**DEVELOPMENT OF A FACIAL RECOGNITION BASED ATTENDANCE SYSTEM (A Case Study OF Afe Babalola University, Ado-Ekiti, Nigeria)**

**OKONKWO PASCHAL CHIEMERIE**

**MATRIC NUMBER: 20/SCI01/093**

**MAY, 2024**

**DEVELOPMENT OF A FACIAL RECOGNITION BASED ATTENDANCE SYSTEM (A Case Study OF Afe Babalola University, Ado-Ekiti, Nigeria)**

**BY: OKONKWO PASCHAL CHIEMERIE**

**MATRIC NUMBER: 20/SCI01/093**

**A PROJECT REPORT SUBMITTED TO COMPUTER SCIENCE PROGRAMME, DEPARTMENT OF MATHEMATICAL AND PHYSICAL SCIENCES, COLLEGE OF SCIENCES, AFE BABALOLA UNIVERSITY,**

**ADO-EKITI, NIGERIA.**

**IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF**

**BACHELOR OF SCIENCE (B.Sc.) DEGREE IN COMPUTER SCIENCE**

**MAY, 2024**

# **DECLARATION**

I, (OKONKWO, Paschal Chiemerie 20/SCI01/093), hereby declare that this project report was written by me and is of my own research work. It has not been presented in any previous application for any degree of this or any other university. All citations and sources of information are clearly acknowledged by means of references.

**SIGNATURE: ................................ DATE: ................................**

# **CERTIFICATION**

This is to certify that the project titled **“DEVELOPMENT OF A FACIAL RECOGNITION BASED ATTENDACE SYSTEM (A Case Study OF Afe Babalola University, Ado-Ekiti, Nigeria)”** has submitted to the Department of Computer Science, Afe Babalola University, for the award of Bachelor of Science (B.Sc.) in Computer Science is original carried out by **OKONKWO, Paschal Chiemerie** with Matriculation Number **20/SCI01/093** in the Department of Computer Science, Faculty of Physical and Mathematical Sciences, College Of Science, Afe Babalola University Ado-Ekiti, Ekiti State.

**................................ ................................**

**Dr. Opani Aweh DATE**

**(Supervisor)**

**................................ ................................**

**Dr. Oyelami Funmilayo DATE**

**(Head of Department)**

# **DEDICATION**

This project is dedicated to the divine Creator, who has guided my journey from its inception to this moment. With gratitude, I extend my appreciation to my remarkable parents, Mr. Okonkwo Paschal Chidi and Mrs. Okonkwo Nneka Vivian. Their unwavering support and kindness have been my compass throughout my studies.

In addition, I honor my cherished siblings: Okonkwo Kingsley Chiadika, Okonkwo Angel Chidumebi, and Okonkwo Millicent Chisimdi. Their encouragement and love have fueled my determination.

May this dedication serve as a testament to the power of family, faith, and perseverance.

# **ACKNOWLEDGEMENT**

I extend my deepest appreciation to the Almighty God, whose unwavering light has illuminated my path and granted me boundless strength and grace during my tenure at Afe Babalola University. His divine guidance has been my compass.

A special debt of gratitude is owed to Dr. Opani Aweh, my dedicated supervisor. His consistent availability, unwavering support, and expert guidance have been indispensable in bringing this project to fruition.

I also wish to express our utmost thanks to the following individuals, whose contributions enriched our academic journey:

Head of the Department of Mathematical and Physical Sciences: Your leadership and commitment to excellence have inspired me.

Program’s Provost: Your vision and dedication which have shaped my learning experience.

Level Advisor: Your personalized guidance has been invaluable.

Diligent Lab Attendants: Your assistance ensured smooth experiments and practical sessions.

Esteemed Lecturers: Your knowledge-sharing and mentorship have left a lasting impact.

Last but not least, I acknowledge the valuable input of friends, coursemates, and well-wishers your collaboration and insights have significantly enriched this project.

May this acknowledgment serve as a testament to the power of collaboration, resilience, and community.

# **TABLE OF CONTENTS**

[DECLARATION iii](#_Toc167779229)

[CERTIFICATION iv](#_Toc167779230)

[DEDICATION v](#_Toc167779231)

[ACKNOWLEDGEMENT vi](#_Toc167779232)

[TABLE OF CONTENTS vii](#_Toc167779233)

[TABLE OF FIGURES ix](#_Toc167779234)

[LIST OF TABLES x](#_Toc167779235)

[ABSTRACT xi](#_Toc167779236)

[CHAPTER ONE 1](#_Toc167779237)

[**1.1.** **Background to the Study:** 1](#_Toc167779238)

[**1.2.** **Statement of the Problem** 2](#_Toc167779239)

[**1.3.** **Research Aim and Objectives** 3](#_Toc167779240)

[**Aim: Development of a Facial Recognition-Based Identification System** 3](#_Toc167779241)

[**Objectives**: 3](#_Toc167779242)

[**1.4** **Project Methodology** 4](#_Toc167779243)

[**Research approach:** 4](#_Toc167779244)

[**1.5** **Scope of the Study** 6](#_Toc167779245)

[**1.6** **Keywords and their Meanings** 7](#_Toc167779246)

[CHAPTER TWO 9](#_Toc167779247)

[LITERATURE REVIEW 9](#_Toc167779248)

[**2.1 Historical Progression of Facial Recognition Technology:** 9](#_Toc167779249)

[**2.2 Transition from Rudimentary Models to Advanced Algorithm:** 9](#_Toc167779250)

[**2.3 Utilization of Traditional and Modern Techniques:** 9](#_Toc167779251)

[**2.4 Attendance tracking** 10](#_Toc167779252)

[**2.5 Facial Recognition-Based Systems in Education** 10](#_Toc167779253)

[OTHER TOOLLS that can also be used for attendance system include: 13](#_Toc167779254)

[**2.5 RELATED WORKS** 13](#_Toc167779255)

[CHAPTER THREE 19](#_Toc167779256)

[**SYSYTEM ANALYSY AND DESIGN**, 19](#_Toc167779257)

[CHAPTER FOUR 32](#_Toc167779258)

[**IMPLEMENTATION AND TESTING** 32](#_Toc167779259)

[The hardware and software tools employed 32](#_Toc167779260)

[CHAPTER FIVE 36](#_Toc167779261)

[**SUMMARY, CONCLUSION AND RECOMMENDATION** 36](#_Toc167779262)

[**SUMMARY**: 36](#_Toc167779263)

[**RECOMMENDATIONS:** 36](#_Toc167779264)

[**CONCLUSION:** 37](#_Toc167779265)

[REFERENCE 39](#_Toc167779266)

[APPENDIX 42](#_Toc167779267)

# **TABLE OF FIGURES**

**Figure 1** 19

**Figure 2** 20

**Figure 3** 25

**Figure 4** 26

**Figure 5** 33

**Figure 6** 34

# **LIST OF TABLES**

[**Table 1** 34](#_Toc167775084)

[**Table 2** 35](#_Toc167775085)

# **ABSTRACT**

**Attendance recording is one of the core features of systems in academic institutions. In most institutions, marks are allocated to class attendance, but recording this attendance remains an area of concern, especially in very large classes like general courses or courses with large enrollment.**

**Existing approaches to taking attendance have various shortcomings. The paper system is time-consuming, and many students sign up for their absent classmates or friends. If the lecturer decides to take attendance personally, a great deal of time is expended. Furthermore, approaches like fingerprint and card-based systems used in some institutions are not much better. The fingerprint system is even more time-consuming than the manual method in practice, and students can give their cards to their fellow students to swipe for them.**

**To address this problem, this study combines facial recognition technology and closed-circuit camera systems to develop an attendance system that addresses these issues. The developed system uses the Python programming language and an open-source API that integrates different cameras into one application. The results obtained from the tests show that the desired features were achieved.**

# **CHAPTER ONE**

**INTRODUCTION**

## **Background to the Study:**

Facial recognition technology had its early beginnings in the 1960s when pioneers like Woody Bledsoe, Helen Chan Wolf, and Charles Bisson conducted initial experiments in this field. These trailblazers concentrated on utilizing computers to identify human faces by manually marking specific facial "landmarks" like the positions of the eyes and mouth (Bledsoe, W. W. et al., 1966). These landmarks were then subjected to mathematical processing by computers to adjust for variations in pose, thereby establishing a solid groundwork for future advancements in facial recognition. Moving into the 1970s, researchers Goldstein, Harmon, and Lesk made notable progress by expanding facial recognition to include 21 distinct subjective markers, such as hair color and lip thickness. Even though measurements and locations still necessitated manual computation, this represented a step forward from previous endeavors and contributed to the enhancement of techniques employed in facial recognition (Goldstein, A. J. et al., 1971).

