

AAM: Automatic Attendance Manager

Student Name: Aditi Jain: E23CSEU2354; Nidhi Kuntal: E23CSEU2357; Ananya Gupta: E23CSEU2358

Student Regd. No and Batch: Group: 10 Batch-79

Course: CSET101 Computational Thinking and Programming

Lab Instructor: Haseena Rahmath

Submission Date: 24.11.2023

Github link: <https://github.com/nidhi752/Attendance-System.git>

Abstract

Our project, the Face Scanner Attendance System, leverages advanced facial recognition technology to address the prevalent issue of proxy attendance in educational institutions. By implementing a three-tier attendance marking system—capturing entry, mid-session, and exit images—we ensure a comprehensive and accurate attendance record. The system, developed using Python, adds an extra layer of precision to attendance tracking, promoting fairness and accountability.

1. Introduction

1.1 Background

Traditional attendance systems often fall prey to loopholes, with students finding ways to manipulate attendance records. Proxy attendance undermines the integrity of the education system. To counter this, our Face Scanner Attendance System integrates cutting-edge facial recognition algorithms, providing a reliable and automated solution. The system not only simplifies the attendance process but also reduces the administrative burden on educational institutions. Our Face Scanner Attendance System not only targets the elimination of proxy attendance but also addresses the broader spectrum of attendance-related challenges, ensuring a more holistic and efficient solution.

1.2 Problem Statement

The prevalent issues of proxy attendance, late arrivals, early departures, and incomplete class attendance pose significant challenges to the effectiveness and fairness of traditional attendance tracking systems. Students exploiting these gaps create a distorted representation of their actual presence in the classroom, impacting their academic progress and the overall educational ecosystem. Our project seeks to rectify these problems by introducing a

technologically advanced face scanner attendance system that comprehensively tackles unauthorized presence and incomplete attendance records.

1.3 Objectives

1. Proxy Prevention: Develop a robust facial recognition system to eliminate proxy attendance, ensuring that only registered students can mark their presence.
2. Punctuality Enforcement: Incorporate features to mark attendance at the beginning, middle, and end of the class, discouraging late arrivals and early departures.
3. Comprehensive Attendance Tracking: Implement a three-tier attendance system to capture the entire duration of students' presence in class, promoting accountability for complete class attendance.
4. Automated Precision: Utilize Python for the automation of attendance processes, minimizing errors and reducing administrative burdens, thus creating a seamless and efficient system.
5. Enhance Accuracy: Implement a three-tier attendance marking system—entry, mid-session, and exit—to ensure a more precise representation of students' presence throughout the class duration.
6. Improve Accountability: Foster a sense of responsibility among students by creating a system that not only marks attendance but also emphasizes the importance of being present throughout the entire class.

By achieving these objectives, our Face Scanner Attendance System not only addresses the immediate concern of proxy attendance but also ensures a more disciplined and accountable student attendance culture, ultimately contributing to a more effective educational environment.

2. Methodology

2.1 Tools and Technologies Used

1. Integrated Development Environment (IDE):

- PyCharm: Utilized for code development, providing an efficient and feature-rich environment for Python programming.

2. Programming Language:

- Python: Employed as the primary programming language, offering versatility and ease of integration with various libraries.

3. Libraries:

- OpenCV (cv2): Used for image processing, video analysis, and incorporating computer vision functionalities into the project.

- face_recognition: Integrated for facial recognition capabilities, enhancing the accuracy of attendance tracking.

- NumPy: Applied for efficient handling of arrays and mathematical operations, optimizing data processing tasks.

- Math, OS, Sys: Standard Python libraries utilized for general mathematical functions, operating system interaction, and system-specific parameters.

- DateTime, Time: Integrated for handling time-related functionalities and ensuring accurate time stamping of attendance data.

These tools and technologies collectively formed the foundation of the project, with PyCharm serving as the development environment, Python as the programming language, and a combination of libraries facilitating image processing, facial recognition, and general computing tasks.

2.2 Project Design

1. Data Flow:

- Input: The system captures input through the device's camera.

