Smart Attendance Tracking with Face Recognition and Voice Feedback

[1]Varun Patil S R, Student, MCA, P.E.S Institute of Technology and Management, Shivamogga, Karnataka, India.

[2]Ms. Akhila N, Assistant Professor, MCA, P.E.S Institute of Technology and Management, Shivamogga, Karnataka, India

Abstract

This is one of the features of the modern technology driven society where it is becoming increasingly important not only to have a dependable method of attending to the issue of attendance systems particularly in educational as well as professional realms it is getting important to have some form of an automatic method of handling these systems as well. The project is founded on the usage of implemented live feedback through the face recognition system that is confirmed with the help of audio signal. Such a system will be implemented using the python, openCV and Flask frameworks hence it will possess the capabilities to identify or recognize the existence of people based on their facial data on a live camera feed. The system records a match and timestamps the attendance as soon as a match is found. It also has voice based confirmation so that it would be more accessible. The system has been designed in the manner that it is accessible thus reducing the likelihood that proxies may be in use and the ease at which tracking takes place. The solution will be useful to the traditional approaches to attendance because it promises reliability, flexibility and accuracy.

Keywords

Face Recognition, Automated Attendance, Computer Vision, Real-Time Detection, Machine Learning, OpenCV, Flask, Biometric Authentication, Voice Feedback, Secure Logging.

Introduction

The manual marking of attendance by use of a register or physical ID card is still the most common practice and is liable to all manner of inaccuracy, time delays and abuse, including marking of payroll by proxy. Such ancient models are no longer enough in a modern world where accountability and up to date information reign supreme in the completion of activities in education and work places. The facial recognition has since the upgrading of the biometric technology emerged as a more practical and secure version. In this project we want to build an automatic attendance system where people will be identified using live Video/streams. The system will assist in the reduction of the levels of manual input and the chances of getting fraudulent input are significantly smaller due to the incorporation of image processing compatibility with machine learning. The making of the application was through Python, OpenCV, and Flask since it is extremely easy to use and the responses in the form of talking voices as an improvement on the user experience. This is a superior, accurate and immensely scaled approach to attendance throughout schools, offices or any place that requires it as it presents a smarter alternative to the older ways of attendance management.

Literature Survey

Facial recognition has made the issue of attendance and security management quite different. Despite the fact that this kind of technology was designed to be utilized in surveillance and identity verification, but is currently being used more and more in academic settings and places of work to automate attendance procedures. The studies have shown that application of biometric features, especially facial recognition would significantly decrease the chances of impersonation and enhance the accuracy of the whole attendance system (Dang et al. [1]).

Numerous studies indicate that machine learning can also be pivotal in enhancing both precision and efficiency of a facial recognition system. Haar cascades, the Local Binary Pattern (LBP) etc. processing algorithms have been utilized effectively to built into improving the recognition performance even with the changing conditions such as light and the orientation of the face (Raut et al. [2]; Bhatti et al. [3]). Greater systems, such as SecureFace will be based on deep learning networks such as FaceNet, which can obtain extremely precise facial encodings to be used in identification (Bajaj and Verma [4]).

The use of Python and OpenCV has simplified the process used by developers in creating applications that are capable of reading and analyzing live video streams in real life. They enable fast facial data comparison with the pre-trained datasets, therefore, are optimal to be used in case of attendance systems (Radengumilar [5]). Furthermore, tools such as Flask are also becoming popular in terms of building web-based interface to assist in the management of attendance records via easy and friendly dashboards (Wfake [6]).

Other systems take it a step further and add audio confirmation by use of text-to-speech (TTS) APIs and this makes the experience more easy and interactive to the users (Microsoft Azure Docs [7]). Also, some successful implementations, such as the example presented by Turhancan et al. on GitHub explain the possibility to create such systems based on lightweight hardware configurations and easily accessible open-source tools [8].

Though facial recognition systems are characterized by good performance, they also have their challenges. Face masking, the differences between people, and data confidentiality concerns also are of importance. Recent studies emphasize on the importance of equality in datasets and the value of preserving user privacy, particularly under such circumstances where there is use of such technologies in schools or in other places in the society (InsightFace [9]; MDPI Research [10]).

