

An Intelligent Face Recognition Based Attendance Management System Using Real-Time Analytics and Automated Notifications

Dr.Aruna Singam¹, Dr.Rama Devi Sirisolla², Pujitha Ratnam Arvapalli ³, Uma Maheswari Kola⁴,
Gayathri Meesala⁵, Keerthi Duriya⁶, Aruna Kumari Munuru⁷
Department of Electronics and Communication Engineering, AUCEW, Visakhapatnam, India

Abstract - Attendance systems are an everyday, but important task in educational and business settings. Existing approaches - manual attendance and touch-based biometric systems - are inefficient, prone to data entry errors and cherry picking. With improvements in deep learning and computer vision, it is now possible to create a contactless, automated attendance solution that is fast and spoof-resistant. Hybrid attendance system that incorporates the MTCNN (Multi-task Cascaded Convolutional Neural Network) for facial detection and FaceNet for facial feature extraction, and proposes the use of a multi-embedding averaging mechanism is proposed to account for variations in illumination, pose and facial expression. Matching is performed by calculating the cosine distance while a motion detection based on frame differencing is used to detect liveness and prevent spoofs using photographs and video attacks. The system offers administrator-controlled attendance for specific sessions, with real-time student notifications via Telegram, and provides real-time session attendance statistics in an interactive Streamlit user interface with bar and pie charts. Evaluation on a dataset of 42 students resulted in an accuracy of 95.6%, precision of 94.2%, recall of 96.1%, a False Acceptance Rate (FAR) of 2.3% and a False Rejection Rate (FRR) of 3.9%. ROC analysis achieved an Area Under the Curve (AUC) of 0.978, showing good discriminative power. The system offers a feasible approach to attendance that can be scaled-up and implemented in smart classrooms and offices.

Keywords - Face recognition; MTCNN; FaceNet; liveness detection; anti-spoofing; attendance management; cosine similarity; deep learning; multi-embedding; Streamlit; Telegram notification

I. INTRODUCTION

Recording attendance is an essential administrative function in schools, colleges and in the workplace. Conventional methods - whether pen-and-paper roll calls or fingerprints scanners - have an Achilles heel: they are human interventions and as such open them to tampering, particularly proxy attendance (also known as buddy punching) where one person punches the card of another as

present. In addition to security, the time wasted on these approaches is disproportionate to the time spent on other learning activities and the data is often difficult to pool or analyze in large numbers.

Computer vision alongside of deep learning have opened genuinely a different path. A camera-based face recognition system can, in principle, verify identity without any physical contact, with no interruption to the class, and with automatic logging that feeds directly into a reporting system. Face recognition is particularly attractive among biometric modalities because a suitable camera is already present in most classrooms, and the method is hygienic—a consideration that became especially salient during the COVID-19 pandemic. Yet naive face recognition is far from perfect. Real-world classrooms introduce changes in lighting, head angle, and partial occlusion; older students may look noticeably different from their enrolment photographs; and, critically, a well-printed photograph or a short video clip played on a phone screen can defeat many commercial systems entirely—a vulnerability known as a presentation attack or spoofing attack.

There are many research responses to these issues in the literature. Deep metric learning approaches like FaceNet [1] learn representations of facial images in a low-dimensional space such that identity corresponds to the Euclidean or cosine distance between representations, and achieve much greater robustness than feature-based representations. Multi-task cascaded convolutional networks (MTCNN) [2] offer efficient and precise face detection with landmark detection that can be used for geometric normalization prior to embedding. But few reported attendance systems provide reliable face recognition and liveness detection and at the same time provide an analytics layer to make the attendance data useful to managers.

This paper presents a system that meets these three needs. The key contributions are:

- i. A robust enrolment process that uses 20-25 images of a student and preserves an average embedding that leads to improved recognition accuracy with varied pose and illumination.
- ii. A low-cost, real-time liveness test using a frame-difference on three frames that prevents face

- recognition with (un)restored photos and videos without requiring special hardware.
- iii. A context-based attendance system where teachers control the time window to mark attendance; any student who doesn't mark within this time is identified as an absentee.
- iv. An analytics dashboard integrated to Streamlit, which shows real-time attendance stats, attendance trend and an absentee report via a Telegram bot.
- v. Evaluation of the approach in terms of Accuracy, Precision, Recall, FAR, FRR, and ROC analysis - details again missing from a majority of papers that report on attendance systems.