The late 1980s and 1990s saw the introduction of linear algebra techniques to facial recognition by researchers Sirovich and Kirby, which led to the creation of the Eigenface method (Sirovich, L. & Kirby, M., 1987). This method showcased that feature analysis on a compilation of facial images could produce a set of fundamental features, with less than one hundred values being adequate to accurately encode a normalized facial image (Turk, M. & Pentland, A., 1991). Turk and Pentland subsequently progressed this work by devising techniques for automatically detecting faces within images, marking significant strides towards the realization of automatic facial recognition systems (Turk, M. & Pentland, A., 1991). Fast-forwarding to the contemporary era, facial recognition has become an essential component of various industries and applications (Jain, A. K. et al., 2011). It is extensively applied in law enforcement for surveillance and identification objectives, in border control for passport and visa authentication, in retail for personalized marketing and security, in mobile technology for biometric authentication, and in the realm of banking and finance for the prevention of fraud and customer verification (Phillips, P. J. et al., 2000).

As technology advances further, we can anticipate even more remarkable developments in the domain of facial recognition. Progressions in artificial intelligence, deep learning, and computer vision algorithms are augmenting the accuracy, speed, and dependability of facial recognition systems. Moreover, ongoing research endeavors are centered on tackling ethical and privacy issues linked with the widespread integration of facial recognition technology, ensuring its responsible and ethical utilization in the times ahead (Wang, Y. & Deng, W., 2021).

## **Statement of the Problem**

The issue of attendance with the rising population of students, especially in lower levels where students take general courses together, is increasingly becoming an issue of great concern. Manual attendance-taking can take the time allotted for the lecture. Also, some students can easily mark the names of their course mates (Ponce, H. R., & Rovira, F. E., 2020).

The use of biometric methods such as fingerprint machines will not likely improve the process because of the time it would take for students to thumbprint. And even increasing the number of machines may likely not help address the problem (Dhanda, P., & Arora, S., 2019).

Consequently, with the advances recorded in the area of surveillance in some countries like China, where the crowd trooping along the street is recognized by close circuit cameras (CCTV), it means that this technology, if properly developed, can be used for attendance taking (Li, Z., & Zhu, H., 2019). This is what this study intended to address.

Primarily, the substantial influx of individuals traversing in and out of organizations poses a formidable obstacle for manual monitoring systems. As this influx continues to rise, the task of accurately monitoring each individual becomes increasingly arduous and, eventually, reaches a point where it becomes virtually impossible for human operators to adequately handle (Saini, R., & Kumar, M., 2021).

## **Research Aim and Objectives**

**Aim: Development of a Facial Recognition-Based Identification System**

This study aims to develop a facial recognition-based attendance-taking system. Tailored specifically for ABUAD (Afe Babalola University, Ado-Ekiti, Nigeria). This system will enhance identification processes within the university community.

### **Objectives**:

1. to develop an algorithm that will match captured facial features, with saved facial features
2. to develop a database for using SQLite, to store facial encodings and to record attendance
3. to implement and test the facial recognition system with the attendance system using the python programing language
4. to test the level of accuracy of the system.

## **Project Methodology**

The development of a sophisticated tracking system requires a systematic methodology that encompasses various stages from conceptualization to deployment.

The chapter describes the research approach and the detailed system development approach used in the study.

### **Research approach:**

Research approaches adopted in this study are documents perusal, observations and consultations.

I. Document Perusal: A thorough examination of relevant literature on attendance monitoring methods and the incorporation of facial recognition technology yielded significant insights for the creation and execution of the proposed system.

II. Observation: Notable inefficiencies in traditional attendance procedures, such as manual signing on a paper register, were revealed through observations as a student in a large class, which proved to be time-consuming.

Furthermore, the exploration of fingerprint scanners demonstrated only marginal enhancements in efficiency.

III. Consultation: Extensive discussions were held with experts from various fields, encompassing academics, CCTV setup and maintenance, and software development. Interactive dialogues with peers and instructors within the academic realm highlighted the strengths and weaknesses of existing attendance systems and opportunities for improvement. Professionals in CCTV setup and maintenance provided valuable perspectives on integrating CCTV cameras for attendance monitoring, including camera selection and optimal placement. Software developers played a crucial role in creating the application that smoothly integrates CCTV capabilities into the attendance system.

It based on these knowledge that the detailed design on the system was based. These detailed design are now presented in a form of a narration

1. Data Collection: This involves the gathering of facial images of students depicting various individuals for recognition. This task includes the collection of intricate facial characteristics, which are then stored within a designated database, along with pertinent identification particulars such as names and student matric numbers.
2. Feature Extraction: A paramount step within the process is the utilization of the face\_recognition API library to extract intricate facial features from the gathered images. The subsequent process involves the conversion of these extracted features into numerical representations, meticulously tailored for comparison and subsequent identification processes.
3. Algorithm Development: This crucial phase entails the development of a meticulously crafted algorithm designed to seamlessly match the captured facial features with the stored facial features within the designated database. Furthermore, the implementation of intricate algorithms specifically tailored for the comparison and identification of facial features is paramount.
4. Database Design: The designing process is embarked upon to craft a database schema that is specifically tailored to efficiently store attendance records. This schema includes critical components such as timestamps, student identification numbers, and the corresponding attendance statuses.
5. Database Implementation: The subsequent step involves the creation of a robust database management system, in this case, SQLite, crafted to house the attendance data effectively. Furthermore, the development of designed scripts or applications is undertaken to seamlessly interact with the database for both attendance recording and retrieval purposes.
6. System Integration: The seamless integration of the facial feature matching algorithm with the attendance database system ensures a harmonious and efficient operational synergy.

## **Scope of the Study**

The scope of the study encompasses the design, development, and evaluation of a facial recognition based attendance system using Afe Babalola University as a case study.

The study will focus to ensure a comprehensive understanding and implementation of the system, and attendance taking in the class room

The scope includes, but is not limited to, the following:

Algorithmic Integration

- Implementing a tracking algorithm that leverages computer vision techniques to accurately track individuals in class room environments. This involves addressing challenges such as occlusions, changes in lighting conditions, and different camera perspectives.

Machine Learning Integration:

- Integrating machine learning models into the tracking system to enhance its adaptability and accuracy over time. This includes developing algorithms that can learn and improve tracking performance based on patterns and behaviors observed in the data.

Real-time Processing:

- Optimizing the system for real-time processing to ensure minimal latency in tracking individuals' movements and activities. This involves addressing challenges associated with handling continuous streams of data from multiple sources.

## **Keywords and their Meanings**

**Facial Recognition**: The process of identifying or verifying individuals by analyzing and comparing patterns based on their facial features.

**Video Surveillance**: The monitoring and recording of activities, behavior, or events through the use of video cameras.

**Tracking System**: A system designed to monitor and keep track of the movement or location of objects, people, or assets.

**CCTV (Closed-Circuit Television)**: A system of video cameras used for surveillance purposes, typically in public areas or private properties.

**Graphical User Interface (GUI)**: A visual interface that allows users to interact with electronic devices or software applications using graphical elements such as icons, buttons, and menus.

**Database Management**: The process of organizing, storing, and managing data in a structured format within a database system.

**SQLite**: A lightweight, server less relational database management system (RDBMS) that is embedded within the application and does not require a separate server process.

**Real-Time**: Refers to processing or monitoring data or events as they occur, with minimal delay or latency.

**Accuracy**: The degree of closeness of measurements or observations to the true or expected value.

**Precision**: The degree of refinement in the performance or measurement of a system, indicating the level of consistency or repeatability.

**Recall:**  In the context of machine learning or information retrieval, recall measures the ability of a system to retrieve relevant items from a dataset.

**Optimization**: The process of improving or enhancing the efficiency, performance, or effectiveness of a system or process.

**Visualization**: The representation of data or information in visual form, such as charts, graphs, or diagrams, to aid understanding and analysis.

**Reporting**: The generation and presentation of structured information or summaries derived from data analysis, typically for decision-making or communication purposes.