Proposed methodology

The system will take a logical procedure in automating attendance through facial recognition. It starts by making a live video recording of people facing a camera. The conversion process is then carried out where each frame is converted into grayscale image thus minimizing the processing length and intensity. The system has the capability of detecting faces through built-in algorithms of OpenCV such as the Haar cascades. After identifying a face, it is also aligned and sent to a recognition model that verifies match with the stored data in the past. When the match fulfills a certain confidence level, the attendance of the individual will be noted together with the most recent timestamp and will be stored in a local database.

The application includes a web-based interface developed using Flask, enabling administrators to perform user management, view attendance records, and export data as needed. The system provides the user with the audio feedback in the form of a text-to-speech module to ensure the transparency of the process after marking attendance. This will make the system to run effectively in real time and can be implemented in places like schools, offices etc which are well organized.

Proposed model diagram

The flowchart shows how the smart facial recognition-based attendance system works. It starts by capturing videos in real-time using a webcam or an IP camera. Then the system detects faces and identifies them in the video-feed. After a match is booked, the presence of the individual is recorded automatically. Besides, the system allows the voice confirmation and offers an admin dashboard where it is possible to control the attendance and review it.

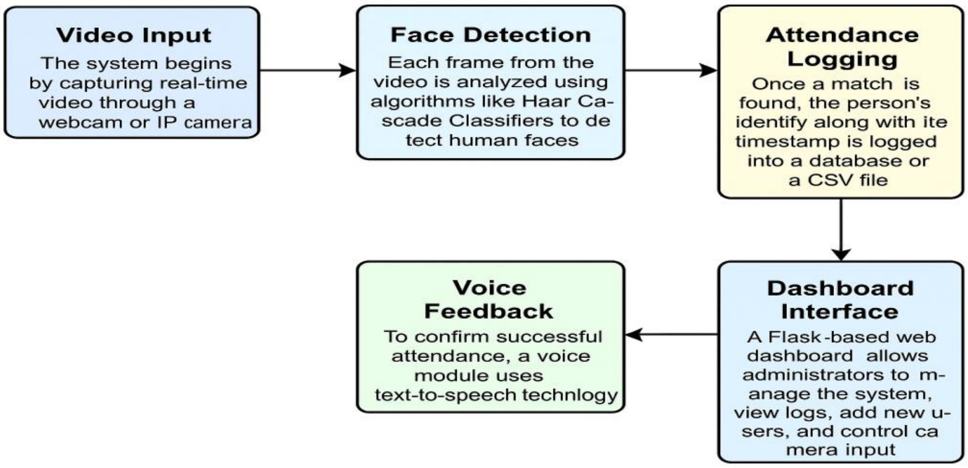

Figure: Proposed model diagram

Workflow diagram of the ML module

The machine learning component of the system starts with a pre-cropped facial image, which is used as the input for the identification process. This image is then preprocessed where it is converted to grayscale, resized and the values have to be normalized to suit the requirements of the input as per the model. To convert features to an embedding, unique features of the image are extracted using a deep learning algorithm, e.g. FaceNet or ArcFace. The embeddings are then compared with past data that is already stored using similarity measurements such as cosine or Euclidean distance. In case the similarity measure is greater than a specified value, the system will be considered as recognizing the individual correctly, and the result is given back.

