Smart Attendance System: Using Facial Recognition Technology

Amaan Ansari[1], Anita Pujari[2], Dr. Kamika Chaudhary[3]

[1,2]Master of Computer Science, M.B.G.P.G. College Haldwani, Kumaun University Nainital (263139)
[3]Assistant Professor, Department of Computer Science, M.B.G.P.G. College Haldwani, Kumaun University Nainital (263139)

Abstract:
The Smart Attendance system automates attendance tracking using advanced face recognition technology integrated with cloud-based data storage. This system includes components for generating face encodings, populating a database with student details, and tracking attendance in real time through live video feeds. The face encoding generation processes student images to create unique facial encodings stored alongside student IDs, Real-time attendance tracking captures live video, detects faces, and matches them against stored encodings to update attendance records in the Firebase Real-time Database. This paper describes a method for Student Attendance system that will record the attendance of the students in the classroom environment automatically and it will provide the facilities to the faculty to access the information of the students easily.

Keywords: Face recognition, python, pycharm, libraries, firebase, opencv, database.

Introduction:

Attendance tracking is essential in educational institutions and workplaces, yet traditional methods like manual sign-ins and roll calls are inefficient and prone to inaccuracies and manipulation. This research explores a Face Recognition Attendance System to overcome these challenges, leveraging the unique characteristics of human faces for secure, contactless, and accurate identification. The system comprises three main components: face encoding generation, database population, and real-time attendance tracking. Face encoding generation processes student images to create unique facial encodings stored with student IDs, managed by the encodegenerator.py script, which also uploads images to Firebase Storage. The database population module, implemented in addtodatabase.py, initializes the Firebase Real-time Database with student details, ensuring necessary information is available for attendance tracking. The real-time attendance tracking module, detailed in main.py, captures live video feeds, detects faces, and matches them against stored encodings to update attendance records in the Firebase Real-time Database.

Additional functionalities include automatic data ingestion from external sources and the generation of monthly attendance reports in Excel format. Extensive testing has demonstrated the system's high accuracy and reliability under various conditions, including changes in lighting and occlusions. Integration with Firebase ensures real-time data synchronization and scalability, making the system suitable for deployment in educational institutions and other organizational settings. The objectives of this study include developing a robust facial recognition algorithm capable of handling diverse conditions, ensuring data privacy and security, integrating the system with existing attendance management systems, and evaluating its performance and user acceptance. The system aims to be scalable and cost-effective, providing significant administrative efficiencies compared to traditional methods.

Literature Review:

Face recognition has been a significant area of research within computer vision, driven by both its practical applications and theoretical interest from cognitive scientists. While other biometric identification methods, such as fingerprint or iris scans, may offer higher accuracy, face recognition remains a primary focus due to its non-invasive nature and because it mirrors the primary way humans identify one another. The technology behind face recognition is steadily advancing towards becoming a universal biometric solution, mainly because it requires minimal effort from users compared to other biometric options. Biometric face recognition is predominantly used in three main areas: time and attendance systems, visitor management systems, and access control systems.

Traditionally, student attendance is recorded manually using attendance sheets provided by faculty members in class. This method is time-consuming and inefficient, especially in large classrooms with numerous students, making it challenging to verify each student individually. The authors of this paper propose a system that uses face recognition to automate attendance recording, ensuring that the presence of enrolled individuals is accurately logged within the respective venue. The proposed system also maintains a log file to keep records of each individual's entry based on a universal system time.

Early efforts in face recognition began in the 1960s with semi-automated systems where key facial features were marked on photographs, and distances and ratios were calculated and compared to reference data. In the early 1970s, Goldstein, Harmon, and Lesk [1] developed a system using 21 subjective markers such as hair color and lip thickness. This method proved challenging to automate due to the subjective nature of the measurements, which were performed manually.