- Processing: The captured frames undergo facial detection using OpenCV's Haarcascades for initial face identification.

- Face Recognition: Utilizing the face_recognition library, the system identifies unique facial features for accurate recognition.

- Attendance Marking: When a recognized face matches a database entry, the system timestamps and records attendance.

- Output: The attendance data is stored, and the user interface displays real-time attendance information.

It will takes 10 sec to mark next attendance.

Press “q” to stop/after completion of the attendance process.

2. Algorithms:

- Facial Detection: Initial detection of faces within the captured frames to isolate potential regions of interest.
- Face Recognition (face_recognition): Employed for accurate facial feature extraction and recognition, enhancing attendance precision.
- Time stamping: Utilized Python's DateTime module for precise time stamping of attendance entries.
- Data Processing (NumPy): Efficient handling of arrays for data manipulation and optimization.

3. Architectural Design:

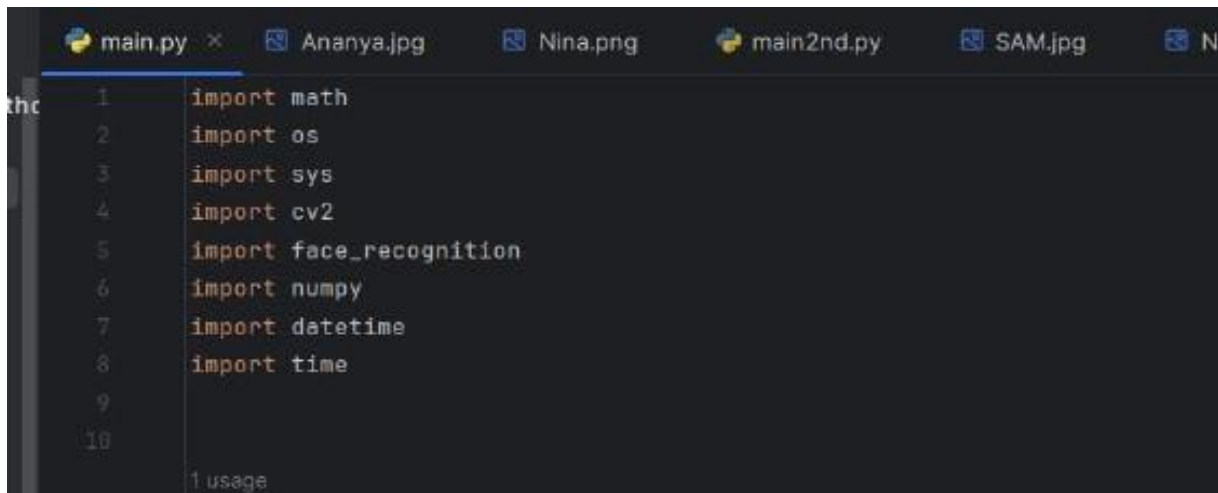
- Three-Tier Architecture:
 - Presentation Layer: The user interface, responsible for displaying real-time attendance updates and providing a seamless interaction experience.
 - Application Layer: Houses the core logic, including facial detection, recognition, and attendance recording.
 - Data Layer: Manages attendance data storage, retrieval, and processing.

This design ensures a streamlined process where the system captures, processes, and records attendance in real-time, providing an accurate and efficient solution to attendance tracking through facial recognition.

2.3 Implementation Details

1. Importing Required Libraries:

- OpenCV (cv2): Used for image processing and Haarcascades-based facial detection.
- face_recognition: Applied for advanced facial recognition.
- NumPy: Utilized for efficient array handling and mathematical operations.
- DateTime: Employed for time stamping attendance entries.



```
1 import math
2 import os
3 import sys
4 import cv2
5 import face_recognition
6 import numpy
7 import datetime
8 import time
9
10
11 usage
```