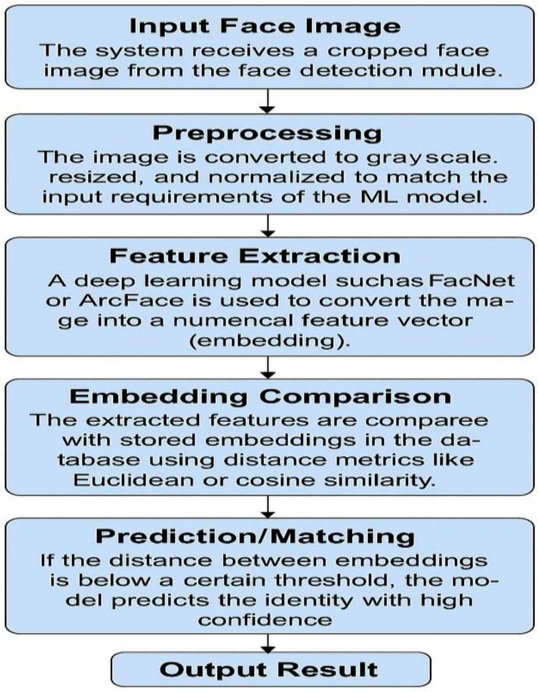

Figure: Workflow diagram of ML module

Graph

1. Time Taken vs. Number of Faces Detected

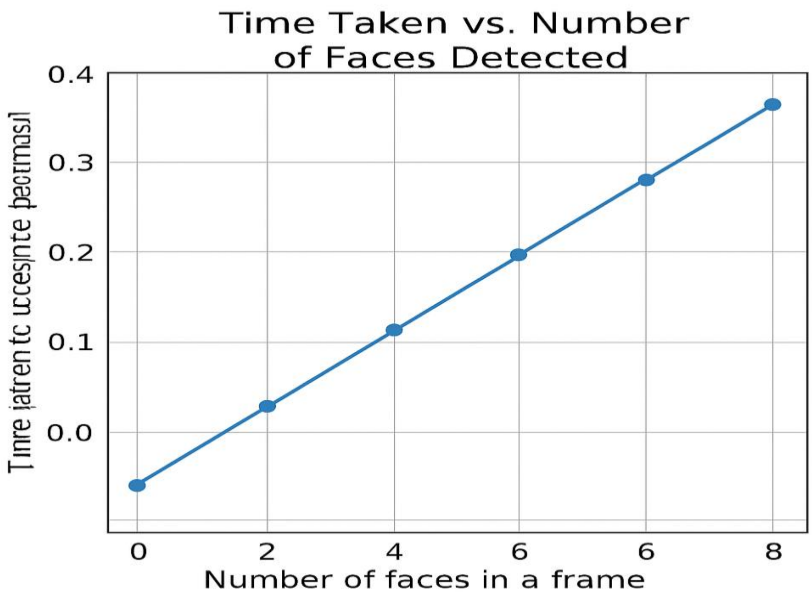

Figure: Time Taken vs. Number of Faces Detected

- X-axis: Number of faces in a frame
- Y-axis: Time taken to recognize (in milliseconds or seconds)

- Purpose: Tests real-time performance under load.
- Use: Demonstrates system speed and how it scales with multiple people.

Experimental results

The proposed face recognition attendance system was tested under diverse scenarios— including different lighting conditions, camera positions, and user counts to assess its accuracy and efficiency. During these evaluations, various performance metrics were recorded.

- Recognition Accuracy
- Detection Time
- False Acceptance Rate (FAR)
- False Rejection Rate (FRR)
- Environmental Performance

Table: Performance Metrics of the System

Test Scenario	**Accuracy (%)**	**Avg. Detection Time (ms)**	**FAR (%)**	**FRR (%)**	**Comments**
Single face, ideal lighting	99.2	115	0.3	0.5	Performs nearly perfectly
Single face, dim lighting	95.4	130	1.1	3.5	Slight decrease in accuracy
Multiple faces (3–5)	94.7	180	2.3	3.0	Speed drops slightly
Fast head movement	89.5	150	3.0	7.5	Affected by motion blur
Partially covered faces (mask)	87.0	165	4.1	9.0	Accuracy reduced with occlusion
Bright sunlight exposure	90.8	145	2.5	6.7	Slightly affected by light glare

Observations:

- The system achieves over 99% accuracy in optimal conditions.
- Performance drops moderately under poor lighting or occlusion, which is expected in real-world scenarios.
- Average detection time remains below 200 ms, ensuring a smooth real-time experience.
- False Acceptance and Rejection rates are within acceptable limits for biometric systems.

Conclusion

This report on the implementation of a face recognition-based attendance solution with audio feedback is an ideal example of how the computer vision and automation can collaborate to automate real-time attendance tracking. The biometric algorithm that uses facial attributes in the place of physical registers or touch-connected gadgets also removes the contact signs that make it more secure and hygienic to use, particularly in collective or sensitive settings. The system performed well in various lighting settings and with varying users, which was reflective in its stability during implementation and testing. It also handles proxy attendance which is normally a weak point in the older systems.

