**Smart Attendance Management System Using Face Recognition And**

**Fingerprint Authentication**

Mr. D. W. Chinchkhede (Guide)

*Satyam Dnyaneshwar Gole Aniket Bhausaheb Dalavi*

*Information Technology SCVP, Pune Information Technology SCVP, Pune*

[*Saytamgole4@gmail.com*](mailto:Saytamgole4@gmail.com)[*ad3436141@gmail.com*](mailto:ad3436141@gmail.com)

---------------------------------------------------------------------\*\*\*--------------------------------------------------------------------

**Abstract** This paper presents a Smart Attendance Management System Using Face Recognition that automates attendance marking using a camera and OpenCV. The system captures and matches faces against a local SQLite database, recording attendance in real time. Developed using Java, Python, and OpenCV, it features a user-friendly interface and communicates status updates to a Python-based dashboard. A QR code module is included as a backup for cases where face recognition fails. The system auto-starts on boot, reducing manual effort and improving accuracy. Future upgrades may include AI-based emotion detection and cloud integration for scalability.

**Keywords:** Face Recognition, Smart Attendance System, OpenCV, Real-Time Monitoring, Automation

**1.INTRODUCTION**

Effective attendance tracking is a critical requirement in educational institutions and corporate environments. Traditional methods of attendance, such as manual sign-ins or RFID cards, are often time-consuming, prone to human error, and susceptible to fraudulent marking. These limitations highlight the need for a more reliable, contactless, and automated system. To address these challenges, intelligent attendance systems leveraging face recognition technology have emerged as a modern solution to ensure accuracy, security, and operational efficiency.

This paper presents a Smart Attendance Management System Using Face Recognition, designed to automatically identify individuals and mark their attendance through facial recognition. The system combines computer vision, automation, and IoT technologies to deliver a real-time, userfriendly solution.

At the core of the system is an Android device or camera module, interfaced with OpenCV for live face detection and recognition. The captured facial data is compared against a pre-stored dataset saved in a SQLite database. The system identifies the individual and automatically records their attendance with a time stamp.

For accurate recognition and robust performance, the system uses the following components:

* Camera module to capture real-time face images.
* OpenCV framework for detecting and matching facial features.
* SQLite database to store registered user face encodings and related information.
* QR Code scanner as a fallback for cases when facial recognition fails (e.g., poor lighting, occluded face).

To handle system logic, data matching, and user interface functions, a backend is developed using Java and Python. The system includes a Python-based desktop UI that displays realtime recognition status, attendance logs, and error alerts. The face recognition engine runs concurrently, and all interactions are synchronized using local network protocols or MQTT, ensuring secure and responsive communication.

The system is configured to auto-start on device boot, enabling it to be used continuously without manual intervention. This feature is especially useful in classrooms or offices where daily attendance needs to be monitored consistently.

By automating attendance tracking with face recognition, this solution minimizes manual efforts, eliminates fraudulent attendance, and enhances record-keeping accuracy. Future upgrades may include integrating cloud storage for centralized data access, AI-based analytics for behavioral insights, and scalability features for use across larger campuses or enterprises.

**2. LITERATURE SURVEY**

**1. Face Recognition Based Attendance System using Machine Learning**

**Authors:** Shaik Nayab Rasool, V. Surendra

**Journal:** *International Journal of Advanced Research in*

*Electrical, Electronics and Instrumentation Engineering*

**Year:** 2022

**Summary:** This study presents a facial recognition-based attendance system that uses Haar Cascade and LBPH algorithms for face detection and recognition. The system is implemented using OpenCV and Python, with a local database for storing attendance logs. It emphasizes accuracy in realtime face detection and efficient time tracking in classroom environments. [ijareeie.com](https://www.ijareeie.com/)

**2. Real-Time Face Recognition for Attendance System Using Python and OpenCV**

**Authors:** A.V. Chandrawanshi, M. Balani, H. Jain

**Journal:** *International Journal of Computer Science and*

*Mobile Computing*

**Year:** 2021

**Summary:** This paper proposes a smart attendance system that captures student images using a webcam and matches them against a trained dataset using the OpenCV library in Python. The system automatically marks attendance upon successful recognition and updates a CSV log. It is a costeffective solution for small to medium-scale institutions. [ijcsmc.com](https://www.ijcsmc.com/)

**3. IoT-Based Smart Attendance System Using Facial**

**Recognition**

**Authors:** H. Saha, P. Bera, A. Mondal

**Journal:** *International Journal of Engineering Research &*

*Technology (IJERT)*

**Year:** 2020

**Summary:** The authors propose an IoT-enabled face recognition system for attendance that uses Raspberry Pi, Pi camera, and a cloud database. Face data is processed using OpenCV and stored in Firebase. The system also features realtime alerts and attendance monitoring from remote locations.

