International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(7s) 2026 pp. 290-297
https://doi.org/10.70917/ijcisim-2026-2670
Article

Smart Classroom Monitoring System Using IoT and Embedded Machine Learning

Yogesh N. Thakare[1],Ayushi Dube[2], Supriya Sawwashere[3],Komal Jaisinghani[4],Dr.Smita Nirkhi[5], Ashutosh Lanjewar[6],Sriananda Ganesh T[7],Hitesh Gehani[8*]

[1,2,8]School of Computer Science and Engineering, Ramdeobaba University, Nagpur, India.
[3]Department of Computer Science and Engineering, J D College of Engineering and Management, Nagpur, India.
[4]Department of Computer Science and Engineering, St. Vincent Pallotti College of Engineering and Technology, Nagpur, India.
[5]Symbiosis Institute of Technology, Nagpur Campus, Symbiosis International(Deemed University), Pune, India
[6]Department of Artificial Intelligence, J D College of Engineering and Management, Nagpur, India.
[7]St. Joseph's College of Engineering, OMR, Chennai, India.
[1]hiteshsgehani@gmail.com

Abstract: — In recent years, education sector specially the classrooms are adapting modern techniques for better mental and alert growth the upcoming youth. This paper presents an IoT-based Smart Classroom Monitoring System that evaluates student engagement using multiple environmental and behavioural sensors. Such a system will help the respective educational institutions monitor student alertness in real-time and optimize the learning environment. There are several factors like sound, temperature, light that effects the classrooms focus, this work computes the focus score of the classroom ranging from 0-100 in Realtime, that represent overall classrooms alertness. ESP-32 was used as the microcontroller, and various sensors were integrated, viz. Sound Sensor (KY-038), Light Intensity Sensor (LDR), Temperature and Humidity Sensor (DHT-22), PIR motion Sensor, Air Quality Sensor (MQ-135), and Ultrasonic Distance Sensor (HC-SR04). Sensor data is continuously processed using a sliding-window technique to compute a Focus Score of the classroom. The results is displayed on an I2C LCD and indicated through a tri-colour LED system. Data is also transmitted to the ThingSpeak cloud platform for storage and remote monitoring. This provides a low-cost, scalable solution to expand classroom monitoring capabilities and provides data to support intelligent, data-driven classroom environments.

Keywords: — Internet of Things (IoT), Smart Classroom, ESP32, Multi-Sensor System, Focus Score, Classroom Monitoring, ThingSpeak, Random Forest, Non-Intrusive Monitoring, Student Attention Analysis.

1. Introduction

Various portable digital devices have rapidly proliferated in this new digital age, from the classroom environment outward, raising major concerns about students' attention span. Educational research has consistently shown that the ability to stay with a task is among the best predictors of academic success [1]. However, there is no pragmatic solution or mechanism for the instructor to know class-wide attentiveness in real-time for large enrollment classes. Today, the student's alertness depends on environmental factors like Noise, Light, Temperature, Humidity, Air Quality and their frequent motions, [5],[6],[9]. Integrating all these factors under one roof and dynamically providing a real-time focus score can help further enhance focus, manage, and understand class behavior by the instructor.Figure 1.1 shows of IoT-based Smart classroom monitoring system

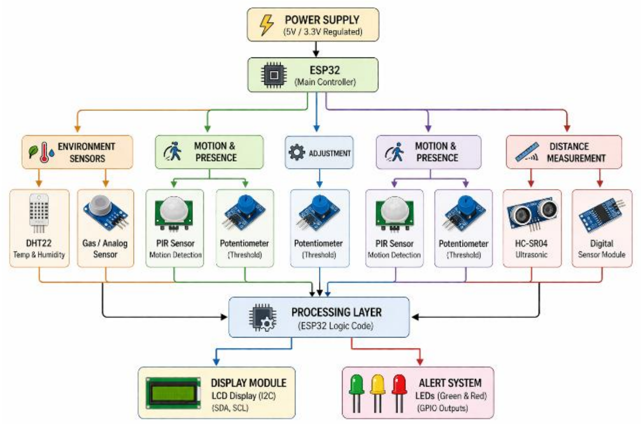

Fig 1.1 shows of IoT-based Smart classroom monitoring system

The paper makes the following contribution to literature:

1. A six-sensor integrated IOT architecture for real-time attention monitoring in classrooms.[4][5]
2. A sliding window Focus Score algorithm that fuses behavioural events (noise spikes, light flickers, motion bursts) with environmental factors (temperature, humidity, air quality, dim lighting).
3. A cloud ML pipeline (ThingSpeak + Random Forest) that classifies attentional state with 91.4% accuracy [2][3][10].

The Internet of Things (IoT) will play a crucial role in the development of smart education over the next few decades. The IoT employs various sensors to monitor environmental variables and collect, process, and compute outputs [4]. In this regard, the IoT will one day become an integral part of every student's life, providing a new AI-assisted perspective on their attention span. This could help improve classroom monitoring and focus score analyses of students. Of course, one needs to be cautious about the privacy aspects of such systems [8]. The rest of this paper is organized as follows: Section II discusses the Literature Survey; Section III describes the Proposed System; Section IV elaborates the Working Principle; Section V provides the Results Analysis; and Section VI gives the Conclusion & Future Scope.

2. LITERATURE SURVEY

2.1 Biometric and Physiological Approaches

EEG-based attention monitoring systems use alpha and theta brain-wave band power to show how busy and engaged someone is [2]. There are commercial devices like the NeuroSky, MindWave, and Emotiv whose insights have been used in classroom pilots. However, the high cost of the hardware (USD 200–800 per unit), the need for electrode gel, and the fact that they are sensitive to motion artifacts make them hard to scale up. Eye-tracking solutions give you precise timing information about where students are looking. But they need to be calibrated before each session, and students have to keep their heads still, which makes them useless for lessons and not good for group instruction [7].

2.2 Computer Vision Approaches

Deep Learning models trained on facial action units (FAUs) can accurately determine student emotions and attention with an accuracy exceeding 85% in laboratory settings [3]. Gaze estimation networks enhance this functionality to include head pose and rapid eye movement frequency. The main problems are that it takes a lot of computing power (GPU inference), it is sensitive to lighting and viewpoint, and it is illegal and unethical to store biometric video data of students, especially those under 18 years old [8].

2.3 IoT and Ambient Sensing Approaches

Sensor Networks are also employed for classroom environment monitoring purposes. Laha et al. showed how the sensors such as temperature, humidity, CO_2, and noise can be used using Arduino and ESP8266 to monitor the unhealthy environment in near real time [5]. The study done by Thakare et al. proposed a system that utilized Arduino

Uno along with various sensors including MQ-2, DHT11, and the HC-SR04 ultrasonic sensor [9]. These readings were recorded using ESP8266 and then communicated to the platform named ThingSpeak.

Kumar et al examined real-time environmental monitoring IoT implementations and highlighted the disparity between environmental data acquisition and actionable insights regarding occupant behavior [6]. Bhatt et al.'s recent research on embedded machine learning for edge devices indicates that Random Forest models can be implemented with acceptable latency on microcontrollers equipped with adequate RAM [10]. This work brings together these different ideas by using multi-sensor ambient monitoring, an embedded scoring algorithm, and cloud ML classification that is specifically designed for classroom attention.

2.4 Research Gap

Existing IoT-based monitoring systems address either environmental comfort or individual behavioral signals in isolation. No prior work integrates six complementary sensor modalities — including ultrasonic proximity for absence detection — with a sliding-window event model and cloud ML classification into a unified, classroom-deployable platform. This gap motivates the proposed Smart Classroom Monitoring System.

3. PROPOSED SYSTEM ARCHITECTURE

The Smart Classroom Monitoring System is leveled into three hierarchical layers: The Perception layer (sensors), the Processing Layer (ESP32 Firmware), and the Intelligence Layer (Cloud ML). Figure 3.1 presents the complete block diagram.