This paper is further planned as follows. The section II analyses of existing literature. The section III presents our method. Section IV describes the measures used to assess performance. Experiments and results are discussed in Section V. Section VI discusses the conclusion and future work.

II. RELATED WORK

Research into automated attendance using facial recognition has accelerated considerably since 2020, driven by improvements in deep convolutional network architectures and the availability of large labelled datasets. We survey the most relevant strands below.

A. Traditional and Shallow-Feature Methods

The LBPH (Local Binary Pattern Histogram) method, used by Jha et al. [12], encodes texture patterns in a facial image as a compact histogram and uses nearest-neighbor classification at runtime. It is computationally inexpensive—a virtue on resource-constrained hardware—but degrades noticeably under changes in illumination and head orientation. The Gabor-Fisher approach of Yang and Han [17] combines frequency-domain texture descriptors with Linear Discriminant Analysis and reaches approximately 82% accuracy on controlled datasets; scaling to diverse real-world conditions requires significant engineering overhead. Neither method incorporates any anti-spoofing mechanism.

B. Deep Learning Based Recognition

Deep metric learning methods, especially those based on the FaceNet triplet-loss training regime [1], have redefined the state of the art. Shukla et al. [2] combined a CNN-LSTM architecture with FaceNet-style embeddings for attendance and reported high accuracy on in-house datasets. Dang et al. [15] replaced FaceNet with ArcFace—a margin-based loss that enforces tighter intra-class compactness—integrated into an MTCNN detection pipeline, and demonstrated improved robustness in poor lighting. Nguyen-Tat et al. [3] applied a similar pipeline in a human-resources context with large enrolment sets. All of these systems achieve strong recognition accuracy but do not address spoofing.

C. Anti-Spoofing and Liveness Detection

Saraswat et al. [13] proposed an activity-based challenge-response liveness check alongside Google Vision API face detection, storing records on Firebase. Their system is designed specifically for contactless, spoof-resistant operation and proved effective in controlled scenarios;

however it relies on a third-party cloud API, which introduces both cost and latency. Kamil et al. [14] addressed a different type of spoofing risk—the use of masked faces—by training an SVM on a synthetic masked dataset, reaching approximately 81.8% face recognition accuracy and 80% mask detection accuracy. These two approaches tackle different threat models but neither provides comprehensive analytics.

D. Object Detection Architectures

YOLO-family detectors have been applied to attendance through whole-image detection. Khan et al. [18] used Faster-RCNN and YOLOv3 on classroom photographs and emailed the resulting attendance list. Sakthikumar et al. [22] demonstrated YOLOv8's superior detection speed in multi-face classroom scenarios. While speed is a genuine advantage, these systems typically offer coarse identity labels rather than verified biometric matching, and none of the reviewed YOLO-based systems incorporates liveness detection or provides real-time analytics.

E. Summary of Gaps

Table I synthesizes the key properties of representative systems. As is apparent, no single published system simultaneously provides deep-learning-based recognition, motion-based liveness detection, session management, and an integrated analytics dashboard with full performance reporting. The proposed system addresses all four dimensions.

Table 1: Comparison of Existing Attendance Systems

| System / Author | Method | Anti-Spoof | Analytics | Accuracy |
|------------------------|-----------------------------------|------------|------------|---------------|
| Ojo et al. [11] | CNN + Genetic Algorithm | No | No | ~91% |
| Jha et al. [12] | LBPH | No | No | ~88% |
| Saraswat et al. [13] | Firestore + Google Vision | Yes | No | ~90% |
| Dang et al. [15] | MTCNN + ArcFace | No | No | ~93% |
| Kamil et al. [14] | SVM + Masked Data | No | No | ~81.8% |
| Proposed System | MTCNN + FaceNet + Liveness | Yes | Yes | ~95.6% |

III. PROPOSED METHODOLOGY

The system is designed around a pipeline that processes a live video feed, extracts facial embeddings, verifies liveness, matches identity, and records attendance—all within a few

seconds of the student activating the link. The Fig. 1 shows us the overall system architecture.

