A Mini Project report on

**SMART ATTENDANCE SYSTEM**

*Submitted in Partial fulfilment of the requirements for the award of the Degree of*

**Bachelor of Technology**

**In**

**Artificial Intelligence & Machine Learning**

Submitted by

**B Vardhan Kumar - 21EG107A05**

**A Goutham Sai - 21EG107A04**

**K Sri Ankith – 21EG107A23**

Under the Guidance of

**Dr. M Srinadh Swamy**

Assistant Professor



**Department of Artificial Intelligence**

**School of Engineering**

**ANURAG UNIVERSITY**

**2024-2025**

****

### **CERTIFICATE**

This is to certify that the project report titled **“Smart Attendance System”** is being submitted by B Vardhan Kumar (21EG107A05)**,** A Goutham Sai (21EG107A04)**,** K Sri Ankith (21EG107A23)**,** in IV B. Tech I semester *Artificial Intelligence and Machine Learning* is a record bonafide work carried out by them. The results embodied in this report have not been submitted to any other University for the award of any degree.

Student’s Name Student’s Signature

1. B Vardhan Kumar 1.

2. A Goutham Sai 2.

3. K Sri Ankith 3.

**Dr M Srinadh Swamy Dr A Mallikarjun Reddy**

**Assistant Professor Head of Department**

**EXTERNAL EXAMINER**

**Acknowledgement**

We are indebted to our project guide Dr. M Srinadh Swamy – Assistant Professor, Department of Artificial Intelligence, Anurag University. We feel it’s a pleasure to be indebted to our guide for his valuable support, advice, and encouragement and we thank him for his superb and constant guidance towards this project.

We sincerely thank Dr. A. Mallikarjuna Reddy, Associate Professor and the Head of the Department of Artificial Intelligence, Anurag University, for all the facilities provided to us in the pursuit of this project.

We wish to record our profound gratitude Dr. V. Vijay Kumar, Dean – School of Engineering, for his motivation and encouragement.

We acknowledge our deep sense of gratitude to Prof. Balaji Utla, Registrar, Anurag University, for being a constant source of inspiration and motivation.

We owe our gratitude to Prof. Archana Mantri, Vice-Chancellor, Anurag University, for extending the University facilities to the successful pursuit of our project so far and her kind patronage.

**CONTENTS**

**ABSTRACT i**

**LIST OF FIGURES ii**

1. **INTRODUCTION 1**
2. **LITERATURE SURVEY 2-3** 
   1. Existing System 2
   2. Limitation of Existing system 2
   3. Gaps Identified 2
   4. Problem Statement 3
   5. Objectives 3

1. **PROPOSED SYSTEM 4-5** 
   1. Architecture/Algorithms/Methods 4
   2. Requirements & Specifications 5
      1. Client requirements 5
      2. Software requirements 5
      3. Hardware requirements 5
2. **DESIGN 6-8** 
   1. DFD / ER / UML diagram (any other project diagrams) 6
   2. Module design and organization 7
3. **IMPLEMENTATION & TESTING 9-12** 
   1. Technology Used 9
   2. Procedures 9
   3. Testing & Validation 10
      1. Design test cases and scenarios 10
      2. Validation 10
4. **RESULTS 13-16** 
   1. Output 13
   2. Result Analysis 15

**CONCLUSION 17**

**FUTURE WORK 18**

**REFERENCES 20**

**ABSTRACT**

"Smart Attendance System" utilizes facial recognition technology to streamline the attendance marking process. Traditional methods of attendance tracking, such as manual sign-ins or RFID cards, are time-consuming and susceptible to errors. In contrast, the proposed system allows for a more efficient and secure approach by identifying individuals through their facial features. Users are required to interact with login and logout buttons to mark their presence and absence, making the system simple to use while reducing the possibility of errors associated with manual entry.

The facial recognition functionality is implemented using Python libraries, including the "ageitgey/face\_recognition" repository for face detection and recognition. To enhance security, the system integrates anti-spoofing techniques from the "Silent-Face-Anti-Spoofing" repository, ensuring that only legitimate users can mark their attendance. This document outlines the development process, system architecture, and performance evaluation, demonstrating the potential of facial recognition technology in enhancing attendance management systems.

**LIST OF FIGURES**

Fig. 4.1: Data Flow Diagram of Smart Attendance System 6

Fig. 6.1: Real-Time Attendance Logging Interface 13

Fig. 6.2: User Registration Interface 13

Fig. 6.3: User Registration Interface 14

Fig. 6.4: Spoofing Detection Interface 14

Fig. 6.5: User Not Recognized 15

**CHAPTER-1**

**INTRODUCTION**

Attendance management is a critical aspect of administrative operations in educational institutions, corporate environments, and other organizations. Traditional methods such as manual sign-ins, punch cards, and RFID-based systems often face challenges such as inefficiency, time consumption, and susceptibility to manipulation. These systems can also lead to human errors or intentional misuse, affecting the accuracy and reliability of attendance records.

To address these issues, biometric systems—specifically facial recognition—have emerged as a promising solution. Facial recognition technology offers a contactless, efficient, and secure way to automate the process of identifying individuals. It eliminates the need for physical interaction, reduces the risk of errors, and enhances the overall user experience.

This project focuses on developing a Smart Attendance System that leverages facial recognition technology to streamline attendance tracking. The system allows users to log their presence and absence by using facial recognition along with a simple login and logout interface. By integrating advanced facial recognition algorithms and anti-spoofing mechanisms, the system ensures that only authorized individuals can mark attendance, minimizing fraud and improving accuracy.