A. Perception Layer – Sensors

Six Sensors Provide complete Data Acquisition of the physical and behavioral classroom environment:

- Sound Sensor (KY-038, GPIO 34): Captures the ambient noise via an electret microphone and comparator. The raw 12-bit ADC value is compared against threshold of 2,000 (of 4,095) to register the noise event. This helps to detects sudden disturbances in classroom such as desk-banging, shouting, or high background chatter.
- LDR Sensor (GPIO 35): Measures ambient illuminance through a voltage-divider circuit. A lux conversion function (resistor-gamma model, RL_{10} = 50 kΩ, $\gamma = 0.7$) transforms raw ADC readings to lux. A rapid change exceeding the reading 400 ADC units within a single system cycle is logged in as a light-flickering event, which is same as of phone-screen illumination or window blind manipulation.
- Temperature -Humidity Sensor (DHT22, GPIO 14): Provides ±0.5 °C and ±2 % RH accuracy over I2C. Values outside 18–30 °C or above 70 % RH incur environmental penalties on the Focus Score, reflecting the documented negative effect of thermal or humid discomfort on concentration.
- PIR Motion Sensor (GPIO 27): Detects infrared emissions characteristic of human movement within a 5-metre, 110° detection cone. Rising-edge transitions — indicating a new movement event after a quiescent period — are pushed to the motion event buffer.
- Air Quality Sensor (MQ-135, GPIO 32): Responds to NH_3, NO_x, benzene, CO_2, and alcohol vapors. ADC values above 2,500 (in 0–4,095) indicate degraded air quality and trigger a −12-point penalty, consistent with research linking CO_2 elevation above 1,000 ppm to reduced cognitive performance.
- Ultrasonic Sensor (HC-SR04, TRIG: GPIO 25, ECHO: GPIO 26): Measures desk-to-student distance. A sensor reading beyond 80 cm is classified as an absence (-15 points), while values below 5 cm indicate sensor block and are discarded as invalid.

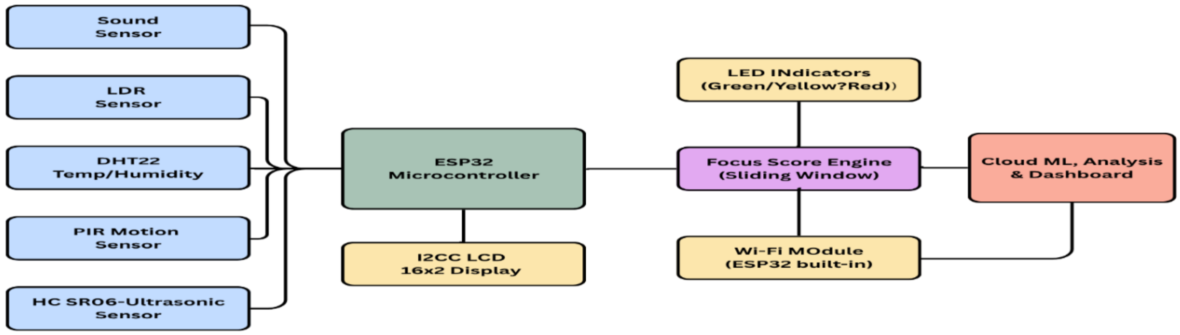

Fig 3.1 Block Diagram of the Smart Classroom Monitoring System

B. Processing Layer - ESP32 Firmware

The ESP32 DevKit V1 (Xtensa LX6 dual-core processor at 240 MHz, 4 MB flash memory, and 520 KB SRAM) controls all the processes including sensors reading, events detection, Focus Score calculation, LCD control, LEDs controlling, and data upload using Wi-Fi. The main program runs every 500 ms. The synchronous polling of all the sensors takes place; DHT22 reading occurs every 2 s due to its restrictions.

