

Smart Attendance System Using Facial Recognition

Suraj P Madiwal, Vittal S Goudra, Shreepad Gundupkar, Uday Kushalapur,
Professor Deepak Sharma

Department of E&CE, KLS Vishwanathrao Deshpande Institute of Technology,
Haliyal, Karnataka, India

Abstract- Accurate and efficient attendance tracking plays a key role in educational institutions. It helps maintain discipline, ensures transparency, and supports academic evaluation in various ways. Conventional methods for taking attendance tend to be slow and full of errors. They are also easy to manipulate with proxy marking, which makes them unfit for today's academic settings. This paper presents a fully automated Smart Face Recognition Attendance System. The system handles student registration, verifies at gates, monitors entry and exit in classrooms, and offers centralized real-time reporting. It combines face encoding algorithms with computer vision techniques. A MongoDB database backs it up for reliable storage and quick retrieval of attendance events. Multiple cameras allow for ongoing monitoring of student movements across campus locations. This setup relies on stable facial detection and records events with time stamps. Experiments show the system cuts down on manual work a lot. It stops proxy attendance effectively and delivers precise summaries for daily, weekly, and course-wise attendance. The overall solution scales well, stays cost-effective, and proves dependable. That makes it ideal for use in academic institutions.

Keywords: Medical Chatbot, Natural Language Processing, Artificial Intelligence, Healthcare, Patient- Doctor Interaction, Doctor Appointment System.

I. INTRODUCTION

Automated attendance systems are getting more essential in schools and colleges these days. They help fix the issues that plague older methods like paper records and RFID setups. Manual attendance recording consumes instructional time, is prone to duplication or manipulation, and fails to provide administrators with immediate access to attendance data. Likewise, RFID cards and biometric fingerprints require direct physical interaction and do not fully eliminate the possibility of identity misuse, such as students exchanging cards or bypassing verification. Face recognition technology provides a more secure, contactless, and user-friendly alternative by enabling identification without active participation from the user. With recent progress in machine learning, deep neural networks, and scalable data storage technologies, facial biometrics have become reliable enough for practical, large-scale applications.

This research proposes an integrated attendance management framework that includes student enrolment, automated gate-level authentication, classroom entry and exit detection, and centralized monitoring through a web-based dashboard. The

system incorporates Python, OpenCV, facial encoding models, and Fast API to deliver a real-time, precise, and fully automated attendance pipeline.

II. BACKGROUND

Attendance management really matters in the day to day admin work at schools and colleges. It ties right into keeping track of how students do academically, handling discipline issues, and maintaining all those official records. Old school ways of doing attendance, like calling out names or having kids sign sheets or scan their ID cards, have stuck around for years. Still, these methods tend to be pretty inefficient most of the time. They leave room for mistakes from people, take up a lot of time, let folks cheat the system sometimes, and do not give solid info on where students move around on campus.

To address these challenges, biometric systems particularly fingerprint scanners and RFID card-based mechanisms were introduced as improved alternatives. Although these systems enhance accuracy to some extent, they present their own limitations. Fingerprint devices require physical contact, which can lead to hygiene concerns and

slow processing during high-traffic periods like morning entry. RFID cards, on the other hand, may be faster but remain vulnerable to misuse because cards can be shared among students. Additionally, most biometric and RFID-based methods focus solely on classroom attendance and do not support multi-point verification across different campus entry and exit points.

III. RELATED WORK

Several studies have looked into automated attendance systems. Educational institutions are shifting away from old school manual and RFID tracking ways. Those earlier setups mostly depended on biometric fingerprints and smart cards. They did boost authentication a bit. Still, they had problems like needing physical touch, worries over sensor cleanliness, and folks sharing cards to fake attendance. All that pushed researchers toward tougher contactless options that would not run into the same snags.

With progress in computer vision and artificial intelligence, face recognition stands out now for quick identity checks. Plenty of research shows how deep learning based models work well. Things like CNNs, HOG based encoders, and Siamese networks handle face detection reliably. They manage different lights, angles, and even crowds without much trouble.

