Employee Attendance Tracking Using Facial Recognition System

Bukola Meka OWOLABI
LCU/PG/002526

Being a MSc Thesis Presentation Submitted to the Department of Computer Science, Faculty of Natural and Applied Sciences, Lead City University, Ibadan, Oyo State, Nigeria

In Partial Fulfilment of the Requirement for the Award of Master of Science (MSc) Degree in Computer Science

2023

Certification

This is to certify that Bukola Meka OWOLABI with matriculation number LCU/PG/002526 carried out this research work titled "Employee Attendance Tracking Using Facial Recognition System in the Department of Computer Science, Faculty of Natural and Applied Sciences, Lead City University, Ibadan, Oyo State, for the award of Master of Science (MSc) in Computer Science and that this has not been previously submitted.

……... ……

Dr.A.Waheed
Supervisor

Date

……... ……

Dr.W.Sakpere
Head of Department

Date

Dedication

This research work is dedicated to God, my parents, my husband and my siblings.

Acknowledgement

Foremost, I would like to express my gratitude to the leadership of the Lead City University, Ibadan for creating an a medium for us to acquire knowledge for self-reliance.

I acknowledge my supervisor Dr.A.Waheed, for the continuous support for M.Sc study and research, for his motivation, enthusiasm, to guide me through the research. The immense knowledge and plentiful experience have encouraged me in all the time of my academic research and daily life. Besides my supervisor, my sincere thanks go to the Head of Department Dr. Wilson Sakpere, the Postgraduate Coordinator Dr. Azeez Waheed, all other lecturers and staff members in the department of computer science for their guidance encouragement, and insightful comments.

I thank my parents Revd.Dr & Mrs Owolabi for their guidance, parental advice and commitment instilled in me towards learning. May you both live long in good health to enjoy and reap fruits of your labour. My sincere thanks to my loving husband Mr Dolapo Adeyemi,ACA for the understanding, sacrifice, and prayers during my study. Also, to my siblings Dr Femi Owolabi, Mrs Adebayo, Mr Gbenga Owolabi for their continuous support and advice.

Abstract

Traditional pen-and-notebook methods for employee attendance are often susceptible to inaccuracies and falsifications. Biometric systems, despite being more secure, confront issues such as high acquisition costs and inefficiencies in capturing fingerprints, especially when hands are unclean or injured. In this study, a cutting-edge Employee Attendance Tracking System using Facial Recognition is developed, addressing the shortcomings of conventional attendance methods and biometric systems. The proposed system employs an array of Python libraries including Django, face_recognition, OpenCV (cv2), numpy, and PCA. These libraries are utilized for their strengths in image processing, facial recognition, and efficient data management. The primary objective is to create a reliable, cost-effective, and efficient alternative for recording employee attendance, overcoming the limitations of existing methods.The system utilizes advanced image processing techniques to tackle common challenges in facial recognition, such as noise interference, varying lighting conditions, and physical obstructions like occlusions. This is achieved through innovative approaches like noise reduction, illumination normalization, and occlusion handling, significantly improving the accuracy of facial recognition under diverse environmental conditions. A key component of the system is the "Capture_Image" module, which establishes a reference database by capturing and storing employee images. Concurrently, the "Recognize" module employs machine learning algorithms for facial recognition, ensuring accurate and timely recording of attendance. The effectiveness of the system is demonstrated in its ability to adapt to a variety of environments, attributed to its advanced image processing capabilities and robust algorithmic framework. This innovative system is particularly advantageous for institutions, corporate offices, and industries seeking secure, precise, and efficient attendance tracking solutions. It marks a significant advancement in the field of attendance management, offering a blend of enhanced security, accuracy, and operational efficiency. The study recommends further enhancements, such as incorporating advanced algorithms to improve recognition accuracy in different lighting and noise conditions.

Keywords: Accuracy, Biometric system, Employee Attendance Tracking, Facial recognition, Machine learning algorithm

Word Count: 295 Words

Table of Contents

Title page

Certification ii

Dedication iii

Acknowledgement iv

Abstract v

Table of Content vi

List of Tables ix

List of Figures x

List of Acronyms xi

Chapter One: Introduction

1.1. Background to the Study 1

1.2 Statement of the Problem 3

1.3 Aim and Objectives of the Study 4

1.6 Significance of the Study 4

1.6 Scope of the Study 4

1.7 Operational Definition of Terms 5

Endnotes

Chapter Two: Literature Review

2.1. Conceptual Review 7

2.1.1 History of Facial Recognition 9

2.1.2 Facial Recognition 10

2.1.3 Advantages of Facial Recognition 11

2.1.4 Draw Back of Facial Recognition 12

2.1.5 Face Recognition Methods 13

2.1.5.1 Principal Component Analysis (PCA) 13

2.1.5.2 Discriminant Analysis (LDA) 14

2.1.5.3 Independent Component Analysis (ICA) 14

2.1.5.4 Kernel Principal Component Analysis (KPCA) 15

2.1.5.5 Kernel Linear Discriminant Analysis (KLDA) 16

2.1.6 Distance Measurement 16

2.1.6.1 Euclidean Distance 16

2.1.6.2 City Block Distance 17

2.1.6.3 Mahalanobis Distance 17

2.1.6.4 Chebyshev Distance 18

2.2 Methodological Review 18

2.2.1 Principal Component Analysis (PCA) 18

2.2.2 OpenCV (Open Source Computer Vision Library) 20

2.2.3 Other Face Recognition Algorithms 21

2.3 Review of Related Works 21

Endnotes

Chapter Three: Methodology

3.1. Software Analysis and Design 46

3.1.1 Phases of Waterfall Model 46

3.2. Software Requirements and Approach 47

3.3 Requirement Specification 48

3.3.1 Functional Requirement 48

3.4 Face Recognition Design 50

3.5 Face Recognition Design 52

3.5.1 Classification or Recognition. 53

3.6 System Architecture 53

3.6.1 System Flowchart 55

3.6.2 Use case diagram 56

3.7 Choice of Programming Language 59

Endnotes

Chapter Four:Results and Discussion of Findings

4.1 System Overview 61

4.1.1 Face Image Capture 61

4.1.2 Training the Images: A Comprehensive Exploration 63

4.1.3 Feature Extraction: Unveiling Hidden Patterns 63

4.1.4 Model Training: Imbuing Intelligence 64

4.1.5 Recognizing and Recording Attendance 65

4.2 Results 65

4.2.1 Admin Login 65

4.2.2 User Interface Page 68

4.2.3 Capture Interface 68

4.2.4 Main Dashboard 69

4.2.5 Employee Profile Interface 71

4.3 Performance Evaluation 72

4.3.1 System Performance Insights 73

4.3.2 Camera Testing 74

4.3.3 Image Quality Challenges 74

4.3.4 Potential Mitigation Strategies 75

Endnote

Chapter Five : Conclusion

5.1 Summary 77

5.2 Conclusion 77

5.3 Recommendations 78

5.4 Contribution to Knowledge 79

5.5 Suggestions for Further Research 80

5.6 Limitation of Study 81

Bibliography 82

Appendix 90

Bio Data 144

University Compliance Form 145

List of Tables

Table	Title	Page
4.1	Performance Evaluation Table	72

List of Figures

Figure	Title	Page
2.1.	The Operate Process of Face Recognition System	8
3.1.	Conceptual Model of the Proposed Design	52
3.2	System Flow Chart	54
3.3.	Administrator Use Case	55
3.4	Staff Use Case	56
4.1	Admin Login	65
4.2	User Interface Page	66
4.3	Capture Interface	68
4.4	Main Dashboard	69
4.5	Employee Profile Interface	71

List of Acronyms

CNNs-	Convolutional Neural Networks
FLD-	Fisher's Linear discriminant
FMLA-	Family and Medical Leave Act
HOG-	Histogram of Oriented Gradient
ICA-	Independent Component Analysis
KLDA-	Kernel Linear Discriminant Analysis
KPCA-	Kernel Principal Component Analysis
LBPH-	Local Binary Patterns Histograms
LPB-	Local Binary Pattern
MFCC-	Mel Frequency Cepstral Coefficient
ML-	Machine Learning
OPENCV -	Open Source Computer Vision Library
PCA-	Principal Component Analysis
SVM-	Support Vector Machine
VQ-	Vector Quantization

Chapter One

Introduction

1.1 Background to the Study

Face recognition is as old as computer vision, both because of the practical importance of the topic and theoretical interest from cognitive scientists. Despite the fact that other methods of identification (such as fingerprints, or iris scans) can be more accurate, face recognition has always remains a major focus of research because of its non-invasive nature and because it is people's primary method of person identification. Face recognition technology is gradually evolving to a universal biometric solution since it requires virtually zero effort from the user end while compared with other biometric options. Biometric face recognition is basically used in three main domains: time attendance systems and employee management; visitor management systems; and last but not the least authorization systems and access control systems[1].

The pipeline of general enterprises needs to record the attendance of personnel, which has become a basic requirement of the company. However, when these attendance systems are formulated, unnecessary errors often occur. Taking the current fingerprint attendance system as an example, study has found that the fingerprint attendance system has an error rate of about 5%, and there will be a phenomenon that fingerprints cannot be hit, which seriously affects the efficiency of attendance, especially in large attendance sites, which is more likely to cause congestion[2]. However, the card attendance system has the phenomenon of employees swiping cards for someone else, and it is difficult to achieve the purpose of real time attendance[3]. Compared with the two attendance systems, the face recognition system has higher accuracy and stability, because there are more points for face recognition, which is more accurate than other systems[4,5].

Personnel management is very significant in the administration and management of organizations. Lack of security and extensive paperwork in the traditional systems reveal that there is a need for new approach for personal verification and employee identification technologies. In recent times, organizations have started to benefit from fingerprints and facial recognition more than before for personnel attendance. Moreover, the fact that no two individuals have the-same finger prints and face shape increases the system security. It can be also asserted that, since fingerprints and facial structure remains unchanged throughout lifetime, it remains unique and sustainable[6].

Organizations of all sizes use time and attendance systems to record when employees start and stop work, and the department where the work is performed. However, it's also common to track meals and breaks, the type of work performed, and the number of items produced. In addition to tracking when employees work, organizations also need to keep tabs on when employees are not working. Vacation time, compensation time, Family and Medical Leave Act (FMLA) time, and jury duty must be recorded. Some organizations also keep detailed records of attendance issues such as who calls in sick and who comes in late.

Facial recognition or face recognition as it is often referred to as, analyses characteristics of a person's face image input through a camera. It measures overall facial structure, distances between eyes, nose, mouth, and jaw edges[7]. These measurements are retained in a database and used as a comparison when a user stands before the camera. One of the strongest positive aspects of facial recognition is that it is non-intrusive. Verification or identification can be accomplished from two feet away or more, without requiring the user to wait for long periods of time or do anything more than look at the camera[9].

Facial recognition is an easy and secure way of taking down attendance. Facial recognition in the workplace is going to become more and more common in a post-pandemic world. Business owners across the globe are looking for contactless ways for employees to clock in

and out and calculate payroll. As a result, solutions will become more utilized than they previously were especially using image capture device that captures the images of employees, train & processes the information into a database. Then the trained face images coordinate structure are mapped and stored into database which will make the faces of employees suitable for recognition by the deployed machine. This study adopted the use of computer vision method by applying face recognition techniques using CLAHE for illumination control, median filtering for noise removal, Principal Component Analysis for feature extraction and Euclidean distance for template matching so as to collectively enhance the rate the recognition accuracy rate of Facial Recognition System in Employee Attendance Tracking and Management System.

1.2 Statement of the Problem

Most organizations make use of a pen and notebook to check attendance of its employees. With this system, employees write their names, time of arrival and signature in ruled columns in the notebook. This data is used to process the salary of employees given account of times when they were late and absent. The issue with this system is that employees tend to falsify entries and indirectly. Also attendance taking using biometrics is relatively slow because only one user could be recognized at a time, other identified issues is listed thus[10,11]:

1. Cost of acquiring the fingerprint devices is relatively high and difficulty in capturing fingerprints of users when thumbs are cut or dirty.
2. Misplacement or unrecorded temporary clock out of employees.
3. No central system for compiling attendance for individual work unit.

1.3 Aim and Objectives

The aim of this study is to develop an Employee Attendance Tracking Using Facial Recognition System, The Specific objectives are to:

i. design a central web based attendance system that allows employees to take attendance at their designated attendance point.

ii. implement a facial recognition pattern using image processing algorithms

iii. develop the platform for employee attendance, query issuance, and image real time acquisition.

iv. test and evaluate the proposed system.

1.4 Significance of Study

The implemented software is advantageous in diverse ways for businesses because it provides advanced tracking for job-costing and labour distribution. It also gives you a better picture of the company's important performance indicators, such as hours and wages, personnel numbers, and scheduled/working status. Employee productivity is also improved by the systems, which relieve payroll staffs of the time-consuming chore of entering each employee's daily hours into payroll programs. It also provides data diddling by employees thereby eliminating attendance manipulation of ghost workers and unavailable workers.

1.5 Scope of Study

This study is proposed for usage at the human resource unit of the Computer Department of Leads University Ibadan and as a desktop based application of any organization supporting IT facility and human resource management. The face recognition algorithm will make use of the Median Filtering, Contrast Limited Adaptive Histogram Equalization, Principal Component Analysis and Euclidean Distance to build all its functional components and the MYSQL database will used as storage for staff's data.

1.6 Operational Definition of Terms

Face Recognition: Facial recognition is a method of recognizing or verifying a person's identification by looking at their face. People can be identified in pictures, films, or in real time using facial recognition technology.

Attendance Management System: An attendance management system is software that keeps track of employees' working hours. It keeps precise time records for your employee's attendance, breaks, time off, and clock in and out.

Face Detection: Face detection is a computer technique that recognizes human faces in digital images and is used in a range of applications. The psychological process through which humans locate and attend to faces in a visual context is known as face detection.

Feature Extraction: The process of translating raw data into numerical features that may be handled while keeping the information in the original data set is known as feature extraction. It produces better outcomes than applying machine learning to raw data directly.

Template Matching: Template matching is a technique for looking for and locating a template image within a larger image.

Image Pre-processing: The use of computerized algorithms for the analysis of images with respect to an application is referred to as image processing.

Endnotes

1. S Dargan, M Kumar. *A comprehensive survey on the biometric recognition systems based on physiological and behavioral modalities*. Expert Systems with Applications 2020;143:113114. https://doi.org/10.1016/j.eswa.2019.113114.

