A

Final Project Report

On

"FACE AUTHENTICATION BASED ATTENDANCE SYSTEM"

Submitted in Partial Fulfillment of the Requirement for Degree of Bachelor of Science in Computer Science and Information Technology (B.Sc. CSIT) Awarded by Tribhuvan University

Submitted by:

Naresh Adhikari(23034/076)

Yogesh Bhattarai(23107/076)

Basan Ghimire(23011/076)

Under the Supervision of:

Roshan Kumar Nandan

DEPARTMENT OF STATISTICS AND COMPUTER SCIENCE

Patan Multiple Campus

Patandhoka, Lalitpur

March, 2024

Certificate of Declaration

Award Title: BSc. Computer Science and Information Technology

Declaration Sheet

(Presented in partial fulfilment of the assessment requirements for the above award)

This work or any part thereof has not previously been presented in any form to the University or to any other institutional body whether for assessment or for other purposes. We confirm that the intellectual content of the work is the result of our own efforts and of no other person.

It is acknowledged that the author of any project work shall own the copyright, However, by submitting such copyright work for assessment, the author grants to the University a perpetual royalty-free license to do all or any of those things referred to in section 16(I) of the Copyright Designs and Patents Act 1988. (viz. to copy work; to issue copies to the public; to perform or show or play the work in public; to broadcast the work or to make an adaptation of the work).

Declared By:

…….	……	……
Naresh Adhikari	Yogesh Bhattarai	Basan Ghimire
TU Symbol No: 23034	TU Symbol: 23107	TU Symbol: 23011

Date: 16/03/2024

Acknowledgement

This project is prepared in the partial fulfillment of the requirement for the degree of Bachelor in Computer Science and Information Technology (BSC.CSIT). The satisfaction and success of completion of this task would be incomplete without heartfelt thanks to people whose constant guidance, support and encouragement made this work successful. On doing this undergraduate project we have been fortunate to have help, support and encouragement from many people. We would like to acknowledge them for their co-operation.

Our first thanks go to Tribhuvan University for designing such a worthy syllabus and making us do this project. Our next batch of thanks goes to the Institute of Science and Technology without whose help our project would have been impossible. Our very sincere and heartfelt thanks go to Mr. Roshan Kumar Nandan, our project supervisor who constantly guided us through the project time period. Without his guidance, our project would have been impossible. Last but not the least, we want to thank every direct and indirect hands that were involved in completion of this project for final year project and thanks for sharing his valuable time with us to evaluate our work and provide professional suggestions with us. This project has been a wonderful experience where we have learned and experienced many beneficial things.

Naresh Adhikari(23034/076)

Yogesh Bhattarai(23107/076)

Basan Ghimire(23011/076)

Abstract

The suggested system uses facial recognition and detection based on Haar Cascade algorithms. Individuals are reliably detected and recognized by the system thanks to their distinct facial traits. In order to isolate the essential facial information for additional analysis, facial detection algorithms recognize and extract facial areas from input photos or video frames. The identities of the individual users are then ascertained by LBPH face recognition algorithms, which match these attributes to pre-registered faces, kept in the system's database and compute confidence scores. Because of the system's user-friendly interface, admin can handle attendance data with ease. They may easily obtain attendance statistics, add and remove students from the system's database, and keep an eye on real-time attendance data.

With improved security, efficiency, and accuracy along with real-time monitoring and extensive reporting features, the suggested Attendance Management System completely transforms the traditional attendance tracking method. This system is a major step forward in the direction of efficient and intelligent attendance management, and it has the potential to be used in a wide range of educational institutions, businesses, and industries.

Keywords: *Face Detection, Face Recognition, Attendance, LBPH, Database*

Table of Contents

List of Abbreviations

AI: Artificial Intelligence

CNN: Convolutional Neural Networks

CAGR: Compound Annual Growth Rate

CSV: Comma-Separated Values

HOG: Histogram of Oriented Gradients

IDE: Integrated Development Environment

JPEG: Joint Photographic Experts Group

LBPH: Local Binary Patterns Histogram

ML: Machine Learning

MySQL: My Structured Query Language

OpenCV: Open Source Computer Vision

PNG: Portable Network Graphic

SDLC: Software Development Life Cycle

SRS: Software Requirements Specific

SVM: Support Vector Machine

UML: Unified Modelling Language

List of Figures

List of Tables

Chapter 1

Introduction

1.1. Introduction

Facial recognition and detection systems for automated attendance have been developed and deployed in a variety of settings, including offices, event spaces, and educational institutions. Compared to more conventional methods like sign-in sheets or roll calls, these systems provide more accuracy and efficiency by using biometric data—specifically, face features—to identify people and track their attendance.

Through continuous face analysis, this approach seeks to improve the precision and effectiveness of current attendance monitoring techniques. Due to its increased precision and efficiency, facial detection and identification technology has the potential to revolutionize traditional attendance tracking techniques. Nevertheless, there are some drawbacks and worries with this technology, such privacy issues and the possibility of bias in the recognition algorithms. In order to guarantee the moral and appropriate application of this technology, these issues need to be addressed.

All things considered, automated attendance systems that make use of facial detection and identification technology mark a substantial progress in the field of intelligent and efficient attendance management in a variety of settings. Additional investigation and advancement are required to tackle any obstacles and worries related to this technology and guarantee its moral and conscientious application.

1.2. Problem Statement

The traditional method of recording student attendance is frequently plagued by several difficulties. By doing away with traditional methods of recording student attendance, such contacting students by name or examining their corresponding identity cards, the facial recognition student attendance system highlights its simplicity. There aren't any not only obstructs the teaching process but also diverts students' attention during exam periods. During the lecture periods, an attendance list is circulated throughout the classroom in addition to calling names. It might be challenging to circulate the attendance sheet around the lecture hall, particularly in classes with a lot of pupils.

The manual signing of students' presence, which is tedious and distracts them from signing for their attendance, is thus intended to be replaced by a facial recognition attendance system. Additionally, the automatic facial recognition student attendance system eliminates the need for lecturers to repeatedly count the number of students in order to verify that they are there. It also overcomes the issue of fraudulent approaches.
Making the distinction between recognized and unidentified photos is one of the challenges in face recognition. Therefore, in order to prevent omission, the identification procedure must be completed within predetermined time limitations. This calls for the development of a real-time functioning student attendance system.

When the background, lighting, posture, and expression of the students vary, the traits that were taken from the face photos that best capture their identities must remain constant. Fast calculation times and high precision will be used to evaluate the performance.

1.3. Objectives

The objective of this project is to develop face recognition attendance system.
Expected achievements in order to fulfill the objectives are:

- To detect the face segment from the video frame.
- To extract the useful features from the face detected.
- To classify the features in order to recognize the face detected.
- To record the attendance of the identified student.

1.4. Scope and Limitation

1.4.1. Project Scope

The scopes of our initiative, as well as how we intend to distribute it broadly and effectively, are explained in this part. The following features of the desktop-based face recognition attendance system program are covered:

- **Student Registration:** Capture and store facial images along with student details.
- **Check-in and Check-out:** Allow students to check in and check out using facial recognition.
- **Attendance Tracking:** Record attendance data in a database.

- **User Management:** Administer accounts and permissions for system access.
- **Reporting:** Generate reports on attendance statistics and trends.

1.4.2. Project Limitation

This project is correct in what it does, but it might yet be improved. Since it relies on fixed patterns and features to identify faces, it can be vulnerable to spoofing attacks where an attacker uses a photo or a video of a person's face instead of their actual face to trick the system. This limitation highlights the importance of incorporating liveliness detection or using more advanced facial recognition techniques to enhance security and prevent such attacks. Furthermore, System may struggle with detecting faces in challenging conditions such as poor lighting, occlusions (where part of the face is hidden), or variations in facial expressions. These limitations can affect the overall accuracy and reliability of the system, especially in real-world scenarios where such conditions are common. Thus, there a big room for improvement in the future.

1.5. Development Methodology

For this project, the incremental process was selected since the project's needs are well-established, understood, and specified. Given the methods the project, which involves face detection and identification, would benefit greatly from the use of increment methodology, which facilitates the design, implementation, and testing processes with each increment added over time.

In order to meet the intended project's functional requirements and attain high precision, this endeavour will require a great deal of trial and error.

The fundamental prerequisite for the attendance system is to be able to enter a photo into the system and receive output related to attendance and categorization. This leads to the division of the system development process into several smaller development initiatives.

These initiatives for the partial development of the system were:

- The creation of the system that takes the pictures as input and sends them to the model.
- The system that receives an image as input and sends it to the model.

- The system that receives the output produced by the model and outputs a result.

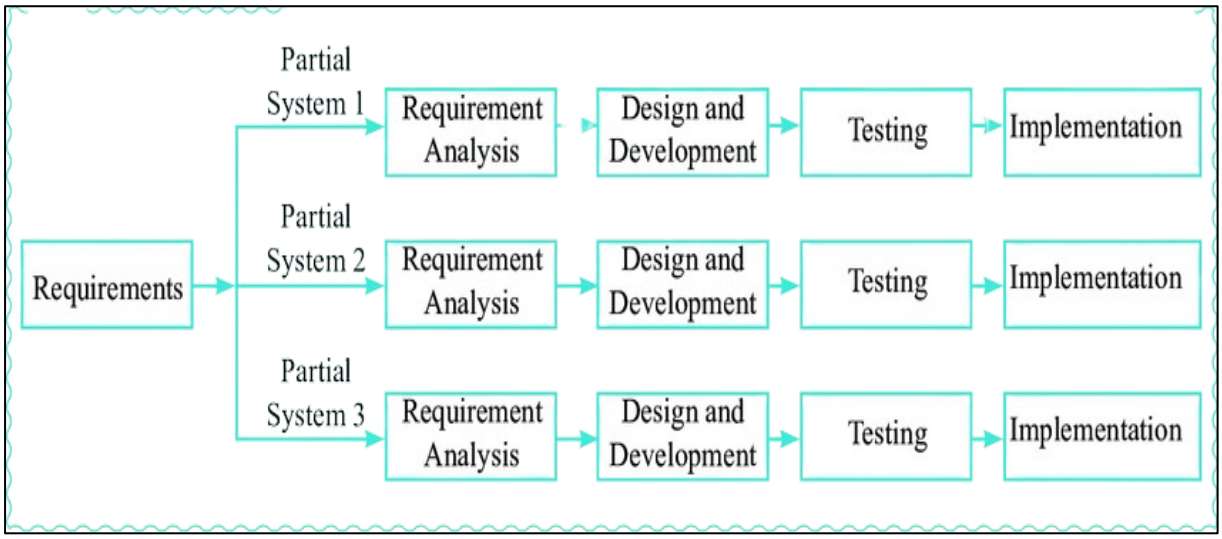


Figure 1: Block Diagram of Incremental Model

Face Recognition

A computer vision approach called facial detection locates and recognizes human faces in pictures or video streams. In order to identify facial landmarks and differentiate faces from other objects or backgrounds, it employs algorithms. Face identification is accomplished by a variety of techniques, such as Convolutional Neural Networks (CNN), Support Vector Machines (SVMs), and Haar cascade. These techniques reliably recognize faces in real-time by analyzing visual patterns, geometric characteristics, and texture information.

Recognition of Faces

By recognizing and authenticating a person's identity using only their facial traits, facial recognition technology surpasses face detection. It entails taking pictures of people's faces or video frames, identifying their distinctive facial features, and comparing them to a database of people who have already been identified. Eigenfaces, Fisherfaces, Local Binary Patterns Histogram (LBPH), and Deep Learning-based techniques (e.g., Convolutional Neural Networks) are some of the techniques utilized for facial identification. According to [1], these techniques allow for excellent accuracy and resilience in identifying people in a variety of stances, emotions, and lighting situations.

A Program for Automatic Attendance

Technology for facial detection and recognition offers a reliable and effective remedy for creating methods for online attendance. These systems provide precise monitoring and identification of people through the use of computer vision algorithms and deep learning techniques, guaranteeing a quick and safe attendance management procedure. Technology has enormous potential to improve attendance monitoring in a variety of industries and foster an atmosphere that is more dependable and efficient as it develops. There are several benefits to incorporating facial recognition and detection technologies into an online attendance system.

The main benefit is that it does away with the necessity for manual check-ins, which saves time and lowers administrative work. Employees or students only need to point their faces toward a camera or webcam for the system to automatically identify and detect who they are. This process ensures a streamlined and contactless attendance experience [2].

1.6. Report Organization

1st Portion: Report Introduction

This section contains the skeleton view of the report's introduction. It includes elements like Declaration certificate, Acknowledgement, Abstract, Table of Contents, List of Figures, List of Tables, and List of Abbreviations.

2nd Portion: General Introduction and Literature Review

This section describes the overall project concept in brief with a look at the past research and existing applications related to our own. Research, analysis, and investigation have been conducted on previously done work. It contains all the comparisons of those systems along with the designed system. Altogether, work on 2 different pieces of research has been performed.

3rd Portion: Main Content

This is where all the system details have been mentioned. It is composed of different system analysis techniques, SRS documentation, and process design with necessary UML diagrams along with user interface diagrams, system architecture, and testing methodologies.

4th Portion: Conclusion and Critical Evaluation

This portion includes the answer to all the objectives and aims from the general introduction. Also, the evaluation of the report, findings, learnings, and future recommendations are included in this portion.

