IoT Cyber-physical System with Computer Vision for Driver Drowsiness Monitoring in e-Health Applications

RICARDO YAURI, RAFAEL ESPINO, ANTERO CASTRO
Facultad de Ingeniería, Universidad Tecnológica del Perú,
Lima,
PERÚ

Abstract: -This research presents the design and implementation of a cyber-physical system based on IoT technologies and computer vision aimed at the early detection of drowsiness in land transport drivers with applications in the field of occupational e-Health. The system integrates a camera, a facial analysis model in Python using Mediapipe and OpenCV, and embedded devices such as the ESP32 for continuous monitoring of visual signals to detect eye closure or yawning. The proposed architecture includes real-time communication via the MQTT protocol, triggering local alerts (sound and light), storing visual evidence, and sending notifications to Telegram. In addition, a remote monitoring interface using Node-RED and a mobile application in Kotlin for local event visualization are implemented. Functional tests were conducted, demonstrating the viability of the system as a preventive tool for reducing fatigue-related risks in the context of monitoring ground vehicle drivers. The solution's modular and portable approach allows for adaptation to other biomedical monitoring environments and suggests a low-cost alternative for digital health applications.

Key-Words: - Telecare, Work Fatigue, OpenCV, Mediapipe, Android, Facial analysis, Driver monitoring

Received: May 22, 2025. Revised: July 15, 2025. Accepted: September 21, 2025. Published: May 22, 2026.

1 Introduction

The advancement of digital technologies has generated the development of intelligent systems aimed at monitoring the physical and cognitive state of people in critical work environments [1], [2]. In the particular case of the land transport sector, there are high levels of risk associated with driver fatigue and drowsiness, where these factors affect driving activities [3], affecting road and pedestrian safety [4], [5]. In this context, the integration of cyber-physical systems with visual detection capabilities and real-time communication has allowed the generation of solutions for the development of systems and tools to prevent and alert to anomalous events related to e-Health [6], [7].

In this context, one of the main problems is to have real-time solutions for the detection [8] and alert of drowsiness states in land transport drivers, but using portable devices integrated with computer vision technologies [9], [10]. Although there are rules and regulations to limit driving hours, there are no modern or efficient continuous monitoring mechanisms to identify physical signs of drowsiness, adapted to contexts of driver fatigue before critical events occur [11], [12].

The novelty of this research lies in its difference from other systems that rely solely on computer vision and expensive hardware. The article proposes a portable architecture that uses Mediapipe-based facial landmark extraction, facial feature detection, and monitoring via MQTT, Node-RED, and Telegram with a native Android application.

It is important to note that this research follows a descriptive and qualitative methodological approach. Therefore, the aim of this work is not to provide a quantitative comparison but rather to validate the cyber-physical architecture for drowsiness detection. In this case, statistical validation is not within the scope of this study.

2 Literature Review

Some previous works have addressed the problem through physiological sensors such as EEG, heart rate monitors [1], or sensors that require physical contact with the monitored people [13], which makes it difficult to adequately detect the signals of interest due to sensitivity to noise [14], [15]. On the other hand, other papers describe the use of cameras for the visual analysis of the eyes or face using artificial intelligence algorithms [16] that require high computational resources [17], [18]. However, many of these solutions require specialized equipment and are not easily integrated into real-world environments.

Previous research describes systems that use computer vision to detect indicators such as blinking rate, prolonged eye closure, or yawning using techniques such as Haar Cascades [19], [20] or deep learning models [21], [22]. On the other hand, remote monitoring platforms have also been implemented using IoT and MQTT communication, applied to the monitoring of biomedical variables in public transport applications [23], [24]. However, the integration of these technologies into a single system to act in real time or generate alerts using portable devices still represents an opportunity for improvement in practical applications because these are still being studied in research [25], [26].

The novelty of this article lies in the technological gap described in the reviewed studies, which lack a platform integrating local detection with computer vision, event transmission via MQTT, or activation of physical alerts. Most solutions focus solely on computer vision techniques and do not integrate Mediapipe with an ESP32 alert device.

Based on the above, this research shows the design and implementation of a cyber-physical system using Internet of Things technologies and computer vision mechanisms to detect events related to drowsiness in land transport drivers in real time. In this case, the solution integrates local visual processing using Python and the Mediapipe framework, embedded devices based on the ESP32 microcontroller, along with data transmission using the MQTT protocol. Event recognition models are also integrated into a smartphone for notification and monitoring. These components help achieve the goal of contributing to road safety through a low-cost, scalable, and adaptable tool within the e-Health field.

3 Methodology

The cyber-physical system design is composed of visual processing components, embedded hardware control, wireless communication with an MQTT broker, and remote monitoring with Node-Red (Figure 1). The four components of the system are: a computer vision model for drowsiness detection and visual feedback, an ESP32-based IoT device for alert triggering, a monitoring stage using the MQTT protocol, and a mobile application for local event visualization using integrated computer vision. These components capture, analyze, and notify drivers in real time when fatigue-related behaviors such as yawning and eyelid closing occur. The data flow begins with facial image capture, continues with Python processing, risk event detection, physical alert activation, and topic updates in the MQTT broker.

The novelty of the system lies in the combination of a facial mark detector with Mediapipe, an ESP32 IoT module, and MQTT communication, which differentiates it from existing solutions.

