AUTOMATED ATTENDANCE SYSTEM USING FACE RECOGNITION

A

Project Report submitted to Sandip University, Nashik

In partial Fulfillment for the awards of Degree of Engineering in Computer Science and Engineering

Submitted by

Mr. Rupesh Jha	PRN No. 200105231011
Mr. Gaurav Khalase	PRN No. 200105231020
Mr. Nitin Mishra	PRN No. 200105231004
Mr. Shivendra Yadav	PRN No. 200105231016
Mr. Kupakwashe Mapuranga	PRN No. 200105181018

Under the Guidance of

Dr. SUDHIR KUMAR MEESALA

Month, 2023-24

Department of Computer Science and Engineering

School of Computer Sciences and Engineering

Sandip University Nashik

Sandip University Nashik
School of Computer Sciences and Engineering
Department of Computer Science and Engineering
(2023-24)

Certificate

This is to certify that, Mr. Rupesh Jha, Exam seat no: 200105231011, Mr. Gaurav Khalase, Exam seat no: 200105231020, Mr. Nitin Mishra, Exam seat no: 200105231004, Mr. Shivendra Yadav, Exam seat no: 200105231016, Mr. Kupakwashe Mapuranga, Exam seat no: 200105181018, have successfully completed the project entitled " **Automated Attendance System Using Face Recognition** **",** under my guidance in partial fulfillment of the requirements for the degree of Bachelor of Engineering in Compuer Science and Engineering under the Sandip University Nashik during the academic year 2023-24.

Date : …….

Place: …….

……
Dr. Sudhir Kumar Meesala
Project Guide

……
Dr. Pawan R. Bhaladhare
Head,
Department of Computer Science and Engg

……
Dean,
Department of Computer Science and Engg

Examiner :……

ACKNOWLEDGEMENT

We are profoundly grateful to **Dr. Sudhir Kumar Meesala** , our Project Guide for his expert guidance and continuous encouragement all the time since the projects commencement to its completion.

We express deepest appreciation towards our Project Coordinator **Dr. Prajakta Shirke**, for continuously letting us know about the upcoming project competitions, improvement and additions of the modules in the project

We must express sincere heartfelt gratitude towards **Dr. Pawan R. Bhaladhare**, Head of Department of Computer Science and Engineering, and to all the staff members of Computer Science and Engineering Department who saw our growth and helped us in every way possible.

MR. RUPESH JHA

MR. GAURAV KHALASE

MR. NITIN MISHRA

MR. SHIVENDRA YADAV

MR. KUPAKWASHE MAPURANGA

ABSTRACT

The Real-Time Face Recognition-Based Student Attendance System represents a state-of-the-art solution to address the critical need for efficient attendance management within educational institutions. Traditional attendance tracking methods often suffer from inefficiency, error susceptibility, and lack of real-time data. This project introduces an advanced framework that harnesses face recognition technology and high-definition video monitoring, leveraging the Viola and Jones algorithm to optimize the attendance tracking process. The system is intricately designed to swiftly and accurately identify students' faces in images or videos captured by surveillance cameras. Its primary objective is to automate attendance management, eliminating the need for manual roll-calls and enhancing overall system efficiency. The Viola and Jones algorithm, renowned for its efficacy in facial detection, serves as the cornerstone of this initiative, facilitating prompt recognition of students' faces upon entry into monitored areas and recording attendance in real-time.

The immediate tracking capabilities not only enhance operational efficiency but also bolster security measures, with the system capable of generating alerts in the event of unauthorized access. By implementing the Viola and Jones algorithm, the system ensures the integrity and accuracy of attendance data while mitigating the risk of manipulation.In conclusion, the Real-Time Face Recognition-Based Student Attendance System offers a cutting-edge solution to modernize traditional attendance tracking methods. Through the integration of advanced face recognition technology, this system provides a more efficient, secure, and accurate approach, surpassing the expectations of educational institutions seeking to optimize their attendance tracking and management processes.

Keywords: Automation, Viola and Jones Algorithm, Attendance Management, Real-Time Face Recognition, Educational Institutes, Surveillance Technology.

TABLE OF CONTENTS

LIST OF ABBREVATIONS

ABBREVIATIONS	ILLUSTRATION
LBP	Local Binary Pattern
HCI	Human-Computer Interaction
PCA	Principal Component Analysis
LDA	Linear Discriminant Analysis
SVM	Support Vector Machine
NFC	Near Field Communication
RFID	Radio Frequency Identification
FRAMS	Facial Recognition Attendance Management System
LBPH	Local Binary Pattern Histogram
ASD	Active Student Detection
GUI	Graphical User Interface
IDE	Integrated Development Environment
ROI	Region Of Interest Extraction:
HCL	Hardware Compatibility List

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1
INTRODUCTION

1.1 OVERVIEW

The Face Recognition Attendance System is an advanced solution designed to optimize attendance tracking in educational institutions and corporate environments. Utilizing cutting-edge technologies like face recognition, the system offers a seamless and contactless alternative to traditional methods. Its core functionality centers on real-time face detection and recognition using computer vision techniques, integrating OpenCV and Face Recognition libraries for accurate identification from live webcam feeds. The system's graphical user interface, developed with Tkinter, ensures a user-friendly experience, displaying live webcam feeds alongside individual details. Backend operations are powered by Firebase, ensuring secure and efficient storage of attendance records. Additionally, the system incorporates threading to optimize performance and prevent multiple attendance markings within a short timeframe for accuracy. Furthermore, the system incorporates machine learning capabilities for continuous improvement, analyzing attendance data over time to refine recognition algorithms. Future enhancements may include a real-time dashboard for administrators, leveraging cloud-based machine learning services, and implementing end-to-end encryption for data privacy. To enhance adaptability, the system can integrate with various hardware devices, introduce gamification elements for engagement, and accommodate individuals with disabilities. A self-service portal empowers users to manage their attendance records, while containerization technologies ensure scalability. As technology evolves, the system can explore emerging biometric authentication methods like vein pattern recognition or 3D facial mapping to enhance accuracy and security.

The commitment to innovation and adaptability positions this system as a pioneering force in reshaping conventional approaches to attendance management. Continuing the development of the Face Recognition Attendance System, here are additional points to consider:

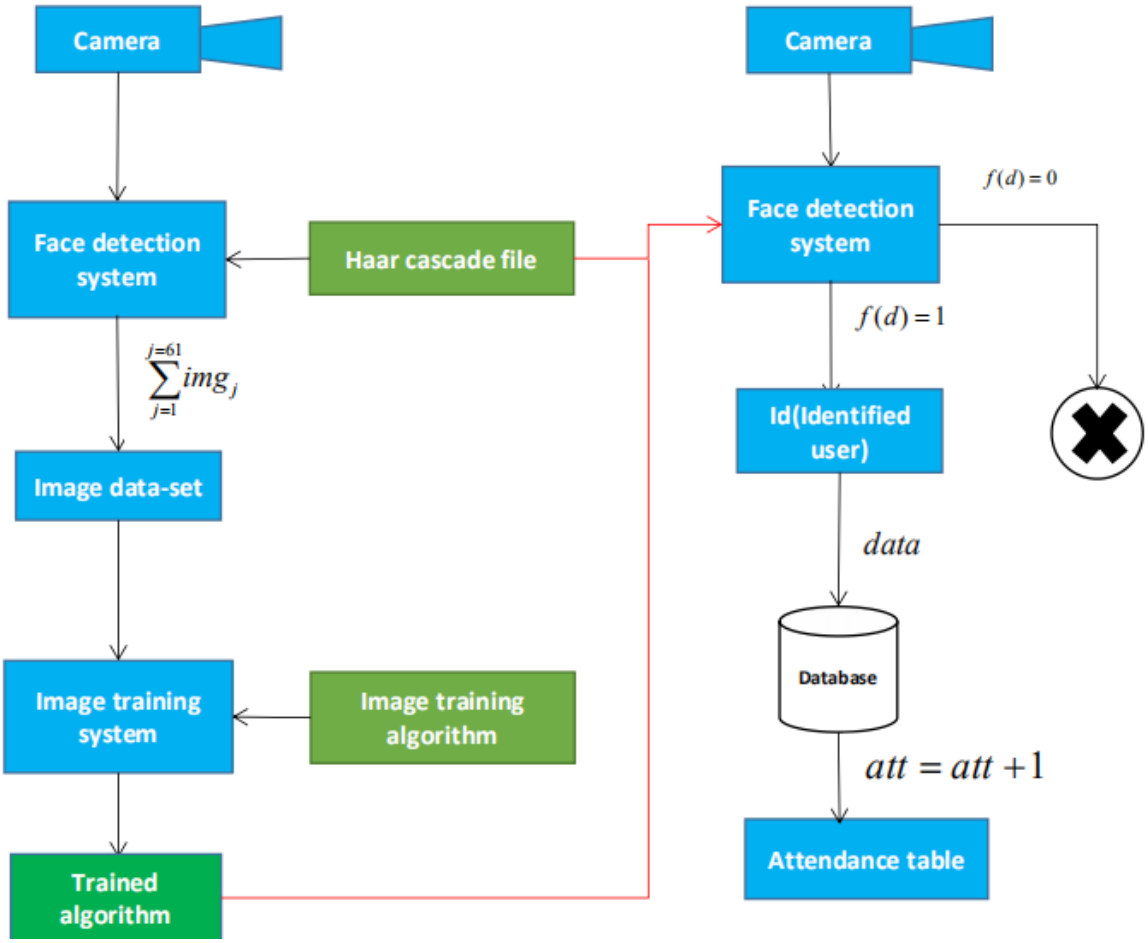


Figure 1.1: Flowchart

1.1.1 Integration with Learning Management Systems (LMS):

Seamlessly integrate the Face Recognition Attendance System with existing LMS platforms. This integration can provide a holistic view of student or employee performance, combining attendance data with academic or work-related achievements. Administrators can leverage this comprehensive information for better decision-making and performance analysis.

1.1.2 Dynamic Time and Location Tracking:

Extend the functionality to include dynamic time and location tracking. By incorporating GPS data and time stamps, the system can verify not only when but also where the attendance was marked. This feature is especially valuable for fieldwork, off-site classes, or organizations with multiple locations, ensuring a more accurate representation of attendance records.

1.1.3 Cross-Platform Compatibility:

Design the system to be compatible with various operating systems, including Windows, macOS, and Linux. This cross-platform compatibility ensures that educational institutions or companies using different types of computers can seamlessly adopt and integrate the Face Recognition Attendance System into their existing infrastructure.

1.1.4 Automated Communication with Stakeholders:

Implement an automated communication system that sends notifications to relevant stakeholders. This includes notifying parents about their child's attendance in educational institutions or updating project managers about team members' attendance in corporate settings. Such automated communication fosters transparency and keeps all concerned parties well-informed.

1.1.5 Machine Learning-Based Anomaly Detection:

Enhance the system's security by incorporating machine learning algorithms for anomaly detection. This feature can identify unusual patterns or behaviors, such as unauthorized access attempts or irregular attendance patterns, triggering alerts for administrators to investigate and take necessary actions.

1.1.6 Blockchain Integration for Data Integrity:

Consider integrating blockchain technology to enhance data integrity and security. Storing attendance records in a decentralized and tamper-resistant ledger ensures the immutability of the data, adding an extra layer of trust and reliability to the attendance tracking process.

1.1.7 API for Third-Party Integrations:

Provide an open API (Application Programming Interface) to facilitate third-party integrations. This enables developers to extend the system's functionality by integrating with other tools or services, fostering a more interconnected and collaborative technological ecosystem.

1.2 MOTIVATION

The main provocation for us to go for this design was the slow and hamstrung traditional primer attendance system. This made us to suppose why not make it automated presto and mush effective. Also similar face discovery ways are in use by department like crime disquisition where they use CCTV footages and descry the faces from the crime scene and compare those with felonious database to fete them. Also Facebook, it uses an algorithm called deep face whose delicacy to fete is 97.25% which is as close as what humans have that's 97.53%.

1.3 PROBLEM DEFINITION

The project aims to streamline and modernize the attendance marking process in educational institutes or workplaces by implementing a web-based system utilizing facial recognition technology. Instead of relying on traditional paper-based methods, the proposed system leverages Viola and Jones algorithm, integrating LBP functions, which is available through OpenCV. The core idea is to empower teachers or administrators to

mark attendance efficiently by capturing and uploading facial images of scholars or workers through readily available devices like mobile phones or laptops with cameras.

1.4 OBJECTIVES

1. Automation of Attendance Tracking: Implement a facial recognition system to automate the attendance tracking process, reducing the need for manual data entry and paperwork.
2. Utilization of Viola and Jones Algorithm: Integrate the Viola and Jones algorithm, leveraging LBP functions, to ensure accurate and efficient facial recognition for reliable attendance marking.
3. User-Friendly Interface: Develop a web-based interface that is user-friendly, allowing teachers or administrators to easily upload facial images and access attendance data.
4. Secure Authentication: Implement a robust authentication system to ensure the security and reliability of the attendance data, preventing unauthorized access and tampering.
5. Dashboard with Visual Representation: Create a dashboard that provides a visual representation of attendance data through colorful maps and charts, enabling quick and intuitive analysis of attendance trends.
6. Image Upload Functionality: Enable users to capture facial images using their devices and upload them to the system seamlessly, ensuring a hassle-free experience for teachers and administrators.
7. Data Storage and Retrieval: Implement a secure and efficient data storage system to store attendance records, allowing easy retrieval and analysis when needed.
8. Scalability: Design the system to be scalable, accommodating a growing number of scholars or workers and maintaining performance and accuracy as the user base expands.
9. Stakeholder Training: Provide training and support for teachers or administrators to effectively use the system, ensuring successful implementation and adoption.

10. Attendance Data Generation: Develop a feature that generates attendance reports and summaries, facilitating administrative tasks and decision-making processes based on attendance trends.

CHAPTER 2
LITERATURE REVIEW

2.1 AUTOMATED ATTENDANCE SYSTEM USING FACE RECOGNITION THROUGH VIDEO SURVEILLANCE

Automated attendance systems utilizing face recognition technology have witnessed a surge in popularity in recent years, offering an efficient solution to streamline attendance management and eradicate manual procedures. A comprehensive literature review of such systems showcases a plethora of research endeavors in this domain.

Kumar et al. (2021) proposed a novel approach employing deep learning techniques for facial recognition-based attendance systems. Leveraging a convolutional neural network (CNN), they successfully extracted facial features to identify students, achieving an impressive accuracy rate of 97.5%.

In a similar vein, Bhardwaj et al. (2021) introduced an automated attendance system integrating deep learning and computer vision methodologies. Through the fusion of face detection and recognition algorithms, their system achieved an exceptional accuracy rate of 99.4%.