5th Portion: Citations and References

All the works in this project other than the authors' have been carefully cited and referenced in this section. Sources include books, websites, journals, magazines, case studies, and thesis documents.

Chapter 2

Background Study and Literature Review

2.1. Fundamental Theories

Woodrow Wilson Bleadsoe invented face recognition technology in 1966. It compares human faces in digital photos or video clips using a facial database. The concept of face recognition emerged to enable computers to locate and identify human faces with accuracy and speed.

To enhance face recognition performance, numerous algorithms have been created [3]. Although facial recognition technology has been around since the 1960s, it really took off in the late 1990s and early 2000s. Despite the accuracy and processing power of early systems were limited, they have since significantly improved. The global facial recognition market size was valued at USD 4.35 billion in 2020 and is expected to reach USD 12.92 billion by 2028, with a CAGR of 14.5% from 2021 to 2028 [4].

2.2. Terminologies

These are few basic terminologies based on this report about e-commerce and developed application:

Table 1: Terminologies to Face Authentication Attendance System

Terminologies	**Meaning**
Facial Recognition Technology	That identifies or verifies a person's face.
Biometric Authentication	Using unique physical characteristics to verify identity.
Face Detection	Identifying and locating faces in images or videos.
Feature Extraction	Identifying unique features of a face for recognition.
Face Matching	Comparing faces to determine if they match.
Liveness Detection	Ensuring that a face being presented is from a live person.
Deep Learning	A type of machine learning that mimics how the brainworks.
Algorithm	A set of rules or instructions for solving a problem.

Accuracy	How correct a facial recognition system is.
Privacy Concerns	Issues related to the protection of personal information.
Histogram	A visual representation of the distribution of data.
Trained Data	Data used to teach a facial recognition system to recognize faces.

2.3. Literature Review

Facial recognition technology has emerged as a powerful tool with diverse applications, including attendance tracking in educational institutions and workplaces. The ability to accurately and efficiently identify individuals based on their facial features makes facial recognition an attractive solution for attendance systems, offering advantages such as automation, convenience, and enhanced security.

In general, facial recognition algorithms are divided into two: algorithms whose facial features are regulated by humans and algorithms that let the system choose the best facial features themselves. Good face recognition can contribute to high feature extraction, and better classification. Currently, face recognition can be used for many things, one of which is student attendance at universities [5]. Face recognition can improve and manage the attendance system, reduce errors in the manual recording process by providing an automatic and reliable attendance system, increase privacy and security, prevent fake attendance, and provide regular attendance reports [6]. Such face recognition can be used to save time for educators and students and to stop false attendance. The steps taken in facial recognition include recording student videos, converting them into frames, connecting them to a database to ensure their presence or absence, and marking the presence of certain students to maintain records. All the data will be stored and displayed using the display screen. One of the examples is a web page [7]. In general, the use of an attendance system with facial recognition is influenced by many things, such as the database and algorithm. Research suggests providing more precise specifications on our database to improve the training of our facial recognition system. In addition, in the paper, the data/images that we enter in the database affect our facial recognition systems, such as the use of makeup, glasses, or hairstyle changes. The traditional algorithm widely used is

LBPH, a combination of LBP with an Oriented Gradient Histogram (HOG). Local Binary Patterns (LBP) is a visual descriptor used for computer vision classification. LBP was first described in 1994 by Timo Ojala and David Harwood at the University of Maryland and has since been found to be a powerful feature for texture classification. This amalgamation significantly improves the detection performance across multiple data sets [8].

The Face Recognition algorithm that we know is divided into two: algorithms that work traditionally and algorithms that use deep learning. The working principle of the traditional method is to take an image as input, then resize the image to fit the algorithm. After that, the image that was originally RGB will be changed to gray or according to the filter in the algorithm. Then, a face detection algorithm is needed so that the face recognition algorithm can recognize the faces that are successfully detected; the datasets were obtained from the predefined number of the image that was taken by the camera. A comparison of the accuracy between traditional facial recognition algorithms from research of several papers is for LBPH to get 86.47% accuracy, Eigenfaces get 15.09% accuracy, Fisherfaces gets 36.4% accuracy. The LBPH algorithm is proven better than the fisherface and eigenface algorithms. Furthermore, it has been explained in that the LBPH Algorithm's accuracy is better than the other because the LBPH Algorithm can recognize not only the front face but also the side face and also the faces under any light [9]. After the successful review of the literature, we conclude to use the Haar cascade algorithm for face detection and the LBPH algorithm for face recognition.

2.4. Existing Systems

For studying existing systems, research has been conducted over two attendance system. It gives a general idea about where the current attendance system lies.

2.4.1. Register-Based Attendance System

A register-based attendance system is a commonly used method for tracking attendance, where individuals are required to physically sign a register or logbook upon arriving at a specific location. This system is widely used in schools, workplaces, events, and other scenarios where monitoring who is present is essential. In this system, individuals typically provide their signature, along with the date and time of their arrival, and sometimes additional details like the purpose of their visit or how long they plan to stay. The completed

register acts as a record of attendance, offering insights into who was present at a specific time.

One of the main advantages of a register-based attendance system is its simplicity and low-tech requirements, needing just pen and paper or digital equivalents. It's also cost-effective and easy to implement and maintain. However, there are drawbacks such as it being time-consuming for individuals to sign the register and the potential for errors or fraud if not managed properly. Despite these drawbacks, the register-based approach remains popular for tracking attendance in various environments [10].

2.4.2. Fingerprint-Based Attendance System

A fingerprint-based attendance system is a widely embraced method for monitoring attendance, where individuals use their fingerprints to register their presence at a specific location. This system is commonly seen in schools, workplaces, and other settings where precise attendance tracking is essential. In this system, individuals place their finger on a fingerprint scanner, which captures and analyzes the unique patterns of their fingerprint. The captured fingerprint is then compared to a database of stored fingerprints to identify the individual and verify their identity. Once identification is successful, the system records their attendance and updates its records accordingly [11] [9].

One notable advantage of a fingerprint-based attendance system is its high level of accuracy and difficulty to forge, given that fingerprints are unique to each individual. Additionally, the process is relatively swift and efficient as individuals only need to place their finger on the scanner to register their attendance. However, it's important to acknowledge that implementing and maintaining a fingerprint-based system may incur higher costs compared to other attendance tracking methods. Furthermore, concerns about privacy and potential misuse of fingerprint data may arise among some individuals.

2.4.3. Problem with Existing System

The register-based attendance method has shortcomings, including lengthy individual sign-in procedures and a greater possibility of mistakes and discrepancies in the reported data. Furthermore, this system's manual nature leaves it open to fraudulent acts, such as someone signing in on another people's account.

Similar difficulties arise with fingerprint-based attendance systems. Due to conditions like dry or damaged skin, they can have problems with unreadable fingerprints, which would compromise the precision of the attendance monitoring system. Performance issues with

the fingerprint scanners themselves may potentially be a hindrance to the system. Furthermore, because biometric data collection and storage raises worries about data security and potential abuse, privacy issues are raised [12].

Chapter 3

System Analysis

3.1. System Analysis

The system is developed based on the Object-Oriented Approach. In this analysis phase, necessary Class and Object Diagrams using object modelling, State and Sequence Diagrams using dynamic modelling, Activity Diagrams using Process modelling are designed for the proper analysis of the system.

3.1.1. Requirement Analysis

Face Authentication Attendance System is a desktop-based system to detect faces to help in attendance of students in a classroom. It primarily focuses on the detection of faces with a higher level of accuracy. The requirement of this project is given with reference to [13] [14] [15] [2].

- System Camera
- Student Details
- Photo Sample of the Student
- Python IDE
- MYSQL
- Necessary Libraries

These are the basic requirements of the project. Also, there are various functional and nonfunctional requirements of this particular project, Face Authentication Attendance System.

3.1.1.1. Functional Requirement

A functional requirement is something a system must do. Use case diagrams help in visualizing the functional requirements by showing how users will interact with the system to take the attendance.

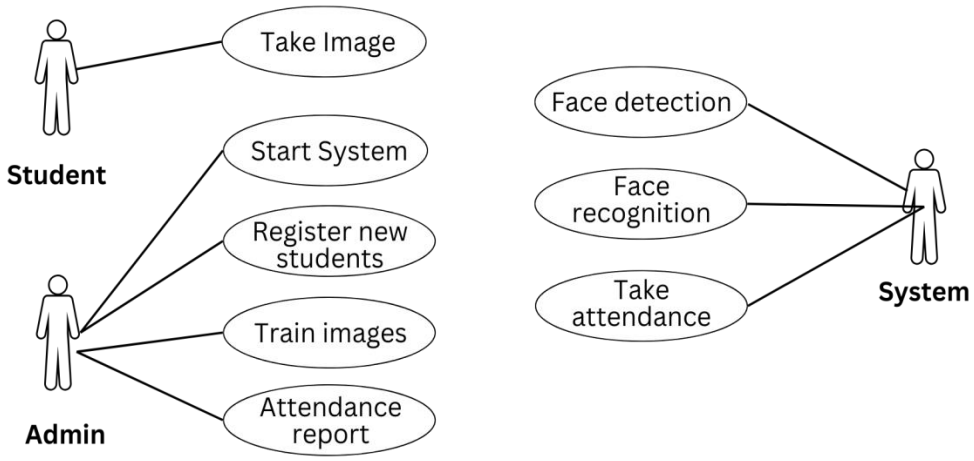


Figure 2: Use Case Diagram of Face Recognition System

3.1.1.2. Non- Functional Requirements

The non-functional requirements in context to the project are as follows:

a) Performance Requirements

- System can produce results faster on 4GB/8GB of RAM.
- The system will be available 100% of the time. Once there is a fatal error, the system will provide understandable feedback to the user.

b) Safety and Security Requirements

The system is designed in modules where errors can be detected and fixed easily.

c) Software Quality Attributes

- **Reliability:** The system should be reliable in accurately verifying and recording attendance based on facial recognition.
- **Usability:** The system should be user-friendly, with a clear interface for both administrators and users. It should also provide easy enrollment of new users and straightforward attendance tracking.
- **Maintainability:** The system will be developed using the standard software development conventions to help in easy review and redesigning of the system.'

- **Accuracy**: The facial recognition algorithm should have a high level of accuracy in identifying individuals to avoid errors in attendance tracking.
- **Performance:** The system should perform efficiently, especially in high-traffic scenarios where many individuals need to be authenticated quickly during attendance recording.

3.1.2 Feasibility Analysis

The goal of feasibility study is to logically and objectively identify the advantages and

Disadvantages of a proposed project or an already-existing business, as well as environmental risks and opportunities, the resources needed to proceed, and the likelihood of success. To ascertain whether developing a new or enhanced system is compatible with the costs, advantages, operation, technology, and time, a feasibility study is required.

3.1.2.1. Technical Analysis

The project is developed for general use. In order to access the application developed in this project, the user requires a desktop/laptop computer. The main requirement of the system from a developer's view are Python and desktop/laptop computer. The required technologies existed. These requirements can be fulfilled. The following points were considered for the project's technical feasibility:

- The system will detect face from real time video/photo.
- After detecting the face, it will process and give accurate class result.
- If any error is occurred, system will handle with ease.

3.1.2.2. Operational Analysis

It is focused on the system's operational capabilities. Because it's a computer application, anyone with a basic understanding of user interface design may easily navigate the system. All that is needed for the smooth operation is a computer device for ordinary use. Additionally, the user interface is friendly. As a result, the system can function.

3.1.2.3. Economic Analysis

A system request is economically feasible if the projected benefits of the proposed system outweigh the estimated costs involved in developing or purchasing, installing, and operating it. Costs can be one time or continuing and can be incurred at various times during project development and us. Hence, the proposed system is economically feasible as it passes the cost of existing system and has many benefits along with less development cost. The budget for the development has been allocated.

3.1.2.4. Schedule Analysis

Keeping the requirements, operation and performance specification, we feel the time period given for us for this project is more than enough for us. The project can be completed within the given time frame. So, this project is feasible in context of our time allocated to us.