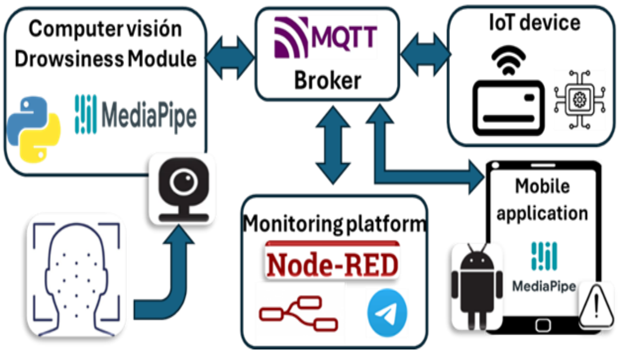

Fig. 1. System Architecture
Source: created by the authors

3.1 Drowsiness detection stage using a computer vision model

This component focuses on integrating a computer vision model to detect the face and feature points related to mouth opening and eyelid closure, using Mediapipe and OpenCV (Figure 2). This is achieved by using a camera connected to a computer to capture real-time images of the driver's face. The model extracts facial landmarks, focusing on the eyes and mouth, to identify signs of drowsiness such as prolonged eye closure or frequent yawning. The algorithm generates visual alerts on the screen, plays a sound when events are detected, and stores images or videos at that time. In addition, temporal signals of facial variables are graphed. This component is the core of the analysis for locally classifying risk events before transmitting or triggering alerts.

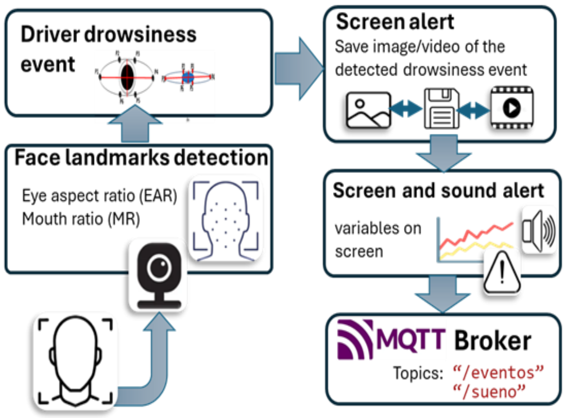

Fig. 2. Components of the detection algorithm
Source: created by the authors

Table 1 shows the selected facial landmarks from the Mediapipe model for detecting drowsiness-related gestures. In this case, the most important points around the eyes and mouth are considered, which are necessary to calculate the eye opening (EAR) and mouth opening (MAR) metrics. These landmarks allow for the noninvasive identification of fatigue-associated behaviors, such as excessive blinking or yawning.

Table 1. Comparison of noise signals over time

Region	Indexes	Description	Function in detection
Left eye	[27, 23, 130, 243]	Upper and lower eyelids	Calculating the EAR Index
Right eye	[257, 253, 463, 359]	Upper and lower eyelids	Calculating the EAR Index
Mouth	[11, 16, 57, 287]	Upper and lower borders of the lips	Calculating the MAR Index to Detect Yawning

Source: created by the authors

Furthermore, to obtain the values related to the detection of yawning or eye closure, the relative distance between the characteristic points described in Table 1 has been considered. Using these reference points, the calculation is performed based on the Euclidean distance, where, in the case of the Eye Aspect Ratio (EAR), it is determined if the eyes are closed by measuring the relationship between the height and the width of the eye, while for the Mouth Aspect Ratio (MAR), they are used to detect the yawning event.

$$EAR_{left} = \frac{\|P_{27} - P_{23}\|}{\|P_{130} - P_{243}\|} x100 \quad (1)$$

$$EAR_{left} = \frac{\|P_{257} - P_{253}\|}{\|P_{463} - P_{359}\|} x100 \quad (2)$$

$$EAR = \frac{vertical\ eye\ distance}{Hhorizontal\ eye\ distance} * 100 = \frac{EAR_{left}}{EAR_{right}} \quad (3)$$

$$MAR = \frac{Vertical\ distance\ from\ the\ mouth}{Horizontal\ distance\ from\ the\ mouth} * 100 = \frac{\|P_{11} - P_{16}\|}{\|P_{57} - P_{287}\|} x100 \quad (4)$$

Threshold values were defined through preliminary calibration tests. Mediapipe facial landmarks produce normalized distances, for which experimental calibration was performed with test subjects, obtaining an average EAR of 53–55 under normal conditions and values below 50 during eyelid closure. Similarly, reported MAR values are typically between 25 and 35 for yawn detection, with MAR values of 10–15 in normal states and values above 35 during yawning.

The algorithm integrates the process of calculating the distances between reference points and, using the "detect_drowsiness" function, analyzes whether the EAR value is less than 50 for more than thirty consecutive frames, indicating the presence of a sleep event. On the other hand, if the MAR value is greater than thirty-five, a yawn event is present. In both cases, a visual and audible alert is triggered, a 3-second video is recorded, an image of the event is captured, and the event is logged in a CSV file. Finally, the event is published via MQTT using the topic "/monitor/user1/event" with data in JSON format, and the topic "monitor/user1/sound" is used to send the status to the subscribed IoT device (Figure 3).

