Drowsiness Detection Based on Eye Aspect Ratio and Head Pose Estimation with IoT Integration

Rama Amirul Ramadhan[1], Kurniawan D. Irianto[2*]

[1,2] Department of Informatics, Universitas Islam Indonesia, Yogyakarta, Indonesia
Jalan Kaliurang Km 14,5, Umbulmartani, Kec. Ngemplak, Sleman, Yogyakarta, Indonesia
Email: [1]22523127@students.uii.ac.id, [2*]k.d.irianto@uii.ac.id

Abstract−This study proposes a real-time driver drowsiness detection system integrating computer vision with Internet of Things (IoT) technology on an affordable embedded platform. The system uses a Camera Module 3 Wide NoIR connected to a Raspberry Pi 3 Model B+ to capture driver facial images. Two visual indicators are computed in parallel from 68 facial landmarks extracted using dlib: Eye Aspect Ratio (EAR) for detecting prolonged eye closure, and Head Pose Estimation using solvePnP for detecting head nodding. An OR-logic decision mechanism triggers an audio alarm when EAR falls below 0.25 for five consecutive frames or Pitch angle exceeds 15 degrees for ten consecutive frames. Events are classified as KANTUK_MATA, KANTUK_KEPALA, or KEDUANYA and sent to firebase Realtime Database for remote monitoring. *Black Box* testing with 11 scenarios confirms all core functions operate correctly. Average response times of 2.00 seconds via EAR and 2.70 seconds via Head Pose are within acceptable ranges for early drowsiness warning. The multi-indicator approach demonstrates that head nodding is detected earlier than eye closure in gradual drowsiness scenarios, providing earlier warning than single-indicator systems.

Keywords: Driver Drowsiness Detection; Eye Aspect Ratio; Head Pose Estimation; Raspberry Pi; IoT; Firebase

1. INTRODUCTION

Road safety is a global issue that remains a primary concern in many countries, including Indonesia. The World Health Organization (WHO) reports that traffic accidents cause over 1.19 million deaths annually worldwide and are a leading cause of death among the working-age population [1]. In Indonesia, the majority of traffic accidents are attributed to human factors, such as negligence, traffic violations, and suboptimal physical conditions of the driver [2]. Among these factors, driver drowsiness is often underestimated, despite its significant contribution to accident risk. Drowsy drivers tend to lose concentration, experience reduced alertness, and exhibit slower reaction times compared to normal conditions [3].

Driver drowsiness is generally caused by sleep deprivation, excessive workloads, long working hours, and long-distance travel without adequate breaks [15]. The risk is heightened during night driving, a time when the body is physiologically in a phase of reduced alertness [4], [5]. Research indicates that drowsiness-related accidents frequently occur between midnight and 5:00 AM, particularly on relatively quiet roads [6], [7]. The economic impact of such accidents cannot be overlooked either, encompassing medical costs, vehicle damage, and lost productivity that place a burden on the national economy [8].

Various efforts have been made to mitigate the risk of accidents caused by drowsiness, ranging from regulations on professional drivers' working hours to road safety campaigns. However, preventive, real-time, technology-based approaches are considered more effective, as they can provide immediate warnings to drivers before drowsiness escalates into a dangerous incident [2]. Advances in computer vision and embedded systems have opened up new opportunities to develop non-invasive drowsiness detection systems that do not require physical contact with the driver [8].

Several previous studies have examined driver drowsiness detection using various approaches.

Table 1. Previous Research

No	Authors	Summary
1	Shimpi et al.	Implementing an EAR-based system on a Raspberry Pi using a conventional webcam, albeit limited to a single indicator without IoT integration [12]
2	Florez et al.	Combining EAR and Mouth Aspect Ratio (MAR) on an embedded platform, but without Head Pose Estimation and without remote monitoring capabilities [13]
3	Chen et al.	Implementing multi-feature capabilities—including head pose—on a Raspberry Pi 4 using LSTM, yet requiring more expensive hardware and lacking integration with IoT cloud platforms [17].
4	Kukudkar et al.	Developing a drowsiness and seatbelt monitoring system for buses, but without cloud-based data logging [21]

5	R. Kaur and S. Bhatt	Facial landmark-based drowsiness detection and audio alarm, limited to non-cloud platforms [20]

A review of previous research identified three key gaps:

a. Most systems on affordable embedded platforms rely solely on EAR as an indicator and are unable to detect drowsiness manifested as head nodding.
b. IoT integration for remote monitoring and historical data logging remains very limited.
c. The potential of NoIR cameras on low-cost embedded platforms has not been extensively explored.