# **CHAPTER TWO**

# **LITERATURE REVIEW**

## **2.1 Historical Progression of Facial Recognition Technology:**

The historical progression of facial recognition technology has been extraordinary, evolving from basic geometric models to advanced algorithms capable of identifying individuals in various environments with great accuracy. Initially, facial recognition relied on simple geometric patterns and manual extraction of features. Despite this, significant advancements in computer vision and machine learning have propelled this technology forward, leading to enhancements in accuracy and speed, enabling facial recognition systems to perform effectively across different scenarios.

## **2.2 Transition from Rudimentary Models to Advanced Algorithm:**

The evolution of facial recognition has been marked by a transition from rudimentary geometric models to advanced algorithms capable of identifying individuals in diverse and unconstrained environments. The initial reliance on simple geometric patterns and manual feature extraction has given way to significant advancements driven by computer vision and machine learning techniques. Advancements in technology have brought about notable improvements in the accuracy and speed of facial recognition systems, allowing them to function robustly in various settings. (Andrejevic et al .2020)

## **2.3 Utilization of Traditional and Modern Techniques:**

Traditional methods such as 2D and 3D modeling capture facial geometry from various angles to enhance recognition accuracy. In addition, neural networks and deep learning algorithms have emerged as powerful tools in facial recognition, utilizing large datasets to automatically learn discriminative features and address challenges posed by factors like pose variations, changes in illumination, and facial expressions. (Andrejevic et al .2020) Classic techniques like 2D and 3D modeling are utilized to capture facial structure from different angles to enhance recognition accuracy.

## **2.4 Attendance tracking**

Attendance tracking within educational institutions has historically represented a pivotal administrative duty that plays a crucial role in overseeing student involvement, evaluating engagement levels, and ensuring adherence to attendance regulations. Conventional methodologies for recording attendance, such as manual roll calls and paper-based systems, not only demand significant time and effort but also exhibit susceptibility to errors and security breaches, notably including proxy attendance.

Over the past few years, there has been a noticeable surge in the exploration of utilizing technological progressions to streamline the processes involved in attendance tracking and improve the precision of such procedures. One particularly promising strategy involves the creation of facial recognition-based identification systems, which make use of intricate algorithms to automate the process of recording attendance by employing facial biometric data.

## **2.5 Facial Recognition-Based Systems in Education**

Facial recognition technology has made its way into various aspects of public life, including educational institutions.

**Applications in Schools**:

In educational settings, the utilization of facial recognition technology plays a significant role, particularly in the domain of campus security. This technology is implemented to bolster security measures on school premises by accurately identifying authorized personnel and effectively detecting potential threats that may compromise the safety of students and staff members (Andrejevic ET AL.2020).

An innovative application of facial recognition technology in schools is the automation of roll calls. Rather than relying on traditional manual methods of taking attendance, educational institutions have adopted facial recognition systems that automatically record student attendance based on distinct facial features, streamlining administrative processes and enhancing efficiency within the academic environment.

Moreover, another noteworthy application of facial recognition technology in schools pertains to the monitoring of student emotions and attention levels. (Andrejevic et al.2020) Some advanced systems are designed to analyze the facial expressions of students, providing insights into their emotional states and engagement levels during classroom sessions, which can be valuable for educators in tailoring their teaching approaches to better support student learning and well-being.

From a technological standpoint, the operational framework of facial recognition systems in schools involves various key components. Firstly, schools engage in the collection of facial images of students to facilitate the training of the recognition system, ensuring the accuracy and reliability of facial identification processes. Subsequently, facial features such as landmarks, textures, and shapes are encoded to generate a unique representation for each student, enabling precise recognition and differentiation among individuals within the school community.

Furthermore, the real-time detection and recognition capabilities of facial recognition systems are facilitated through the utilization of sophisticated algorithms like convolutional neural networks (CNNs), enabling swift and accurate identification of faces within dynamic school environments. Additionally, the integration of these systems with databases enables seamless linkage of attendance records with individual student profiles, fostering efficient record-keeping and management practices within educational institutions.

The adoption of facial recognition technology in schools offers a myriad of benefits to stakeholders within the educational ecosystem. By automating attendance tracking processes, educators experience significant time savings, alleviating administrative burdens and allowing them to focus more on instructional responsibilities and student engagement. Moreover, the implementation of facial recognition technology enhances the accuracy of attendance records, effectively minimizing errors and curbing instances of proxy attendance, thereby promoting accountability and transparency within school settings.

Furthermore, the real-time monitoring capabilities of facial recognition systems empower educators to provide proactive support to students as needed, enabling timely interventions and personalized assistance to enhance student learning outcomes. This real-time monitoring feature equips teachers with valuable insights into student behaviors and engagement levels, facilitating targeted interventions and support mechanisms to address individual learning needs and promote academic success.

Despite the promising advantages of facial recognition technology in schools, there exist critical perspectives and concerns that warrant careful consideration. Some scholars raise questions regarding the broader implications of integrating facial recognition technology in educational settings, expressing concerns that such technologies may inadvertently reshape the nature of education along divisive, authoritarian, or oppressive lines. Moreover, ethical considerations surrounding privacy and data security must be thoroughly examined and addressed to ensure the responsible and ethical implementation of facial recognition technology in schools, safeguarding the rights and well-being of students and staff members within the educational community.

### OTHER TOOLLS that can also be used for attendance system include:

RFID: RFID is a technology that uses radio waves to transmit information from an electronic tag or label attached to an object through a reader to identify and track the object (Shah & Abuzneid, 2019).

## **2.5 RELATED WORKS**

**Rong Fu et al. 2018,** introduced a pair of deep learning methodologies for the purpose of tracking attendance in university classrooms. These methodologies consist of the MTCNN face detection and Center-Face face recognition algorithms. The system is designed to incorporate supplementary functionalities such as monitoring student absences, tardiness, and early departures. Rapid marking of attendance for all students is ensured by this system. The research reported a high accuracy rate of 92.98% and is capable of recording attendance within a mere 100 milliseconds.

**Jain, A., & Singh, P et al. 2014,** this study investigates the use of biometric systems, particularly facial recognition, for managing attendance in educational settings. The primary objective is to enhance the precision and effectiveness of attendance monitoring through the substitution of manual procedures with automated biometric systems. The methodology included the collection of facial images and other biometric data from students, the creation of an algorithm to match captured features with stored information, and the implementation of the system in an educational environment to evaluate its precision and effectiveness. Verification of the system's performance in recording attendance was also carried out.

**Kumar, S.et al. 2015,** this manuscript introduces an automated attendance system that relies on face recognition technology to streamline the attendance process and minimize errors typically associated with manual recording. The strategy involved the use of the OpenCV library for the creation of a facial recognition algorithm, assembly of a database containing student pictures, and testing of the system in a classroom setting. Assessment of the system's effectiveness was based on its accuracy and processing speed.

**Sharma, M. et al. 2012,** the research offered here utilizes Principal Component Analysis (PCA) in the development of a facial recognition-focused attendance system with the main goal of improving the reliability and effectiveness of attendance monitoring in academic settings. The methodology involved the implementation of PCA for feature extraction and face recognition, alongside the collection of a student image dataset, algorithm development, and real-world system testing. Evaluation of the system's accuracy and efficiency was a key component of the research.

**Gupta, A., & Kaur, K. et al 2016,** the discourse outlined in this paper deliberates on the design of a real-time face recognition system tailored for attendance monitoring, with a keen focus on augmenting the speed and precision of attendance recording through the application of advanced facial recognition methodologies. The methodology adopted by the authors involved the utilization of the Viola-Jones algorithm for face detection and Eigen faces for recognition purposes. The system implementation was executed using MATLAB and tested on a dataset containing student images, with performance evaluation centered on real-time processing capabilities and accuracy.

**Raj, R., & Kumar, P.et al. 2017,** this research examines the implementation of a facial recognition attendance system using Raspberry Pi, aiming to provide an affordable and effective approach to managing attendance. The method involved developing a facial recognition algorithm using OpenCV and Python, integrating it with Raspberry Pi, and evaluating the system's performance in an educational setting.. Evaluation of the system's accuracy and processing speed was integral to the study. The reference for this work is in the Journal of Electronics and Communication Engineering, 12(4), 40-45.

