ISSN 2278-2540 | DOI: 10.51583EMAS | Volume XV, Issue X, October 2026

Smart Attendance Automation Using Camera-Based Face Recognition

Roshan Malga*, T. Sai Kumar, V. Jashwanth, P. Sai Charan, S. Karthik

Department of Computer Science and Engineering, Hyderabad Institute of Technology and Management Hyderabad, Telangana, India

DOI: https://doi.org/10.51583/IJLTEMAS.2026.151000105

Received: 01 November 2026; Accepted: 06 November 2026; Published: 14 November 2026

ABSTRACT

Attendance management in educational institutions is often performed through manual registers or roll calls, which require staff time and can lead to recording errors, proxy attendance, and difficulty retrieving historical records. This paper presents a web-based Smart Attendance Automation prototype that integrates camera-based face detection and recognition with digital attendance management. The application is developed using Python and Flask, with HTML, CSS, Bootstrap, and JavaScript for the user interface and SQLite for student and attendance records. During enrollment, an administrator enters student details and captures ten facial images per student. The images are organized by roll number and used to train an OpenCV Local Binary Patterns Histograms (LBPH) recognizer. During attendance, browser camera frames are sent to the Flask backend, Haar Cascade detects face regions, and LBPH predicts the registered identity. A configured threshold is applied before a student is accepted. The application checks for an existing record before inserting attendance for the same student and date, and provides student management, daily and monthly reports, individual attendance history, charts, and CSV export. Preliminary testing involved six registered students under good and low lighting, and six unregistered people. The system correctly identified 6/6 registered-student attempts under good lighting and 4/6 under low lighting; the remaining two low-light attempts were rejected. All six unregistered-person tests were rejected as unregistered, with no incorrect acceptance observed. The resulting genuine-student identification rate was 83.33%, the false rejection rate was 16.67%, and the observed false acceptance rate was 0% in this small test sample. These results are preliminary and require validation using a larger, more diverse dataset.

Keywords—Smart attendance, face recognition, OpenCV, Haar Cascade, LBPH, Flask, Python, SQLite, biometric privacy.

INTRODUCTION

Attendance management is an important activity in educational institutions, as it helps monitor student participation and maintain academic records. Traditional attendance methods, such as calling student names and signing paper registers, require considerable time and administrative effort. These methods may also result in recording errors, proxy attendance, and difficulties in maintaining and retrieving historical attendance information.

To overcome these limitations, automated attendance management systems using facial recognition have gained attention. Facial recognition provides a contactless approach to identifying individuals by analyzing their facial features. By integrating facial recognition with database management and web technologies, attendance can be recorded automatically, reducing the need for manual intervention.

The proposed project, Smart Attendance Automation Using Camera-Based Face Recognition, is a web-based application developed to simplify student attendance management. The system uses a browser-based camera to capture facial images and communicates with a Python Flask backend for processing. OpenCV Haar Cascade is used for face detection, while the Local Binary Patterns Histograms (LBPH) algorithm is used for facial recognition. SQLite is used to store student details and attendance records.

ISSN 2278-2540 | DOI: 10.51583EMAS | Volume XV, Issue X, October 2026

The application provides several functionalities, including student registration, multiple-image face registration, model training, real-time camera-based recognition, automatic attendance recording, duplicate attendance prevention, and attendance report generation. The administrator can manage student information, monitor daily attendance, view monthly reports, and export attendance data.

The system is designed as an integrated attendance-management prototype using lightweight computer vision techniques. It focuses on combining facial recognition with database management and reporting functionalities rather than developing a new facial recognition algorithm.

The main contributions of this work are:

- Development of a web-based attendance management application using Python Flask.
- Integration of Haar Cascade face detection and LBPH-based facial recognition.
- Collection and storage of multiple facial images for each registered student.
- Automatic attendance recording with date-based duplicate prevention.
- Development of administrator facilities for student management and attendance report generation.
- Preliminary evaluation of the system under good lighting, low lighting, and unregistered-person testing conditions.

Project Statement

Attendance management in educational institutions is commonly performed using manual registers or conventional attendance-recording methods. These approaches require time, involve administrative effort, and may result in inaccurate records or proxy attendance.

Although facial recognition technology provides an automated method for identifying individuals, developing a complete attendance-management solution requires additional functionalities such as student registration, facial-image collection, model training, database management, duplicate attendance prevention, and report generation.

The problem addressed in this research is the development of an integrated camera-based attendance-management system that can identify registered students, reject unregistered individuals, and automatically maintain attendance records.

The proposed system uses OpenCV Haar Cascade for face detection and LBPH for face recognition. A Python Flask backend processes camera frames received from the browser, while SQLite maintains student and attendance information.

The system aims to reduce manual attendance work, improve record management, and provide an organized platform for monitoring student attendance.

LITERATURE SURVEY

Facial recognition has been widely studied in computer vision for identifying individuals from digital images. Different techniques have been developed for face detection and recognition, ranging from traditional image-processing methods to advanced deep-learning architectures. The following studies provide the technical foundation for the proposed system.

Haar Cascade-Based Face Detection

Viola and Jones [1] introduced a real-time object-detection framework using Haar-like features, integral images, AdaBoost, and cascade classifiers. The method detects objects by examining image features and rejecting non-object regions through multiple classifier stages.

ISSN 2278-2540 | DOI: 10.51583EMAS | Volume XV, Issue X, October 2026

Haar Cascade is widely used for face detection because it is computationally efficient and suitable for real-time applications. In the proposed system, Haar Cascade is used to identify facial regions from camera frames before passing the detected face to the recognition module.

However, face detection performance can be affected by poor lighting conditions, facial angles, and partial occlusion.

Local Binary Patterns for Face Recognition

Ahonen, Hadid, and Pietikäinen [2] studied Local Binary Patterns (LBP) for face description and recognition. LBP represents local facial texture by comparing the intensity of a pixel with its neighbouring pixels and encoding the resulting pattern.