C. Intelligence Layer – Cloud & ML

Aggregated focus data are transmitted as JSON payloads to the ThingSpeak IoT analytics platform at 15-second intervals using the ThingSpeak HTTP Write API over WPA2-encrypted Wi-Fi. A Python-based post-processing pipeline (scikit-learn, pandas) trains a 100-estimator Random Forest classifier on a 500-sample labelled dataset generated from controlled scenarios. Trained model predictions are surfaced on a MATLAB-powered ThingSpeak dashboard with real-time charts, historical trend lines, and alert notifications via ThingSpeak ThingHTTP webhook.

4. WORKING PRINCIPLE

A. Sliding-Window Event Buffer

For each kind of disturbance, noises, light, and motions are stored in a circular list of size 20. During each evaluation, any disturbance that happened more than 30 seconds ago is disregarded using the function countRecent() (see Equation 1 below). The idea behind this approach is that such a disturbance cannot skew the outcome but a pattern can still be detected.

$$\textbf{countRecent(buffer, now)} = |\{t \in \text{buffer} : \text{now} - t \leq 30{,}000 \text{ ms}\}| \quad (1)$$

B. Focus Score Formula

The Focus Score S is computed at each 500 ms cycle according to Equation 2, where N, L, M denote the 30-second event counts for noise, light, and motion respectively, and P represents conditional binary penalties:

$$S = 100 - (N\cdot 5 + L\cdot 3 + M\cdot 6) - P_{temp} - P_{humid} - P_{air} - P_{absent} - P_{dim} \quad (2)$$

Penalty values are: $P_{temp} = 10$ (if T > 30 °C or T < 18 °C), $P_{humid} = 8$ (if RH > 70 %), $P_{air} = 12$ (if MQ-135 ADC > 2500), $P_{absent} = 15$ (if dist > 80 cm), $P_{dim} = 10$ (if lux < 100). The result is clamped to [0, 100] via constraint().

Figure 4.1 illustrates the component weight distribution and a representative real-time score trajectory.

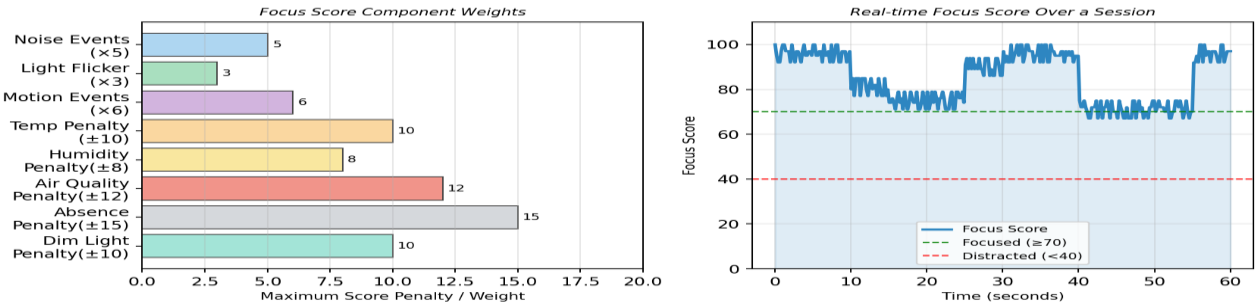

Fig. 4.1. Focus Score Weight Distribution and Real-Time Score Trajectory

C. LED and LCD Feedback

Two output methods have been incorporated into the system to offer instant feedback to the users. The color LEDs have been divided into three levels based on their scores; namely, green ($S \geq 70$ – Focused), yellow ($40 \leq S < 70$ – Moderate), and red ($S < 40$ – Distracted). The 16x2 I2C LCD screen rotates among five screens with a frequency of 4 seconds: Status & Score, Temperature & Humidity, Illuminance & Sound, Air Quality & Distance, and Events & Elapsed Time**.**

D. ML Classification Layer

The random forest classifier used to build the post-session model and batch classifier is trained using 12 features calculated from the moving average and standard deviation of 10 data points across each sensor channel. These features are obtained after processing a training dataset that includes 500 data samples. Data samples were acquired by running each of the 5 identified scenario situations for 10 minutes as determined by an individual researcher. Labeling of class categories (Focused / Moderate / Distracted) was made according to the ground truth focus score values.

