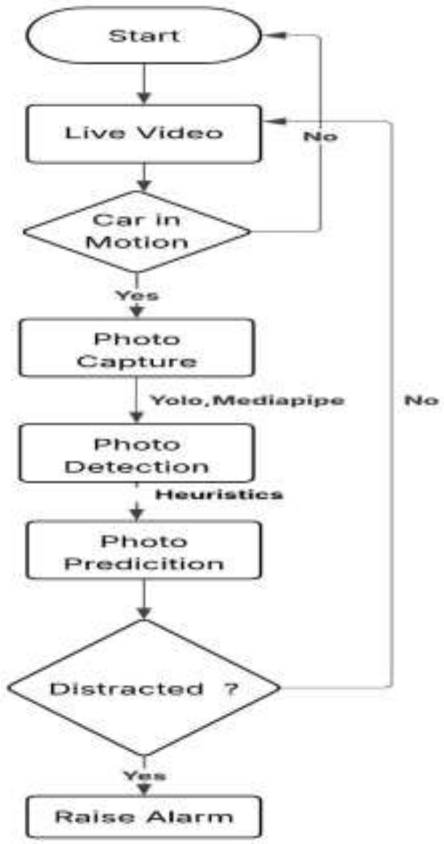

Figure 3: Flowchart of the proposed distraction detection system.

The proposed distraction detection framework operates in a sequence of well-defined steps, starting from video input and ending with an alarm in case of distraction. The workflow is explained below in detail:

4.1 System Initialization (Start)

The system begins by setting up all necessary components to get ready for operation. This step involves switching on the camera, loading the pre-trained YOLO and MediaPipe models, and preparing the environment to handle live video input. By completing these preparations in advance, the system ensures that it can start processing video frames instantly once activated, without any noticeable delay.

4.2 Continuous Video Monitoring (Live Video)

The system constantly monitors the driver using a live video feed from the camera. This video stream is the main source of information for detecting the driver’s face, eyes, and actions. Instead of saving the entire video, the system captures only certain frames at intervals. This makes the process faster and reduces unnecessary storage and processing.

4.3 Frame Extraction (Photo Capture).

Frames are captured only when the vehicle is moving. This ensures that image processing happens only when it is needed, avoiding extra work when the car is parked or waiting in traffic. Each frame shows the driver’s current state and helps the system check for signs of sleepiness or distraction. By focusing only on moving conditions, the system works faster and responds in real time when it truly matters for safety.

4.4 Feature Detection (Photo Detection with YOLO and MediaPipe)

Each frame is analyzed using two tools:

YOLO (You Only Look Once): Detects objects like mobile phones or steering wheels that may indicate distraction.

MediaPipe: Tracks facial points such as eyes, head, and hands to understand the driver's focus and movements. Using both together helps the system accurately detect both external objects and the driver's behavior in real time.

4.5 Behaviour Prediction (Photo Prediction using Heuristics)

After detection, the results are checked using simple rules. For example:If the driver's eyes stay closed for a few seconds, it may mean drowsiness.If the head is turned away or facing down, it suggests distraction.If a phone is held near the face, it indicates phone usage.
These checks help the system turn raw data into clear, meaningful actions.

4.6 Decision-Making (Distraction Check)

The system then decides whether the driver is distracted or not. If yes, it immediately prepares to alert the driver. If no, it continues to monitor as usual. This continuous process keeps the system active and attentive at all times.

4.7 Alert Generation (Raise Alarm)

When distraction or drowsiness is detected, the system quickly alerts the driver. It may use a buzzer sound or a visual warning on the dashboard to grab attention and help the driver refocus on the road.

4.8 Continuous Feedback Loop

After sending an alert, the system keeps watching the driver without any pause. Each new frame is analyzed instantly to maintain constant safety. This continuous cycle ensures real-time accuracy, efficient use of resources, and consistent driver awareness throughout the journey.