The facial image is divided into regions, and histograms of local binary patterns are used to describe facial features. The LBPH method uses these descriptions for facial recognition.

The proposed system uses the LBPH recognizer available in OpenCV because it is relatively simple to implement, requires fewer computational resources, and can operate on CPU-based systems.

Deep-Learning-Based Face Recognition

Deep-learning-based facial recognition methods use Convolutional Neural Networks (CNNs) and learned facial representations to identify individuals. These approaches can provide improved recognition performance under variations in lighting, facial expressions, and viewing angles when appropriately trained.

A 2023 systematic literature review compared LBPH and CNN-based approaches for student attendance systems [3]. The review discussed the advantages of deep-learning methods in handling variations in facial appearance while identifying LBPH as a comparatively lightweight approach.

Although deep-learning techniques can provide improved recognition capabilities, they may require greater computational resources and training data. Therefore, the selection of a recognition method depends on the application's computational requirements and implementation environment.

Comparison of LBPH and CNN-Based Face Recognition

Table I. Comparison of LBPH and CNN-Based Face Recognition Methods

Parameter	LBPH-Based Recognition	CNN-Based Recognition
Technology	Traditional computer vision	Deep learning
Feature extraction	Local Binary Patterns	Automatically learned deep features
Computational requirements	Relatively low	Generally high
Training requirements	Simple and requires fewer resources	Requires more data and computational resources
Recognition performance	Suitable for controlled environments	Can provide better performance under varying conditions
Lighting variations	Sensitive to illumination changes	Can be more robust when appropriately trained

ISSN 2278-2540 | DOI: 10.51583EMAS | Volume XV, Issue X, October 2026

Facial angle variations	May experience reduced recognition performance	Can handle variations better with suitable training
Implementation complexity	Simple to implement using OpenCV	Requires deep-learning frameworks and models
Suitability	Lightweight CPU-based applications	Advanced facial recognition applications

The comparison indicates that LBPH provides a lightweight facial recognition approach with relatively low computational requirements, making it suitable for CPU-based applications. In contrast, CNN-based methods use deep feature extraction techniques and can provide improved recognition performance under variations in lighting, facial expressions, and viewing angles when appropriately trained. However, these methods generally require greater computational resources and training data. Therefore, LBPH was selected for the proposed system based on its simplicity, lower computational requirements, and ease of integration with OpenCV. A direct experimental comparison between LBPH and CNN-based methods has not been performed in the present study and is considered future work.

Existing Facial Recognition Attendance Systems

Darmawan (2023) implemented a face-recognition attendance system for online learning using Haar Cascade and LBPH and reported an accuracy of 93.55% under that study's experimental conditions [4].

Beri, Srivastava, and Malik (2024) also described an attendance-management system using Haar Cascade and LBPH [5]. These studies demonstrate the application of traditional face detection and recognition methods in automated attendance management.

Recent research has also explored more advanced and hybrid methods. An IEEE conference paper published in 2025 described a face-recognition attendance system using LBPH with a deep-learning architecture [6]. A 2024 design-science study examined computer-vision-based attendance management in a human-resources setting [7].

The reported results of these studies cannot be directly compared with the present prototype because their datasets, testing procedures, environmental conditions, and evaluation methods differ.

Summary of Literature Survey

The literature indicates a trade-off between computational simplicity and recognition robustness. Haar Cascade and LBPH provide a lightweight approach suitable for CPU-based applications, whereas CNN-based methods offer advanced facial representations but may require additional computational resources. The proposed system adopts Haar Cascade and LBPH because of their simplicity and suitability for a lightweight web-based attendance application. The literature also highlights the importance of evaluating recognition performance under different environmental conditions.

Research Gap

Existing research has demonstrated the use of Haar Cascade, LBPH, and CNN-based algorithms for facial recognition and automated attendance management. However, developing a practical attendance-management application requires more than implementing face detection and recognition.

The following research gaps are identified:

Limited System Integration

Several implementations focus mainly on facial detection and recognition, while a complete attendance

ISSN 2278-2540 | DOI: 10.51583EMAS | Volume XV, Issue X, October 2026

application also requires student registration, facial-image collection, database management, duplicate attendance prevention, and reporting facilities.

Limited Environmental Evaluation

Recognition performance may change under different lighting conditions and facial orientations. Testing only under good lighting may not provide sufficient information about the system's behaviour in practical environments.

Unregistered-Person Handling

Attendance applications must distinguish registered students from individuals whose facial images are not available in the trained dataset. Incorrectly accepting an unregistered individual may result in inaccurate attendance records.

Duplicate Attendance Prevention

A practical attendance system must prevent multiple attendance entries for the same student on the same date to maintain accurate attendance records.

Dataset and Testing Transparency

Experimental evaluation should clearly specify the number of participants, facial images collected, training procedures, testing conditions, and recognition outcomes.

Biometric Data Protection

Facial images contain sensitive biometric information. Therefore, attendance systems must consider controlled access, secure storage, authorized usage, and protection against unauthorized access.

Research Contribution

To address these practical requirements, the proposed research develops a web-based attendance-management prototype integrating student registration, multiple-image face collection, LBPH model training, camera-based recognition, duplicate attendance prevention, and attendance report generation.

The system is evaluated using six registered students under good and low lighting conditions and six unregistered individuals to examine its preliminary recognition and rejection behaviour.

The contribution of this work is an implementation-level integration of these functionalities into a lightweight attendance-management application. It does not propose a new facial recognition algorithm or claim superiority over deep-learning-based methods.

PROPOSED METHODOLOGY

The proposed methodology consists of student registration, facial-image collection, dataset preparation, model training, face detection, face recognition, attendance validation, and report generation.