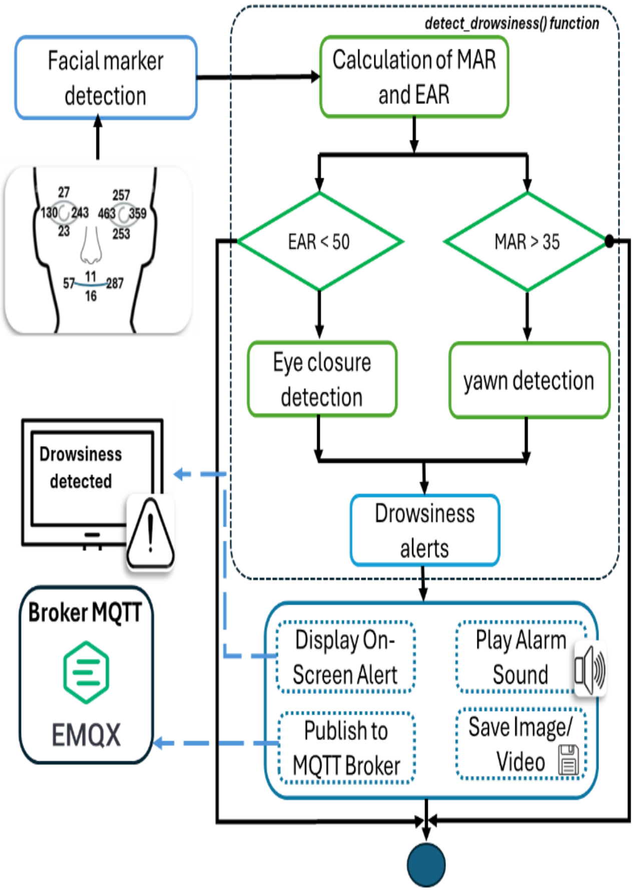

Fig. 3. Algorithm flowchart
Source: created by the authors

The real-time display scheme was designed based on the variables monitored by the algorithm (Table 2) on the screen as a feedback component and a graphical user interface. In this case, numerical indicators of the EAR and MAR values, their evolution over time, and the appearance of visual alert messages that notify the user of drowsiness are included (Figure 4).

Table 2. Indicators shown on the display screen

Parameter	Chart	Utility
EAR Left Ratio	Text	Fluctuates with blinks
EAR Right Ratio	Text	Fluctuates with blinks
Mouth Ratio	Text	Fluctuates with mouth movements
Sleep count	Text	Prolonged eye closure
Yawn Count	Text	Prolonged mouth opening
Number of events	Vertical bars	Increases over time
State: ACTIVO / DROWSY	Text	Visual identification of sleepiness lapses
Time with your eyes closed.	Horizontal bar	Useful for detecting prolonged microsleeps
Average EAR ratio	Time series	Evolution of eye closure patterns
Average mouth ratio	Time series	Evolution of yawning patterns
Event detected	Alert text	Triggered when the event is detected

Source: created by the authors

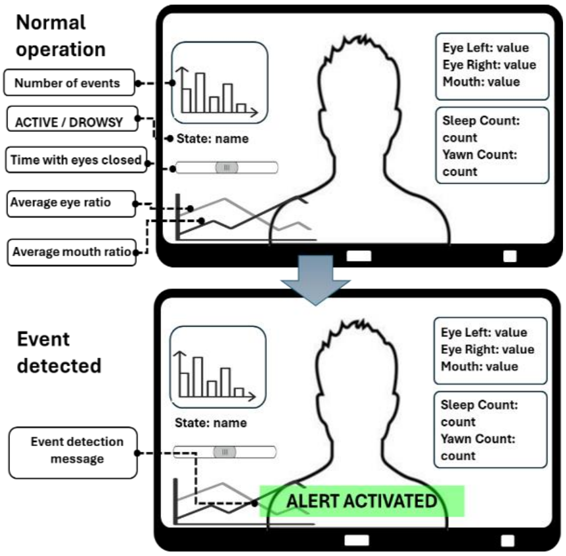

Fig. 4. Indicators shown on the display screen
Source: created by the authors

The system establishes communication with an MQTT broker to report drowsiness or yawning events. When the algorithm detects a risk event, it packages the data, including the type ("SLEEP" or "YAWN"), facial ratios (EAR and MAR), and a timestamp.

This information is packaged in JSON format and sent via an MQTT topic (/monitor/user1/events). Additionally, a simpler message ("SLEEP" or "YAWN") is also published to a second topic (/monitor/user1/sueno), where the subscribed IoT device receives these alerts and can trigger physical responses. Periodic data with the "NORMAL" status and blink rate are also sent for continuous monitoring (Figure 5).

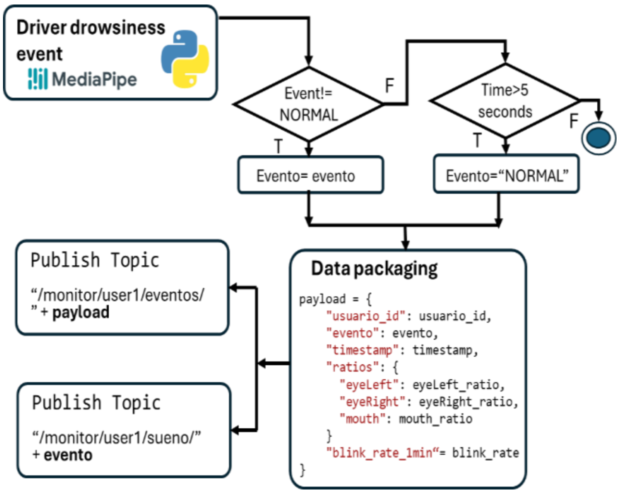

Fig. 5. MQTT Communication scheme with the MQTT broker
Source: created by the authors

3.2 Development of the IoT device for alarm activation

The control device was implemented based on the ESP32 microcontroller, which triggers physical signals such as a buzzer, LED lights, and an OLED display (to show the type of event detected). Communication with the MQTT broker is via Wi-Fi, and the embedded firmware processes the received messages to trigger real-time actions (Figure 6).