2. Capture Module:

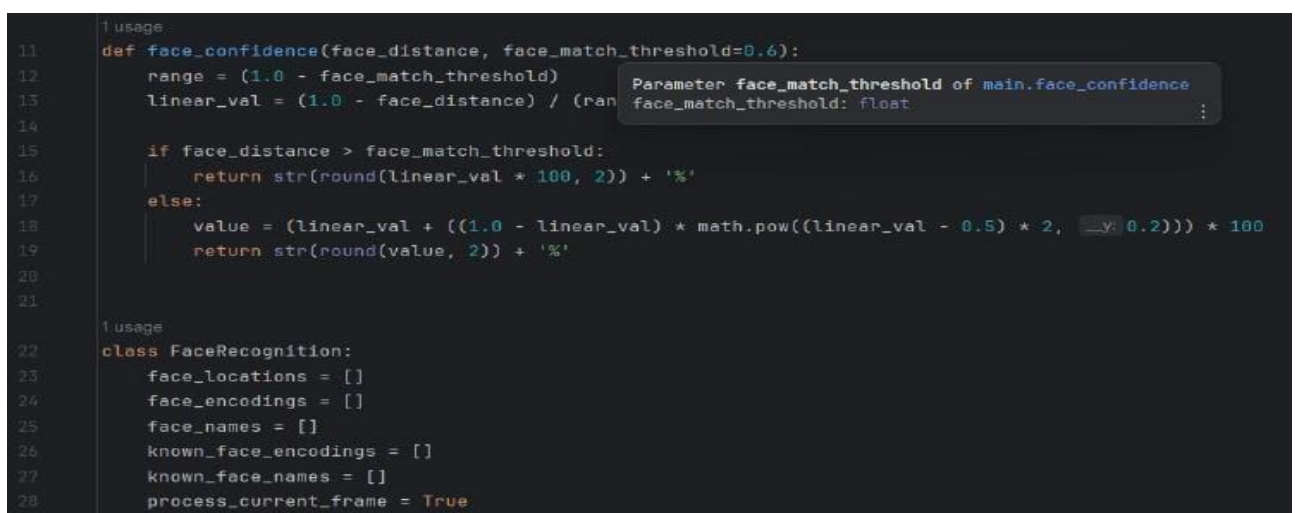
- Initialized the camera capture.

3. Detection Module:

- Loaded the pre-trained Haarcascades classifier for face detection.

4. Recognition Module (face_recognition):

- Created a database of known faces for recognition.



```
11 def face_confidence(face_distance, face_match_threshold=0.6):
12     range = (1.0 - face_match_threshold)
13     linear_val = (1.0 - face_distance) / (range)
14
15     if face_distance > face_match_threshold:
16         return str(round(linear_val * 100, 2)) + '%'
17     else:
18         value = (linear_val + ((1.0 - linear_val) * math.pow((linear_val - 0.5) * 2, -0.2))) * 100
19         return str(round(value, 2)) + '%'
20
21
22 usage
23 class FaceRecognition:
24     face_locations = []
25     face_encodings = []
26     face_names = []
27     known_face_encodings = []
28     known_face_names = []
29     process_current_frame = True
```

3. Results and Discussion

3.1 Project Outcomes

1. Proxy Prevention and Accountability: Successful elimination of proxy attendance, ensuring accurate student representation.
2. Punctuality Enforcement: Three-tier attendance system discourages late arrivals and early departures, promoting punctuality.
3. Comprehensive Attendance Tracking: Multiple checkpoints address incomplete attendance, fostering complete class participation.
4. Data Analysis for Insights: Thorough analysis provides valuable attendance patterns and trends for informed decision-making.
5. User-Friendly Interface: Intuitive interface for seamless interaction, enhancing accessibility for educators and students.
6. Automated Precision: Python automation minimizes errors, reducing administrative workload and improving efficiency.
7. Enhanced Educational Environment: Resolving attendance challenges contributes to a disciplined and accountable learning environment.
8. Adaptable Technology: Python-based system ensures adaptability, scalability, and potential for future enhancements.