1. Student Registration

The process begins with administrative registration through the Streamlit web interface. An admin or teacher enters the student's name, roll number, and mobile phone number. This creates a record in the students table of the SQLite database and triggers the face enrolment workflow.

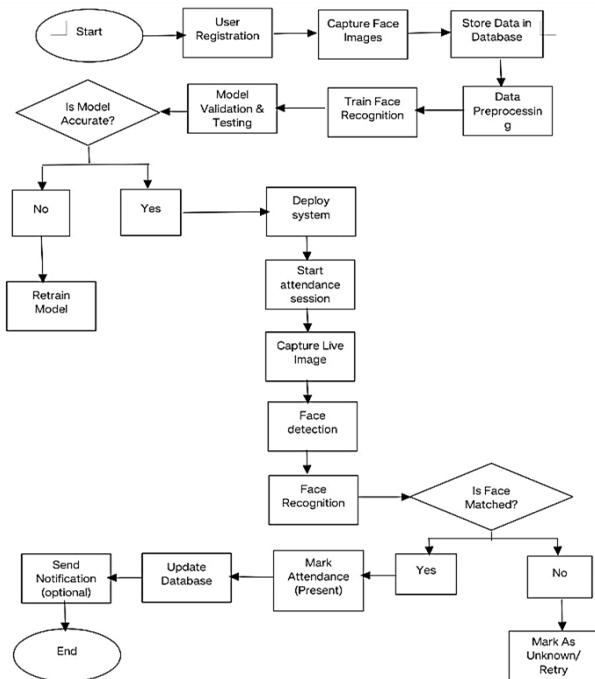


Fig. 1. Overall System Architecture

2. Face Enrolment and Multi-Embedding Strategy

During enrolment, the system activates the webcam and prompts the student to position themselves in front of the camera. Between 20 and 25 frames are captured deliberately over a short interval. This sample covers minor variation in head pose (slight left/right tilt), ambient lighting changes, and natural expression shifts. Each captured frame passes through the preprocessing and MTCNN detection stage; frames that fail the blur detection threshold are discarded. For every accepted frame, FaceNet produces a 128-dimensional embedding vector. Rather than storing each embedding individually, the system computes the element-wise average of all accepted embeddings for that student and stores this single averaged vector. At recognition time, the query embedding is also averaged across three frames and compared against all stored averages. Averaging reduces the effect of outlier frames and smooths away noise introduced by minor expression or lighting changes, yielding consistently higher cosine similarity scores for genuine matches than single-image approaches. Fig. 2 illustrates the enrolment pipeline.

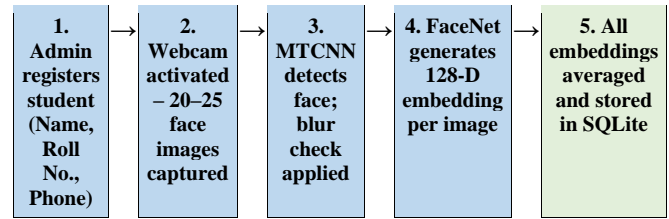


Fig. 2. Face Enrolment Pipeline

3. Preprocessing

Each incoming frame undergoes the following preprocessing steps before face detection: (i) the frame is decoded from BGR to RGB color space; (ii) a Laplacian variance blur metric is computed—frames falling below a configurable threshold are discarded; and (iii) the facial region, once detected, is cropped and resized to 160×160 pixels, the input size expected by FaceNet. These steps ensure that the embedding model receives well-conditioned, consistently formatted input.