In addition, the algorithm integrates the PubSubClient communication libraries and subscribes to the "sensor" and "data" topics, which are processed in a separate main loop thread to update the information on an OLED display. This component aims to generate an immediate and visible response in the driver's environment, even in the absence of a graphical interface (Figure 7).

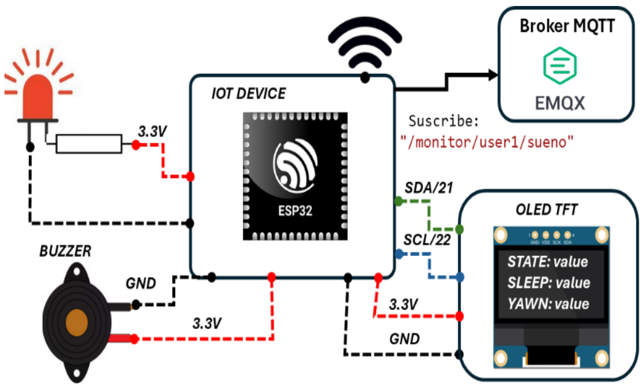

Fig. 6. ESP32 Schematic of the embedded device based on the ESP32 microcontroller
Source: created by the authors

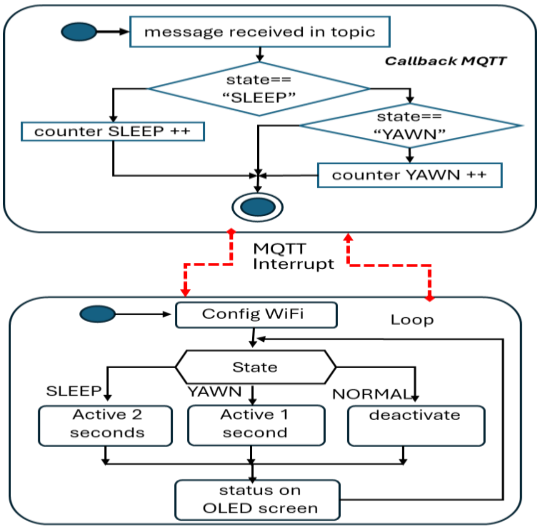

Fig. 7. Firmware flowchart
Source: created by the authors

3.3 Monitoring platforms (Node-RED and Telegram)

The MQTT broker receives information generated by the drowsiness detection stage, such as prolonged eye closure or yawning. From the Node-RED service, a web dashboard is displayed with visual indicators using gauges, charts, and a template to display a captured image of the driver's condition. To do this, the data received in JSON format must be processed and decoded to send it to the appropriate widget. Additionally, if the creation of a new image is detected (from the detection stage), the image is uploaded to the web monitoring interface (Figure 8). On the other hand, communication nodes are used with a Telegram bot to send images or video captures when the event is detected (Figure 9). To do this, it is necessary to encode the image in base 64, which is compatible with Telegram. In addition, the information received through the MQTT subscription is decoded into functions displayed in each dashboard widget.

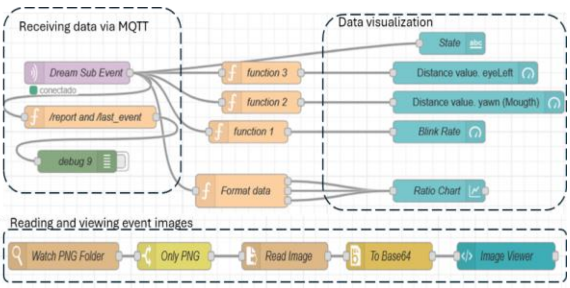

Fig. 8. MQTT subscription nodes for updating variables in the dashboard
Source: created by the authors

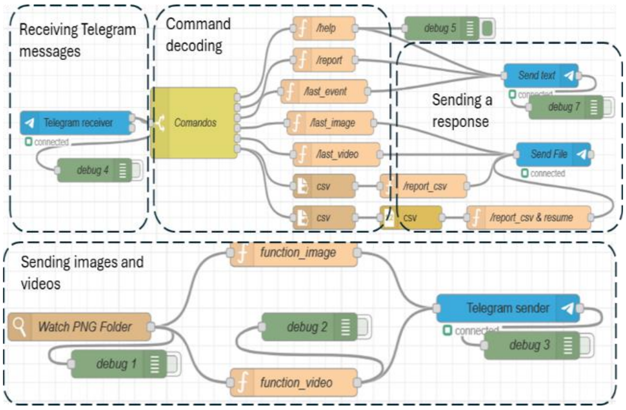

Fig. 9. Communication nodes for Telegram alerts
Source: created by the authors

3.4 Mobile application

Considering the system's multiplatform approach, a mobile application for Android devices was developed using the Kotlin language to provide a portable monitoring interface. The application is designed to display the driver's status in real time using graphical indicators of eye opening/closing and yawn frequency. In addition, it emits alert sounds when an event is detected and sends the captured image to the Telegram bot (Figure 10). To achieve this, functions were created for managing MQTT communication, Telegram communication, managing the screen's visual components, and event detection. The mobile app also allows facial indicators to be displayed in text and time-based graph formats.