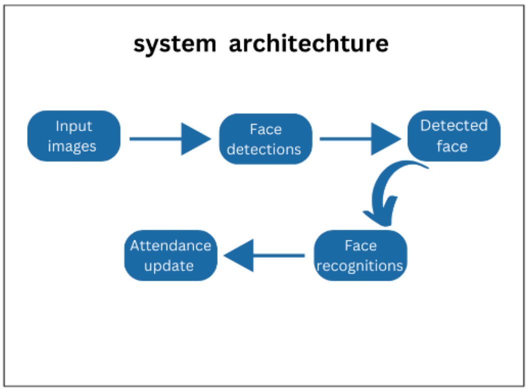
Fisher and Elschlager [2] developed methods to measure different facial features and map them onto a global template, but these features did not provide enough unique data to represent an adult face accurately. Another method, the Connectionist approach [3], attempted to classify human faces using a combination of gestures and identifying markers, typically implemented using 2D pattern recognition and neural network principles. However, this approach required many training faces to achieve reasonable accuracy, limiting its large-scale implementation.

The first fully automated face recognition system [4] was developed based on general pattern recognition principles. It compared faces to a generic face model of expected features and generated patterns for an image relative to this model. This statistical approach relied on histograms and grayscale values, forming the foundation for future advancements in face recognition technology. [5]

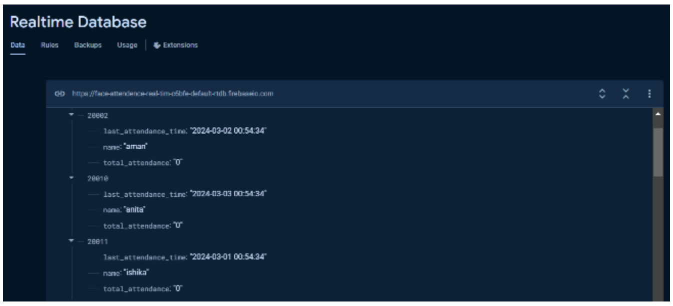
Methodology:

This research paper details how face recognition can be effectively employed to automate attendance tracking, significantly improving accuracy and efficiency in educational settings. The proposed system not only automates attendance recording but also ensures real-time updates and maintains a log for verification purposes, highlighting its potential to streamline administrative tasks and enhance overall management.

The system comprises several scripts, each with specific responsibilities. The encodegenerator.py script handles loading images, detecting faces, generating encodings, and saving them to a database or file. The addtodatabase.py script manages adding new face images or encodings to the database, ensuring no duplicates, and updating the database with new entries. The main.py script captures real-time video feeds, detects and recognizes faces, and logs attendance for recognized faces. A system flow diagram outlines the sequence of operations: initialization with encoding generation and storage, adding new data via the database script, and tracking attendance through real-time video capture and recognition. The programming languages and libraries used include Python for core functionality, JavaScript for a web-based interface, OpenCV for video processing, and various other Python libraries for data manipulation, Firebase interaction, and report generation. The Face Recognition Attendance System utilizes a modular architecture to ensure efficient and accurate attendance tracking. This modular approach supports scalability, maintainability, and ease of development, with each component playing a critical role in the overall functionality, contributing to a robust and automated solution for attendance management.



Implementation and Data Collection:



```
https://face-attendence-real-tim-c6bfe-default-rtdb.firebaseio.com
▾ — 20002
    — last_attendance_time: "2024-03-02 00:54:34"
    — name: "aman"
    — total_attendance: "0"
▾ — 20010
    — last_attendance_time: "2024-03-03 00:54:34"
    — name: "anita"
    — total_attendance: "0"
▾ — 20011
    — last_attendance_time: "2024-03-01 00:54:34"
    — name: "ishika"
    — total_attendance: "0"
```

The project was developed using PyCharm IDE, chosen for its robust features supporting Python development, including code completion, debugging tools, and version control integration.

Key technologies such as Python for its simplicity and extensive library support, face recognition for face detection and recognition, opencv-python for image processing, and firebase-admin for database integration were pivotal in achieving real-time attendance tracking with face recognition. The implementation started by setting up Firebase credentials and configuring the video capture device. Precomputed face encodings were loaded from files to facilitate efficient face recognition. The core functionality involved capturing video frames, detecting faces using face recognition, comparing face encodings, and updating attendance records in Firebase based on recognized faces.