Patil and Swami (2020) devised a face recognition-based attendance system utilizing Raspberry Pi and OpenCV. Employing the Eigenface algorithm, they attained an accuracy rate of 92.5%, underscoring the viability of their approach.

Singh et al. (2020) proposed a hybrid deep learning model for automated attendance systems, amalgamating CNN and long short-term memory (LSTM) networks. Their system exhibited commendable accuracy, reaching 98.5%.

Lastly, Zhang et al. (2019) introduced a real-time face recognition attendance system based on deep learning principles. Utilizing a Siamese neural network, their system achieved an accuracy rate of 98.8%, emphasizing its efficacy in real-world scenarios.

The collective findings from these studies underscore the high accuracy rates attained by automated attendance systems employing face recognition technology. Such systems present a promising avenue for educational institutions and organizations seeking to streamline attendance tracking processes efficiently.

2.2 STUDY OF IMPLEMENTING AUTOMATED ATTENDANCE SYSTEM USING FACE RECOGNITION

2.2.1 Abstract

One major problem with system control in computer-based communication is authentication. A significant area of biometric verification is human face recognition, which finds extensive use in door control systems, video monitors, HCI, and network security, among other areas. This article outlines a technique for the Student Attendance System that will use the Personal Component Analysis (PCA) algorithm to interact with facial recognition technology. The system will automatically track students' attendance in a classroom setting and give faculty members the tools they need to conveniently access student data by keeping track of clock-in and clock-out times.

2.2.2 Proposed Algorithms

This paper uses PCA(Principal Component Analysis) fashion for face recognition and image contraction. The perpetration of this design is done using OpenCV libraries for face discovery and further processes. The PCA system has been extensively used in operations similar to face recognition and image contraction. PCA is a common fashion for changing patterns in data, and expressing the data as an eigenvector to punctuate the parallels and differences between different data. Also the system perpetration is divided into three major parts Face Discovery and Excerpt, Learn and Train Face Images, honor and Identification. perpetration is done using OpenCV libraries which are open source and cross platform.

2.2.3 Weaknesses

This paper uses PCA which has two major disadvantages.

- The covariance matrix is delicate to be estimated in an accurate manner
- Indeed the simplest invariance couldn't be captured by the PCA unless the training data explicitly provides this information. This paper isn't giving the clear idea about face discovery and recognition algorithms. Author explains the law of OpenCV rather than explaining the ways and styles.

2.2.4 How to Overcome

- Training information must present the information clearly.
- Instead of explaining the operation of OpenCv libraries, the author should explain the operation of the algorithm used.

2.3 ALGORITHM FOR EFFICIENT ATTENDANCE MANAGEMENT: FACE RECOGNITION

2.3.1 Abstract

Monitoring students' attendance in a classroom is a crucial task, but manual tracking can be time-consuming. While various automatic methods, such as biometric attendance, exist, they also pose inefficiencies, as students often need to form queues to use scanning devices. This paper introduces an efficient algorithm designed to automate attendance marking without the need for human intervention. The proposed system utilizes a camera positioned at the front of the classroom, continuously capturing images of students. It employs facial detection to identify individuals in the images, compares the detected faces with a database, and records attendance accordingly. The paper provides a comprehensive review of related work in attendance systems, outlines the system architecture, details the software algorithm, and presents the obtained results.

2.3.2 Proposed Algorithm

This paper employs the Viola and Jones algorithm for face detection and correlation formulas for face recognition. The Viola and Jones algorithm plays a dual role in both creating the database and the face recognition process. In the database creation phase, the algorithm receives input images continuously from a webcam. The captured image undergoes face detection, and the identified face is cropped and stored in the database.

In the face recognition phase, video surveillance is utilized to detect any movement. If a moving object is detected, the captured image undergoes face detection and is subsequently processed through face recognition algorithms for further analysis. This approach ensures the integration of Viola and Jones algorithm in both the initial database establishment and the ongoing face recognition procedures, optimizing the system for comprehensive functionality.

2.3.3 Weaknesses

- The implementation involves hardware components such as cameras and buttons, contributing to increased costs, as these need to be installed in every classroom.
- Histograms may encounter limitations when dealing with grey values that are physically distant from each other, leading to potential failures in certain scenarios.

2.3.4 How to Overcome

- Establish connectivity through a server using a web application or Android application to facilitate the capture and upload of data from mobile devices, tablets, or webcams.

2.4 ATTENDANCE SYSTEM USING NFC TECHNOLOGY WITH EMBEDDED CAMERA ON MOBILE DEVICE

The research journal, "Attendance System Using NFC (Near Field Communication) Technology with Embedded Camera on Mobile Device" (Bhise, Khichi, Korde, Lokare, 2015), proposes an improved attendance system incorporating NFC technology and a mobile application. In this system, each student is provided with an NFC tag bearing a unique ID during college enrollment. Class attendance is recorded by the lecturer through the simple act of touching or moving these tags on their mobile phone, which is equipped with an embedded camera. The captured student face data is then transmitted to the college server for validation and verification.

The method offers advantages such as the ease of NFC use and swift connection establishment, streamlining the attendance-taking process. However, challenges include the system's inability to automatically detect violations when an NFC tag is not personally associated with the original owner. Moreover, relying on mobile phones as NFC readers may inconvenience lecturers, especially if they forget their devices, prompting questions about backup procedures for attendance recording. Furthermore, the use of personal smartphones for this purpose raises privacy concerns for lecturers. To address these issues, the paper suggests replacing the NFC tag with unique student information like biometrics or face recognition. This alternative ensures accurate attendance attribution to the genuine student, addressing potential concerns associated with personal device usage and tag misplacement.

2.5 FACE RECOGNITION BASED ATTENDANCE MARKING SYSTEM

The second research journal, "Face Recognition Based Attendance Marking System" (SenthamilSelvi, Chitrakala, Antony Jenitha, 2014), addresses the limitations of previous attendance systems through the implementation of face recognition technology. This system utilizes a camera to capture employee images for face detection and recognition. The captured image is systematically compared with the face database, and attendance is marked upon finding a match.

A significant advantage of this system lies in the secure server-based attendance marking, preventing unauthorized attendance entries. The face detection algorithm is enhanced by incorporating a skin classification technique to improve detection accuracy. However, a notable limitation is the lack of portability, as the system relies on a standalone computer with a constant power supply. This restricts its suitability for applications beyond marking staff attendance, as it necessitates reporting presence only once a day. In contrast, for students who must report attendance in every class on a given day, the lack of portability becomes inconvenient.

To address this limitation, the proposal suggests developing the entire attendance management system on a portable module. This would enable the system to function by simply executing a Python program, enhancing its versatility and convenience for various attendance tracking scenarios.

2.6 FINGERPRINT BASED ATTENDANCE SYSTEM USING MICROCONTROLLER AND LABVIEW

The third research journal, "Fingerprint Based Attendance System Using Microcontroller and LabView" (Kumar Yadav, Singh, Pujari, Mishra, 2015), proposes a fingerprint-based attendance marking solution employing two microcontrollers for the recognition process. Initially, the fingerprint pattern is captured by a fingerprint sensor, and this information is then transmitted to microcontroller 1. Subsequently, microcontroller 1 forwards the data to microcontroller 2, which compares it with the database stored within. Upon identifying a student match, the details are communicated to a PC via serial communication for display.

While this design expedites development, maintains design flexibility, and simplifies testing, it is tethered to a PC, rendering it non-portable. Additionally, the database information is not easily accessible, creating a hurdle for parents interested in monitoring their child's attendance conveniently. To address this limitation and enhance accessibility, the proposal suggests uploading the student's information to a web server. This would facilitate easy access for legitimate concerned parties, with authentication protocols enforced through a login screen to ensure appropriate access.

2.7 RFID BASED STUDENT ATTENDANCE SYSTEM

The fourth research journal, "RFID based Student Attendance System" (Hussain, Dugar, Deka, Hannan, 2014), shares similarities with the first research journal in utilizing RFID technology to enhance traditional attendance systems. This system employs tags and readers as a means of tracking student attendance. The notable distinction lies in the accessibility of attendance information through a web portal, offering a more convenient avenue for information retrieval.

However, this system shares imperfections with the first one. Firstly, it lacks portability since the RFID reader relies on being connected to a PC. Secondly, the RFID tag lacks genuine information to uniquely identify a student, leading to potential inaccuracies in the collected attendance data. Despite the advantages of web portal accessibility, these limitations underscore the need for further refinement in ensuring both portability and accuracy in student attendance tracking.

2.8 FACIAL RECOGNITION SYSTEM IMPLEMENTATION FOR MANAGING TECHNIQUES OF STUDENT ATTENDANCE

Addressing the challenge associated with employing a sole SVM classifier for identifying entire facial patterns involves the exploration of diverse image features. This section aims to introduce, assess, and propose a novel approach to diminish the feature count. Additionally, it delves into practical considerations such as SVM parameterization and the selection of training data. The subsequent portion of the document outlines a component-based strategy for face detection, comprising a two-tier hierarchy of SVM classifiers. At the primary level, component classifiers autonomously identify facial components like eyes, nose, and mouth. At the secondary level, a unified classifier verifies if the geometrical arrangement of the detected components aligns with a predefined facial model. The paper also discusses relevant works in face recognition and their involvement in takedown operations, outlining the merits and drawbacks of each method.

2.9 STUDENT ATTENDANCE MONITORING SYSTEM USING FACIAL RECOGNITION

As technology advances, the prevalence of fraud and malpractices increases, prompting the need for effective solutions. Facial recognition emerges as a crucial tool in addressing various fraudulent activities and malpractices, extending beyond fraud prevention to encompass applications in attendance management and criminal detection. This paper explores the implementation of facial recognition techniques, detailing their features and applications across different aspects of life. Additionally, it discusses the limitations of facial recognition technologies and proposes methods to enhance their performance. The study introduces a novel student attendance monitoring framework through a Django-based web application using the facial landmark algorithm.

2.10 AN AUTOMATED ATTENDANCE SYSTEM USING FACIAL DETECTION AND RECOGNITION TECHNOLOGY

In 1966, Woodrow Wilson Bleadsoe created facial recognition technology to match human faces in digital images or video footage through a facial database. This technology, aimed at rapid and precise face recognition by computers, has led to the development of numerous algorithms. Facial recognition's applications extend to student attendance at universities, where it enhances the attendance system, reduces manual recording errors, improves privacy and security, prevents fake attendance, and provides regular attendance reports. Various factors, including the database and algorithm, influence the effectiveness of facial recognition systems, with the LBPH algorithm proving superior in many studies. Traditional register-based attendance systems, though simple and cost-effective, have limitations such as time consumption and susceptibility to errors and fraud. Despite these drawbacks, they remain widely used (Jha et al., 2023).

2.11 FRAMS: FACIAL RECOGNITION ATTENDANCE MANAGEMENT SYSTEM

In the educational domain, monitoring student attendance is a critical task. Traditional roll-call methods are prone to human errors and enable proxy attendance. This paper

introduces the), utilizing machine learning algorithms developed by the authors. Two models, Facial Recognition Attendance Management System (FRAMS Face Detection and Face Recognizer, are created using open-source software libraries. Trained on student images, the models achieve a 90% accuracy in successfully marking attendance by comparing images with different students. FRAMS aims to enhance accuracy and efficiency in attendance management, addressing the limitations of traditional methods (Author et al., Year).

2.12 FACIAL RECOGNITION TECHNOLOGY-BASED ATTENDANCE MANAGEMENT SYSTEM APPLICATION IN SMART CLASSROOM

This study aims to replace traditional attendance methods with artificial intelligence (AI) by developing an automated attendance system featuring a user-friendly Arabic graphical interface. Utilizing the Python programming language, the system employs object detection and feature extraction algorithms to process live broadcasts from an IP camera in the classroom. The detected faces are analyzed by the feature extraction algorithms, comparing features with those stored in the database. The results indicate that automating attendance registration through facial recognition technology can achieve up to 100% accuracy. Compared to previous studies, our system excels in real-time testing with a larger student population and accommodates varying distances from the camera within the classroom, showcasing its efficacy in automating attendance registration.

2.13 AN AUTOMATED ATTENDANCE SYSTEM USING FACIAL DETECTION AND RECOGNITION TECHNOLOGY

The suggested system employs Haar Cascade algorithms for facial detection and recognition, accurately identifying individuals based on their distinct facial features. Facial detection algorithms isolate facial regions from input images or video frames, extracting necessary details. Subsequently, LBPH facial recognition algorithms compare these features with pre-registered faces in the system's database, calculating confidence scores for identity determination. The user-friendly interface facilitates easy attendance record management, allowing administrators to add or remove students, access reports, and monitor real-time data. This Attendance Management System revolutionizes

traditional attendance tracking, providing heightened accuracy, efficiency, and security with real-time monitoring and comprehensive reporting capabilities. Its potential adoption across educational institutions, organizations, and industries signifies a significant advancement in streamlined and intelligent attendance management. Keywords: face detection, face recognition, attendance, LBPH.

2.14 AUTOMATED ATTENDANCE MARKER USING FACIAL RECOGNITION

Attendance management is vital in various sectors, including education, workplaces, and events, for accurate tracking of individuals' presence. Traditional manual methods are time-consuming, error-prone, and inefficient. This abstract introduces a solution, "Attendance Marker Using Facial Recognition," capitalizing on facial recognition technology to automate attendance. The system eliminates manual data entry, enhancing accuracy and efficiency, and ensures reliable identification due to the difficulty of forging facial features. Additionally, it accommodates a large number of attendees simultaneously, making it ideal for events with high footfall, marking a significant advancement over traditional methods.

2.15 ATTENDANCE MANAGEMENT SYSTEM USING GOOGLE FACENET FOR FACIAL RECOGNITION

An essential tool in environments requiring attendance, attendance management systems often suffer from time-consuming, invasive, and manual processes. This research endeavors to create a more efficient, cost-effective, and non-invasive automated student attendance management system, utilizing facial recognition with OpenCV functions. The system incorporates a user-friendly interface for attendance marking, updating, and viewing, leveraging OpenCV's facial recognition libraries. Attendance data is stored in an administrator-maintained database, accessible for viewing, updating, and modification. Developed on an open-source image processing library and utilizing the Python Tkinter module for interface development, the system ensures software independence and vendor hardware flexibility.