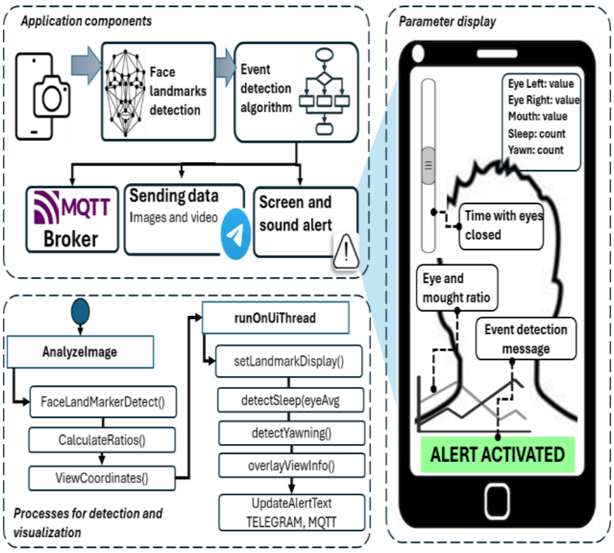

Fig. 10. Mobile application operation diagram
Source: created by the authors

4 Results and discussion

The results demonstrate that the system differs from other existing systems that only use machine vision or IoT because it integrates an alerting and remote monitoring stage.

The evaluation focused on observing the system's behavior, rather than applying statistical validation procedures. Since the research objective is descriptive and qualitative, EAR and MAR values are used to understand the system's behavior instead of performing hypothesis tests or statistical estimations.

4.1 Drowsiness detection with the computer vision model

Figure 11 shows the EMQX MQTT broker display console, where the verification of the data sent from the event detection stage was performed, considering the verification of reception of the Sleep, YAWN, and NORMAL events. Figure 12 shows the camera screen in operation during drowsiness detection, where the facial markers, the temporal values of the eye-opening index (EAR) and mouth opening index (MAR), and the activation of alert messages are displayed.

Fig. 11. Mobile application operation diagram
Source: created by the authors

Fig. 12. Visualization of the running drowsiness detection algorithm
Source: created by the authors

Additionally, Table 3 shows a comparison of the average values of the EAR and MAR indices under different driver conditions: normal state, onset of fatigue, drowsiness (SLEEP), and yawning (YAWN). These values allow us to observe how facial metrics behave during drowsiness-related events.

Table 3. Comparison of facial metrics

State	Average EAR	Average MAR	Observations
Normal	53.1	11	Normal blinking
SLEEP	45.5	10	Prolonged eye closure
YAWN	54.6	46.7	Excessive mouth opening

Source: created by the authors

The chosen EAR and MAR values showed that under normal driving conditions, EAR values of 53 and MAR values of 11 were obtained, while sleep events obtained values of 45, and yawning MAR values of 46. This validates that the selected thresholds (EAR < 50, MAR > 35) allow for the correct classification of normal drowsiness events.

The results obtained in the detection algorithm evaluation stage demonstrate the effectiveness of the computer vision model in identifying events. The accuracy in detecting prolonged eye closure and yawning demonstrates that the integration of Mediapipe and OpenCV is suitable for handheld environments and edge computing. Compared with invasive or sensor-dependent methods, this technique proved to be non-intrusive and adaptable to portable platforms. Furthermore, it is evident that during sleep or fatigue, the EAR decreases due to prolonged eyelid closure, while during yawning, the MAR increases significantly, validating the use of these metrics as reliable indicators.

4.2 IoT device

Figure 13 shows the ESP32-based embedded device running using the Wokwi simulation platform and receiving data with the display of information on an OLED screen and the reception of MQTT messages by subscription.

In the case of the embedded system, tests showed that MQTT communication is dependable and low-latency, enabling buzzers, LEDs, and OLED display messages to be activated in less than half a second after detection. Furthermore, compared to more expensive solutions with proprietary hardware, the use of open components, such as the ESP32, allows for replication in different work environments.

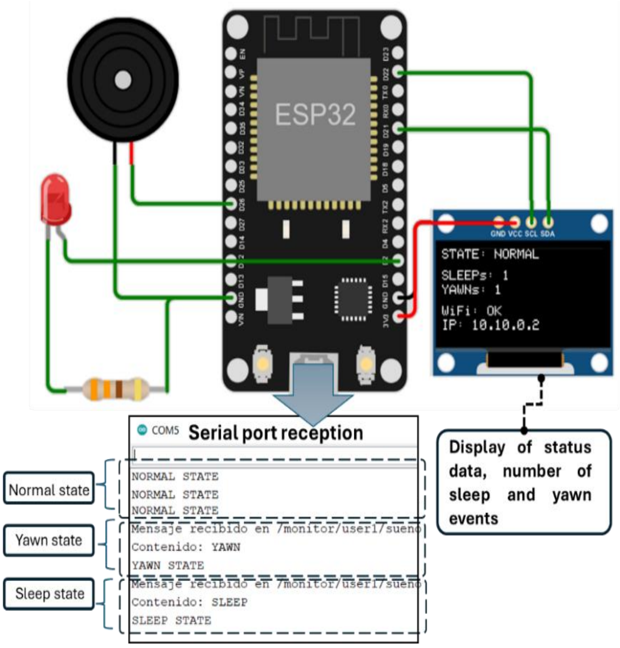

Fig. 13. Operation of the embedded IoT system for alert activation
Source: created by the authors

4.3 Evaluation of the Node-RED Monitoring Platform

This section validates the remote communication and visualization mechanisms using Node-RED and Telegram. Figure 14 shows the dashboard, which includes a drowsiness level gauge, EAR and MAR trend graphs, and a time-stamped log of events. Also shown are the visualization of automatic notifications sent by Telegram, including evidence of the detection of drowsiness or a yawn (Figure 15) and the commands used to access information.