This research aims to address this gap by proposing a multi-indicator drowsiness detection system that combines Eye Aspect Ratio (EAR) and Head Pose Estimation on the affordable Raspberry Pi 3 Model B+, integrated with Firebase as an IoT platform for real-time monitoring and data logging. The system is expected to provide more comprehensive drowsiness detection than single-indicator systems while remaining implementable on a low-cost embedded platform. Specifically, this research makes three key contributions. First, it demonstrates that EAR and Head Pose Estimation can be combined within a single processing pipeline using the same 68 dlib facial landmarks, without requiring additional hardware or imposing a significant computational load on the Raspberry Pi 3B+. Second, it implements OR logic as a decision mechanism, enabling the system to independently detect two distinct manifestations of drowsiness—specifically, head nodding and eye closure. Third, it integrates the Firebase Realtime Database as an IoT platform to facilitate remote, real-time monitoring of the driver's condition, alongside historical logging that distinguishes the specific triggering indicator for each drowsiness event.

This study employs a prototyping methodology that enables the empirical calibration of system parameters through iterative testing and refinement cycles, ensuring that the established thresholds reflect the actual characteristics of the device used. Through this approach, the research aims to produce a drowsiness detection system prototype that is not only technically functional but also offers tangible practical value for driver safety in Indonesia—particularly for professional drivers undertaking long-distance journeys at night or in the early hours of the morning, when the risk of drowsiness is highest [4], [6]. Consequently, this research is expected to serve as a foundation for developing affordable, technology-based driving safety systems that can be more widely adopted in the future.

2. RESEARCH METHODS

2.1 Research Steps

This research employed a prototyping methodology, enabling the iterative development, testing, and refinement of the system based on evaluation results [18]. This methodology was selected because the system involves the integration of hardware and software requiring empirical calibration—specifically to establish optimal EAR thresholds and pitch angles on the actual device. The research process was conducted in six stages:

a. Requirement identification through a literature review and driver interviews;
b. System design;
c. Prototype development;
d. Testing;
e. Evaluation and analysis;
f. Report preparation.

The developed system architecture consists of four main components: the Camera Module 3 Wide NoIR as the image acquisition unit, the Raspberry Pi 3 Model B+ as the main processing unit, a speaker for alert output, and the Firebase Realtime Database as the IoT platform.

The system's hardware configuration used in this research is shown in Figure 1. The figure illustrates the physical connections between the power bank (power source), the Raspberry Pi 3 Model B+ (main processing unit), the Camera Module 3 Wide NoIR (connected via the CSI port), and the speaker (connected via the 3.5mm audio port).