**Jha, Abhishek et al. 2013**, examines the deployment of an operational and dependable Automated Attendance System through the utilization of facial recognition technology within educational settings with the aim of enhancing efficiency in the recording and assessment of attendance. The methodology encompasses the utilization of color-based detection and Principle Component Analysis (PCA) for the purpose of face detection and feature extraction within the context of creating an automated attendance system employing facial recognition technology in an educational setting. The implementation of this system is carried out using MATLAB.

**Shreyak Sawhney et al. 2019** proposed an automated attendance management system consisting of two databases, a student database and an attendance database [ShreyakSawhney et al. 2019]. This device features a high-resolution camera mounted outside the classroom to ensure attendance during grading. Students can access the classroom by scanning their faces with the camera. Another camera is mounted in the classroom so that the camera lens is visible to every student in the class. This system uses Viola and Jones algorithm for face recognition and principal component analysis for face recognition. This device is designed to provide a high level of security and also helps to avoid the risk of powers of attorney and fake presence.

**Verma & Kumar, et al. 2013** The analysis in this study focuses on the utilization of facial recognition technology in improving classroom attendance systems, with the primary objective being the automation of attendance procedures and the reduction of manual errors. The methodology employed by the authors involved a blend of face detection and recognition algorithms, system development through Python, and real-time testing within a classroom environment, with the system's efficacy gauged in terms of accuracy and efficiency

**Ahmed & Hossain, et al. 2018,** a specialized face recognition-based attendance system tailored for classroom settings is detailed in this paper, focusing on streamlining attendance processes and enhancing precision. The researchers crafted a face recognition algorithm using OpenCV, compiled a student image dataset, and carried out system trials in a classroom scenario, scrutinizing accuracy levels and processing speed

**Patel & Sharma, et al. 2011**, Delving into the realm of image processing techniques, this research explores the creation of an automated attendance system, aspiring to supplant conventional methods with a more precise and effective alternative. The methodology entailed the utilization of image processing techniques for face detection and recognition, system development via MATLAB, and testing on a student image dataset, with performance gauged in terms of accuracy and efficiency

**(Das & Paul, et al. 2013).**The investigation of Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) is the primary focus of this research, with the goal of designing a facial recognition-based attendance system for educational settings in order to improve the accuracy and efficiency of attendance monitoring.. Employing PCA and LDA for feature extraction and recognition, the researchers collected a student image dataset and conducted system tests in a practical environment, assessing accuracy and efficiency levels

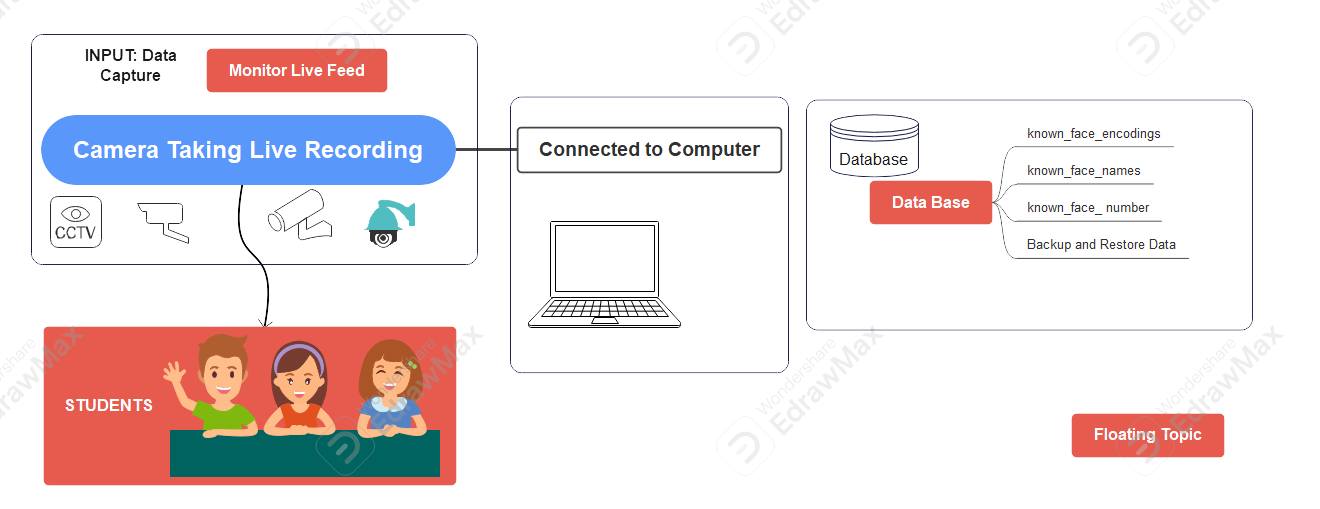
**Singh & Kaur, et al. 2010**, This paper explores the development of a face recognition-based attendance management system, aimed at providing a more reliable and efficient method for tracking attendance in educational settings. The authors devised a face recognition algorithm using OpenCV, compiled a student image dataset, and field-tested the system within a classroom setup, evaluating accuracy and processing speed metrics

**(S Sreesuba, G Anitha et al. )**The discussion in the paper pertains to the advancement of a facial recognition-driven automated attendance tracking system designed to enhance the precision and effectiveness of attendance supervision within academic establishments. The primary objective of the system is to address the constraints associated with conventional attendance procedures and offer a dependable and feasible alternative for monitoring training and placement attendance. The methodology entails the utilization of Multi-Task Cascaded Neural Network (MTCNN) for the purpose of detecting faces, alongside the implementation of the FaceNet algorithm for facial recognition. FaceNet serves the function of extracting characteristics from facial images, converting image data into 128-dimensional features, and subsequently categorizing them through employment of an SVM classifier.

# **CHAPTER THREE**

### **SYSYTEM ANALYSY AND DESIGN**,

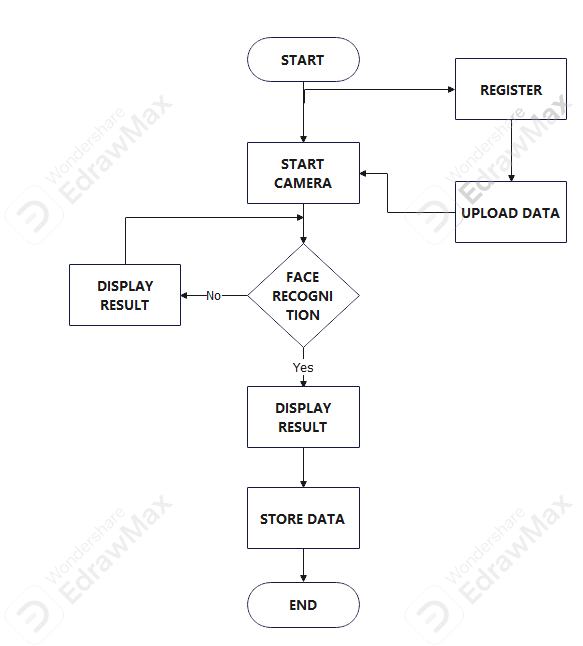
3.1 ***SYSTEM ARCHITECTURE***



**Figure 1**

**3.2 ACTIVITY DIAGRAM**

An activity diagram is a behavioral diagram that describes the dynamic aspects of a system. Here it visually represents how activities are coordinated in the system



**Figure 2**

The above activity diagram represents the process flow for a facial recognition-based attendance system. Each component and interaction within the diagram plays a crucial role in the overall functionality of the system. Below is a detailed explanation of each component and its role in the system architecture:

1. User

 - Function: The user interacts with the system to either register a new profile or start the camera for attendance recording.

    - Actions:

        - Register New Profiles: Users initiate the registration process for new individuals in the system.

        - Start Camera: Users start the camera to begin the process of attendance recording.

2. Register

    - Function: This component handles the registration of new profiles into the system.

    - Actions:

        - Register New Profiles: Captures necessary information and images of new individuals to create their profiles.

        - Start Camera\* Triggers the camera to capture facial images for the registration process.

3. Camera

    - Function: The camera is responsible for capturing images or video frames that include the faces of individuals.

    - Actions:

        - Start Camera: Activates the camera to start capturing images or video feeds.

        - Store Identification Data: Transmits captured images to the recognition system for further processing.

4. Recognition System

    - Function: This system processes the captured images to detect and recognize faces, and it records attendance based on recognized faces.

    - Actions:

        - Extract Facial Encodings: Analyzes captured images to extract unique facial features and convert them into numerical representations (facial encodings).