2.16 SMART MANAGEMENT ATTENDANCE SYSTEM WITH FACIAL RECOGNITION USING COMPUTER VISION TECHNIQUES ON THE RASPBERRY PI

This research introduces a sophisticated attendance system employing computer vision techniques on the Raspberry Pi device. Initial steps involve recording students and capturing facial images for the system database. The system then organizes based on lecture entry times, identifying students entering classrooms by capturing facial images with the Raspberry Pi camera. Utilizing the Viola-Jones method for face detection and the LBP method for feature extraction, each image is processed, and results are stored for facial recognition. The final stage utilizes the normalized cross-correlation (NCC) technique, achieving a remarkable 97.54% accuracy in attendance management.

2.17 AUTOMATIC CLASS ATTENDANCE SYSTEM USING BIOMETRIC FACIAL RECOGNITION TECHNIQUE BASED ON RASPBERRY PI

This research introduces an innovative automatic class attendance system utilizing biometric facial recognition technology on Raspberry Pi with a camera module. The detection algorithm calculates distances between various facial features, such as eyes, forehead, chin, nose, mouth, eye sockets, cheekbones, and contours of lips, ears, and chin. The system operates in four steps: capturing and scanning, extracting, comparing, and matching images in real-time, replacing manual attendance tracking. Implemented on a Raspberry Pi 3 model B+ microprocessor with Python's OpenCV, the system achieves 100% accuracy and fast detection in 1 second, even capturing faces at a 1.25 cm distance and varying orientations. The system's swift and accurate performance recommends its integration into diverse applications.

2.18 AUTOMATIC ATTENDANCE SYSTEM USING FACIAL RECOGNITION

Artificial Intelligence (AI) has evolved from a concept in science fiction to an integral part of our daily lives, enhancing efficiency and reducing errors with remarkable speed

and precision. The incorporation of computing aims to make technology smarter, capable of decision-making and minimizing human efforts. The implementation of an "Automatic Attendance System using Facial Recognition" stands out as an intelligent application of AI. Upon facing the camera, the system captures and sends the photo for database verification. Matching the photo with database records, the system marks attendance and updates the sheet for viewing. This paper primarily employs Convolutional Neural Network and a pre-trained FaceNet model, achieving an accuracy of approximately 94.85% with 100 different faces. Additionally, the paper proposes a rapid face detection algorithm utilizing facial expression and Linear Support Vector Machines (LSVM). Color information is initially used, followed by LSVM to segregate non-face regions, and face identification is finalized by detecting eyes and mouth.

2.19 ONLINE ATTENDANCE WITH PYTHON FACE RECOGNITION AND DJANGO FRAMEWORK

Online education necessitates an attendance system accessible from anywhere with minimal risk of fraud. This study focuses on constructing an online attendance system using face recognition to prevent unauthorized attendance submissions in online learning. Employing the object-based system approach and the waterfall development method, the system utilizes the Django Python framework, face recognition library, and OpenCV. The outcome is an attendance system resistant to impersonation. During lectures, students access the online system to record attendance, a process restricted to once per lecture schedule. The camera captures the student's face, comparing it with stored facial data, allowing only registered students to record attendance. Moreover, attendance cannot be recorded outside scheduled lecture hours. Black-box testing ensures the system's functionality, covering access button operations, facial recognition accuracy, attendance data storage, and recap viewing. Test results demonstrate the facial recognition attendance application functions at 100% accuracy, meeting expectations.

2.20 IMPLEMENTATION OF CLASSROOM ATTENDANCE SYSTEM BASED ON FACE RECOGNITION IN CLASS

The system consists of a camera that captures the images of the classroom and sends it to the image enhancement module. To enhance the captured image histogram normalization, median filtering and skin classification methods are used. Face detection is done using Viola-Jones algorithm. Initially face detection algorithm was tested on variety of images with different face positions and lighting conditions and then algorithm was applied to detect faces in real time video. Algorithm is trained for the images of faces and then applied on the class room image for detection of multiple faces in the image. The next step is face recognition, where a hybrid algorithm from PCA and LDA is used. The detected faces are cropped from the image and compared with the face database using an Eigen face method. The face database consists of templates of face images of individual students that was collected and stored by an enrollment process. In this way the faces of students are verified one by one and the attendance is marked on the server. A time table module is attached to the system to obtain the subject, class, date and time. Teachers come in the class and just press a button to start the attendance process.

2.21 FACE RECOGNITION-BASED LECTURE ATTENDANCE SYSTEM

The system consists of two cameras; one for determining the seating positions (fixed at the ceiling) and the other for capturing the students face (Fixed in front of the seats). To determine the target seat Active Student Detection(ASD) method is used to estimate the existence of a student on a seat. One seat is targeted and camera is directed to capture the image. The face image capture is enhanced and recognized and are recoded into the database. Every seat has a vector of values that represent relationship between the student and seat. Attendance is estimated by interpreting the face recognition data obtained by continuous observation. The position and attendance of the student are recorded into the database.

2.22 STUDY OF IMPLEMENTING AUTOMATED ATTENDANCE SYSTEM USING FACE RECOGNITION TECHNIQUE

The proposed system has been implemented in three basic steps. The first step is face detection and extraction. The user stands in front of the camera and an image is captured, which is taken as input. The frontal face is captured by using the OpenCVHaarCascade method. After the face is detected, it is converted into a gray scale image of 50x50 pixels. The second step is to learn and train face images. The system needs to be initialized by feeding it a set of training images of faces. The PCA algorithm is performed on it. All the learned data is stored in an xml file. The third step is the recognition and identification. In this step the frontal face that is to be recognized, test face, is extracted from the image. The Eigen value for the test face is re-calculated and is matched with the stored data for the closest neighbor. Finding the closest neighbor is implemented as a function that computes distance from the projected test face to each projected training set. The distance basis here is "Squared Euclidean Distance." When a face is matched the corresponding information is obtained from the database. The log table is then updated with the system time to mark the attendance of that person.

2.23 FACE RECOGNITION BASED ATTENDANCE MARKING SYSTEM

The system consists of a camera that must be positioned in the office room to take snap shots of the room. These images are then sent to an enhancement module where Histogram Normalization is used for the contrast enhancement of the image, Median Filter is used for removing noise from the image. To avoid false detection skin classification technique is used. This process first classifies the skin and then retains only the skin pixels and the other pixels are set to black. The enhanced image is then sent to a face detection and recognition module. This requires MATLAB software version 7.6. Two databases are maintained, the first one is the Face database to store the face images and extracted features at the time of enrolment process and the second attendance database contains the information about the employees and is also used to mark attendance.

2.24 ATTENDANCE MANAGEMENT SYSTEM USING FACE RECOGNITION

In this system, the CCTV is fixed at the entry of the class room and is used to capture an image of the entering student. The detected faces are stored in a database and is compared with the existing images using Eigen faces methodology. To identify if the student image is matching, a 3D face recognition technique is used. If a match is found, that image is processed for attendance management. For attendance management, the attendance will be marked for the student image matched and the information is sent to the server which controls the overall database of the student. The software is installed in a smart phone that would help to improve the report features. When the server receives the message of student who are absent that particular day will send an SMS to the parent of that particular student.

CHAPTER 3
SOFTWARE REQUIREMENTS SPECIFICATION

3.1 SYSTEM REQUIREMENTS

This section outlines the essential system software requirements for the project. It defines the operating system, programming languages, databases, and other necessary software components needed to develop and deploy the system. These requirements ensure compatibility, functionality, and performance of the software within its intended environment.

3.1.1 Supportive Operating Systems

The supported operating systems for client include:

- Windows XP/7/8/10
- Mac OS
- Linux Ubuntu/Mint ect
- Android/iOS/Windows Mobile

Software requirements pertain to the specification of necessary software resources and prerequisites essential for the optimal operation of an application. These requirements and prerequisites are typically not integrated into the software installation package, necessitating separate installation before the actual software installation process.

Web Browser :

- Google Chrome
- Mozila FireFox
- Internet Explorer/Edge
- Opera

3.2 SOFTWARE REQUIREMENT

Software requirements outline the functional and non-functional specifications that the software system must fulfill to meet the needs of its users. Functional requirements describe the specific actions the software must perform, such as data processing, user interactions, and system behavior. Non-functional requirements specify the quality attributes of the software, including performance, reliability, security, and usability. These requirements serve as the foundation for the design, development, testing, and evaluation of the software, ensuring that it meets the expectations and objectives of the stakeholders.

The Software Requirement for this project include:

- OpenCV 2.4.11 with gtk+
- python 3.11
- Visual Studio Code
- NoSQL

OpenCV (Open Source Computer Vision Library) is a widely-used open source software library dedicated to computer vision and machine learning. Developed to serve as a common infrastructure for computer vision applications. The library, licensed under BSD, offers businesses the flexibility to utilize and modify the code. Boasting over 2500 optimized algorithms, OpenCV includes a comprehensive array of both classic and state-of-the-art computer vision and machine learning algorithms. These algorithms cater to various tasks such as face detection and recognition, object identification, human action classification in videos, camera movement tracking, object motion tracking, 3D model extraction, generation of 3D point clouds from stereo cameras, image stitching for high-resolution scene images, similarity searches in image databases, red-eye removal from flash photography, eye movement tracking, scenery recognition, and establishing markers for augmented reality overlays. OpenCV has garnered a user community of over 47 thousand people, with an estimated download count exceeding 7 million. The library is extensively employed by companies, research groups, and governmental bodies. NoSQL, recognized as the world's most popular open source database, is chosen for its

proven performance, reliability, and user-friendly interface. As the preferred database for web-based applications, NoSQL is widely used by high-profile web properties.

3.3 HARDWARE REQUIREMENT

The typical set of requirements outlined by an operating system or software application pertains to the physical computer resources, commonly referred to as hardware. A hardware requirements list is frequently accompanied by a hardware compatibility list (HCL), particularly in the case of operating systems. The HCL serves as a compilation of tested, compatible, and occasionally incompatible hardware devices specifically designated for use with a particular operating system or application.

Table 3.1: Hardware Requirement

Component	Minimum	Recommended
Processor	1.8 Ghz Dual Core Intel Pentium/AMD Athlon 64 X2	Intel Core i3-2100 2nd Generation
RAM	2 GB	4 GB
Camera	8 Mega-Pixel	16 Mega-Pixel DSLR
Disk	128 GB	512 GB
Network	1 MB/s plan	3 MB/s

CHAPTER 4
SYSTEM DESIGN

4.1 DESIGN APPROACH

Design is the initial step in the development phase, involving the application of techniques and principles to define a device, process, or system in sufficient detail for its physical realization. Once the software requirements have been analyzed and specified, the software design encompasses three technical activities: design, coding, implementation, and testing, all of which are necessary for constructing and verifying the software. Among these activities, design holds particular importance in this phase, as it involves decisions that ultimately impact the success of the software implementation and its ease of maintenance. These decisions significantly influence the reliability and maintainability of the system. Design serves as the crucial bridge for accurately translating customer requirements into a finished software product or system. It is the pivotal stage where quality is nurtured in the development process. Software design is a systematic process that transforms requirements into a tangible representation of the software. This process unfolds in two key steps. Preliminary design is primarily concerned with the conversion of requirements into data structures and architectural foundations.

4.2 SYSTEM ARCHITECTURE

Our system adheres to a three-tier architecture, consisting of the GUI (Graphical User Interface), Recognition tier, and the Database.

4.2.1 GUI:

- Register/Login: Upon accessing the web app, users encounter a login/register page to establish their identity.

- Dashboard: After logging in, users are directed to the dashboard, the primary interface for user interaction. It provides statistics related to attendance and facilitates the upload of group images to generate attendance data sheets. Lecture details accompany each upload.

4.2.2 Trainer:

- Training: The system undergoes initial training by being provided with 5-10 grayscale images of individuals. Histograms are created from these images to enhance recognition accuracy.
- Detection: When an image is uploaded, all faces within it are cropped and stored for subsequent comparison.
- Face Recognition: Cropped faces are compared with trained images from the database using correlation. If a match is found, the corresponding ID is marked as present in the attendance data sheet.

4.2.3 Database:

- We utilize a centralized NoSQL database containing details of students and staff.
- Every cropped or detected image undergoes comparison with trained images in the database
- for retrieval, facilitating the generation of attendance sheets through updates to the database.

