Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 47 No. 01 (2026)

Implementation of a Facial Recognition Attendance System using Raspberry Pi

Yusuf O. Isaac[1]*, Ariba O. Folashade [2], Onyemenam O. John [3], Adeyemi A. Tobiloba [4], Ochuma J. Enemona [5], Obayemi T. Adeniyi [6]

[1,2,3,4,5,6] *Department of Electrical and Information Engineering Landmark University SDG 9 (Industry, Innovation and Discoveries), Omu-Aran, Kwara State, Nigeria,*

Abstract: Manual attendance systems, such as manual sign-in sheets in academic institutions, are often error-prone, time-consuming, and can be easily compromised. Fingerprint biometric systems can pose hygiene risks due to shared contact surfaces. This motivated the development of an automated, contactless facial recognition-based attendance system to enhance accuracy, efficiency, and hygiene in educational settings. This project aimed to design and implement a system that uses facial recognition technology to accurately log student attendance in real time, while leveraging the use of the Raspberry Pi. The system integrates a Raspberry Pi Camera Module for capturing facial images, a 16x2 I2C LCD for displaying recognized names and matric numbers, and a buzzer for auditory confirmation. Using Python, the system was built with key libraries such as face recognition, dlib, and OpenCV. A dataset comprising 60 students, each with about 15 facial images, was used for training. Facial features were extracted using 128-dimensional embedding from dlib and classified with a Support Vector Machine (SVM). The system was tested under different lighting and angles, which resulted in producing an accuracy range of 83–96%, an average recognition time of 6 seconds, a false acceptance rate (FAR) of 15%, and a false rejection rate (FRR) of 10%. This paper is subdivided into 6 sections. Section 1 is the introduction, Section 2 is the literature review, Section 3 is Methodology, Section 4 results and discussion, Section 5 recommendations, Section 6 conclusion.

Keywords: *Facial Recognition, Attendance System, Biometrics, Raspberry Pi, OpenCV, SVM*

1. Introduction

In today era of an amazing convergence of emerging technological innovations, present a chance to tackle the problems that prevail in attendance capturing of students [1]. In many actual and practical systems, attendance is thought to have a central role in forecasting the academic performance of students [2] [3] [4] [5]. That is why, educational institution has mandated lectures to record and monitor student's attendance [6]. The traditional process of taking students' attendance manually with the use of an attendance sheet by a faculty member in each class is a tedious process particularly in a huge class. These processes have introduced unnecessary crowd, wastage of manpower, and tardiness [7]. Due to the ease in applying traditional approaches, technology such as image processing and biological attendance records proves easier to implement [8]. The implementation of biometrics identity authentication techniques such as, voiceprints, iris scanning, fingerprints and facial recognition over the last few years, founded on biological principles has gained wider popularity due to ease and anonymity [9] [10] [11] [12]. Notwithstanding, in computer vision during the last decades, face recognition has been the most essential use of biometric-based systems [13].

Although other identification processes may be more accurate, face recognition has always been a great area of research because it does not require any physical invasion or touch [14]. Face recognition devices are gradually becoming a pan biometric solution due to the fact that they require almost zero effort on the part of the user relative to other biometric modalities. Facilitated by inventions in Artificial Intelligence (AI), computer vision, and deep learning, face recognition devices can present identification in actual detection with high reliability today [15] [16]. The production of face recognition device on low-cost and small-scale hardware platforms like the Raspberry Pi has further increased their applications in classroom attendance systems [17], [18].