2. H Xan, X Han. *Face recognition attendance system based on real-time video processing*. IEEE Access 2020;8:159143–50. https://doi.org/10.1109/access.2020.3007205.

3. E Jain, S Mishra. *Face recognition attendance system based on real-time video*. **International Journal of Computer Applications** 2022;184:12–8. https://doi.org/10.5120/ijca2022921983.

4. S Dargan, M Kumar. *A comprehensive survey on the biometric recognition systems based on physiological and behavioral modalities*. Expert Systems with Applications 2020;143:113114. https://doi.org/10.1016/j.eswa.2019.113114.

5. Q Miao, F Xiao, H Huang, L Sun, R Wang. *Smart attendance system based on frequency distribution algorithm with passive RFID tags.* Tsinghua Science and Technology 2020;25:217–26. https://doi.org/10.26599/tst.2018.9010141.

6. SD Hodge Jr. T*he claimed ability to date fingerprints-is this another example of junk science or a scientifically accurate technique*?. **J. Health & Biomedical L.**. 2022;18:171.

7. P Payal, MM Goyani. *A comprehensive study on face recognition: methods and challenges*. **The Imaging Science Journal** 2020;68:114–27. https://doi.org/10.1080/13682199.2020.1738741.

8. G Tonguç, B Ozaydın Ozkara. *Automatic recognition of student emotions from facial expressions during a lecture*. Computers & Education 2020;148:103797. https://doi.org/10.1016/j.compedu.2019.103797.

9. AM Umar, OU Muhammad, PW Asua. *A generic face detection algorithm in electronic attendance system for educational institute.* **World Journal of Advanced Research and Reviews** 2022;15:541–51. https://doi.org/10.30574/wjarr.2022.15.2.0864.

10. T Aparna, MT Chandan, P Yusuf, KS Ashish, K Neha. *Face recognition based automated attendance management system*. **International Journal of Scientific Research in Science and Technology** 2022:261–8. https://doi.org/10.32628/ijsrst229147.

11. W Deng, J Hu, J Guo. F*ace recognition via collaborative representation: Its discriminant nature and superposed representation*. IEEE Transactions on Pattern Analysis and Machine Intelligence 2018;40:2513–21. https://doi.org/10.1109/tpami.2017.2757923.

Chapter Two

Literature Review

2.1 Conceptual Framework

Face Recognition has sparked a lot of interest in recent years, and it's become a hot topic of research. Furthermore, it is a vital application in image analysis, and developing an automated system based on face recognition that can accurately distinguish human faces is a difficult task. Because resolving the manual attendance problem is time-consuming, several studies have been undertaken using an automated or smart attendance management solution to handle manual attendance concerns. However, the biometric attendance management system finds it difficult to verify each student in the classroom, especially when there are a large number of students present, and it interrupts the student when the system responds to the student if detection or recognition is not possible.

Furthermore, the biometric system necessitates additional gear or connection with the student, who must spend money on peripherals and may still be time-consuming. Building a real-time facial recognition research system with an attendance management system has been a tremendous challenge. Automatic attendance marking can address major concerns such as manual attendance errors that occur when the academic office enters data from a sheet into the system. Because one can't guarantee the data in the university office is valid 100 percent of the time, this is a problem when we utilize the traditional technique of attendance marking in the university, which has a large number of students enrolled each year, numerous departments, and many classes. Face recognition entails a number of phases, including capture, extraction, comparison, and matching[1]. Each process has the following operations:

Step one: capture is the process of taking an image during the system's enrolment. The extraction stage is used to locate or extract a specific object from the face. Then comes the comparison process, where new data is compared to the enrollment or database (sample data).

The final phase is matching: using an extraction and comparison process, the system will be able to determine if the new face matches the registration face or not.

Figure 2.1 below shows the result of the preceding.

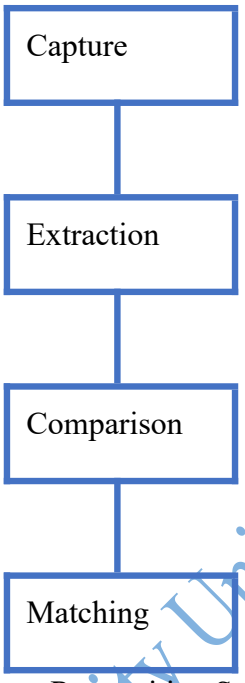

Figure 2.1. The Operate Process of Face Recognition System
Source: Research Work, 2023

Facial recognition, according to the given procedure, is a difficult task. Face recognition necessitates the use of different approaches and algorithms.

2.1.1 History of Facial Recognition

The advent of facial recognition is dated back to 1964 and 1965, where computer was used to recognize human faces[2]. Face were marked severally on the face manunally, such as the centers of the eyes, the nose, and the lips. They then employed the computer to correct for pose fluctuation by mathematically rotating[2]. To match the identity, the distances between the facial landmarks were estimated automatically and compared to the image. This was the beginning of facial recognition technology. Face recognition became a viable biometric for business after applying linear algebra to it[3]. They devised the "Eigenface" technology, which required fewer than a hundred values to accurately code the facial image.

The discovery of face detection within an image in 1991 paved the way for automated facial recognition. This paved the path for facial recognition technology to expand and flourish. DARPA and NIST launched the FERET initiative for commercial facial recognition in the early 1990s[4]. They built a database of facial images that contained 2413 photographs representing 856 persons. FRVTs were created in the early 2000s to enable unbiased government reviews of face recognition systems and prototype technologies[4]. These studies gave law enforcement agencies and the government the knowledge they needed to best employ facial recognition technology. In 2006, the Face Recognition Grand Challenge was started to test the current face recognition algorithms[4]. The new method was found to be ten times more accurate than the algorithms used in 2002 and more than 100 times more accurate than the algorithm used in 1995[4].

Face recognition technology has been introduced by Facebook in recent years to identify people who appear in a user's daily updates. Apple released the iPhone-X in 2017, which was the first iPhone to use face recognition to unlock the device[5,6].

2.1.2 Facial Recognition

An overview of face recognition in attendance marking systems was already given in the introduction section. The many approaches to facial recognition will be discussed in part, including:

i. Geometric Methodology: Based on a geometrical point on the face, this approach is recognized by humans. Many aspects of the geometric method have been investigated: used a geometric technique to increase the system's accuracy when the photos had a complicated background[7].

ii. PCA-Based: PCA is a statistical method that has been used in face recognition to extract features and reduce redundant data. PCA was also utilized to reduce dimensionality and

conserve information[8]. Given images into the system; face recognition will process to find the matching image between the given images with the images inside the databases. Principal component or Eigenface is a face that contains feature and characteristic. The image with the can be written to 1dimensional to reduce the dimensionality[8].

iii. LDA-Based: Is used to maximize the scatter image in the same class or in a different class which is enhanced from PCA. In PCA is selected subspace that contains most of the variation[9]. In 1997 " Fisherfaces" has been used in the application of (FLD) Fisher's linear discriminant. The Fisher's linear discriminant (FLD) choose the linear subspace that makes the ration more maximize.

iv. LBP (Local Binary Pattern): Is a classification method used in computer vision that was initially introduced in 1990. Based on research publications it was demonstrated in 1994 that LBP works with HOG (Histogram of Oriented Gradient), which has a high power feature for texture classification, and improves performance detection on various relevant data sets[10].

v. Active Appearance Model: This method has been employed in a wide range of applications, including face matching and medical interpretation[11].

2.1.3 Advantages of Facial Recognition

i. **Easy to Manage:** Dealing with records and monitoring daily activities will be much easier with the AI-based attendance system than with the manual approach, because it is totally automated[11]. The system will take care of everything. Several products can be adjusted to indicate the exact time of how many hours or minutes an individual worked at his or her workstation throughout the day. All activities may be readily tracked to keep track of them.

ii. **Time and Money are Saved:** This framework has the potential to save businesses a lot of time and money. Organizations will not need to hire additional labor because the face

recognition framework monitors employees' or students' working hours and access to various zones within the premises[12]. The automatic system also helps to avoid human error and keeps track of correct hours. Students can be easily monitored and identified. Face recognition has been implemented at schools, universities, and colleges to track attendance and deter criminal activities on campus.

iii. **Increases the Effectiveness of Security Mechanisms:** This framework also aids in the enhancement of security and safety protocols. Since a long time, facial recognition has been a standard feature of airport security evaluations, assisting in the identification of lawbreakers and potential threats to carriers and passengers. Facial recognition is also used by banks and other organizations to avoid fraud, as the technology may identify people who have recently been accused of wrongdoings and inform the bank or institution.

iv. **Automatic and Seamless Verification Process:** It is not necessary to wait for your turn, as with fingerprint scanners or other security measures; instead, the facial recognition for attendance framework provides a quick, programmed, and consistent verification experience.

v. **During pandemics, it reduces interaction and device contact.**COVID-19 is affecting the entire planet, and it's past time for us to pay attention to social distancing. Keeping a safe distance from people has become a necessity in recent years. If you have a manual attendance system, conditions like this can be dangerous. A face recognition-based attendance system, on the other hand, will allow you to not only register an individual's attendance but also keep you at a safe distance from them because you can work remotely while still seeing who is coming and going. This requires the point that, this entire framework is a lot more secure, efficient, and faster method to record attendance.

2.1.4 Drawbacks of Facial Recognition

Image Quality: In the identification procedure, the resolution of the reference image is critical. If a picture's quality isn't high or good enough, cameras may be misled into thinking the person being scanned isn't the same as in the photo[13]. A easy solution to this problem is to use the same cameras for both the reference pictures and the scanning.

Produces Data Security Flaws: The storing of facial recognition data is a source of concern, as these data sets could be breached or exploited.

Performance may differ from one system to the next: For a smooth and lag-free program execution, this framework requires a processor that is fast and of decent quality Configuration is simple. PCs and devices may experience latency or experience delayed data processing.

Technology could be deceived. (On rare occasions.): Camera angles, lighting conditions, and picture or video quality may all have an impact on this framework's capacity to recognize people's faces. Facial recognition technology can also be thrown off by people wearing masks or changing their appearance somewhat. However, the incidence of this problem is quite unlikely.

2.1.5 Face Recognition Methods

2.1.5.1 Principal Component Analysis (PCA)

PCA is a method for extracting features and reducing dimensions that can be used to tackle recognition and compression issues. PCA, also known as Eigen space projection, Karhunen and Loeve (KL) transformation, or Hotelling, is a prominent linear projection method. By extracting the main components (PCs) of multidimensional data, PCA reduces dimensionality [43]. PCA can extract key features, collect sample data components that are nearly variable, and then choose multiple significant individuals from all of the feature components[22]. When compared to other approaches, PCA provides an efficient and simple recognition process. Individual recognitions that incorporate PCA features also perform better in single variation cases [42]. Without any mid-level or low-level processing, raw intensity data is used for

recognition and learning. PCA is unsupervised and requires no prior knowledge of facial images. The subspace of the low dimensional representation can also be used to compute data compression [8].

Face identification was accomplished using a sparse representation-based classification algorithm, and PCA was utilized to extract features from face photos. When there are multiple observed variables and you want to develop a smaller number of unknown variables that account for practically all of the variability in recognition variables, PCA is a good choice.

2.1.5.2 Discriminant Analysis (LDA)

LDA is a feature extraction and dimension reduction algorithm that is extensively used to find linear groups of features while keeping the classes separate. It is also known as Fisherface. LDA is a supervised learning method in which each class is taught with several training images [7]. The Fisherface projection method's original work consisted on optimizing the ratio of between-class scatter to within-class scatter to address the illumination problem. Pose, lighting, and expression changes have less impact on Fisherface. LDA improves the representation of things in low dimensions [9].

The classic LDA's singularity problem is a drawback. When the total number of training photos is fewer than the feature space dimension, face recognition is challenged by the short sample size (SSS) problem via LDA. Between-class and within-class photos are segregated and labeled. The image variants of the same individual are captured between classes. Within-class captures the differences in image between individuals in the same class. LDA, unlike PCA, tries to model distinctions between classes. LDA generates unique projection vectors for each class, and multi-class LDA, which is commonly used in biometric systems, can handle more than two classes[39].

$$SB = \sum_{n=1}^{\breve{} } \sum_{m=1}^{''} (S_m^n - u_n)\,(S_m^n - u_n)^T$$

$$SW = \sum_{n=1}^{C} (u_n - u)\,(u_n - u)^T$$

2.1.5.3 Independent Component Analysis (ICA)

ICA tries to provide an independent image representation rather than an uncorrelated one by transforming data as linear combinations of statistically independent data points[41]. The picture elements are treated as random variables in PCA with Gaussian distribution, and the second order statistics are minimized. Dividing a significant variation does not match to the PCA fundamentals vector if PCA is non-Gaussian. ICA seeks to determine the essentials using statistically independent data while minimizing both higher-order and second-order dependencies in the input data. ICA is a statistical approach for converting observed multidimensional random vectors into statistically independent component vectors [10]. PCA, on the other hand, can only depict second-order inter-pixel correlations or associations that capture the amplitude spectrum of an image, not the phase spectrum of an image, because it relies on the pair wise relationship between pixels in the image database. ICA aims to take advantage of higher-order interactions between pixels. The most common application of ICA is to overcome the problem of blind source separation [27]. Image processing, reflection canceling, biological signal processing, time series forecasting, data mining, radio communications, text document analysis, and discovering hidden elements in financial data are some of the uses [27].

2.1.5.4 Kernel Principal Component Analysis (KPCA)

Non-linear PCA is also generated from several ways; diverse kernel methods are generalized to form Kernel PCA, which is one of the most used variants of non-linear PCA (KPCA). KPCA calculates PCA primarily by employing different kernel approaches to map the original input into a high-dimensional feature space. In order to determine the performance

variation, compared GABOR-PCA and GABOR KPCA versions. To test the system's performance, the ORL database is employed[28]. The GABOR-PCA approach outscored GABOR-KPCA by 6.67 percent (Euclidean), 0.83 percent (Cosine), 12.00 percent (City Block), and 4.17 percent (City Block) respectively (MAHCOS). This increase runs counter to the popular belief that KPCA is the best option by default.