4.3 DATA FLOW DIAGRAMS

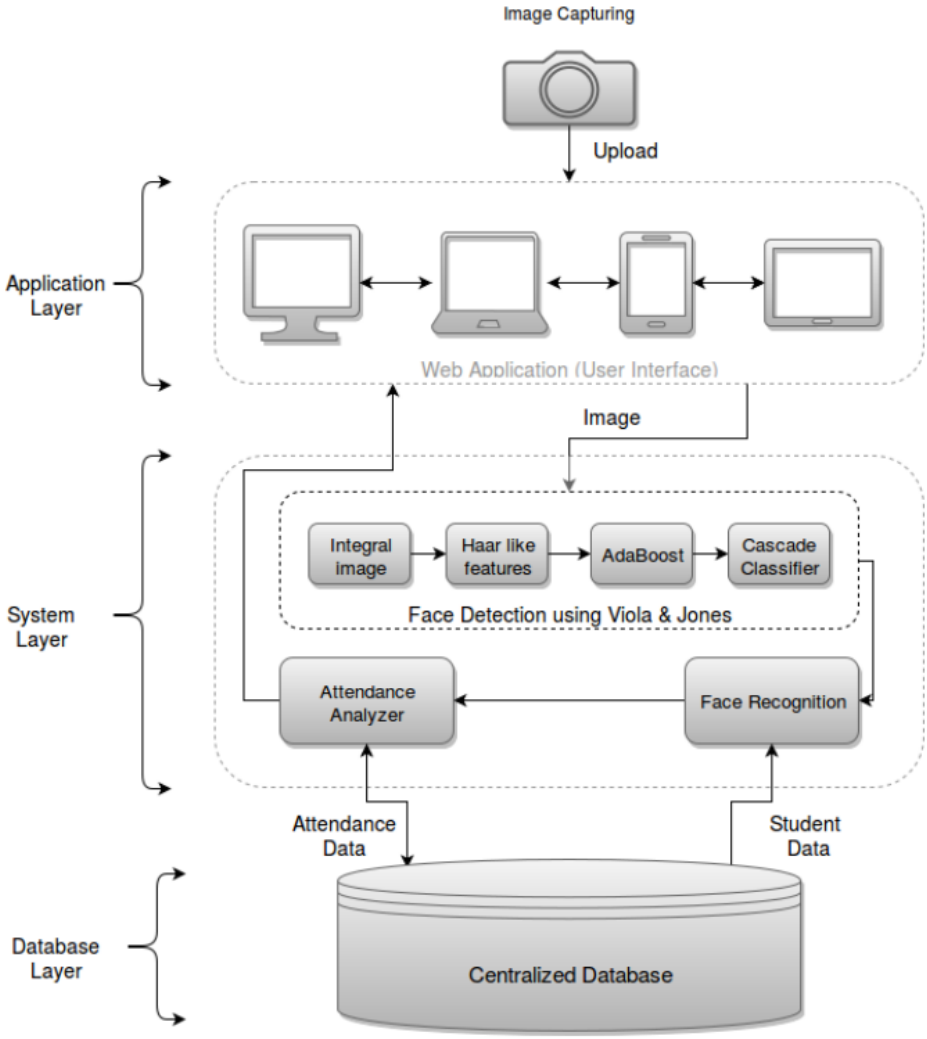


Figure 4.1: System Architecture

4.4 ENTITY RELATIONSHIP DIAGRAMS

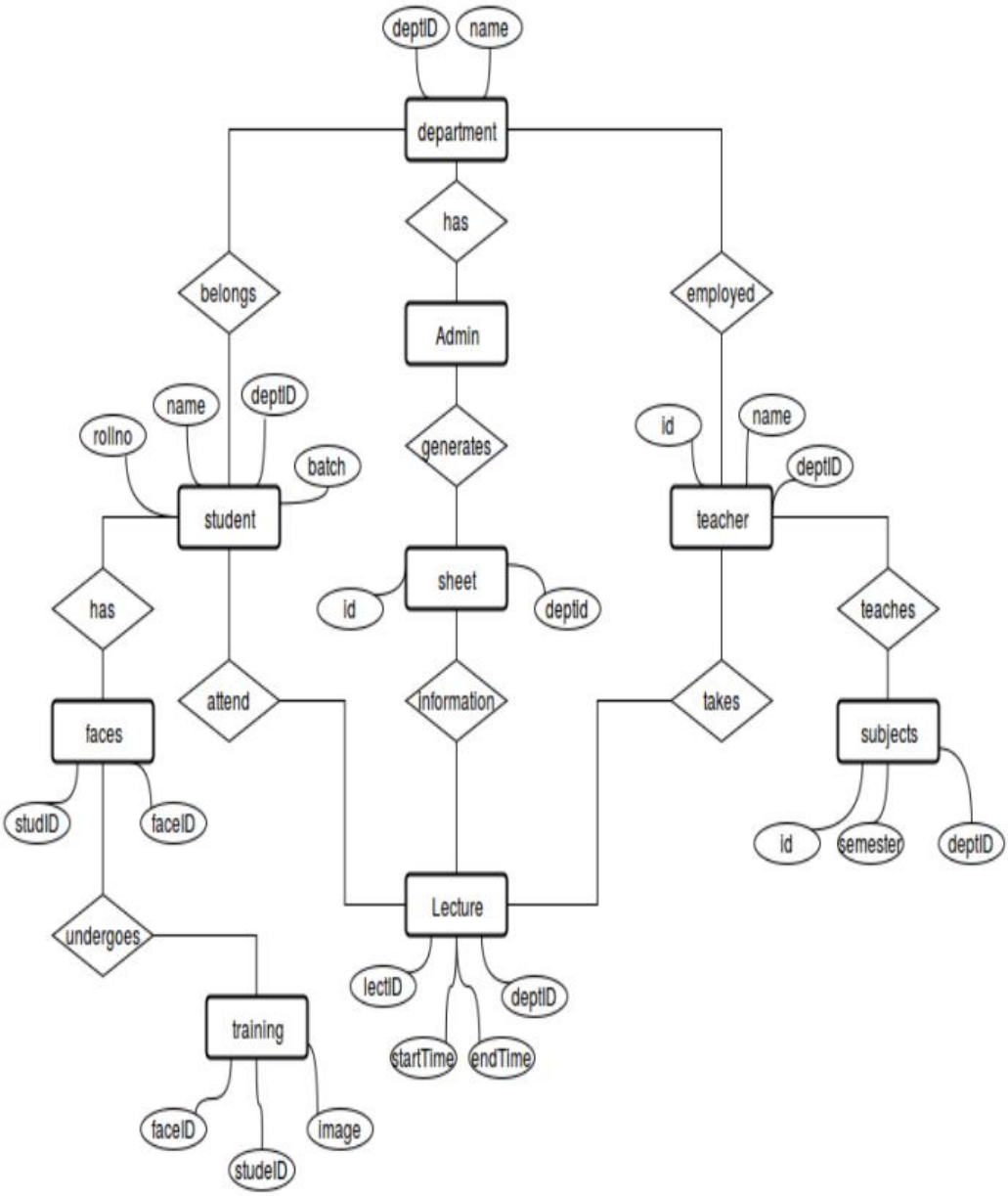


Figure 4.2: E-R Diagram of Real Time Product Analysis system

4.5 USE CASE DIAGRAM

A use case diagram visually depicts the interactions between a system and external entities, referred to as actors, to achieve specific tasks or objectives. It presents the functional requirements of the system from the viewpoint of external users, emphasizing the system's actions rather than the underlying mechanisms. Actors may include users, other systems, or external entities, while use cases represent the functionalities or services provided by the system. Through scenarios and interactions depicted between actors and the system, this diagram offers a broad overview of the system's behavior. It serves as a valuable tool for comprehending, communicating, and analyzing the system's functionality and user interactions at a high level.

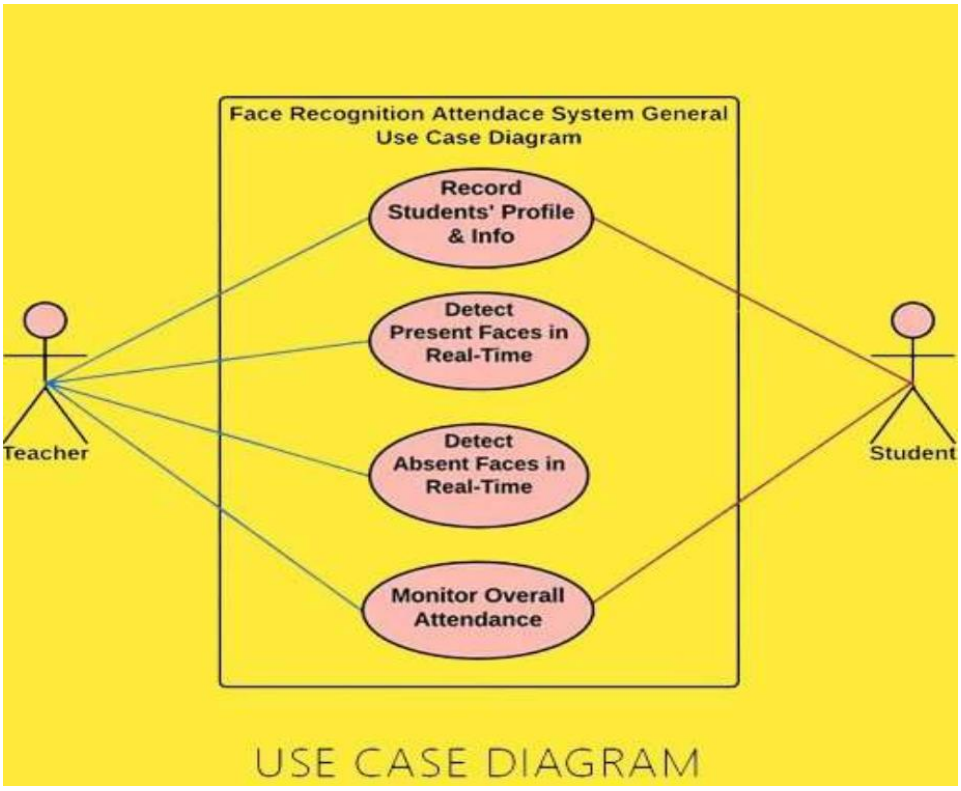


Figure 4.3: Face Recognition Attendance System General UCD

4.6 BLOCK DIAGRAM

A block diagram is a visual depiction of a system or process, employing blocks to represent its distinct components and lines to signify their relationships or interactions. It offers a top-level view of a complex system or process by breaking it down into simpler elements and illustrating how they are interconnected or associated. Blocks within the diagram may symbolize physical components like hardware devices, subsystems, or software modules, as well as abstract concepts or procedures. The connections between blocks denote the flow of data, signals, or other inputs and outputs among the components, serving to elucidate the logic or functionality of the system or process. Widely utilized in engineering, science, and technology, block diagrams facilitate the modeling and analysis of intricate systems or processes, while also enabling clear and succinct communication of their design and operation to stakeholders.

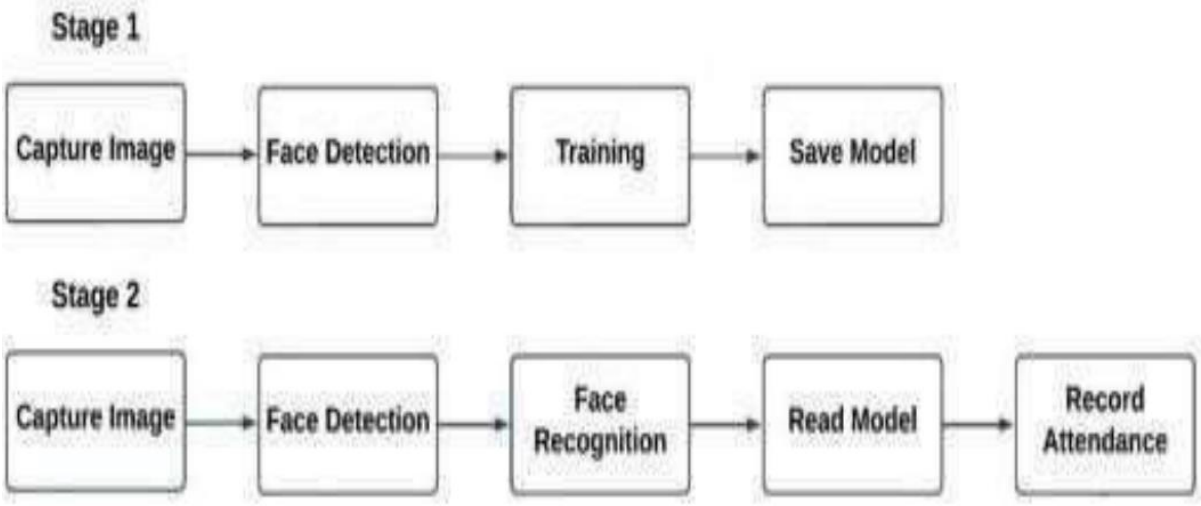


Figure 4.4: Stages of Proposed Methodology

CHAPTER 5
PROJECT PLAN

5.1 PROJECT ESTIMATES

5.1.1 Reconciled Estimates

Reconciled estimates involve a comprehensive evaluation and adjustment of project estimates to ensure precision in resource allocation and scheduling. This process incorporates several key elements:

Detailed Task Analysis:

- Every project task, from requirement analysis to documentation, undergoes a thorough examination. This includes breaking down complex activities into smaller, more manageable sub-tasks.

Risk Assessment:

- Potential risks and uncertainties are identified and factored into the estimates. This involves considering external factors that may impact task durations or resource requirements.

Contingency Planning:

- Contingency plans are developed to address unforeseen challenges. Time and resource buffers are incorporated into estimates to account for any deviations from the planned schedule.

Historical Data Utilization:

- Data from past projects or industry benchmarks are utilized to validate and refine estimates. This historical perspective aids in anticipating potential challenges and streamlining the planning process.

Stakeholder Consultation:

- Collaboration with stakeholders, including end-users and project sponsors, helps in refining estimates based on their insights and expectations. This ensures that the project aligns with stakeholders' needs and priorities.

Expert Opinions:

- Input from subject matter experts and experienced team members is sought to validate estimates. This collective expertise enhances the accuracy of estimations, especially in complex technical domains.

Feedback Loops:

- Continuous feedback loops are established throughout the estimation process. Regular reviews and updates ensure that estimates remain aligned with the evolving project landscape.

Documentation of Assumptions:

- Assumptions underlying the estimates are explicitly documented. This includes assumptions related to resource availability, task dependencies, and potential risks. Documenting assumptions facilitates transparent communication within the project team.

Regular Updates:

- Estimates are not static; they are subject to change based on evolving project dynamics. Regular updates and adjustments are made as the project progresses and new information becomes available.

Tool Utilization:

- Project management tools and software are employed to facilitate accurate estimation. These tools provide features such as Gantt charts and critical path analysis, aiding in visualizing task dependencies and optimizing resource allocation.

5.2 PROJECT RESOURCES

To successfully execute the final year project, a comprehensive set of resources is required:

Human Resources:

- Project Manager: Oversees the entire project, ensuring alignment with goals and timelines.
- Software Developers: Responsible for coding and implementing the face recognition system.
- Database Administrator: Manages the design and integration of the centralized database.
- Web Developers: Develops the web-based application for user interaction.
- Testers: Ensure the reliability and accuracy of the face recognition system.

Hardware Resources:

- Servers: Needed for hosting the application.
- High-definition cameras: Used for capturing images for face recognition.
- Computing devices for testing: Essential for validating system performance.

Software Resources:

- OpenCV: Utilized for image processing and implementing the Viola and Jones algorithm.
- Web development tools: Employed for creating the user interface.

- Database management system: Required for the storage and retrieval of attendance data.
- Integrated Development Environment (IDE): Facilitates software development.

5.3 PROJECT SCHEDULE

5.3.1 Project Task Set

Requirement Analysis (2 weeks):

- Identify and document system requirements based on user stories and use cases.

System Design (3 weeks):

- Architectural design of the system, including front-end and back-end components.

Implementation (8 weeks):

- Develop the face detection module using the Viola and Jones algorithm.
- Implement the face recognition module.
- Create the web-based application for user interaction.

Database Implementation (4 weeks):

- Design and implement the centralized database.
- Integrate the database with the system.

Testing (3 weeks):

- Conduct unit testing for each module.
- Perform system testing and integration testing to ensure reliability.

Documentation (2 weeks):

- Prepare comprehensive user manuals.
- Document the system architecture and design for future reference.

5.3.2 Task Network

Tasks are interdependent, with some tasks relying on the completion of others. For example, system design must be finalized before implementation can commence.

5.3.3 Timeline Chart

A Gantt chart will be created to visualize the timeline for each task. This chart will include start and end dates, helping the project team manage and track progress effectively.

5.4 TEAM ORGANIZATION

5.4.1 Team Structure

The project team is structured to optimize collaboration and task completion:

Project Manager:

- Oversees the entire project.
- Ensures team alignment with project goals.

Development Team:

- Software Developers: Responsible for coding and implementing the face recognition system.
- Database Administrator: Manages the design and integration of the centralized database.
- Web Developers: Develops the web-based application for user interaction.