[ijert.org](https://www.ijert.org/)

**4. Automatic Attendance System Using Face Recognition and Deep Learning**

**Authors:** V. S. Patil, S. B. Kshirsagar

**Journal:** *IEEE Xplore (Conference Paper)*

**Year:** 2019

**Summary:** This study implements a deep learning-based attendance system using convolutional neural networks (CNNs) to improve face recognition accuracy. The system operates in real time and is optimized for classroom

environments with varying lighting and angles. It emphasizes high detection accuracy and low false recognition rates. [ieeexplore.ieee.org](https://ieeexplore.ieee.org/)

**4. Smart Attendance Monitoring System Using Face Recognition and RFID**

**Authors:** V. Shyam, M. Divya, S. Monisha

**Journal:** *International Journal of Scientific Research in*

*Engineering and Management (IJSREM)*

**Year:** 2023

**Summary:** This hybrid model integrates face recognition and RFID scanning to enhance reliability. Face data is processed using Haar cascades and stored in a local server. RFID is used as a fallback in the event of recognition failure. The dualsystem improves system flexibility and minimizes errors. [ijsrem.com](https://www.ijsrem.com/)

# 3. METHODOLOGY

The proposed Smart Attendance Management System Using Face Recognition employs an integrated approach combining computer vision, database management, wireless communication, and real-time monitoring. The methodology consists of several stages, including system architecture design, hardware and software integration, face detection and recognition flow, and user interface communication.

## 3.1 System Architecture

The system architecture includes the following key components:

* Camera Module – Captures live images of individuals for attendance verification.
* Raspberry Pi / Android Device – Acts as the central processing unit, running face detection and recognition algorithms using OpenCV.
* OpenCV Library – Used for real-time face detection and recognition.
* SQLite Database – Stores registered face encodings and corresponding student or employee information.
* QR Code Scanner (Backup) – Used when face recognition fails due to low lighting or occlusion.
* Python-based UI – Displays real-time attendance updates, recognition status, and logs on a connected laptop or desktop.
* MQTT Protocol / Local Server – Handles communication between the recognition module and the monitoring interface.

**3.2 Hardware Design and Integration**

The system's hardware is assembled in the following steps:

* Face Registration and Dataset Creation :
* Users are registered by capturing multiple face images using the camera.
* Face encodings are generated and stored in the SQLite database alongside user details.
* Live Face Detection and Recognition:
* A live video stream is processed using OpenCV.
* Detected faces are compared with registered encodings using algorithms such as LBPH (Local Binary Patterns Histograms) or EigenFaces.
* QR Code Module Integration:
* A fallback QR code scanner is included for backup attendance marking if facial recognition fails.
* The system prioritizes facial data but switches to QR scanning when needed.
* Data Logging and Real-Time Display:
* Recognized faces are logged with time stamps.
* The system sends data via MQTT protocol or API to the Python-based UI, which visually tracks attendance.

**3.3 Working Procedure**

The complete system follows a sequential workflow:

Step 1: Face Detection and Matching

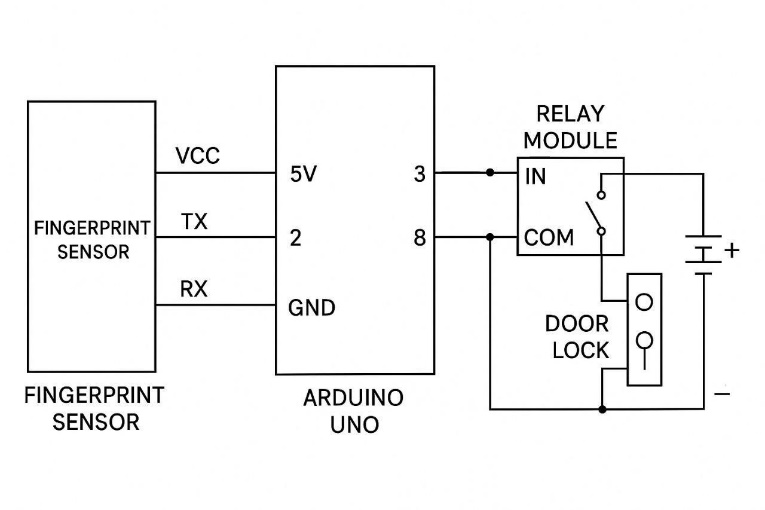
* The system captures a live frame using the camera.
* The OpenCV module detects faces and encodes the image.
* The encoded image is compared to the stored dataset.
* If a match is found, the attendance is recorded with the user's name, ID, and timestamp.

Step 2: Attendance Marking and Fallback

* If the face is unrecognized or detection fails:
* The user is prompted to scan their QR code.

**Step 3: Data Logging and Monitoring**

* Attendance data is transmitted to the Python UI using MQTT or a local network.
* The UI displays real-time recognition status, recorded entries, and alerts.
* Logs are saved in a structured format (CSV or SQL) for future analysis or report generation.

Fig 1. Block Diagram

**5. ALGORITHM & PROTOCOL**

The system’s decision-making logic is implemented using the following algorithms:

**1. Face Recognition Algorithm:**

* Continuously activates the camera to scan for a face.
* If a face is detected, extract facial features using OpenCV.
* Compare the detected face against the stored face encodings in the SQLite database.
* If a match is found:
* Mark attendance with the user’s ID and timestamp.  end attendance confirmation to the UI via MQTT.
* If no match is found, proceed to fingerprint verification.