2.1.5.5 Kernel Linear Discriminant Analysis (KLDA)

Kernel is a statistical method that takes into account higher-order statistics. First, The PCs in the feature space are determined when the input space is transferred onto the feature space via nonlinear mapping. Even if they don't know what the vectors are, they can compute the dot products of two feature vectors. The kernel technique allows non-linear versions of any method that can be stated purely in terms of dot products to be constructed. The mapping is made implicit and efficient by utilizing kernel functions that meet Mercer's theorem, which is a result of the rise in dimensionality.

2.1.6 Distance Measurement

The majority of facial recognition technologies developed in the last decade assist in making decisions based on distance measurements. In image recognition and computer vision, the distance between two images is a crucial challenge. The distance between two photos is measured as the final stage in face recognition. The distance between the vectors of two images is referred to as image similarity. Recognition judgments are based on the distances between feature space representations[8]. Because many existing image distance algorithms have sophisticated measure computations, merging the metric with various face recognition methods is problematic [52].

2.1.6.1 Euclidean Distance

Euclidean Distance is a popular classifier for determining similarity levels since it is straightforward and faster than other classifiers[65]. The Euclidean Distance is a straight line

distance between two points that looks at the root of square discrepancies between the coordinates of two subjects [1]. The most advantageous criterion for normally distributed classes is a minimum Euclidean Distance classifier [23]. Assume that X is a test image and Y represents a train image. Then, as stated in the equation below, Euclidean Distance can be calculated:

$$ED(X,Y) = \sqrt{\sum_{i=1}^{No.Of\ Images} (X_i - Y_i)^2}$$

2.1.6.2 City Block Distance

The total of the absolute differences between two vectors is the City Block Distance, also known as L1 Distance or Manhattan Distance classifier [1]. It's particularly handy for discrete descriptions. Because it responds to triangle inequality, City Block Distance is a valid distance function. It also assumes that the distribution is triangular [23]. The following equation can be used to calculate city block distance:

$$CB(X,Y) = |X - Y| = \sum_{i=0}^{No.Of\ Images} |X_i - Y_i|$$

2.1.6.3 Mahalanobis Distance

Mahalanobis distance It is based on the correlations between variables, allowing for the identification and analysis of various patterns. This distance determines how similar an unknown sample set is to a known sample set. Mahalanobis Distance, in contrast to Euclidean Distance, takes into account the database's correlations and is scale-invariant [14]. It's a useful statistic for determining the similarity of a set of values from an unknown sample to a set of values measured from a collection of known samples in multivariate analysis and distance measuring. The Mahalanobis Distance can be calculated using the following equation:

$$MD(X,Y) = \sqrt{(X-Y)^T\ S^{-1}\ (X-Y)}$$

[66] uses 27 RGB photos to compare Mahalanobis distance to city-block distance. Out of 27 photos, Mahalanobis distance recognizes 12 of them. Furthermore, it outperformed the city-block distance, which only identifies 11 of the 27 photos.

2.1.6.4 Chebyshev Distance

The Chebychev distance is a metric that is induced by the uniform norm or supermum norm. It's also known as the maximum value distance, and it's an injective measure. This distance is specified in a vector space, where the biggest of the discrepancies between two vectors along any coordinate dimension equals the distance between them [14].

2.2 Methodological Review

This section will discuss the algorithms used in this study

2.2.1 Principal Component Analysis (PCA)

Principal Component Analysis (PCA) is a statistical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components[65]. The number of principal components is less than or equal to the number of original variables. This transformation is defined in such a way that the first principal component has the largest possible variance (that is, accounts for as much of the variability in the data as possible), and each succeeding component, in turn, has the highest variance possible under the constraint that it is orthogonal to the preceding components.

Standardization: The first step is to standardize the range of the continuous initial variables so that each one of them contributes equally to the analysis. Standardization typically involves subtracting the mean and dividing by the standard deviation for each value of each variable.

Covariance Matrix Computation: The next step is to take the standardized data and calculate the covariance matrix. The covariance matrix expresses the correlation between the different variables in the dataset.

Eigenvalue and Eigenvector Computation: After the covariance matrix is formed, the next step is to decompose it into eigenvectors and eigenvalues.

Eigenvectors (principal components) determine the directions of the new feature space, and eigenvalues determine their magnitude. In other words, the eigenvalues explain the variance of the data along the new feature axes. Eigenvectors are sorted by corresponding eigenvalue, highest to lowest. This gives the components in order of significance. This step is crucial because it tells us the principal components that maximize the variance.

Projection Onto the New Feature Space: This step involves projecting each data point onto the principal components to get new data points.

The formula for this projection is: $Y = X.P$, where X is the original data matrix, P is the matrix of eigenvectors, and Y is the matrix of the data represented in terms of principal components[66].

In the context of face recognition, PCA is used to reduce the dimensionality of the data by transforming the original high-dimensional data (e.g., pixel values of face images) into a lower-dimensional space (e.g., "eigenfaces") while retaining most of the information or variance. The lower-dimensional space represents the most significant features necessary to distinguish faces from each other.

The steps for PCA in face recognition might involve:

i. Creating a set of face images (the training set).

ii. Flattening each image into a single vector of pixel values.

iii. Performing PCA on these vectors to find the eigenvectors.

iv. Using a subset of the eigenvectors (eigenfaces) to project the high-dimensional data into a lower-dimensional space.

v. Using this lower-dimensional space as features to train a classifier or for direct matching in a recognition task.

Principal Component Analysis (PCA) is used to reduce dimensionality while retaining the most variance in the data. A set of eigenfaces is generated by finding the eigenvectors (`e`) of the covariance matrix (`S`) of the training image set[66]. The covariance matrix is calculated as

$$S = \frac{1}{N}\sum_{i=1}^{N} (x_1 - \mu)(x_1 - \mu)^T \quad 4.1$$

Where x_1 are the face images and μ is the mean image

A new face image (`x`) is transformed into its eigenface components (`W`): $W = e^T (x_1 - \mu)$. Recognition is performed by comparing the eigenface components of the new image with those of the known images.

2.2.2 OpenCV (Open Source Computer Vision Library)

OpenCV is an open-source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in commercial products. It has C++, Python, Java, and MATLAB/OCTAVE (the last two in a limited way) interfaces and supports Windows, Linux, Android, and Mac OS. OpenCV leans mostly towards real-time vision applications and takes advantage of MMX and SSE instructions when available[58]. In the context of face recognition, OpenCV provides several methods and algorithms to perform this task. Some of the most common include:

1. Haar Cascades: For face detection, OpenCV has the `CascadeClassifier` class that is trained on a set of positive (faces) and negative (non-faces) images using Haar features. It uses an XML file containing the data to detect objects, which for face detection, typically includes `haarcascade_frontalface_default.xml[58].

2. Local Binary Patterns Histograms (LBPH): OpenCV implements LBPH for face recognition through the `face.LBPHFaceRecognizer_create()` function. It considers the pixel values as local binary patterns and uses them to generate histograms, which are then used for face recognition[58].

3. Eigenfaces and Fisherfaces: OpenCV also provides methods to use PCA (Eigenfaces) and LDA (Fisherfaces) for face recognition through the `face.EigenFaceRecognizer_create()` and `face.FisherFaceRecognizer_create()` functions. These algorithms transform faces into a lower-dimensional space where they can be more easily compared. Haar cascade is used to detect faces in real-time from a video camera feed. Once faces are detected, they can be further processed for recognition using one of the face recognition methods mentioned earlier. For a full face recognition pipeline, after detecting the face, the image will be preprocessed , extract features using one of the face recognition methods, and then compare those features to a known database of faces to find a match. OpenCV's face recognition methods come with functions to train on such a database and to predict the identity of a face given its features.

2.2.3 Other Face Recognition Algorithms

1. Fisherfaces

Linear Discriminant Analysis (LDA) is used to maximize the ratio of between-class scatter to within-class scatter. The goal is to find the vectors (`v`) that maximize $v^T S_B v$ and minimize $v^T S_W v$, where S_B is the between-class scatter matrix and S_W is the within-class scatter matrix. Recognition is performed in a manner similar to eigenfaces, but using the Fisherfaces for comparison.

2. **Local Binary Patterns Histograms (LBPH)**: For each pixel in the image, compare it with its neighbors by subtracting the central pixel's value and thresholding: Compute the local binary pattern for the central pixel by considering the neighbors in a circularly symmetric neighbor set, construct a histogram from the patterns obtained for the entire image.

Recognition is performed by comparing the histograms of the new image with those of the known images using a metric like the Chi-square distance.

3. **Neural Network-Based Algorithms**:

i. Convolutional Neural Networks (CNNs): CNNs learn to recognize faces through numerous layers of convolution, non-linearity, and pooling. Convolutional layers apply filters to the input to create feature maps:

$$F_{ij}^{l} = \sigma(\sum_{m}\sum_{n} I_{(m+n)\,(j+n)} K_{mn}^{l} + b^{l}) \quad 4.2$$

where F_{ij}^{l} is the feature map, σ is a non-linear activation function, I is the input, K is the kernel for convolution, and b is the bias. Through backpropagation and optimization algorithms like stochastic gradient descent, the network's weights are adjusted to minimize a loss function (e.g., cross-entropy).

ii. DeepFace: DeepFace uses a deep CNN to learn a compact representation of faces. The detailed architecture and loss functions are proprietary, but it follows a similar learning approach as other CNNs.

iii. FaceNet: FaceNet learns a mapping from face images to a compact Euclidean space where distances directly correspond to a measure of face similarity. It uses a triplet loss function that aims to ensure that an anchor image `a` is closer to all other images `p` of the same identity than it is to any image `n` of a different identity by a margin[69].

$$\text{`}\alpha\text{'}: \mathrm{L} = \max\left(||\mathrm{f(a)} - \mathrm{f(p)}||_2^2 - ||\mathrm{f(a)} - \mathrm{f(n)}||_2^2 + \alpha, 0\right) \quad 4.3$$

2.2 Review of Related Works

In a proposal to reduce the errors that occur in the standard attendance system by utilising facial recognition and presented a model that used deep learning algorithms and had a recognition accuracy of 98.3 percent. The method also addresses the problem of face identification within a biometric system when the system is being subjected to real-time settings such as illumination, rotation, and scaling. The input images are obtained by the

model through the usage of a camera. Following that, a linear support vector machine is utilised to detect a face within the image. Finally, attendance is recorded within a spreadsheet before being converted into a PDF file. students from Mumbai University collaborated on the development of an Attendance Management System that is based on ordinary Bluetooth technology. This system ensures that a student's attendance is at least 75% before allowing them to take an examination[6].

In an android-based course attendance system that uses face recognition in a manner that is comparable to that of 46 built an android-based course attendance system that uses face recognition. Electronic tags (serial numbers) are used to permit automatic wireless identification using a Bluetooth smart based device that is developed and set to work in conjunction with an Android application that records attendance as the lecturer goes around the class to detect the tags. The fault in this technique is that it places an excessive amount of reliance on the ownership and use of a Bluetooth-enabled Android phone, which leaves teachers and pupils in the dark regarding the location of important information[55].

In a research that utilized Principal Component Analysis (PCA) so that people could recognise faces. The finished output is a transportable system that can be accessed and used on any computer or mobile phone. The system is able to maintain distinct attendance records for each class and area of study that it supports.has previously published. The following steps are utilised by the system in order to record attendance: To start, a video clip of the classroom is recorded and stored in the database. Next, a count of the number of people present is calculated. Second, the video is then converted to still images, and face detection methods are used to locate the individuals inside the film. In order to extract features (LBPA), the Histogram of Oriented Gradients approach and the Local Binary Pattern method are both utilised. Face recognition is achieved by first projecting a new image in the eigenface subspace, which is a space in which a person is categorised by comparing their position in

eigenface space with that of known individuals, and then extracting the image using the PCA technique. Face recognition software can also use a combination of these two methods. Through the use of the system, the report is delivered to both the instructor and the parents.

Similarly, another studt employed principal component analysis (PCA) to develop an automatic attendance system. They did this by extracting the same distinctive features, which are referred to as eigenfaces. This system is safe, dependable, and does not call for any specialised equipment because it is constructed with a camera, a computer, and face recognition algorithms. Face recognition, which is by far the most important use of the technology, is used for a range of purposes, including safety and surveillance, commerce and education. Even while PCA-based systems are effective, it is quite time consuming to extract the primary components unless a specialised method is used in addition to it[67].

Another study proposed a robust four-layer convolutional neural network architecture as a solution to the challenge of face recognition. This design is capable of handling occlusion, postures, facial emotion, and changeable illumination. The face recognition operation was completed by the model in less than 0.01 seconds when it was evaluated on the AR database. On the AR database, the model achieved an accuracy of 99.5 percent, while on the FERET database, the model achieved an accuracy of 85.15 percent[47].

A novel automated class attendance system based on illumination and an invariant facial recognition system was proposed. In a similar vein, this new system was inspired by comparable ideas. The Viola and Jones algorithms are utilised to accomplish face detection, the Non-subsample contourlet transform is utilised to accomplish face feature extraction, and face classification is completed by comparing an unknown face to a known face included within the class database. The method was evaluated using the use of the 3D face datasets from Yale and BU. In the University's system, face authentication was performed using linear binary pattern histograms and face detection was performed using the Viola-Jones method

(also known as Haar's Cascade). Before being converted to grayscale, the photographs taken by the students were initially stored in a database in their original colour space (RGB) after being obtained from the camera and stored there. Second, the OpenCV framework is used to monitor attendance by comparing an input face to previously stored images of people's faces in a database. Third, when the day comes to a close, the output is printed on a notepad, the system updates the student's attendance, and it communicates this modification to the Department Head. When it comes to these kinds of systems, speed is an asset; nevertheless, utilising a database system that is unable to provide the essential speed as a result of the diverse face inputs may block the conclusion that is wanted[29].

In the Smart Attendance Management System Based on Face Recognition Using CNN, a CNN architecture with 20 layers is utilised. These layers include Conv2D, Batch Normalisation, Max Pooling, and Dense layers, among others. The dataset contains 10,029 face shots that have been separated into four classes, each of which has 2,500 images (with the exception of class-3, which has 2,529 images). This results in a high level of accuracy as a result of the big dataset that was used to train the model; nevertheless, training will take a significant amount of time. The accuracy of the model is 99.86 percent, and it has a loss of 0.0057 percent[52].