Student Registration

The administrator registers students by entering the required information, including:

- Student name
- Roll number
- Branch
- Academic year

ISSN 2278-2540 | DOI: 10.51583EMAS | Volume XV, Issue X, October 2026

- Email address

The information is stored in the SQLite database and used to associate the student's identity with the facial recognition model.

Face Sample Collection

During face registration, the browser camera captures ten facial images for each registered student.

The captured images are stored in separate folders named using the corresponding student roll number. Each image is saved with a unique filename.

The dataset is organized as follows:

```
dataset/
│
├── 101/
│   ├── face_1.jpg
│   ├── face_2.jpg
│   ├── face_3.jpg
│   └── ... face_10.jpg
│
├── 102/
│   ├── face_1.jpg
│   ├── face_2.jpg
│   ├── face_3.jpg
│   └── ... face_10.jpg
│
└── Other student folders
```

This directory structure helps organize facial samples according to student identity.

Dataset Preparation

The collected facial images are organized according to student identity. OpenCV is used to process the images before they are provided to the LBPH recognizer.

The images are converted into grayscale representations and associated with numerical labels corresponding to registered students.

For the evaluation cohort, six registered students were considered, with ten facial images collected per student, resulting in 60 enrollment images.

ISSN 2278-2540 | DOI: 10.51583EMAS | Volume XV, Issue X, October 2026

These images are used for model training and are not treated as an independent testing dataset.

Model Training

The collected facial images are used to train the OpenCV LBPH face recognizer.

Each student's facial images are associated with a numerical label. During training, LBPH extracts local texture patterns from the images and uses them to build the recognition model.

After training, the model is saved in a file and loaded during the attendance-recognition process.

Face Detection Using Haar Cascade

The system uses the OpenCV Haar Cascade classifier to detect faces in captured camera frames.

The process involves:

- Capturing an image frame from the browser camera.
- Converting the image into grayscale.
- Applying the Haar Cascade classifier.
- Identifying the detected facial region.
- Extracting the face for recognition.

The extracted face region is passed to the LBPH recognition module.

Face Recognition Using LBPH

After face detection, the extracted facial region is passed to the trained LBPH recognizer.

The recognizer compares the facial features with the trained model and predicts the corresponding student label.

It also generates a confidence/distance value. The application compares this value with the configured recognition threshold to determine whether the detected face should be accepted or rejected.

A lower LBPH distance generally indicates a closer match. If the prediction does not satisfy the configured threshold, the face is treated as unknown or rejected.

LBPH is selected because of its relatively low computational requirements and ease of integration with OpenCV. However, its performance under changing environmental conditions must be evaluated experimentally.

Attendance Validation and Storage

After successful face recognition, the system retrieves the corresponding student details from the SQLite database.

Before inserting an attendance record, the application checks whether attendance has already been recorded for the student on the current date.

- If an attendance record already exists, another entry is not created.
- If no attendance record exists, a new Present entry is inserted into SQLite.

ISSN 2278-2540 | DOI: 10.51583EMAS | Volume XV, Issue X, October 2026

This mechanism prevents duplicate attendance records for the same student on the same day.

Report Generation

The application provides an administrator dashboard for monitoring attendance information.

The reporting module includes:

- Daily attendance reports.
- Monthly attendance reports.
- Individual student attendance history.
- Present and absent student counts.
- Attendance percentage calculations.
- Graphical attendance summaries.
- CSV export functionality.

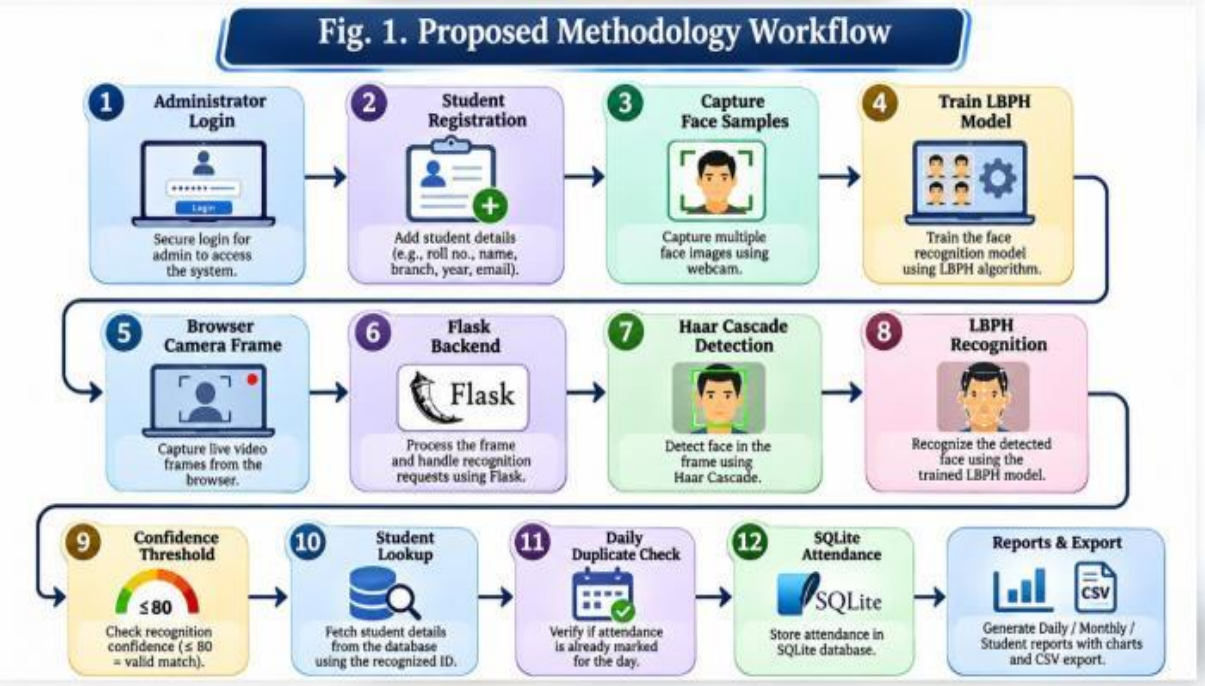


Fig. 1. Proposed methodology workflow

System Architecture

The system follows a client-server architecture consisting of a browser-based frontend, a Flask backend, computer-vision components, and an SQLite database. The frontend is developed using HTML, CSS, Bootstrap 5.3.3, JavaScript, and Jinja2 templates. It provides the administrator interface for login, student registration, face capture, attendance, and reports. The Flask backend handles application routes, student operations, incoming camera frames, face-processing requests, attendance insertion, and report generation.