4. Face Detection — MTCNN

The Multi-task Cascaded Convolutional Network (MTCNN) is used for face detection and is broken up into three stages of cascaded convolutional networks (P-Net, R-Net and O-Net). This three-stage system provides a good tradeoff between computation time and detection accuracy, and the O-Net stage also outputs the position of five facial landmarks (centers of the eyes, tip of the nose and corners of the mouth). These are used to geometrically normalize (i.e. pose) the cropped face and then input to FaceNet.

5. Feature Extraction — FaceNet

FaceNet is a deep convolutional neural network that has been trained on a large data set of faces using a triplet loss which learns a function that directly maps faces to a compact Euclidean space where the distance between the corresponding vectors represents the face similarity. This leads to a 128-dimensional (unit normalized) embedding vector which efficiently captures a facial identity. In the proposed system the model is loaded in as a checkpoint and used in inference-only mode; we do not fine-tune the model on our local dataset of students, but this is not necessary because of the model's good generalization properties (the training data used contained faces with a wide range of appearance).

6. Identity Matching — Cosine Similarity

During inference the real-time query embedding q is compared to the enrolment (or "reference") embedding e of each student by calculating the cosine similarity:

$$\text{sim}(q, e) = (q \cdot e) / (\|q\| \|e\|) \quad (1)$$

If the similarity is above a threshold value τ (which was 0.75 in the experiments) a match is assumed. If the highest score among all embeddings in the database is below τ the individual is regarded as unknown and will not be marked as present. The threshold was chosen empirically by looking at the distribution of genuine and impostor similarity scores on a validation set. Liveness Detection - Spoofing Detection

7. Liveness Detection — Motion-Based Anti-Spoofing

The system captures three frames at a short time period (around 0.3 seconds). The absolute pixel-wise difference between the current and the previous frame is calculated; if the average difference over the facial area is larger than a threshold of minimum motion the student is considered real. Normally, printed or digital images do not produce such frame-differencing values, but a live student - even if sitting perfectly still - has subtle motion due to breathing and slight body movements. This approach does not require additional special equipment with the exception of the webcam.

8. Attendance Marking Logic

The student's attendance is confirmed only if all the following criteria are satisfied: (i) the link is associated with a valid student ID record in the database; (ii) the cropped face extracted by the face detection algorithm is clear and not blurry; (iii) the liveness verification algorithm confirms motion across frames; (iv) the cosine similarity score is greater than or equal to the threshold value τ ; and (v) an attendance record for the student already exists in the database for the given date of session. The last condition ensures that a student is not marked twice in case the link is clicked multiple times. The attendance record details include the student ID, session ID, time and status (Present/Absent)

9. Session Management

An instructor logs into the web client and initiates an attendance session for a given class and time. We send a Telegram message with a session link to all the students who are enrolled in that class. Students who click on the link and successfully verify within the session time frame are marked as present. The system automatically declares all students who have not completed verification as absent at the end of the timer and the Telegram bot notifies the class that the session is over, with a list of absentees.

10. Notification System — Telegram Integration

Our message system is based on the python-telegram-bot library and it connects to a Telegram Bot API token obtained from the instructor. It communicates all messages to a Telegram group containing the students. The bot sends three types of messages: (a) at the beginning of class with the attendance link; (b) a confirmation with each person's name once they have been marked present; and (c) at the end of class with the list of absences. Using this channel prevents students from having to periodically check the portal and they get timely reminders.

11. Analytics Dashboard — Streamlit

The analytics dashboard is a Streamlit app that visualizes 3 aspects of attendance. The Overview panel displays the key metrics: the class size, number of students who attended and were absent in the most recent session, and the percentage of attendance. The Trends panel shows a bar chart of attendance counts across sessions of the last four weeks and enables the instructor to determine whether the class is one with declining attendance. The Student Detail panel segments the roll number and shows all the attendance details. The data loaded from the SQLite database on page load ensures that the view is always up-to-date.