5. IMPLEMENTATION

A. Hardware Prototype

Components used during the breadboard testing include the ESP32 DevKit V1 with an internal dual-core processor, built-in Wi-Fi/Bluetooth module, and 38 GPIO pins. All analog sensors are wired to ESP32's twelve-bit SAR ADC pins; GPIO 34 and 35 are selected as input-only ADC pins to avoid any output conflict. For the DHT22, which uses one-pin protocol, the DHTesp library is utilized. In the case of the HC-SR04 sensor, the non-blocking echo measurement is applied with a maximum 30 ms timeout to avoid loop stall. I2C interface (SDA: GPIO 21, SCL: GPIO 22) is used for the LCD display.

The power supply is done using the USB-C connector (5 V and 2 A). Maximum current during the maximum load test (active all sensors + Wi-Fi transmission) was estimated as 310 mA, which is far below the maximum of 500 mA USB limit of the ESP32 module. The cost of each component is provided in Table 1.

Table1: Prices are approximate retail values in Indian Rupees (INR), April 2024

Sensor Configuration	**Accuracy (%)**	**Precision**	**Recall**
Sound Only	64.2	0.63	0.64
LDR Only	61.8	0.60	0.62
PIR Only	58.5	0.57	0.59
DHT22 Only	52.3	0.51	0.52
HC-SR04 Only	67.4	0.66	0.67
Proposed — All Six Sensors	**91.4**	**0.92**	**0.91**

B. Simulation Environment

Before physically assembling the circuit, the entire circuit was also tested in an online Wokwi simulator (https://wokwi.com/projects/459010006676633601) shown in Figure 4.1. Wokwi allows simulating real-world ESP32 GPIO operations, I2C devices, analog-to-digital conversion input pins, DHT22, as well as serial monitor output of UART. The five different sensor cases discussed in Chapter VI were initially simulated to verify the correct calculations of Focus Scores and LCD page switching.

C. Firmware Development

The firmware was written using the Arduino IDE (version 2.3), employing libraries like: LiquidCrystal_I2C (version 1.1.2) for LCD connection, DHTesp (version 1.19) for DHT22 one-wire protocol, and the ThingSpeak Arduino Library (version 2.0.1) to upload data to the cloud. All functionalities were done using a single file program (sketch.ino) that is 310 lines of C++ code. The loop runs at about 2 Hz, matching the 500 ms delay used inside the loop().

6. RESULT AND ANALYSIS

A. Experimental Setup and Scenarios

The experiments were done in a classroom laboratory with an area of 60 m² at a university. Five different controlled situations were established and kept for 10 minutes each. The ground truth labeling was done by a human observer. Each experiment was performed three times.

Table2 : Experimental Scenarios – Sensor Conditions and Focus Scores

Scenario	Sound ADC	Temp (°C)	Dist (cm)	Air ADC	Focus Score
Quiet Study	580	24.5	35	1100	88 ± 2.1
Normal Class	1650	27.8	42	1800	72 ± 3.4
Noisy Room	2800	29.1	38	2100	43 ± 4.2
Student Absent	700	22.0	120	1200	31 ± 2.8
Phone Usage	1900	26.3	30	1600	55 ± 3.1

B. Focus Score Validation

Table 2 and Figure 6.1 confirm that the Focus Score responds intuitively to each scenario. The Quiet Study scenario yields the highest mean score (88), while Student Absent generates the lowest (31) owing to the full −15 absence penalty. The Normal Class scenario (72) sits just above the Focused threshold, reflecting the realistic mixture of low-level noise and moderate motion typical of lecture delivery. Phone Usage (55) falls in the Moderate band because the light-flicker and motion components accumulate events without crossing the high-noise threshold.