Therefore, the paper's primary contribution is a low-cost, Raspberry Pi 4-based facial-recognition attendance system that balances high accuracy with fast processing, filling a gap where few studies combine this hardware platform with a lightweight HOG + SVM pipeline. It implements a complete end-to-end solution camera module, I²C LCD, buzzer, and Python-based software (OpenCV, dlib, face-recognition library) trained on 60 students (≈15 images each) to produce 128-dimensional embedding and classify faces with a linear SVM The system achieves **85 %–95 % accuracy**, a 5 % false-acceptance rate, and recognition latency of 5.4–6.3 s, outperforming earlier Raspberry Pi 3 implementations that reported 10–12 s latency (Sharma & Patel 2021) and matching or exceeding desktop-based LBP + SVM (80–85 % accuracy, 7–9 s) while remaining portable. By selecting HOG features instead of deep CNNs, the approach reduces computational load, enabling real-time operation on the modest Pi 4 processor without sacrificing robustness. The work provides a design flowchart and modular architecture (hardware configuration, software stack, integration) that can be readily replicated or extended, and it outlines concrete upgrades (higher-resolution camera, Pi 5) to further improve performance. Overall, the study demonstrates that a carefully engineered, inexpensive Raspberry Pi platform coupled with classical computer-vision descriptors can deliver attendance-tracking performance comparable to more resource-intensive solutions, thereby expanding the practical applicability of biometric attendance systems in educational settings. This paper is structured as follows. Section 2 provides the literature review. The methodology is discussed in Section 3. In Section 4, the results discussions are presented. Section 5 concludes the study.

The review of core concepts and real-world use of facial recognition in attendance systems. It highlights challenges such as reduced accuracy due to facial variation, lighting, and poor angles. Privacy concerns also emerge from the risk of unauthorized access to stored biometric data. These findings help identify gaps in current systems and guide the development of improved solutions. For instance, [19], [20] proposed the integration of Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) models for face recognition to monitor online and offline education attendance. The proposed approach aims to increase accuracy by considering both temporal and spatial data. From the experimental results, the proposed approach gave higher accuracy when compared with the existing approach, indicating that the system will be able to monitor the attendance of student effectively, thereby providing accurate data for the institution. [21] focus on applying a webcam with the Viola-Jones method for face recognition that monitors class attendance in real-time. The method resizing the face detected to actual size, uses basic Local updates the collected recognised attendance and important information into a database referred to as SQLite, helping the institution to reduce time and error sin the attendance process.

[22], [23] demonstrated the development of an attendance face recognition system for companies' employees and time management attendance with the aid of Machine Learning (ML) and Artificial Neural Networks (ANN) models. The authors developed a facial recognition system model via an ANN architecture and a Raspberry Pi. The ANN architecture consists of a malfunctioning back propagation method to prompt knowledge from input data. The algorithms are coded in Python, and the results are highly precise in identifying and categorizing people. This study was able to leverage AI, ML, and ANN innovations to create an effective time control system and attendance list for companies.

[24] proposed Haar Cascade algorithms attendance management system for face detection and LBPH for face recognition, real-time checking, and generation of attendance. The proposed system identifies individual faces positively by extracting and analyzing unique facial features. These detected faces are then matched with saved faces in the database, with confidence values determining identity. Also, the authors incorporate a simple-to-use interface allowing administrators to manage students' records and attendance easily.

[25] demonstrated a face recognition attendance technology based on an Android device. The system captures students' faces, generates and displays a QR code that has all the course data on their mobile phone. The outcome of the demonstrated system depicts an accuracy of 97.29% face recognition was derived using linear discriminant analysis. Also, [23] developed an improved attendance management system that combines GPS and facial recognition innovations in academic institutions. The developed system is able to enhance security, accuracy, and attendance pattern insight by performing two-step verification.

[27] developed a Local Binary Pattern (LBP) algorithm combined with advanced image processing techniques to solve the problem of the accuracy of face recognition. The authors employed image processing methods to enhance LBP codes, namely: Image Blending, Bilateral Filter, Contrast Adjustment, and histogram equalization. The study shows, the system developed in be implemented in real real-world automatic attendance scenario.

[28] Implemented a reliable attendance face recognition mechanism for effective attendance monitoring than the traditional approach. The study highlighted factors that can affect fingerprint and RFID systems, namely, injury, moisture, and ID card mishandling. To address these factors in places with little data, the authors make use of MTCNN for face detection, FaceNet for feature extraction, and Support Vector Machine (SVM) for classification.

[29] developed an automatic lecture attendance facial recognition technology to address the problem of traditional methods of marking attendance. The proposed model includes features such as security interaction, accurate monitoring with the facial recognition, and continuous tracking. The authors demonstrated that the proposed method limits human mistakes and improves dependability

[30] suggested a student attendance system for schools and colleges using Face Recognition along with Geo-Fencing based on Google Play services, location services, Firebase, and Geofire. The system tracks the real-time location of students within a classroom geo-fenced, and identifies them as attended if they have attendance that is over 90% of the class time. The authors developed the system in Android Studio IDE and integrated the geolocation-based tracking with face recognition of preregistered students. Also, a questionnaire survey and responses were carried out and analyzed using SPSS. The findings stated that the system significantly enhanced identification, verification, and accuracy of attendance records.