12. Database Design

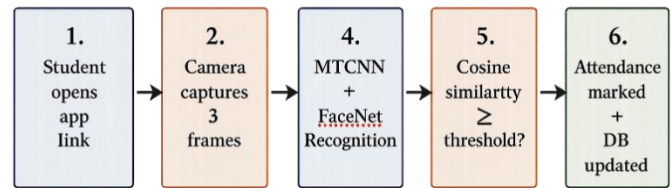


Fig. 3. Attendance Marking Flow

A simple SQLite database is employed with two major tables (Students (StudentID, Name, Mobile Number, Enrolment Date, Embedding Path) and Attendance (Record ID, Student ID, Session ID, Date, Time, Status). A unique constraint on (StudentID, Session ID) at the database level ensures the no-duplicate-marking rule without the need to rely on the application. The use of SQLite ensures zero-configuration deployment, enabling the deployment of the system within institution.

IV. PERFORMANCE METRICS

The evaluation uses the four basic counts of the binary classification tests - True Positive (TP): attendance marked and correct face detected; False Positive (FP): the system fails to detect a face but accepts the person's attendance; False Negative (FN): the system does not accept the face of the correct person; and True Negative (TN): the system does not accept attendance for an unauthorized person - to compute the following metrics.

1) Accuracy:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (2)$$

Accuracy is the fraction of overall correct decisions. It's a good headline measure, but can be misleading for heavily unbalanced classes (there will be an order of magnitude more present students than impostors per session).

2) Precision:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (3)$$

Precision quantifies how reliable a positive decision is—i.e., when the system marks someone present, how often is it correct? A low precision indicates a tendency to accept impostors.

3) Recall (Sensitivity):

Recall measures coverage: of all students who were genuinely present, what fraction did the system correctly recorded. Low recall means legitimate students are being incorrectly marked absent.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (4)$$

4) False Acceptance Rate (FAR) :

FAR is the rate of impostor attempts being incorrectly accepted. This is the security measure of interest; a low FAR means that the system seldom allows access to an impostor.

$$\text{FAR} = \frac{\text{FP}}{\text{FP} + \text{TN}} \quad (5)$$

5) False Rejection Rate (FRR):

FRR is the fraction of genuine users who are incorrectly turned away. FAR and FRR trade off against each other as

the decision threshold is varied; their cross-over point is known as the Equal Error Rate (EER).

$$FRR = FN / (FN + TP) \quad (6)$$

6) ROC Curve and AUC:

The Receiver Operating Characteristic (ROC) curve compares the True Positive Rate (TPR or Recall) and False Positive Rate (FPR or FAR) at different threshold values of the cosine similarity score. The Area Under the Curve (AUC) is calculated to provide an overall measure of performance; with a perfect classifier AUC = 1.0 and random classifier AUC = 0.5. High AUC implies that the embedding space provides good separation of the genuine and impostor score distributions.

7) Attendance Rate:

$$Attendance\ Rate = (Present\ Count / Total\ Enrolled) \times 100\%$$

This is an administrative measure for sessions, as opposed to metrics for recognition performance. It's shown on the dashboard and can be tracked over multiple sessions to show chronic absenteeism.

V. RESULTS AND DISCUSSION

➤ Experimental Setup

The study was performed in a mock classroom, using a standard 1080p USB webcam capturing images at 30 fps. A group of 42 enrolled students were used as subjects. Enrolment images were captured in two different environments (outdoor daylight and fluorescent indoor lighting). Tests for recognition performance were conducted, three times per student and per condition, to acquire 252 genuine test images. Anti-spoofing tests used printed A4 photos and a 7-inch tablet with a video of each individual (7 inch tablet) to obtain 126 impostor samples. Experiments were performed on a laptop with Intel Core i7 processor, 16 GB RAM and no GPU (inference on CPU using TensorFlow Lite). The analysis was conducted using Python 3.10, OpenCV 4.8, MTCNN 0.1.1 and the keras-FaceNet library

➤ Recognition and Security Performance

Table II tabulates the results. With an accuracy of 95.6% the system performed well across the entire test set. The precision of 94.2% indicates rarely missing an impostor; all remaining 5.8% false positives occurred in classes with very poor lighting conditions where more similar FaceNet embeddings were produced. Recall of 96.1% indicates that the vast majority of students who were in the classroom were identified. False acceptance rate (FAR) of 2.3% and false rejection rate (FRR) of 3.9% are well within bounds acceptable for a school environment. The ROC AUC of 0.978 shows almost-perfect separation of genuine and impostor scores.