Python libraries, such as the "ageitgey/face\_recognition" for face detection and recognition, and the "Silent-Face-Anti-Spoofing" for security against spoofing attacks, form the core of the system's functionality. This documentation covers the design, development, and testing of the system, highlighting its potential benefits in replacing traditional attendance methods with a more secure and efficient alternative.

**CHAPTER -2**

**LITERATURE SURVEY**

**2.1 Existing System**

The current attendance systems implemented in educational and corporate environments predominantly rely on manual or semi-automated processes. Traditional methods like sign-in sheets, punch cards, or RFID systems are still widely used, though they come with several limitations, such as time consumption and susceptibility to human errors. Some organizations have adopted biometric systems, such as fingerprint or iris recognition, to automate attendance tracking. However, these systems require physical interaction, which can lead to inefficiencies during high-traffic times or when there are hardware issues (e.g., dirty sensors, misplaced cards). Additionally, the manual process is time-consuming, taking several minutes out of each session or workday.

**2.2 Limitation of Existing System**

* **Manual Dependency:** Traditional methods require manual input from users, which increases the chance of human error.
* **Time-Consuming:** Marking attendance manually or using RFID cards takes time, particularly in large groups.
* **Proxy Attendance:** In many systems, there is a high risk of proxy attendance, where someone signs in for someone else.
* **Physical Contact:** Biometric systems like fingerprint scanners need physical interaction, which can be inconvenient or unhygienic.
* **Inflexibility in Poor Conditions:** Many biometric systems perform poorly in conditions such as low light or with compromised hardware

**2.3 Gaps Identified**

Existing systems, while useful to an extent, are limited by their dependence on manual intervention or physical contact. These systems are prone to:

* **Proxy Attendance:** As there are minimal security measures to ensure only the rightful individual logs in, these systems are often abused.
* **Lack of Automation:** Most systems do not automatically mark attendance, requiring users to engage with the system manually, which is inefficient.
* **Limited Anti-Spoofing:** Systems that incorporate biometrics often do not include anti-spoofing measures, leaving them vulnerable to fake entries via printed images or other means.
* **Lighting and Camera Constraints:** Systems relying on video or image processing are significantly hindered by poor lighting or suboptimal camera angles.

**2.4 Problem Statement**

The traditional methods of attendance marking, including manual systems and basic biometric tools, are inefficient and prone to misuse. These systems also require physical interaction, which can be cumbersome in high-traffic environments. Additionally, most solutions do not incorporate advanced security measures, such as anti-spoofing, to prevent fraudulent entries. Therefore, a need exists for an automated, secure, and efficient attendance system that minimizes manual input, eliminates proxy attendance, and adapts to varying environmental conditions.

**2.5 Objectives**

The objectives of this project are:

1. **Automation:** To develop a facial recognition-based attendance system that automates attendance marking, eliminating the need for manual input.
2. **Security:** To incorporate anti-spoofing measures that ensure only authorized individuals can mark their attendance.
3. **Efficiency:** To create a system that is user-friendly, reducing time spent on attendance procedures, and overcoming the limitations of existing methods.
4. **Real-Time Processing:** To allow real-time face detection and recognition, ensuring that attendance is recorded as soon as a person is recognized

**CHAPTER -3**

**PROPOSED SYSTEM**

**3.1 Architecture/Algorithms/Methods**

The architecture of the Smart Attendance System is designed to capture, recognize, and record attendance using the following key components:

* **Face Detection:** The system utilizes the **OpenCV** library to capture live video feed and detect faces. The **Histogram of Oriented Gradients (HOG)** method is employed to detect facial features in the captured frames.
* **Face Recognition:** Once a face is detected, the system uses the **FaceNet** algorithm to extract facial embeddings, creating a unique digital signature for each individual. These embeddings are compared with the stored database using the **Euclidean Distance** metric to determine whether a match exists.
* **Anti-Spoofing Measures:** The system implements anti-spoofing techniques, such as **Fourier Spectrum Analysis**, to ensure that only live, real faces are recognized, thus preventing the system from being fooled by photos or videos.
* **Attendance Marking:** If the system identifies a valid match, the attendance is automatically logged along with the timestamp. Depending on the system setup, the attendance may also differentiate between "check-in" and "check-out" based on time intervals​

The overall flow of the system architecture can be summarized as follows:

1. Webcam captures live video feed.
2. Face detection using OpenCV and HOG.
3. Face recognition using FaceNet embeddings.
4. Anti-spoofing validation using Fourier Spectrum Analysis.
5. Automatic attendance marking based on face recognition.

**3.2 Requirements & Specifications**

**3.2.1 Client Requirements**  
The system is designed to be deployed in educational institutions or corporate environments where attendance tracking is critical. The primary client requirements are:

* An automated system that replaces manual attendance marking.
* High accuracy in recognizing faces with minimal false positives or negatives.
* The ability to store and retrieve attendance records efficiently.
* Integration with existing database systems for easy report generation.
* A user-friendly interface for administrators to manage attendance data**.**
  + 1. **Software Requirements**

The software stack required for implementing this Smart Attendance System includes:

* **Python 3.x:** The core programming language used for developing the system.
* **OpenCV:** Library for real-time image capture and face detection.
* **face\_recognition Library:** A pre-trained library for facial recognition based on FaceNet.
* **Tkinter:** For building the graphical user interface (GUI).
* **Pillow:** For image processing tasks like resizing and saving.
* **SQLite/MySQL:** A database to store and manage attendance records.
* **pickle:** For serializing and saving the face embeddings and other data