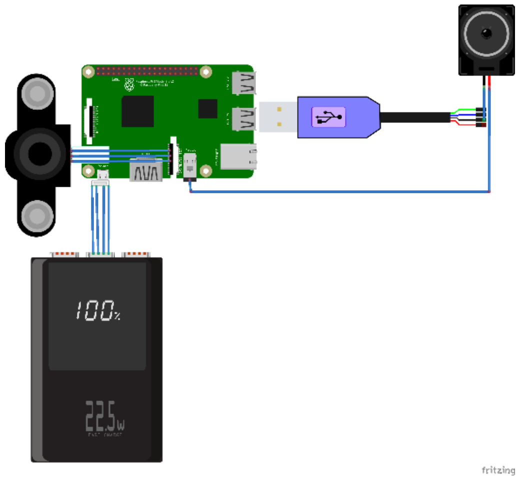

Figure 1. Hardware Assembly for a Driver Drowsiness Detection System

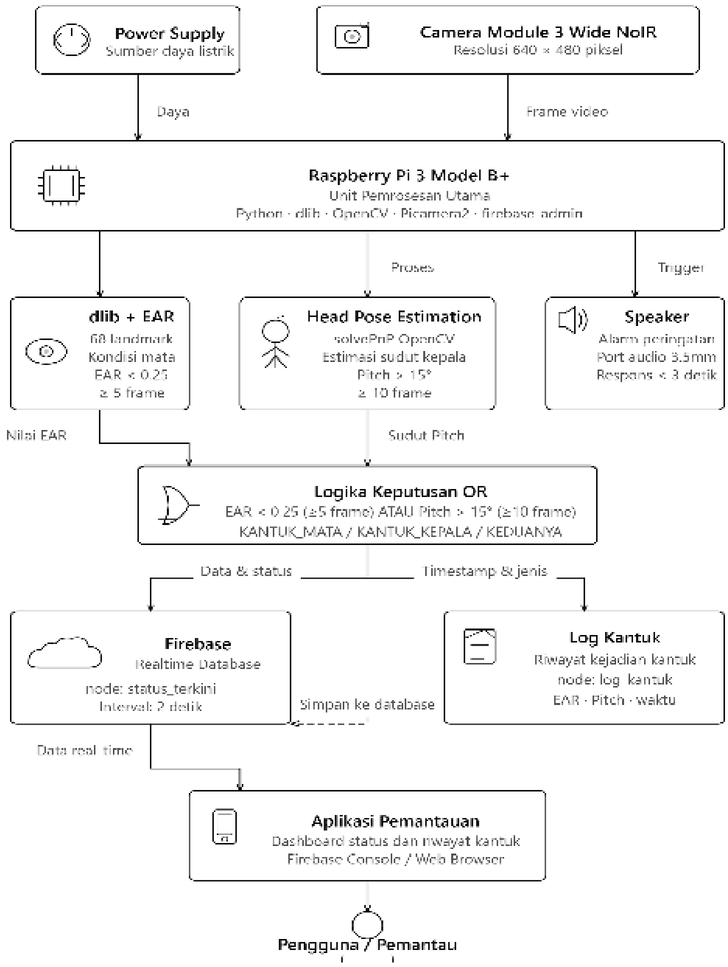

Figure 2. System Architecture Diagram for Driver Drowsiness Detection

Figure 2 illustrates the hardware and software architecture employed in this study. The Camera Module 3 Wide NoIR captures images of the driver's face and transmits video frames to the Raspberry Pi 3 Model B+, which serves as the primary processing unit. The Raspberry Pi executes face detection using dlib, calculates the Eye Aspect Ratio (EAR), and performs head pose estimation using OpenCV. The results from these two indicators are evaluated using OR logic to determine the driver's drowsiness state. Upon detecting drowsiness, the system activates a speaker to sound a warning alarm and simultaneously transmits status data and event history to the Firebase Realtime Database, enabling real-time monitoring via a dedicated application.

In greater detail, the system begins by continuously capturing camera frames, followed by face detection and the extraction of 68 landmark points using the dlib library [12]. Based on these landmarks, the system calculates the EAR and the head-pose pitch angle in parallel. If either or both drowsiness conditions are met, the system triggers an alarm and transmits data to Firebase. The complete system workflow is illustrated in Figure 3.

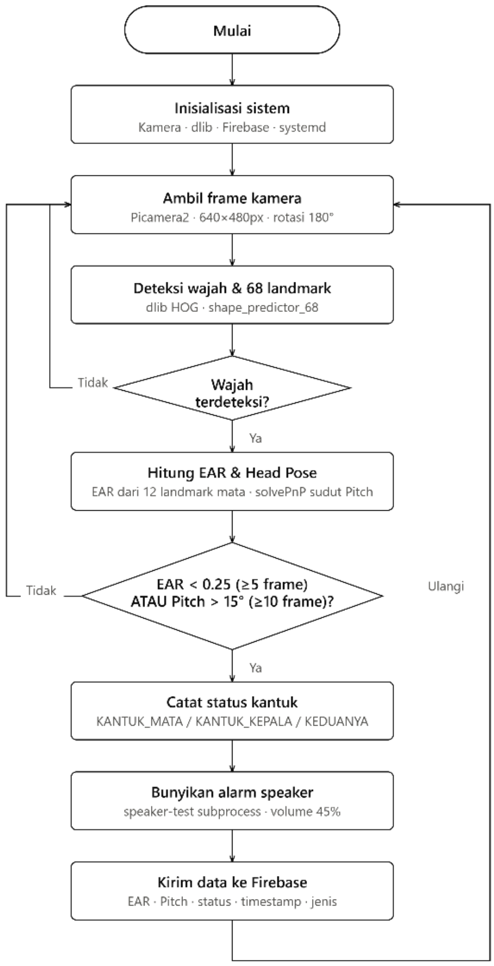

Figure 3. Driver Drowsiness Detection System Workflow Diagram

Figure 3 illustrates the system's continuous workflow, ranging from initialization, frame capture, face detection, and the parallel calculation of two indicators, to the evaluation of an OR logic condition that triggers an alert and transmits data to Firebase if drowsiness is detected.