In all the work reviewed, only a few authors work on the implementation of Raspberry Pi and SVM for the facial recognition of the attendance system. Therefore, this research aims to address these gaps by developing an SVM-based Raspberry Pi face recognition system for attendance monitoring and recording, designed specifically for educational institutions.

2. Methodology

This section covers the hardware and software setups used in creating the facial recognition attendance system, as well as the integration of all components to enable real-time attendance tracking. The subsections of this chapter are section A- Hardware configuration, section B- Software configuration, and section C- Integration of the system. Below is the design flow chart and the step-by-step procedure carried out.

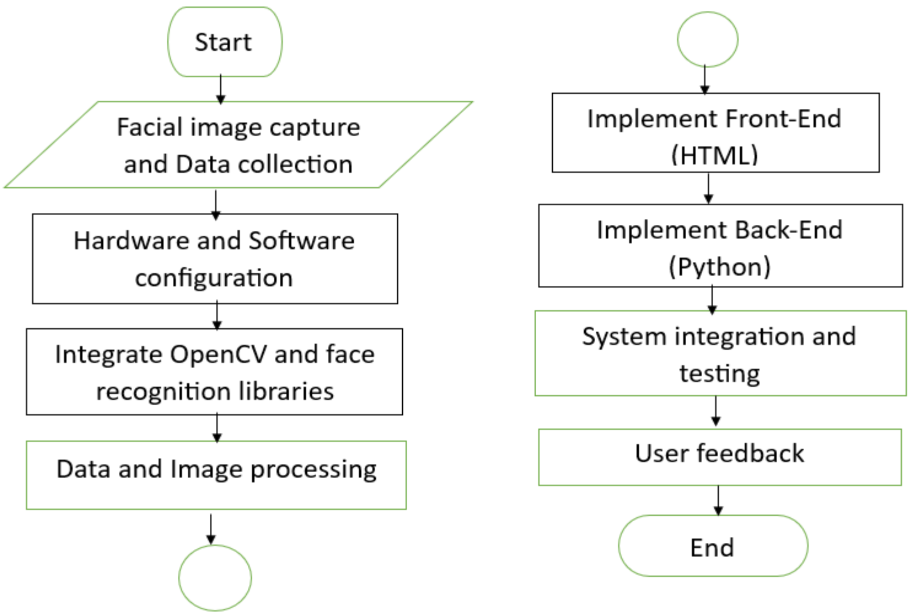

Figure 1 Design flowchart

The facial recognition attendance system was designed and implemented using an organized eight-step process:

Step 1: Facial image capture and data collection: To provide a trustworthy dataset for training, the names, matriculation numbers, and email addresses of 500-level Electrical and Information Engineering students were recorded together with their facial photos.

Step 2: Hardware and Software Configuration: The Raspberry Pi OS, Python, OpenCV, NumPy, PIL, Picamera, and CSV software tools, as well as the Raspberry Pi 4 camera module, LCD, cooling fan, and power supply, were all set up to support the system.

Step 3: Integration of Face Recognition and OpenCV Libraries: To manage tasks like face identification, picture preprocessing, and facial encoding, OpenCV and face recognition packages were combined.

Step 4: Data and image processing: Facial traits were extracted and encoded from captured photos through processing, guaranteeing precise identification and trustworthy attendance recording.

Step 5: Back-End Implementation: Python and Flask were used in the development of the system's back-end, which controls data management, file uploads, face recognition logic, and the coordination of all system functions.

Step 6: Development of the Front End: To ensure usability and convey attendance data in an understandable manner, a basic front-end interface was created using HTML.

Step 7: Testing and System Integration: Every module was combined to create a system that worked flawlessly. The Raspberry Pi was used as the primary controller during testing to assess accuracy, speed, and overall performance.