**3.2.3 Hardware Requirements**  
The system requires the following hardware components:

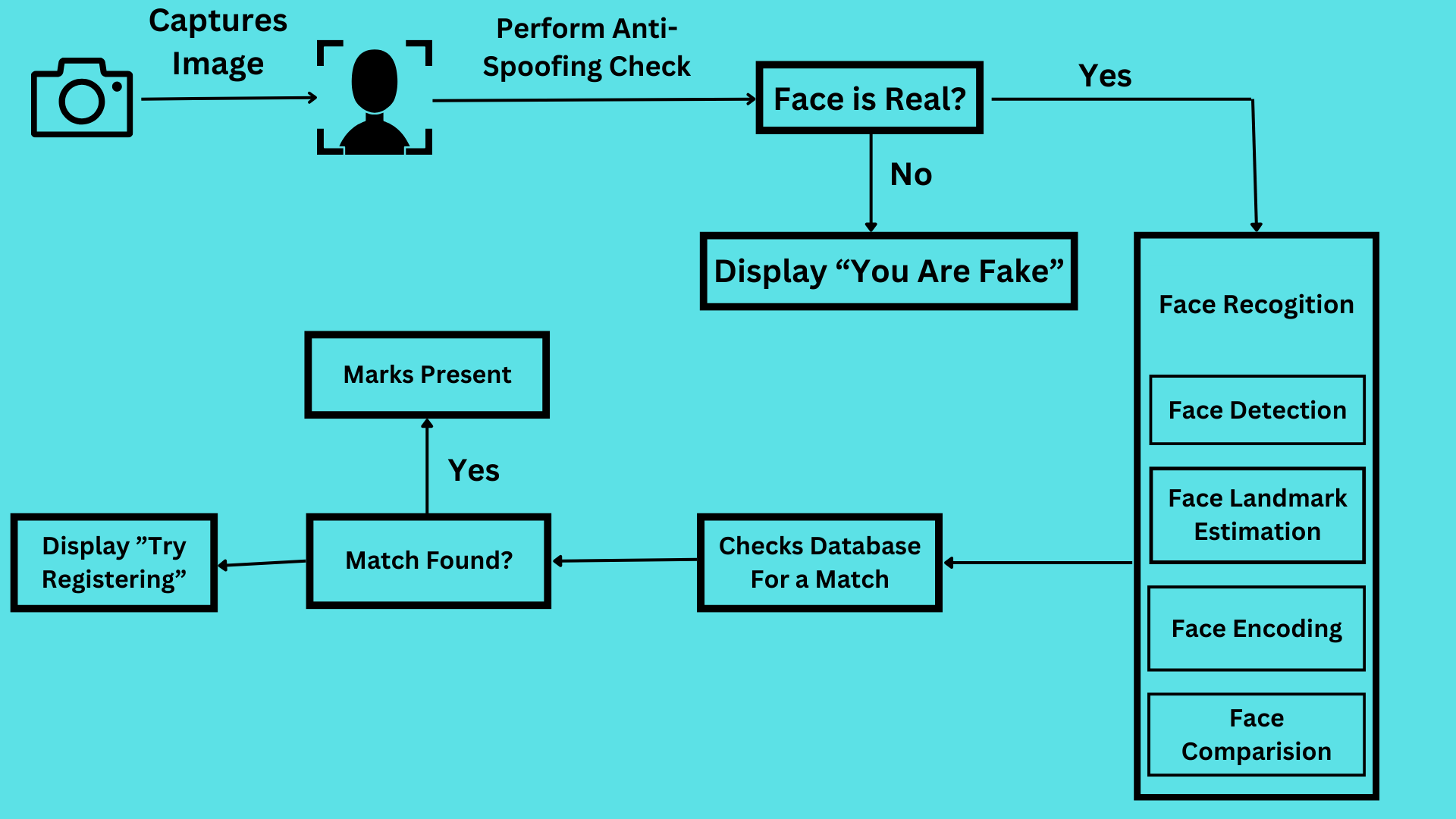
* **Webcam or IP Camera:** To capture real-time video feed for face detection.
* **Computer/Server:** A computer with at least 4GB of RAM and sufficient storage space to store attendance logs and face embeddings.
* **Hard Drive:** Adequate space to store images and attendance data.

**CHAPTER -4**

**DESIGN**

**4.1 Data Flow Diagram (DFD)**

The system first captures an image and performs an anti-spoofing check to determine if the face is real. If the face is fake, a message "You Are Fake" is displayed. If the face is real, it moves to face recognition, which involves face detection, landmark estimation, encoding, and comparison. The system then checks the database for a match. If no match is found, a message "Try Registering" is displayed. If a match is found, attendance is marked.

****

***Fig-4.1 Data flow diagram of smart attendance system***

**4.2 Module Design and Organization**

The Smart Attendance System is divided into several key modules, each responsible for handling different functionalities of the system.