Testing Team:

- Ensures the reliability and accuracy of the face recognition system.

Documentation Team:

- Prepares user manuals and documents the system architecture and design.

5.4.2 Management Reporting and Communication

Regular progress meetings will be held to bandy design status, address challenges, and maintain effective communication within the platoon. Project reports will be participated with stakeholders to keep them informed of developments and mileposts.

Communication channels will include design operation tools and listed meetings to ensure transparent and effective collaboration.

CHAPTER 6

PROJECT IMPLEMENTATION

6.1 OVERVIEW OF PROJECT MODULES

The implementation phase of the "Automated Attendance System Using Face Recognition" project involves the development and integration of several key modules. Each module plays a crucial role in achieving the overall objective of automating attendance management. Below is an overview of the primary project modules:

6.1.1 Face Detection Module

The Face Detection Module utilizes the Viola and Jones algorithm to identify and locate human faces within digital images. This module is a fundamental component as it forms the basis for subsequent recognition processes. The algorithm involves the generation of an integral image and the use of Haar-like features to detect facial features such as eyes, nose, and mouth. The application of the Adaboost algorithm enhances performance by selecting the most relevant features. The result is a reliable face detection mechanism crucial for accurate attendance recording.

6.1.2 Face Recognition Module

The Face Recognition Module is responsible for verifying and identifying individuals based on facial features extracted during the detection phase. This module employs various facial recognition algorithms, including Principal Component Analysis using eigenfaces, Linear Discriminate Analysis, and Normalized Cross-Correlation. The system extracts features such as eye position, nose shape, and jawline to create a unique facial signature. The recognition process involves comparing these features with a database of stored images, allowing the system to accurately identify and record attendance for recognized individuals.

6.1.3 Database Management Module

The Database Management Module is central to the project, serving as a repository for student information and attendance records. During the initial phase, student images are captured, processed, and stored in the database. This module facilitates the retrieval and comparison of facial features during recognition. Additionally, it maintains an up-to-date attendance record for each student. The database layer ensures data integrity, security, and seamless integration with the other modules.

6.1.4 Web-Based Application Module

The Web-Based Application Module provides the user interface for interacting with the system. Users, including professors and administrators, can upload image files containing student faces for attendance recording. Authentication mechanisms are in place to ensure secure access. The application also offers features for viewing attendance reports, providing a user-friendly experience. This module enhances the system's accessibility and usability, making it practical for deployment in educational institutions.

6.1.5 System Layer (Integration Module)

The System Layer, or Integration Module, acts as the glue that brings together the various components of the system. It manages the flow of data between the face detection, recognition, database, and web application modules. This layer ensures seamless communication and coordination, allowing the system to function as a unified and efficient solution for automated attendance management.

6.1.6 Future Enhancements

While the core modules cover the essential functionalities of the Automated Attendance System, future enhancements could include:

- Robustness testing to handle variations in environmental conditions.
- Integration of additional recognition algorithms for improved accuracy.

- Implementation of a mobile-based face recognition application for increased accessibility.
- Continuous improvement of face recognition effectiveness through advanced techniques.

6.2 TOOLS AND TECHNOLOGIES USED

The implementation of the "Automated Attendance System Using Face Recognition" project involves the utilization of various tools and technologies to ensure the seamless development of different components. The key tools and technologies employed in this project include:

Programming Languages:

- Python: Used for coding the face detection and recognition modules.

Image Processing Library:

- OpenCV (Open Source Computer Vision Library): Utilized for image processing tasks, particularly in implementing the Viola and Jones algorithm for face detection.

Database Management System:

- NoSQL: Selected as the non-relational database management system for storing and managing attendance data.

Integrated Development Environment (IDE):

- PyCharm: Used as the primary integrated development environment for Python coding.

Version Control:

- Git: Employed for version control to track changes and facilitate collaborative development.

Project Management:

- Jira: Utilized for project management, issue tracking, and agile development.

Documentation:

- Microsoft Word and LaTeX: Utilized for creating comprehensive project documentation, including user manuals and system architecture documentation.

Communication and Collaboration:

- Slack: Employed for team communication and collaboration.
- Microsoft Teams: Used for virtual meetings, discussions, and document sharing.

Virtual Environment:

- Anaconda: Utilized for managing Python packages and creating a virtual environment to ensure project-specific dependencies.

Diagram Creation:

- Lucid chart: Utilized for creating system architecture diagrams and flowcharts.

6.3 ALGORITHM DETAILS

The implementation of the "Automated Attendance System Using Face Recognition" project relies on sophisticated algorithms for face detection and recognition. Here are the algorithmic details of the key processes involved:

6.3.1 Face Detection Algorithm - Viola and Jones:

The Viola and Jones algorithm, originally crafted for real-time face detection, is a resilient and efficient object detection framework that has expanded to various recognition tasks. Its core components include integral images, which swiftly compute pixel sums within rectangular regions, and Haar-like features, rectangular filters that pinpoint facial traits across different scales and positions. Adaboost, a boosting

algorithm, amalgamates weak classifiers based on Haar-like features to create a robust classifier, assigning weights to emphasize misclassification-prone features. The cascade classifier, organized in stages, rapidly discards non-face regions, optimizing computational resources by advancing regions that pass each stage.

The detection process involves utilizing integral images to compute Haar-like features, applying cascades to reject non-face regions, and classifying features that pass all stages as faces. This algorithm boasts benefits like real-time processing, computational efficiency, and robustness to lighting conditions and facial expressions, yet faces challenges such as false positives and sensitivity to pose variations. Its applications span from facial recognition systems to video surveillance and human-computer interaction, extending to object detection tasks, showcasing its versatility and adaptability across diverse domains.

Steps involved in the Viola and Jones Algorithm:

Step 1: Integral Image Calculation

Calculate the integral image (also known as a summed area table) from the input image. This efficiently computes the sum of pixel values in any rectangular region.

Step 2: Haar-like Feature Extraction

Extract Haar-like features from the integral image. These features are simple rectangular filters used to represent specific characteristics of faces.

Step 3: Feature Selection

Select a subset of Haar-like features that are most informative for distinguishing between faces and non-faces.

Step 4: Adaboost Training

Train a cascade of classifiers using the Adaboost (Adaptive Boosting) algorithm. This combines multiple weak classifiers to create a strong classifier.

Step 5: Cascade Classifier Construction

Construct a cascade classifier consisting of multiple stages, each containing a set of weak classifiers. The stages are organized in a cascade, and each stage aims to quickly eliminate non-face regions.

Step 6: Feature Evaluation

Evaluate the selected Haar-like features at each stage of the cascade classifier to determine whether a region contains a face or a non-face.

Step 7: Region Rejection

If a region passes all stages of the cascade classifier, classify it as a face. Otherwise, reject it as a non-face.

Step 8: Face Detection

Detect faces in the input image by applying the cascade classifier to different regions of the image. If a region is classified as a face, mark it as a potential face detection.

Step 9: Post-processing

Perform post-processing steps, such as non-maximum suppression, to remove overlapping detections and refine the final set of detected faces.

Step 10: Output

Output the coordinates and sizes of the detected faces in the input image, along with any associated confidence scores or probabilities.

6.3.2 Face Recognition Algorithm - Correlation Technique:

The face recognition algorithm employs the correlation technique to match detected faces with images stored in the database, utilizing normalized cross-correlation to compare facial features. Key components include Region of Interest (ROI) Extraction to isolate facial features, Total Offset Calculation for alignment, and Face Extraction Confirmation

for verification. In the recognition process, faces are compared using normalized cross-correlation, and successful extraction and alignment confirm identity. Benefits include precision and reliability, while challenges arise from variability in expressions and lighting. Its applications span biometric identification needs like attendance systems and security access control. Integrated with Viola and Jones face detection, it enhances attendance management systems comprehensively.

Steps involved in the Face Recognition Algorithm using the Correlation Technique:

Step 1: Face Detection

Detect faces in the input image using a face detection algorithm like Viola-Jones.

Step 2: Region of Interest (ROI) Extraction

Extract the region of interest (ROI) from the detected face. This region specifically contains the facial features needed for recognition.

Step 3: Normalized Cross-Correlation

Perform normalized cross-correlation between the extracted face in the image and the face images stored in the database. This measures the similarity between the two images.

Step 4: Total Offset Calculation

Calculate the total offset or translation needed to align the detected face with the stored face image. This offset is crucial for accurate matching.

Step 5: Face Extraction Confirmation

Verify the successful extraction of the face from the target image. Ensure that the detected face aligns correctly with the stored facial features.

Step 6: Recognition Process

Compare the extracted face from the image with the face images stored in the database using normalized cross-correlation.

Step 7: Identity Confirmation

Determine if the normalized cross-correlation score meets a predefined threshold to confirm the identity of the detected face.

Step 8: Result Output

Output the recognized identity if the correlation score exceeds the threshold; otherwise, label the face as unrecognized.

6.3.3 Local Binary Pattern Histogram (LBPH) Algorithm - Face Recognition:

The LBPH algorithm stands out as a texture-based method for face recognition, adept at capturing and leveraging local texture patterns for robust identification, forming a detailed representation of facial features by encoding pixel intensity relationships across different face regions. Its key components include image grid partitioning, where the face image undergoes division into cells for localized analysis, and histogram calculation within each cell, generating a Local Binary Pattern (LBP) descriptor to form histograms based on LBP pattern frequencies. These histograms are then concatenated to form a feature vector, capturing the unique texture patterns of the entire face. During recognition, LBPH compares the test image's feature vector with those in the database, using similarity metrics like Euclidean distance or cosine similarity to determine identity.

Advantages of LBPH include robustness to illumination changes and detailed texture information, while challenges include limited pose variation adaptability and sensitivity to noise. LBPH finds applications in face recognition systems, especially in challenging lighting conditions, and security and surveillance systems. In this project, LBPH integrates as an additional technique alongside the correlation method, aiming to enhance overall system accuracy and reliability. Future research focuses on enhancing LBPH's robustness against pose changes and noise, potentially incorporating advanced techniques to address these challenges effectively.

Steps involved in the Local Binary Pattern Histogram (LBPH):

Step 1: Image Preprocessing

Convert the input grayscale image into a uniform size and format.

Step 2: Divide Image into Cells

Partition the preprocessed image into smaller cells or blocks.

Step 3: Local Binary Pattern (LBP) Calculation

For each pixel in each cell, calculate the LBP value based on its neighboring pixels' intensity values.

Step 4: Histogram Calculation

Compute a histogram for each cell based on the frequency of different LBP patterns within that cell.

Step 5: Concatenate Histograms

Combine the histograms from all cells into a single feature vector.

Step 6: Feature Normalization

Optionally, normalize the feature vector to ensure robustness against variations in illumination or contrast.

Step 7: Classification

Train a classifier, such as a support vector machine (SVM) or a neural network, using the feature vectors obtained from a training dataset.

Step 8: Recognition

Given a new input image, extract the LBPH feature vector using the same procedure and classify it using the trained classifier to determine the object or face identity.

CHAPTER 7
RESULTS AND SCREENSHOTS

7.1 ADMIN PANEL

In this section, we present the results and screenshots of the admin panel interface developed for the project. The admin panel serves as a user-friendly graphical interface designed to facilitate the addition of student data into the system database. This includes essential information such as student names, PRN (Permanent Registration Number), subjects, class, and semester details. The admin panel interface is a pivotal component of the project, providing administrators with a centralized platform to efficiently manage student data. Through intuitive design and seamless functionality, the admin panel streamlines the process of data entry, ensuring accuracy and consistency in the student records. With a focus on user experience and accessibility, the admin panel interface enhances the overall efficiency of the attendance management system, empowering administrators to maintain comprehensive student databases effortlessly.

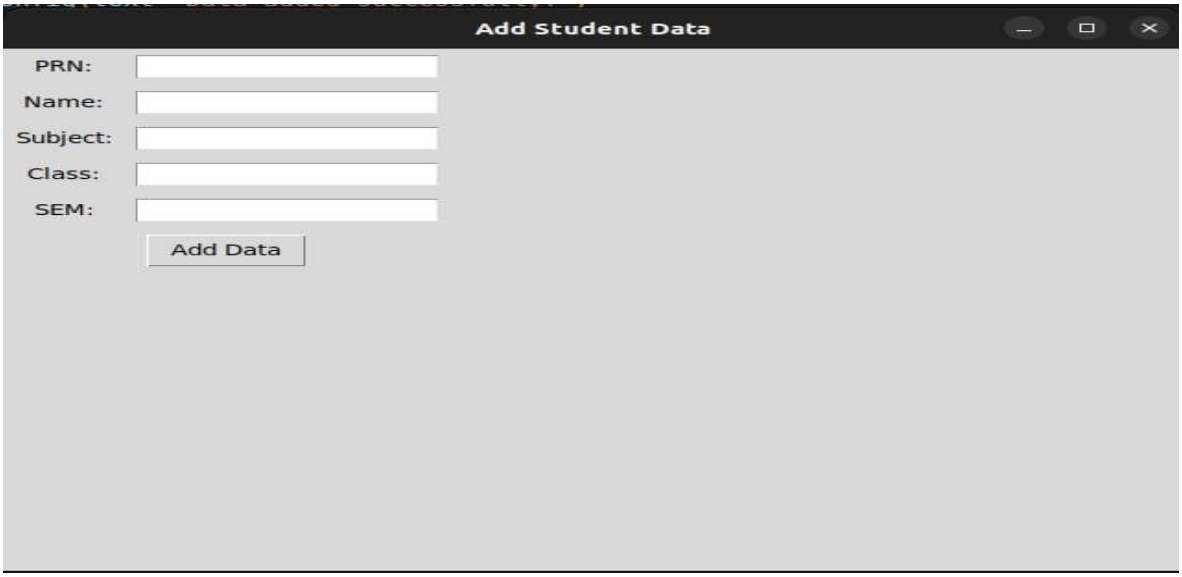


Figure 7.1: Add Student Data

7.2 STORAGE BUCKET FOR IMAGES

This section focuses on the image storage bucket, a vital aspect of our project. The storage bucket holds student images essential for the face detection model. Through screenshots, we illustrate its user interface and backend operations, highlighting its seamless integration within the system. This section demonstrates the importance of the storage bucket in enabling efficient face detection.