In the study Automatic Face Detection and Recognition for Attendance Maintenance, the authors generated their very own dataset by employing still images and frames taken from videos of individuals and fellow students. A number of different combinations of models, including MTCNN, YOLO, FACENET, and SVM, were analysed and compared. They selected the most effective algorithm combination and, as a result, were able to achieve training accuracy of 99.21% in the lowest amount of time feasible, in comparison to the effectiveness of other algorithm combinations. Image processing that begins with pre-processing, including de-noising and quality control checks, is referred to as image

enhancement. The authors identified faces using a pre-trained version of FaceNet and SVM in addition to employing YOLO and MTCNN for detection. They combined these models, which resulted in increased productivity, longevity, and dependability[69].

In the research paper titled Face Recognition for Attendance System Detection, an authentication method based on facial recognition was developed by combining the Haar Cascade methodology with the detection of skin colour. Following the identification of the skin colour in the photographs, the images are then transformed to a grayscale format. After that, the Haar Cascade method is applied in order to identify faces, and after a face has been identified, a bounding box is drawn around it. After that, the face input from the ROI that was obtained in the previous step is aligned for feature extraction. The LBPH algorithm was selected for this work because of its minimal resource utilisation, its ability to detect faces in real time, and its high accuracy of roughly 98.2 percent for photos with a reasonable amount of illumination and 94.7 percent for images with a little less illumination[70].

In a work on Realtime Face-Detection and Emotion Identification Using MTCNN and miniShuffleNet V2 (2019) integrated face detection and emotion recognition into a single model that can complete the task in real time. Facial detection is not achievable if the face tilt is greater than 30%, for example. They used the FER2013 dataset, which comprises of grayscale images with seven different emotions labelled on them. Each photograph is 48 pixels by 48 pixels. The MTCNN and ShuffleNet V2 architectures are utilised in order to capitalise on the benefits offered by both models. These benefits include the detection of faces and the production of bounding boxes, as well as the recognition of emotions to the greatest extent possible. There is space for advancement, as the best level of accuracy on the test data that could be reached was 71.19 percent[71].

In the article titled "A survey on face detection in the wild: Past, Present, and Future, the author conducted research on the history, current state, and potential future of face detection

methods. The contributions made by the community as well as the advancements made in algorithm development are available to the general public. The authors of this work present recent developments in real-world face recognition algorithms in addition to other face detection approaches. The majority of early research failed to achieve the greatest results under uncontrolled conditions and was not successfully deployed in real-world scenarios, despite the fact that numerous face detection and identification technologies are currently available. This issue is resolved by the method developed by Viola and Jones, which makes it possible to carry out essential applications. The Local Binary Pattern technique for face identification and other Yale database approaches are being utilised so that an overall efficiency of 83.2 percent can be achieved. Additionally, the face detection method developed by Viola Jones is also being used. described the automatic Attendance Management system that was based on Face Recognition. In order to improve the overall system performance and efficacy, the author discusses how faces are identified and clipped before doing background removal on the image. Eigen face is recommended by the knowledgeable authors due to the fact that it is simple to use and performs to a high standard when it comes to facial recognition. The document also said that the detection and recognition rate of the face was 45 percent and 10 percent, respectively, in the case of women while wearing a veil, whereas without the veil, it was 93 percent and 87 percent. Bearded men, on the other hand, had a recognition and identification percentage of 79% and 65%, respectively, when compared to the general population[72].

The system that validates the model that aids in facial recognition has been expanded to accommodate 44 users. The Haar classifiers, which use a technique called cascade, are used first, and then Fisher face recognition is performed. When modelling with more than one face and distinguishing characteristics like a cap and glasses, the system is able to deliver ideal efficacy of up to 50 percent within 15 pupils. The proposed method of accessing the

classroom makes use of a video source, and the resulting frames are analysed in order to determine the identities of the students. If the procedures are carried out in the correct order, the overall speed and accuracy of the model will increase.

Face detection and recognition are handled by the Viola Jones and Eigen faces, respectively, in the multiple face identification system developed in a study. Because it is a combination of Eigen face and PCA, it was explained that face identification is independent of the lighting conditions. The face recognition rate is not established at a greater distance than it is in the classroom; nonetheless, various lighting conditions have no impact on the recognition of many faces. By the year 28, an automatic attendance system had been put into place. This system makes use of the facial detection and face recognition algorithm that was developed by Viola Jones. The database of 20 students is initially generated using a number of different head postures before the final recognition results are obtained. After that, the face-finding method was utilised, and its effectiveness was evaluated based on the number of faces that were discovered by the programme. The accuracy of the facial recognition algorithm can be determined by employing the same way[33].

It was recommended in the year 1969 that a lecture attendance system be based on a newly developed technology known as continuous monitoring. In this system, a student's attendance is automatically recorded by a camera that records a photo of the student while they are present in the lecture hall. The construction of the system is basic, and it consists of two cameras that are attached to the wall of the classroom. The first camera is called a capturing camera, and its purpose is to take a picture of a student while they are in class. The second camera is called a sensor camera, and its function is to locate a student's seat within the classroom. The capturing camera is the one that takes the picture of the student. The technology compares the faces shown in the images taken by a camera to those stored in a database, a process that required a lot of practise to perfect.

In a different article that was proposed a real-time computer vision technology was presented as a possible component of an autonomous attendance management system[67].

The system placed an unobtrusive camera in the classroom that is capable of taking images. It then compared the face that was extracted from the image that was taken by the camera with the faces that were already stored in the system. This system made use of machine learning algorithms, which are often applied in the field of computer vision. These algorithms were also utilised by this system. Additionally, HAAR CLASSIFIERS were utilised in order to educate the images that were taken by the camera. After being converted to grayscale and having any unnecessary information removed, the image of the face that was taken by the camera will be uploaded to the server so that it may be processed later.

In addition, in a work that offered a method for achieving automatic attendance that makes use of face recognition technology. The system is able to extract objects from the face, such as the lips and the nose, by utilising MATLAB and Principal Component Analysis (PCA). Concerns with the attendance marking method, such as the amount of time it took, led to the development of system, which was designed to remedy these issues. As a result of the test, the system is now able to identify this piece of paper in the classroom even when there is a dim background or when the face is seen from a different angle[53].

Principal Component Analysis and Artificial Neural Networks were presented as the two distinct algorithms that should be included in the intelligent attendance marking system that was proposed. The traditional way of marking attendance and the time-consuming process are the foci of the author's efforts to find a solution. The method utilises Principal Component Analysis (PCA) to accomplish both the gathering of photographs and the extraction and discovery of commonalities within the face database. Artificial Neural Networks, often known as ANNs, are put to use to either find solutions to problems involving input data or to learn from the data itself, in addition to determining the predicted value. In order to

accomplish what it sets out to do, the method proposed by the author makes use of a back propagation algorithm in conjunction with mathematical functions. According to the findings of the author's research, as a consequence, the system is able to recognise in a wide variety of settings[53].

According to the research paper titled "Attendance System Using NFC (Near Field Communication) Technology with Embedded Camera on Mobile Device," "Attendance System Using NFC (Near Field Communication) Technology with Embedded Camera on Mobile Device" the attendance system can be improved by utilising NFC technology and a mobile application. According to the research report, each student is provided with an NFC tag that has a unique ID at the time of their enrollment into the educational institution. Tapping or relocating these tags will ultimately allow the lecturer's mobile phone to be utilised in the process of tracking attendance for each individual class. After that, the built-in camera on the student's phone will take a photo of the student's face, which will be sent along with the rest of the data to the server at the institution for confirmation and verification. The quick speed at which a connection may be established is one of the benefits of using this technology, along with how simple it is to use NFC. It cuts down the amount of time needed to collect attendance by a significant amount. This system, however, is unable to automatically detect a violation in the event that the NFC tag has not been physically marked by the proprietor of the item in question. Aside from that, the professor found the convenience of the system to be inconvenient. The system uses the lecturer's phone as an NFC reader, which the professor found to be inconvenient. Imagine for a moment that a lecturer arrived at work without their phone. What would transpire? What is the alternative method for keeping track of who was present at the meeting? In addition, the majority of the lecturers probably will not like it if their personal smart phones are exploited in this manner because they are concerned about their privacy. As a result, in place of an NFC tag, the use of

information that is exclusive to the student, such as biometrics or face recognition, which can be relied upon to identify the student should be encouraged. This will serve as evidence that the attendance is being recorded by the correct student[54].

In a study titled "Facial Recognition Based Attendance Marking System", the research is founded on the identification of facial recognition as a means to address the issues that have been raised regarding the previous attendance system. This technology is able to detect and recognise faces by making use of a camera to take photographs of the employee in question. When there is a match found in the face database, the image that was captured is compared one at a time with the face database in order to look for the employee's face, and attendance is then recorded. The fact that attendance is recorded on a server that is well protected and that no one else is able to record the attendance of other individuals is the primary advantage of utilising this method. In addition, the suggested system includes a face detection approach that has been enhanced by the application of a skin categorization technique, which helps to improve the accuracy of the detection process. Even though more work is being made into increasing the precision of the face recognition algorithm, the technology is not yet portable. This is despite the fact that more work is being put into enhancing the accuracy of the algorithm. Because this system requires a computer that is self-contained and has its own dedicated power supply, it cannot be transported from place to place. Because staff members only need to report their presence once per day, this type of system is the only one that is suitable for tracking staff attendance. In contrast, students need to record their attendance for each individual class that they attend on any given day. It will be inconvenient if the mechanism for marking attendance cannot be moved from place to place. In order to solve this issue, the complete system for managing attendance can be constructed as a portable module, which means that it can be put into action by only putting into action a Python programme[55].

In a study on "Fingerprint Based Attendance System Using Microcontroller and LabView", the authors proposed a method of recording attendance by utilising fingerprints as the identifying factor. In this particular system, the process of identifying fingerprints is taken care of by two separate microcontrollers. After the fingerprint pattern has been obtained through the use of a fingerprint sensor, the data will then be transferred to the microcontroller1. After then, the data will be sent from microcontroller 1 to microcontroller 2, which will be responsible for doing the database check. After it is determined that two students are a match, the relevant data is transferred to a personal computer (PC) using serial transmission and then displayed. The development time is cut in half while maintaining design flexibility and simplifying the testing process thanks to this design's clever layout. However, portability is not an option with this system because it must be connected to a computer. Aside from that, the information contained in the database is tough to acquire. This means that parents who are interested in learning about their child's attendance at school will not have an easy or simple way to do so in the future. As a consequence of this, the information on the student can be uploaded to a web server in order to make it simple for the legitimate party who should be concerned to access the information. While a login screen can be used to ensure that only authorised users have access, suitable access authentication can also be enforced[56].

In a research titled "RFID based Student Attendance System, the proposed solution is almost comparable to the one that was presented in the first research magazine. In that publication, RFID technology was used to improve an outdated attendance system. Once more, the attendance of the students may be monitored with the use of a tag and a reader thanks to this setup. The first journal does not allow you to view attendance statistics via a web interface, however this one does. This is the primary distinction between the two journals. The information can be retrieved more quickly and easily as a result. This technology has a

problem in that it cannot be transported from place to place because the RFID reader is dependent on being physically connected to a computer in order to function. Second, the RFID tag does not have any guanine information on it, which is information that may be utilised to identify a student in a way that is completely unique, which results in false attendance statistics[57].

In a study using AttenFace, a standalone system built for real-time attendance tracking in educational settings that makes use of facial recognition, the researchers tracked students' presences in real time using the system's face recognition capabilities. It uses a live camera feed to take still images of the class and then identifies the students in the class, noting their attendance based on whether or not they were present in several still images taken throughout the lesson. The process is entirely hands-free and interacts seamlessly with existing attendance monitoring tools like as Moodle. It solves concerns such as proxy attendance and guarantees that students are physically present in the classroom for the required amount of time to be counted as present. It is equipped with a login gateway, a dashboard for attendance details, facial recognition in real time, and the capacity to manage many classes at the same time[75].

In a study on using facial recognition in employee attendance systems is investigated. The authors place an emphasis not only on the advantages offered by this technology but also on the factors that must be considered during its implementation and any problems that may arise as a result. It explains how facial recognition works, the advantages it offers in terms of accuracy, convenience, and safety, and the factors that should be taken into account when selecting a system. Accuracy, scalability, user friendliness, and the ability to interact with other systems are some of the variables that go into this consideration. The article examines problems surrounding data privacy and security, user acceptance, and environmental factors

that affect the validity of the system. It also explores interaction with workforce management systems like as HR and payroll systems[76].

In a study that offered a web-based application that involves identifying GPS coordinates as well as facial id based upon front-facing cameras for the purpose of registering the attendance of the student, which makes the process simple and more productive, the researchers found that the proposed application made both of these things possible. There is an implementation of the Google GPS API and the camera API within this programme, which enables the application to function more smoothly. The GPS location of the user's device and the camera API are used to validate the user's presence. In addition, the authorization is determined based on the user's face using face recognition technology. This system's functionality is dependent on the location of the campus; hence, with the exception of authorised administrators, it should not be used in places that are located outside of the institution plant. The information on each user's attendance is stored in a database for easy access. The application has a straightforward user interface (UI) that allows the administrator to see the specific reports generated by all of the users. During the next day's entering of their attendance, a user will have the opportunity to receive a report detailing their previous month's activity[77].

In thus work, the face of an individual is utilised for the goal of performing automatic attendance taking. In order to gauge a student's level of proficiency, it is necessary for instructors at every educational establishment to record the individual's attendance. Attendance is recorded in a unique fashion at each and every school. Some of the institutes rely on an approach that is based on paper or files, while others employ techniques of automatic attendance taking that make use of biometric methods, which is a procedure that takes a lot of time. There is a wide variety of approaches that can be used in this regard. The ability to recognise a person's face can be unmistakably distinctive or corroboratory of an

individual by comparing and studying the patterns supported by the contours of a person's face. The primary objective of using facial recognition software at so many different locations is for security considerations. This procedure can be broken down into two distinct phases: the processing that comes before face recognition, in which step-by-step face detection is carried out; and the processing that follows face recognition, in which feature extraction and matching are carried out. Face recognition is utilised by this technology in order to take automatic attendance of students in the classroom. This occurs without the participation of the students themselves. This attendance is recorded with the help of a camera that takes pictures of the students and can recognise their faces; after that, the software consults a database and determines whether or not the students were present based on whether or not their faces were recognised there. A record of attendance will be made, and a message will be relayed to the parents of students number[78].