2.2 Eye Aspect Ratio (EAR)

The Eye Aspect Ratio (EAR) is a geometry-based quantitative method for measuring eye openness using facial landmark coordinates [13]. The EAR is calculated based on the ratio of the vertical and horizontal distances between the eyelids, using the following formula:

$$EAR = \frac{(||p2-p6|| + ||p3-p5||)}{(2 \times ||p1-p4||)} \quad (1)$$

where p1 through p6 represent six landmark points in the area of each eye. In this study, the EAR is calculated from 12 landmark points (6 per eye) extracted from the dlib 68-point set [13]. An alarm is triggered if the EAR drops below a threshold of 0.25 for 5 consecutive frames—a value empirically calibrated to the Raspberry Pi 3B+'s frame rate of 5–8 fps.

2.3 Head Pose Estimation

Head pose estimation is implemented using the Perspective-n-Point (PnP) algorithm via the `solvePnP` function in OpenCV [17]. Six generic 3D facial reference points serve as the basis: the nose tip (0, 0, 0), chin (0, −330, −65), left eye corner (−225, 170, −135), right eye corner (225, 170, −135), left mouth corner (−150, −150, −125), and right mouth corner (150, −150, −125), all in millimeters. The rotation vector is decomposed using `RQDecomp3x3` to obtain the Pitch, Yaw, and Roll angles. An alarm is triggered if the Pitch angle exceeds 15 degrees for 10 consecutive frames; this threshold was determined through iteration, starting from an initial value of 10 degrees that caused false alarms during normal head movements.

2.4 Multi-Indicator Decision Logic and IoT Integration

The system employs OR logic as its decision-making mechanism: an alarm is triggered if the EAR condition is met (EYE_DROWSINESS), the Head Pose condition is met (HEAD_DROWSINESS), or both are active simultaneously (BOTH). This approach enables the independent detection of two distinct manifestations of drowsiness within a single processing pipeline [17]. Firebase Realtime Database is used to store two data structures: a `status_terkini` node—updated every 2 seconds—containing EAR values, pitch angles, and condition statuses; and a `log_kantuk` node that records each drowsiness event, including the triggering indicator type and a timestamp [11]. An error-handling mechanism ensures that local detection and alerting continue to function even if the internet connection is lost [10].

2.5 System Configuration

Table 2 summarizes all system configuration parameters used in the final implementation, including those adjusted through iterative calibration cycles.

Table 2. System Configuration Parameters

Parameter	Value	Basis for Determination
Image resolution	640 × 480 pixels	Accuracy vs. speed trade-off [12]
EAR threshold	0.25	Empirical calibration [13]
EAR consecutive frames	5 frames	Adjusted RPi 3B+ frame rate [12]
Head pose pitch threshold	15 degrees	Iterative calibration starting from 10° [17]
Head pose consecutive frames	10 frames	More gradual head movement [17]
Subject-to-camera distance	40–60 cm	Optimal for landmarks and head pose
Firebase transmission interval	2 seconds	Network load balancing [11]
Image rotation	180 degrees	Physical camera orientation compensation [19]

3. RESULT AND DISCUSSION

3.1 Face Detection and EAR Value Test Results

The system is capable of consistently detecting the subject's face at a distance of 40–60 cm under bright lighting conditions, in accordance with the characteristics of the dlib HOG model [12], [13]. Table 3 presents the measured EAR value ranges for the three tested eye conditions.