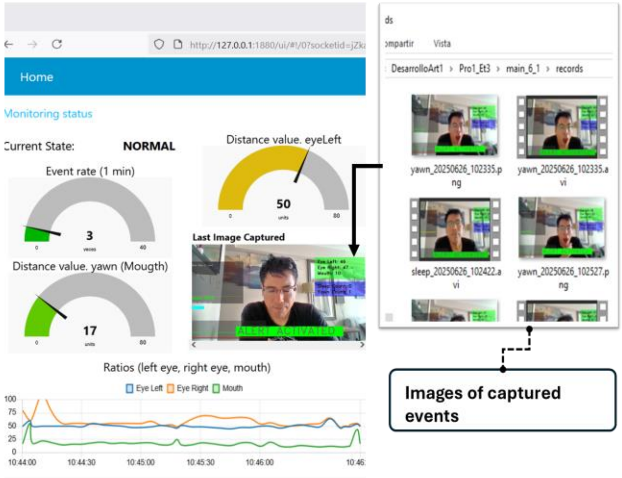

Fig. 14. Web server-based monitoring interface
Source: created by the authors

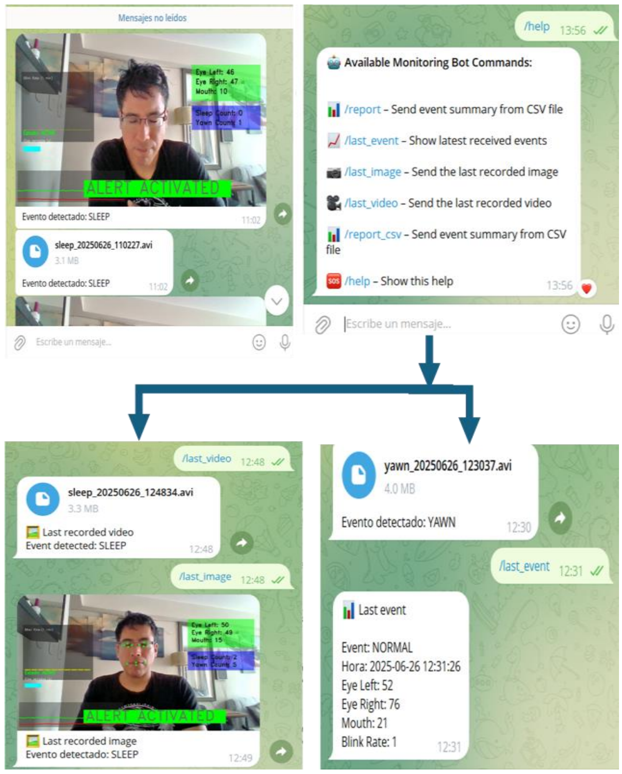

Fig. 15. Receiving alerts in the Telegram application
Source: created by the authors

The implementation of the monitoring platform in Node-Red demonstrates the system's ability to transmit data and images to external applications. Furthermore, its integration with Telegram enabled visual alerts to reach staff who may be serving as caregivers or supervisors, validating its use in telecare, remote monitoring, and e-Health applications.

4.4 Mobile App Evaluation

The results related to the mobile app demonstrate its ability to visualize the driver's alertness locally. Figure 16 shows evidence of the dynamic update of the eye-opening and drowsiness event indicators. An image was also transmitted to Telegram, integrated from the mobile app. These results validate the achievement of the objective of providing a portable, real-time solution for use by drivers.

Furthermore, the mobile app served as a suitable tool to complement the monitoring process. The ability to generate alerts and display drowsiness data on a smartphone screen was accessible to drivers who needed a quick overview. Unlike centralized systems, the use of a mobile device promoted user autonomy, ensuring portability in environments where a fixed monitoring facility is unavailable.

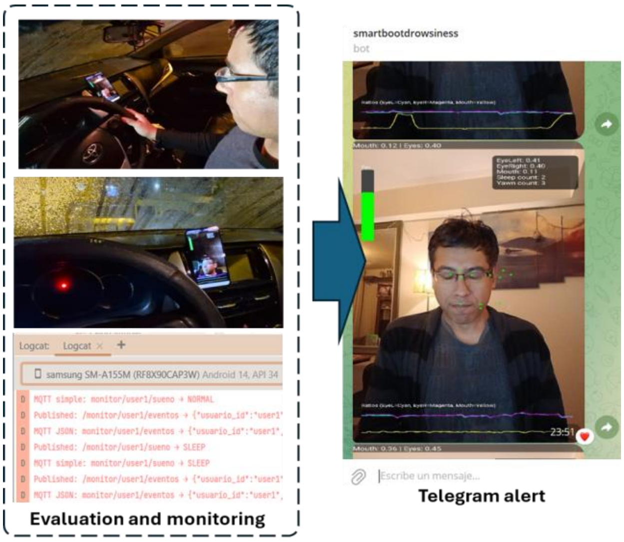

Fig. 16. Mobile app interface for local driver monitoring
Source: created by the authors

5 Conclusion

The developed system allowed for the integration of computer vision for detecting drowsiness using the EAR and MAR indices, an ESP32 embedded device that triggers physical alerts, a remote monitoring platform via Node-RED, and Telegram for distributed supervision. The computer vision model detected facial indicators such as prolonged eye closure and yawning frequency, where experimental results confirmed the effectiveness of the solution.