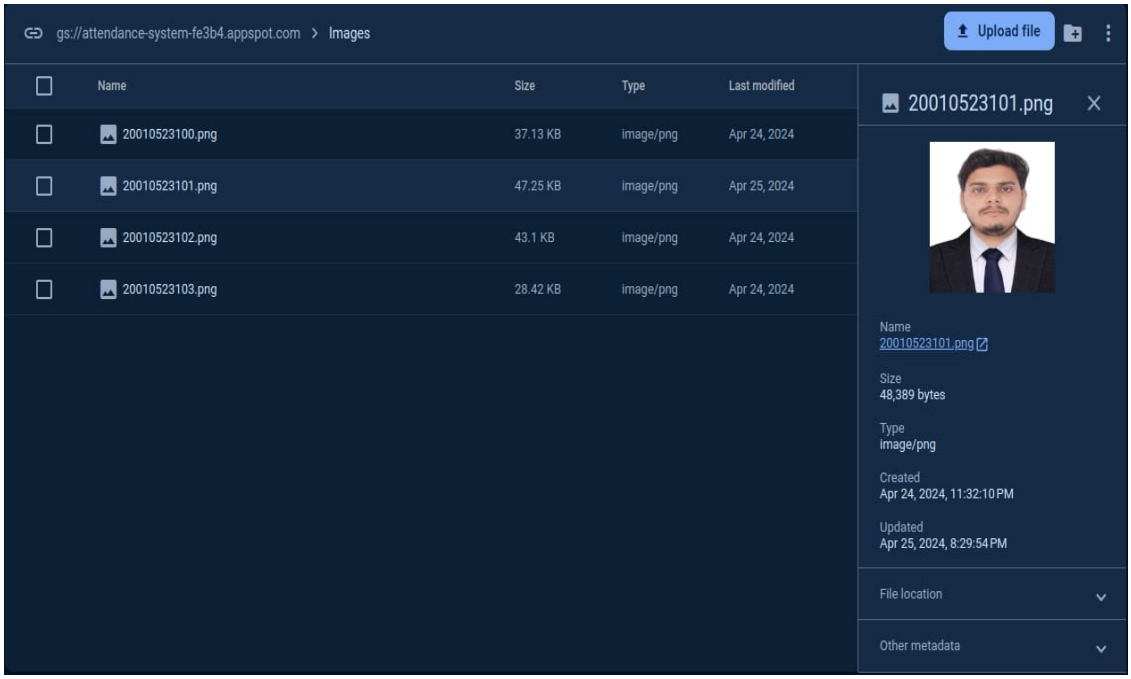


Figure 7.2: Storing and Feeding Images

7.3 FACE RECOGNITION

In this chapter, we delve into the intricacies of the face recognition interface, a pivotal component of our project. This interface serves as a real-time detection mechanism for students, providing valuable information instantly. We explore its intuitive user interface, showcasing its seamless integration with the underlying face detection model. Through a screenshot, we elucidate the functionality and effectiveness of the face recognition interface. By highlighting its ability to detect students in real-time and display pertinent information, we underscore its significance in automating attendance management. This section offers insights into the user experience and operational efficiency achieved through the face recognition interface.

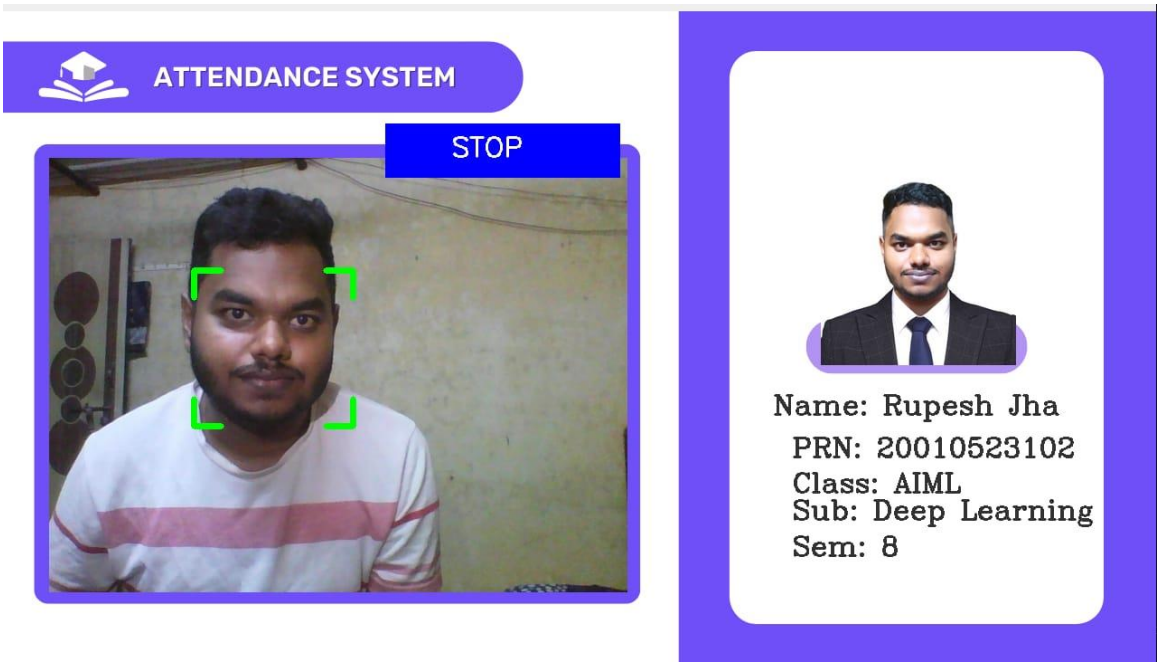


Figure 7.3: Output Screen

7.4 REAL-TIME ATTENDANCE MARKING

Our project presents a real-time student attendance tracking system. Using advanced face recognition technology, it accurately identifies students and marks their attendance. The interface displays student details along with timestamps for each attendance record. Through screenshot, we demonstrate the user-friendly interface and the system's efficiency in automating attendance management.

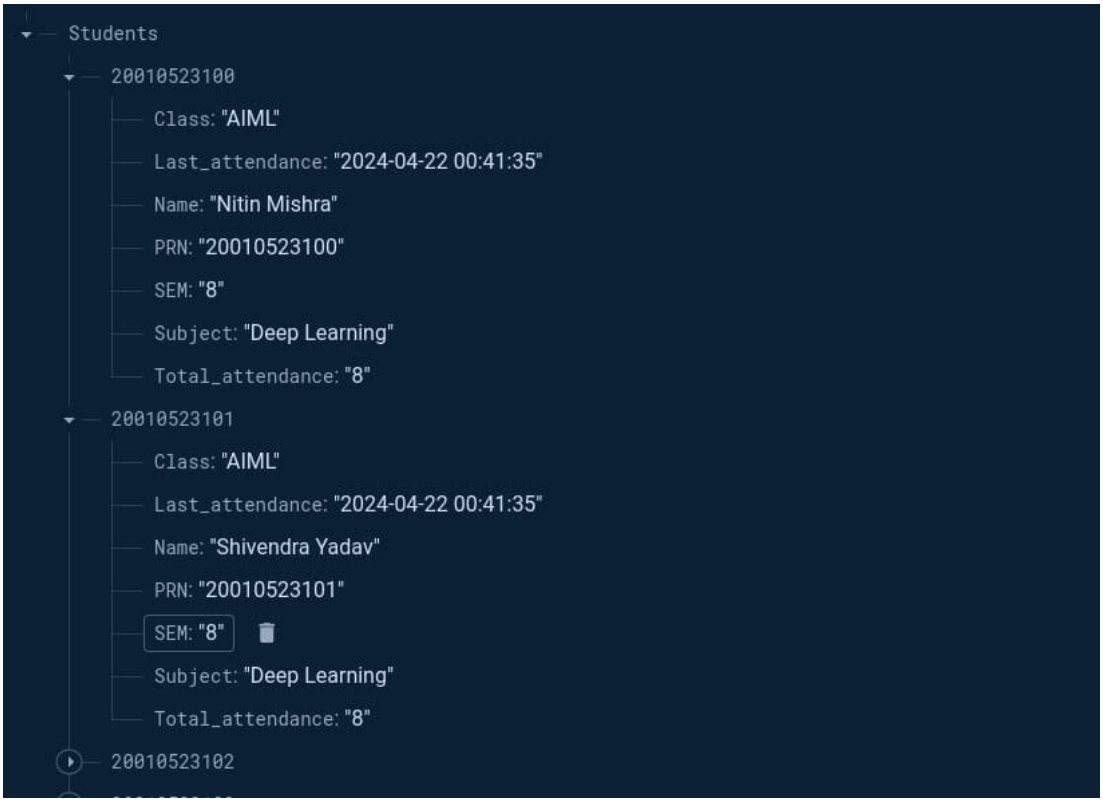
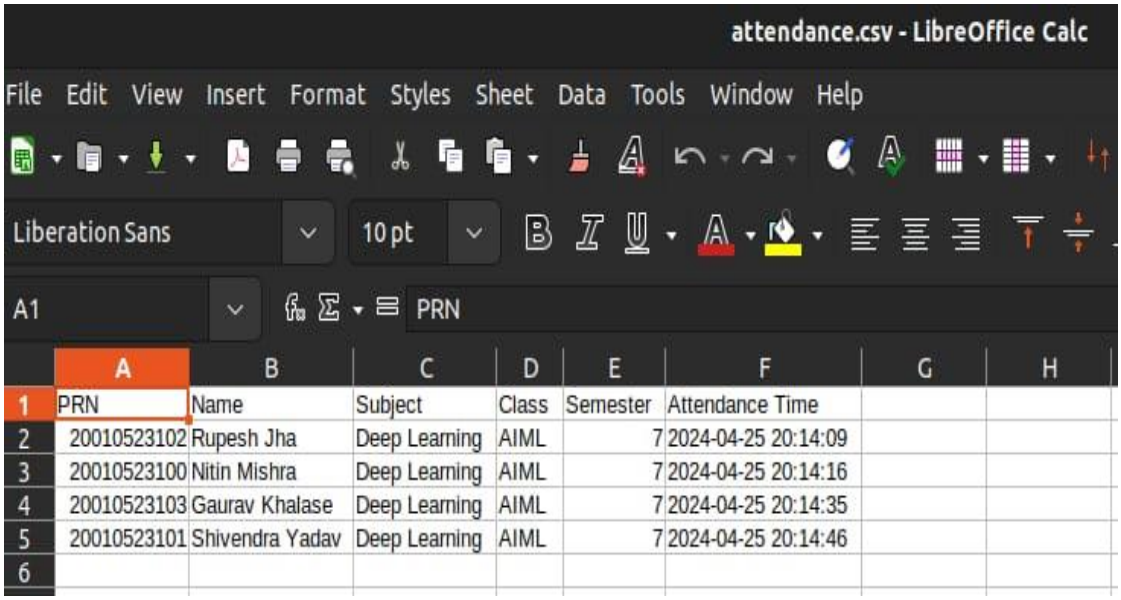


Figure 7.4: Real-Time Database Update

7.5 RECORD FOR ATTENDANCE

This section offers a concise overview of the Excel file generation feature, which organizes attendance data into a structured format for convenient access and analysis. Through screenshots, we illustrate the user interface and guide through the process of generating attendance reports. These Excel files encompass extensive information such as student names, attendance status, dates, and other pertinent details, simplifying record-keeping and analysis tasks. By automating the generation of attendance reports, this feature optimizes administrative workflows, thereby improving productivity and accuracy in attendance management.



	A	B	C	D	E	F	G	H
1	PRN	Name	Subject	Class	Semester	Attendance Time		
2	20010523102	Rupesh Jha	Deep Learning	AIML	7	2024-04-25 20:14:09		
3	20010523100	Nitin Mishra	Deep Learning	AIML	7	2024-04-25 20:14:16		
4	20010523103	Gaurav Khalase	Deep Learning	AIML	7	2024-04-25 20:14:35		
5	20010523101	Shivendra Yadav	Deep Learning	AIML	7	2024-04-25 20:14:46		
6								

Figure 7.5: Excel Record

CHAPTER 8
CONCLUSION

8.1 CONCLUSION

In conclusion, the Real-Time Face Recognition-Based Student Attendance System, built upon the robust Viola and Jones algorithm, stands as a groundbreaking solution to the challenges of attendance management in educational institutions. The application of advanced face recognition technology not only streamlines the attendance tracking process but also significantly enhances efficiency and security. By replacing the manual system with an automated one, this design offers a swift, effective, and cost-saving alternative, eliminating the need for stationary materials and paperwork. The system's real-time capabilities, coupled with the precision of the Viola and Jones algorithm, ensure reliable attendance data while minimizing the risk of manipulation. Looking ahead, the proposed system has the potential for further optimization, and future advancements could include the integration of new approaches to improve overall efficiency. As a result, this innovative solution not only meets the current demands of attendance management but also lays the foundation for continuous enhancement and adaptation to evolving technological advancements.

8.2 FUTURE SCOPE

It can be effortlessly implemented in any educational institution or organization. A methodology could be suggested to demonstrate resilience against variations, indicating that in the near future, we could develop a system that is robust and effective even in unfavorable conditions. While the proposal is currently focused on student attendance in educational institutes, there is potential for future expansion to encompass attendance tracking at both entry and exit points. The authors are actively working on enhancing the effectiveness of face recognition to create more efficient systems in the near future. In upcoming research, the authors plan to elevate face recognition effectiveness by incorporating interaction between the system, users, and administrators. Additionally, the

system holds promise for a novel face recognition application—mobile-based face recognition. This application could serve as a tool for the general public to identify individuals captured by a cell phone camera, complete with proper authorization to access a centralized database.

REFERENCES

1. Rekha A , Dr. Chethan H, Automated Attendance System Using Face Recognition Through Video Surveillance, International Journal For Technological Research In Engineering, Volume 1, Issue 11, July-2014.

2. Nirmalya Kar, Mrinal Kanti Debbarma, Ashim Saha, and Dwijen Rudra Pal, Study Of Implementing Automated Attendance System Using Face Recognition Technique, International Journal Of Computer And Communication Engineering, Vol. 1, No. 2, July 2012.

3. Naveed Khan Balcoh, M. HaroonYousaf, Waqar Ahmad And M. Iram Baig, Algorithm For Efficient Attendance Management: Face Recognition Based Approach, International Journal Of Computer Science Issues, Vol. 9, Issue 4, No 1, July 2012.

4. Kwok-Wai Wong, Kin-Man Lam, Wan-Chi Siu, An efficient algorithm for human face detection and facial feature extraction under different conditions, The Journal Of the Pattern Recognition Society, 25 Aug-2000.

5. Sarabjit Singh, Amritpal Kaur, Taqdir, A Face Recognition Technique using Local Binary Pattern Method, International Journal of Advanced Research in Computer and Communication Engineering Vol. 4, Issue 3, March 2015.

6. Honen, Timo, Abdenour Hadid, and Matti Pietikainen. “Face description with local binary patterns: Application to face recognition.” IEEE transactions on pattern analysis and machine intelligence 28.12 (2006): 2037–2041.