5. System Development

The proposed system is designed to monitor the driver's state in real time and provide instant alerts whenever signs of distraction or drowsiness are detected. The system captures a live video feed from a camera placed in front of the driver and processes it continuously while the vehicle is in motion.
Unlike other research works that depend on separate datasets for training, our project uses pre-trained algorithms such as YOLOv8 and MediaPipe, which already include large, well-trained models. This allows the system to analyze live input directly without the need for additional training data, making it more efficient and practical for real-world use.
The integration of these models helps the system recognize both objects (like mobile phones or steering wheels) and human features (like eyes and head movements) accurately. Heuristic rules are then applied to interpret these detections — for example, if the eyes remain closed for a few seconds or the driver's head is turned away, the system immediately raises an alert through sound or visual signals.
This real-time, camera-based approach ensures that the system responds instantly to risky driving behavior, improving road safety without requiring complex hardware or extra datasets.

5.1 Data Acquisition and Preprocessing

Video data is continuously captured from a camera placed in front of the driver, but only while the vehicle is moving. The captured frames are pre-processed to enhance visibility and detection accuracy. Steps such as grayscale conversion, noise reduction, and frame normalization are applied to make the input consistent and improve computational efficiency. This preprocessing helps the models operate smoothly and maintain stable real-time performance.

5.2 Face and Landmark Detection using MediaPipe

The MediaPipe Face Mesh model is used for identifying facial landmarks, providing accurate detection of key regions such as the eyes, nose, and mouth. These points are essential for tracking the driver's alertness by monitoring eye openness, blinking rate, and mouth movements.
Using these features, the system computes measures like the Eye Aspect Ratio (EAR) and estimates head pose orientation to detect signs of fatigue, yawning, or distraction when the driver looks away from the road.

5.3 Distraction Detection using YOLOv8

For identifying external distractions, the system incorporates YOLOv8, a deep learning model known for fast and precise object detection. It detects objects such as mobile phones, bottles, or other handheld devices that may indicate distracted driving. This allows the system to go beyond facial analysis and capture broader behavioral cues that contribute to driver inattention.

5.4 Heuristic-based Behavioral Analysis

Alongside the deep learning models, heuristic rules are used to interpret behavioral patterns more effectively. Some examples include:

EAR Threshold: If the eye aspect ratio stays below a set limit for a continuous period, the system flags possible drowsiness.
Head Movement: Quick or prolonged deviation in head direction suggests distraction or loss of focus.
These rule-based checks complement the learning models and ensure consistent performance even under changing lighting conditions or partial occlusions, such as glasses or masks.

5.5 Alert Mechanism and Integration

When the system identifies drowsiness or distraction, it immediately triggers audio and visual alerts to prompt the driver's response.
The implementation is carried out in Python, using libraries such as OpenCV, MediaPipe, and Ultralytics YOLOv8 for detection and tracking, while heuristic logic manages real-time decision-making. The modular architecture also allows for future integration with vehicle sensors or IoT platforms to support remote monitoring.

6. Results and Analysis

The real-time driver distraction detection system was developed and tested successfully to observe the driver's behavior through a live video feed. The program continuously monitored facial features such as the eyes, head position, and face orientation to identify any signs of inattention or drowsiness. During testing, the system effectively detected conditions like prolonged eye closure (more than three seconds), frequent blinking, and head movement away from the road. When such behaviors occurred, the Pygame module immediately triggered a beep sound, alerting the driver to stay attentive.
The system processed video frames smoothly and responded quickly, ensuring that alerts were generated in real time without noticeable delay. It was also designed to capture frames only when the vehicle was moving, which helped reduce unnecessary processing when the car was stationary. This approach improved the system's overall efficiency and made it suitable for continuous monitoring.
Throughout testing, the system performed reliably in normal lighting conditions and handled moderate head movements well. The combination of YOLOv8 for face detection, MediaPipe for facial landmark tracking, and heuristic methods for behavior analysis worked effectively together to identify driver distraction. Overall, the project achieved its goal of providing a practical, real-time solution for monitoring driver alertness. It demonstrated that even with limited hardware, such as embedded systems or standard laptops, accurate and timely detection can be achieved to enhance road safety.
When The Vehicle is Stationary:
The driver monitoring system is designed to observe and analyze various driver behaviors that may indicate distraction, fatigue, or inattention. These behaviors include using a mobile phone, turning the head away from the road, closing the eyes for extended periods, or bending forward and looking down. By monitoring these actions in real time, the system can detect potentially unsafe situations before they lead to accidents.
To prevent unnecessary alerts, the system is programmed to recognize the vehicle's state. When the vehicle is stationary, any detected behaviors—such as looking at a phone, checking the side, or momentarily closing eyes—do not trigger alarms. This approach avoids false warnings while still maintaining readiness to respond when the vehicle starts moving.
Overall, the system balances safety and practicality by continuously monitoring driver attention while reducing distractions from unnecessary alerts. It acts as an intelligent assistant, alerting drivers only when there is a real risk, ensuring both vigilance and comfort during driving.

