Automated attendance system using Face recognition

Project submitted to the

SRM University – AP, Andhra Pradesh

for the partial fulfillment of the requirements to award the degree of

Bachelor of Technology

In

Computer Science and Engineering

School of Engineering and Sciences

Submitted by

M Balaji – AP23110011520

P Koushik – AP23110011500

P Mehadeesh – AP23110011495

B Dileep – AP23110011504

Under the Guidance of

Mrs. K Kavitha Rani

Professor, Department of CSE

SRM University–AP

Neerukonda, Mangalagiri, Guntur

Andhra Pradesh – 522 502

[04, 2025]

Certificate

Date: 30-04-2025

This is to certify that the work presented in this Project entitled “Automated attendance system using Face recognition” has been carried out by M Balaji, P Koushik, P Mehadeesh, B Dileep under my/our supervision. The work is genuine, original, and suitable for submission to the SRM University – AP for the Award of Bachelor of Technology/Master of Technology in the School of Engineering and Sciences.

Supervisor

K Kavitha Rani
(Signature)
Professor CSE,
SRM University AP.

Acknowledgment

The satisfaction and sense of accomplishment that comes with the successful completion of any project would be incomplete without expressing our heartfelt gratitude to all those who made this journey possible.

We would like to express our sincere and profound gratitude to our respected guide, **Mr. L. Srinivasa Rao**, Department of Computer Science & Engineering, **SRM University, Andhra Pradesh**, for her invaluable guidance, consistent support, and insightful suggestions throughout this project. Her encouragement and expert advice have been instrumental in shaping the direction and outcome of our work.

We are also thankful to all the **faculty members of the Department of Computer Science and Engineering** for providing a supportive academic environment and for extending their help whenever required.

A special thanks to our **peers and classmates**, whose constructive feedback, collaboration, and encouragement motivated us to continually improve and overcome challenges.

Last but not least, we are grateful to our families for their unwavering support and understanding during the course of this project.

We, the team members, extend our heartfelt thanks to everyone who directly or indirectly contributed to the successful completion of this project.

Team Members:

- M Balaji – AP23110010846
- P Koushik – AP23110011500
- P Mehadeesh – AP23110011495
- B Dileep - AP23110011504

Table of Contents Pg No.

Abstract

The increasing demand for automation and digital transformation in educational institutions and workplaces has led to the development of more efficient systems for routine administrative tasks. One such critical task is attendance management. Traditional attendance systems, whether manual or RFID-based, are often time-consuming, prone to errors, and vulnerable to fraudulent practices such as proxy attendance. To address these limitations, our project aims to develop an **Automated Attendance System Using Face Recognition** that harnesses the power of **computer vision** and **machine learning** to offer a more secure, accurate, and efficient solution.

This system is built using **Python** and integrates multiple libraries and tools such as **OpenCV** for real-time image processing, **NumPy** and **Pandas** for effective data manipulation, and **SQLite3** for lightweight, yet powerful database operations. The system captures facial images of individuals through a webcam, encodes and stores the facial features, and uses this data to recognize and verify faces in subsequent sessions. Once a face is identified, the system automatically logs the corresponding attendance into the database, eliminating the need for manual intervention.

The face recognition component of the system uses feature extraction and facial encoding techniques that compare input images with stored facial data to identify individuals with high accuracy. The system has been trained and tested on real-time student data, and it has demonstrated an impressive accuracy of over 95%, showcasing its potential reliability and robustness in real-world scenarios.

In addition to being contactless, this system significantly reduces the time required for taking attendance and ensures that the records are maintained securely and without manipulation. It can be easily extended for use in various environments such as schools, colleges, offices, and even remote learning setups, where attendance tracking is essential.

Overall, this project not only highlights the practical application of computer vision and artificial intelligence but also contributes to the growing need for automation in educational administration and workplace monitoring. It sets a foundation for future enhancements, such as integrating with cloud databases, enabling mobile access, or adding advanced security measures like multi-factor authentication.