OpenCV Haar Cascade is used for face detection, while the LBPH recognizer identifies registered students. SQLite stores student details and attendance entries. NumPy supports image processing, Chart.js is used for charts, and CSV export supports downloading report data.

ISSN 2278-2540 | DOI: 10.51583EMAS | Volume XV, Issue X, October 2026

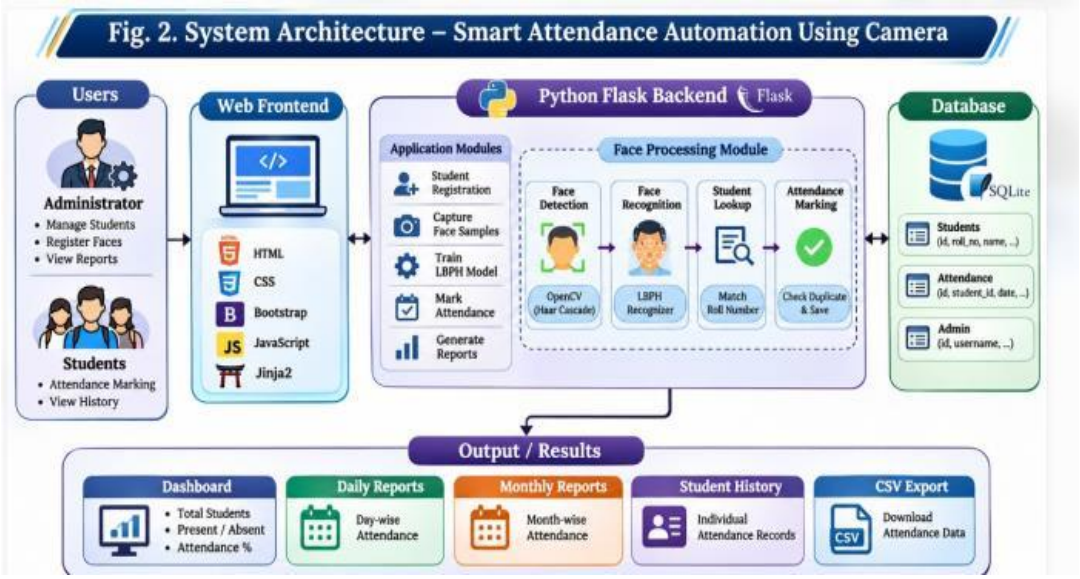


Fig. 2. System architecture of Smart Attendance Automation using Camera.

Implementation

Student Registration and Dataset Preparation

Student registration stores student details in SQLite. During face registration, the browser camera captures ten images for each student. The application stores the images in a folder named with the student's roll number, using sequential image filenames. These enrollment images form the LBPH training dataset. The application dashboard screenshot supplied for this revision displayed six registered student records. If all six records have ten successfully saved images, the corresponding enrollment set contains up to 60 images. This count describes enrollment data, not independent test images.

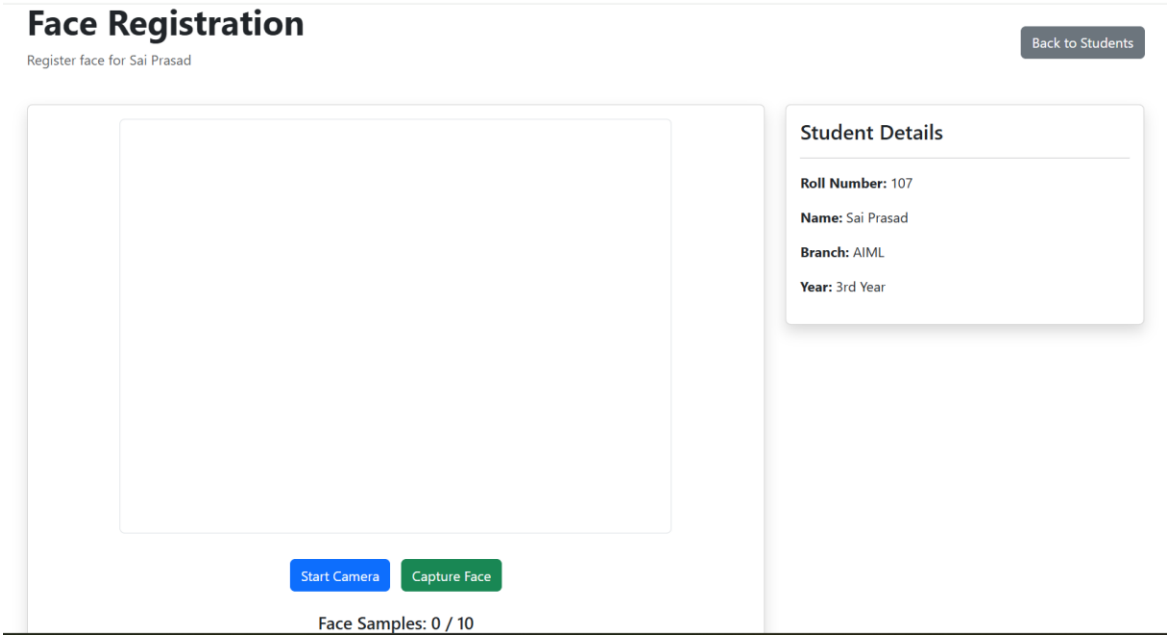


Fig. 3. Student face-registration interface used to capture facial samples.