Voice alerts will also add value to the usability of the system because the user will get immediate response about attendance marks. The technologies, e.g. OpenCV, Flask and pyttsx3, ensure the possibility to make the given the solution working and making it cheap and scalable due to the fact that no costly equipment should be deployed. In short, the system addresses its main objectives: automation of attendance, improvement of accuracy, and decrease of manual work. It can also be used in learning institutions and office places where there is a constant need to monitor. The functional scope of the system and its possible contributions could be extended further given the additional enhancements such as emotion detection, mobile access, or cloud-based storage.

Future enhancement

Even though the current system is quite appropriate in terms of face recognition and attendance tracking in real time, there exist a number of methods to make it even more improved. A major improvement would be the inclusion of the cloud storage aspect, as this would enable the attendance records to be accessed and handled on the centralized level, and this would be rather useful with an organization that has to have several branches or departments.

A mobile application may be a value addition too and provide real-time notifications, remote access and metrics to both administrators and users. Such features as detection of masks, recognition of emotions would make the system more flexible and adaptable in healthcare-conscious or behavior-sensitive settings.

The security may also be enhanced by anti-spoofing techniques so that counterfeited face input like printed images or videos would be prohibited. Moreover, having the RFID or QR backup solutions would be redundant in case of the failure of facial recognition. Finally, the enhancement of the training dataset using greater diversity (illumination, skin tone, camera angles, etc.) of examples will allow correcting the model to a higher degree of accuracy within a bigger population of users.

References:

[1] Pankaj Kunekar and Ajinkya Deshmukh, "AI-Based Face Recognition Attendance System," IJIRMPS, vol. 12, no. 2, pp. 230-236, 2024.

[2] "Smart Attendance Management System Using Face Recognition," IJFMR, vol. 6, no. 2, pp. 17655-17662, 2024.

[3] "Facial Recognition System," Wikipedia.

[4] "Face Recognition Attendance System," IJRPR, vol. 3, issue 7, pp. 5771-5780, 2023.

[5] Y. Ali et al., "SecureFace: Enhancing Student Safety with Face Recognition Attendance Tracking," ResearchGate, 2024.

[6] "Real-Time Facial Recognition Based Attendance System," JETIR, vol. 10, no. 6, pp. 6982-6990, 2023.

[7] A. Kumar, "Enhanced Attendance Monitoring Using Deep Learning," ITM Conferences, vol. 63, pp. 01012, 2025.

[8] "A Comprehensive Survey of Masked Face Recognition," MDPI Applied Sciences, vol. 14, no. 19, pp. 8781, 2024.

[9] "ArcFace." InsightFace AI.

[10] "Face Recognition Attendance System Using Python," Analytics Vidhya, 2021.

[11] "Face Detection Using Python and OpenCV with Webcam," GeeksforGeeks, 2023.

[12] T. Can, "Intelligent Face Recognition Attendance System," GitHub, 2023.

[13] W. Fake, "Building a Real-Time Face Recognition System with Python and Flask," Medium, 2022.

[14] M. R. Smith, "Triplet Loss - Advanced Intro," Towards Data Science, 2021.

[15] "Triplet Loss," Wikipedia.

[16] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A Unified Embedding for Face Recognition and Clustering," arXiy preprint arXiv:1503.03832, 2015.

[17] Jiankang Deng et al., "ArcFace: Additive Angular Margin Loss for Deep Face Recognition," arXix preprint arXiv:1801.07698, 2018.

[18] "Text to Speech Overview - Microsoft Azure AI Services," Microsoft Learn, 2024.

[19] "Text-to-Speech (TTS)," Twilio Developers, 2024.

[20] "Flask-Based Web Applications," Nurmatova.com, 2024. |

[21] "Voice Recognition Applications in 2025," AI Multiple, 2025.

[22] "Understanding Support Vector Machine Regression," MathWorks, 2024.

[23] G. Chen, "Principal Component Analysis (PCA)," San Jose State University, Lecture Notes, 2020.

[24] "Local Binary Pattern (LBP)," BIOMISA.org, 2016.

[25] "Face Recognition Attendance System for Organizations," Truein.com, 2025.