Step 8: User feedback: After system integration and testing, feedback was gathered from actual students to evaluate the system's effectiveness, usability and overall satisfaction.

2.1 Hardware configuration

The hardware configuration establishes the physical framework for the facial recognition attendance system, primarily utilizing the Raspberry Pi 4 Model B because of its processing capabilities and efficiency. This central unit coordinates key components such as the camera module for face capture, an I2C LCD for user feedback, a buzzer for audio alerts, and storage devices for data logging. Each component was selected based on its cost-effectiveness, compatibility, and reliability to ensure smooth real-time performance in academic environments. The components used are:

i. Raspberry pi Model 4b: It Serves as the main processing unit that runs the facial recognition code and controls peripheral devices. It has a rating of Quad-core Cortex-A72 (1.5GHz), 4GB RAM, 5V/3A USB-C power input, 2x micro-HDMI, USB 3.0

Figure 2: Raspberry pi 4b module

---

ii. Raspberry Pi rev 1.3 Camera Module 3: It has a specification of 2MP Sony IMX708 sensor Autofocus, 4608×2592 resolution and CSI interface. It is used to captures live video frames used for face detection and recognition.

Figure 3: Raspberry pi rev 1.3 camera

iii. 16x2 I2C LCD Display: It displays recognized student name and matric number or error messages. It has a 5V input and the I2C Address; 0x27 by default.

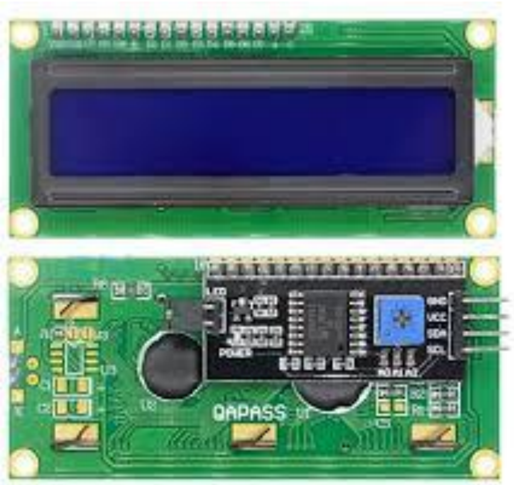

Figure 4: i2c LCD Display

2.2 Software Configuration

The software configuration of the facial recognition attendance system includes the operating environment, programming language, libraries, and development tools that enable face detection, recognition, and attendance logging. The software runs on the Raspberry Pi and was primarily developed using Python due to its simplicity, versatility, and compatibility with machine learning and computer vision libraries.

i. Development environment: The software development for the facial recognition attendance system was carried out using a Windows 11 laptop equipped with an Intel Core i7-10610U / i7-10510U processor and 12GB of RAM. The Raspberry Pi 4 Model B, the primary processor unit used to operate the facial recognition system, was the intended deployment device. The Raspberry Pi was running Raspberry Pi OS (64-bit), a Linux derivative based on Debian that was tailored for the device.

Python 3.9 was chosen as the main programming language because of its versatility and wide range of libraries for hardware control and computer vision. The Thonny IDE was utilized for code development directly on the Raspberry Pi, and Visual Studio Code was sometimes used for more complex editing through remote SSH access. The operating system, project files, training models, and log data were all stored on a 32GB microSD card.

Terminal-based technologies such as the Linux shell, the pip package manager for installing libraries, and crontab for automating activities on boot were also frequently used throughout the development process.

ii. Core Libraries and Their Functions: The software system for the facial recognition attendance project relied on a variety of Python libraries to handle image processing, facial recognition, hardware control, and data management. **OpenCV (**cv2**)** was used extensively for image acquisition from the camera, real-time face detection, and basic image preprocessing tasks such as resizing and color space conversion. The face recognition library, which serves as a high-level wrapper over dlib, was responsible for detecting faces, extracting facial landmarks, and generating 128-dimensional facial embedding used for recognition. The core facial recognition functionality was powered by dlib, which provides robust implementations of machine learning algorithms, particularly the Histogram of Oriented Gradients (HOG) for face detection and a linear Support Vector Machine (SVM) for classification. To enable interaction with the Raspberry Pi's hardware components, such as the buzzer, the RPi.GPIO library was used to control the general-purpose Input/Output (GPIO) pins.