The working schedule is scheduled as follows:

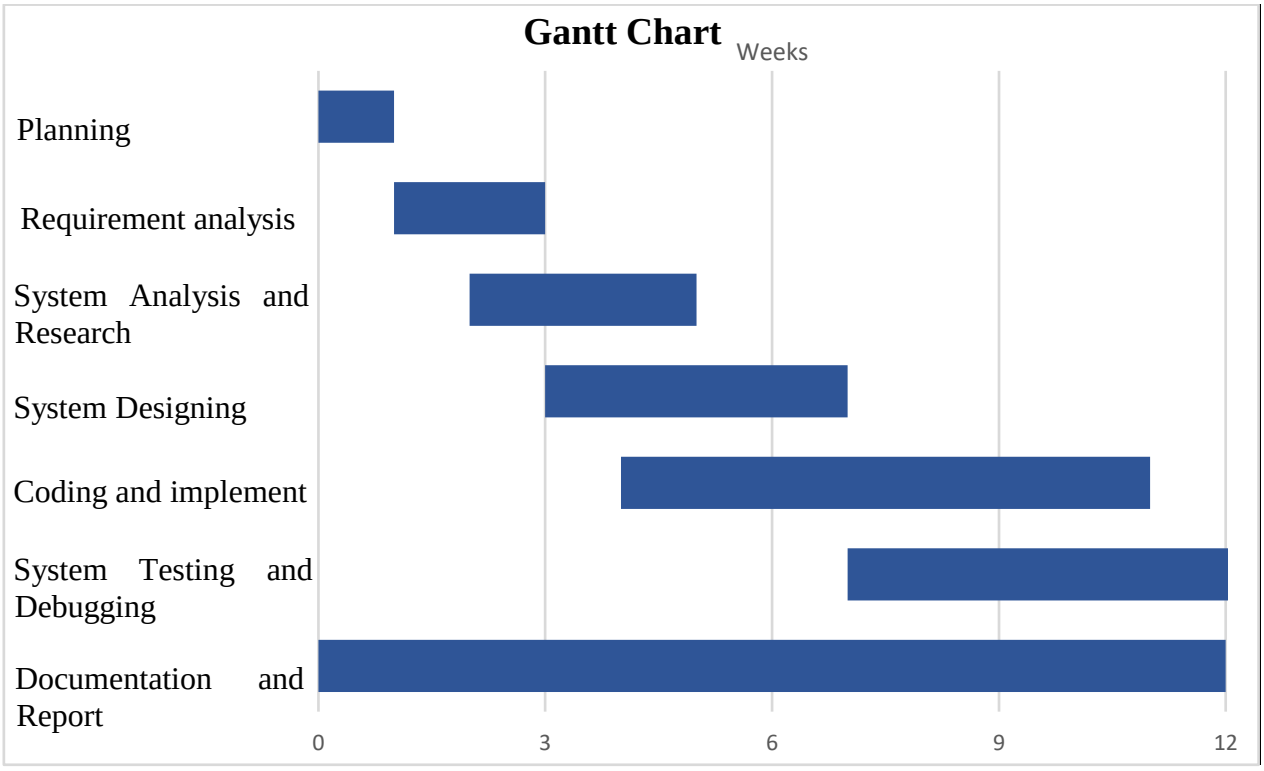


Figure 3: Working Schedule Gantt Chart

3.1.3. Analysis

3.1.3.1. Object Modeling Using Class Diagram

The class diagram for the facial authentication-based attendance system includes classes such as UI_System for the user interface, Face_Recognition_System for core functionality, Login for administrator access, Register for teacher registration, Student for student information, Train for classifier training, and AttendanceApp for overall attendance management. Each class encapsulates specific attributes and methods tailored to its role, enabling the system's seamless operation.

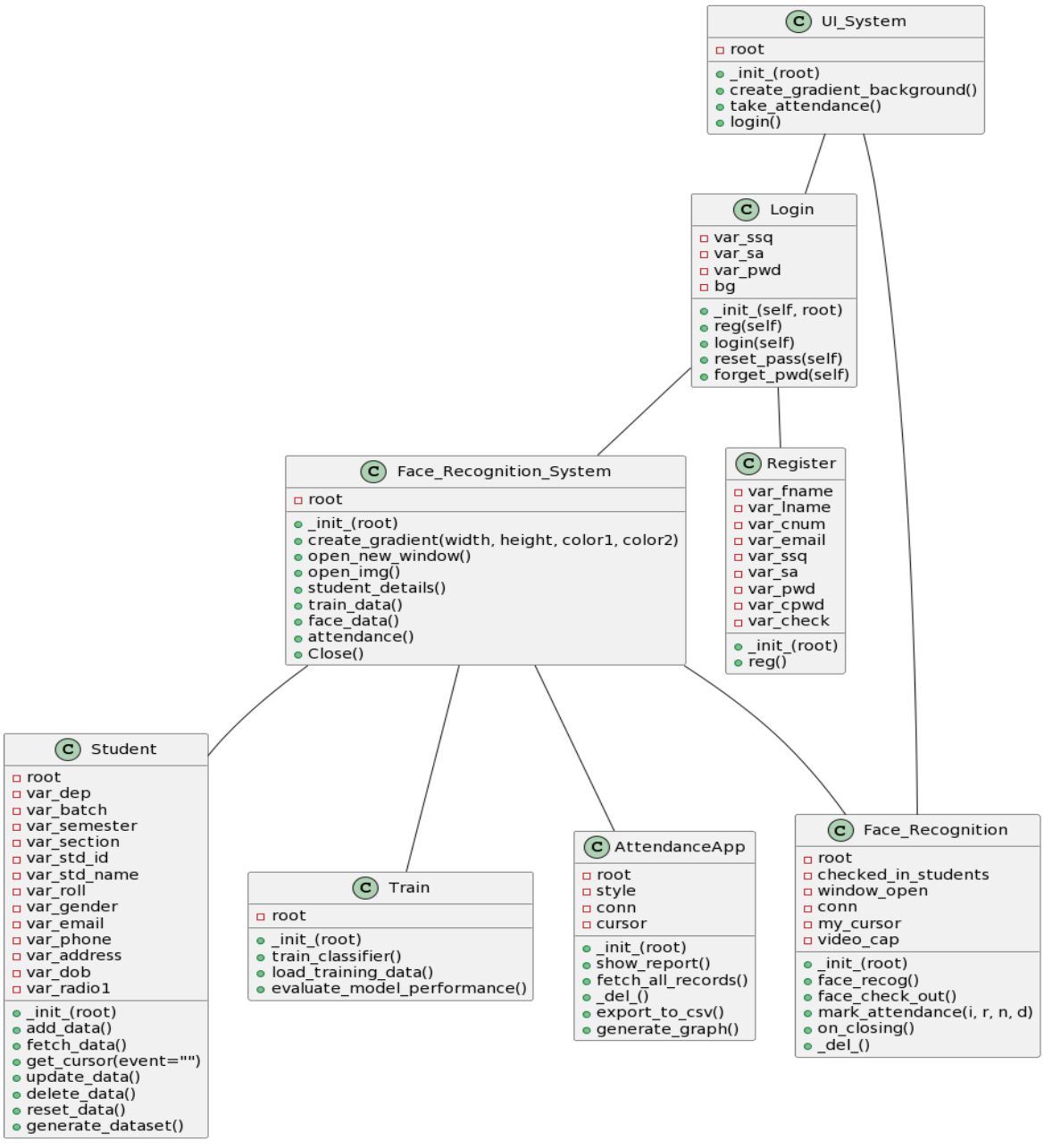


Figure 4: Class Diagram

3.1.3.2. Dynamic Modelling Using Sequence Diagram

The sequence diagram illustrates the flow of interactions within the facial authentication-based attendance system, detailing how students check in and check out and how teachers log in and register.

For Admin (Teacher):

i. UI_System initializes and displays the user interface.
ii. Admin clicks on the "Login" button.
iii. UI_System sends a request to Login for authentication.
iv. Login verifies credentials and grants access.
v. If not registered, admin clicks on "Register" button.
vi. UI_System sends a request to Register for registration.
vii. Register collects admin's information and registers them.
viii. Once logged in, admin interacts with Face_Recognition_System to manage attendance.

For Student:

i. UI_System initializes and displays the user interface.
ii. Student clicks on "Take Attendance" button.
iii. UI_System sends a request to Face_Recognition_System to start facial recognition.
iv. Face_Recognition_System captures student's face and checks against registered faces.
v. If recognized, Face_Recognition_System marks student as checked-in/checked-out.
vi. Face_Recognition_System updates attendance record in database.
vii. System notifies student about check-in/check-out status.

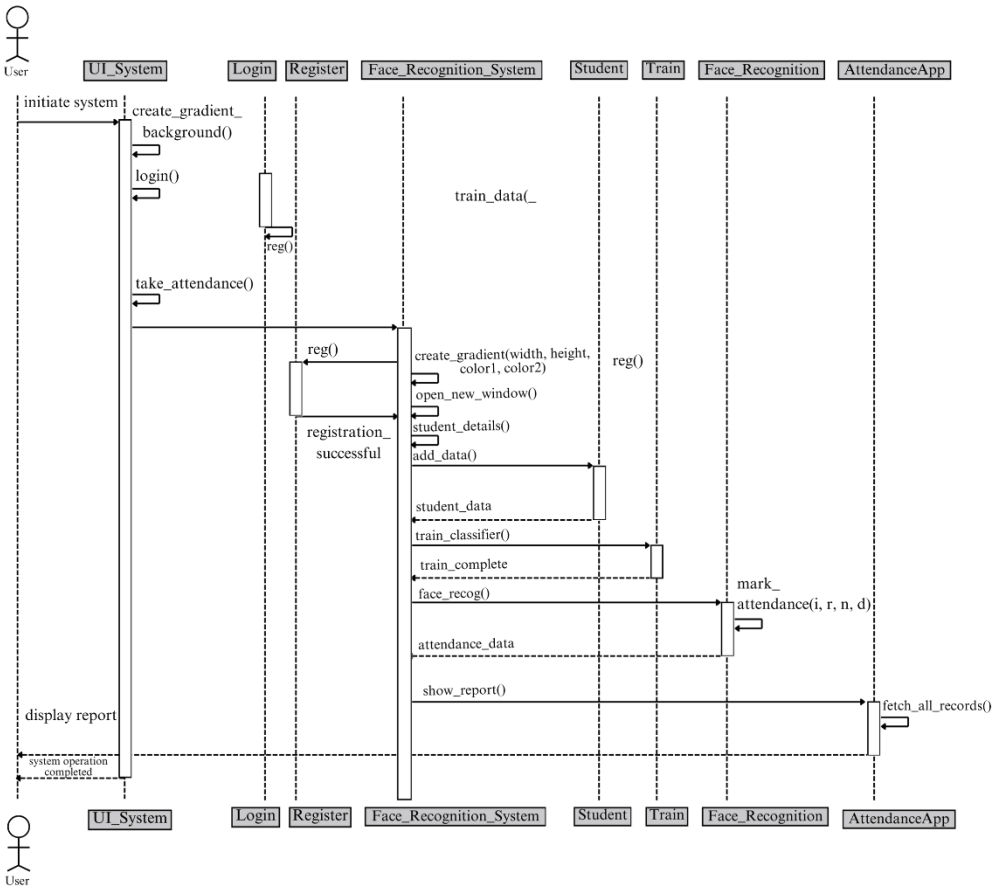


Figure 5: Sequence Diagram

3.1.3.3. Process Modelling Using Activity Diagram

The activity diagram illustrates the workflow of the facial authentication-based attendance system, depicting activities such as student attendance, teacher login, and registration. It provides a visual representation of the system's functionality, showing the sequence of activities and the decision points involved in managing attendance.

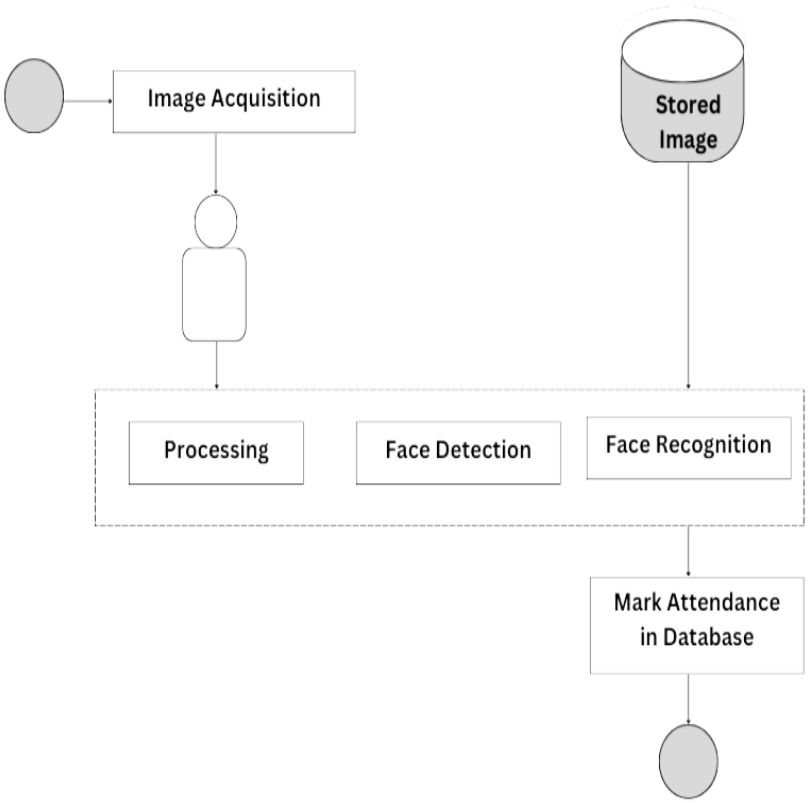


Figure 6: Activity Diagram

Chapter 4

System Design

4.1. Design

In this section, we refine the class diagrams to provide a more detailed representation of the system's structure. Additionally, we introduce various UML diagrams, such as state diagram, component diagram, and deployment diagram to illustrate the dynamic behavior and process flow within the system. These refinements and diagrams serve to enhance our understanding of the system's design and functionality, laying the groundwork for its successful implementation.

4.1.1. Refinement of Class Diagram

1. **Attributes and Methods**
    - Add attributes and methods to classes with more detail.
    - Specify data types and access modifiers for attributes and methods.
    - Consider additional attributes or methods based on system requirements.
2. **Associations**
    - Refine associations by adding multiplicity to indicate the number of instances involved in the relationship.
    - Consider adding roles or names to associations for clarity.
    - Specify the nature of associations (aggregation, composition).
3. **Inheritance and Generalization**
    - Add more subclasses and specify the relationships between them and their superclass.
    - Include abstract classes if needed.