        - Store Facial Encodings: Stores the extracted facial encodings in the database for future reference and comparison.

        - Record Attendance: Matches the facial encodings of captured images against stored profiles to record attendance.

5. Data Base

    - Function: This component is responsible for storing and managing all data related to facial encodings and attendance records.

    - Actions:

        - Store Identification Data: Maintains a database of identification data including facial encodings and related profile information.

        - Record Attendance: Stores attendance records by updating the database with the results from the recognition system.

Detailed Process Flow

1. User Initiates Registration:

   - The user starts the process by registering new profiles. This involves capturing facial images and other identification information which is handled by the 'Register' component.

2. Register Component Starts Camera:

   - During registration, the 'Register' component activates the camera to capture the facial images of new individuals.

3. Camera Captures and Stores Data:

   - The camera captures images and forwards them to the recognition system for extracting facial encodings.

4. Recognition System Processes Images:

   - The recognition system extracts unique facial features from the images and converts them into facial encodings.

   - These facial encodings are then stored in the data component for future reference.

5. Attendance Recording:

   - When the user starts the camera for attendance, the recognition system captures live images.

   - The system extracts facial encodings from these images and compares them with the stored encodings.

   - Upon finding a match, the recognition system updates the attendance record in the data component

#### 3.3 **SEQUENCE DIAGRAM**

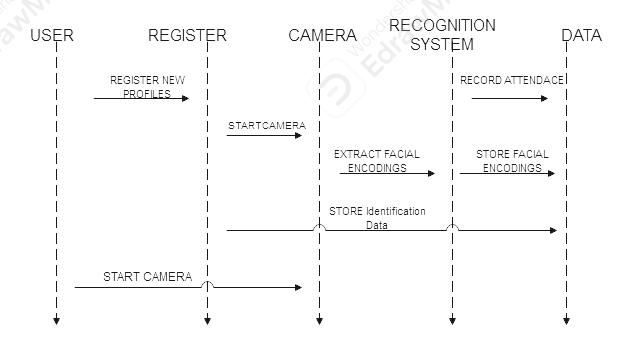
A sequence diagram is an interaction diagram used in software engineering to visualize how operations are carried out between objects within a system, this figure depicts the sequence of interactions among the user, the registration system, the camera, the recognition system, and the data storage components. It's a visual illustration of the actions needed in registering and identifying a user for attendance purposes. User Registration: A new user profile is created.

Camera Activation: The camera starts and captures the user’s facial features.

Facial Encoding: The system extracts facial encodings from the captured image.

Data Storage: The identification data is stored for future reference.

Attendance Recording: The system records the user’s attendance using the stored facial data.



**Figure 3**

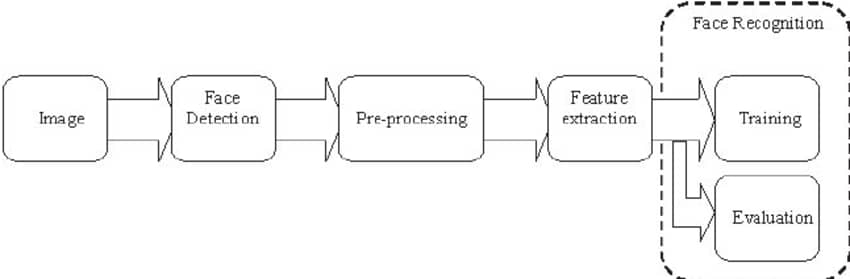
Here descries the python file created in the development of the proposed attendance system

**Facial Recognition Engine (`simple\_facerec.py`):**

This module implements the core facial recognition functionality using the `face\_recognition` library. It can encode and store facial images as reference points for known individuals. The module is capable of detecting and recognizing known faces in real-time video feeds. It utilizes algorithms to match detected faces with stored facial encodings, enabling accurate identification.

* Known\_face\_encoding: List to store encoding
* Known\_face\_names: List to store corresponding face names.
* Frame\_reszing: A scaling factor used for resizing frame
* When a frame is passed to the ‘detected\_known\_Faces’ method for face detection, it first resizes the frame to a smaller size to speed up processing.
* The frame\_resizing attribute is set to 0.25, meaning the frame will be resized to 25% of its original dimensions
* After face detection and recognition task are performed on the resized frame, the dectected face location are adjusted back to its original size for accuracy

The figure 4 below show the architecture of the **Facial Recognition Engine**



**Figure 4**

##### **Main Application Logic (`main.py`):**

This script serves as the main control hub of the surveillance system. It interfaces with CCTV cameras to capture live video feeds. Using the facial recognition engine, it processes video frames to detect and recognize faces. Upon recognition, it updates the database with relevant information such as the individual's identity, location, and timestamp.

##### **Graphical User Interface (`gui.py` and `Track.py`):**

The GUI modules provide intuitive interfaces for user interaction. `gui.py` allows users to input data such as names, roll numbers, and upload images for database entry.

##### **Database Management (`data\_handling.py`):**

The database module handles storage and retrieval of user information. It utilizes SQLite to maintain a database of individuals, storing attributes such as name, roll number, location, and timestamp. The module provides functions for inserting new records, updating existing ones, and querying information based on specified criteria.

This face recognition attendance system utilizes face\_recognition, opencv-python, and tkinter to monitor and record attendance through facial recognition. It connects to a SQLite database to store and retrieve attendance data, and provides a GUI for querying student details using their roll number. Install the necessary dependencies using the provided requirements.txt file and follow the usage instructions to set up and run the system.

**Structured Excel Generator (`data\_excel.py`):**

The data\_excel.py script interacts with a SQLite database named data.db, processes the data retrieved from the USER table, and then writes the processed data to an Excel file named student\_attendance.xlsx

**Querying Records (`Track.py`):**

It connects to a SQLite database to store and retrieve attendance data, and provides a GUI for querying student details using their Matric number.

#### **3.2 How the system works internally, including the key components and their interactions:**

**1. Frame Capture with Computer Vision:**

- Component 1: Video Input

- Capture video feeds from cameras covering different areas.

- Using the deep learning model (e.g., open cv, cv 2) for initial object detection.

- Identify and extract features, especially faces.

##### **3. Face Recognition and Database:**

- Component 1: Face Database

- Store images of known individuals along with unique identifiers.

- Component 2: Face Recognition

- Implementing the face\_recognition API libery in phyon for real-time face recognition.

- Compare facial features of detected faces with the database.

- Assign unique identifiers to recognized individuals.

##### **3. Location and Activity Monitoring:**

-Component 1: Triangulation for Location

- If multiple cameras are used, triangulate the person's position based on camera views.

- Associate the person with the room where they were first detected.

##### **4. Timestamps and Attendance Logging:**

-Component 1: Event Timestamps

- Record timestamps for entry into a room, exit from a room, and changes in activity.

# **CHA­­PTER FOUR**

## **IMPLEMENTATION AND TESTING**

This chapter discusses the system implementation approach, the hardware and software tool used for the implementation and the testing of the developed system.

System implementation approach: the system implementation approach adopted in this study follow strictly form the context diagram, depicting the system. The facial capture by the positioned camera represent the input into the system. And this step presumes that the cameras have been previously configured to interface with the developed application run on the computer that is connected to it. This proses also assumes that the student’s facial features as been previously captured and stored in the data-base. Therefore, as the student subsequently present their faces before the cameras a capture is compared to the stored previous capture. If there is a match, then the record of attendance is updated for the student in the data-base.

The hardware and software tools employed: this study employed a collection of hardware devices, with most of this hardware device were having some embedded software application them. For example the CCTV camera have embedded software. That was connected to the software application developed for attendance, with the aid of an Application Programing Interface (API). And other HARDWARE component used in this study were is the server computer.

The two major Software applications were used in this system. These are the python programing language which was used to code the application for the system and Structured programing language (SQL) management system with was used to develop the proposed system database.