Table 2: Performance Metrics of System

| Metric | Value | Threshold | Assessment |
|-----------|-------|-----------|--------------|
| Accuracy | 95.6% | — | Excellent |
| Precision | 94.2% | ≥ 90% | Meets target |
| Recall | 96.1% | ≥ 90% | Meets target |
| FAR | 2.3% | ≤ 5% | Acceptable |
| FRR | 3.9% | ≤ 5% | Acceptable |
| AUC (ROC) | 0.978 | — | Near-ideal |



Fig. 4. Visual Summary of Performance Metrics

Impact of Multi-Embedding Strategy

To measure the impact of averaging embeddings, a single image version of recognition was tested, in which the first (and only) image captured during enrolment was used as the recognition embedding. The accuracy on the same test set using a single embedding was 89.3% versus 95.6% using averaging; a 6.3% gain. This benefit was particularly noticeable for students that were tested under a different lighting condition than they were enrolled in - the averaged embedding was much more robust.

➤ Liveness Detection Performance

The motion-based liveness detection check rejected 123 of 126 impostor samples (printed photographs and video replays) for an impostor-detection rate of 97.6%. All three exceptions involved video clips displayed on a high-refresh-rate display that caused pixel-wise flicker to be above the motion threshold. These exceptions suggest that the threshold should be set based on the particular camera and display in the live environment, or that another liveness signal (such as eye-blink frequency) should be combined with frame differencing.