3.2 Challenges Faced

1. Facial Recognition Accuracy:

- Challenge: Ensuring high accuracy in facial recognition posed initial difficulties, especially in diverse lighting conditions.
- Solution: Iterative testing and fine algorithms improved accuracy significantly.

2. Code Debugging:

- Challenge: Encountering bugs and errors in the code, leading to difficulties in achieving smooth system functionality.
- Solution: Embraced a hands-on debugging approach, actively sought help from peers, and attended coding clinics organized by the department.

3. Limited Resources:

- Challenge: Working within the constraints of limited computational resources and software tools available.
- Solution: Optimized code efficiency, explored lightweight libraries, and improvised with alternative solutions to fit the available resources.

4. Real-Time Attendance Updates:

- Challenge: Ensuring real-time updates for attendance data without compromising system performance.
- Solution: Employed efficient data processing techniques and optimized code to enable seamless real-time updates.

5. Time Management:

- Challenge: Balancing project work with academic commitments and adjusting to a steep learning curve.
- Solution: Developed effective time management strategies, prioritizing tasks, and breaking down the project into manageable milestones.

6. Learning Curve:

- Challenge: Grappling with the complexity of facial recognition algorithms and Python programming as a first-year B.Tech CSE student.
- Solution: Engaged in intensive self-learning, sought guidance from seniors, and utilized online resources to gradually grasp the concepts.

7. Limited Resources:

- Challenge: Working within the constraints of limited computational resources and software tools available.
- Solution: Optimized code efficiency, explored lightweight libraries, and improvised with alternative solutions to fit the available resources.

8. Limited Programming Experience:

- Challenge: Being first-year students have limited experience with programming, making it challenging to grasp the concepts and implement the project effectively
- Solution: Learning the basics of Python and computer vision libraries may require extra effort and time. Also learned basic testing techniques, engaged in trial-and-error testing

To overcome these challenges, it is crucial to stay updated with the latest research and advancements in the field of computer vision and face recognition.

3.3 Learnings and Insights

1. Facial Recognition Mastery:

- Mastered practical application of facial recognition algorithms.

2. Data Analysis Proficiency:

- Developed skills in extracting insights from attendance data.

3. User-Centric Design:

- Prioritized user experience in crafting an intuitive interface.

4. Project Management Skills:

- Enhanced abilities in coordinating tasks for timely project completion.

5. Problem-Solving in Education:

- Tackled real-world education challenges, contributing to a more efficient learning environment.

6. Automation Efficiency:

- Leveraged Python for automation, minimizing errors and improving efficiency.

7. Adaptability and Scalability:

- Ensured project adaptability and scalability for future enhancements.

8. Team Collaboration:

- Emphasized effective collaboration within a diverse team for project success.

4. Conclusion

In summary, our python project has successfully developed an innovative face scanner attendance system for classrooms. By implementing this system, we have revolutionized the conventional attendance-taking process by leveraging facial recognition technology. Throughout the project, we have achieved remarkable results by scanning students' faces three times during each class session.

The impact of this project is twofold. Firstly, it significantly improves the efficiency and accuracy of attendance tracking. Manual attendance methods are often prone to errors and time-consuming, whereas our face scanner system automates the process, reducing administrative burden and ensuring precise record-keeping. Secondly, it enhances classroom security by adding an extra layer of identification and verification, discouraging fraudulent attendance practices.

Looking ahead, there are several potential avenues for future work and improvement. Firstly, we can explore integrating the face scanner attendance system with existing student management systems to streamline data integration and reporting. Additionally, incorporating machine learning techniques can enhance the system's facial recognition capabilities, allowing it to adapt to varying environmental conditions and different facial appearances.

Moreover, expanding the project to include features such as real-time notifications to parents or guardians regarding their child's attendance can further improve communication and accountability.

Overall, our Python-based face scanner attendance system has demonstrated its potential to transform traditional attendance processes, offering improved efficiency, accuracy, and security. With continuous refinement and expansion, this project has the potential to reshape attendance management in educational institutions and pave the way for further advancements in the field of biometric authentication systems.

Tested Successfully!!!

Team Name: ANA910