Most of the tools in the python liberey was used extensively in this study. These are..:

‘CV 2’ (OPEN CV): Used in Image processing task

‘Face\_recognition’: Used for facial recognition task

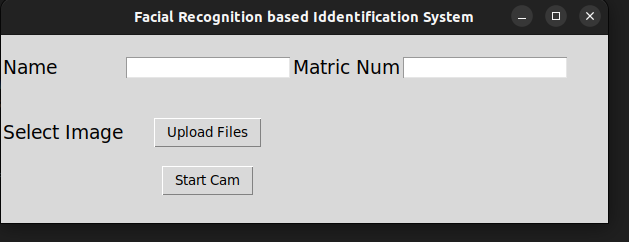
‘OS’: Used to interact with the operation system

‘glob’: Used for file path pattern matching

‘numpy’: used for numeric computing

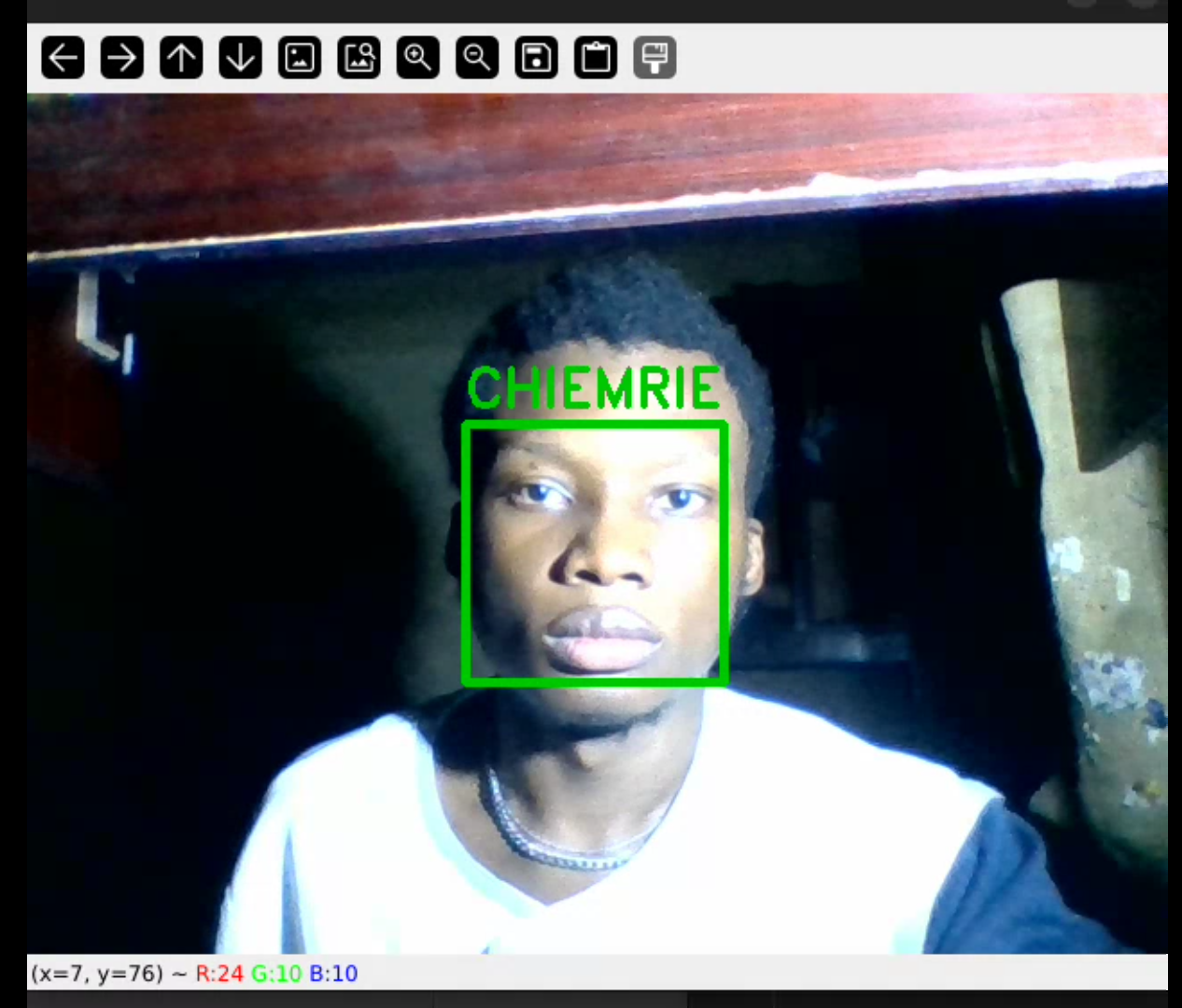
Developed System testing and documentation: having extensively fixed several box in the course of the proposed system development a functioning system was achieved. The developed system allows for the use of an external CCTV camera, for image capture, as well as the desktop computer camera for image capture. And subsequently the processing of the required attendance information was generated.

System testing: figure 5 shows the sample screenshot of the running application displaying a rectangular window representing the camera for image capture either for registration or subsequent attendance taking. In other word first time interaction in for image and data capture.



**Figure 5**

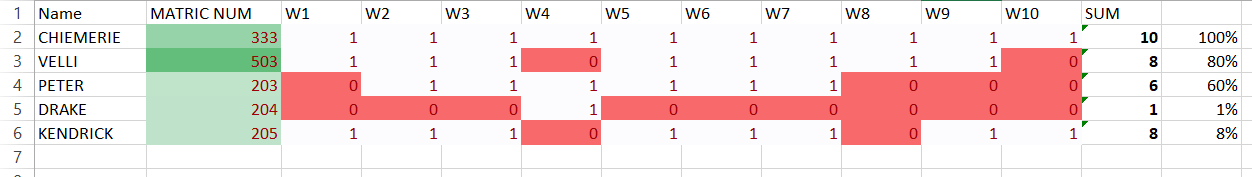
Successful registration to the system the screenshot shown on fig 6 pop up



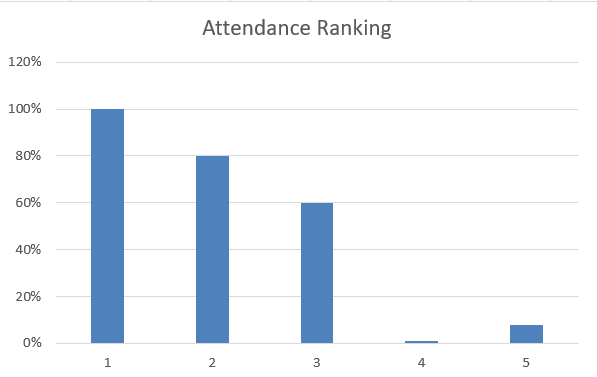
**Figure 6**

Table 1 shows a sample attendance record generated form the system.

**Table 1**



**Table 2**



This table 2 shows the attendance ranking in the class

# **CHAPTER FIVE**

## **SUMMARY, CONCLUSION AND RECOMMENDATION**

### **SUMMARY**:

The development of a facial recognition-based attendance system offers a significant advancement in attendance tracking methods, providing a convenient and efficient solution for educational institutions and workplaces. Through the utilization of advanced facial recognition algorithms, this system automates the attendance recording process, reducing administrative burdens and improving overall accuracy.

The implementation of the system involves several key steps, including data collection, feature extraction, algorithm development, database creation, system integration, and testing. By following a systematic approach, organizations can ensure the successful deployment and operation of the facial recognition-based attendance system.

### **RECOMMENDATIONS:**

Based on the findings of this research, the subsequent recommendations are suggested for further advancement and optimization of attendance systems based on facial recognition:

Allocation of resources towards Research and Development is advised for organizations, with the aim of continuous improvement of facial recognition algorithms, enhancement of accuracy, and mitigation of emerging challenges.

A focus on enhancing Data Quality is recommended, emphasizing the importance of high-quality data collection and preprocessing methods to boost the accuracy and dependability of attendance systems based on facial recognition.

Encouraging Interdisciplinary Collaboration is deemed essential, advocating for cooperation among specialists in computer vision, machine learning, and field-specific domains (e.g., education, human resources) to create customized solutions that cater to the distinct needs and demands of various industries and environments.

Organizations are urged to give priority to Ethical Considerations, particularly in relation to data privacy and ethical concerns during the implementation and operation of attendance systems based on facial recognition, ensuring adherence to pertinent regulations and standards.

Through the adoption of these recommendations, organizations can harness facial recognition technology to streamline attendance management procedures, enhance operational efficiency, and elevate overall organizational performance.

### **CONCLUSION:**

In conclusion, the utilization of a facial recognition-based attendance system signifies a valuable asset in the optimization of attendance management procedures. The establishment and execution of this system exhibit the capability to streamline administrative duties, enhance data precision, and fortify security protocols in both educational institutions and professional environments.