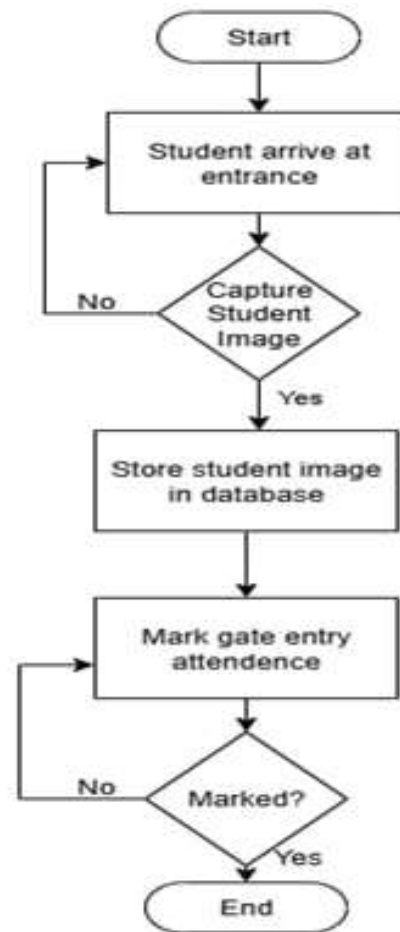
Some other work focuses on systems tied into CCTV. These can grab student faces on their own. Then they match against stored facial data and refresh the database all without anyone lifting a finger. Some studies also focused on multi-point attendance recording, where student tracking is not limited to classrooms but extends to entry gates and common areas. These systems highlight the importance of continuous monitoring, event logging, and cloud-based data synchronization to maintain accurate attendance records

IV. OUR PROPOSED METHODOLOGY

A. System Design

The proposed system is designed as a unified, multi-stage attendance management ecosystem that automates student identification across different campus locations. The architecture integrates four core modules: student enrolment, gate-level detection, classroom entry and exit monitoring, and centralized analytics. Each module communicates with a MongoDB database that stores facial encodings, timestamps, and attendance logs. IP or USB cameras capture real-time video streams, which are processed through Python-based face recognition and Fast API services running on a local server. A web dashboard provides administrators with immediate access to attendance statistics, daily summaries, and movement traces. This modular architecture ensures scalability, high performance, and smooth integration with existing campus infrastructure.

B. Face Recognition Process



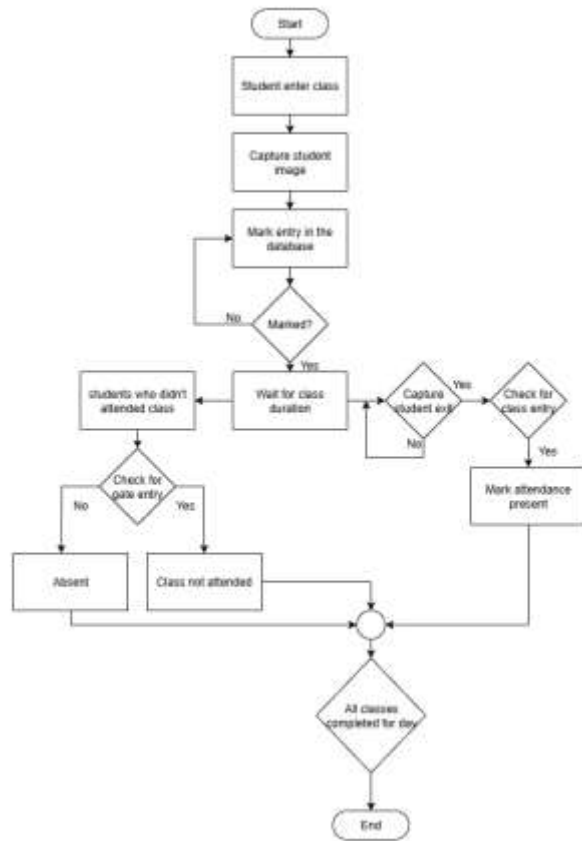


Figure: The Flowchart for the proposed system

The face recognition workflow begins with capturing live frames from the connected cameras. Each frame is preprocessed using OpenCV to enhance clarity and optimize lighting conditions. The system then extracts facial regions using a detection algorithm and converts these regions into numerical encodings using a feature extraction model. During identification, these encodings are compared with the stored student encodings using distance-based matching techniques. When a match is confirmed, the system logs the event with a timestamp and location tag—such as gate entry, classroom arrival, or exit. This continuous and automated pipeline ensures accurate, contactless verification without requiring students to pause or interact with the system.

V. ALGORITHM USED

A. Image Processing

Image processing in the proposed system focuses on preparing raw camera footage so that face detection

and recognition can operate efficiently and accurately. The captured frames are first converted to a consistent colour format, typically RGB, because most facial models are trained on such representations. Techniques such as resizing, noise reduction, histogram equalization, and contrast adjustment are applied to improve visibility and reduce the effects of poor lighting or motion blur. These preprocessing steps help ensure that the face detection module receives cleaner and more uniform images, ultimately enhancing recognition accuracy and reducing the chances of false detections.