In a work to develop a face recognition attendance system based on real-time video processing, the phrase "in an article that aims to design" This article focuses primarily on establishing four perspectives from which to evaluate the issues at hand: the accuracy rate of the face recognition system in the actual check-in; the stability of the face recognition attendance system with real-time video processing; the truancy rate of the face recognition attendance system with real-time video processing; and the interface settings of the face recognition attendance system that makes use of real-time video processing. Research is being done on a face recognition attendance system that is based on real-time video processing. The notion of an attendance system that is based on face recognition technology has been developed after conducting an analysis of the situation involving these challenges. According to the findings of the experiments, the accuracy rate of the video facial recognition system can reach as high as 82%. The time needed to check in using the conventional way can be cut by approximately sixty percent when using the facial recognition attendance

system. The phenomena of students departing early and skipping courses has significantly decreased as a result of the rate of students skipping classes. Through the above experimental certification, the face recognition time and attendance system with real-time video processing is able to quickly complete the tasks of students in the time and attendance check-in system, eliminate the complex naming phenomenon, significantly improve the efficiency of class, and play an important role in guiding the development of the time and attendance system[79].

This research paper presents a new method that uses the Local Binary Pattern (LBP) algorithm in conjunction with advanced image processing techniques such as contrast adjustment, bilateral filtering, histogram equalisation, and image blending to address some of the issues that hinder the accuracy of face recognition. The goal of this method is to improve LBP codes, which will ultimately improve the accuracy of the face recognition system as a whole. The findings of our experiments indicate that our approach is very accurate, reliable, and robust for the purpose of developing a face recognition system that is capable of being effectively applied in real-world environments as an automatic attendance management system[79].

In a study that attempted to accomplish digitization of the traditional way of taking attendance by calling names and storing pen-paper records. The methods that are currently being used to take attendance are laborious and time consuming. Manual recording of attendance data makes it straightforward to falsify attendance statistics. Both the conventional method of recording attendance and the already in use biometric technologies are susceptible to being hacked by proxies. It is for this reason that this paper is presented as a solution to all of these issues. The suggested system makes use of many machine learning algorithms, including Gabor filters, KNN, CNN, SVM, Generative adversarial networks, and Haar classifiers. Following the completion of the face recognition process, attendance reports are going to be generated and saved in excel format. The system is put through its paces in a

number of different scenarios, such as with varying degrees of illumination and head movement, as well as with varying distances between the learner and the cameras. Following a battery of meticulous tests, the overall complexity and correctness are determined. The proposed method was demonstrated to be an effective and reliable apparatus for recording attendance in a classroom without requiring any time investment or labour on the part of the instructor[80].

Another research aim to develop a facial recognition-based attendance system that has a high level of confidence and a low rate of false positives. This study exhibits the potential of facial identification by merging two methods known for their resistance to monotonic grayscale transformations: the Local Binary Pattern Histogram (LBPH) algorithm and the Haar cascade algorithm. Both of these algorithms are binary pattern histograms. This results in the creation of a facial map of the subject, which is helpful in the post-image processing of the image of the individual collected during attendance. This technology is able to recognise students in spite of the fact that some of them may have facial hair or may be wearing eyeglasses. In comparison to more conventional approaches, the effectiveness of this method was significantly higher; despite this, it did have a number of drawbacks that could be easily remedied by improving the surrounding environment and using deep learning through machine computing with the use of artificial intelligence[81].

In a research to accurately record attendance for each and every student in a given class. In the system that has been proposed to take the attendance of all of the students at the same time, a live video is processed for each frame, and to recognise the faces of all of the students, it uses a deep convolutional neural network face recognition algorithm that extracts 128-dimensional face encodings from images and then compares these encodings with the faces that have been stored in the dataset to determine the best match. Additionally, the attendance of the students who are present in the classroom is recorded using an excel sheet The results

of the experiment overcame the challenges presented by the existing technologies and provided a glimpse into the transition towards a more futuristic method of recording attendance[82].

Endnotes

1. Y Kortli, M Jridi, A Al Falou, M Atri. *Face recognition systems: A survey. Sensors* 2020;20:342. https://doi.org/10.3390/s20020342.

2. RV Virgil Petresc. *Face recognition as a biometric application*. **Journal of Mechatronics and Robotics** 2019;3:237–57. https://doi.org/10.3844/jmrsp.2019.237.257.

3. T Tran. *Automatic face recognition system using principal component analysis and eigenface approach*. **International Research Journal of Modernization in Engineering Technology and Science** 2022. https://doi.org/10.56726/irjmets30006.

4. EC Sai, SA Hussain SA, S Khaja, A Shyam. *Student attendance monitoring system using face recognition.* **SSRN Electronic Journal 2021**. https://doi.org/10.2139/ssrn.3851056.

5. N Aljojo, R Alotaibi, B Alharbi, A Alshutayri, J Tariq, A Banjar. *Alzheimer assistant: a mobile application using Machine Learning*. Revista Română de Informatică Și Automatică 2020;30:7–26. https://doi.org/10.33436/v30i4y202001.

6. I Adjabi, A Ouahabi, A Benzaoui, A Taleb-Ahmed. *Past, present, and future of face recognition: A review*. Electronics 2020;9:1188. https://doi.org/10.3390/electronics9081188.

7. A Hassouneh, AM Mutawa, M Murugappan. *Development of a real-time emotion recognition system using facial expressions and EEG based on machine learning and deep neural network methods.* Informatics in Medicine Unlocked 2020;20:100372. https://doi.org/10.1016/j.imu.2020.100372.

8. P PengI Portugal, P Alencar, D Cowan. *A face recognition software framework based on principal component analysis*. PLOS ONE 2021;16:e0254965. https://doi.org/10.1371/journal.pone.0254965.

9. MdP Uddin, MdA Mamun, MdA Hossain. P*ca-based feature reduction for hyperspectral remote sensing image classification*. IETE Technical Review 2020;38:377–96. https://doi.org/10.1080/02564602.2020.1740615.

10. TI Simanjuntak, M Muhathir, F Fadlisyah, I Safira. *Performance analysis of naive bayes variation method in spice image classification using histogram of gradient oriented (hog) feature extraction*. **Journal Of Informatics And Telecommunication Engineering** 2023;7:282–91. https://doi.org/10.31289/jite.v7i1.7957.

11. AS Nadhan, C Tukkoji, B Shyamala, N Dayanand Lal, AN Sanjeev Kumar, V Mohan Gowda. *Smart attendance monitoring technology for industry 4.0*. **Journal of Nanomaterial**s 2022;2022:1–9. https://doi.org/10.1155/2022/4899768.

12. E Chhatar, H Chauhan, S Gokhale, S Mukherjee, N Jha. "*Survey of student attendance management system*", S.B. Jain Institute , Nagpur. 2016.

13. G. J. Edwards, C. J. Taylor, and T. F. Cootes, "I*nterpreting face images using active appearance models*," Proc. Third IEEE Int. Conf. Autom. Face Gesture Recognit., 2015.

14. M.P Gawande & DG Agrawal. *Face recognition using PCA and different distance classifiers*. **IOSR Journal of Electronics and Communication Engineering** (IOSR-JECE), 9(1), pp. 1-5. 2014.

15. RM Ghofrani & S. Ghanbari, "Realtime face-detection and emotion recognition using MTCNN and miniShuffleNet V2," 5th Conference on Knowledge Based Engineering and Innovation (KBEI), 2019.

16. J. Joseph and K. P. Zacharia, "Automatic Attendance Management System Using Face Recognition," Int. J. Sci. Res., vol. 2, no. 11, pp. 327–330, 2013.

17. J. Kanti and A. Papola, "Smart Attendance using Face Recognition with Percentage Analyzer," vol. 3, no. 6, pp. 7321–7324, 2014.

18. J Taylor. *Managing information technology projects*, p.39, 2014.

19. J Joseph & KP Zacharia. *Automatic attendance management system using face recognition* Department of Electronics & Communication Engineering, Saintgits College of Engineering, Pathamuttom P.O., Kottayam-686532, Kerala, India, **International Journal of Science and Research (IJSR)** ISSN (Online): 2319-7064.

20. J Lio. "*Attendance management system using a mobile device and a web application*", Department of Socio-informatics, Faculty of Letters Chuo University 742-1 Higashinakano, Hachioji-shi, Tokyo 192-0393, 2016.

21. K. Kendall, J. Kendall. system analysis and design 8th edition, Pearson Education, Inc., 2011.

22. KD Kadam. *Face recognition using principal component analysis with DCT*. **International Journal of Engineering Research and General Science,** ISSN, pp.2091-2730. 2014.

23. S Kapoor, S Khanna & R Bhatia. *Facial gesture recognition using correlation and mahalanobis distance*. 2010.

24. S Kewalramani. *Automatic attendance system by face recognition using machine learning*. **International Journal of Engineering Sciences and Research Technology.** 7(10): 116 - 121. 2018.

25. M Murtaza, M Sharif, M Raza & J Shah. *Face recognition using adaptive margin fisher's criterion and linear discriminant analysis*. **International Arab Journal of Information Technology**, 11(2), pp.1-11. 2014.

26. N. Darapaneni, "*Automatic face detection and recognition for attendance maintenance,"* IEEE 15th International Conference on Industrial and Information Systems (ICIIS), pp. 236-241 2020.

27. GR Naik & DK Kumar *An overview of independent component analysis and its applications*. Informatica, 35(1). 2019.

28. KJ Nazare, B Surekha."*Attendance management system using hybrid face recognition techniques"* Conference on Advances in Signal Processing (CASP) Cummins College of Engineering for Women, Pune. Jun 9-11, 2019.

29. B Ozmen & K Yurtkan. *Automatic exam attendance system based on illumination invariant face recognition*. International Conference on Education in Mathematics, Science and Technology. 2: 139 - 143. 2018.

30. P. Mehta, "An efficient attendance management sytem based on face recognition using matlab and raspberry pi 2," **Int. J. Eng. Technol. Sci. Res. IJETSR**, vol. 3, no. 5, pp. 71–78, 2020.

31. P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "*Eigenfaces vs. fisherfaces: Recognition using class specific linear projection,*" IEEE Trans. Pattern Anal. Mach. Intell., 2019.

32. H Pooja, K Shivani, K., Apurva, J Akshata, J & B Krushna. *Automatic attendance using face recognition*. **International Journal on Recent and Innovation Trends in Computing and Communication**. 5(12): 24 - 26. 2017.

33. W Priyanka, T Roshani, P Shweta, C Jagruti"*Attendance system based on face recognition using Eigen face and PCA Algorithms*" International Conference on Green Computing and Internet of Things (ICGCIoT) IEEE 2020.

34. R. Hartanto & M. N. Adji, "*Face recognition for attendance system detection*," 10th International Conference on Information Technology and Electrical Engineering (ICITEE), pp. 376-381, 2018.

35. S Refik. *Face recognition-based mobile automatic classroom attendance management system*. International Conference on Cyberworlds, 2017.

36. P Riddhi & B Shruti. *A literature survey on face recognition techniques*. **International Journal of Computer Trends and Technology**. 5(4): 189-193. 2023.

37. L Riya, G Suruchi, J Harshil & N Harish. *Bluetooth smart based attendance management system*. International Conference on Advanced Computing Technologies and Applications. pp. 524 - 527, 2021.

38. S. Jeng, H. Y. M. Liao, C. C. Han, M. Y. Chern, and Y. T. Liu, "*Facial feature detection using geometrical face model: An efficient approach*," Pattern Recognit., vol. 31, no. 3, pp. 273–282, 2022.

39. RK Sahu, YP Singh, Y.P. and Kulshrestha, A.. *A comparative study of face recognition system using PCA and LDA*. **International Journal of IT, Engineering and Applied Sciences** Research, 2(10), 2019.

40. G Salh, G., Mansour, A. H., and Mohammed, M. F. E. *Hand Geometric Recognition System based on Support Vector Machines (SVM)*. **International Journal of Advanced Research in Computer and Communication Engineering**, vol. 4, no. 3, 2019, pp. 445–452.

41. DH Shah, DJ Shah & D.T.V Shah. *The exploration of face recognition techniques*. **International Journal of Application or Innovation in Eng. And Management (IJAIEM)**, 2020.

42. N SharmDubey. *Face recognition analysis using PCA, ICA And Neural Network*. **International Journal of Digital Application & Contemporary research, 2(9). 2020.**

43. M Slavković & D AnJevtić.,. *Face recognition using eigenface approach. Serbian* **Journal of Electrical Engineering**, 9(1), pp.121-130. 2018.

44. H Smit, P Nikhil, P Nandana, P Mayur – "A*utomated attendance system using image processing*" Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), 2018.

45. Z Stefanos, Z Cha, Z Zhengyo, *A survey on face detection in the wild: Past, present and future,* Computer Vision and Image Understanding, Volume 138, 2021, Pages 1-24, ISSN 1077-3142. 2015.

46. D Sunaryono, J Siswantoro & R Anggoro. *An android based course attendance system using face recognition*, **Journal of King Saud University** - Computer and Information Sciences. 2019.

47. A Syafeeza, R Khalil-Hani, M Liew, S.S., Bakhteri. *Convolutional neural network for face recognition with pose and illumination variation*. **International Journal of Engineering and Technology**. 6(1): 44-46, 2014.

48. T. H. Le, "*Applying artificial neural networks for face recognition*," Adv. Artif. Neural Syst., 2019.

49. T. Ojala, M. Pietikainen, and D. Harwood, "*Performance evaluation of texture measures with classification based on Kullback discrimination of distributions," in Pattern Recognition*. Vol. 1 - Conference A: Computer Vision amp; Image Processing., Proceedings of the 12th IAPR International Conference on, 2014.

50. V. Shehu and A. Dika, "U*sing real time computer vision algorithms in automatic attendance management systems,*" Inf. Technol. Interfaces (ITI), 2014.

51. W. Deng, J. Hu, and J. Guo. ``*Face recognition via collaborative representation: Its discriminant nature and superposed representation*," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 10, pp. 2513-2521, 2018.

52. Wang, L., Zhang, Y. and Feng, J. *On the Euclidean distance of images, IEEE transactions on pattern analysis and machine intelligence*, 27(8), pp.1334-1339. 2015.

53. K Yamini, K Mohan, P Sonia, V Yugandhar & K Bharath. *Class attendance using face detection and recognition with opencv*, **International Research Journal of Engineering and Technology**. 6(4): 3822 - 3825. 2019.

54. B Khichi, K Lokare. "*Attendance system using nfc technology with embedded camera on mobile device*" 2018.

55. K.SenthamilSelvi, P.Chitrakala, A.AntonyJenitha, "Face recognition based attendance marking system", IJCSMC, Vol. 3, Issue. 2, 2021.