2.3 Integration of the system and model training

The facial recognition attendance system's main functionality is based on the close integration of hardware, software, and machine learning components. Facial data collection is the first step in the process, followed by feature extraction, model training, and real-time inference for attendance recording. This section explains how the machine learning model was trained, along with pertinent equations, and how the system combines each module.

i. Facial data and data collection: Accurate facial recognition relies heavily on the quality and diversity of the training data. In this project, facial data collection and image acquisition were conducted under controlled but varied conditions to ensure robustness during real-time recognition. The dataset was compiled from 60 students, with each student contributing 15 facial images, resulting in a total of 900 images. Image capture was carried out under different lighting conditions, angles, and facial expressions. All images were resized to 640×480 pixels for uniformity and efficient processing on the Raspberry Pi. Face detection and alignment were performed using dlib's HOG-based detector.

ii. Feature extraction: Feature extraction is an important step in face recognition because it converts raw facial photos into organized numerical data that machine learning algorithms can examine and classify. In this project, feature extraction was carried out using the face recognition package, which is based on dlib's deep learning facial recognition model. Each identified face in the dataset was transformed to a 128-dimensional embedding vector. This vector depicts distinctive facial characteristics such as the distance between the eyes, the shape of the cheekbones, the width of the jaw, and other key facial features. The embedding are highly discriminative, serving as the foundation for comparing and distinguishing faces.

iii. Model training: After extracting the 128-dimensional facial embedding, a Support Vector Machine SVM classifier with a linear kernel was trained to recognize individual students. Each embedding was paired with a label representing the student's identity, forming the training dataset. The model was implemented using Scikit-learn function and trained on 80% of the data, with 20% reserved for validation. The SVM was chosen for its simplicity, speed, and effectiveness in high-dimensional classification tasks. During prediction, the system compares a new face encoding with the trained model to determine the closest match using:

$$d = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \quad (1)$$

3. Results And Discussion

Evaluating the performance of the facial recognition attendance system involved testing the system under realistic conditions using 500-level EIE students. Each student was tested over 10 separate sessions to ensure a reliable data sample across various scenarios. The performance of the developed system was evaluated in terms of accuracy, False Acceptance Rate (FAR), recognition speed, and confusion matrix under different conditions as shown in Table 1 based on the following equations:

Table 1: Performance evaluation metrics of the system

Metrics	Values obtained
Accuracy	85% - 95%
False Acceptance Rate (FAR)	5%
Recognition Speed (Latency)	5.4 to 6.3 seconds
Confusion matrix	True positive > 90% False positive <2%

3.1 System validation

To validate the performance of this designed system, the developed system was compared with existing work on facial recognition attendance systems namely; CNN [28], LBP +SVM [29], and Web-based JavaScript app [30]. Table 2 shows the summary of existing work against the developed system in terms of accuracy, speed, and hardware efficiency,

Table 2: Summary of existing work against the developed system

Metrics	[28]	[29]	[30]	This study
Model type	CNN	LBP +SVM	Web-based JavaScript app	HOG + SVM
Accuracy	90 – 92%	80 – 85%	80%	85 – 95%
Recognition speed	10 – 12 sec	7 – 9 secs	3 – 5 secs	5 to 6 secs
Hardware Efficiency	Raspberry pi 3	Desktop PC	Browser/ mobile device	Raspberry pi 4

The facial recognition system developed in this study was compared with three other existing systems: [28], [29], and [30], based on model type, accuracy, recognition speed, and hardware platform.

[28] utilized a Convolutional Neural Network (CNN) model deployed on a Raspberry Pi 3. Their system achieved an accuracy of 90–92%, but with relatively slow recognition speeds of 10–12 seconds, primarily due to the computational load of the CNN model on limited hardware.

[29] employed a combination of Local Binary Pattern (LBP) for feature extraction and an SVM classifier, running on a desktop PC. The system achieved an accuracy of 80–85%, with a recognition time of approximately 7–9 seconds. While suitable for controlled environments, its dependency on a desktop platform limits portability and scalability.