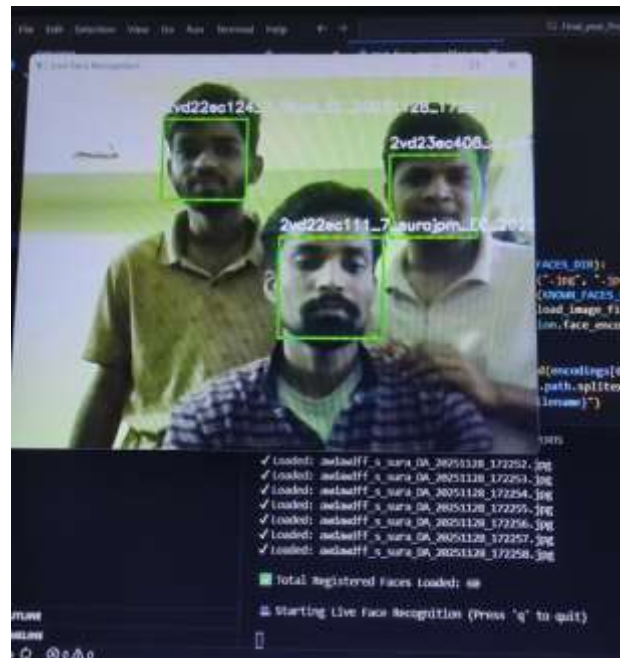


Figure: Detection of Face with IP web Camera

B. Face Detection

The face detection stage identifies and isolates the region of the image that contains a human face. The system uses a trained detection algorithm—commonly based on HOG descriptors, Haar cascades, or deep learning models—to locate facial boundaries within each frame. The detector looks over the image using different scales. This helps deal with changes in how far away things are, their angle, and the light conditions. When it spots a face, the system pulls out the coordinates for the bounding box. Those get sent along to the next part for more detailed work. Getting face detection right matters a

lot. Errors here mess up the whole process of pulling out features and doing recognition later on.

C. Feature Extraction

Step 1: Face Alignment

Before extracting features, the detected face is aligned to match a canonical orientation.

This uses:

- Eye detection
- Affine transformation

Mathematically:

$$x' = Ax + b$$

Where

A = rotation + scaling matrix

b = translation vector

x = original pixel coordinate

x' = aligned pixel coordinate

Alignment ensures:

- Eyes are horizontal
- Nose and mouth are centered.
- Pose variations are

Step 2: Normalization

The aligned face is converted into a normalized tensor:

$$I_{norm} = \frac{I - \mu}{\sigma}$$

Step 3: CNN-Based Feature Extraction

A Convolutional Neural Network (CNN) processes the normalized face.

A CNN performs:

- Convolution
- Re LU activation
- Max pooling
- Residual connections (in Res Net architectures)

The convolution formula:

$$F_{i,j} = \sum_m \sum_n I_{i+m,j+n} \cdot K_{m,n}$$

Where,

I = input image matrix

K = kernel/filter

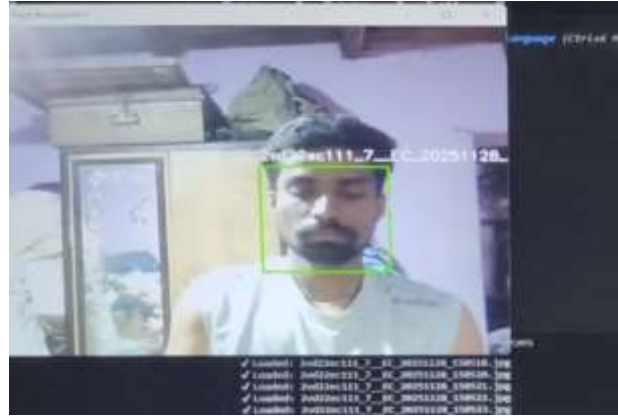


Figure: Face Detection

The CNN gradually learns:

- Eye shapes
- Nose contour
- Jaw structure
- Texture patterns

Step 4: Embedding Generation

The final output layer of the network projects the features into a fixed-length embedding vector:

$$E = f_{\theta}(I)$$

D. Face Recognition

The recognition module compares the extracted feature vector from a live camera frame with the stored face embeddings of registered students. A distance-based matching algorithm—such as Euclidean or cosine similarity—is used to determine how closely the two feature vectors resemble each other. If the computed distance is below a predefined threshold, the system confirms a match and identifies the student. The recognized identity is then logged alongside the timestamp and location of the event, such as gate entry or classroom arrival. This algorithm enables fast, reliable, and contactless verification, making it suitable for real-time attendance tracking.