56. Y Kumar, P Singh, K Mishra, "*fingerprint based attendance system using microcontroller and labview*" 2017

57. D Hussain, H Deka"RFID based Student Attendance System" 2019.

58. OpenCvDocumentation -https://opencv.org

59. Numpy - https://numpy.org

60. A brief history of facial recognition. URL: https://www.nec.co.nz/market-leadership/publications-media/abrief-history-of-facial-recognition/ 2020

61. OpenCV Documentation [online]. URL: https://opencv.org/

62. D Farah, A Aftab, M Hira "*Real time face recognition of human faces by using lbph and viola jones algorithm.*" . URL: https://pdfs.semanticscholar.org/3255/0898eec9c4424932e70e5e32c98 b0220a747.pdf. 2019

63. S Chatterjee, A Jana, A Ganguly & A Ghosh. Automated attendance system using face recognition technique. **International Journal of Engineering and Applied Sciences (IJEAS),** 5(7) 2019.

64. F Hannan, F Abdul. Comparative analysis of face recognition methodologies and techniques," 2019.

65. Y Taghizadegan, H Ghassemian & M NaserMoghaddasi. 3d face recognition method using 2DPCA-Euclidean Distance Classification. **ACEEE International Journal on Control System and Instrumentation**, 3(1). 2018

66. MD Malkauthekar & SD Sapkal. Comparision of mahalanobis and manhattan distance measures in pca based face recognition. International Journal of Computer & Technology (IJCET), 5(5), pp 01-11. 2018

67. V. Shehu & A. Dika, "Using real time computer vision algorithms in automatic attendance management systems," Inf. Technol. Interfaces (ITI), 2019 32nd Int. Conf., pp. 397–402, 2019.

68. J. Kanti & A. Papola, "*Smart Attendance using Face Recognition with Percentage* Analyzer," vol. 3, no. 6, pp. 7321–7324, 2019.

69. Y. Kawaguchi. *Face recognition-based lecture attendance system*," 3rd AEARU …, no. October, 2018.

70. N. Kar, M. K. Debbarma, A. Saha, and D. R. Pal, "study of implementing automated attendance system using face recognition technique," **Int. J. Comput. Commun. Eng.**, vol. 1, no. 2, pp. 100–103, 2012.

71. S Chatterjee, A Jana, A Ganguly & A Ghosh. Automated attendance system using face recognition technique. **International Journal of Engineering and Applied Sciences (IJEAS),** 5(7) 2019.

72. M Murtaza, M Sharif, M Raza & J Shah. *Face recognition using adaptive margin fisher's criterion and linear discriminant analysis*. **International Arab Journal of Information Technology**, 11(2), pp.1-11. 2014.

73. J. Joseph and K. P. Zacharia. *Automatic attendance management system using face recognitio*n. **Int. J. Sci. Res**., vol. 2, no. 11, pp. 327–330, 2013.

74. S. Jeng, H. Y. M. Liao, C. C. Han, M. Y. Chern, and Y. T. Liu, *Facial feature detection using geometrical face model: An efficient approach," Pattern Recognit*., vol. 31, no. 3, pp. 273–282, 1998.

75. Rao A. AttenFace: A Real Time Attendance System Using Face Recognition. In2022 IEEE 6th Conference on Information and Communication Technology (CICT) 2022 Nov 18 (pp. 1-5). IEEE.

76. Jabez J, Keerthanaa V, Kaviya V, Gowri S. An Enhanced Web Based Attendance Application Using Global Positioning System and Face Recognition. Journal of Computational and Theoretical Nanoscience. 2020 Aug 1;17(8):3344-8.

77. J Jabez, V Keerthanaa, V Kaviya, S Gowri. *An enhanced web based attendance application using global positioning system and face recognition*. **Journal of Computational and Theoretical Nanoscience**. 2020 Aug 1;17(8):3344-8

78. A Praveena, MK Eriki, DT Enjam. *Implementation of smart attendance monitoring using open-CV and python*. **Journal of Computational and Theoretical Nanoscience**. 2019 Aug 1;16(8):3290-5.

79. H Yang, X Han. *Face recognition attendance system based on real-time video processing*. IEEE Access. 2020 Jul 10;8:159143-50

80. S Dev, T Patnaik. *Student attendance system using face recognition*. In2020 international conference on smart electronics and communication (ICOSEC) 2020 Sep 10 (pp. 90-96). IEEE.

81. S Barik, S Mohanty, D Singh, SN Sahoo, S Sahoo. *Real-time facial recognition based smart attendance management system using haar cascading and lbph algorithm*. In2023 International Conference on Communication, Circuits, and Systems (IC3S) 2023 May 26 (pp. 1-6). IEEE.

82. PN Kulkarni, TS Anjali, B Rohith, R Nadagouda, S Veeresh. *Novel Approach to Record the Attendance of Students Using Facial Recognition*. In2023 International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE) 2023 Apr 29 (pp. 1-6). IEEE

Chapter Three

Methodology

3.1 Software Analysis and Design

The System Development Life Cycle (SDLC) research approach will be employed in the realization of the software design objectives of this research having investigating the need for automating Health Insurance Management Information. The project life cycle covers all project activities, whereas the systems development life cycle concentrates on meeting product requirements[1]. The SDLC is utilized during the creation of an Information Technology (IT) project, and it explains the many stages involved in the project from conception to completion.

3.1.1 Phases of Waterfall Model:

i. Requirement Analysis and Definition

ii. System and Software Design

iii. Implementation and testing

iv. System testing

v. Maintenance

Requirement Analysis and Definition: The system's potential users will be consulted at this stage to determine the system's goals, requirements, and services that the end-user requires and expects. This entailed a proper characterization of the problem's scope and peculiarities. The security flaws in the current way of designing health assurance applications are the problem on which this project is built.

System and Software Design: The method of defining the interface, architecture, modules, components, and data for a system in order to meet the defined requirements is referred to as design. The philosophy of application systems to system development can be considered as systems design. The design was built to assist in defining the software and hardware needs as

well as outlining the entire system architecture, based on the criteria stated in the first stage. The utilization of System Architecture and Use Cases are examples of this.

Implementation and Unit Testing: This entails writing actual code to implement the design's system needs. This comprises constructing the graphical user interface, implementing the model in PHP, and setting up the system database in MySQL.

System Testing: After the coding is complete, every unit of the system is tested and then incorporated as a full system to ensure that the entire system performs according to the requirement specification.

Maintenance: This is the final stage of development, during which all necessary maintenance tasks were completed to ensure that the software would continue to function even if there was a future development

3.2 Software Requirements and Approach

The basic requirements of the system were identified and outlined. The system consists of the two major use case namely the Administrator and the Employee. The Administrator acts as the moderator over the entire application while the application is tailored to address the Employer which acts as the nominal user of the developed system. The system consists of the Employee Registration Module which involves two phase registration, personal information registration and facial image capturing, the facial image training module using computer vision techniques, the employee attendance module and the administrator module.

The graphical user interface was designed to produce an executable application with Matlab Guide (GUI), the MATLAB programming language was used to also developed and build every facial recognition module inclusive with all image processing algorithms, while also creating connections and database relationship with the backend engine use via the MATLAB ODBC connector. The MYSQL was integrated with MATLAB as the database engine. MySQL database tables are then created using the PHPMyAdmin database to store the users'

information as they are submitting their information through various provided forms. The implemented system consists of three use case models namely the health care insurance company, the hospitals and the enrolees/patients.

The Employer is privy to the administrative role of the implemented system, which invariably give the admin the access to moderate over all registered employees. The employer moderates over the system by generating attendance reports, deleting unavailable employers, editing employee's information, capturing of employees faces and training of all captured faces. The employee which are the staffs of the human resource unit of the Computer Department of Leads University Ibadan are required to perform image capturing which involves capturing fove to six frontal poses of their face, registration of their job personnel information will also be collected which will serve as an input to the developed system. The staff are also required to perform every morning attendance marking by providing their face at the system webcam at the administrative desk.

3.3 Requirement Specification

3.3.1 Functional Requirement

A functional requirement specifies a system's or component's function. A function is defined as a set of inputs, outputs, and behavior. The Development of the Employee Attendance Tracking Using Facial Recognition System Information consists of two use case models namely the Employer (Administrator). The input design was modelled using the MATLAB Graphical User Interface (GUIDE). This will give the users the best experience to use the application which runs an executable file. The system development modules and analysis is shown below for extensive breakdown.

1. Administrative Module
2. Employee Registration Module
3. Face Training Module
4. Employee Attendance Module

Administrative Module

The Administrative Module or (employer module) of the system performs the following function in the system

1. Login: admin user will login into the system as an admin user
2. Moderate Employee's Account: In moderating the employee's, the admin can:
    i. Capture staff's facial pose.
    ii. Register staff personal information.
    iii. Delete staff account if necessary
    iv. Train the facial recognition system anytime there are new facial updates.
    v. Generate attendance report

Employee Registration Module

The Staff Registration Module of the system performs the following function in the system

1. Register personal information
2. Perform face image capturing with different poses

Face Training Module

The face training module of the system performs the following function in the system by the administrator

1. Image pre-processing
2. Feature extraction
3. Saving of facial patterns
4. Training of saved features

Employee Attendance Module

The employee attendance module triggers the face recognition process to mark attendance, the process is highlighted below.

1. Present face at administrative desk webcam.
2. Face recognition check
3. Mark attendance if subject is found in database

3.4 Face Recognition Design

Local facial images of the staffs of the human resource unit of the Computer Department of Leads University Ibadan will we adopted as the facial database collection. The, performance evaluation of the face recognition model will be obtained in terms of training time, testing time, accuracy and recognition rate. The face recognition design consists of number of modules: geometric extraction, Illumination Control, Feature extraction, classification and performance evaluation.

Image capturing: The image acquisition toolbox of the MATLAB will be integrated to capture real time live images from the staffs.

Image Prepossessing: In this module the images are normalized to improve the recognition of the system. The pre-processing steps implemented are as follows:

• Image size normalization

• Illumination normalization: This was done with the three techniques

I. Histogram Equalization
II. Adaptive Histogram Equalization
III. Contrast Adaptive Histogram Equalization

Feature Extraction: Principal component analysis (PCA) is a statistical dimensionality reduction method, which produces the optimal linear least-square decomposition of a training set. Kirby and Sirovich applied PCA for representing faces and Turk and Pentland extended PCA for identifying faces. In applications such as image compression and face recognition a helpful statistical technique called PCA is utilized and is a widespread technique for determining patterns in data of large dimension. The PCA approach is then applied to reduce

the dimension of the data by means of data compression, and reveals the most effective low dimensional structure of facial patterns. In case of this approach a complete front view of face is needed; or else the output of recognition will not be accurate. The major benefit of this method is that it can trim down the data required to recognize the entity to 1/1000th of the data existing.

The following steps summarize the process PCA. Let a face image $X(x, y)$ be a two dimensional m x n array of intensity values[2]. An image may also be considering the vector of dimension mn, so that a typical image of size 112x92 becomes a vector of dimension 10304. Let the training set of images

$\{X_1, X_2, X_3 \ldots X_N\}$. The average face of the set is defined by[3]:

$$\bar{X} = \frac{1}{N}\sum_{i=1}^{N} X_i \tag{3.1}$$

Calculate the estimate covariance matrix to represent the scatter degree of all feature vectors related to the average vector. The covariance matrix C is defined by[4]:

$$C = \frac{1}{N}\sum_{i}^{N}(\bar{X} - X_i)(\bar{X} - X_i)^{\mathrm{T}} \tag{3.2}$$

3.5 Training

The obtained feature from the PCA will be used to train the system and create a pattern extraction for each of the supplied face emotion via the support vector machine.

3.5.1 Classification or Recognition.

After feature extraction step next is the classification step which makes use of Euclidean Distance for comparing/matching of the test and trained images[5]. In the testing phase each test image is centred and into the same eigenspace as defined during the training phase. This

projected image is now compared with projected training image in eigenspace. Images are compared with similarity measures. The training image that is the closest to the test image will be matched and used to identify. Calculate relative Euclidean distance between the testing image and the reconstructed image of ith person, the minimum distance gives the best match. The Euclidean distance was used for the distance similarity measure.

3.6 System Architecture

The system architectures show the framework of the implemented system in alignment with the system use cases and modules.

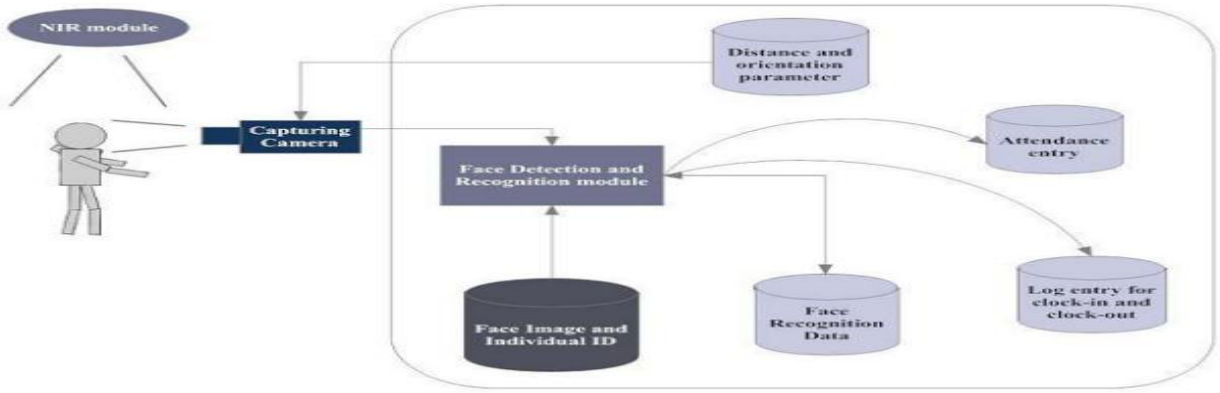

Figure 3.1: System Framework[6].

The image depicts a flowchart for a facial recognition-based attendance system. The process illustrated can be explained thus:

NIR Module: Used to improve the capturing of images in various lighting conditions. It helps in capturing better quality images that are necessary for accurate facial recognition.Capturing Camera: This is the hardware that captures the images of individuals to be processed. It may work in tandem with the NIR module for enhanced image quality.