Face Detection Using Haar Cascade

At attendance time, the backend receives a camera frame and prepares it for OpenCV processing. The Haar Cascade classifier locates face regions in the image. The detected face region is then passed to the recognition stage. The cascade-based detection approach is associated with the work of Viola and Jones [1].

ISSN 2278-2540 | DOI: 10.51583EMAS | Volume XV, Issue X, October 2026

Face Recognition Using LBPH

The detected face region is passed to the trained LBPH recognizer. LBPH describes local facial texture through local binary patterns and histograms. The recognizer returns a predicted label and a confidence/distance value. The application applies its configured threshold to accept or reject the prediction. Because LBPH distance is implementation- and dataset-dependent, the threshold must be selected using validation data and documented with the test protocol; a threshold value alone does not establish accuracy.

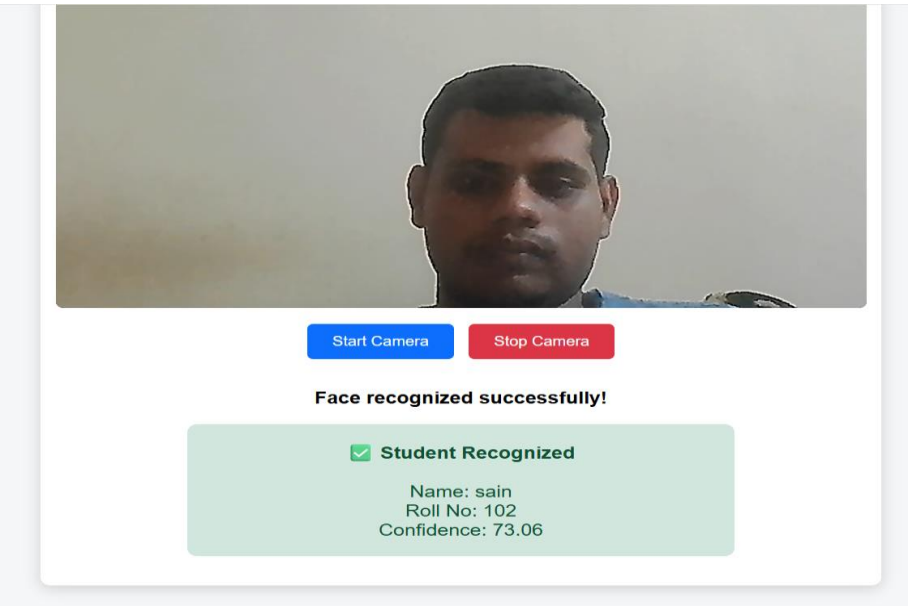


Fig. 4. Example live-camera screen showing a registered student recognized by the prototype.

Attendance Recording and Duplicate Prevention

After a prediction is accepted, the application uses the recognized roll number to retrieve the corresponding student record. It checks whether an attendance entry already exists for that student on the current date. If no entry exists, a Present record is inserted into SQLite. If a record already exists, the application avoids inserting another entry for the same date.

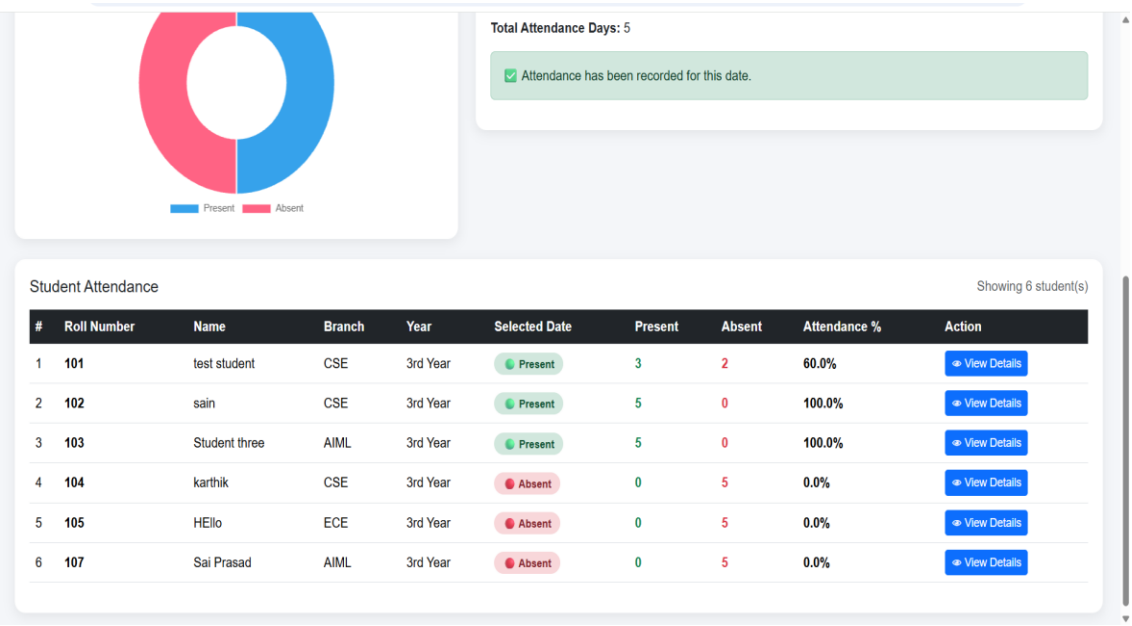


Fig. 5. Attendance reports interface showing date filters, attendance summary, and student-wise records.