Image Collection:

Images are sourced from a local directory named 'images', where each image corresponds to a unique student identified by their student ID. This structured approach facilitates easy access and management of student data.

Firebase Storage Integration:

Collected images are uploaded to Firebase Storage, ensuring centralized and secure storage. This integration enables seamless access to images across various components of the system. Face encoding extracts numerical representations of facial features crucial for recognition. Leveraging the face recognition library, each image undergoes processing to generate these encodings. The find Encodings function converts images to RGB format and computes corresponding face encodings, which are stored for subsequent recognition tasks. Integration with Firebase encompasses leveraging Firebase Real-time Database and Firebase Storage for effective data management. Firebase is initialized with appropriate credentials, facilitating secure interactions with Firebase Real-time Database and Storage components. Images collected from local directories are uploaded to Firebase Storage, establishing a centralized repository accessible to all system modules. This centralized approach simplifies image management and enhances scalability.

Database Design, Testing and Results:

Database Schema Design outlines the structured representation of student information and attendance records. Each student is uniquely identified by a Student ID and associated with attributes such as name, subject, starting year, total attendance count, current year and last attendance timestamp. This structured schema facilitates efficient data organization and retrieval.

Integration with Firebase Real-time Database involves initializing Firebase using the Admin SDK and configuring database access through a service account key. This setup ensures secure authentication and enables seamless interaction with Firebase services. Storing Data utilizes Firebase's set method to write student information into the database under their respective Student IDs. Data is structured as dictionaries, ensuring clarity and accessibility of stored information. Retrieving Data involves using Firebase's get method to fetch specific student information from the database based on their Student ID. This real-time retrieval capability supports dynamic updates and verification of attendance records during face recognition processes. Real-Time Applications leverage the retrieved data to dynamically verify attendance and update records as needed. For instance, the system can automatically update total attendance counts and timestamps based on real-time face recognition results.

Database Integration:

Firebase Real-time Database is employed as the backend solution for storing attendance records due to its real-time data synchronization capabilities and scalability. Firebase ensures immediate updates to attendance records as faces are detected and recognized, thereby maintaining accurate and current attendance data. Security features like Firebase Authentication are utilized to restrict access to authorized personnel only, ensuring data integrity.

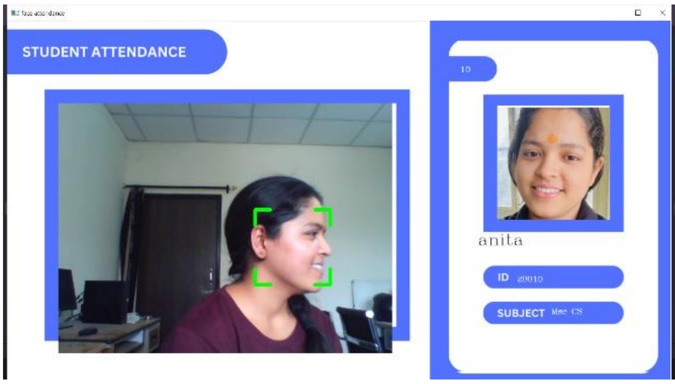
Single Instance Attendance:

The system utilizes single face detection and recognition techniques where each student registers their face through images for training the facial recognition model. During operation, the system captures and matches a single face in real-time against the stored dataset. Upon recognition, attendance records are updated instantly in the Firebase database. This approach minimizes manual intervention and ensures efficient attendance tracking.

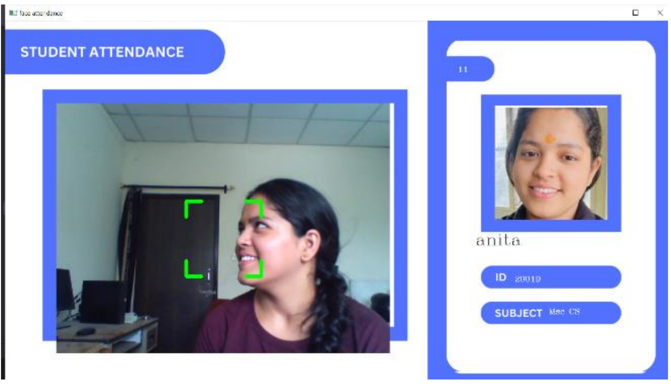
Multiple Instance Attendance:

For environments requiring simultaneous recognition of multiple faces (e.g., classrooms or conferences), advanced computer vision and machine learning techniques are applied. Users register their faces through images to train a robust model capable of identifying multiple individuals in real-time. The system continuously scans the environment, updates attendance records promptly in Firebase as faces are recognized, and ensures accurate attendance logging across diverse settings.