Figure 4: Driver using phone, but no alert since the vehicle is stationary.

Figure 5: Driver turning head to the right, but no alarm as detection is paused during the stationary state.

Figure 6: Driver turning head to the left, but no alert triggered because the vehicle is not in motion.

Figure 7: Driver's eyes closed, but no alarm triggered since the vehicle is stationary.

When The Vehicle is in Movement:

Figure 8: Alert triggered as driver's eyes remain closed while vehicle is moving.

If the driver's eyes remain closed for more than 3 seconds while the vehicle is moving, the system detects possible drowsiness and instantly triggers an alarm to alert the driver and prevent accidents.

Figure 9: Alert triggered as driver turns left while distracted.

When the driver turns left and remains distracted for over 3 seconds, the system identifies unsafe behavior. An alert is activated to remind the driver to stay focused and maintain proper lane control.

Figure 10: Alert triggered as driver bends down while vehicle is in motion.

If the driver bends down or looks away from the road, the system detects this unsafe movement and activates an alarm to regain the driver's attention.

Figure 11: Alert triggered as driver turns right while distracted.

If the driver turns right and continues to be inattentive for more than 3 seconds, the system issues a warning alarm. This helps the driver regain focus and avoid potential road hazards.

Figure 12: Alert triggered as driver uses phone in left while driving.

Figure 13: Alert triggered as driver uses phone in right while driving.

The system detects if the driver uses a phone with the right or left hand while driving and immediately alerts them, encouraging full attention on the road.

7.Limitations

While the system is effective in detecting driver drowsiness and distraction, it has some inherent limitations. Its accuracy can be impacted by challenging lighting conditions, such as very low light at night or harsh sunlight, which may make facial feature detection less reliable. Accessories like sunglasses, masks, or hats can partially obstruct key landmarks, affecting detection precision.

Extremely brief or subtle inattentive actions might not be captured, as the system is designed to respond to sustained patterns of distraction. The system’s real-time performance is also dependent on hardware capabilities. On lower-end or resource-constrained devices, processing delays may occur, potentially affecting timely alerts. Lastly, environmental factors such as vibrations in the vehicle or sudden camera shifts can momentarily reduce detection efficiency.

8. Future Work and Enhancement

The current system provides real-time driver monitoring, but several enhancements can be made in the future. One key improvement could be integrating an emergency response feature, allowing the system to automatically contact services like an ambulance in case of a critical event or accident.

Additional enhancements may include incorporating extra sensors, such as eye trackers or vehicle telemetry, to better assess driver state and behavior. Adaptive algorithms could personalize alerts based on individual habits, reducing unnecessary warnings and improving usability.The system could also be optimized for low-power embedded devices to ensure smooth operation in vehicles and support cloud-based analytics for long-term monitoring and data-driven insights. Expanding these features would make the system more robust, intelligent, and capable of providing timely assistance in emergencies.

9. References

[1]. A. Ziryawulawo, M. Kirabo, C. Mwikirize, J. Serugunda, E. Mugume and S. P. Miyingo, "Machine learning based driver monitoring system: A case study for the Kayoola EVS," in SAIEE Africa Research Journal, vol. 114, no. 2, pp. 40-48, June 2023, doi: 10.23919/SAIEE.2023.10071976.