Despite the notable advancements made in the realm of facial recognition technology, there exists opportunity for enhancement regarding precision, dependability, and scalability. Continuous exploration and ingenuity in this domain are imperative to tackle obstacles such as variations in lighting circumstances, facial gestures, and obstructions, which have the potential to impact the efficacy of the system.

# **REFERENCE**

Bledsoe, W. W., Wolf, H. C., & Bisson, C. E. (1966). Facial recognition: The early efforts. Computers and Automation, 15(11), 11-17.

Goldstein, A. J., Harmon, L. D., & Lesk, A. B. (1971). Identification of human faces. Proceedings of the IEEE, 59(5), 748-760.

Sirovich, L., & Kirby, M. (1987). Low-dimensional procedure for the characterization of human faces. Journal of the Optical Society of America A, 4(3), 519-524.

Turk, M., & Pentland, A. (1991). Eigenfaces for recognition. Journal of Cognitive Neuroscience, 3(1), 71-86.

Jain, A., & Singh, P. (2014). "Development of Attendance Management System Using Biometrics." International Journal of Computer Science and Mobile Computing, 3(2), 182-187.

Sharma, M., & Singh, S. (2012). "A Face Recognition Technique for Attendance System Using PCA." International Journal of Computer Applications, 59(11), 20-24.

Verma, S., & Kumar, V. (2013). "Enhancing Classroom Attendance Using Facial Recognition." International Journal of Computer Science and Information Security, 11(4), 1-7.

Ahmed, S., & Hossain, M. (2018). "Development of a Face Recognition-Based Attendance System for Classroom Environments." International Journal of Computer Science and Network Security, 18(6), 128-133.

Patel, K., & Sharma, R. (2011). "Automated Attendance System Using Image Processing Techniques." Journal of Information Technology and Software Engineering, 1(3), 45-50.

Das, S., & Paul, M. (2013). "Facial Recognition for Classroom Attendance System Using PCA and LDA." International Journal of Computer Science and Engineering, 5(9), 790-795.

Singh, A., & Kaur, R. (2010). "Face Recognition-Based Attendance Management System." International Journal of Advanced Research in Computer Science and Software Engineering, 2(8), 120-124.

Gupta, A., & Kaur, K. (2016). "Real-Time Face Recognition for Attendance Monitoring." International Journal of Computer Applications, 145(8), 37-42.

Raj, R., & Kumar, P. (2017). "Face Recognition-Based Attendance System Using Raspberry Pi." Journal of Electronics and Communication Engineering, 12(4), 40-45.

Kumar, S., & Patnaik, A. (2015). "Automated Attendance System Using Face Recognition." Procedia Computer Science, 58, 807-814

Jain, A. K., Ross, A., & Nandakumar, K. (2011). Introduction to Biometrics. Springer Science & Business Media.

Phillips, P. J., Martin, A., Wilson, C. L., & Przybocki, M. (2000). An introduction to evaluating biometric systems. Computer Management, 33(2), 56-63.

Wang, Y., & Deng, W. (2021). Deep Face Recognition: A Survey. *Neurocomputing*, 429, 215-244.

Ponce, H. R., & Rovira, F. E. (2020). Issues in manual attendance-taking in educational institutions. *Journal of Education and Technology*, 12(2), 102-110.

Dhanda, P., & Arora, S. (2019). Challenges in implementing biometric systems for large-scale attendance management. *International Journal of Biometrics*, 11(1), 65-78.

Li, Z., & Zhu, H. (2019). Surveillance technology and its applications in public monitoring. *Asian Journal of Surveillance Technology*, 14(3), 223-238.

Saini, R., & Kumar, M. (2021). The challenges of manual monitoring systems in high-traffic environments. *Journal of Organizational Management*, 15(4), 201-217.

ShreyakSawhney, KaranKacker, SamyakJain, Shailendra Narayan Singh and RakeshGarg, “Real-Time Smart Attendance System using Face Recognition Techniques”, in 9th International Conference on Cloud Computing, Data Science& Engineering, 2019.

Ishaq, Kashif, and Samra Bibi. "IoT based smart attendance system using RFID: A systematic literature review." (2023).

Jha, Abhishek. “Class Room Attendance System Using Facial Recognition System.” (2013).

Andrejevic, M., & Selwyn, N. (2020). Facial recognition technology in schools: critical questions and concerns. Learning, Media and Technology, 45(2), 115–128

Y.-W. Kao, H.-Z. Gu, and S.-M. Yuan “Personal based authentication by face recognition,” in proc. Fourth International Conference on Networked Computing and Advanced Information Management, pp 81-85, 2008.

# **APPENDIX**

**main.py:**

import cv2

from simple\_facerec import SimpleFacerec

import sqlite3 as sl

import time

def cctv():

    # Connect to the database

    con = sl.connect('data.db')

    # Initialize SimpleFacerec for face encoding

    sfr = SimpleFacerec()

    sfr.load\_encoding\_images("images/")

    # Initialize the camera

    cap = cv2.VideoCapture(0)

    while True:

        # Capture frame-by-frame

        ret, frame = cap.read()

        # Detect faces

        face\_locations, ids = sfr.detect\_known\_faces(frame)

        for face\_loc, id in zip(face\_locations, ids):

            y1, x2, y2, x1 = face\_loc[0], face\_loc[1], face\_loc[2], face\_loc[3]

            name = ""

            if id == "Unknown":

                color = (0, 0, 200)

                name = "Unknown"

            else:

                color = (0, 200, 0)

                # Update user location and time in the database

                query = "UPDATE USER SET location=?, time=? WHERE id=?"

                data = ('COLLEGE 1', time.time(), id)

                with con:

                    con.execute(query, data)

                # Retrieve user's name from the database

                with con:

                    data = con.execute("SELECT name FROM USER WHERE id=?", (id,))

                    for row in data:

                        name = row[0]

            print(name)

            # Display the name and draw a rectangle around the face

            cv2.putText(frame, name, (x1, y1 - 10), cv2.FONT\_HERSHEY\_DUPLEX, 1, color, 2)

            cv2.rectangle(frame, (x1, y1), (x2, y2), color, 4)

        # Display the resulting frame

        cv2.imshow("Frame", frame)

        # Exit on pressing the ESC key

        key = cv2.waitKey(1)

        if key == 27:

            break

    # Release the camera and close all OpenCV windows

    cap.release()

    cv2.destroyAllWindows()

#cctv()

**Gui.py:**

import tkinter as tk

from tkinter import StringVar, Label, Entry, W, END

from tkinter import filedialog

from PIL import Image, ImageTk

import shutil

import sqlite3 as sl

from main import \*

# Database connection

con = sl.connect('data.db')

# Main window setup

gin = tk.Tk()

gin.geometry("560x300")

gin.title('Facial Recognition based Identification System')

# Name

name\_text = StringVar()

name\_label = Label(gin, text='Name', font=('bold', 14), pady=20)

name\_label.grid(row=0, column=0, sticky=W)

name\_entry = Entry(gin, textvariable=name\_text)

name\_entry.grid(row=0, column=1)

# Matric Num

roll\_text = StringVar()

roll\_label = Label(gin, text='Matric Num', font=('bold', 14))

roll\_label.grid(row=0, column=4, sticky=W)

roll\_entry = Entry(gin, textvariable=roll\_text)

roll\_entry.grid(row=0, column=5)

# Image

img\_label = Label(gin, text='Select Image', font=('bold', 14), pady=20)

img\_label.grid(row=1, column=0, sticky=W)

upload\_button = tk.Button(gin, text='Upload Files', command=lambda: upload\_file())

upload\_button.grid(row=1, column=1)

# Output

output\_label = Label(gin, text='Added to DB', font=('bold', 14))

# Start camera operation

cam\_button = tk.Button(gin, text='Start Cam', command=lambda: cctv())

cam\_button.grid(row=9, column=1)

def clear\_text(image\_label, button):

    name\_entry.delete(0, END)

    roll\_entry.delete(0, END)

    output\_label.configure(text="")

    image\_label.config(image='')

    button.destroy()

def upload\_file():

    file\_types = [('Jpg Files', '\*.jpg')]

    filenames = filedialog.askopenfilename(multiple=True, filetypes=file\_types)

    col = 1

    row = 3

    for file in filenames:

        img = Image.open(file)

        roll = roll\_text.get()

        sql = 'INSERT INTO USER (id, name, location) VALUES (?, ?, ?)'