VI. EXPERIMENTATION AND RESULTS

A. Login Page for Principal or HOD

A secure login page was developed for administrators, including the Principal and HOD. This page ensures that only authorized personnel can

access attendance records and student information. The login interface requires a username and password, which are validated against the system database. Upon successful authentication, the administrator is redirected to the main dashboard. This feature provides a controlled and secure entry point for managing attendance data.

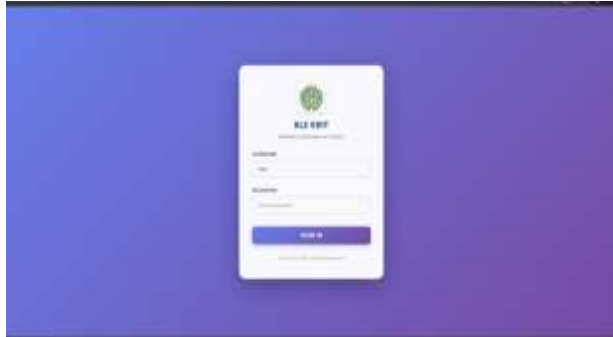


Figure: Principal / HOD Login page

B. Student Information After Login

Once logged in, the system displays detailed student information in a structured format. The data includes:

- Student Name
- USN/Roll Number
- Department and Semester
- Division
- Registered face image

This module allows administrators to verify student enrolment, cross-check face registration status, and access individual records efficiently. It ensures transparency and provides a centralized view of all registered students in the system.

A screenshot of a table displaying student information. The table has columns for 'ID', 'Name', 'Gender', 'Status', 'Division', and 'Image'. The data is organized into rows, with some cells containing red 'X' marks, likely indicating registration status or errors.

Figure: Student Information

C. Student Attendance System Dashboard

The dashboard is the core result of the project, offering real-time attendance tracking and visualization. Its features include:

- **Daily Attendance Summary:** Shows which students were present, absent, or partially

present (e.g., entered the classroom but did not exit properly).

- **Color-Coded Status:** Green indicates present, yellow indicates partial attendance, and red indicates absent.
- **Time-Stamped Events:** Each student's gate entry, classroom entry, and exit times are recorded and displayed.
- **Search and Filter:** Administrators can filter by department, class, or student to quickly access specific attendance data.
- **Auto-Refresh:** The dashboard updates every few seconds to reflect the latest attendance captured by the system.

The dashboard provides a complete, real-time view of student attendance and is a practical tool for both academic monitoring and administrative reporting. Its visual interface and automated data collection significantly reduce manual effort and errors compared to traditional methods.



Figure: Student Daily Attendance Dashboard

VII. CONCLUSION

The smart face recognition attendance integrating system shows off a fully automated way to track student attendance. It works without any contact and stays reliable for schools and colleges. The setup combines face recognition tech with checks at several points around campus. That means it catches when students enter gates or classrooms and when they leave. All this helps keep attendance records straight and stops anyone from marking for someone else.

Deep learning helps create face embeddings that the system uses to identify people. MongoDB stores everything in one central spot. So the whole thing

runs smoothly even with lots of students coming through. Updates happen in real time too. That keeps everyone in the loop without delays.

Admins like the principal or head of department get a web dashboard to check things out. It has an easy layout for pulling up daily attendance overviews. They can also look at details for single students or go back through past records. Reports come with colours to show status and timestamps for each event. This setup makes things clearer and cuts down on admin hassle.

In the end the project cuts out a ton of manual work. It frees up time in class that would otherwise go to roll calls. Plus it scales up well for bigger schools or universities.

REFERENCES

1. Shilpa M Pande, Subhassini S, Swastik Raj Singh."Smart Attendance and Attention Monitoring System." 14th International Conference on Computing, Communication and Networking Technologies (IEEE), July 6–8, 2023.
2. Chunduru, S Annapurna. "Smart Attendance System with Face Recognition using OpenCV." ICAISS 2023 (Second International Conference on Augmented Intelligence & Sustainable Systems).
3. Saurabh S. R., Komal Saxena. "Face Recognition Based Attendance System." 2022 – 1st IEEE ICIDEA (International Conference on Industrial Electronics: Developments & Applications).