Table 3. Measurement Results of EAR Values Under Various Eye Conditions

Eye Condition	EAR range	Average	Information
Open (normal)	0.25 – 0.37	0.31	Above threshold; does not trigger an alarm [13]
Briefly blinking	0.14 – 0.23	0.19	Counter reset; no false alarms [13]
Closed (DROWSY_EYES)	0.13 – 0.20	0.16	Below the threshold ≥5 frames; alarm sounds [13]

The clear separation between the EAR ranges for open eyes (0.25–0.37) and closed eyes (0.13–0.20) confirms that a threshold of 0.25 effectively discriminates between alert and drowsy states. No false alarms occurred during normal blinking, as the duration of natural blinks did not reach five consecutive frames at a frame rate of 5–8 fps [12]. This indicates that the consecutive frame counting mechanism functions effectively as a filter to distinguish normal blinks from actual drowsiness. The distinct separation of EAR values between the open (0.25–0.37) and closed (0.13–0.20) states demonstrates that the empirically calibrated threshold of 0.25 lies precisely between these ranges, providing an adequate margin for discrimination. The mean EAR value of 0.31 for the open-eye state provides a 0.06 margin from the threshold, which is sufficient to accommodate natural EAR fluctuations caused by slight facial movements while driving without triggering false alarms.

3.2 Head Pose Estimation Test Results

The Head Pose Estimation module successfully estimates head orientation in real-time using solvePnP [17]. Table 4 presents the measured pitch angles for four head position conditions, obtained from 10 measurements per condition.

Table 4. Head Pose Pitch Angle Measurement Results Under Various Head Conditions

Head Condition	Pitch Range	Average	Information
Upright (normal)	−5° to +5°	0,8°	Below threshold; NORMAL [17]
Lightly bowed	6° to 13°	9,5°	Below threshold; not triggered [17]
Moderately bowed	15° to 28°	21,3°	Exceeds threshold for ≥10 frames; alarm sounds [17]
Deeply bowed	29° to 45°	36,7°	Well above threshold; alarm sounds [17]

The system successfully distinguished between an upright head position (Pitch < 5°) and a head-nodding posture indicative of drowsiness (Pitch > 15°; averaging 21.3° during moderate nodding). No false alarms were triggered by turning (Yaw) or tilting (Roll) movements, as the threshold was applied solely to the Pitch angle [17]. Selecting Pitch as the only monitored angle was a sound design decision, given that forward nodding (positive Pitch) is a characteristic manifestation of driver drowsiness, whereas turning the head sideways is a normal behavior associated with checking mirrors or changing lanes. This also explains the choice of a 15° threshold: this value accommodates normal, slight head movements (below 13°) while remaining sensitive to the head-nodding associated with drowsiness (averaging 21.3° during moderate nodding).

3.3 Results of Simultaneous Drowsiness Detection Testing (Multi-Indicator)

Simultaneous drowsiness testing was conducted by simulating concurrent eye closure and head nodding. The system successfully detected this condition and recorded it as a "BOTH" status in the Firebase log [11]. A crucial finding was that the Head Pose indicator triggered before the EAR during simulated gradual drowsiness, as the head began to nod before the eyes fully closed. This aligns with patterns of gradual drowsiness described in the literature [17], [15] and confirms the practical value of a multi-indicator approach in providing earlier warnings.

The advantage of the OR logic lies in its ability to independently capture two distinct manifestations of drowsiness. Relying solely on EAR would fail to detect instances where the head nods while the eyes remain partially open; conversely, relying only on Head Pose would miss cases where the eyes close while the head remains upright. Trigger indicator data stored in Firebase enables the analysis of individual drowsiness patterns, providing valuable insights for preventive decision-making [11]. Across five trials involving simultaneous drowsiness indicators, the HEAD_DROWSINESS indicator was triggered before the EYE_DROWSINESS indicator in four instances. The average time difference between the triggering of the Head Pose indicator and the EAR indicator was approximately 0.7 seconds. This indicates that during the onset of gradual drowsiness, a driver's head tends to nod before the eyes fully close—a pattern physiologically consistent with the loss of neck muscle tone during the early stages of drowsiness [4]. These findings confirm that a multi-indicator system not only detects a broader range of drowsiness manifestations but also, on average, provides earlier warnings than a system relying solely on EAR.

3.4 Response Time Test Results

Response time was measured manually using a stopwatch over five repetitions per indicator, calculated from the onset of drowsiness to the moment the alarm sounded [20]. Table 5 presents the measurement results.