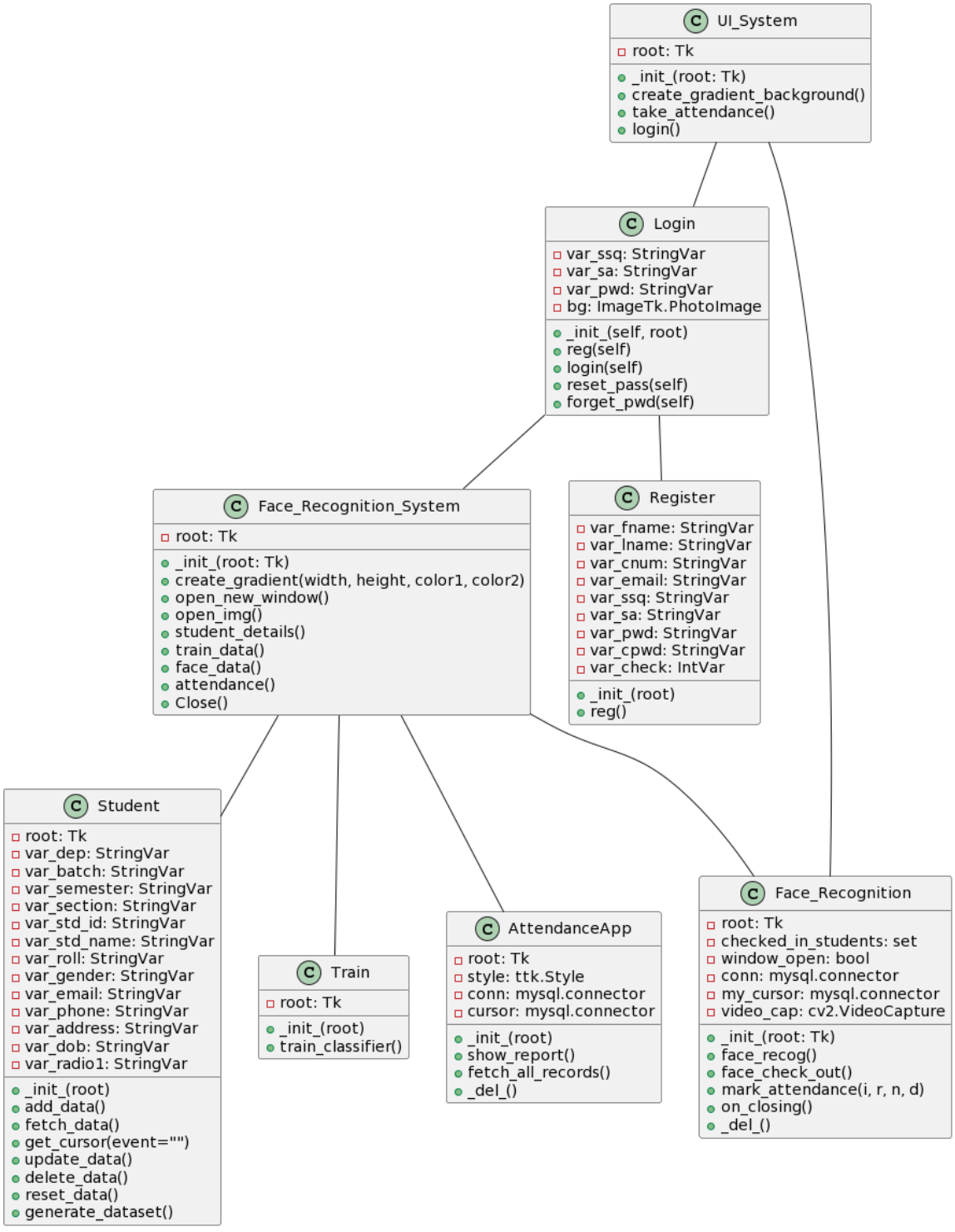


Figure 7: Refinement of Class Diagram

4.1.2. Refinement of Sequence Diagram:

1. Lifelines

- Elaborate on lifelines by adding more details about the objects involved.
- Include activation bars to represent the duration of an object's activity.

2. Messages

- Specify the types of messages (synchronous, asynchronous).
- Add parameters to messages for clarity.
- Include return messages where applicable.

4.1.3. Refinement of Activity Diagram

1. Actions and Nodes

- Break down complex actions into smaller, more manageable sub-actions.
- Add decision nodes to represent conditional behavior.
- Use merge nodes to consolidate parallel flows.

2. Annotations

- Include annotations or notes to provide additional details about actions or decisions.
- Specify any conditions or constraints associated with transitions.

3. Resources

- Specify resources (human or system) associated with each activity.
- Add swimlanes to represent responsibilities or roles.

4.1.4. State Diagram

The state diagram illustrates the various states and transitions in the facial authentication-based attendance system. It depicts how the system progresses from the initial state to final states based on user interactions, authentication processes, and attendance recording. The diagram provides a visual representation of the system's behavior, helping to understand its operational flow and state transitions.

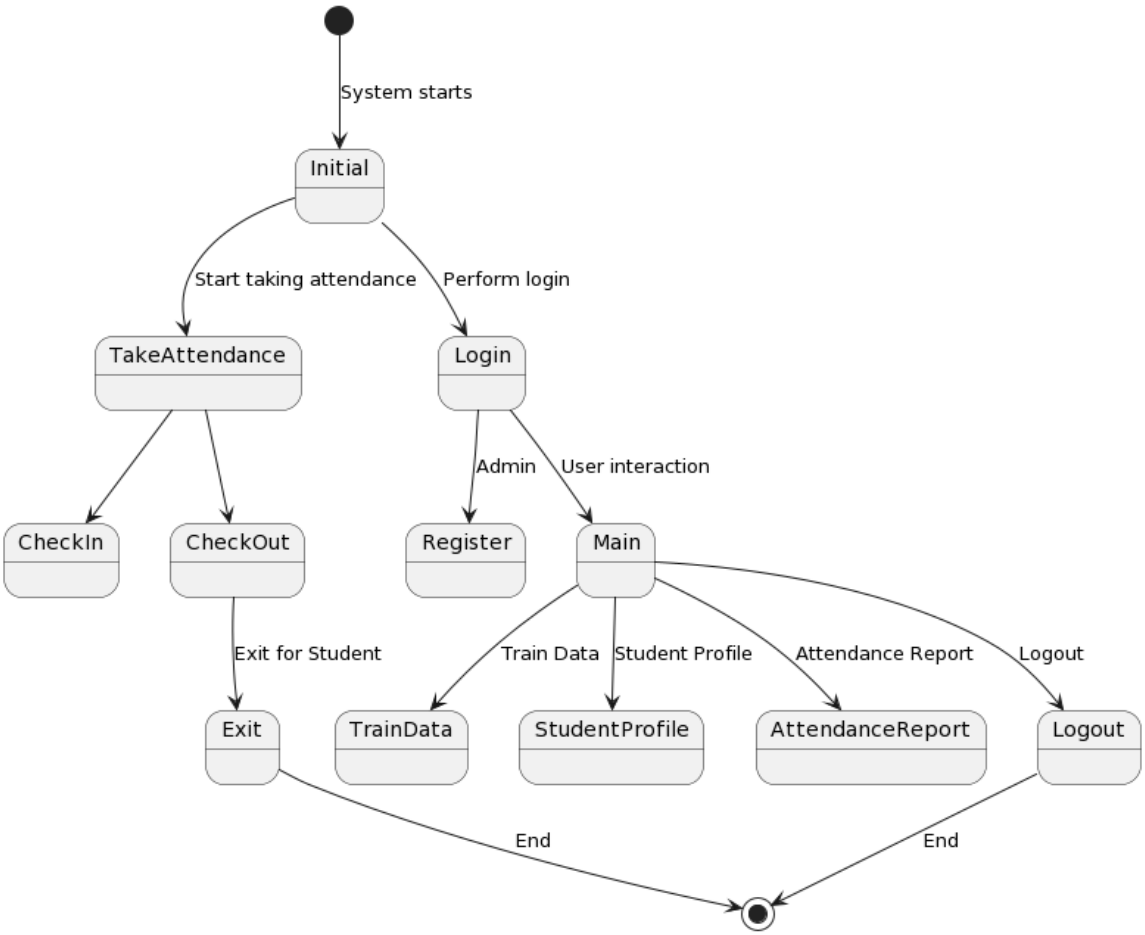


Figure 8: State Diagram

4.1.5. Component Diagram

A component diagram is used to break down a large object-oriented system into the smaller components, so as to make them more manageable. It models the physical view of a system such as executables, files, libraries, etc. that resides within the node. It visualizes the relationships as well as the organization between the components present in the system.

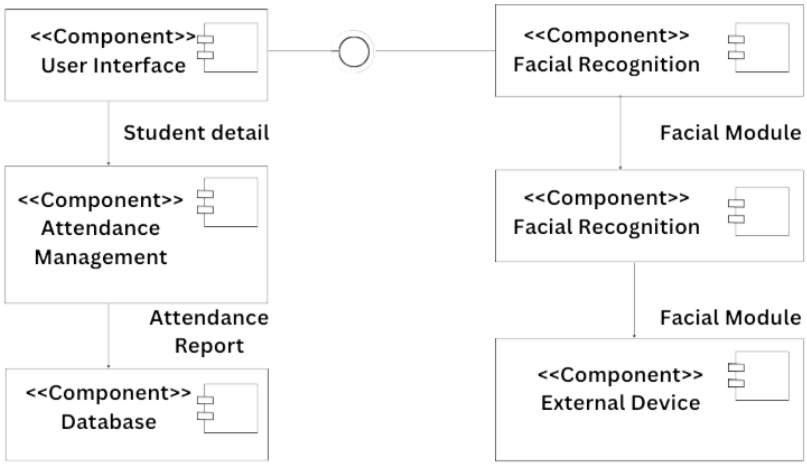


Figure 9: Component Diagram

4.1.6. Deployment Diagram

The deployment diagram for the facial authentication-based attendance system illustrates the physical arrangement of components. It includes the desktop/laptop application for students and teachers, along with the MySQL database server for storing attendance data. This diagram shows how these components interact across different nodes, providing a clear overview of the system's physical deployment.

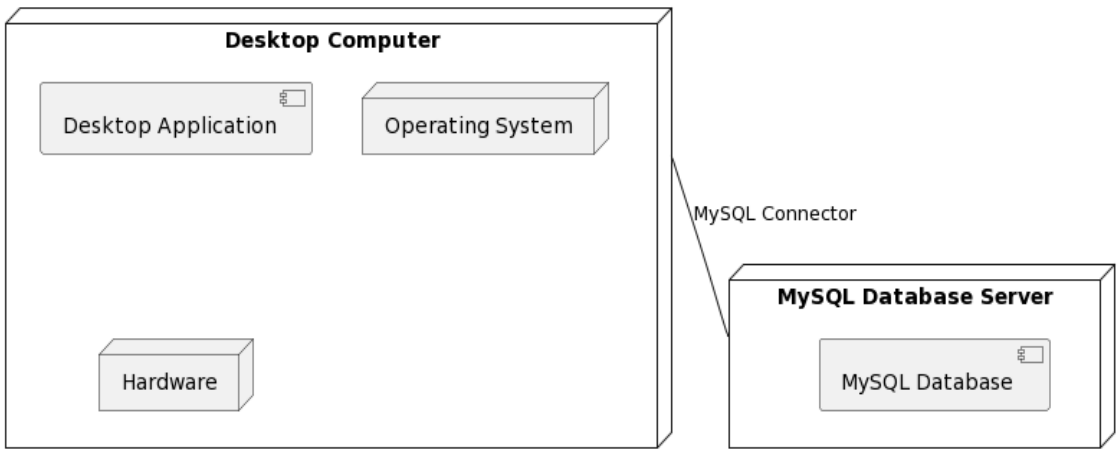


Figure 10: Deployment Diagram

4.2. System Architecture

Our facial recognition attendance system utilizes a combination of Haarcascade for face detection and LBPH for face recognition. The system architecture consists of several key components working in tandem to capture, detect, recognize, and mark attendance based on recognized faces. The system starts by capturing images from a camera, which are then processed for face detection using the Haarcascade algorithm. If a face is detected, the system proceeds to the face recognition stage using LBPH. Once a face is recognized, the system marks the attendance in the database. This process continues in a loop, ensuring continuous monitoring and marking of attendance. Additionally, error handling states such as 'Insufficient Light' or 'Camera Error' are considered, and the system can also manage user enrollment for capturing and storing new faces in the database. Overall, the system architecture is designed to be efficient, reliable, and scalable, providing a robust solution for facial recognition-based attendance tracking.

In general, the working of the face authentication attendance system can be explained as below:

Step 1: The system is always open. The student can click on the Take Attendance button for check-in and check-out.

Step 2: To check-in, the student can click on the “Checkin” Button which will open the camera to detect student and save the time of check-in along with student’s details.

Step 3: If already checked in, it will show message that user has already checked in.

Step 4: If the student wants to check out, s/he can click on “Checkout” button which will save the required details in the database by detecting student.

Step 5: If the student is not registered, the system will show message “Unknown Face”.

Step 6: To register, the admin needs to log in to the system. The admin shall fill the details of students taking the images. The images are trained. Student is now registered.