[30] designed a web-based JavaScript facial recognition app intended for use on mobile devices or browsers. Though portable and accessible, the system had a lower accuracy of around 80%, with faster recognition times between 3–5 seconds. However, its reliance on front-end browser processing limited the system’s consistency and security.

In comparison, the system developed in this project uses HOG (Histogram of Oriented Gradients) features with an SVM classifier, deployed on a Raspberry Pi 4. It achieved an accuracy range of 85–95% and a recognition speed of 5–6 seconds, offering a good balance between performance, speed, and hardware efficiency. This makes it particularly suitable for real-time classroom environments where affordability and reliability are essential.

4. Recommendations

To enhance the better performance, accuracy, and reliability of the facial recognition

attendance system, the following improvements are proposed:

i. Enhance Dataset Diversity: Collecting more facial images per individual under different lighting, angles, and expressions will improve recognition accuracy and help the model perform better in real-world conditions.

---

ii. Upgrade the Camera Module: Improving the image acquisition hardware can significantly increase detection precision. Replacing the current camera with a higher-resolution or wide-angle camera module will provide clearer and more comprehensive facial data, particularly in low-light or multi-subject scenarios.

iii. Upgrade the Processing Hardware: While the Raspberry Pi 4 performed adequately during testing, future implementations involving more advanced features (such as deep learning models or multi-face recognition) may benefit from greater processing power. It is therefore recommended to upgrade to Raspberry Pi 5, which offers improved CPU speed, GPU performance, and overall system responsiveness.

5. Conclusion

The facial recognition attendance system met its primary objective of automating the attendance process. It demonstrated practical applicability in small to medium-scale classroom environments, providing both visual and auditory feedback. The system effectively logged attendance data and offered a user-friendly interface for administrators via a web dashboard. Despite certain limitations, including false acceptance rates, recognition delays, and lighting dependency, the project served as a robust proof-of-concept. Its modular design allows for scalability, and its use of widely accessible hardware and open-source libraries makes it both cost-effective and adaptable.

CONFLICT OF INTEREST

The authors declare that they have no conflicts of interest.

ACKNOWLEDGMENTS

The authors would like to express sincere gratitude to Landmark University Centre for Research, Innovation and Discoveries for fostering a supportive atmosphere that allowed this research to be conducted effectively in term of publication Sponsorship. In particular, the authors appreciate the University's provision of key equipment, resources, and institutional information, all of which played a vital role in gathering data, performing analyses, and confirming the study's results.

References

[1] T.-V. Dang, *Smart attendance system based on improved facial recognition*, vol. 4, no. 1. 2023, pp. 46–53. Accessed: Aug. 27, 2025. [Online]. Available: https://journal.umy.ac.id/index.php/jrc/article/view/16808

[2] R. N. Khan, 'Attendance matters: Student performance and attitudes', *Int. J. Innov. Sci. Math. Educ.*, vol. 30, no. 4, 2022, Accessed: Aug. 27, 2025. [Online]. Available: https://openjournals.test.library.sydney.edu.au/CAL/article/view/15655

[3] H. Zeineddine, U. Braendle, and A. Farah, *Enhancing prediction of student success: Automated machine learning approach*, vol. 89. Elsevier, 2021, p. 106903. Accessed: Aug. 27, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0045790620307552

[4] L. Cagliero, L. Canale, L. Farinetti, E. Baralis, and E. Venuto, *Predicting student academic performance by means of associative classification*, vol. 11, no. 4. MDPI, 2021, p. 1420. Accessed: Aug. 27, 2025. [Online]. Available: https://www.mdpi.com/2076-3417/11/4/1420

[5] Y. Baashar *et al.*, *Toward predicting student's academic performance using artificial neural networks (ANNs)*, vol. 12, no. 3. MDPI, 2022, p. 1289. Accessed: Aug. 27, 2025. [Online]. Available: https://www.mdpi.com/2076-3417/12/3/1289

[6] K. Ishaq and S. Bibi, 'IoT Based Smart Attendance System Using Rfid: A Systematic Literature Review', Aug. 03, 2023, *arXiv*: arXiv:2308.02591. doi: 10.48550/arXiv.2308.02591.