1. **Face Detection Module:**
   * + **Functionality:** Captures live video feed using a webcam and detects faces using the OpenCV library. This module extracts facial features in real-time using the HOG (Histogram of Oriented Gradients) algorithm.
     + **Inputs/Outputs:** Inputs live video; outputs detected faces.
2. **Face Recognition Module:**
   * + **Functionality:** Matches detected faces with stored face embeddings using the FaceNet algorithm. It converts faces into embeddings and uses Euclidean distance to check for matches.
     + **Inputs/Outputs:** Inputs detected faces; outputs recognized individuals or null if no match is found.
3. **Anti-Spoofing Module:**
   * + **Functionality:** Verifies the authenticity of the captured face using Fourier Spectrum Analysis to differentiate between live and spoofed faces (e.g., printed photos or videos).
     + **Inputs/Outputs:** Inputs detected faces; outputs a boolean (real/spoof).
4. **Attendance Logging Module:**
   * + **Functionality:** Logs attendance records (date, time, status) into the system database once a valid user is recognized.
     + **Inputs/Outputs:** Inputs recognized users; outputs attendance records stored in the database.
5. **User Registration Module:**
   * + **Functionality:** Allows the registration of new users by capturing their face data and storing the corresponding face embeddings.
     + **Inputs/Outputs:** Inputs new user details and face data; outputs a stored user profile.
6. **Attendance Report Generation Module:**
   * + **Functionality:** Provides a user-friendly interface for administrators to view and manage attendance records. Reports can be generated based on specific criteria (e.g., date range, user).
     + **Inputs/Outputs:** Inputs administrator queries; outputs attendance reports.

**CHAPTER -5**

**IMPLEMENTATION & TESTING**

**5.1 Technology Used**

The Smart Attendance System is built using various technologies that enable facial recognition, data processing, and user interaction. The following technologies are used:

* **Programming Language:** Python 3.x, chosen for its extensive libraries and ease of use in image processing.
* **Face Detection Library:** OpenCV, used for real-time image and video capture, along with face detection functionality.
* **Face Recognition Library:** face\_recognition (based on FaceNet), for accurate face recognition by converting facial features into embeddings and matching them with stored records.
* **Anti-Spoofing Library:** Fourier Spectrum Analysis, used to verify that the captured face is real and not a spoof.
* **GUI Framework:** Tkinter, for building the graphical user interface for administrators to interact with the system.
* **Database:** SQLite or MySQL, used for storing user profiles, face embeddings, and attendance records.
* **Libraries:**
  + - **Pillow:** For image processing.
    - **Pickle:** For serializing and deserializing face embeddings.

**5.2 Procedures**

The system implementation follows these key steps:

1. **Face Detection and Capture:**
   * + The webcam or camera captures a live video feed.
     + OpenCV detects faces in real time using the HOG algorithm.
2. **Face Recognition:**
   * + Detected faces are passed to the face\_recognition library, which uses the FaceNet algorithm to extract facial embeddings.
     + These embeddings are compared to stored user profiles using the Euclidean distance metric. If a match is found, the user’s attendance is recorded.
3. **Anti-Spoofing Mechanism:**
   * + The system employs Fourier Spectrum Analysis to ensure the detected face is live, preventing spoofing attempts with photos or videos.
4. **Attendance Marking:**
   * + Once a face is recognized, the system logs attendance automatically, marking the entry or exit time.
5. **Administrator Interface:**
   * + A Tkinter-based GUI allows administrators to view and manage attendance records, generate reports, and register new users.

**5.3 Testing & Validation**

**5.3.1 Design Test Cases and Scenarios**

Below are some example test cases designed to validate the system:

1. **Test Case 1:** Face Detection Accuracy
   1. **Scenario:** The system should correctly detect a face in various lighting conditions.
   2. **Expected Result:** The system should accurately detect faces under normal, low-light, and bright-light conditions.
2. **Test Case 2:** Face Recognition Accuracy
   1. **Scenario:** The system should correctly identify registered users and reject unregistered individuals.
   2. **Expected Result:** The system should accurately match faces with stored profiles and correctly mark attendance.
3. **Test Case 3:** Anti-Spoofing
   1. **Scenario:** The system should prevent spoofing attempts with printed photos or videos.
   2. **Expected Result:** The anti-spoofing mechanism should detect live faces only and reject static images.
4. **Test Case 4:** System Response Time
   1. **Scenario:** The system should process face detection and recognition quickly enough for real-time attendance marking.
   2. **Expected Result:** The system should detect, recognize, and log attendance within 2-3 seconds of capturing the face.
5. **Test Case 5:** Multiple Users in Frame
   1. **Scenario:** The system should handle multiple faces in the frame and mark attendance for all recognized users.
   2. **Expected Result:** All detected and recognized users should have their attendance marked.
6. **Test Case 6:** User Registration
   1. **Scenario:** The system should allow the registration of new users by capturing their facial features.
   2. **Expected Result:** The new user's face should be correctly stored, and their attendance should be marked in subsequent sessions.