Table 5. Response Time Measurement Results by Indicator

Test	EAR Response Time (seconds)	Head Pose Response Time (seconds)
1	1.90	2.40
2	2.90	3.10
3	1.85	2.65
4	1.75	2.80
5	1.62	2.55
Average	2.00	2.70
Minimum	1.62	2.40
Maximum	2.90	3.10

The average response times were 2.00 seconds for EAR and 2.70 seconds for Head Pose. This difference in response time stems from the differing consecutive-frame thresholds (5 versus 10 frames) established to prevent false alarms caused by normal head movements. Both values fall within an acceptable range for early drowsiness warnings, given that hazardous situations typically require more than 2–3 seconds before posing a risk of losing vehicle control [4], [5]. The significant variation in EAR response times (1.62–2.90 seconds) is attributed to the inconsistent frame rate of the Raspberry Pi 3B+, which fluctuates between 5 and 8 fps depending on the prevailing computational load. At 5 fps, five consecutive frames equate to 1.0 second of actual detection time plus processing latency, whereas at 8 fps, the required time is shorter. A similar variation was observed for the Head Pose indicator (2.40–3.10 seconds), consistent with its higher consecutive-frame threshold (10 frames). Future development involving an upgrade to a platform with a more stable frame rate would yield more consistent and predictable response times, thereby enhancing system reliability in real-world usage scenarios.

3.5 Firebase IoT Testing Results

The `status_terkini` node successfully receives updates every 2 seconds containing the EAR value, Pitch/Yaw/Roll angles, condition status, and a timestamp [11]. The `log_kantuk` node records every drowsiness event along with the specific type of triggering indicator. No data loss was observed under stable connection conditions. When the internet connection was disconnected, local detection and alerting continued without interruption, demonstrating that the error-handling mechanism functions as designed and that safety functions do not rely on internet connectivity [10], [11].

This is practically significant, given that unstable internet signals are frequently encountered on toll roads, in tunnels, and in areas with limited network infrastructure. Furthermore, the dual-node data structure offers monitoring flexibility: the `status_terkini` node enables real-time monitoring of the driver's condition, while the `log_kantuk` node allows for historical analysis of drowsiness patterns, which transportation companies can utilize for periodic driver performance evaluations. By distinguishing between trigger indicator types (EYE_DROWSINESS, HEAD_DROWSINESS, or BOTH) in each log, the stored data is far more informative than that of systems that merely record drowsiness status in a binary format.

3.6 Black Box Testing Results

Table 6 summarizes the results of Black Box testing for 11 scenarios covering all the system's main functions.

Table 6. Black Box Testing Results Summary

No	Tested Features	Expected Output	Results
1	Face detection	Detection box and landmarks displayed	It worked
2	EAR – eyes open	EAR > 0.25; normal status	It worked
3	EAR – normal blink	Counter reset; no false alarm	It worked
4	Eye-based drowsiness detection	Status change; alarm sounds	It worked
5	Head-based drowsiness detection	Status change; alarm sounds	It worked
6	Simultaneous detection of both	Status of both recorded	It worked
7	Audio alarm	Speaker emits sound	It worked
8	Firebase transmission	EAR, Pitch, and type data saved	It worked
9	Internet interruption	Local detection continues running	It worked
10	Live streaming	Video, EAR, and Pitch displayed in browser	It worked
11	Auto-start	Program runs automatically	It worked

All 11 scenarios were declared successful. Note on scenario 10: using live streaming in conjunction with a multi-indicator detection pipeline causes high CPU load on the Raspberry Pi 3B+ which has the potential to disrupt the stability of the Wi-Fi connection in long-term use [12]. However, it should be emphasized that the live streaming feature is optional and does not affect the system's primary safety functions. The core functions, namely face detection, calculating EAR and Head Pose, providing voice alarms, and sending data to Firebase still work well without relying on the live streaming feature. The success of 10 of 11 scenarios with no noted limitations and 1 scenario with minor notes indicates a good level of functional maturity of the prototype for the initial development stage using the prototyping methodology [18].