Step 7: If the student is registered, s/he can go to step 1 to step 4.

(For admin)

Step 8: The admin can manage the system. H/she shall be responsible for managing the attendance report and generating the CSV/Excel file. The admin can also see the attendance

of particular student of particular month to see total days of present and average check in and check out time.

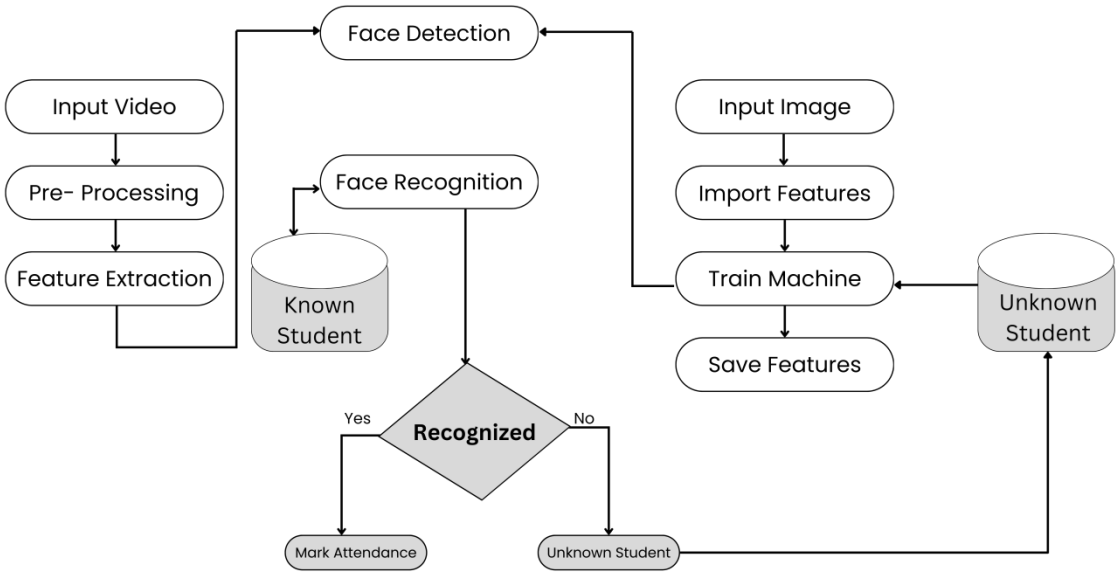


Figure 11: Generalization view of System Architecture

4.3. Algorithm Details

The project will implement two algorithms namely Haar Cascade and LBPH algorithms. Haar cascade algorithm is used for face detection and LBPH algorithm for face recognition. After successful review of the literature we concluded to make the use of Haar cascade and LBPH collaboratively for better accuracy and faster processing.

4.3.1 Haar Cascade Algorithm

The Haar Cascade algorithm is a machine learning-based approach used for object detection in images or video. It was proposed by Viola and Jones in their seminal 2001 paper titled "Rapid Object Detection using a Boosted Cascade of Simple Features."

The Haar Cascade algorithm is named after Haar wavelets, which are used as the basic features for detecting objects. It is based on the concept of cascades of classifiers, where each stage of the cascade progressively eliminates unlikely regions from consideration.

This Algorithm is based on a Machine Learning approach in which lots of images are used, whether positive or negative, to train the classifier [16].

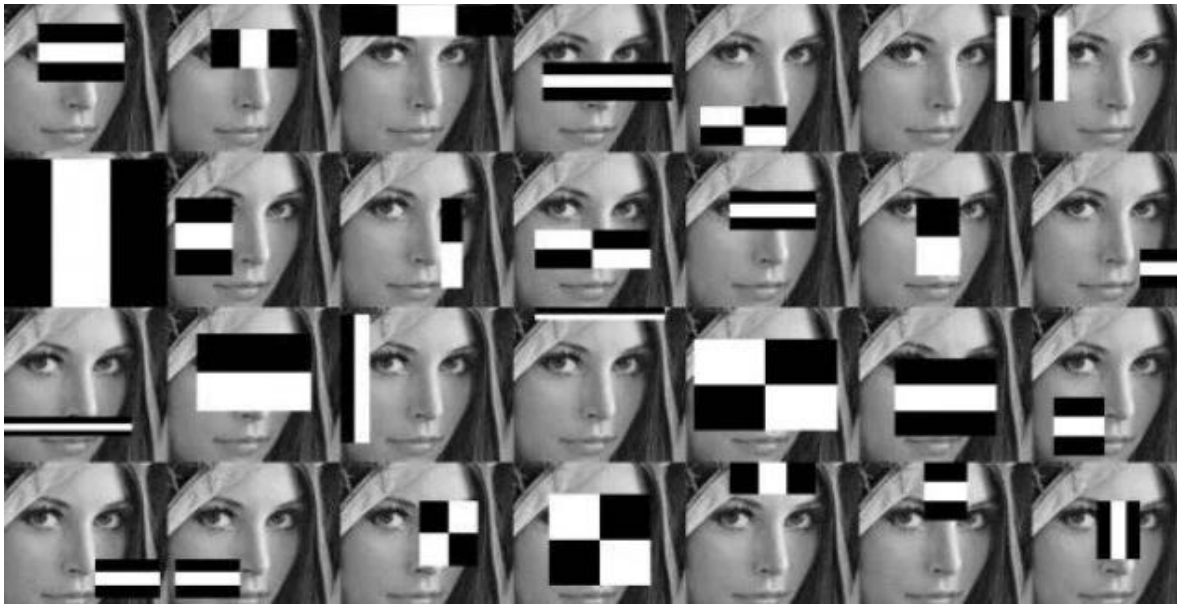


Figure 12: Haar-like feature applied

4.3.1.1. Haar Cascade Algorithm Explanation:

The Haar cascade algorithm is machine learning based object detection method used to identify and locate objects within images or video frames. It is widely used for tasks such as face detection, pedestrian detection, and object detection. The algorithm works by training a classifier using positive and negative samples. Positive samples contain images or examples of the object we want to detect, while negative samples contain images without the object. During the training process, the algorithm extracts Haar-features from these samples. Haar-like features are simple rectangular patterns that capture contrast differences in the image. They can represent various aspects like edges, lines, and textures. These features are calculated at different scales and positions in the image, creating a feature vector that is fed into a classifier. During the training phase, the algorithm iteratively selects the best features and adjusts their weights to accurately classify positive and negative samples. This process creates a strong classifier capable of distinguishing the object of interest from the background. Once the classifier is trained, it can be used to detect the object in new images or video frames. This is done by sliding a window of different sizes across the image and applying the classifier to each window. If the window matches the object's characteristics, it is considered detection.

Advantages

- Efficiency: It efficiently rejects non-object regions, enabling real-time applications.
- Robustness: It handles variations in lighting, object orientations, and backgrounds.
- Training: It learns from positive and negative samples to distinguish objects.
- Multiple Objects: It can detect multiple objects simultaneously.
- Pre-Trained Models: Pre-trained models are available for common objects.

4.3.2. Local Binary Patterns Histograms

Local Binary Pattern *(*LBP) is a simple yet very efficient texture operator which labels the pixels of an image by thresholding the neighborhood of each pixel and considers the result as a binary number.
It was first described in 1994 (LBP) and has since been found to be a powerful feature for texture classification. It has further been determined that when LBP is combined with histograms of oriented gradients (HOG) descriptor, it improves the detection performance considerably on some datasets. Using the LBP combined with histograms we can represent the face images with a simple data vector [17].

The steps of LBPH algorithm:

1. **Parameters**: the LBPH uses 4 parameters:

- **Radius**: the radius is used to build the circular local binary pattern and represents the radius around the central pixel. It is usually set to 1.
- **Neighbors**: the number of sample points to build the circular local binary pattern. Keep in mind: the more sample points you include, the higher the computational cost. It is usually set to 8.
- **Grid X**: the number of cells in the horizontal direction. The more cells, the finer the grid, the higher the dimensionality of the resulting feature vector. It is usually set to 8.

- **Grid Y**: the number of cells in the vertical direction. The more cells, the finer the grid, the higher the dimensionality of the resulting feature vector. It is usually set to 8.

2. Training the Algorithm: First, we need to train the algorithm. To do so, we need to use a dataset with the facial images of the people we want to recognize. We need to also set an ID (it may be a number or the name of the person) for each image, so the algorithm will use this information to recognize an input image and give you an output. Images of the same person must have the same ID. With the training set already constructed, let's see the LBPH computational steps.

3. Applying the LBP operation: The first computational step of the LBPH is to create an intermediate image that describes the original image in a better way, by highlighting the facial characteristics. To do so, the algorithm uses a concept of a sliding window, based on the parameter's **radius** and **neighbors**.

The image below shows this procedure:

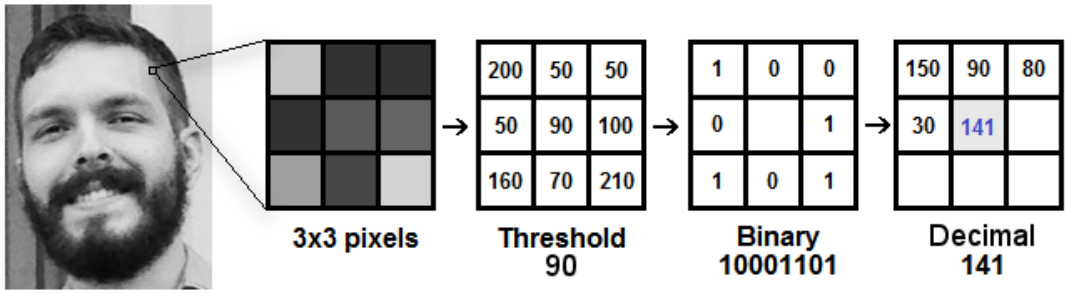


Figure 13: The matrix of the threshold of 3*3 of LBPH

Based on the image above, let's break it into several small steps so we can understand it easily:

- Suppose we have a facial image in grayscale.
- We can get part of this image as a window of 3x3 pixels.
- It can also be represented as a 3x3 matrix containing the intensity of each pixel (0~255).
- Then, we need to take the central value of the matrix to be used as the threshold.
- This value will be used to define the new values from the 8 neighbors.
- For each neighbor of the central value (threshold), we set a new binary value. We set 1 for values equal or higher than the threshold and 0 for values lower than the threshold.
- Now, the matrix will contain only binary values (ignoring the central value). We need to concatenate each binary value from each position from the matrix line by line into a new binary value (e.g. 10001101). Note: some authors use other approaches to concatenate the binary values (e.g. clockwise direction), but the final result will be the same.
- Then, we convert this binary value to a decimal value and set it to the central value of the matrix, which is actually a pixel from the original image.
- At the end of this procedure (LBP procedure), we have a new image which represents better the characteristics of the original image.
- **Note**: The LBP procedure was expanded to use a different number of radius and neighbors, it is called Circular LBP.

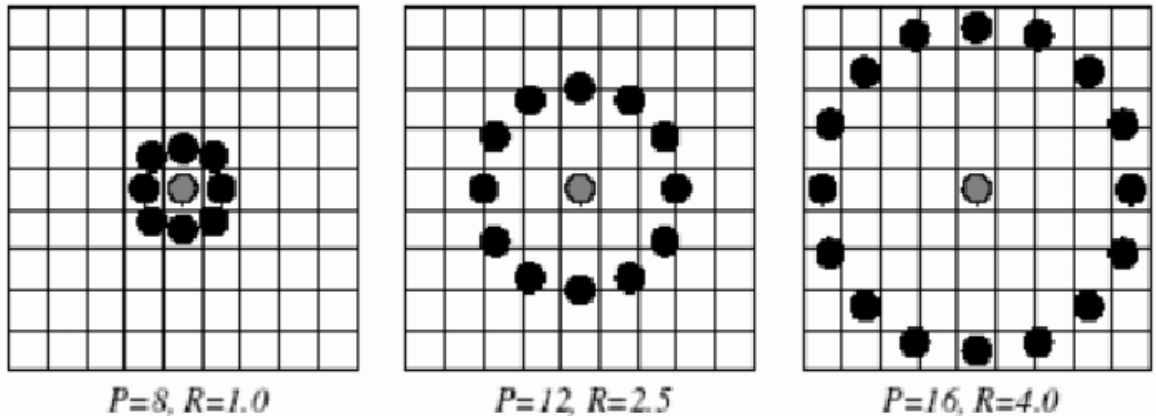


Figure 14: Circular LBP Operator

It can be done by using bilinear interpolation**.** If some data point is between the pixels, it uses the values from the 4 nearest pixels (2x2) to estimate the value of the new data point.

4. Extracting the Histograms: Now, using the image generated in the last step, we can use the Grid X and Grid Y parameters to divide the image into multiple grids, as can be seen in the following image:

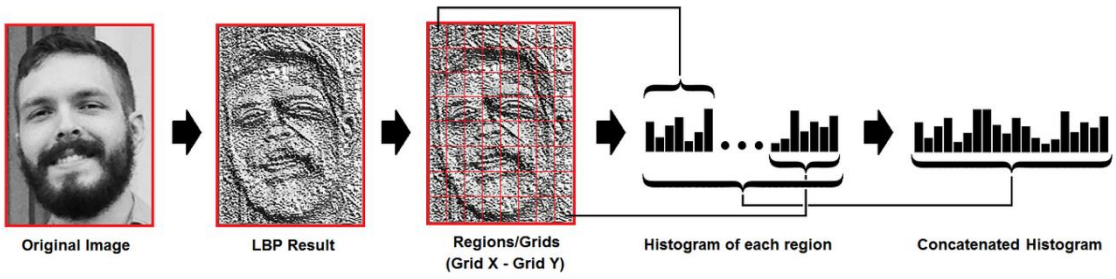


Figure 15: Extracting the Histogram from the sample

Based on the image above, we can extract the histogram of each region as follows:

- As we have an image in grayscale, each histogram (from each grid) will contain only 256 positions (0~255) representing the occurrences of each pixel intensity.