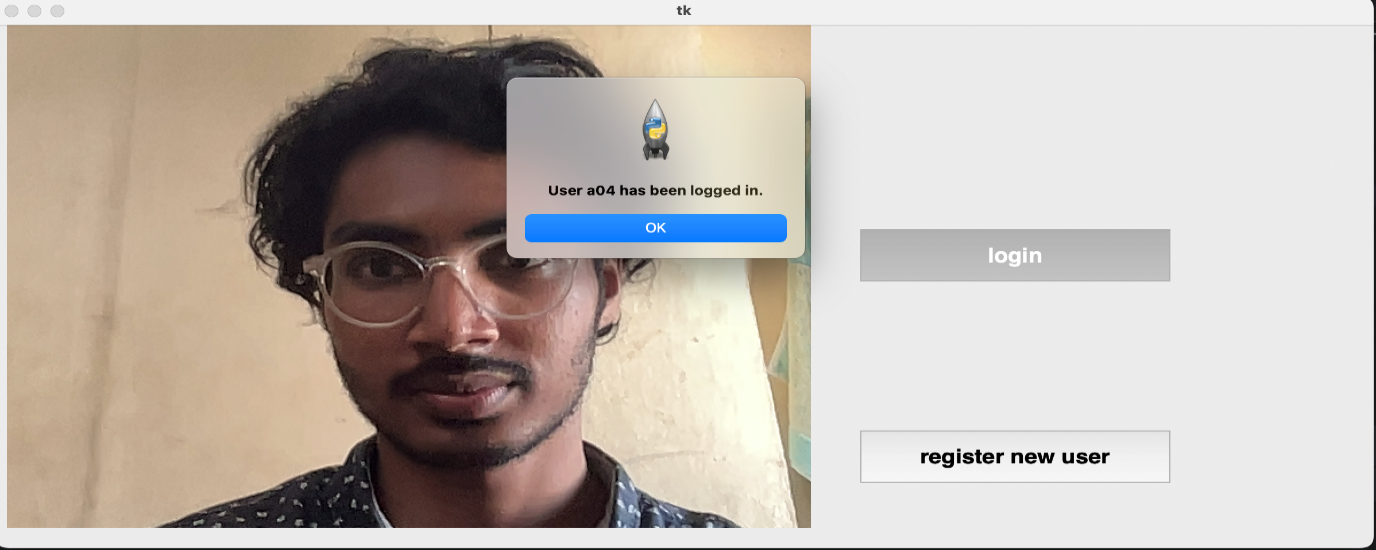
**5.3.2 Validation**

The validation process ensures the system meets the required specifications and objectives:

* **Performance Validation:** The system was tested under various conditions to ensure reliable performance in terms of face detection, recognition accuracy, and processing speed.
* **Security Validation:** Anti-spoofing tests were conducted to ensure the system could not be tricked by non-live faces.
* **Usability Validation:** The user interface and overall system workflow were validated by testing with potential users (e.g., administrators) to ensure ease of use and clarity in accessing attendance data.
* **Error Handling:** The system was tested for its ability to handle errors, such as camera disconnections, database issues, or unrecognized faces, ensuring robustness and stability in practical scenarios.

**CHAPTER -6**

**RESULTS**

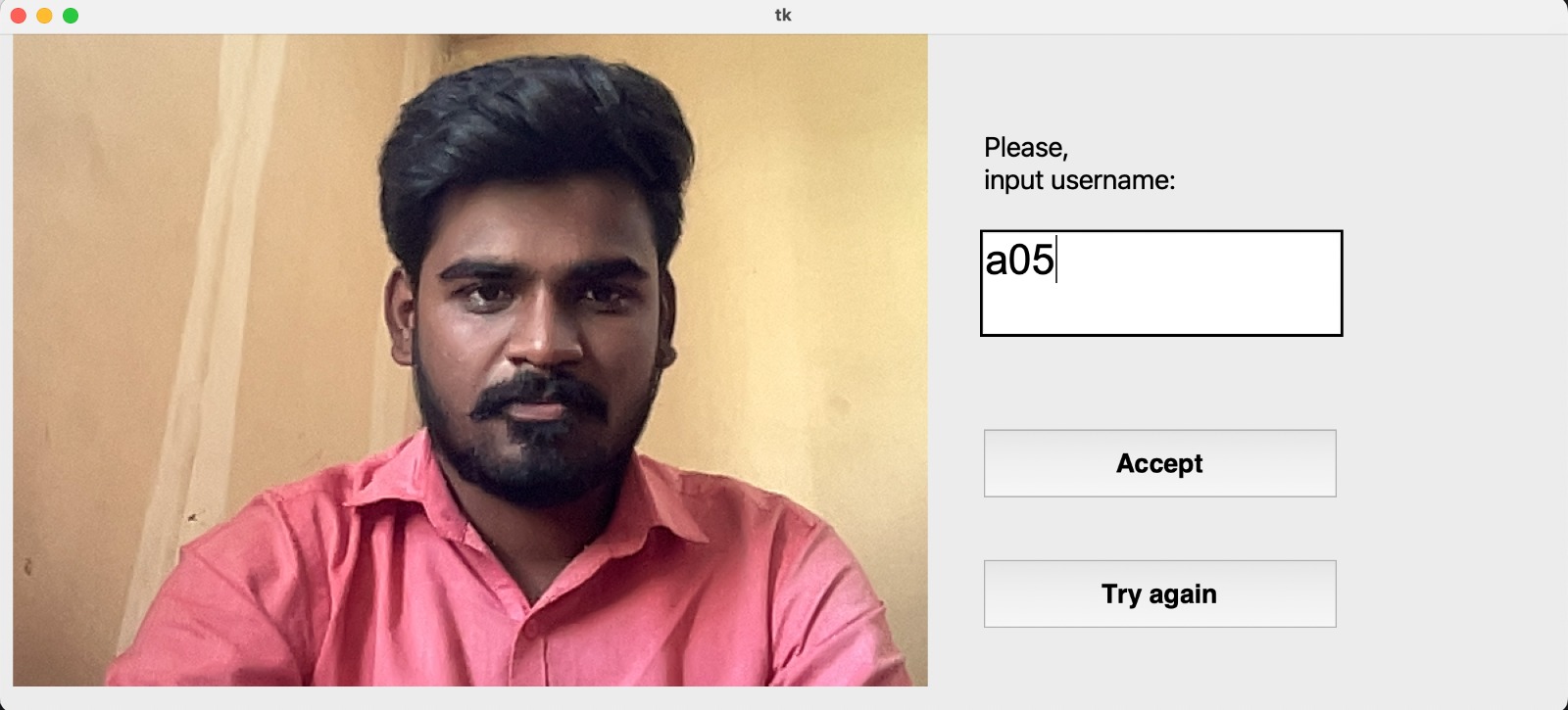
**6.1 Output**

The image shows a facial recognition-based login system where user "a04" has been successfully logged in.