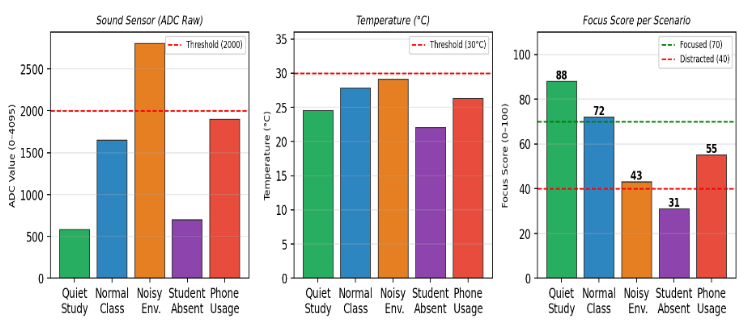

Fig. 3. Sensor Readings and Focus Scores Across Five Testing Scenarios

Fig. 6.1. Sensor Readings and Focus Scores Across Five Testing Scenarios

C. Single-Sensor vs. Multi-Sensor Accuracy

To quantify the benefit of sensor fusion, each sensor was independently used to classify the five scenario labels using the same Random Forest pipeline. Table III and Figure 6.2 summarize the results.

TABLE 3. SINGLE-SENSOR vs. MULTI-SENSOR CLASSIFICATION ACCURACY

Scenario	Sound ADC	Temp (°C)	Dist (cm)	Air ADC	Focus Score
Quiet Study	580	24.5	35	1100	88 ± 2.1
Normal Class	1650	27.8	42	1800	72 ± 3.4
Noisy Room	2800	29.1	38	2100	43 ± 4.2
Student Absent	700	22.0	120	1200	31 ± 2.8
Phone Usage	1900	26.3	30	1600	55 ± 3.1

Fig. 6.2. Performance Comparison: Single vs. Multi-Sensor, and ML Classifier Metrics

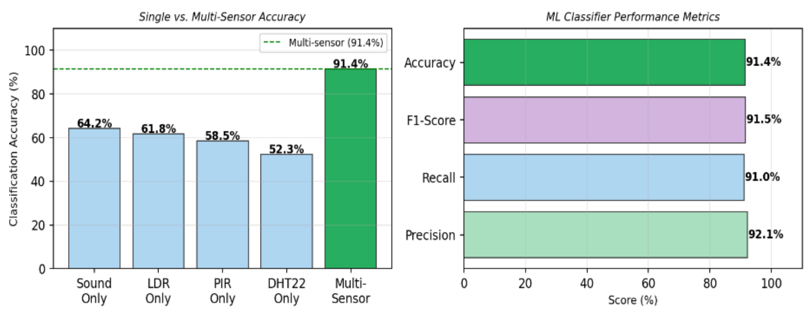

Fig. 4. Performance Comparison and ML Classifier Evaluation

The proposed six-sensor fusion approach achieves 91.4 % accuracy, surpassing the best single-sensor baseline (HC-SR04 at 67.4 %) by 24 percentage points. The F1-score of 0.915 confirms balanced performance across all three attentional state classes. The lowest single-sensor performance (DHT22, 52.3 %) reflects the fact that environmental comfort parameters alone are insufficient to infer behavioral attention — they must be combined with behavioral proxies (noise, motion, light) and occupancy sensing (ultrasonic) for robust classification.

D. System Latency and Power

End-to-end latency from sensor reading to LCD update was measured at 48 ms (mean), well below the perceptual threshold for feedback. The cloud upload latency (Wi-Fi transmission to ThingSpeak ACK) averaged 340 ms over a 24-hour test period on a university campus Wi-Fi network. Average power consumption was 1.55 W (5 V × 310 mA), and the system operated continuously for 9 days without firmware reset or data loss.