ISSN 2278-2540 | DOI: 10.51583EMAS | Volume XV, Issue X, October 2026

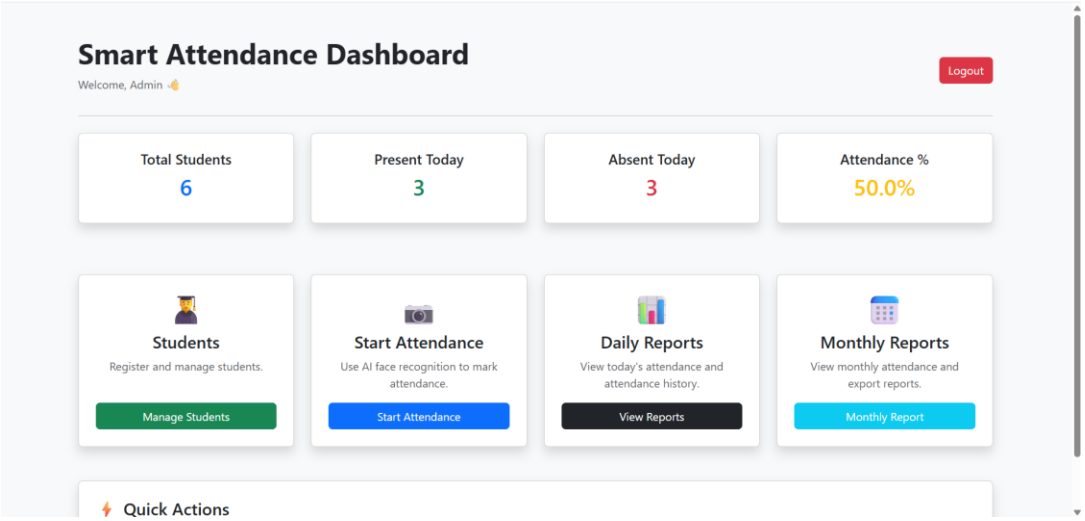


Fig. 6. Administrator dashboard showing student and daily attendance summary cards.

Technology Stack

- Frontend: HTML, CSS, Bootstrap 5.3.3, JavaScript, and Jinja2 templates.
- Backend: Python and Flask.
- Computer Vision: OpenCV Haar Cascade and LBPH face recognizer, with NumPy for image processing.
- Database: SQLite and SQL.
- Reporting and Development: Chart.js, CSV export, and Visual Studio Code.

Equations

Let d denote the LBPH distance returned for a prediction and T denote the configured acceptance threshold. The identity decision is:

Decision = Accept predicted identity when $d \leq T$; otherwise, reject the prediction as unknown. (The exact comparison direction and threshold must match the implementation used in the submitted source code.)

Attendance insertion is performed only when the student has no existing attendance record for the current date:

Insert Present = 1, if Student ID + Current Date has no existing record; otherwise, do not insert.

For a controlled evaluation, report at least: (i) correct identification rate = correct genuine identifications / all genuine test attempts × 100; (ii) wrong-identification rate = genuine attempts assigned to an incorrect enrolled identity / all genuine test attempts × 100; (iii) false-rejection rate (FRR) = genuine attempts rejected as unknown / all genuine test attempts × 100; and (iv) false-acceptance rate (FAR) = unknown-person attempts accepted as any enrolled identity / all unknown-person attempts × 100. Report the numerator and denominator for every metric. Overall accuracy should be calculated only from a clearly defined labelled test set and should not combine incompatible test categories.

Experimental Setup

The prototype was developed using Python, Flask, OpenCV, SQLite, and browser-based camera capture. The enrollment procedure collects ten facial images per registered student and stores them by roll number for LBPH training. The preliminary live-camera evaluation covered six enrolled students under two illumination conditions: good lighting and low lighting. Six observations were recorded under each condition. In addition, six unregistered people were tested to check unknown-person rejection. The reported outcomes were: 6/6 correct identifications under good lighting; 4/6 correct identifications and 2/6 rejections

ISSN 2278-2540 | DOI: 10.51583EMAS | Volume XV, Issue X, October 2026

under low lighting; and 6/6 unregistered-person rejections. No wrong identity assignment or unknown-person acceptance was reported. The available test log does not specify repeated trials per person, camera distance, lux level, exact pose angle, or processing time; these variables are therefore not claimed as controlled measureents.

RESULTS AND DISCUSSION

The preliminary results show that recognition performance varied with illumination. All six registered-student tests under good lighting were correctly identified, while four of six were correctly identified under low lighting and two were rejected. No wrong identity assignment was reported. All six unregistered-person tests were rejected as unregistered. Across the 12 genuine-student observations, the correct identification rate was 83.33%, the wrong-identification rate was 0%, and the false-rejection rate was 16.67%. For the six unknown-person observations, the observed false-acceptance rate was 0%. These figures describe only the small test sample supplied for this revision.

Table II. Functional observations of the implemented prototype.

Module / Test	Expected Function	Project Report Observation
Administrator login	Allow administrator access	Login and dashboard workflow reviewed
Student registration	Save and display student details	Student management interface implemented
Face registration	Capture multiple face samples	Samples stored by student roll number
Model training	Train and save LBPH model	Trained model used for recognition
Face recognition	Detect and predict registered identity	Haar Cascade and LBPH workflow implemented
Duplicate check	Avoid a second entry on the same date	Existing daily entry is checked before insertion
Reports	View attendance summaries and history	Daily, monthly, student history, charts and CSV export available

The functional observations show that the prototype connects enrollment, recognition, attendance storage, duplicate checking, and reporting. The preliminary recognition tests indicate that illumination affects the result: the system recognized all tested enrolled students under good lighting but rejected two enrolled-student attempts under low lighting. The system rejected all six tested unregistered people. Because the sample contains only six enrolled and six unregistered people, these results should be treated as an initial demonstration rather than a definitive measure of reliability. Further testing should use more participants, repeated attempts, different cameras, distances, poses, and lighting levels.