3.7 Comparison with Previous Studies

Compared to previous studies, the developed system offers several distinct contributions. Shimpi et al. [12] utilized only EAR, excluding head pose and IoT capabilities. Florez et al. [13] combined EAR and MAR but lacked head pose estimation and cloud integration. Chen et al. [17] implemented a multi-feature approach—including head pose—on the more expensive Raspberry Pi 4, without IoT integration. This study addresses these gaps by combining EAR and head pose estimation on the more affordable Raspberry Pi 3B+ while integrating Firebase for real-time remote monitoring. Utilizing the same set of 68 dlib landmarks for both indicators enhances computational efficiency, distinguishing this approach from systems that employ separate modules for each indicator [12], [17].

Furthermore, the integration of Firebase with a logging structure that categorizes triggering indicators yields richer data compared to systems that merely record binary drowsiness states. Regarding implementation costs, the Raspberry Pi 3 Model B+ used in this study is priced at approximately one-fifth of the Raspberry Pi 4, thereby creating opportunities for wider adoption among small- and medium-scale transport operators in Indonesia.

Overall, the combination of EAR, head pose estimation, and Firebase IoT on the affordable Raspberry Pi 3B+ platform distinguishes this study from all previously reviewed research and addresses the three gaps identified in the Introduction. The practical implications of these findings are significant. The developed system has the potential to serve as an affordable driving safety solution accessible to both professional and individual drivers in Indonesia, without requiring expensive hardware or complex infrastructure. Additionally, the remote monitoring capability provided by Firebase enables transport companies to integrate the system into their existing fleet management platforms.

4. CONCLUSION

This study successfully developed and evaluated a real-time driver drowsiness detection system combining Eye Aspect Ratio (EAR) and Head Pose Estimation on a Raspberry Pi 3 Model B+ with Firebase IoT integration. The system demonstrated that both indicators could be calculated in parallel using the same 68 dlib facial landmarks, enabling multi-indicator detection on an affordable embedded platform without significant computational overhead.

Test results showed that EAR consistently distinguished between open (0.25–0.37) and closed (0.13–0.20) eye states without triggering false alarms during normal blinking. Head Pose Estimation successfully distinguished an upright head position (Pitch $< 5°$) from nodding (Pitch $> 15°$, averaging 21.3°). Average response times—2.00 seconds via EAR and 2.70 seconds via Head Pose—fell within an acceptable range for early warning systems. A key finding was that the Head Pose indicator triggered earlier than the EAR in gradual drowsiness scenarios, confirming the superiority of the multi-indicator approach over a single-indicator EAR system. Firebase integration, combined with the differentiation of triggering indicator types, yielded more informative historical data. All 11 Black Box testing scenarios were successful, confirming the system's readiness for deployment in vehicles.

Future research should focus on integrating an infrared illuminator module for nighttime operation, testing with a more diverse group of subjects to validate threshold generalizability, and considering an upgrade to the Raspberry Pi 4 or 5 to address the observed computational limitations.

REFERENCES

[1] D. Andarini, A. Camelia, and M. M. Ibrahim, "Factors related to road accidents in Palembang, South Sumatera, Indonesia," Int J Publ Health Sci, vol. 10, no. 3, pp. 638–645, 2021, doi: 10.11591/ijphs.v10i3.20768.

[2] H. Pranoto, A. Adriansyah, D. Feriyanto, A. Wahab, and S. Zakaria, "Propose Safety Engineering Concept Speed Limiter and Fatigue Control Using SLIFA for Truck and Bus," SINERGI, vol. 24, no. 3, p. 237, 2020, doi: 10.22441/sinergi.2020.3.009.

[3] R. Zuraida and B. S. Abbas, "The factors influencing fatigue related to the accident of intercity bus drivers in Indonesia," International Journal of Technology, vol. 11, no. 2, pp. 342–352, 2020, doi: 10.14716/ijtech.v11i2.3792.

[4] S. Soares, T. Monteiro, A. Lobo, A. Couto, L. Cunha, and S. Ferreira, "Analyzing driver drowsiness: From causes to effects," Sustainability, vol. 12, no. 5, 2020, doi: 10.3390/su12051971.

[5] M. A. Puspasari et al., "Prediction of drowsiness using EEG signals in young Indonesian drivers," Heliyon, vol. 9, no. 9, 2023, doi: 10.1016/j.heliyon.2023.e19499.

[6] A. A. D. Parami Dewi, "Investigating motorists perceptions towards road safety," Civil Engineering and Architecture, vol. 9, no. 5, pp. 1339–1346, 2021.