[2]. H. Hayashi, N. Oka, S. Sugano and M. Kamezaki, "TUN-DAS: Time-Series Analysis and Unsupervised Learning Based Driving Behavior Assessment System," in IEEE Transactions on Intelligent Transportation Systems, vol. 26, no. 1, pp. 604-619, Jan. 2025, doi: 10.1109/TITS.2024.3484452**.**

[3]. C. Ryan et al., "Real-Time Multi-Task Facial Analytics With Event Cameras," in IEEE Access, vol. 11, pp. 76964-76976, 2023, doi: 10.1109/ACCESS.2023.3297500.

[4]. M. Ramzan, A. Abid, M. Fayyaz, T. J. Alahmadi, H. Nobanee and A. Rehman, "A Novel Hybrid Approach for Driver Drowsiness Detection Using a Custom Deep Learning Model," in IEEE Access, vol. 12, pp. 126866-126884, 2024, doi: 10.1109/ACCESS.2024.3438617

[5]. L. Sharara et al., "A Real-Time Automotive Safety System Based on Advanced AI Facial Detection Algorithms," in IEEE Transactions on Intelligent Vehicles, vol. 9, no. 6, pp. 5080-5100, June 2024, doi: 10.1109/TIV.2023.3272304.

[6]. A. Altameem, A. Kumar, R. C. Poonia, S. Kumar and A. K. J. Saudagar, "Early Identification and Detection of Driver Drowsiness by Hybrid Machine Learning," in IEEE Access, vol. 9, pp. 162805-162819, 2021, doi: 10.1109/ACCESS.2021.3131601.

[7]. A. Jarndal, H. Tawfik, A. I. Siam, I. Alsyouf and A. Cheaitou, "A Real-Time Vision Transformers-Based System for Enhanced Driver Drowsiness Detection and Vehicle Safety," in IEEE Access, vol. 13, pp. 1790-1803, 2025, doi: 10.1109/ACCESS.2024.3522111.

[8]. S. P. Soman, G. Senthil Kumar and A. K M, "Internet-of-Things-Assisted Artificial Intelligence-Enabled Drowsiness Detection Framework," in IEEE Sensors Letters, vol. 7, no. 7, pp. 1-4, July 2023, Art no. 6003704, doi: 10.1109/LSENS.2023.3289143.

[9]. Q. Wu, N. Li, L. Zhang and F. Richard Yu, "Driver Drowsiness Detection Based on Joint Human Face and Facial Landmark Localization With Cheap Operations," in IEEE Transactions on Intelligent Transportation Systems, vol. 25, no. 12, pp. 19633-19645, Dec. 2024, doi: 10.1109/TITS.2024.3443832.

[10]. J. Bai et al., "Two-Stream Spatial–Temporal Graph Convolutional Networks for Driver Drowsiness Detection," in IEEE Transactions on Cybernetics, vol. 52, no. 12, pp. 13821-13833, Dec. 2022, doi: 10.1109/TCYB.2021.3110813.

[11]. S. A. El-Nabi et al., "Driver Drowsiness Detection Using Swin Transformer and Diffusion Models for Robust Image Denoising," in IEEE Access, vol. 13, pp. 71880-71907, 2025, doi: 10.1109/ACCESS.2025.3561717.

[12]. L. Yang, H. Yang, H. Wei, Z. Hu and C. Lv, "Video-Based Driver Drowsiness Detection With Optimised Utilization of Key Facial Features," in IEEE Transactions on Intelligent Transportation Systems, vol. 25, no. 7, pp. 6938-6950, July 2024, doi: 10.1109/TITS.2023.3346054.

[13]. H. A. Madni, A. Raza, R. Sehar, N. Thalji and L. Abualigah, "Novel Transfer Learning Approach for Driver Drowsiness Detection Using Eye Movement Behavior," in IEEE Access, vol. 12, pp. 64765-64778, 2024, doi: 10.1109/ACCESS.2024.3392640.