The Node-RED monitoring platform enabled convenient event visualization via a web console and verification of messages in JSON format. Additionally, the Android mobile app offered a portable interface for local monitoring, validating the system's portability and replicability in telecare applications. Furthermore, integration with the Telegram platform enabled images to be sent to remote users or supervisors, promoting its use in distributed monitoring contexts.

Despite its successful functional validation, the system presents limitations related to illumination variability, computational scalability, and ethical considerations of continuous monitoring. These aspects will be addressed in future work through improved sensing technologies and embedded optimization. On the other hand, from an ethical perspective, continuous monitoring requires guaranteeing privacy and appropriate use of visual information, making it necessary to ensure that processing is done locally to protect user autonomy.

Since this research followed a descriptive and qualitative approach, no quantitative comparison was made with other methods of drowsiness detection, as the contribution focuses on demonstrating the viability of the system.

References:

[1] N. Bienefeld, E. Keller, and G. Grote, “AI Interventions to Alleviate Healthcare Shortages and Enhance Work Conditions in Critical Care: Qualitative Analysis,” *J. Med. Internet Res.*, vol. 27, p. e50852, Jan. 2025, doi: 10.2196/50852.

[2] T. S. Delwar, M. Singh, S. Mukhopadhyay, A. Kumar, D. Parashar, Y. Lee, M. H. Rahman, M. A. S. Sejan, and J. Y. Ryu, “AI- and Deep Learning-Powered Driver Drowsiness Detection Method Using Facial Analysis,” *Appl. Sci.*, vol. 15, no. 3, p. 1102, Jan. 2025, doi: 10.3390/app15031102.

[3] T. Agostinelli, A. Generosi, S. Ceccacci, and M. Mengoni, “Validation of computer vision-based ergonomic risk assessment tools for real manufacturing environments,” *Sci. Rep.*, vol. 14, no. 1, p. 27785, Nov. 2024, doi: 10.1038/s41598-024-79373-4.

[4] A. Giorgi, G. Borghini, F. Colaiuda, S. Menicocci, V. Ronca, A. Vozzi, D. Rossi, P. Aricò, R. Capotorto, S. Sportiello, M. Petrelli, C. Polidori, R. Varga, M. Van Gasteren, F. Babiloni, and G. Di Flumeri, “Driving Fatigue Onset and Visual Attention: An Electroencephalography-Driven Analysis of Ocular Behavior in a Driving Simulation Task,” *Behav. Sci. (Basel).*, vol. 14, no. 11, p. 1090, Nov. 2024, doi: 10.3390/bs14111090.

[5] A. Soultana, F. Benabbou, N. Sael, and S. Bouhsissin, “Driver inattention detection system using multi-task cascaded

convolutional networks," *IAES Int. J. Artif. Intell.*, vol. 13, no. 4, p. 4249, Dec. 2024, doi: 10.11591/ijai.v13.i4.pp4249-4262.

[6] N. A. Othman and I. Aydin, "A New UAV-Based Social Distance Detector for COVID-19 Outbreaks Reduction, Using IoT, Computer Vision and Deep Learning Technologies," *Trait. du Signal*, vol. 39, no. 6, pp. 1951–1959, Dec. 2022, doi: 10.18280/ts.390607.

[7] D. Neogi, M. A. Chowdhury, M. M. Akter, and M. I. A. Hossain, "Mobile detection of cataracts with an optimised lightweight deep Edge Intelligent technique," *IET Cyber-Physical Syst. Theory Appl.*, vol. 9, no. 3, pp. 269–281, Sep. 2024, doi: 10.1049/cps2.12083.

[8] M. A. Khan, T. Nawaz, U. S. Khan, A. Hamza, and N. Rashid, "IoT-Based Non-Intrusive Automated Driver Drowsiness Monitoring Framework for Logistics and Public Transport Applications to Enhance Road Safety," *IEEE Access*, vol. 11, no. 1, pp. 14385–14397, 2023, doi: 10.1109/ACCESS.2023.3244008.

[9] W. Fang, L. Tang, and J. Pan, "AGL-Net: An Efficient Neural Network for EEG-Based Driver Fatigue Detection," *J. Integr. Neurosci.*, vol. 22, no. 6, pp. 1–10, Oct. 2023, doi: 10.31083/j.jin2206146.

[10] H. A. Khalil, S. A. Hammad, H. E. Abd El Munim, and S. A. Maged, "Low-Cost Driver Monitoring System Using Deep Learning," *IEEE Access*, vol. 13, no. 1, pp. 14151–14164, 2025, doi: 10.1109/ACCESS.2025.3530296.