7. Ojala, Timo, Matti Pietikainen, and Topi Maenpaa. “Multiresolution gray-scale and rotation invariant texture classification with local binary patterns.” IEEE Transactions on pattern analysis and machine intelligence 24.7 (2002): 971–987.

8. Ahonen, Timo, Abdenour Hadid, and Matti Pietikäinen. “Face recognition with local binary patterns.” Computer vision-eccv 2004 (2004): 469–481.

9. Paul Viola and Michael Jones, Rapid Object Detection using a Boosted Cascade of Simple Features. Accepted Conference on Computer Vision and Pattern Re cognition, 2001.

10. Face Detection with Haar Cascade,Towards Data Science-727f68dafd08,Girija Shankar Behera, India, Dec 24, 2020.

11. Face Recognition: Understanding LBPH Algorithm,Towards Data Science-90ec258c3d6b,Kelvin Salton do Prado, Nov 11, 2017.

12. Kushsairy Kadir , Mohd Khairi Kamaruddin, Haidawati Nasir, Sairul I Safie, Zulkifli Abdul Kadir Bakti,"A comparative study between LBP and Haar-like features for Face Detection using OpenCV", 4th International Conference on Engineering Technology and Technopreneuship (ICE2T), DOI:10.1109/ICE2T.2014.7006273, 12 January 2015.

13. Senthamizh Selvi.R,D.Sivakumar, Sandhya.J.S , Siva Sowmiya.S, Ramya.S , Kanaga Suba Raja.S,"Face Recognition Using Haar - Cascade Classifier for Criminal Identification", International Journal of Recent Technology and Engineering(IJRTE), vol.7, issn:2277-3878, issue-6S5, April 2019.

14. Margaret Rouse, What is facial recognition? - Definition from WhatIs.com, 2012.

15. Robert Silk, Biometrics: Facial recognition tech coming to an airport near you: Travel Weekly, 2017.

16. Sidney Fussell, NEWS Facebook's New Face Recognition Features: What We Do (and Don't) Know, 2018.

17. Reichert, C. Intel demos 5G facial-recognition payment technology | ZDNet, 2017.

18. Mayank Kumar Rusia, Dushyant Kumar Singh, Mohd. Aquib Ansari, “Human Face Identification using LBP and Haar-like Features for Real Time Attendance Monitoring”, 2019. Fifth International Conference on Image Information Processing (ICIIP) ,Shimla, India, DOI: 10.1109/ICIIP47207.2019.8985867 10 February 2020.

APPENDIX A

CODES

GUI ADD DATA

The below code is a Python script that implements a graphical user interface (GUI) using the Tkinter library for adding student data to a Firebase database. It initializes Firebase using a service account key and defines a function 'add_data()' to collect input data from the user and add it to the database. The GUI window includes entry fields for the student's PRN, name, subject, class, and semester, along with a button to trigger the data addition process. Upon successful addition, a status label updates to indicate success. The script utilizes Firebase Admin SDK for Python to interact with the Firebase Realtime Database.

```
import tkinter as tk
from firebase_admin import credentials, db, initialize_app
from datetime import datetime

# Initialize Firebase
cred = credentials.Certificate("serviceAccountKey.json")
initialize_app(cred , {
    'databaseURL' : "https://attendance-system-fe3b4-default-
rtdb.asia-southeast1.firebasedatabase.app/"
})

# Function to add data to Firebase
def add_data():
    prn = prn_entry.get()
    name = name_entry.get()
    subject = subject_entry.get()
    clas = class_entry.get()
    sem = sem_entry.get()

    default_last_attendance = "2024-04-22 00:41:35"  # Default
value for Last_attendance

    data = {
        "Name": name,
        "PRN": prn,
        "Subject": subject,
        "Class": clas,
```

```
        "SEM": sem,
        "Last_attendance": default_last_attendance,
        "Total_attendance": "0"
    }

    ref = db.reference('Students')
    ref.child(prn).set(data)

    status_label.config(text="Data added successfully!")

# Create Tkinter window
window = tk.Tk()
window.title("Add Student Data")
window.geometry("640x480")  # Set window size

# PRN
prn_label = tk.Label(window, text="PRN:")
prn_label.grid(row=0, column=0, padx=5, pady=5)
prn_entry = tk.Entry(window)
prn_entry.grid(row=0, column=1, padx=5, pady=5)

# Name
name_label = tk.Label(window, text="Name:")
name_label.grid(row=1, column=0, padx=5, pady=5)
name_entry = tk.Entry(window)
name_entry.grid(row=1, column=1, padx=5, pady=5)

# Subject
subject_label = tk.Label(window, text="Subject:")
subject_label.grid(row=2, column=0, padx=5, pady=5)
subject_entry = tk.Entry(window)
subject_entry.grid(row=2, column=1, padx=5, pady=5)

# Class
class_label = tk.Label(window, text="Class:")
class_label.grid(row=3, column=0, padx=5, pady=5)
class_entry = tk.Entry(window)
class_entry.grid(row=3, column=1, padx=5, pady=5)

# SEM
sem_label = tk.Label(window, text="SEM:")
sem_label.grid(row=4, column=0, padx=5, pady=5)
sem_entry = tk.Entry(window)
sem_entry.grid(row=4, column=1, padx=5, pady=5)
```

```
# Button to add data
add_button = tk.Button(window, text="Add Data", command=add_data)
add_button.grid(row=5, column=0, columnspan=2, padx=5, pady=5)

# Status label
status_label = tk.Label(window, text="")
status_label.grid(row=6, column=0, columnspan=2, padx=5, pady=5)

window.mainloop()
```

ENCODE GENERATOR

This script is used for encoding student images using the face_recognition library and storing the encodings along with student IDs in a pickle file. It imports necessary libraries such as cv2 (OpenCV), face_recognition, pickle, os, and firebase_admin. The script initializes Firebase using a service account key and imports student images from a specified folder path. It then iterates through each image, converts it to RGB format, extracts facial encodings using face_recognition library, and stores the encodings along with student IDs in a list. This list is then serialized using pickle and stored in a file named "encodedFile.p". Finally, the script uploads the images to a Firebase Storage bucket for future reference.

```
import cv2
import face_recognition
import pickle
import os
import firebase_admin
from firebase_admin import credentials
from firebase_admin import db
from firebase_admin import storage


cred = credentials.Certificate("serviceAccountKey.json")
firebase_admin.initialize_app(cred , {
```

```
    'databaseURL' : "https://attendance-system-fe3b4-default-
rtdb.asia-southeast1.firebasedatabase.app/",
    'storageBucket' : "attendance-system-fe3b4.appspot.com"

})

# importing the student images

folderImagePath = "Images"
imagePathList = os.listdir(folderImagePath)
studentIds = []

studimgList = []
for path in imagePathList:
    studimgList.append(cv2.imread(os.path.join(folderImagePath,
path)))
    studentIds.append(os.path.splitext(path)[0])


    fileName = f'{folderImagePath}/{path}'
    bucket = storage.bucket()
    blob = bucket.blob(fileName)
    blob.upload_from_filename(fileName)




def findEncodings(imagesList):
    encodedList = []

    for img in imagesList:
        img = cv2.cvtColor(img , cv2.COLOR_BGR2RGB)
        encode = face_recognition.face_encodings(img)[0]
        encodedList.append(encode)

    return encodedList


print("Encoding Started...")


encodeListKnown = findEncodings(studimgList)
print(encodeListKnown)
encodeListKnownwithIds = [encodeListKnown ,  studentIds]
print("Encoding Ended...")
```

```
file = open("encodedFile.p" ,'wb')

pickle.dump(encodeListKnownwithIds , file)
file.close()
print("File generated")
```

APP

This Python script integrates various functionalities for real-time face recognition and attendance management. It utilizes OpenCV, face_recognition, and Firebase libraries to implement features such as face detection, recognition, student data retrieval, attendance marking, and CSV file generation. The script initializes Firebase using a service account key and loads pre-encoded student face data from a pickle file. It then captures frames from the webcam, detects faces, compares them with stored encodings, and marks attendance in real-time. Additionally, it displays student details upon successful recognition and updates the Firebase database with attendance records. The script also includes a graphical interface with a "STOP" button to exit the program. This comprehensive functionality enables efficient attendance management with real-time monitoring and data storage capabilities.

```
import csv
import cv2
import os
import pickle
import face_recognition
import numpy as np
import cvzone
import firebase_admin
from firebase_admin import credentials
from firebase_admin import db
from firebase_admin import storage
from datetime import datetime

cred = credentials.Certificate("serviceAccountKey.json")
firebase_admin.initialize_app(cred, {
```

```
    'databaseURL': "https://attendance-system-fe3b4-default-
rtdb.asia-southeast1.firebasedatabase.app/",
    'storageBucket': "attendance-system-fe3b4.appspot.com"
})

bucket = storage.bucket()

cap = cv2.VideoCapture(0)
cap.set(3, 640)
cap.set(4, 480)

imageBackground = cv2.imread("Resources/background.png")

folderModePath = "Resources/Modes"
modePathList = os.listdir(folderModePath)
imgModeList = [cv2.imread(os.path.join(folderModePath, path)) for
path in modePathList]

file = open('encodedFile.p', 'rb')
encodeListKnownwithIds = pickle.load(file)
file.close()
encodeListKnown, studentIds = encodeListKnownwithIds

modeType = 2
counter = 0
id = -1
imgStudent = []

# Dictionary to track whether attendance has been taken for each
student
attendance_taken = {student_id: False for student_id in studentIds}
show_details_timer = 0  # Timer to control how long to show student
details on the display

# Create a CSV file to store attendance
with open('attendance.csv', 'w', newline='') as file:
    writer = csv.writer(file)
    writer.writerow(['PRN', 'Name', 'Subject', 'Class', 'Semester',
'Attendance Time'])


def stop_program(event, x, y, flags, param):
    if event == cv2.EVENT_LBUTTONDOWN:
```

```
        if 427 <= x <= 686 and 124 <= y <= 183:  # Check if mouse
click is within button coordinates
            exit(0)


cv2.namedWindow('Face recognition')
cv2.setMouseCallback('Face recognition', stop_program)

while True:
    success, img = cap.read()

    imgS = cv2.resize(img, (0, 0), None, 0.25, 0.25)
    imgS = cv2.cvtColor(imgS, cv2.COLOR_BGR2RGB)

    faceCurFrame = face_recognition.face_locations(imgS)
    encodeCurFrame = face_recognition.face_encodings(imgS,
faceCurFrame)

    imageBackground[162:162 + 480, 55:55 + 640] = img
    imageBackground[44:44 + 633, 808:808 + 414] =
imgModeList[modeType]

    for encodeFace, faceLoc in zip(encodeCurFrame, faceCurFrame):
        matches = face_recognition.compare_faces(encodeListKnown,
encodeFace)
        faceDist = face_recognition.face_distance(encodeListKnown,
encodeFace)

        matchIndex = np.argmin(faceDist)

        if matches[matchIndex]:
            y1, x2, y2, x1 = faceLoc
            y1, x2, y2, x1 = y1 * 4, x2 * 4, y2 * 4, x1 * 4
            bbox = 55 + x1, 162 + y1, x2 - x1, y2 - y1
            imageBackground = cvzone.cornerRect(imageBackground,
bbox, rt=0)
            id = studentIds[matchIndex]

            if not attendance_taken[id]:
                attendance_taken[id] = True

                # Get the data of students
                student_info = db.reference(f'Students/{id}').get()
                print(student_info)
```

```
                # Get the Image of the students
                blob = bucket.get_blob(f'Images/{id}.png')
                array = np.frombuffer(blob.download_as_string(),
np.uint8)
                imgStudent = cv2.imdecode(array,
cv2.COLOR_BGRA2BGR)

                # Update the date of attendance
                ref = db.reference(f'Students/{id}')
                ref.child('Last_attendance').set(datetime.now().str
ftime("%Y-%m-%d %H:%M:%S"))

                # Set timer to show student details for 3 seconds
                show_details_timer = 90  # Assuming 30 frames per
second

                # Write attendance to CSV file
                with open('attendance.csv', 'a', newline='') as
file:
                    writer = csv.writer(file)
                    writer.writerow([id, student_info['Name'],
student_info['Subject'], student_info['Class'],
                                     student_info['SEM'],
datetime.now().strftime("%Y-%m-%d %H:%M:%S")])

    # Decrement timer
    if show_details_timer > 0:
        show_details_timer -= 1
    else:
        # Reset student details when timer expires
        modeType = 2
        counter = 0
        id = -1
        imgStudent = []

    if show_details_timer > 0:
        # Show student details on display
        classs = f"Class: {student_info['Class']}"
        prn = f"PRN: {id}"
        sub = f"Sub: {student_info['Subject']}"
        sem = f"Sem: {student_info['SEM']}"
        name = "Name: " + student_info['Name']
```

```
        (w, h), _ = cv2.getTextSize(name, cv2.FONT_HERSHEY_COMPLEX,
1, 1)
        offset = (414 - w) // 2
        cv2.putText(imageBackground, classs, (877, 531),
                    cv2.FONT_HERSHEY_COMPLEX, 1, (50, 50, 50), 2)
        cv2.putText(imageBackground, prn, (877, 492),
                    cv2.FONT_HERSHEY_COMPLEX, 1, (50, 50, 50), 2)
        cv2.putText(imageBackground, sub, (877, 561),
                    cv2.FONT_HERSHEY_COMPLEX, 1, (50, 50, 50), 2)
        cv2.putText(imageBackground, sem, (877, 601),
                    cv2.FONT_HERSHEY_COMPLEX, 1, (50, 50, 50), 2)
        cv2.putText(imageBackground, name, (808 + offset, 445),
                    cv2.FONT_HERSHEY_COMPLEX, 1, (50, 50, 50), 2)
        imageBackground[175:175 + 216, 909:909 + 216] = imgStudent

    # Display the button
    cv2.rectangle(imageBackground, (427, 124), (686, 183), (255, 0,
0), cv2.FILLED)
    cv2.putText(imageBackground, "STOP", (500, 160),
cv2.FONT_HERSHEY_SIMPLEX, 1, (255, 255, 255), 2)

    cv2.imshow("Face recognition", imageBackground)
    cv2.waitKey(1)
```