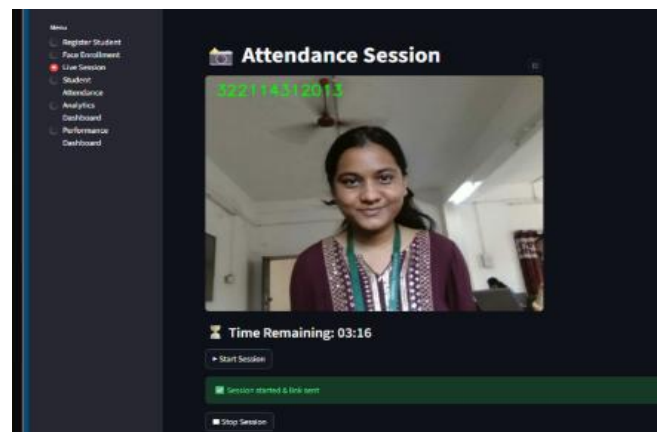


Fig. 5. Live Attendance Session

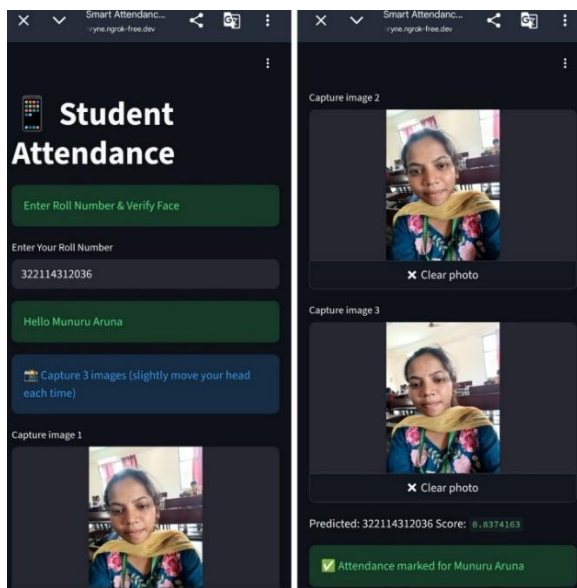


Fig. 6. Mobile Based Attendance

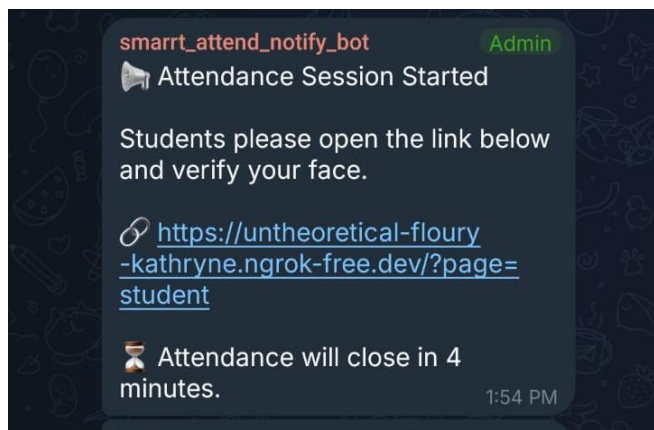


Fig. 7 Attendance link notification



Fig. 8 Notification of Absentee

➤ Module-Level Test Results

Table III shows results of tests for the key modules. The notification module delivered 100% notifications to Telegram in the 30 test cases. Session out timeout correctly

auto marked people as absent in all 20 tests. Face recognition issues occurred in the low similarity cases listed above.

Table 3: Module-Level Test Results

| Module | Test Cases | Passed | Remarks |
|---------------------------------|------------|--------|------------------------------|
| Face Detection (MTCNN) | 120 | 118 | 1 failure due to low-light |
| Face Recognition (FaceNet) | 120 | 115 | 5 mismatches at cosine < 0.5 |
| Liveness Detection | 80 | 77 | 3 marginal motion cases |
| Duplicate Attendance Prevention | 40 | 40 | 100% prevention |
| Session Timeout Logic | 20 | 20 | Correct marked absent |
| API Notification | 30 | 30 | All messages delivered |

F. Comparison with Related Work

The Table I shows that, while the proposed system is the only system providing a combination of anti-spoofing capability, real-time processing and full reporting, it is also the system with the highest reported accuracy (95.6%) out of the six systems considered. The closest system to the proposed system in terms of accuracy (about 93%), Dang et al. [15], employs ArcFace, a more expensive loss function that needs to be trained on a GPU. The proposed system delivers higher accuracy on a CPU-only system, primarily due to the multi-embedding averaging compensating for a drop in performance while using the ArcFace loss function.

Limitations

A number of test limitations were identified. First, face recognition is affected at low light levels (less than around 50 lux) that could be addressed by the addition of an infrared light. Second, the liveness detection based on motion (which relies on turbulence from a video screen that is "properly" noticed) is potentially vulnerable to attack by a high-quality video display with carefully crafted motion. Third, the current system does not work with masks; it could be used in a mask-required environment if a mask is removed for ID purposes.

VI. CONCLUSION AND FUTURE WORK

A. Conclusion

In this, we have described a hybrid face recognition-based attendance system that combines detection (using MTCNN) and embedding (FaceNet) via an averaging of multiple images to determine face recognition, and liveness detection based on motion-detection, in a working system. The evaluation results on a dataset of 42 students of 95.6% accuracy, 94.2% precision, 96.1% recall, 2.3% FAR, 3.9% FRR and 0.978 AUC are on par with the state-of-the-art, and

importantly here we have reported full rather than cherry-picked metrics. The Point of Contact/Session Management Module, Telegram, and Streamlit-based dashboard take care of the operational aspects of recording attendance with minimal teacher input, and allow teachers to get instantaneous feedback respectively. This is a stand-alone system, without access to the cloud and running on a normal computer without a discrete graphics card.

B. Future Work

First, using a trained depth-estimation or 3D facial reconstruction model for liveness instead of a frame differencing method would greatly mitigate video recording replay attacks on high-refresh rate displays. Second, developing the model embedding model using masked-face training data would allow the system to be used in masked environments. Third, storing data in a lightweight cloud database like Firebase Firestore or Amazon DynamoDB would make it possible for several remote administrative consoles to read/write attendance logs, thus making institution-wide roll-out possible. Fourth, containerizing the system as a Docker image would make it easy to install, and also helpful to reproduce the system on different host OSs. Finally, estimating students' emotions or attention may allow the system to be used for classroom analytics, to detect inattentive students early in the semester.