- Then, we need to concatenate each histogram to create a new and bigger histogram. Supposing we have 8x8 grids, we will have 8x8x256=16.384 positions in the final histogram. The final histogram represents the characteristics of the image original image.

The LBPH algorithm is pretty much it.

5. Performing the face recognition: In this step, the algorithm is already trained. Each histogram created is used to represent each image from the training dataset. So, given an input image, we perform the steps again for this new image and create a histogram which represents the image.

- So to find the image that matches the input image we just need to compare two histograms and return the image with the closest histogram.

- We can use various approaches to compare the histograms (calculate the distance between two histograms), for example: **euclidean distance**, **chi-square**, **absolute value**, etc. In this example, we can use the Euclidean distance (which is quite known) based on the following formula:

$$D = \sqrt{\sum_{i=1}^{n}(hist1_i - hist2_i)^2}$$

- So the algorithm output is the ID from the image with the closest histogram. The algorithm should also return the calculated distance, which can be used as a 'confidence' measurement.

- We can then use a threshold and the 'confidence' to automatically estimate if the algorithm has correctly recognized the image. We can assume that the algorithm has successfully recognized if the confidence is lower than the threshold defined [17].

Conclusions

- LBPH is one of the easiest face recognition algorithms.

- It can represent local features in the images.
- It is possible to get great results (mainly in a controlled environment).
- It is robust against monotonic gray scale transformations.
- It is provided by the OpenCV library (Open Source Computer Vision Library).

Chapter 5

Implementation and Testing

5.1. Implementation

The first step in the procedure is to train the system with the faces of the pupils whose attendance has to be recorded soon. Various faces are given many names throughout the system. Since light is a major factor in image processing, algorithms like Eigenface, Fisherface, and LBPH may be applied under a variety of lighting conditions. Additionally, as a high-quality camera is essential for face identification and recognition, it should be employed. Therefore, the higher the system's efficiency, the better the webcam being utilized.

Even if the documentation process is ongoing throughout the Lifecycle, formal emphasis is paid to it at the implementation stage. This type of document is created to present the knowledge that has been gathered about the system during its creation and application.

Ultimately, this stage made sure the system met all the requirements and goals set forth in previous project stages. In the part that follows, the technologies and tools utilized to carry out this project are briefly covered.

5.1.1. Tools Used

Different tools and technologies have been used to implement this application. They are listed in the table below:

Table 2: Tools Used

Application Platform	Desktop Application
Programming Language	Python
Database Tools	MYSQL
IDE	Visual Studio Code
Python Libraries	Tkinter, OpenCV, Pillow, Numpy

5.1.2. Implementation Detail of Modules

5.1.2.1. Student Management Module

Table 3: Implementation Details of the Student Management Module

Purpose	This module is responsible for collecting facial images of individuals for training and identifications purposes.
Dependencies	None
Input	Webcam or camera device for capturing facial images
Output	Facial images in a standardized format (e.g. JPEG, PNG)

5.1.2.2. Preprocessing Module

Table 4: Implementation Details of the Preprocessing Module

Purpose	This module processes the collected facial images to enhance their quality and prepare them for recognition.
Dependencies	Student Management Module
Input	Facial images
Output	Preprocessed facial images

5.1.2.3. Face Detection Module

Table 5: Implementation Details of the Face Detection Module

Purpose	This module detects and localizes human faces within the preprocessed images.
Dependencies	Preprocessing Module
Input	Preprocessed facial images

Output	Coordinates of detected faces within the images

5.1.2.4. Face Recognition Module

Table 6: Implementation Details of the Face Recognition Module

Purpose	This module detects and localizes human faces within the preprocessed images.
Dependencies	Preprocessing Module
Input	Preprocessed facial images
Output	Coordinates of detected faces within the images

5.1.2.5. Attendance Management Module

Table 7: Implementation Details of the Face Recognition Module

Purpose	This module manages the attendance records based on the recognized identities.
Dependencies	Face Detection Module
Input	Recognized identities
Output	Attendance records with timestamps

5.1.2.6. User Interface Module

Table 8: Implementation Details of the User Interface Module

Purpose	This module provides a user-friendly interface for administrators and users to interact with the system.
Dependencies	Student Management Module, Attendance Management Module
Input	User commands and settings
Output	User commands and settings

5.1.2.7. Reporting Module

Table 9: Implementation Details of the Reporting Module

Purpose	This module generates reports based on the attendance records, allowing administrators to view and analyze attendance data.
Dependencies	Attendance Management Module
Input	Attendance records
Output	Attendance reports in various formats (e.g. CSV)

5.2. Testing

In application development, testing is the process of assessing software to make sure it satisfies specifications and operates as intended. It includes finding defects, confirming functioning, assessing performance, guaranteeing quality, testing security, and making sure the user experience is satisfactory. Before the application is deployed, problems are found and fixed through an iterative process called testing. It is essential to producing software that is dependable and of excellent quality. Finding software bugs is one of testing's main goals in order to identify and fix errors.

As a consequence, testing is done to ensure that project requirements are met, as well as to verify outcomes and report on system performance and procedure.

5.2.1. Test Cases for Unit Testing

Software application components or individual modules are tested separately during a process known as unit testing. Its main objective is to confirm the independent functioning and accuracy of every unit, usually at the code level. Unit tests are brief, automated tests that verify a unit operates as intended by examining certain inputs and outputs. Early defect detection, code restructuring, and the creation of modular and maintainable code are all facilitated by it. There is no table displaying the test cases for the unit testing since all of the unit testing for this system is done at the code level by the individual using the system. Rather, a tabulation of all the test cases is used for the system's functionality testing.

5.2.2. Test Cases for System Testing

System testing is a type of testing where the complete software system is examined to make sure all of its parts function properly and adhere to the prerequisites as stated. It assesses how the system behaves in various situations and determines whether or not it operates correctly in the intended setting. System testing confirms the functionality, security, dependability, performance, and integration of different modules, among other features of the system. It seeks to evaluate the system against the intended results and find flaws that could arise from interactions between various components [18].

Table 10: System Testing of an Attendance System

ID	Test Scenario	Test Case	Test Steps	Expected Result	Actual Result	True
T1	Verify Sign up Functionality	To check Whether the teacher(user) can be registered	(a)Open the Application (b) Click the register button (c) Fill the sign up form	User should be registered	User is registered successfully and redirected towards log in	True
T2	Verify Sign in Functionality	To check whether the person is signed in or not	(a) Open the app (b) Fill the sign in credentials	User should be signed in.	Home window is displayed	True
T3	Verify button functionality	To check whether all buttons in home window are clickable.	(a) Open the app (b) Click on different buttons	The user should be able to click onto different functionalities	User accesses different windows on application.	True
T4	Verify student management	To check whether students are	(a)Fill details from the	User should be able to	Student details	True

	system functionality	being registered or not	student details window	register students	added successfully	
T5	Verify student management system functionality	To check whether the students' sample are taken or not	(a)Click Take photo (b)Face the camera	System camera should open and samples must be collected	Student's samples collected successfully	True
T6	Verify student management system functionality	To check whether students sample trained or not	Click Train Data	All the samples should start training	All the samples are trained	True
T7	Verify student management system functionality	To check whether student's details can be updated or not	(a) Select any student from the table (b) Make the changes in the details (c) Click the update button	Student details should be updated	Student details are updated with details and stored in the table again	True
T8	Verify student management system functionality	To check whether student's details can be deleted or not	(a) Select any student from the table (b) Click the delete button	Student details should be deleted	Student details are deleted from the table and database	True

T9	Verify take attendance functionality	To check whether the new window of check in and check out is open	Click the Take attendance button	The new window containing two buttons should be open	The new window with two buttons is opened	True
T10	Verify Check in button functionality	To check whether the camera opens, detects and recognizes the person	Click the Check in button	The camera should be opened and the person facing camera should be detected and attendance should be marked	The camera is opened and the person facing camera is detected and attendance is marked	True
T11	Verify Check out button functionality	To check whether the camera opens, detects and recognizes the person	Click the Checkout button	The camera should be opened and the person facing camera should be detected and attendance of checked out should be marked	The camera is opened and the person facing camera is detected and attendance of checked out is marked	True
T12	Verify Photos button functionality	To check whether the buttons works to open the	Click on the Photos button.	The captured samples should be shown	The captured samples are shown	True

		folder that contains captured samples				
T13	Verify Show Report button	To check whether the attendance data of month of a student is shown when button is clicked	Click on the Show Report button	The monthly attendance data of student should be shown.	The monthly attendance data of student is shown.	True
T14	Verify Fetch All Records button functionality	To check the data in database is exported to CSV/Excel file.	Click on the Fetch All Records button	The data of database related to attendance should be shown.	The data of database related to attendance is shown.	True

5.3. Result Analysis

After conducting Unit Testing for the abovementioned test cases, we determined that each individual component successfully passes the test with expected results. We then conducted System Testing to determining proper functionality of all the limited features of the application. For this we selected a data set, trained it and activated the system to get a desired result. With this we were able to find the strength and weakness of our application. The application was able to successfully detect the faces. Given ideal condition such as proper lighting, absence of shadows and plain background, the more accurate result can be expected. The application showed the correct result of the image which was taken for prediction. Likewise, different images were taken for prediction in which result was accurate for some image while some images showed incorrect result. Regression testing was also performed to verify every feature of the application. The Buttons clicking

functionalities, camera functionalities, image gallery functionalities all of these tests were carried out in system testing. The overall running of the application can be characterized as smooth and functional. However, it should be noted that the smoothness and fluidity of the application depend on the hardware of the system. Sometimes the application crashed because the integrated model takes more time to load. Also, during the hit and trial phase of our system for increasing the accuracy, we tried taking the 1000 samples of the student but our system was being crashed continuously so we concluded it to be technically not feasible. Similarly, the results of the face recognition were less accurate on taking 50-70 samples. So, taking the 500 samples was the standard fit for our project to be technically feasible along with high accuracy and effectiveness.

Thus, the aim of this project is to capture the video of the students, convert it into frames, and relate it with the database to ensure their presence or absence, mark attendance to the particular student to maintain the record. The Automated Classroom Attendance System helps in increasing the accuracy and speed ultimately achieve the high precision real-time attendance to meet the need for automatic classroom evaluation.

Chapter 6

Conclusion and Future Recommendations

6.1. Conclusion

This system has been proposed for maintaining the attendance record. The main motive behind developing this system is to eliminate all the drawbacks which were associated with manual attendance system. The drawbacks ranging from wastage of time and paper, till the proxy issues arising in a class will completely be eliminated. Hence, desired results with user friendly interface expected from the system is made. The efficiency of the system could also be increased by integrating various steps and techniques in the future developing stages of the system. Eventually, the system is able to meet our desired objectives which are mentioned as follows:

- The face segment from the camera frame is detected.
- The useful features from the face detected are extracted
- The features to recognize the face detected are classified.
- The attendance of the identified student is recorded.
- The attendance details for 30 days of a student is generated.

6.2. Future Recommendations

Although this system delivers what it needs to deliver, the accuracy for facial attendance can be greatly improved. Powerful machines can be used to train the data and make the model much more efficient. It isn't possible for every device to be equipped with a web-camera, where we can still work upon to make the running of facial attendance system seamless with higher efficiency. The face recognition window can be made to run in the background without being displayed in the screen itself. These are few recommendations for this system to be made much more efficient by maintaining greater accuracy.

6.3. Summarizing the project

The face authentication attendance system is a project aimed at automating attendance tracking using facial recognition technology. It involves capturing video, converting it into frames, matching faces with a database, and providing real-time attendance updates. The system underwent Unit Testing and System Testing, showing strengths in face detection under ideal conditions but weaknesses in varying conditions. Regression testing ensured overall functionality and the project's success reflects thorough research and team efforts despite challenges in coordination and remote work.

6.4. Self-Reflection throughout the project

6.4.1. Things we learned

- The ability to write a good project report.
- The capacity for critical thought.
- The capacity for time management.
- Proficiency with libraries and other programming frameworks.
- Capability to utilize python to construct AI and ML.
- Appropriate testing and implementation methodology.
- The general SDLC structure.

6.4.2. Things yet to learn

- Proper way to communicate ideas.
- Accepting mistakes and being committed to finding a solution.
- Thinking about programming logics in a different way.
- Thinking about programming logics in a different way.

References

[1] M. &. P. Truk, "Eigenfaces," *Journal of Cognitive,* vol. 3, pp. 71-86, 1991.

[2] Y. W. H. &. Z. Y. Wang, "A review of face recognition for online attendance management in the era of big data," *Journal of Ambient Intelligence and Smart,* vol. 13, no. 1, pp. 1-17, 2021.