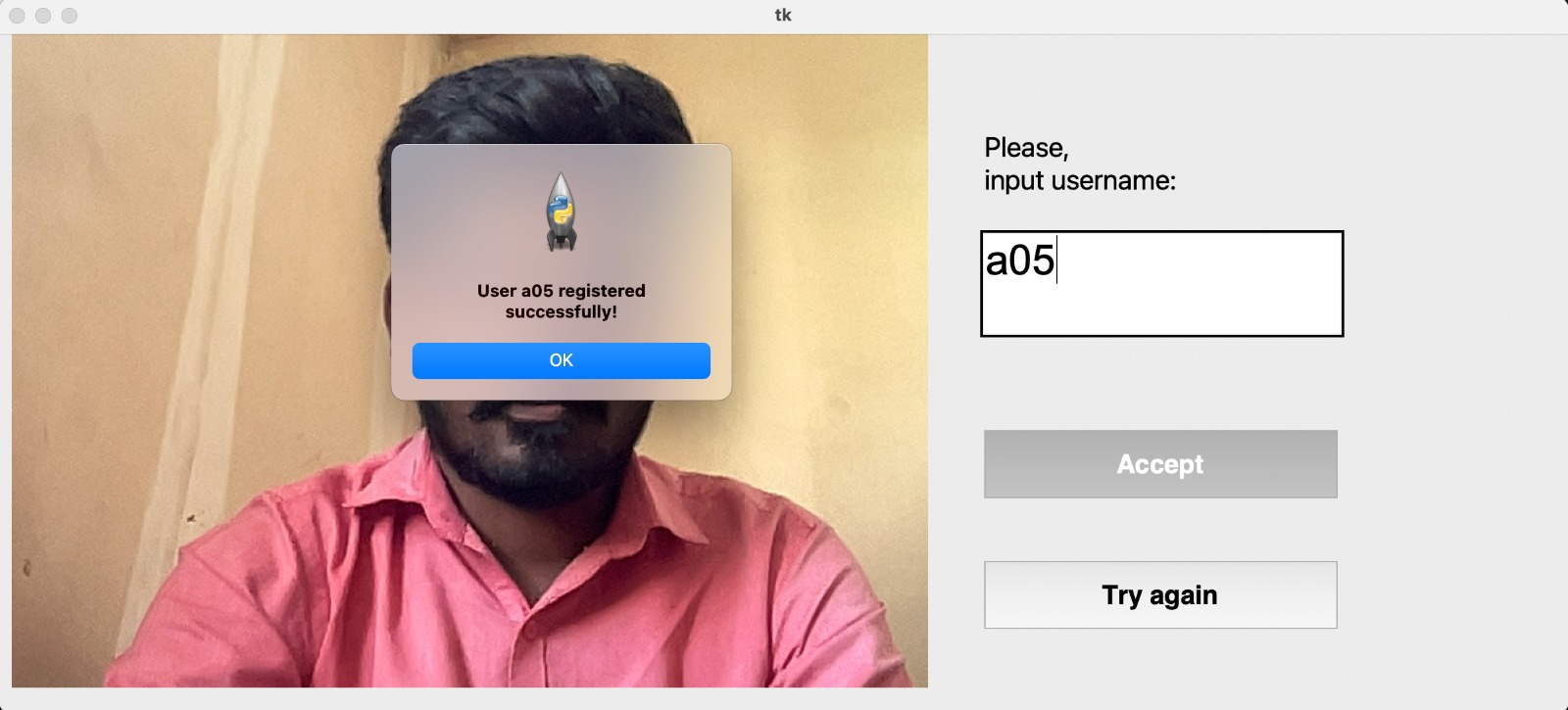
***Fig-6.1 Real Time Attendance Logging Interface***

The image shows a GUI interface prompting the user to input a username, with the username