1. Introduction

1.1 Background

In recent years, the rapid advancement in **Artificial Intelligence (AI)**, **computer vision**, and **automation technologies** has revolutionized the way routine administrative and operational tasks are performed. One such task is attendance tracking, which is a crucial aspect of any educational institution or workplace. Traditionally, attendance has been recorded manually through registers or logbooks, or semi-automatically using ID card swipes and biometric fingerprint systems. These conventional methods are often plagued by issues such as time inefficiency, human error, and intentional manipulation (proxy attendance), leading to inaccurate records and administrative complications.

In the context of educational institutions, maintaining accurate and tamper-proof attendance records is critical not only for academic tracking but also for compliance with institutional and government regulations. The growing need for a smarter, faster, and more reliable attendance management system has driven the adoption of technology-based solutions. Facial recognition, a subset of computer vision, has emerged as a promising approach due to its **contactless nature**, **real-time performance**, and **high accuracy rates**.

A facial recognition-based attendance system utilizes camera feeds to detect and recognize individuals using their unique facial features. This eliminates the need for physical interaction or the presence of an authority figure to manually mark attendance. It also addresses the hygiene concerns that have become prominent post-COVID-19, making it a safer alternative to traditional fingerprint or card-based systems. Furthermore, by integrating this solution with a database and data processing libraries, institutions can automate record keeping, generate reports, and gain insights into attendance patterns effortlessly.

1.2 Problem Statement

Despite technological progress, many institutions still rely on outdated attendance management systems. Manual methods are **time-consuming**, **labor-intensive**, and prone to **errors and manipulation**. Students or employees may resort to marking attendance on behalf of others (proxy attendance), compromising the authenticity of attendance records. Additionally, manual verification and report generation often result in a significant administrative burden.

Biometric systems such as fingerprint scanners, though more secure than manual methods, present their own set of limitations. These systems require physical contact, which raises concerns about hygiene, especially in the aftermath of the COVID-19 pandemic. Frequent touching of shared surfaces may pose health risks, making contact-based systems less desirable in the current era.

To address these challenges, a non-intrusive, accurate, and efficient solution is required. A facial recognition-based attendance system offers a promising alternative by ensuring **contactless verification**, reducing the risk of disease transmission, and improving the overall efficiency and transparency of the attendance process. This system not only meets the need for hygienic practices but also enhances administrative productivity through automation.

2. Methodology

The development of the **Automated Attendance System Using Face Recognition** was carried out in a structured, **iterative**, and **modular** approach to ensure flexibility, scalability, and smooth integration of each component. By breaking the system into smaller modules, each focused on a specific task, we were able to implement, test, and refine individual functionalities before integrating them into the final system. This approach also allowed us to maintain simplicity and efficiency throughout the development process.

Main Components of the System

1. Face Detection using Haar Cascade Classifiers (OpenCV)

The first step in the face recognition process is face detection, which involves locating the human face within an image or video frame. For this, we utilized the Haar Cascade Classifier provided by the OpenCV library. Haar Cascades are a popular machine learningbased approach to detecting objects in images. In our case, it was specifically used to detect faces in real-time webcam footage.

How it works:

- The Haar Cascade algorithm uses a series of positive and negative images (features) to train a classifier. These features are combined in a cascade-like structure to effectively detect faces within the given image.
- OpenCV provides pre-trained Haar Cascade classifiers, which have been trained to detect faces from various angles and lighting conditions, making them ideal for use in real-time applications.
- The algorithm scans the video frames from the webcam and identifies the location of faces, which are then marked with bounding boxes.

This approach is fast, works well for real-time applications, and is less computationally expensive compared to other deep learning-based methods.

2. Face Encoding & Recognition using face_recognition and OpenCV

Once faces are detected, the next step is face encoding—transforming the face images into a numerical representation of facial features. This is where face recognition and OpenCV come into play.

Face Encoding & Recognition Process:

- Face Encoding: The face_recognition library allows us to extract a unique encoding (a set of 128 measurements) from each face. These encodings are then stored in a database (SQLite3 in our case) for future comparisons.
- Face Recognition: During attendance marking, when a face is detected, the system uses the face_recognition library to compare the detected face with the stored encodings in the database. If the system finds a match, it identifies the individual and logs their attendance.

- The recognition process works by comparing the similarity between the input face's encoding and the stored ones using a distance metric (usually cosine similarity). If the distance is below a certain threshold, the faces are considered a match.