Face Detection and Recognition Module: Once the image is captured, it is sent to this module. The face detection algorithm first locates the face within the image, and then the face recognition algorithm compares the detected face against a database of known faces to identify the individual[7].

Face Image and Individual ID: If the face is recognized, the corresponding individual's ID is retrieved from the database. This ID will be used to mark attendance.

Distance and Orientation Parameter: This check within the system to ensure that the face is at the correct distance and orientation for accurate recognition. It may be a part of the recognition module or a separate check before or after the recognition process.

Face Recognition Data: This is a database or repository where the face data and recognition logs are stored.

Attendance Entry: After successful recognition and retrieval of the individual ID, an attendance entry is logged.

Log Entry for Clock-ins and Clock-outs: This system not only registers attendance but also records the time of entry and exit (clock-ins and clock-outs) for individuals.

3.6.1 System Flowchart

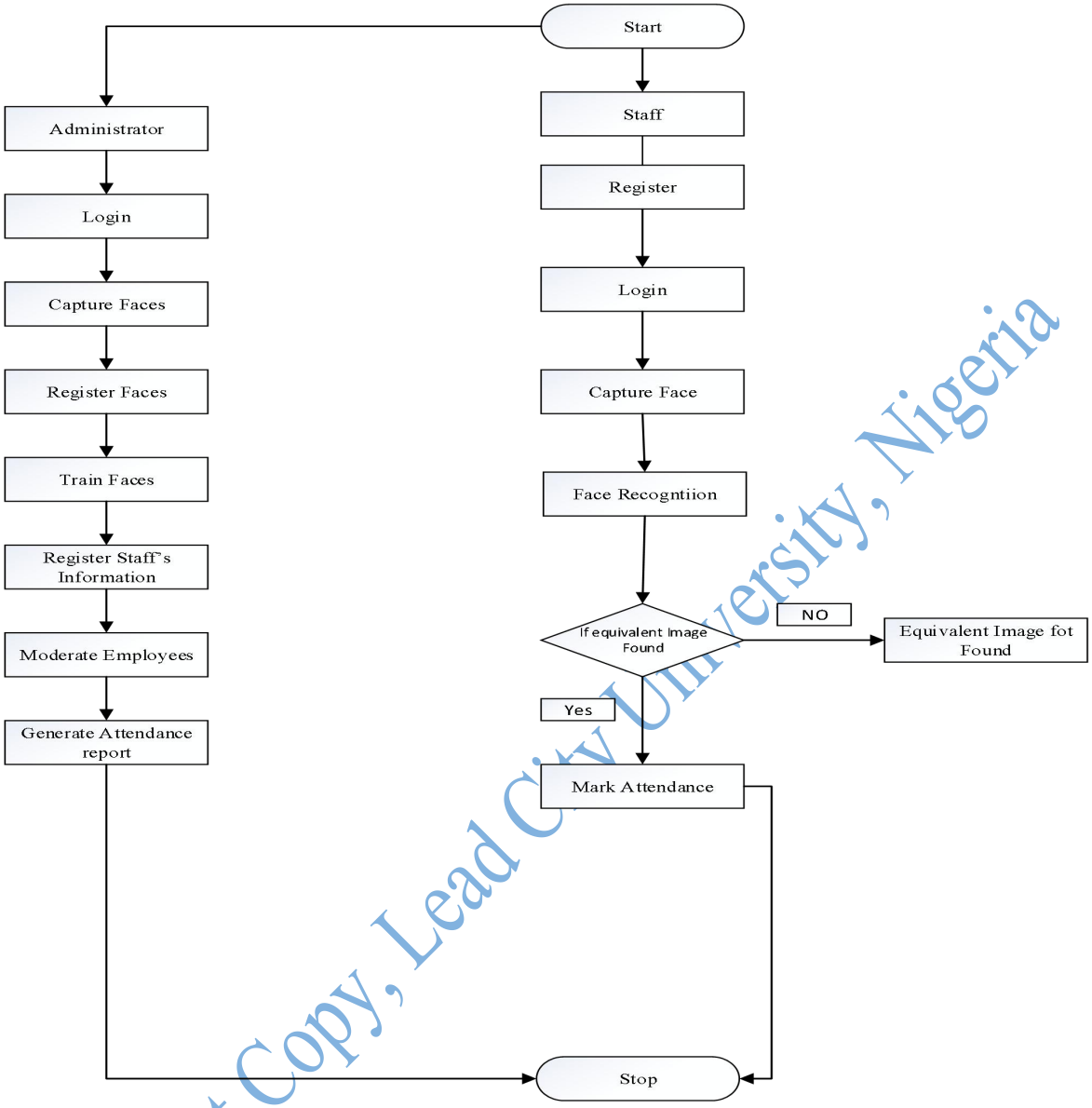

Figure 3.2: System Flow Chart
Source: Research Design, 2023

3.6.2 Use Case Diagram

The use case diagram is a representation of a user's interaction with the system i.e. it identifies all possible actions that can be performed on the system by the user.

A. Employer (Administrator) Use case

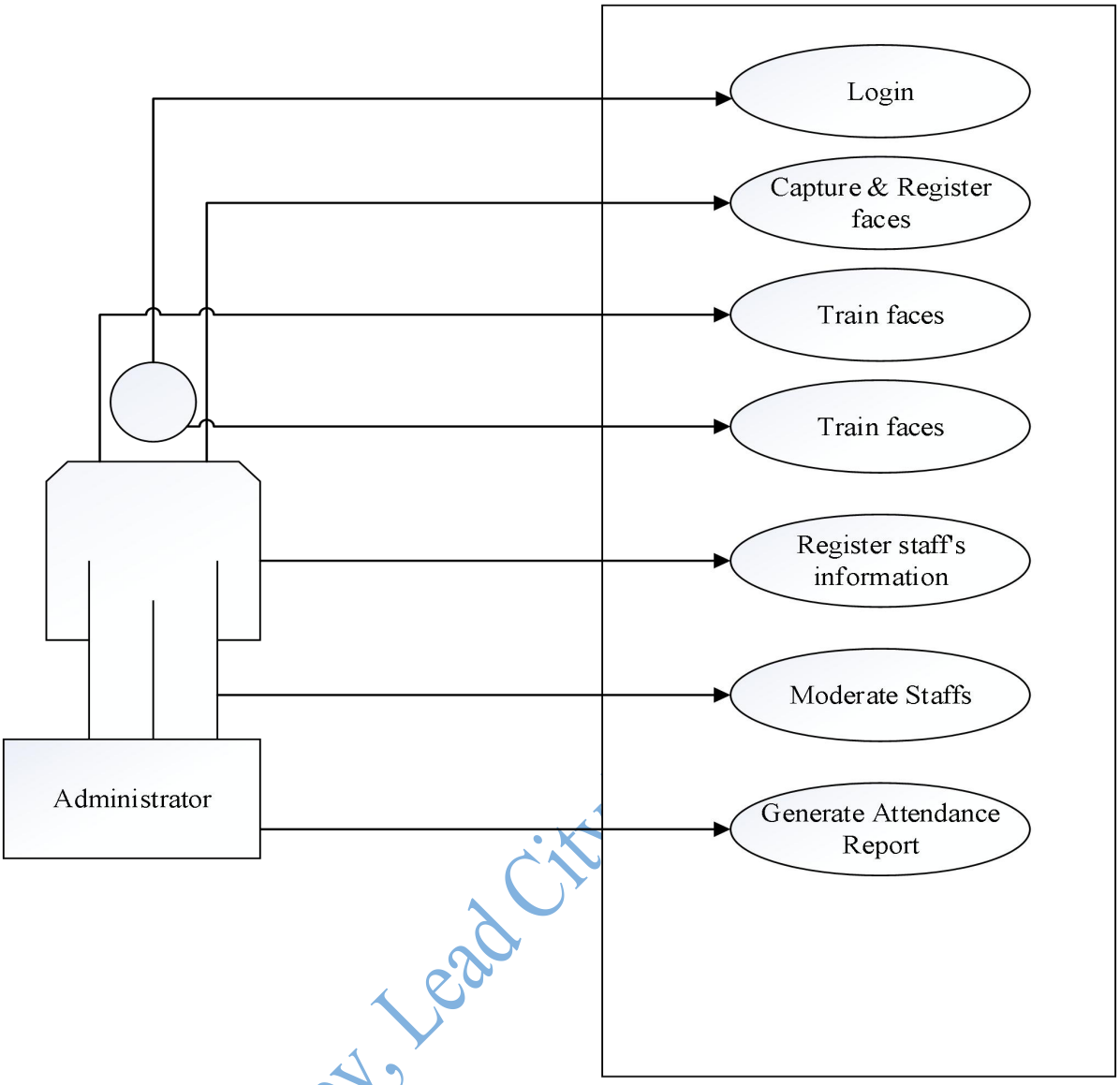

Figure 3.3: Administrator Use Case
Source: Research Design, 2023

The image is a usecase diagram representing the responsibilities or tasks of an "Administrator" within a facial recognition system for attendance tracking. The tasks are sequenced as follows:

Login: The Administrator must first log into the system, which implies a secure access point requiring credentials.

Capture & Register Faces: This step involves taking pictures of individuals' faces and entering them into the system. It's likely the initial enrollment phase where staff members' facial data is collected.

Train Faces: After capturing and registering faces, the Administrator must run or oversee the process of 'training' the system. This usually means that the facial recognition software is learning to identify the registered faces by analyzing the features and patterns of the faces.

Register Staff's Information: Apart from facial data, the Administrator may also enter or update other relevant staff details into the system.

Moderate Staffs: This might involve managing staff profiles, updating information, or handling permissions within the system.

Generate Attendance Report: The final listed task is to compile and produce reports on attendance, likely using the data from the facial recognition logs to show who was present, absent, late, etc.

This flowchart outlines a systematic approach to managing an automated attendance system, where facial recognition technology is used to verify the identity of staff members.

B. Employee (Staff) Use case

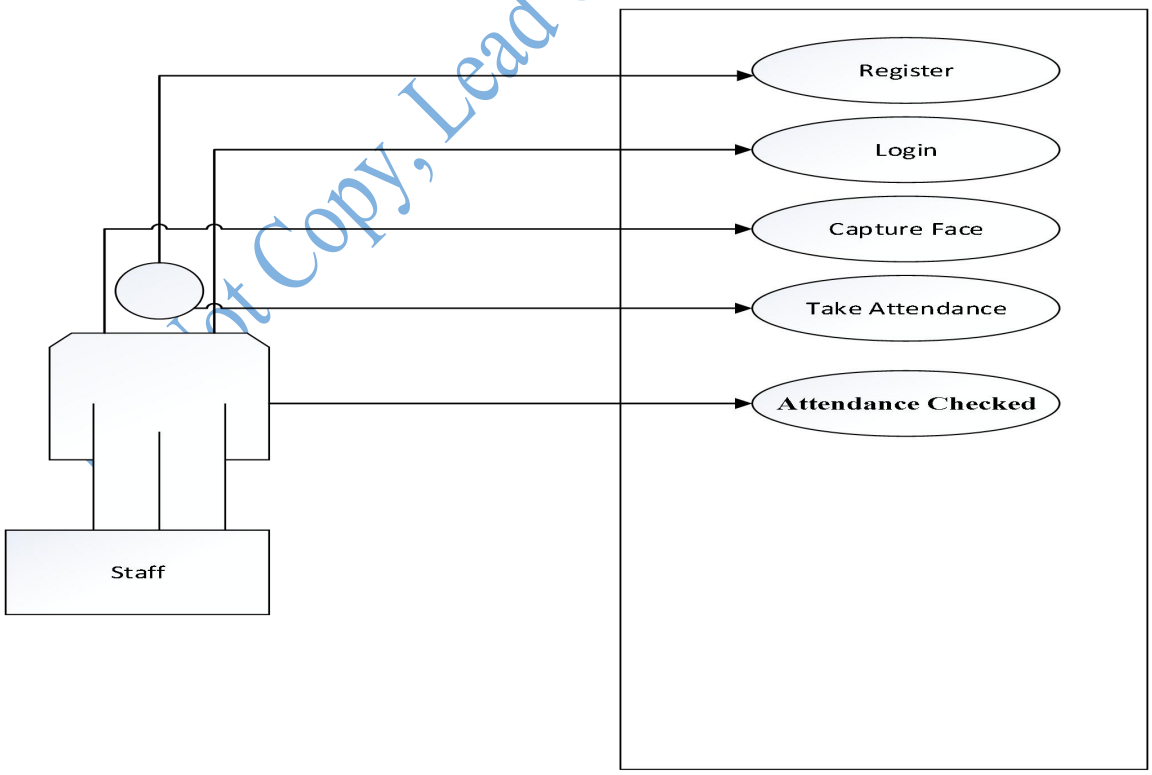

Figure 3.4: Staff Use Case
Source: Research Design, 2023

The image depicts a use case diagram that outlines the steps a staff member needs to follow within a facial recognition attendance system:

Register: The staff member's initial step is to register in the system, which likely involves providing personal details and creating an account.

Login: After registration, the staff member logs into the system, presumably using credentials set up during the registration process.

Capture Face: The system captures the staff member's face, which is a part of the attendance marking process. This step might involve the staff member positioning themselves in front of a camera designed to recognize and verify their identity.

Take Attendance: This step likely refers to the system's process of marking the staff member as present after their face has been recognized.

Attendance Checked: This final step indicates that the staff member's attendance has been verified and logged into the system.

This usecase diagram provides a high-level overview of the interaction between staff and an automated facial recognition attendance system, showing the user journey from registration to the confirmation of attendance.

3.7 Choice of Programming Language

MATLAB: The application was crafted with the MATLAB IDE. MATLAB is a high-performance language for technical computing. It integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation[8]. Typical uses include:

1. Math and computation
2. Algorithm development

3. Modeling, simulation, and prototyping
4. Data analysis, exploration, and visualization
5. Scientific and engineering graphics
6. Application development, including Graphical User Interface building

MATLAB is an interactive system whose basic data element is an array that does not require dimensioning. This allows you to solve many technical computing problems, especially those with matrix and vector formulations, in a fraction of the time it would take to write a program in a scalar noninteractive language such as C or Fortran.

The name MATLAB stands for matrix laboratory. MATLAB was originally written to provide easy access to matrix software developed by the LINPACK and EISPACK projects, which together represent the state-of-the-art in software for matrix computation.