[3] R. &. D. M. Jayaswal, "Comparative analysis of human face recognition by traditional methods and deep learning in real-time environment," in *In 2020 IEEE 9th International Conference on Communication Systems and Network Technologies (CSNT,* 2020.

[4] A. Borgeaud, "https://www.statista.com/," 30 August 2023. [Online]. Available: https://www.statista.com/statistics/1153970/worldwide-facial-recognition-revenue/. [Accessed 15 March 2024].

[5] P. B. M. M. T. &. R. R. Kamencay, "A new method for face recognition using convolutional neural network.," *Advances in Electrical and Electroic Engineering,* vol. 15, no. 4, pp. 63-72, 2017.

[6] G. D. J. S. &. P. D. Anitha, ""Face Recognition Based Attendance System Using Mtcnn and Facenet," *Zeichen Journal,* vol. 6, no. 1, pp. 89-95, 2020.

[7] B. &. K. P. Gornale, "Classroom Attendance Management System Using camera," *International Journal of Research in Engineering, Science,* vol. 8, no. 8, pp. 27-30, 2020.

[8] M. A. I. M. I. H. M. A. Khan, "Face Recognition Using Local Binary Patterns Histograms (LBPH) with Improved Illumination and Pose Robustness.," *IEEE Access,* vol. 2, no. 2, pp. 1188-1197, 2014.

[9] S. &. B. Onur, "Face Recognition Using Principal Component Analysis and Viola-Jones Algorithm," in *in International Conference on Computer Science and (UBMK),* Bursa, Turkey, 2016.

[10] A. &. J. A. Omari, "Attendance Tracking Systems:A Review of the Literature," *International Journal of Advanced Computer Science and Application,* vol. 11, no. 4, pp. 194-202, 2020.

[11] A. D. M. &. K. A. Hairi, "Feasibility Study of a Proposed System for the Detection of Landmines," *IEEE Transactions of System, Man, and Cybernetics Part A: Systems and Humans*, vol. 27, no. 4, pp. 631-638, 1997.

[12] B. Jones, "How to Choose the Right Attendance Tracking System," *Forbes*, 2021.

[13] E. a. H. S. K. S. &. S. SAI, "https://ssrn.com/," 2021. [Online]. Available: https://ssrn.com/abstract=3851056. [Accessed 2024].

[14] O. A. R. R. F. a. W. A. B. Salim, "Class attendance management system using face recognition," in *2018 7th International Conference on Computer and communication*, 2018.

[15] A. R. S. A. S. N. a. M. G. Siswanto, "mplementation of face recognition algorithm for biometrics based time attendance system.," in *2014 International Conference on ICT For Smart Society (ICISS)*, 2014.

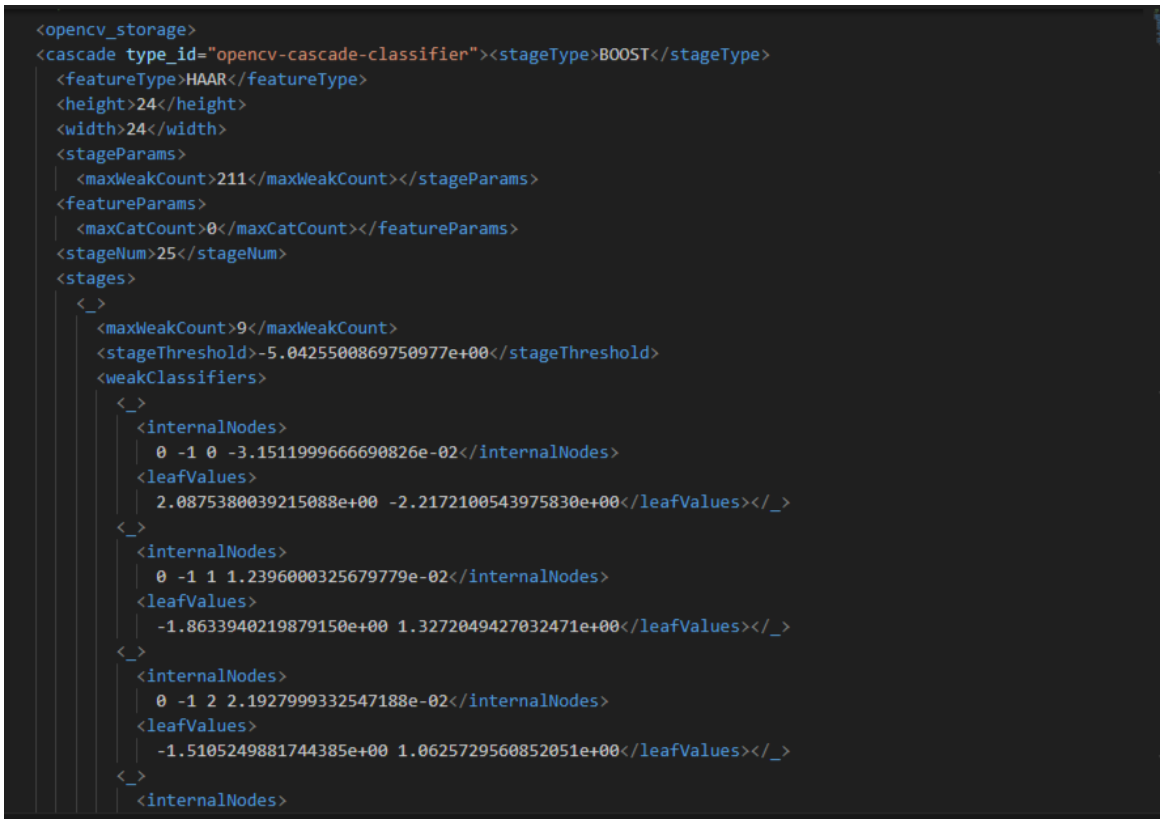
[16] Javatpoint, "www.javatpoint.com," [Online]. Available: https://www.javatpoint.com/haar-cascade-algorithm. [Accessed 2024].

[17] K. S. d. Prado, "https://towardsdatascience.com/," 11 November 2017. [Online]. Available: https://towardsdatascience.com/face-recognition-how-lbph-works-90ec258c3d6b. [Accessed 15 March 2024].

[18] A. B. B. P. B. P. G. K. T. S. C. Phul Babu Jha, "An Automated Attendance System Using Facial Detection and Recognition Technology," *Apex Journal of Business and Management*, vol. 1, no. 1, pp. 103-120, 2023.

Appendices

Face Authentication Attendance System Source Code Snippets



```
<opencv_storage>
<cascade type_id="opencv-cascade-classifier"><stageType>BOOST</stageType>
  <featureType>HAAR</featureType>
  <height>24</height>
  <width>24</width>
  <stageParams>
    <maxWeakCount>211</maxWeakCount></stageParams>
  <featureParams>
    <maxCatCount>0</maxCatCount></featureParams>
  <stageNum>25</stageNum>
  <stages>
    <_>
      <maxWeakCount>9</maxWeakCount>
      <stageThreshold>-5.0425500869750977e+00</stageThreshold>
      <weakClassifiers>
        <_>
          <internalNodes>
            0 -1 0 -3.1511999666690826e-02</internalNodes>
          <leafValues>
            2.0875380039215088e+00 -2.2172100543975830e+00</leafValues></_>
        <_>
          <internalNodes>
            0 -1 1 1.2396000325679779e-02</internalNodes>
          <leafValues>
            -1.8633940219879150e+00 1.3272049427032471e+00</leafValues></_>
        <_>
          <internalNodes>
            0 -1 2 2.1927999332547188e-02</internalNodes>
          <leafValues>
            -1.5105249881744385e+00 1.0625729560852051e+00</leafValues></_>
        <_>
          <internalNodes>
```

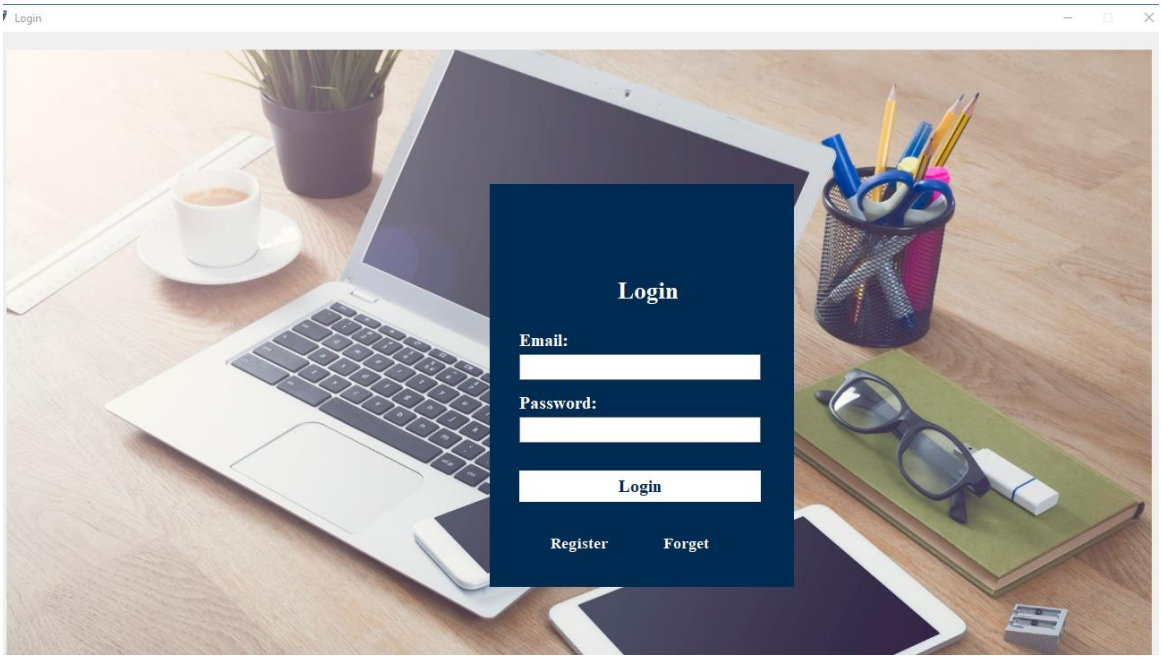
Snapshot of Haarcascade Frontal Code

```python
def face_recog(self):
    faceCascade = cv2.CascadeClassifier("haarcascade_frontalface_default.xml")
    clf = cv2.face.LBPHFaceRecognizer_create()
    clf.read("Classifier.xml")

    video_cap = cv2.VideoCapture(0)

    while self.window_open:
        ret, img = video_cap.read()
        if not ret:
            break
        gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
        faces = faceCascade.detectMultiScale(gray, scaleFactor=1.1, minNeighbors=5, minSize=(30, 30))

        # Flag to track if a recognized face is found
        face_found = False

        for (x, y, w, h) in faces:
            cv2.rectangle(img, (x, y), (x + w, y + h), (0, 255, 0), 2)
            id, predict = clf.predict(gray[y:y + h, x:x + w])
            confidence = int((100 * (1 - predict / 300)))

            if confidence > 77:
                face_found = True  # Set the flag to indicate a recognized face is found

                self.my_cursor.execute("Select `Student Name` from student where `Student ID`=" + str(id))
                n = self.my_cursor.fetchone()
                n = "+".join(n)
```

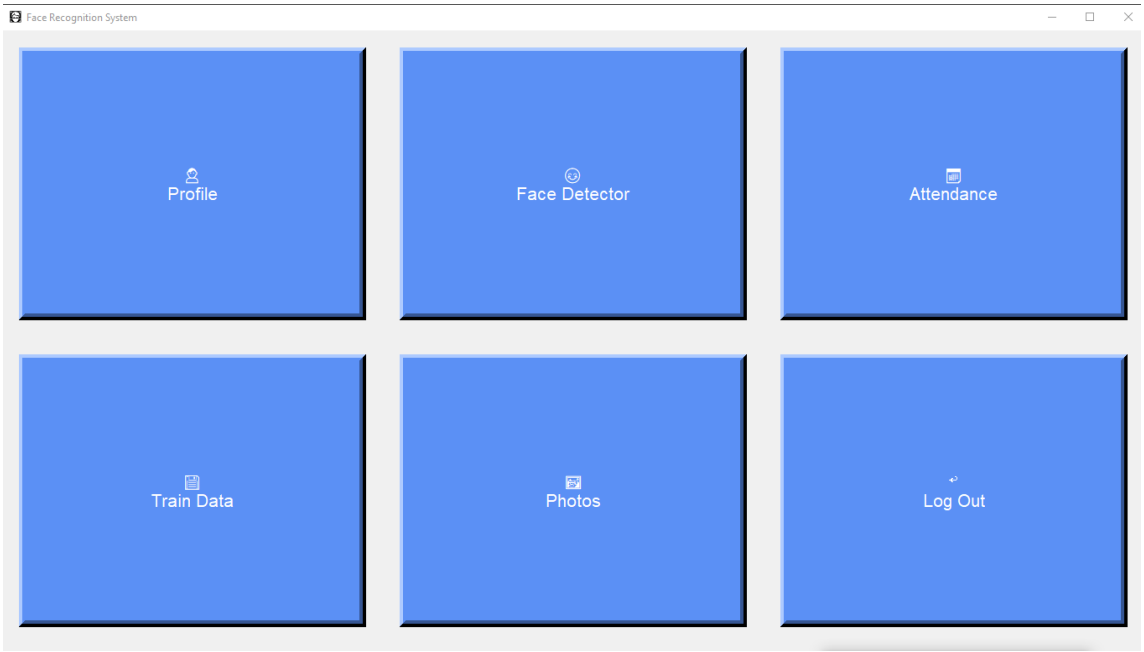
Snapshot of Local Binary Pattern Histogram