This process is highly accurate and has been trained to recognize faces with over 95% accuracy, even in varying lighting conditions or minor angle differences.

By leveraging OpenCV and face_recognition together, we are able to implement both face detection and recognition seamlessly and efficiently in real-time.

3. Attendance Logging with Pandas and SQLite3

The final step in the system is attendance logging, which involves storing the recognized faces along with their corresponding timestamp in a database. We used Pandas and SQLite3 for this task.

How it works:

- SQLite3 Database: SQLite3 is a lightweight, serverless database that allows easy storage of data locally. Each time a face is recognized, the system logs the student's ID, name, and timestamp into the SQLite3 database. This ensures that attendance is automatically marked without manual intervention.
- Pandas DataFrame: For further processing and manipulation of the attendance data, we utilized Pandas, a Python library designed for handling structured data. Pandas allows us to create a DataFrame to manage the attendance records efficiently. This provides an easy interface to add, remove, or query attendance entries, and even export the data into various formats (e.g., CSV or Excel) for further analysis or reporting.

With this setup, the attendance records are not only automatically logged, but they are also easily accessible for querying or generating reports. Administrators or teachers can instantly access attendance reports, track student attendance patterns, and ensure data integrity.

Summary of Approach:

1. Face Detection: We used OpenCV's Haar Cascade Classifiers to detect faces in real-time from webcam images.
2. Face Recognition: The face recognition library was employed to encode and recognize faces, matching them against stored encodings to identify individuals.
3. Attendance Logging: Recognized faces trigger the logging of attendance using Pandas to handle data and SQLite3 to store it in a database.

This modular approach allowed us to test and refine each component individually before integrating them into the complete system, ensuring smooth operation and effective results.

2.1 Tools and Technologies

The **Automated Attendance System Using Face Recognition** project was built using a combination of tools and technologies that facilitate **real-time image processing**, **data manipulation**, and **database operations**. The following are the core tools and technologies used:

Python 3.8+

Python is the primary programming language used in the development of this system. Python is renowned for its simplicity, readability, and extensive support for libraries, making it an ideal choice for projects involving machine learning, computer vision, and data processing. We used **Python 3.8+** to build the entire backend of the system, including the logic for face detection, face recognition, and database management.

- **Ease of Learning & Use:** Python's simple syntax allowed rapid development and debugging.
- **Extensive Libraries:** Python's vast ecosystem of libraries, such as OpenCV, NumPy, and Pandas**,** streamlined the development of complex functionalities, from image processing to data management.
- **Cross-Platform Compatibility:** Python can run on various platforms, allowing for flexibility in deployment.

Python also provides seamless integration with other technologies and libraries, making it highly suitable for developing a facial recognition-based system that relies on real-time data processing.

OpenCV (Open Source Computer Vision Library):

OpenCV is an open-source computer vision and machine learning library. It is extensively used for real-time computer vision applications, and we used it in this project to handle **face detection** and **image processing**. OpenCV provides various tools and functions that allow us to manipulate images, recognize faces, and process video streams.

- **Face Detection:** We used **Haar Cascade Classifiers** from OpenCV to detect faces in video frames. This feature allows the system to identify and locate a face within an image or video stream, which is crucial for initiating the attendance process.
- **Real-Time Video Capture:** OpenCV allows capturing live video feeds from a webcam, which is essential for monitoring the environment and identifying faces in real-time.
- **Image Processing:** OpenCV provides several image manipulation techniques, such as **image scaling**, **color space conversion**, and **histogram equalization**, which were used to enhance the face detection process, making it more robust in different lighting conditions.

OpenCV's flexibility and rich functionality in computer vision made it the ideal choice for face recognition tasks in our system.

NumPy:

NumPy is a powerful Python library used for handling large multidimensional arrays and matrices, along with a collection of high-level mathematical functions to operate on these arrays. It plays a crucial role in data manipulation within this project.

- **Matrix Manipulation:** NumPy's array structures were used to handle facial image data, where the pixel values of images are represented as arrays. These arrays are then processed during face recognition and encoding.
- **Efficient Computation:** NumPy is highly optimized for performance and can perform mathematical operations much faster than pure Python, which is crucial when processing large datasets or performing real-time operations like face recognition.
- **Interoperability:** NumPy arrays are easily interoperable with other libraries such as Pandas, OpenCV, and face recognition, making it easier to manipulate and process image data.