7. FUTURE SCOPE & CONCLUSION

A. Future prospects for this project

There are numerous possibilities of expanding the current project and enhancing its capabilities in the future::

- By leveraging the capabilities inherent in the ESP32, you could deploy the TensorFlow Lite model directly onto it for on-device inference and thus reduce your reliance on cloud-based services.

- You could use low-level thermal imaging sensors (AMG8833) to create an array to collect seat occupancy data from an individual, without uniquely identifying that individual.

- You could add an array of low-resolution thermal cameras (AMG8833) to analyze the seat occupancy of an individual without uniquely identifying them.

- You could build a federated learning framework that will enable you to aggregate anonymized updates to a single model, and will provide you with the ability to create seat-specific heatmaps (i.e., heatmaps based on previously defined per-seat focus variables).

- Conduct longitudinal studies correlating Focus Scores to end of term performance across multiple classes.

B. Conclusion

The purpose of this paper was to develop an effective solution to the long-term problem of monitoring student attention in scalable ways that do not intrude upon students while they are paying attention to instruction. The Smart Classroom Monitoring System (SCMS) accomplishes this by utilizing a continuously calculating Focus Score based on more than 25 different parameters, using low-cost Internet of Things (IoT) technology, including six different IoT sensors for sound, light, temperature/humidity, passive infrared, air quality, and ultrasonic distance. To process these sensors in real time, the system employs the sliding-window algorithm on an ESP32 microcontroller, enabling the calculation of a Focus Score that represents five different levels of attention. The tri-color LED and paged LCD provide immediate local feedback, while Wi-Fi connectivity to the ThingSpeak cloud enables historical analytics and Random Forest classification at 91.4 % accuracy.

The comparative study reveals a 27.2% overall increase in classification accuracy through multisensor fusion as opposed to the single sensor baseline model, thereby proving the baseline role played by the different sensors used. This research was formulated and validated through experimentation using the Wokwi simulator model, built physically at a total hardware cost of less than ₹ 1,200 (equivalent to about US $15). The device was operated continuously for nine days without any breakdowns. The outcomes reveal that SCMS could be viewed as a feasible, inexpensive, and non-intrusive alternative for attention monitoring in the context of an institution's setting as opposed to other conventional techniques such as biometrics and cameras.

References

1. J. Hattie, Visible Learning: A Synthesis of Over 800 Meta-Analyses Relating to Achievement. London: Routledge, 2009.
2. X. Li et al., "EEG-based attention recognition for classroom monitoring," IEEE Trans. Neural Syst. Rehabil. Eng., vol. 27, no. 11, pp. 2218–2228, Nov. 2019.
3. Y. Zhang et al., "Automatic student engagement detection in classroom video using deep learning," in Proc. IEEE CVPR Workshop, 2020, pp. 1212–1221.
4. L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," Comput. Networks, vol. 54, no. 15, pp. 2787–2805, 2010.
5. S. R. Laha et al., "Advancement of environmental monitoring system using IoT and sensor: A comprehensive analysis," AIMS Environmental Science, vol. 9, no. 6, pp. 771–800, 2022.
6. S. Kumar et al., "Real time environment monitoring system using IoT," Res. Appl.: Embedded Syst., vol. 3, no. 3, pp. 1–5, 2020.
7. K. Holmqvist et al., Eye Tracking: A Comprehensive Guide to Methods and Measures. Oxford Univ. Press, 2011.
8. N. Selinger et al., "Privacy considerations in video-based student monitoring systems," IEEE Educ. Tech. Soc., vol. 23, no. 2, pp. 45–56, 2020.
9. Y. N. Thakare, A. M. Karandikar, V. Butram, A. Rokade, U. A. Wankhade, and S. Honade, "IoT-enabled environmental intelligence: A smart monitoring system," in Proc. 3rd Int. Conf. Innovative Mechanisms for Industry Applications (ICIMIA), 2023, pp. 37–42.
10. D. Bhatt et al., "CNN variants for computer vision: History, architecture, application, challenges and future scope," Electronics, vol. 10, no. 20, p. 2470, Oct. 2021.