**2. Fingerprint Authentication Algorithm:**

* Activate fingerprint sensor as a backup method.
* Prompt the user to place their finger on the scanner.
* Compare the fingerprint data with the stored templates.
* If a match is found:
* Mark attendance and log the entry.  Send the data to the UI for display.
* If no match is found:
* Display an alert indicating authentication failure.

**Advantages**

* Automated attendance marking improves accuracy and saves time.
* Dual authentication (face & fingerprint) enhances security and reliability.
* Reduces manual errors and eliminates proxy or fraudulent entries.
* Real-time monitoring and data logging increase administrative transparency.
* Compact and scalable design suitable for classrooms, offices, and enterprises.

## Limitations

1. **Face Recognition Accuracy:**

Performance may vary based on lighting conditions, facial angle, and quality of the captured image.

Users with facial features that change over time (e.g., due to aging, makeup, or facial hair changes) may experience inaccuracies.

1. **Fingerprint Authentication:**

Fingerprint scanners may struggle with dirty, wet, or damaged fingers.

Scanning accuracy can vary based on the quality of the

fingerprint sensor and skin conditions (e.g., calluses, scars)

1. **Environmental Factors:**

The system's performance can be affected by the environment (e.g., poor lighting, extreme temperatures) during face or fingerprint capture.

1. **Hardware Maintenance:**

Periodic cleaning and maintenance of fingerprint scanners and camera hardware are required to ensure accurate readings.

1. **Recalibration and Updates:**

The system may require recalibration after long-term use or when there are updates to the recognition algorithms.

1. **Privacy Concerns:**

Storing biometric data (facial images or fingerprints) requires proper security measures to avoid unauthorized access or misuse of sensitive personal information.

1. **User Enrollment and Management:**

Initial enrollment of users may take time and require multiple attempts for accurate capture of face and fingerprint data.

## 6. CONCLUSION AND FUTURE SCOPE

The proposed Smart Attendance Management System using Face Recognition and Fingerprint Authentication efficiently automates the attendance marking process, ensuring enhanced accuracy and reducing manual intervention. By integrating

cutting-edge technologies such as facial recognition and fingerprint authentication, the system accurately identifies and

records student attendance, providing a secure and reliable solution.

The system uses a combination of OpenCV for facial detection and matching, alongside a high-quality fingerprint scanner, to ensure accurate identification and verification. This dual authentication process improves the overall security and reliability of the attendance system. The data is stored securely in a database, enabling real-time tracking and easy retrieval of attendance records.

With the use of an Android application or web interface, users can monitor and manage attendance data efficiently. The application provides a user-friendly interface, and the system can be integrated with various school or college management software for seamless operation.

The system's automated nature reduces the risk of human error, increases operational efficiency, and ensures that attendance is accurately recorded in real-time. Moreover, this system eliminates the need for traditional manual attendance methods, which are time-consuming and prone to inaccuracies.

**Future Scope:**

1. Integration with Mobile Applications: The system could be enhanced with a mobile app for real-time notifications and reports, allowing teachers and administrators to track attendance on the go.
2. AI and Machine Learning: Future advancements could incorporate AI/ML algorithms for continuous improvement of face recognition accuracy, even in varied lighting conditions and with minor changes in user appearance.
3. Multi-factor Authentication: Adding additional layers of authentication, such as voice recognition or NFC card access, could further enhance security and flexibility.
4. Cloud Integration: Cloud storage could be implemented for centralized data management, making it easier for institutions to access and manage large amounts of attendance data.
5. Biometric Data Encryption: Future versions could focus on improving the security of biometric data by implementing advanced encryption techniques, ensuring better data privacy.

## 7. REFERENCES

1. Krishnapriya S., Aneesh Babu A.K., Aswin Kavil Biju, Abidha Abdul Karim Rawther, Bejoy Antony. "Intelligent IoT Enabled Smart Attendance System Using Face Recognition and Fingerprint Authentication." Research & Reviews: A Journal of Embedded System & Applications, 2024.
2. M.R. Chitale, S.J. Chitpur, A.B. Chivate, P.D. Chopade, S.M. Deshmukh, A.A. Marathe. "Automated Smart Attendance System Using Face Recognition and Fingerprint Authentication." Journal of Physics: Conference Series, 2023.
3. Rashmi Priyadarshni, Monisha D.P., Nikita Y.P., P. Mamatha Reddy, Pavithra B.M. "Design and Development of Automated Attendance System Using Face Recognition and Fingerprint Authentication." International Journal of Advanced Science and Technology, 2020.
4. F.E. Samann. "Design and Implementation of Smart Attendance System Using Face Recognition and Fingerprint Authentication." Academic Journal of Nawroz University, 2017.
5. R.S. Devi, V.R. Vijaykumar, M. Muthumeena. "Attendance Management Using Deep Learning and Biometric Authentication." International Journal of
6. Innovative Technology and Exploring Engineering, 2018.