[7] M. S. Mohammed and K. A. Zidan, *Enhancing attendance tracking using animated QR codes: a case study*, vol. 31, no. 3. 2023, p. 1716. Accessed: Aug. 27, 2025. [Online]. Available: https://www.academia.edu/download/105993938/17583.pdf

[8] N. S. Ali, A. H. Alhilali, H. D. Rjeib, H. Alsharqi, and B. A. Sadawi, 'Automated attendance management systems: systematic literature review', *Int. J. Technol. Enhanc. Learn.*, vol. 14, no. 1, p. 37, 2022, doi: 10.1504/IJTEL.2022.120559.

[9] G. Malik, 'Biometric Authentication-Risks and advancements in biometric security systems', *J. Comput. Sci. Technol. Stud.*, vol. 6, no. 3, pp. 159–180, 2024.

[10] M. Mubeen, 'Biometric authentication: Past, present, and future perspectives', *Int. J. Innov. Res. Technol. Sci.*, vol. 12, no. 2, pp. 351–362, 2024.

[11] S. Minaee, A. Abdolrashidi, H. Su, M. Bennamoun, and D. Zhang, 'Biometrics recognition using deep learning: a survey', *Artif. Intell. Rev.*, vol. 56, no. 8, pp. 8647–8695, Aug. 2023, doi: 10.1007/s10462-022-10237-x.

[12] V. K. Attri, T. Jaiswal, R. N. Jaiswal, and V. Baggan, ‘Exploring integrated biometric surveillance systems: Case studies and future trends’, *World J. Adv. Eng. Technol. Sci.*, vol. 14, no. 2, pp. 208–215, 2025.

[13] V. Upadhyaya, ‘Advancements in Computer Vision for Biometrics Enhancing Security and Identification’, in *Leveraging Computer Vision to Biometric Applications*, Chapman and Hall/CRC, 2024, pp. 260–292. Accessed: Aug. 27, 2025. [Online]. Available: https://www.taylorfrancis.com/chapters/edit/10.1201/9781032614663-14/advancements-computer-vision-biometrics-enhancing-security-identification-vivek-upadhyaya

[14] A. A. Alalwan *et al.*, *Fintech and contactless payment: help or hindrance? The role of invasion of privacy and information disclosure*, vol. 42, no. 1. Emerald Publishing Limited, 2024, pp. 66–93. Accessed: Aug. 27, 2025. [Online]. Available: https://www.emerald.com/insight/content/doi/10.1108/ijbm-08-2022-0339/full/html

[15] B. Amirgaliyev, M. Mussabek, T. Rakhimzhanova, and A. Zhumadillayeva, *A review of machine learning and deep learning methods for person detection, tracking and identification, and face recognition with applications*, vol. 25, no. 5. MDPI, 2025, p. 1410. Accessed: Aug. 27, 2025. [Online]. Available: https://www.mdpi.com/1424-8220/25/5/1410

[16] M. Siddiqui and P. Valsalan, *AI-Based Human Face Recognition System.*, vol. 20. 2024. Accessed: Aug. 27, 2025. [Online]. Available: https://search.ebscohost.com/login.aspx?direct=true&profile=ehost&scope=site&authtype=crawler&jrnl=11125209&AN=181187048&h=dzWo9fgAPg9d%2FZnl27%2BTG87tcYXj2fczAG8HewxHjApdJbOt%2B0%2FW%2Bc7cnGd6CPxCRcX72bR%2FiVPAaDF%2B7zGkjw%3D%3D&crl=c

[17] A. A. A. Aboluhom and I. Kandilli, *Real-time facial recognition via multitask learning on raspberry Pi*, vol. 15, no. 1. Nature Publishing Group UK London, 2025, p. 28467. Accessed: Aug. 27, 2025. [Online]. Available: https://www.nature.com/articles/s41598-025-97490-6

[18] D. K. Oladele, F. S. Cornelius, A. L. Sunday, A. E. Adetola, and E. Gracia, ‘The Crustal Structure of Landmark University and OMU-ARAN Environment, Nigeria, using High-Resolution Aeromagnetic Data’, presented at the 2023 International Conference on Science, Engineering and Business for Sustainable Development Goals (SEB-SDG), IEEE, 2023, pp. 1–9.