[11] B. S and P. Durgadevi, "A Novel Approach to Driver Negligence Detection: Algorithm With IoT Integration," *Trans. Emerg. Telecommun. Technol.*, vol. 36, no. 2, p. 10, Feb. 2025, doi: 10.1002/ett.70068.

[12] J. Sun, R. Zheng, N. Huang, X. Liu, and J. Xu, "Driver behavior monitoring system for explosive transport," in *Third International Conference on Intelligent Traffic Systems and Smart City (ITSSC 2023)*, W. Shangguan and J. Wu, Eds., SPIE, Apr. 2024, p. 42. doi: 10.1117/12.3023935.

[13] R. Yauri, "IoT Edge Device to Estimate Breathing Rate from ECG Signal for Continuous Monitoring," in *2022 IEEE Engineering International Research Conference (EIRCON)*, IEEE, Oct. 2022, pp. 1–4. doi: 10.1109/EIRCON56026.2022.9934805.

[14] W. Dong, W. Fang, H. Qiu, and H. Bao, "Impact of Situation Awareness Variations on Multimodal Physiological Responses in High-Speed Train Driving," *Brain Sci.*, vol. 14, no. 11, p. 1156, Nov. 2024, doi: 10.3390/brainsci14111156.

[15] R. Kaveh, C. Schwendeman, L. Pu, A. C. Arias, and R. Muller, "Wireless ear EEG to monitor drowsiness," *Nat. Commun.*, vol. 15, no. 1, p. 6520, Aug. 2024, doi: 10.1038/s41467-024-48682-7.

[16] F. Aydin, G. Caruso, and L. Mussone, "An Image Processing-Based Method to Analyze Driver Visual Behavior Using Eye-Tracker Data," *Appl. Sci.*, vol. 14, no. 14, p. 6123, Jul. 2024, doi: 10.3390/app14146123.

[17] S. J, "Smart attendance monitoring system using multimodal biometrics," *Sigma J. Eng. Nat. Sci. – Sigma Mühendislik ve Fen Bilim. Derg.*, vol. 43, no. 1, pp. 168–188, Feb. 2025, doi: 10.14744/sigma.2024.00030.

[18] M. Kelley, M. Tiede, X. Zhang, J. A. Noah, and J. Hirsch, "Spatiotemporal processing of real faces is modified by visual sensing," *Neuroimage*, vol. 312, May 2025, doi: 10.1016/J.NEUROIMAGE.2025.121219.

[19] E. Temitayo, Y. Thomas, K. Ayodele, and A. Ogunseye, "Implementation of an On-board Embedded System for Monitoring Drowsiness in Automobile Drivers," *Int. J. Technol.*, vol. 9, no. 4, p. 819, Jul. 2018, doi: 10.14716/ijtech.v9i4.1691.

[20] V. Kumar and K. P. Sharma, "Alert based drowsiness detection using machine learning," in *AIP Conference Proceedings*, American Institute of Physics Inc., Oct. 2022, p. 040001. doi: 10.1063/5.0125988.

[21] H. Zia, I. ul Hassan, M. Khurram, N. Harris, F. Shah, and N. Imran, "Advancing Road Safety: A Comprehensive Evaluation of Object Detection Models for Commercial Driver Monitoring Systems," *Futur. Transp.*, vol. 5, no. 1, p. 2, Jan. 2025, doi: 10.3390/futuretransp5010002.

[22] A. O. Onososen, I. Musonda, D. Onatayo, A. B. Saka, S. A. Adekunle, and E. Onatayo, "Drowsiness Detection of Construction Workers: Accident Prevention Leveraging Yolov8 Deep Learning and Computer Vision Techniques," *Buildings*, vol. 15, no. 3, p. 10, Feb. 2025, doi: 10.3390/buildings15030500.

[23] M. De Vincenzi, J. Moore, B. Smith, S. E. Sarma, and I. Matteucci, "Security Risks and Designs in the Connected Vehicle Ecosystem: In-Vehicle and Edge Platforms," *IEEE Open*

J. Veh. Technol., vol. 6, no. 1, pp. 442–454, 2025, doi: 10.1109/OJVT.2024.3524088.

[24] V. Chowdary, A. Sharma, N. Kumar, and V. Kaundal, *Internet of Things in Modern Computing*. Boca Raton: CRC Press, 2023. doi: 10.1201/9781003407300.

[25] B. B. Gupta, A. Gaurav, K. Tai Chui, and V. Arya, "Deep Learning Model for Driver Behavior Detection in Cyber-Physical System-Based Intelligent Transport Systems," *IEEE Access*, vol. 12, no. 1, pp. 62268–62278, 2024, doi: 10.1109/ACCESS.2024.3393909.

[26] E. Michelaraki, C. Katrakazas, S. Kaiser, T. Brijs, and G. Yannis, "Real-time monitoring of driver distraction: State-of-the-art and future insights," *Accid. Anal. Prev.*, vol. 192, no. 1, p. 107241, Nov. 2023, doi: 10.1016/j.aap.2023.107241.

Contribution of Individual Authors to the Creation of a Scientific Article (Ghostwriting Policy)

The authors equally contributed in the present research, at all stages from the formulation of the problem to the final findings and solution.

Sources of Funding for Research Presented in a Scientific Article or the Scientific Article Itself

The authors express their special gratitude to the Universidad Tecnológica del Perú (UTP) for the valuable support provided through the allocation of material resources as part of the 2025 Research project.

Conflict of Interest

The authors have no conflicts of interest to declare.