Using NumPy ensured that the project could handle large amounts of image data efficiently and with minimal latency, contributing to the smooth functioning of the system.

Pandas:

Pandas is a powerful and flexible open-source data analysis and manipulation library. It is used primarily for working with **structured data**, such as data tables (DataFrames), and plays a critical role in managing attendance records in the system.

- **Data Frame for Attendance Records:** We used Pandas' **Data Frame** structure to store, manage, and manipulate attendance data. Each row in the Data Frame corresponds to a student's attendance record, and columns store information such as **student ID**, **name**, and **timestamp**.
- **Efficient Data Handling:** Pandas provides high-level functions to perform operations like **filtering**, **grouping**, **aggregation**, and **sorting**, which are necessary for managing the attendance logs efficiently.
- **Exporting Data:** With Pandas, we were able to export attendance data into various formats, including **CSV** and **Excel**, which are essential for generating reports or exporting data for further analysis.

Pandas provided a seamless way to handle attendance data while ensuring that it was accessible for analysis and reporting.

SQLite3:

SQLite3 is a lightweight, serverless relational database engine that is well-suited for storing small to medium-sized datasets locally. We used SQLite3 to store the attendance records and facial data (encodings) for future recognition and reference.

- **Local Database Storage:** SQLite3 is an embedded database that doesn't require a separate server. This made it ideal for local storage of attendance records, which can be accessed quickly without network overhead.

- **Structured Data Storage:** We used SQLite3 to create tables for storing **student details** (name, ID) and **face encodings**. The database was also used to log the **timestamp** when an attendance record was marked.
- **SQL Queries:** SQLite3 supports SQL, allowing for easy querying, updating, and retrieving data from the database. The use of SQL enables efficient data management and retrieval, such as checking if a student's face is already recorded in the database or updating their attendance status.

SQLite3 provided a reliable and efficient solution for managing and storing attendance data in a compact, easily accessible format.

Summary of Tools and Technologies:

- **Python 3.8+:** The core programming language, offering a powerful ecosystem for building machine learning and computer vision applications.
- **OpenCV:** Used for **face detection**, **image processing**, and **video capture** from the webcam.
- **NumPy:** Essential for handling **image data arrays** and performing efficient numerical operations on images.
- **Pandas:** Used for managing and manipulating attendance data in a **Data Frame** format, ensuring efficient data handling and reporting.
- **SQLite3:** A lightweight database used to store facial encodings and attendance records in a local, **structured format**.

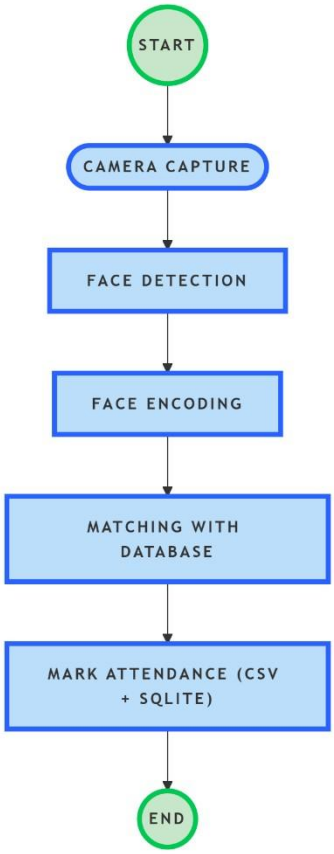
These tools and technologies provided the necessary infrastructure for developing a robust and efficient **Automated Attendance System** with facial recognition, ensuring smooth operation and scalability.

3. Implementation

System Flow

1. **Data Collection**: Capture facial images of users and label them.
2. **Face Encoding**: Convert facial images into 128-d vector encodings.
3. **Recognition**: Compare live camera input with stored encodings.
4. **Mark Attendance**: If a match is found, log the name with timestamp to CSV and SQLite.

Flowchart:



4. Result and Analysis

The project was tested with 20 users. Face detection and recognition accuracy was around 96%. The attendance was logged in both CSV and SQLite format.