ADD DATA TO DATABASE

The below script establishes a connection to the Firebase Realtime Database using a service account key. It initializes Firebase and creates a reference to the "Students" node in the database. It then defines a dictionary containing student data, including their name, PRN (Personal Roll Number), subject, class, semester, last attendance date, and total attendance count. Next, it iterates through the dictionary items and sets each student's data as a child node under the "Students" node in the Firebase database. This script efficiently populates the database with student information, enabling seamless integration with the attendance management system.

```
import firebase_admin
from firebase_admin import credentials
from firebase_admin import db
```

```
cred = credentials.Certificate("serviceAccountKey.json")
firebase_admin.initialize_app(cred , {
    'databaseURL' : "https://attendance-system-fe3b4-default-
rtdb.asia-southeast1.firebasedatabase.app/"

})

ref = db.reference('Students')

data = {
       "20010523101":{
        "Name" : "Shivendra Yadav",
        "PRN" : "20010523101",
        "Subject" : "Deep Learning",
        "Class" : "AIML",
        "SEM" : "8",
        "Last_attendance" : "2024-04-22 00:41:35",
        "Total_attendance" : "8"
    },
    "20010523102":{
        "Name" : "Rupesh Jha",
        "PRN" : "20010523102",
        "Subject" : "Deep Learning",
        "Class" : "AIML",
        "SEM" : "8",
        "Last_attendance" : "2024-04-22 00:41:35",
        "Total_attendance" : "8"
    },
    "20010523103":{
        "Name" : "Gaurav Khalase",
        "PRN" : "20010523103",
        "Subject" : "Deep Learning",
        "Class" : "AIML",
        "SEM" : "8",
        "Last_attendance" : "2024-04-22 00:41:35",
        "Total_attendance" : "8"
    }
}


for key,value in data.items():
    ref.child(key).set(value)
```

APPENDIX B

RESEARCH PAPER AND CERTIFICATES

Ethical and Practical Aspects of Facial Recognition in Attendance Management System

Dr. Meesala Sudhir Kumar[1]

[1]Professor, Department of CSE, SOCSE, Sandip University, Nashik (MH), India.

sudhir.meesala@sandipuniversity.edu.in

Mr. Gaurav Khalase[2]

[2]Scholar, SOCSE, Sandip University, Nashik(MH), India.

gauravkhalase12@gmail.com

Mr. Rupesh Jha[3]

[3]Scholar, SOCSE, Sandip University, Nashik(MH), India.

rupeshjha365@gmail.com

Mr. Nitin [4]

[4]Scholar, SOCSE, Sandip University, Nashik(MH), India.

bcs.animesh@gmail.com

Mr. Shivendra Yadav[5]

[5]Scholar, SOCSE, Sandip University, Nashik(MH), India.

shivendra101ss@gmail.com

Mr. Kupakwashe Mapuranga[6]

[5]Scholar, SOCSE, Sandip University, Nashik(MH), India.

kupakwashemapuranga@gmail.com

Under the Guidance of

Dr. Meesala Sudhir Kumar

Professor, Department of CSE, SOCSE, Sandip University, Nashik (MH), India.

sudhir.meesala@sandipuniversity.edu.in

i

SANDIP
UNIVERSITY
UGC Recognised

CERTIFICATE

OF PARTICIPANTS

PROUDLY PRESENTED TO:

Rupesh Jha.

This certifies that has demonstrated commendable dedication and scholarly engagement as a participant in the Sandip University NextGen Computing 2K23 PhD Symposium held on 25th November 2023

This certificate was awarded by:

DR. ANAND SINGH RAJAWAT
conference chair
Sandip University

DR. PAWAN BHALADHARE
Convener
Sandip University

DR. AMOL POTGANTWAR
Director
Sandip University

DR. PRASAD R. BAVISKAR
REGISTRAR
Sandip University

DR. RAJENDRA SINHA
VICE CHANCELLOR
Sandip University

SANDIP UNIVERSITY
UGC Recognised

CERTIFICATE

OF PARTICIPANTS

PROUDLY PRESENTED TO:

Nitin Abhishek Mishra.

This certifies that has demonstrated commendable dedication and scholarly engagement as a participant in the Sandip University NextGen Computing 2K23 PhD Symposium held on 25th November 2023

This certificate was awarded by:

DR. ANAND SINGH RAJAWAT
conference chair
Sandip University

DR. PAWAN BHALADHARE
Convener
Sandip University

DR. AMOL POTGANTWAR
Director
Sandip University

DR. PRASAD R. BAVISKAR
REGISTRAR
Sandip University

DR. RAJENDRA SINHA
VICE CHANCELLOR
Sandip University

SANDIP
UNIVERSITY
UGC Recognised

CERTIFICATE

OF PARTICIPANTS

PROUDLY PRESENTED TO:

Gaurav Khalase

This certifies that has demonstrated commendable dedication and scholarly engagement as a participant in the Sandip University NextGen Computing 2K23 PhD Symposium held on 25th November 2023

This certificate was awarded by:

DR. ANAND SINGH RAJAWAT
conference chair
Sandip University

DR. PAWAN BHALADHARE
Convener
Sandip University

DR. AMOL POTGANTWAR
Director
Sandip University

DR. PRASAD R. BAVISKAR
REGISTRAR
Sandip University

DR. RAJENDRA SINHA
VICE CHANCELLOR
Sandip University

SANDIP
UNIVERSITY
UGC Recognised

CERTIFICATE

OF PARTICIPANTS

PROUDLY PRESENTED TO:

Shivendra Yadav

This certifies that has demonstrated commendable dedication and scholarly engagement as a participant in the Sandip University NextGen Computing 2K23 PhD Symposium held on 25th November 2023

This certificate was awarded by:

DR. ANAND SINGH RAJAWAT
conference chair
Sandip University

DR. PAWAN BHALADHARE
Convener
Sandip University

DR. AMOL POTGANTWAR
Director
Sandip University

DR. PRASAD R. BAVISKAR
REGISTRAR
Sandip University

DR. RAJENDRA SINHA
VICE CHANCELLOR
Sandip University

SANDIP
UNIVERSITY
UGC Recognised

CERTIFICATE

OF PARTICIPANTS

PROUDLY PRESENTED TO:

Kupakwashe Mapuranga

This certifies that has demonstrated commendable dedication and scholarly engagement as a participant in the Sandip University NextGen Computing 2K23 PhD Symposium held on 25th November 2023

This certificate was awarded by:

DR. ANAND SINGH RAJAWAT
conference chair
Sandip University

DR. PAWAN BHALADHARE
Convener
Sandip University

DR. AMOL POTGANTWAR
Director
Sandip University

DR. PRASAD R. BAVISKAR
REGISTRAR
Sandip University

DR. RAJENDRA SINHA
VICE CHANCELLOR
Sandip University

APPENDIX C

CURRICULUM VITAE OF GROUP MEMBERS

NAME OF STUDENT **MR. RUPESH JHA**

• **PERMANENT ADDRESS** : Room no-12, Tulja appt., Radhakrishna Nagar , Ashok Nagar, Nashik, Maharashtra,India

MOB NO : +917620153337 **EMAIL ID :** rupeshjha365@gmail.com

LINKEDIN : linkedin.com/in/rupesh-jha-

CAREER OBJECTIVE

I am an aspiring DevOps Engineer with a profound interest in cloud technology, automation, artificial intelligence and machine learning. I am a fast learner and committed to give my best to every task I perform.

ACADEMIC CREDENTIALS

Sr.No	DESCRIPTION	BOARD/UNIVERSITY	YEAR	PERCENTAGE/C.G.P.A
1	10th	SSC	2018	83.60
2	12th	HSC	2020	69.85
3	B.Tech	SANDIP UNIVERSITY (UGC)	2024	8.30

AREAS OF INTEREST

- Cloud
- DevOps
- Generative AI
- Blockchain
- Artificial Intelligence
- Machine Learnign

SKILLS & ABILITIES

- Linux
- Docker
- Kubernets
- Cloud Computing
- Scikit Learn
- Tensorflow

DISTINCTIVE HIGHLIGHTS

- Backend Developer (Radiance Innovation)
- DevOps Inern Engineer (Mactores Cognition)

Certification

- AWS Certified DevOps Engineer Professional
- AWS Certified Solution Architect Associate
- Google Cloud Certified Associate Cloud Engineer
- Google Certifed Automation in IT Using Python

DECLARATION

I hereby declare that all the information given above is true and correct to the best of my knowledge.

NAME OF STUDENT

MR. Nitin

• **PERMANENT ADDRESS** : Ramnagar, Behind K.P Market, Purnia,Bihar,854303, India

MOB NO : 7206220843 **EMAIL ID :** bcs.animesh@gmail.com

LINKEDIN : http://www.linkedin.com/in/nitin-mishra-2bb993268

CAREER OBJECTIVE

I am a self-motivated computer science engineer with a strong proficiency in coding and specialized expertise in Anaplan. I am dedicated to the thorough completion of tasks, demonstrating a commitment to achieving objectives from a strategic standpoint.

ACADEMIC CREDENTIALS

Sr.No	DESCRIPTION	BOARD/UNIVERSITY	YEAR	PERCENTAGE/C.G.P.A
1	**10th**	**CBSE**	**2018**	88%
2	**12th**	**CBSE**	**2020**	71.8%
3	**B.Tech**	**SANDIP UNIVERSITY (UGC)**	**2024**	8.84

AREAS OF INTEREST

- Python
- Machine Learning
- Artificial Intelligence
- Automation
- Management
- Office work

SKILLS & ABILITIES

- Python
- Pandas
- Scikit-Learn
- Git
- Anaplan
- Interpersonal skill

DISTINCTIVE HIGHLIGHTS

- Intern (Radiance Innovation)

Certification

• Introduction to Python (INFOSYS) • Application Of Artificial Intelligence & Data Science in Python (WINJIT TECHNOLOGIES, NASHIK) • Data Science And Machine Learning (MITU SKILLOGIES)
• Google IT Automation with Python (COURSERA) • Anaplan Level 1 Model Building (ANAPLAN)

DECLARATION

I hereby declare that all the information given above is true and correct to the best of my knowledge.

NAME OF STUDENT

MR. GAURAV KHALASE

- **PERMANENT ADDRESS** : Plot No. 32, Sai Nagar, Vilhale Road, Varangaon, Dist. Jalgaon, Maharashtra.

MOB NO : 9112458099 **EMAIL ID :** gauravkhalase@gmail.com

LINKEDIN : https://www.linkedin.com/in/gaurav-khalase

CAREER OBJECTIVE

Recent graduate with a degree in Computer Science and a passion for building seamless and functional user interfaces. Seeking a role to apply exceptional ability to switch between programming languages and frameworks as needed, while working quickly and independently.

ACADEMIC CREDENTIALS

Sr.No	DESCRIPTION	BOARD/UNIVERSITY	YEAR	PERCENTAGE/CGPA
1	10th	SSC	2018	91.20 %
2	12th	HSC	2020	81.54 %
3	B.Tech	SANDIP UNIVERSITY, NASHIK	2024	8.54 CGPA

TECHNICAL SKILLS

- HTML
- CSS
- JavaScript
- JAVA
- Tailwind CSS
- ReactJS
- NodeJS
- ExpressJS
- MongoDB

EXPERIANCE

- Front End Development Internship (Radiance Innovations)
- MERN Stack Internship (Paarsh Infotech)

CERTIFICATION

- MERN Stack Certification (CodeHelp)

DECLARATION

I hereby declare that all the information given above is true and correct to the best of my knowledge.

NAME OF STUDENT

MR. SHIVENDRA YADAV

• **PERMANENT ADDRESS** : Sa-5/159 Sanjay nagar colony paharia Varanasi Uttar Pradesh, India

MOB NO : 7275775322 **EMAIL ID :** Shivendra101ss@gmail.com

LINKEDIN : linkedin.com/in/shivendra-yadav-7a2905187

CAREER OBJECTIVE

I am a self-motivated computer science engineer with a profound interest in technology, management and an unwavering commitment to completing tasks without prematurely discontinuing them from a perspective standpoint.

ACADEMIC CREDENTIALS

Sr.No	DESCRIPTION	BOARD/UNIVERSITY	YEAR	PERCENTAGE/C.G.P.A
1	**10th**	**CBSE**	**2017**	**72 %**
2	**12th**	**CBSE**	**2019**	**65 %**
3	**B.Tech**	**SANDIP UNIVERSITY (UGC)**	**2024**	**8.2 GPA**

AREAS OF INTEREST

- Python
- C
- Digital Marketing
- Web Development
- Management
- Office work

SKILLS & ABILITIES

- Python
- Front-end development
- Interpersonal skill
- Business Development
- Digital Marketing
- MS Office

DISTINCTIVE HIGHLIGHTS

- Hr internship (Smm Foundation) , Data analytics internship (Kpmg) , Business development internship(Younity India)

Certification

- Python From (Infosys) • IT Support From (Coursera) • Digital Marketing From (Coursera)
- Business Intelligence From (Coursera)

DECLARATION

I hereby declare that all the information given above is true and correct to the best of my knowledge.

NAME OF STUDENT : KUPAKWASHE T MAPURANGA

PERMANENT ADDRESS: 5040 Manyame Park Chitungwiza, Harare, Zimbabwe

MOB NO: 8080520097 **EMAIL ID:** kupakwashemapuranga@gmail.com

LINKEDIN: www.linkedin.com/in/kupakwashe-mapuranga-18b6a2276

CAREER OBJECTIVE

I am a visionary leader and technocrat, passionate about making a positive impact in the world. With a keen sense of purpose, I aim to leverage my expertise in technology and leadership to drive meaningful change and uplift lives. My commitment to excellence and dedication to my vision for a better Africa underscore my role as a catalyst for progress and innovation. As a leader, I strive to inspire others and create a lasting legacy of empowerment and transformation

ACADEMIC CREDENTIALS

Sr.No	DESCRIPTION	BOARD/UNIVERSITY	YEAR	PERCENTAGE/C.G.P.A
1	**10th**	**ZIMSEC**	**2017**	86.2%
2	**12th**	**ZIMSEC**	**2019**	92%
3	**B.Tech**	**SANDIP UNIVERSITY (UGC)**	**2024**	**8.67 CGPA**

AREAS OF INTEREST

- Python
- Artificial Intelligence
- Machine Learning
- Web Development
- Data Science
- Public speaking

SKILLS & ABILITIES

- Python programming
- Data Manipulation and Analysis:
  - Pandas
  - NumPy
- Machine Learning and Statistical Modelling:
  - Scikit-learn
- Data Visualization:
  - Matplotlib
  - Seaborn
- Deep Learning
  - TensorFlow
  - Keras
- HTML, CSS, JS

DISTINCTIVE HIGHLIGHTS

- Operations direct for Digiworkx, Gaborone, Botswana & AI consultant for Ba Isago University

Certification

- Python From (Infosys) • Data Science from (Mitu Skillilogies), cyber security from (Konsola)

DECLARATION

I hereby declare that all the information given above is true and correct to the best of my knowledge.