ISSN 2278-2540 | DOI: 10.51583EMAS | Volume XV, Issue X, October 2026

Table III. Preliminary live-camera recognition test results reported for the prototype.

Test condition	Attempts	Correct ID	Wrong ID / acceptance	Rejected	Observed result
Registered students—good lighting	6	6	0	0	100% correct identification
Registered students—low lighting	6	4	0	2	66.67% correct; 33.33% rejected
Unregistered people	6	N/A	0	6	All six correctly rejected as unregistered

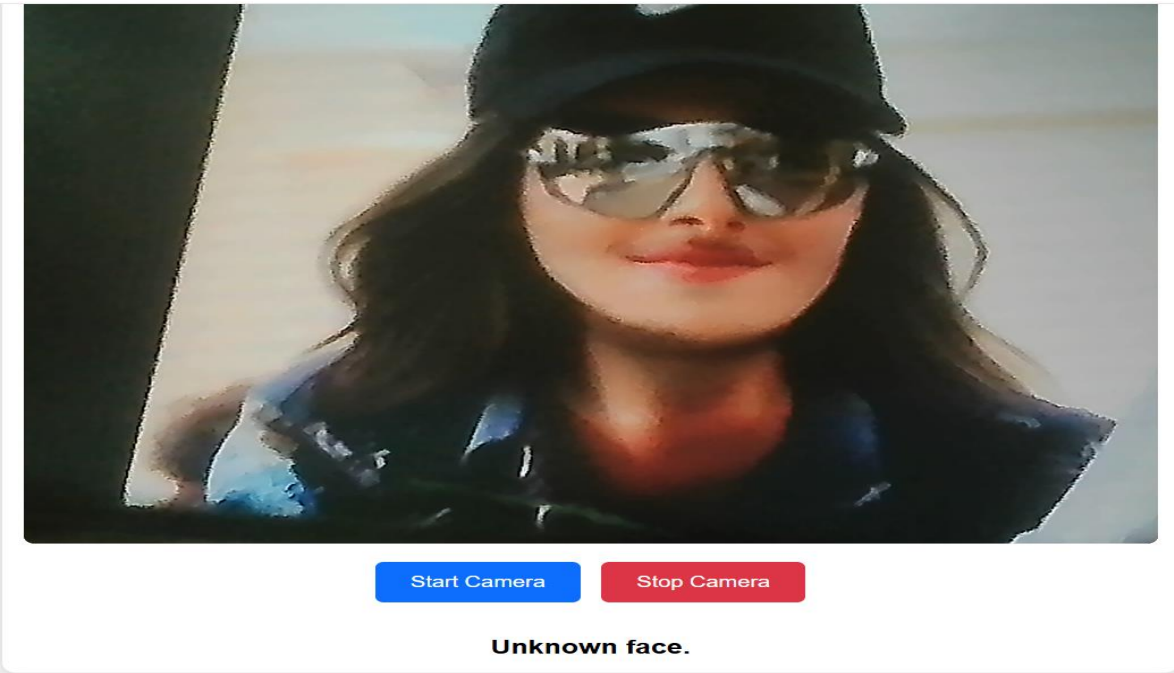


Fig. 7 Detection and rejection of an unregistered face by the proposed system.

An additional test was conducted using a face image that was not included in the registered student dataset. The image was presented to the camera, and the system detected the face but could not match it with any registered student identity. Consequently, the application displayed the message "Unknown face." This observation demonstrates the system's ability to reject an unregistered face image and avoid unauthorized attendance marking.

Table IV. Performance metrics calculated from the reported preliminary test observations.

Note: The values are based on 12 genuine-student observations and six unknown-person observations only. They are preliminary sample results, not a statistically conclusive estimate of performance across all students or conditions.

ISSN 2278-2540 | DOI: 10.51583EMAS | Volume XV, Issue X, October 2026

Metric	Formula	Observed result	
Correct identification rate (genuine)	Correct genuine IDs / all genuine attempts × 100	10 / 12 × 100 = 83.33%	10 of 12 genuine-student observations were correctly identified.
Wrong-identification rate	Wrong genuine IDs / all genuine attempts × 100	0 / 12 × 100 = 0%	No genuine-student observation was assigned to another enrolled identity.
False-rejection rate (FRR)	Genuine attempts rejected / all genuine attempts × 100	2 / 12 × 100 = 16.67%	Two genuine-student observations were rejected under low lighting.
False-acceptance rate (FAR)	Unknown attempts accepted / all unknown attempts × 100	0 / 6 × 100 = 0%	None of the six tested unregistered people was incorrectly accepted.

LIMITATIONS

The prototype uses a traditional Haar Cascade detector and LBPH recognizer. Recognition may be affected by illumination, blur, pose, camera quality, occlusion, and similarity between enrolled faces. The ten enrollment images per student provide training examples, but the manuscript does not yet establish an independent labelled test set or a controlled comparison with CNN/embedding-based methods.

The screenshots included in this manuscript document the student registration interface, successful face recognition, administrator dashboard, attendance reports, and rejection of an unregistered face image. The unregistered-face screenshot demonstrates the system's response to a face image that is not included in the registered dataset. This image-based test should be distinguished from a separate live-person evaluation.

The current application primarily stores positive attendance entries. Absent values shown in summaries may be inferred by report logic rather than independently recorded for every student on every working day. Accurate absence calculation requires a defined academic calendar, working-day rules, holidays, and explicit absent-record generation. The current project materials also do not demonstrate encryption of stored face images/model files, a formal retention policy, or a complete privacy/security audit.

Although the system was evaluated under good and low lighting conditions, a separate controlled experiment involving different facial orientations and viewing angles was not conducted. Therefore, the effect of facial-angle variations on recognition performance has not been quantitatively established. Future testing will include frontal, left-side, and right-side facial orientations. Do not invent angle-based results.