System UI Snippets

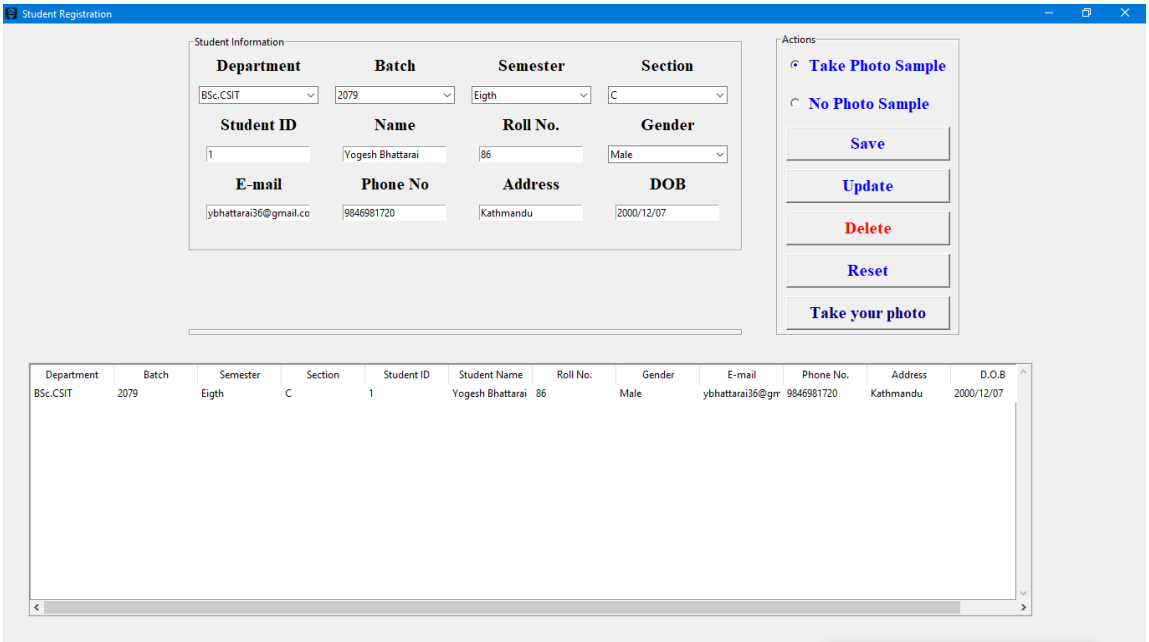
Snapshot of Main Window



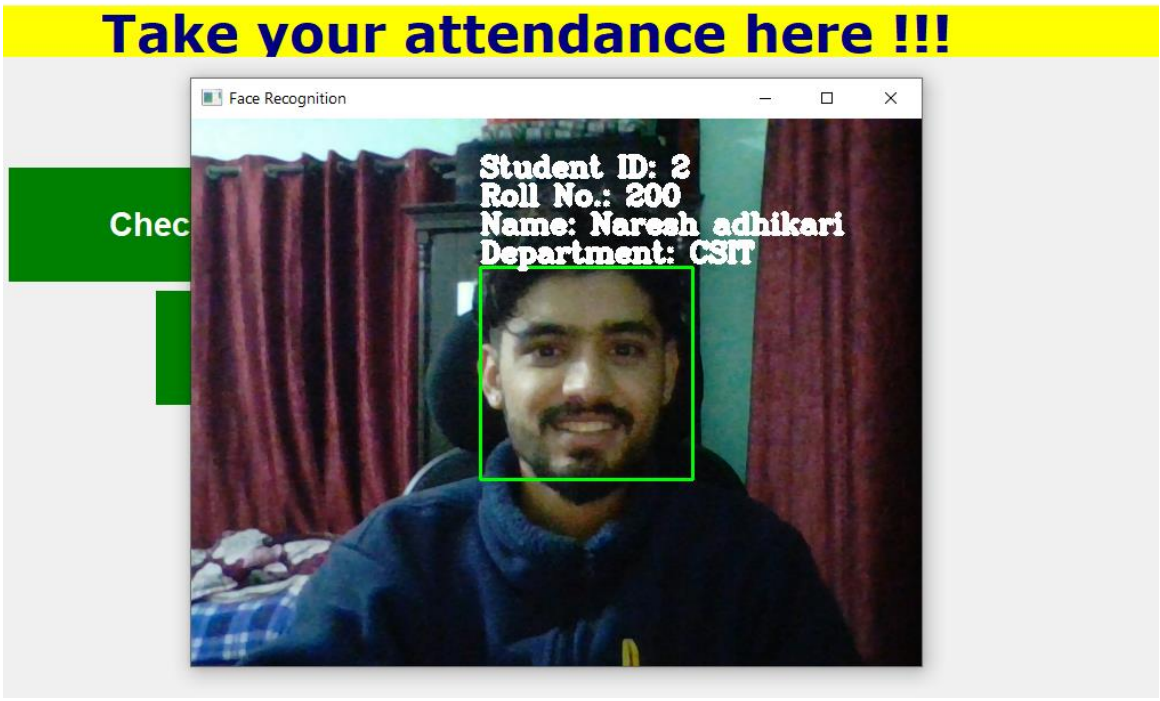
Snapshot of Login



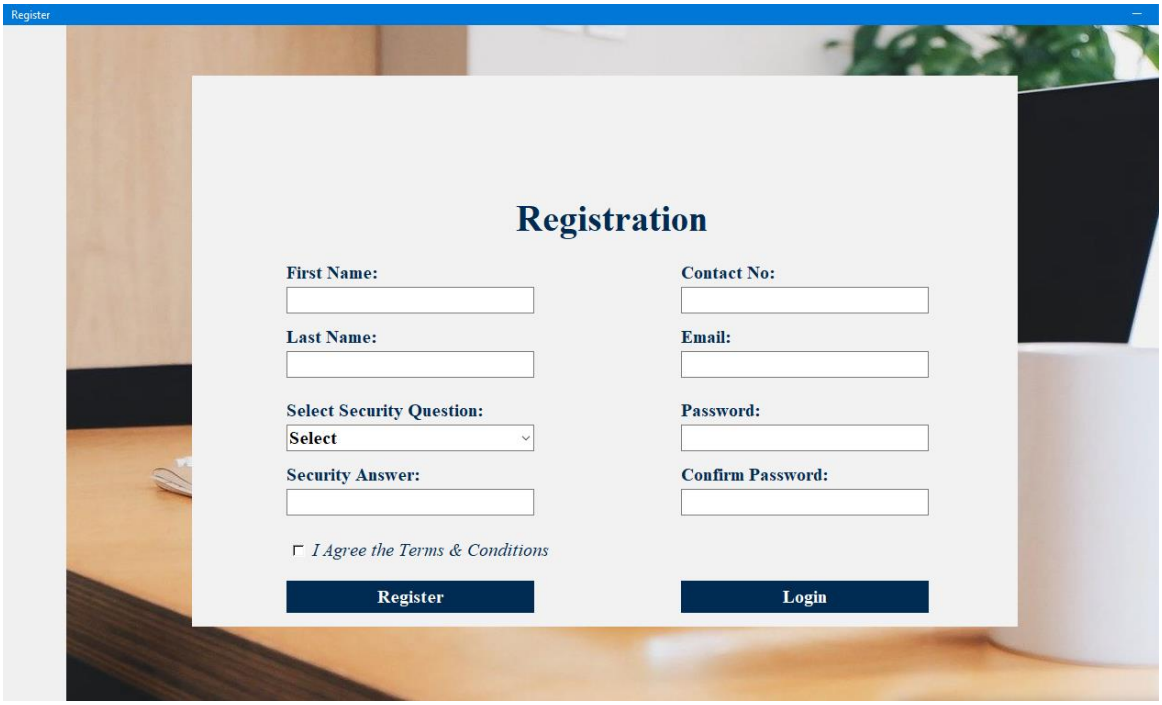
Snapshot of Functional Buttons



Snapshot of Student Profile

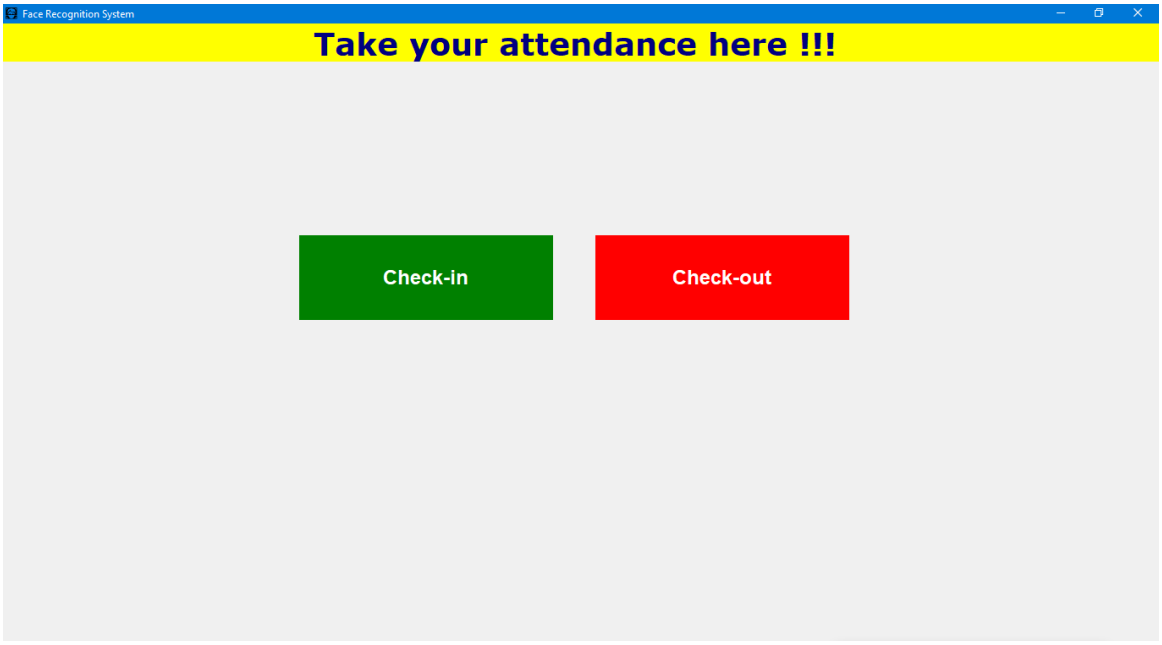


Snapshot of Face Detection

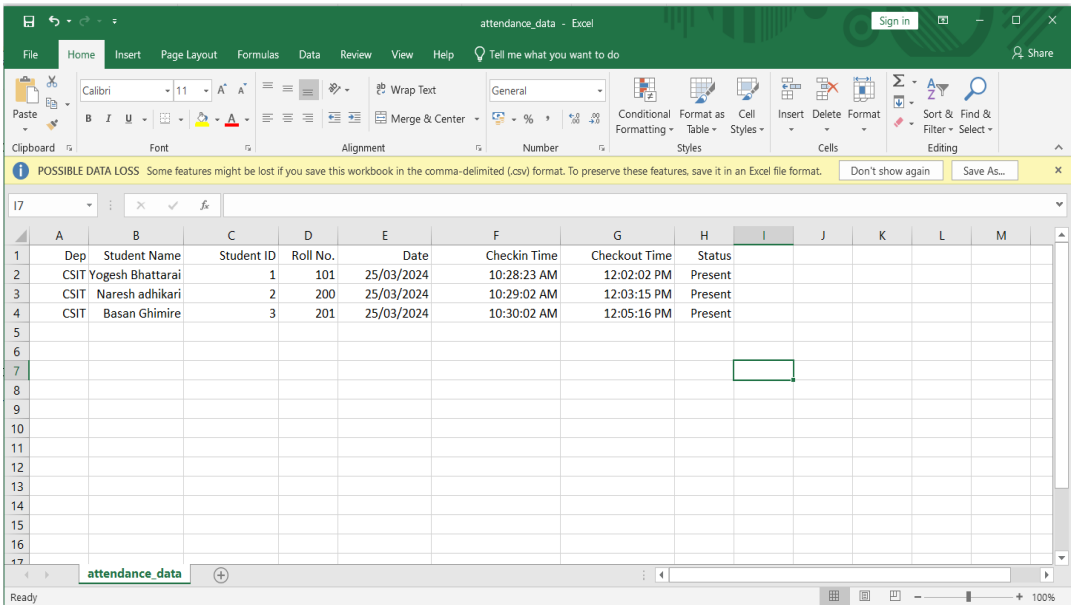


Snapshot of Admin Registration

Snapshot of Train Data



Snapshot of Take Attendance



Dep	Student Name	Student ID	Roll No.	Date	Checkin Time	Checkout Time	Status
CSIT	Yogesh Bhattarai	1	101	25/03/2024	10:28:23 AM	12:02:02 PM	Present
CSIT	Naresh adhikari	2	200	25/03/2024	10:29:02 AM	12:03:15 PM	Present
CSIT	Basan Ghimire	3	201	25/03/2024	10:30:02 AM	12:05:16 PM	Present

Snapshot of Attendance Data