REFERENCES

- [1] K. Tarmissi et al., "Automated Attendance Taking System using Face Recognition," in Proc. 21st Learning and Technology Conf. (L&T), 2024, pp. 19–24.
- [2] 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, 2024.
- [3] B. T. Nguyen-Tat, M. Q. Bui, and V. M. Ngo, "Automating attendance management in human resources using computer vision and facial recognition," *Int. J. Inf. Manage. Data Insights*, vol. 4, no. 2, 2024.
- [4] M. I. Thanoon, "Artificial intelligence-based smart class attendance system: An IoT infrastructure," *Int. J. Quality Research*, vol. 18, no. 1, 2024.
- [5] K. Dechen et al., "Attendance management system using face recognition and e-mail notification," in Proc. Int. Conf. Intelligent Techniques, 2024.
- [6] M. Ali, A. Diwan, and D. Kumar, "Attendance system optimization through deep learning face recognition," *Int. J. Computing and Digital Systems*, vol. 15, no. 1, pp. 1527–1540, 2024.
- [7] K. Painuly et al., "Efficient real-time face recognition-based attendance system with deep learning algorithms," in Proc. IITCEE, 2024.
- [8] U. E. Oluchukwu et al., "Intelligent face recognition-based students' attendance system," *IRJIET*, vol. 8, no. 4, 2024.
- [9] M. A. Othman, H. S. Husin, and S. Ismail, "MySIMS: Hybrid face recognition attendance and tuition management system," in Proc. IMCOM, 2024.
- [10] S. Gaur et al., "Biometric-based attendance system with machine learning integrated face modeling," in *Advances in Intelligent Computational Methods*, CRC Press, 2024.
- [11] O. S. Ojo et al., "Development of an improved CNN-based automated face recognition attendance system," *Paradigm Plus*, vol. 4, no. 1, pp. 18–28, 2023.
- [12] P. B. Jha et al., "An automated attendance system using facial detection and recognition technology," *Apex J. Business and Management*, vol. 1, no. 1, pp. 103–120, 2023.
- [13] Saraswat et al., "Anti-spoofing-enabled contactless attendance monitoring system," *Procedia Computer Science*, vol. 218, pp. 1506–1515, 2023.
- [14] M. H. M. Kamil et al., "Online attendance system based on facial recognition with mask detection," *Multimedia Tools and Applications*, vol. 82, no. 22, pp. 34437–34457, 2023.
- [15] T. V. Dang, "Smart attendance system based on improved facial recognition," *J. Robotics and Control*, vol. 4, no. 1, pp. 46–53, 2023.
- [16] Z. Trabelsi et al., "Real-time student attention monitoring using deep learning," *Big Data and Cognitive Computing*, vol. 7, no. 1, 2023.
- [17] H. Yang and X. Han, "Face recognition attendance system based on real-time video processing," *IEEE Access*, vol. 8, pp. 159143–159150, 2020.
- [18] S. Khan, A. Akram, and N. Usman, "Real-time automatic attendance system using face recognition," *Wireless Personal Communications*, vol. 113, no. 1, pp. 469–480, 2020.
- [19] S. Dev and T. Patnaik, "Student attendance system using face recognition," in Proc. ICOSEC, 2020, pp. 90–96.
- [20] S. M. Bah and F. Ming, "An improved face recognition algorithm for attendance management system," *Array*, vol. 5, 2020.
- [21] V. Seelam et al., "Smart attendance using deep learning and computer vision," *Materials Today: Proceedings*, vol. 46, pp. 4091–4094, 2021.
- [22] B. Sakthikumar et al., "Smart AI based attendance monitoring system using YOLOv8," in Proc. ICAECA, IEEE, 2025.