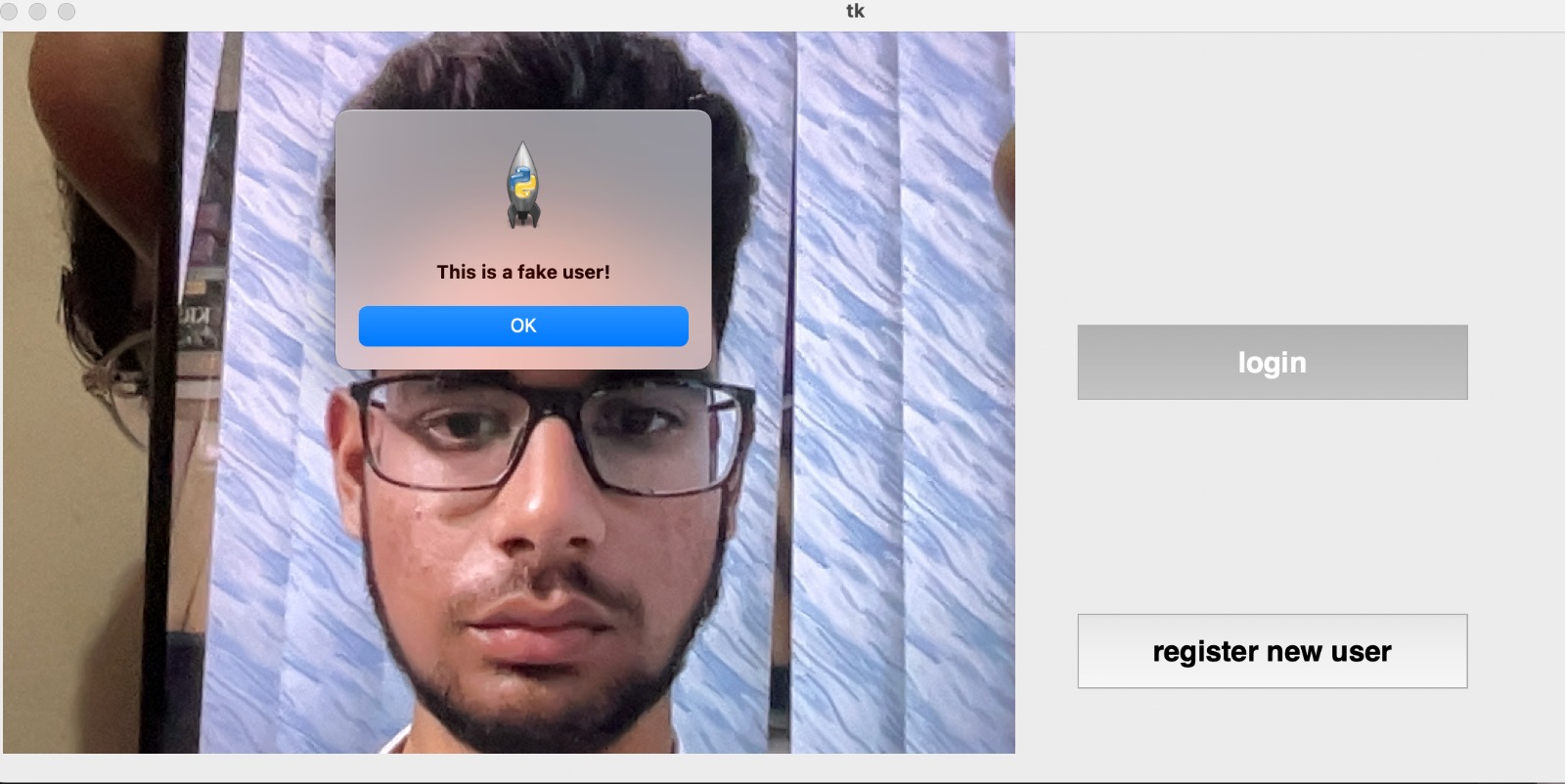
"a05"entered, and two buttons labelled "Accept" and "Try again" below the input field**.**

****

***Fig-6.2 User Registration Interface***

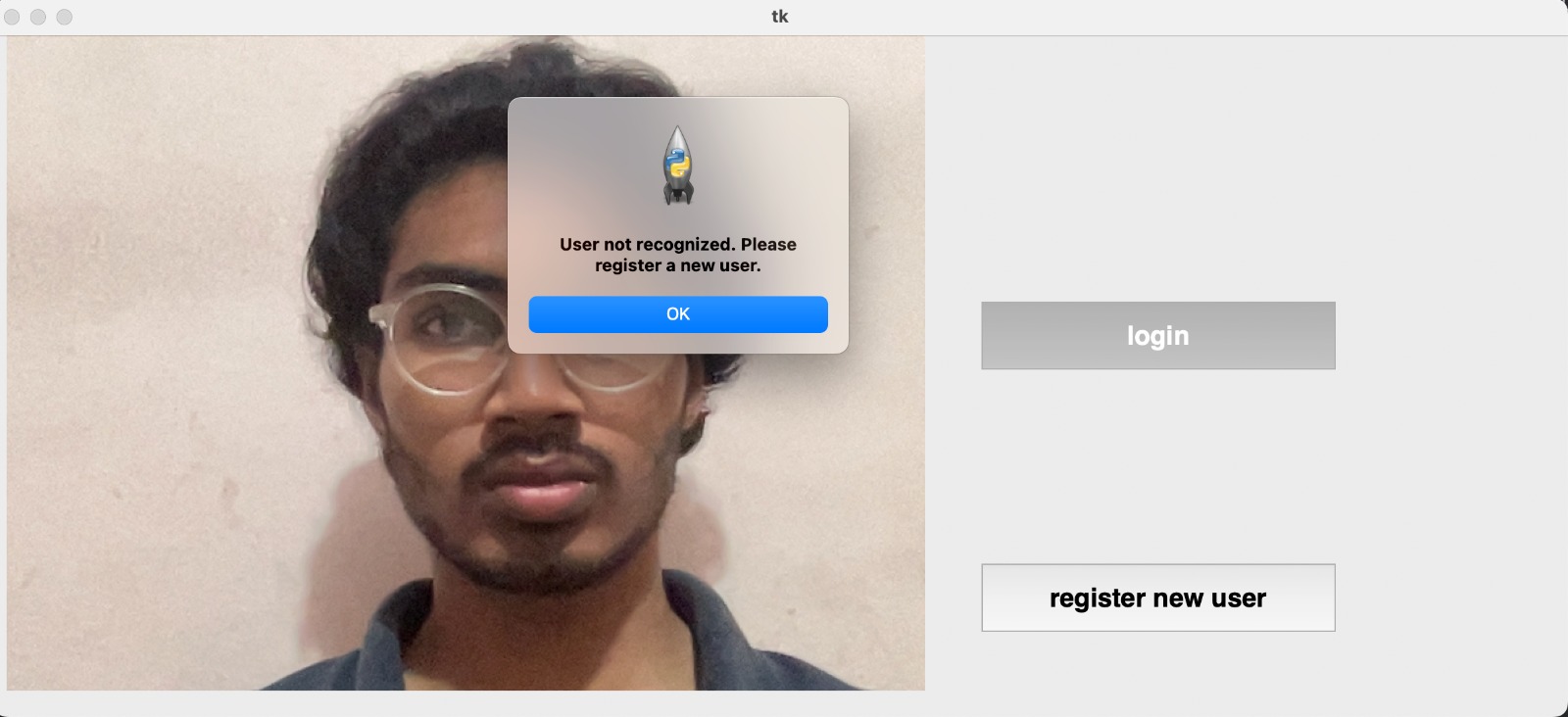
**** The image shows a successful registration message for the user "a05"