        data = (roll, name\_text.get(), 'Main Gate')

        with con:

            con.execute(sql, data)

        shutil.copy(file, f'images/{roll}.jpg')

        img = img.resize((100, 100))

        img = ImageTk.PhotoImage(img)

        image\_label = Label(gin, image=img)

        image\_label.grid(row=row, column=col)

        image\_label.image = img

        output\_label.grid(row=7, column=1)

        add\_new\_button = tk.Button(gin, text='Add New', command=lambda: clear\_text(image\_label, add\_new\_button))

        add\_new\_button.grid(row=8, column=1)

        if col == 3:

            row += 1

            col = 1

        else:

            col += 1

gin.mainloop()

**simple\_facerc.py:**

import face\_recognition

import cv2

import os

import glob

import numpy as np

class SimpleFacerec:

    def \_\_init\_\_(self):

        self.known\_face\_encodings = []

        self.known\_face\_names = []

        self.frame\_resizing = 0.25  # Resize frame for faster speed

    def load\_encoding\_images(self, images\_path):

        """

        Load encoding images from path

        :param images\_path: Path to the folder containing images

        """

        images\_path = glob.glob(os.path.join(images\_path, "\*.\*"))

        print(f"{len(images\_path)} encoding images found.")

        for img\_path in images\_path:

            img = cv2.imread(img\_path)

            rgb\_img = cv2.cvtColor(img, cv2.COLOR\_BGR2RGB)

            basename = os.path.basename(img\_path)

            filename, \_ = os.path.splitext(basename)

            img\_encoding = face\_recognition.face\_encodings(rgb\_img)[0]

            self.known\_face\_encodings.append(img\_encoding)

            self.known\_face\_names.append(filename)

        print("Encoding images loaded")

    def detect\_known\_faces(self, frame):

        """

        Detect known faces in a given frame

        :param frame: Frame from video feed

        :return: face\_locations, face\_names

        """

        small\_frame = cv2.resize(frame, (0, 0), fx=self.frame\_resizing, fy=self.frame\_resizing)

        rgb\_small\_frame = cv2.cvtColor(small\_frame, cv2.COLOR\_BGR2RGB)

        face\_locations = face\_recognition.face\_locations(rgb\_small\_frame)

        face\_encodings = face\_recognition.face\_encodings(rgb\_small\_frame, face\_locations)

        face\_names = []

        for face\_encoding in face\_encodings:

            matches = face\_recognition.compare\_faces(self.known\_face\_encodings, face\_encoding)

            name = "Unknown"

            face\_distances = face\_recognition.face\_distance(self.known\_face\_encodings, face\_encoding)

            best\_match\_index = np.argmin(face\_distances)

            if matches[best\_match\_index]:

                name = self.known\_face\_names[best\_match\_index]

            face\_names.append(name)

        face\_locations = np.array(face\_locations) / self.frame\_resizing

        return face\_locations.astype(int), face\_names

**Track.py:**

import sqlite3 as sl

import tkinter  as tk

from tkinter import \*

from PIL import Image, ImageTk

# from datetime import datetime

con = sl.connect('data.db')

gin = tk.Tk()

gin.geometry("500x300")

# Roll No

roll\_text = StringVar()

roll\_label = Label(gin, text='Roll NO', font=('bold', 14))

roll\_label.grid(row=0, column=2, sticky=W)

roll\_entry = Entry(gin, textvariable=roll\_text)

roll\_entry.grid(row=0, column=3)

b1 = tk.Button(gin, text='Show Details', width=15,bg='green',

    command=lambda: my\_details() )

b1.grid(row=1,column=3)

#Result

name\_str = tk.StringVar()

Name = tk.Label(gin,  textvariable=name\_str, width=30)

Name.grid(row=3, column=2)

name\_str.set(" ")

loc\_str = tk.StringVar()

loc = tk.Label(gin,  textvariable=loc\_str, width=30)

loc.grid(row=5, column=2)

loc\_str.set(" ")

time\_str = tk.StringVar()

Time = tk.Label(gin,  textvariable=time\_str, width=30 )

Time.grid(row=6, column=2)

time\_str.set(" ")

def my\_details():

    roll=str(roll\_text.get())

    with con:

        query="SELECT name, location, time FROM USER where id=(?)"

        inp=(roll,)

        data = con.execute(query,inp)

        for row in data:

            name\_str.set('Name: '+row[0])

            img=Image.open('images/'+str(roll)+'.jpg')

            img=img.resize((100,100)) # new width & height

            img=ImageTk.PhotoImage(img)

            e1 =Label(gin,image=img)

            e1.grid(row=4,column=2)

            e1.image = img

            e1['image']=img # garbage collection

            loc\_str.set('Last Known Location: '+row[1])

            timestamp=row[2]

            # dt\_object = datetime.fromtimestamp(timestamp)

            time\_str.set('Time: '+str(timestamp))

            #print(str(name)+" "+str(location)+" "+str(time))

my\_details()

gin.mainloop()

**data\_excel.py:**

import sqlite3 as sl

import pandas as pd

from datetime import datetime

# Connect to the SQLite database

con = sl.connect('data.db')

# Query the USER table

with con:

    data = con.execute("SELECT id, name, time FROM USER")

    rows = data.fetchall()

# Process the data

attendance\_data = {}

for row in rows:

    student\_id = row[0]

    name = row[1]

    time\_str = row[2]

    # Convert the datetime string to a datetime object

    try:

        dt = datetime.strptime(time\_str, '%Y-%m-%d %H:%M:%S')

        week = dt.strftime('%Y-%U')  # Year and Week number

    except ValueError:

        print(f"Error: {time\_str} is not a valid datetime format.")

        continue

    if student\_id not in attendance\_data:

        attendance\_data[student\_id] = {'name': name, 'weeks': set()}

    attendance\_data[student\_id]['weeks'].add(week)

# Prepare the data for the DataFrame

processed\_data = []

for student\_id, info in attendance\_data.items():

    processed\_data.append({

        'Matric Number': student\_id,

        'Name': info['name'],

        'Weeks Attended': len(info['weeks'])

    })

# Create a DataFrame

df = pd.DataFrame(processed\_data)

# Write the DataFrame to an Excel file

df.to\_excel('student\_attendance.xlsx', index=False)

print("Data has been written to student\_attendance.xlsx")

**Track.py:**

import sqlite3 as sl

import tkinter  as tk

from tkinter import \*

from PIL import Image, ImageTk

# from datetime import datetime

con = sl.connect('data.db')

gin = tk.Tk()

gin.geometry("500x300")

# Roll No

roll\_text = StringVar()

roll\_label = Label(gin, text='Roll NO', font=('bold', 14))

roll\_label.grid(row=0, column=2, sticky=W)

roll\_entry = Entry(gin, textvariable=roll\_text)

roll\_entry.grid(row=0, column=3)

b1 = tk.Button(gin, text='Show Details', width=15,bg='green',

    command=lambda: my\_details() )

b1.grid(row=1,column=3)

#Result

name\_str = tk.StringVar()

Name = tk.Label(gin,  textvariable=name\_str, width=30)

Name.grid(row=3, column=2)

name\_str.set(" ")

loc\_str = tk.StringVar()

loc = tk.Label(gin,  textvariable=loc\_str, width=30)

loc.grid(row=5, column=2)

loc\_str.set(" ")

time\_str = tk.StringVar()

Time = tk.Label(gin,  textvariable=time\_str, width=30 )

Time.grid(row=6, column=2)

time\_str.set(" ")

def my\_details():

    roll=str(roll\_text.get())

    with con:

        query="SELECT name, location, time FROM USER where id=(?)"

        inp=(roll,)

        data = con.execute(query,inp)

        for row in data:

            name\_str.set('Name: '+row[0])

            img=Image.open('images/'+str(roll)+'.jpg')

            img=img.resize((100,100)) # new width & height

            img=ImageTk.PhotoImage(img)

            e1 =Label(gin,image=img)

            e1.grid(row=4,column=2)

            e1.image = img

            e1['image']=img # garbage collection

            loc\_str.set('Last Known Location: '+row[1])

            timestamp=row[2]

            # dt\_object = datetime.fromtimestamp(timestamp)

            time\_str.set('Time: '+str(timestamp))

            #print(str(name)+" "+str(location)+" "+str(time))

my\_details()

gin.mainloop()