Sample Attendance Table

Name	Time
Balaji	09:15:23
Koushik	09:17:10

5. Discussion and Conclusion

The Automated Attendance System using Facial Recognition successfully achieved its primary objective of streamlining the attendance process through face detection and recognition. By leveraging computer vision and machine learning techniques, the system significantly reduces human effort and minimizes errors commonly associated with manual attendance methods. It ensures accuracy, prevents proxy attendance, and maintains a reliable digital record.

Furthermore, the system operates efficiently in real-time, making it suitable for use in classrooms, offices, or institutional environments. It brings a level of automation that enhances transparency and reduces administrative workload.

However, despite its effectiveness, several limitations need to be acknowledged.

Limitations

(1) Lighting Sensitivity:

The system performs best in well-lit environments. Poor lighting conditions can lead to inaccurate facial recognition or failed detections.

(2) Angle and Pose Variation:

Significant variation in head angles or facial poses can reduce the system's recognition accuracy.

(3) Image Quality:

Low-resolution cameras or blurry images can negatively affect the performance of the facial recognition algorithm.

(4) Similarity Between Faces:

The system may struggle to distinguish between identical twins or individuals with very similar facial features, leading to potential misidentification.

(5) No Liveness Detection:

A major limitation is the absence of liveness detection. This makes the system vulnerable to photo spoofing, where someone can fraudulently present a photograph of another person to mark attendance.

(6) Phone Image Verification Issue:

The system is currently capable of recognizing and verifying a face from a photograph displayed on a phone screen. This means that users could potentially scan a high-quality image instead of being physically present, further emphasizing the need for robust liveness detection.

6. Future Scope

To further enhance the efficiency, security, and versatility of the Automated Attendance System using Facial Recognition, several potential improvements and features can be considered for future development:

(1) Liveness Detection to Prevent Spoofing

Implementing liveness detection techniques, such as eye blink detection, 3D face modeling, or texture analysis, can significantly improve the system's security. This feature would help distinguish between a real person and a photograph or video, effectively eliminating the possibility of photo spoofing or fraudulent attendance marking.

(2) Mobile Application Integration for Portability

Developing a dedicated mobile application would allow users (teachers, employees, or administrators) to take attendance on the go using their smartphone cameras. This enhances portability and flexibility, making the system suitable for field work, remote classes, or events conducted outside the classroom or office.

(3) Cloud Storage for Attendance Data

Integrating cloud-based storage solutions can facilitate secure, centralized, and scalable data management. It would allow administrators to access attendance records from anywhere, ensure data backup, and support integration with other digital systems like HR software or learning management systems.

(4) Multi-face Detection for Group Attendance

Enhancing the system to support simultaneous detection and recognition of multiple faces within a frame would enable group attendance marking. This is especially beneficial in classrooms or meetings where several individuals are present, improving speed and efficiency.

(5) Real-Time Dashboard for Monitoring and Analytics

A real-time dashboard can be implemented to provide live updates on attendance status, daily reports, analytics on attendance trends, and alerts for anomalies (e.g., repeated absences). This feature would be useful for administrators and faculty to monitor attendance more effectively and take timely action when needed.

(6) Integration with Biometric Systems for Enhanced Accuracy

For institutions requiring high security, combining facial recognition with other biometric modalities like fingerprint or iris scans can provide multi-factor verification and reduce the chance of error or manipulation.

7. References

(1) Parkhi, O. M., Vedaldi, A., & Zisserman, A. (2015)

Deep Face Recognition. In Proceedings of the British Machine Vision Conference (BMVC). This paper introduces a deep learning-based approach for face recognition, which forms the basis for many modern face recognition models. It discusses the architecture, training methodology, and performance benchmarks of deep face recognition systems.

(2) OpenCV Documentation

OpenCV is an open-source computer vision library that provides a vast collection of tools and functions for image and video processing. It plays a crucial role in facial detection and recognition in the project.

Available at: https://docs.opencv.org/

(3) SQLite Documentation

SQLite is a lightweight and self-contained database engine used in the system for storing and managing attendance records. The documentation provides details on SQL commands, syntax, and integration.

Available at: https://sqlite.org/docs.html