[14]. B. Fu, F. Boutros, C. -T. Lin and N. Damer, "A Survey on Drowsiness Detection: Modern Applications and Methods," in IEEE Transactions on Intelligent Vehicles, vol. 9, no. 11, pp. 7279-7300, Nov. 2024, doi: 10.1109/TIV.2024.3395889.

[15]. P. S. Lamba et al., "Detection of Driver Drowsiness Using Adaptive Eye Characteristic Ratio for Enhanced Road Safety," in IEEE Access, vol. 13, pp. 84313-84328, 2025, doi: 10.1109/ACCESS.2025.3566981.

[16]. E. Perkins, C. Sitaula, M. Burke and F. Marzbanrad, "Challenges of Driver Drowsiness Prediction: The Remaining Steps to Implementation," in IEEE Transactions on Intelligent Vehicles, vol. 8, no. 2, pp. 1319-1338, Feb. 2023, doi: 10.1109/TIV.2022.3224690.

[17]. D. Tan, W. Tian, C. Wang, L. Chen and L. Xiong, "Driver Distraction Behavior Recognition for Autonomous Driving: Approaches, Datasets and Challenges," in IEEE Transactions on Intelligent Vehicles, vol. 9, no. 12, pp. 8000-8026, Dec. 2024, doi: 10.1109/TIV.2024.3405990.

[18]. Y. Zhang, J. Lu, G. Xia and A. Khajepour, "Human–Machine Shared Control for Industrial Vehicles: A Personalized Driver Behavior Recognition and Authority Allocation Scheme," in IEEE Transactions on Intelligent Vehicles, vol. 9, no. 11, pp. 6869-6880, Nov. 2024, doi: 10.1109/TIV.2024.3389952.

[19]. S. Bouhsissin, N. Sael and F. Benabbou, "Driver Behavior Classification: A Systematic Literature Review," in IEEE Access, vol. 11, pp. 14128-14153, 2023, doi: 10.1109/ACCESS.2023.3243865.

[20]. H. Gao, M. Hu and Y. Liu, "Learning Driver-Irrelevant Features for Generalizable Driver Behavior Recognition," in IEEE Transactions on Intelligent Transportation Systems, vol. 25, no. 10, pp. 14115-14127, Oct. 2024, doi: 10.1109/TITS.2024.3396640.

[21]. W. Shariff, M. A. Farooq, J. Lemley, and P. Corcoran, ‘‘Event-based YOLOobject detection: Proof of concept for forward perception system,’’ in Proc.15th Int. Conf. Mach. Vis. (ICMV), vol. 12701. Bellingham, WA, USA:SPIE, Jun. 2023, pp. 74–80.

[22]. P. Liu, G. Chen, Z. Li, D. Clarke, Z. Liu, R. Zhang, and A. Knoll,‘‘NeuroDFD: Towards efficient driver face detection with neuromorphic vision sensor,’’ in Proc. Int. Conf. Adv. Robot. Mechatronics (ICARM),Jul. 2022, pp. 268–273.

[23] G. Moreira, A. Graça, B. Silva, P. Martins, and J. Batista, ‘‘Neuromorphicevent-based face identity recognition,’’ in Proc. 26th Int. Conf. Pattern Recognit. (ICPR), Aug. 2022, pp. 922–929.

[24]. Z. Zhang, K. Chai, H. Yu, R. Majaj, F. Walsh, E. Wang, U. Mahbub,H. Siegelmann, D. Kim, and T. Rahman, ‘‘Neuromophic high-frequency3D dancing pose estimation in dynamic environment,’’ in Proc. CVPR Workshop, 2023.

[25]. L. Berlincioni, L. Cultrera, C. Albisani, L. Cresti, A. Leonardo,S. Picchioni, F. Becattini, and A. D. Bimbo, ‘‘Neuromorphic event-based facial expression recognition,’’ 2023, arXiv:2304.06351.

[26]. Y. Wang, W. Zhou, and J. Zhou, ‘‘2DHeadPose: A simple and effective annotation method for the head pose in RGB