[19] A. K. Shukla, A. Shukla, and R. Singh, ‘Automatic attendance system based on CNN–LSTM and face recognition’, *Int. J. Inf. Technol.*, vol. 16, no. 3, pp. 1293–1301, Mar. 2024, doi: 10.1007/s41870-023-01495-1.

[20] O. Elemile *et al.*, ‘Prediction of Noise Pollution in Omu-Aran Township Using Artificial Neural Network’, presented at the 2024 International Conference on Science, Engineering and Business for Driving Sustainable Development Goals (SEB4SDG), IEEE, 2024, pp. 1–14.

[21] K. Alhanaee, M. Alhammadi, N. Almenhali, and M. Shatnawi, ‘Face recognition smart attendance system using deep transfer learning’, *Procedia Comput. Sci.*, vol. 192, pp. 4093–4102, 2021.

[22] H.-D. Thai, Y.-S. Seo, and J.-H. Huh, *Enhanced Efficiency in SMEs Attendance Monitoring: Low Cost Artificial Intelligence Facial Recognition Mobile Application*. IEEE, 2024. Accessed: Aug. 28, 2025. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10763516/

[23] L. M. Akanbi, M. Bashorun, U. A. Salihu, G. O. Babafemi, and K. Sulaiman, ‘Application of Barcode Technology in Landmark University Centre for Learning Resources, Omu-Aran Experience’, *Libr. Philos. Pract.*, p. 1, 2018.

[24] P. B. Jha, A. Basnet, B. Pokhrel, B. Pokhrel, G. K. Thakur, and S. Chhetri, ‘An automated attendance system using facial detection and recognition technology’, *Apex J. Bus. Manag.*, vol. 1, no. 1, pp. 103–120, 2023.

[25] D. Sunaryono, J. Siswantoro, and R. Anggoro, *An android based course attendance system using face recognition*, vol. 33, no. 3. Elsevier, 2021, pp. 304–312. Accessed: Aug. 28, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1319157818309406

[26] M. R. Wayahdi and F. A. Dzikri, ‘Integrating Facial Recognition and GPS Technology for Efficient Attendance Management in Educational Institutions’, Accessed: Aug. 28, 2025. [Online]. Available: https://www.researchgate.net/profile/Wayahdi/publication/388836854_Integrating_Facial_Recognition_and_GPS_Technology_for_Efficient_Attendance_Management_in_Educational_Institutions/links/67a9659d8311ce680c5ad3f0/Integrating-Facial-Recognition-and-GPS-Technology-for-Efficient-Attendance-Management-in-Educational-Institutions.pdf

[27] S. M. Bah and F. Ming, ‘An improved face recognition algorithm and its application in attendance management system’, *Array*, vol. 5, p. 100014, 2020.

[28] M. Hani’Ah, V. N. Wijayaningrum, V. A. H. Firdaus, and A. Himawan, ‘Leveraging FaceNet and SVM Hyperparameter Optimization in Face Recognition Attendance Systems with Limited Data’, in *2025 International Conference on Computer Sciences, Engineering, and Technology Innovation (ICoCSETI)*, IEEE, 2025, pp. 1–6. Accessed: Aug. 28, 2025. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/11019431/

[29] K. S. Rajpoot, U. Saxena, N. A. Shelke, and R. Agrawal, ‘Revolutionizing Attendance Tracking: A Holistic Automated Facial Recognition System’, in *2025 International Conference on Ambient Intelligence in Health Care (ICAIHC)*, IEEE, 2025, pp. 1–7. Accessed: Aug. 28, 2025. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10957105/

[30] A. N. Babatunde, A. A. Oke, R. S. Babatunde, O. Ibitoye, and E. R. Jimoh, ‘Mobile based student attendance system using geo-fencing with timing and face recognition’, *J Adv Math Comput Sci*, 2022, Accessed: Aug. 28, 2025. [Online]. Available: https://www.researchgate.net/profile/Akinbowale-Babatunde-2/publication/361553012_Mobile_Based_Student_Attendance_System_Using_Geo-Fencing_With_Timing_and_Face_Recognition/links/648b5fb395bbbe0c6ecb4441/Mobile-Based-Student-Attendance-System-Using-Geo-Fencing-With-Timing-and-Face-Recognition.pdf