***Fig-6.3 User registration***

The image shows a GUI interface displaying a warning message, "This is a fake user!" with an "OK" button, alongside options for "login" and "register new user" at the bottom**.**

***Fig-6.4 Spoofing Detection Interface***

The image shows a facial recognition system interface where a user is not recognized, prompting the option to register a new user.

 ***Fig-6.5 User Not Recognized***

**6.2 Result Analysis**

1. **System Performance:**

* The system successfully integrates facial recognition for real-time attendance marking, utilizing Python libraries (OpenCV, face\_recognition).
* It includes anti-spoofing measures, enhancing security against fake attendance attempts.
* During testing, the system was able to recognize faces accurately under various conditions (lighting, multiple users), but it shows sensitivity to environmental factors like low light, which might affect performance.

1. **Accuracy and Efficiency:**

* The use of advanced algorithms (FaceNet for face recognition, Fourier Spectrum Analysis for anti-spoofing) allows for high accuracy in identifying users, effectively reducing false positives and negatives.
* The system's response time was optimized to log attendance within 2-3 seconds of face detection, ensuring efficiency.

1. **Security:**

* The anti-spoofing feature was effective in preventing fraudulent attempts using images or videos, addressing a critical vulnerability in many biometric systems.

1. **Usability:**

* The interface, built using Tkinter, was found to be user-friendly, allowing administrators to manage attendance records and generate reports easily.
* Real-time attendance marking reduces the need for manual input, automating the process for better convenience and reliability.

1. **Limitations:**

* Sensitivity to lighting conditions can lead to issues in recognition accuracy.
* Handling multiple users in a single frame could be improved, as it may not always log all detected faces accurately.

**CONCLUSION**

The Smart Attendance System successfully automates the attendance tracking process using facial recognition technology, eliminating the need for manual input or traditional attendance methods like sign-in sheets and RFID cards. The system achieves high accuracy in detecting and recognizing faces using state-of-the-art algorithms, such as FaceNet, and integrates an anti-spoofing mechanism to prevent fraudulent attendance marking.

Through real-time face detection and recognition, the system simplifies attendance logging while reducing errors like proxy attendance and manual mistakes. The interface provides easy access for administrators to manage records and generate attendance reports efficiently. Testing results demonstrate the system’s reliability, speed, and security in a variety of conditions, proving it to be a practical and effective solution for educational institutions and corporate environments.

Despite its success, the system does show some sensitivity to environmental factors like lighting and the presence of multiple users in the frame. These limitations present opportunities for improvement in future iterations. Overall, the Smart Attendance System meets the project objectives of accuracy, security, and efficiency, making it a valuable tool for modern attendance management.

**FUTURE WORK**

While the current system achieves its primary goals, there are several areas where it can be enhanced for better performance and broader applications:

1. **Improvement in Lighting Adaptability:**
   * Future versions of the system could integrate more advanced image processing techniques to improve recognition accuracy in low-light conditions. Incorporating infrared cameras or advanced lighting compensation algorithms could further enhance performance in diverse environments.
2. **Handling Multiple Users Simultaneously:**
   * The system could be optimized to handle scenarios where multiple users are present in the frame simultaneously. Techniques like parallel processing or faster face detection algorithms could be implemented to ensure efficient attendance marking for large groups.
3. **Cloud Integration for Remote Access:**
   * Integrating cloud-based storage and processing would allow for remote access to attendance records and real-time monitoring. Administrators could view and manage attendance data from any location, making the system scalable for large organizations.
4. **Mobile Application Development:**
   * Developing a mobile application for administrators and users could enhance the system’s usability. Through a mobile app, users could view their attendance records, and administrators could manage data on the go.
5. **Integration with Other Systems:**
   * The system could be extended to integrate with other organizational systems, such as payroll or academic management platforms, automating more workflows and providing comprehensive data management.
6. **Enhanced Anti-Spoofing Measures:**
   * Although the current anti-spoofing mechanism is effective, future iterations could incorporate more advanced machine learning techniques to further improve the detection of fraudulent attempts.

**REFERENCES:**

1. M. Arsenovic, S. Sladojevic, A. Anderla and D. Stefanovic, "FaceTime — Deep learning based face recognition attendance system," 2017 IEEE 15th International Symposium on Intelligent Systems and Informatics (SISY), Subotica, Serbia, 2017.
2. S. Sawhney, K. Kacker, S. Jain, S. N. Singh and R. Garg, "Real-Time Smart Attendance System using Face Recognition Techniques," 2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, 2019.
3. S. Kadambari, G. Prabhu, D. Mistry and M. Khanore, "Automation of Attendance System Using Facial Recognition," 2019 International Conference on Advances in Computing, Communication and Control (ICAC3), Mumbai, India, 2019.