[7] T. Suryadi and Kulsum, "Forensic investigation in a multiple trauma deceleration case due to road traffic accidents," Bali Medical Journal, vol. 9, no. 1, pp. 172–177, 2020.

[8] W. Suprihatiningsih et al., "Impact on Driver Behavior, Performance, Fatigue, Over Speed and Infrastructure as A Factor of Road Accident: A Review," in IOP Conf. Ser.: Mater. Sci. Eng., 2020, doi: 10.1088/1757-899X/864/1/012105.

[9] M. A. Khan, T. Nawaz, U. S. Khan, A. Hamza, and N. Rashid, "IoT-Based Non-Intrusive Automated Driver Drowsiness Monitoring Framework," IEEE Access, vol. 11, pp. 14385–14397, 2023.

[10] S. Ayas, B. Donmez, and X. Tang, "Drowsiness Mitigation Through Driver State Monitoring Systems: A Scoping Review," Hum Factors, vol. 66, no. 9, pp. 2218–2243, 2024.

[11] D. N. Bestari and A. Wibowo, "An IoT-Based Real-Time Weather Monitoring System Using Telegram Bot and Thingsboard Platform," Int. J. Interact. Mobile Technol., vol. 17, no. 6, pp. 4–19, 2023.

[12] R. Shimpi, V. Chaudhari, P. Patil, B. Narkhede, V. D. Chaudhari, and M. N. Patil, "Driver Drowsiness Detection System Using Raspberry Pi," Int. J. Innov. Eng. Sci., vol. 10, no. 4, pp. 22–25, 2025.

[13] R. Florez, F. Palomino-Quispe, A. B. Alvarez, R. J. Coaquira-Castillo, and J. C. Herrera-Levano, "A Real-Time Embedded System for Driver Drowsiness Detection Based on Visual Analysis of the Eyes and Mouth Using CNN and MAR," Sensors, vol. 24, no. 19, 2024, doi: 10.3390/s24196261.

[14] M. M. Naddaf-Sh, A. Lee, K. Yen, E. Amini, and I. Soltani, "Low-Cost Infrared Vision Systems for Improved Safety of Emergency Vehicle Operations," IEEE Access, vol. 13, pp. 173241–173253, 2025.

[15] M. Mansyur et al., "Long working hours, poor sleep quality, and work-family conflict: determinant factors of fatigue among Indonesian tugboat crewmembers," BMC Public Health, vol. 21, no. 1, 2021.

[16] C. Chen, H. Gu, S. Lian, Y. Zhao, and B. Xiao, "Investigation of Edge Computing in Computer Vision-Based Construction Resource Detection," Buildings, vol. 12, no. 12, 2022.

[17] Z. Chen, D. Shi, Y. Guo, and X. Fu, "Real-Time Multi-Feature Driver Drowsiness Detection Based on Facial Landmarks Using LSTM on Raspberry Pi," Sensors, vol. 25, no. 3, 2025, doi: 10.3390/s25030920.

[18] K. A. Obayes and A. Hamzah, "Using of prototyping in develop an employee information management," Measurement: Sensors, vol. 24, p. 100557, Dec. 2022, doi: 10.1016/j.measen.2022.100557.

[19] Raspberry Pi Ltd, "Raspberry Pi Documentation," 2024. [Online]. Available: https://www.raspberrypi.com/documentation/

[20] R. Kaur and S. Bhatt, "Design and Implementation of Drowsiness Detection System for Driver Safety Using Facial Landmarks," Int. J. Comput. Appl., vol. 184, no. 7, pp. 1–6, 2022.

[21] A. Kukudkar, R. Pawar, S. More, V. Gawali, and P. Gawali, "Driver Drowsiness and Seatbelt Monitoring System," Int. J. Innov. Res. Technol., vol. 11, no. 12, 2025.

[22] E. H. Istoto et al., "Monitoring of Soil Humidity and Temperature Using IoT and AI for Remote Management," Int. J. Informatics Comput. (IJICOM), vol. 7, no. 2, pp. 793–805, 2025.

[23] F. N. R. D. Haq and K. D. Irianto, "Indoor Air Quality System: A Design of Monitoring CO and CO_2 Substances for Elderly Home Using IoT," The Indonesian Journal of Computer Science, vol. 13, no. 5, pp. 7230–7243, 2024.