Biometric Data Privacy and Security

Facial images and face-recognition templates are sensitive personal data in practical use. An educational deployment should collect facial data only for a clearly stated attendance purpose and should provide students with an understandable notice describing what is collected, why it is used, who can access it, how long it is retained, and how deletion or correction requests are handled. Where required, valid consent or another appropriate lawful basis must be established before collection.

ISSN 2278-2540 | DOI: 10.51583EMAS | Volume XV, Issue X, October 2026

Recommended safeguards include administrator-only access, strong password hashing, role-based authorization, HTTPS for browser-to-server communication, encryption of facial images and model files at rest, secure backups, access logging, retention limits, and deletion of data when it is no longer required. A non-biometric attendance alternative should be available for students who cannot or do not wish to use face recognition, subject to institutional policy.

The present project uses a local SQLite database and local dataset/model files. The materials reviewed for this paper do not establish that encryption at rest, HTTPS deployment, role-based access, or a formal retention/deletion mechanism has been implemented. These safeguards are therefore presented as deployment requirements and future improvements, not as completed system features. Any deployment in India should be reviewed against applicable requirements, including the Digital Personal Data Protection Act, 2023 [8], and institutional policy.

Future Scope

- Collect varied facial samples and evaluate recognition under different lighting and camera conditions.
- Introduce an academic calendar, holidays, working days, and explicit absent-record generation.
- Strengthen administrator authentication through password hashing and role-based access control.
- Develop a mobile-friendly interface for faculty and students.
- Add secure backup and optional cloud synchronization for authorized access.
- Provide low-attendance notifications and subject-wise, semester-wise, and class-wise analytics.
- Evaluate advanced face-recognition models and anti-spoofing or liveness checks after considering accuracy and resource requirements.

CONCLUSION

This paper presented a web-based Smart Attendance Automation prototype that integrates student management, camera-based face detection and recognition, attendance storage, and report generation. The implementation uses Python and Flask for backend processing, OpenCV Haar Cascade for face detection, LBPH for face recognition, and SQLite for student and attendance records. Preliminary live-camera testing correctly identified six of six enrolled-student observations under good lighting and four of six under low lighting; the other two low-light observations were rejected. Six of six unregistered-person observations were rejected, and no wrong identity assignment or false acceptance was reported in the supplied tests. The results correspond to an 83.33% genuine-student identification rate, a 16.67% false-rejection rate, and a 0% observed false-acceptance rate for this limited sample. The findings suggest that lighting is an important factor for the current Haar Cascade and LBPH implementation. Larger-scale testing, repeated trials, additional pose conditions, and stronger biometric data safeguards are required before deployment in a real educational environment.

The prototype provides an administrator dashboard, student registration, face-data capture, camera-based attendance, daily and monthly reports, individual student attendance history, charts, and CSV export. The preliminary evaluation now provides initial numerical evidence for the tested lighting conditions and unknown-person rejection. Further evaluation should use a larger participant pool, repeated live-camera attempts, varied facial angles, different cameras and distances, and measured processing time. Biometric privacy safeguards must also be implemented and verified before institutional deployment.

REFERENCES

1. P. Viola and M. J. Jones, “Robust real-time face detection,” in Proc. IEEE International Conference on Computer Vision (ICCV), 2001, p. 747, doi: 10.1109/ICCV.2001.937709.
2. T. Ahonen, A. Hadid, and M. Pietikäinen, “Face description with local binary patterns: Application to face recognition,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 28, no. 12, pp. 2037–2041, Dec. 2006, doi: 10.1109/TPAMI.2006.244.

ISSN 2278-2540 | DOI: 10.51583EMAS | Volume XV, Issue X, October 2026

3. A. Budiman, F. Fabian, R. A. Yaputera, S. Achmad, and A. Kurniawan, "Student attendance with face recognition (LBPH or CNN): Systematic literature review," Procedia Computer Science, vol. 216, pp. 31–38, 2023, doi: 10.1016/j.procs.2022.12.108.
4. I. D. M. B. A. Darmawan, "Implementation of face recognition for attendance recording in online learning," Jurnal Ilmu Komputer, vol. 16, no. 2, 2023, doi: 10.24843/JIK.2023.v16.i02.p06.
5. N. Beri, V. Srivastava, and N. Malik, "Face recognition attendance management system using LBPH and Haar Cascade," Journal of Trends in Computer Science and Smart Technology, vol. 6, no. 3, pp. 257–273, 2024, doi: 10.36548/jtcsst.2024.3.004.
6. S. S. Tiwari, S. Kumar, R. V. S. Praveen, G. Kumar, S. C. Sekhar, and R. Bhokare, "Automated smart attendance system for higher education using a lightweight DNN model-based face recognition," in Proc. 8th International Conference on Computing Methodologies and Communication (ICCMC), 2025, pp. 1513–1518, doi: 10.1109/ICCMC65190.2025.11140746.
7. B.-T. Nguyen-Tat, M.-Q. Bui, and V. M. Ngo, "Automating attendance management in human resources: A design science approach using computer vision and facial recognition," International Journal of Information Management Data Insights, vol. 4, no. 2, Art. no. 100253, 2024, doi: 10.1016/j.jjimei.2024.100253.
8. Government of India, "The Digital Personal Data Protection Act, 2023," Act No. 22 of 2023, Aug. 11, 2023. [Online]. Available: https://www.indiacode.nic.in/indiacode/handle/123456789/22037?view_type=browse
9. OpenCV, "Cascade Classifier and Face Recognition Documentation." [Online]. Available: https://docs.opencv.org/
10. Pallets, "Flask Documentation." [Online]. Available: https://flask.palletsprojects.com/
11. Python Software Foundation, "Python Documentation." [Online]. Available: https://docs.python.org/3/
12. SQLite, "SQLite Documentation." [Online]. Available: https://www.sqlite.org/docs.html