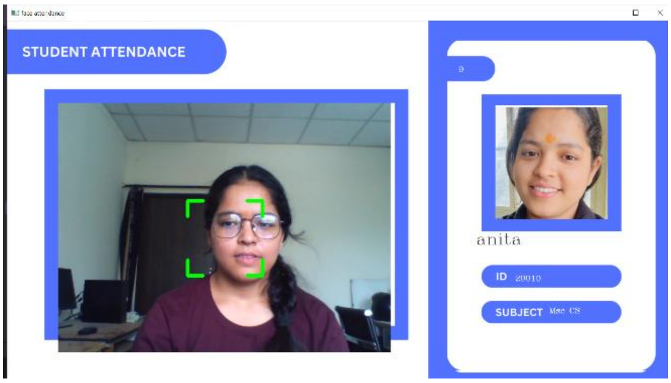
Following images are resultant with different angles:



10
anita
ID 20010
SUBJECT Msc CS



11
anita
ID 20010
SUBJECT Msc CS



9
anita
ID 20010
SUBJECT Msc CS

Testing and Results:

The system's testing phase involved evaluating its performance with faces in various positions and orientations. Results demonstrate its capability to accurately detect and recognize individuals under different conditions, validating its robustness. Real-time updates to attendance records in Firebase Real-time Database reflect the system's effectiveness in maintaining up-to-date attendance information across devices connected to the internet.

Conclusion:

a. **Findings:** Integration of technologies, Efficiency in recognition, Data handling, User interface, Single interface recognition attendance, Multiple interface recognition attendance, Error handling of past attendance system, provide accuracy and time-saving in recording attendance.

b. **Suggestions:** Enhanced Accuracy in Face Recognition Systems Using Deep Learning Techniques, Scalability and Performance Optimization of Real-Time Face Recognition Systems, User Experience

Enhancements in Biometric Attendance Systems, Data Privacy and Security in Cloud-Based Biometric Systems, Real-Time Database Management in Face Recognition Attendance Systems Using Firebase.

References:

1. L. D. H. a. A. B. L. A. J. Goldstein, "Identification of Human Faces," *Proc. IEEE Conference on Computer Vision andPattern Recognition,* vol. 59, pp. 758-760,May 1971.
2. M. A. Fischler and R. A. Elschlager, "TheRepresentation and Matching of Pictorial Structures," in *IEEE Transaction on Computer,* 1973.
3. S. S. R. Abibi, "Simulating Evaluation:connectionist metaphors for studying human cognitive behaviour," *in Proceedings TENCON 2000,* vol. 1, pp. 167-173, 2000.
4. J. S. J. S. L. M. P. a. S. S. L. A. Y. Cui, "Automated Pattern Recognition and Defect Inspection System," in in proc. 5th International Conference on Computer Vision and Graphical Image, May 1992.
5. M. K. D. A. S. a. D. R. P. Nirmalya Kar, "Study of Implementing Automated Attendance System," International Journal of Computer and Communication Engineering, vol. 1, p. 2, 2, July 2012

URLs:

1. Mortaza’s workshop and Robotics: https://youtu.be/iBomaK2ARyI?si=H_dsqWkjqdmI5uyd
2. Ws cube Tech: https://youtu.be/JZZr0PjZsIk?si=9DqHVYOV_oB4DZ5U
3. Pysources: https://youtu.be/5yPeKQzCPdI?si=kxaxMDkHY_yenE1P
4. Firebase Real-time Database: https://console.firebase.google.com/u/0/
5. Design of encoding: https://www.canva.com/