MATLAB has evolved over a period of years with input from many users. In university environments, it is the standard instructional tool for introductory and advanced courses in mathematics, engineering, and science. In industry, MATLAB is the tool of choice for high-productivity research, development, and analysis.

MATLAB features a family of application-specific solutions called toolboxes. Very important to most users of MATLAB, toolboxes allow you to learn and apply specialized technology. Toolboxes are comprehensive collections of MATLAB functions (M-files) that extend the MATLAB environment to solve particular classes of problems. Areas in which toolboxes are available include signal processing, control systems, neural networks, fuzzy logic, wavelets, simulation, and many others[9].

MySQL: MYSQL is a relational database management system that is free and open source. It is based on the structure query language, and it provides constant quick performance, high dependability, and simplicity of usage[10]. MySQL is a database management system. In MySQL, data is saved in database objects known as tables. A

table is made up of columns and rows and contains a collection of connected data elements.

Endnotes

1. P Riddhi & B Shruti. *A literature survey on face recognition techniques*. **International Journal of Computer Trends and Technology**. 5(4): 189-193. 2018.

2. L Riya, G Suruchi, J Harshil, N Harish. *Bluetooth smart based attendance management system*. International Conference on Advanced Computing Technologies and Applications. pp. 524 - 527, 2019.

3. S. Jeng, H. Y. M. Liao, C. C. Han, M. Y. Chern, and Y. T. Liu. *Facial feature detection using geometrical face model: An efficient approach,"* Pattern Recognit., vol. 31, no. 3, pp. 273–282, 2019.

4. RK Sahu, YP Singh, & A Kulshrestha, A. *A Comparative Study of Face Recognition System Using PCA and LDA*. **International Journal of IT, Engineering and Applied Sciences Research**, 2(10), 2020.

5. G Salh, AH Mansour & M.F.E Mohammed. *Hand geometric recognition system based on Support Vector Machines (SVM)*. **International Journal of Advanced Research in Computer and Communication Engineering**, vol. 4, no. 3, 2015, pp. 445–452.

6. DH Shah, DJ Shah & DTV Shah. The Exploration of Face Recognition Techniques. International Journal of Application or Innovation in Eng. And Management (IJAIEM), 2019.

7. N Sharma & APSK Dubey. Face recognition analysis using PCA, ICA And Neural Network. **International Journal of Digital Application & Contemporary research**, 2(9). 2020.

8. M Slavković & D Jevtić. Face recognition using eigenface approach. Serbian Journal of Electrical Engineering, 9(1), pp.121-130. 2022.

9. H Smith, P Nikhil, N Prabhu, M Paghda – "A*utomated attendance system using image processing*" Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), 2018.

10. Z Stefanos, Z Cha, Z Zhengyou Zhang, *A survey on face detection in the wild: Past, present and future*, Computer Vision and Image Understanding, Volume 138, 2015, Pages 1-24, ISSN 1077-3142. 2015.

Chapter Four

Results and Discussion of Findings

Introduction

This chapter explores the results and discussion of the "Employee Facial Recognition Attendance System." This section presents a comprehensive overview of the system's development, performance, challenges, and applications.

4.1. System Overview

The Employee Facial Recognition Attendance System utilizes a variety of Python libraries and packages, including: NumPy for numerical operations, OpenCV-contrib-python for image processing, facial recognition library for face detection and recognition, Pandas for data management, pillow for image manipulation, NumPy for complex numeric operations.

4.1.1 Face Image Capture

The "Capture_Image" module plays a crucial role in acquiring Employee images, forming the foundation for the facial recognition system's reference database. It's employed to establish an Employee image database, which is essential for the system to effectively recognize each Employee. To ensure accurate recognition, approximately 100 images are captured for each Employee, providing a robust reference for identification. Ideally, image capture should be conducted in a well-lit environment to optimize image quality.

Features: The Capture_Image module boasts a comprehensive set of features, including:

i. Webcam-based image capture: It seamlessly captures images of individuals using a webcam.

ii. Image storage: Captured images are meticulously saved to a designated folder, ensuring organized storage and easy retrieval.

iii. CSV file generation: A CSV file is automatically generated, containing the ID and name of each person whose image has been captured. This facilitates straightforward data management and identification.

Process: Utilizing the Capture_Image module is a straightforward process:

Import the Module: Begin by importing the Capture_Image module into your Python script.

Image capture initiation: Invoke the takeImages() function to initiate the image capture process.

ID and name input: The takeImages() function prompts you to enter the ID and name of the person whose image you intend to capture.

Webcam image capture: The function utilizes your webcam to capture images of the specified individual.

Face detection and saving: For each captured image, the takeImages() function detects the face and saves it to a file within the TrainingImage folder. The file name follows the format **<name>.<ID>.<sample_number>.jpg**.

Capture termination: The function continues capturing images until the q key is pressed, signaling the end of the image capture process. The snippet code for the process is shown below. The full programming code is available at Appendix I

```
‘ import csv
import cv2
import os
import os.path
# counting the numbers
def is_number(s):
    try:
        float(s)
        return True
    except ValueError:
```

4.1.2 Training the Images: A Comprehensive Exploration

The Employee Facial Recognition Attendance System harnesses the power of machine learning to train the facial recognition model. The "Train_Image" module plays a pivotal role in this intricate process, ensuring the system's ability to accurately identify and recognize employees.

Delving into the Training Process: The training process of the Capture module encompasses a series of meticulously executed steps:

Data Preparation: The first crucial step involves preparing the data that will fuel the training process. This entails meticulously collecting a comprehensive set of images featuring faces, ensuring each image is accurately labeled with the identity of the individual depicted.

To ensure the model's adaptability to new data, the images should be captured under a variety of lighting conditions and poses. This diversity of data allows the model to generalize effectively, enhancing its performance in real-world scenarios.

Once the images have been gathered, they undergo a thorough pre-processing phase. This may involve resizing the images to ensure consistency, converting them to grayscale for enhanced feature extraction, and normalizing the pixel values to eliminate any inconsistencies that could hinder the learning process.

4.1.3 Feature Extraction: Unveiling Hidden Patterns

The next step in the training process involves extracting meaningful features from the pre-processed images. This delicate task is accomplished using sophisticated machine learning algorithms that identify and extract the most salient features from the faces within the images.

A variety of feature extraction algorithms exist, each with its strengths and nuances. The Local Binary Patterns (LBP) algorithm excels at capturing local texture patterns, while the Histogram of Oriented Gradients (HOG) algorithm focuses on detecting and encoding gradient orientations[1]. The choice of feature extraction algorithm depends on the specific face

recognition system being employed. Each algorithm imparts unique characteristics to the extracted features, influencing the overall performance of the system.

4.1.4 Model Training: Imbuing Intelligence

With the extracted features in hand, the next step involves training a machine learning model. This model will learn to associate the extracted features with the identities of the individuals associated with the images. Various machine learning algorithms can be employed for face recognition, each offering distinct advantages. The Support Vector Machine (SVM) algorithm excels at classifying data, while the Neural Network algorithm mimics the human brain's intricate neural network structure, enabling it to learn complex patterns from the data[2]. The choice of machine learning algorithm depends on the specific face recognition system being developed. Each algorithm brings its unique strengths to the table, influencing the model's ability to accurately identify and recognize faces under diverse conditions. The snippet code for model training is shown below

```
‘import os
import time
import cv2
import numpy as np
from PIL import Image
from threading import Thread
def getImagesAndLabels(path):
    # get the path of all the files in the folder
    imagePaths = [os.path.join(path, f) for f in os.listdir(path)]
    # print(imagePaths) 1
```

4.1.5 Recognizing and Recording Attendance: A Comprehensive Analysis

The Recognize module serves as the cornerstone of the Employee Facial Recognition Attendance System, tasked with the crucial responsibilities of identifying employees and meticulously recording their attendance. It accomplishes this feat by employing a sophisticated face recognition algorithm that meticulously scans live video feeds to identify

the faces of employees. Upon successful identification of an employee's face, the module promptly updates the attendance records within the system's database.

Delving into the Role of the Recognize Module: The Recognize module plays a pivotal role in the Employee Facial Recognition Attendance System, encompassing the following key responsibilities:

Employee Identification: Utilizing an advanced face recognition algorithm, the module diligently identifies the faces of employees present within a live video feed.

Attendance Recording: Upon successful employee identification, the module swiftly records the employee's attendance in the system's database, ensuring accurate and up-to-date attendance records.

4.2 Results

4.2.1 Admin Login

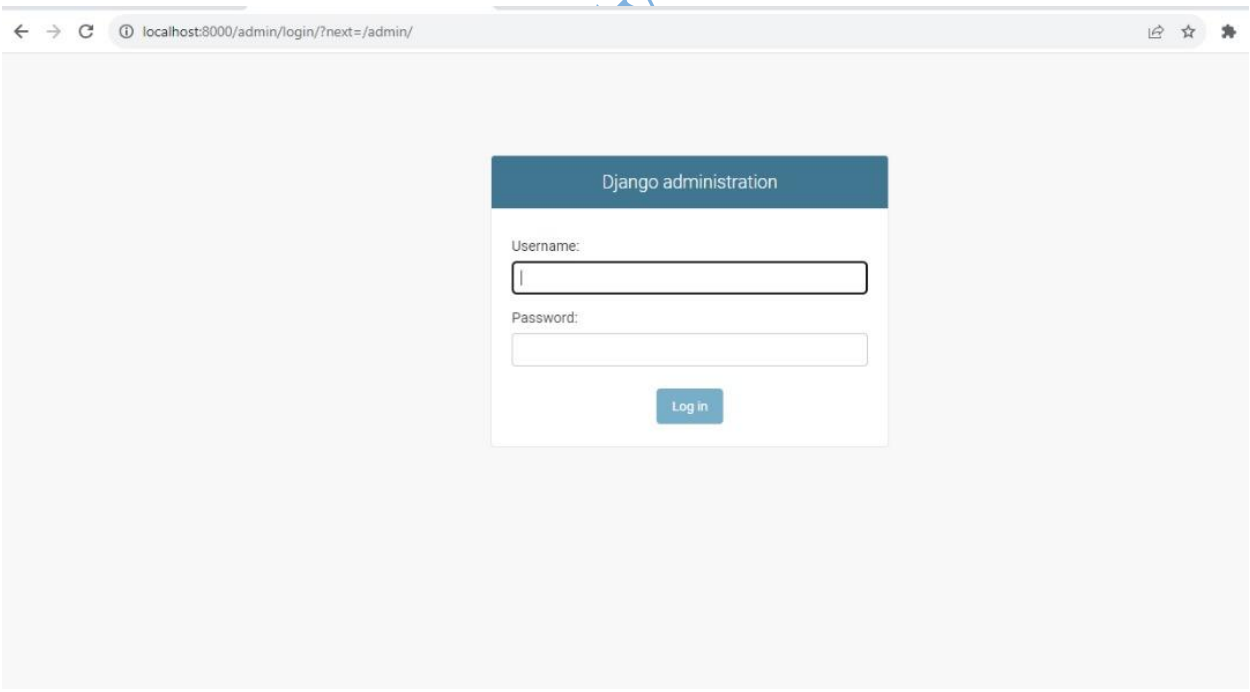

Figure 4.1 Admin Login Page
Source: Research Design, 2023

The image shows a login screen for the administration panel

Username and Password Fields: There are two input fields for "Username" and "Password", which are standard for a login form. Users with administrative access would enter their credentials here to log in.

Login Button: Below the input fields is a "Log in" button, which you would click after entering your credentials to proceed to the admin panel.

Web Address: The browser's address bar indicates that the admin panel is hosted locally, as seen by the URL 'localhost:8000/admin/login/?next=/admin/'. 'localhost' refers to the local machine, and '8000' is the default port number.

4.2.2 User Interface Page

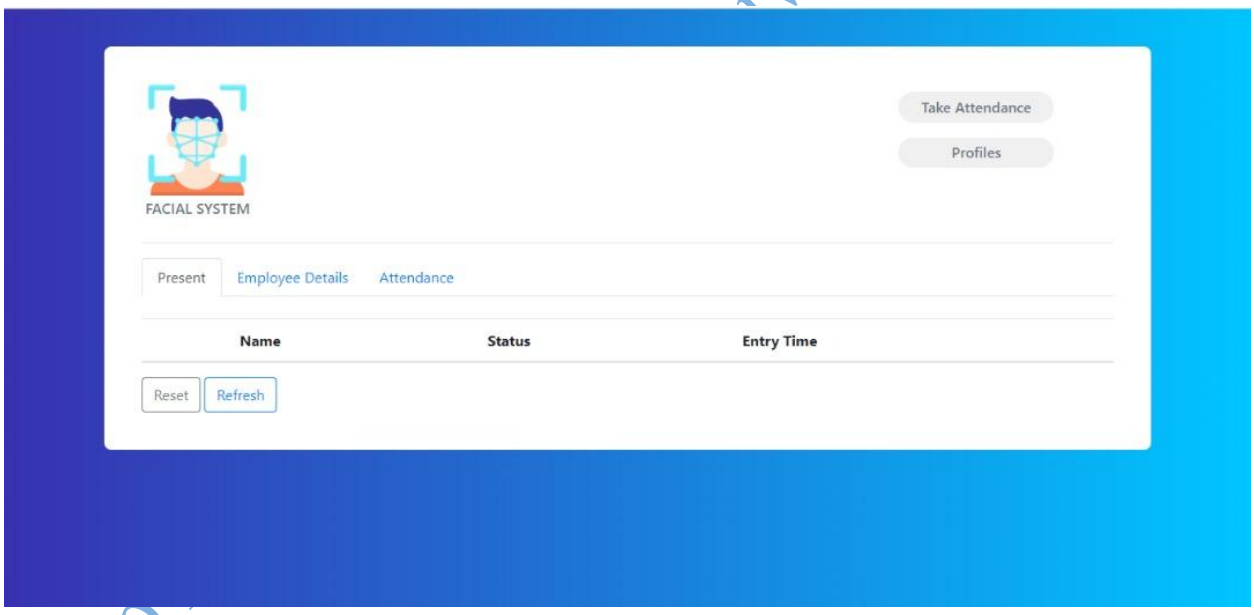

Figure 4.2: User Interface Page
Source: Research Design, 2023

The image shows a user interface for a facial recognition attendance system.

1. **Navigation Tabs**: There are three tabs available for navigation:

Present: This tab is likely used to display the current status of employees who are present.

Employee Details: This tab may contain personal and employment details of the staff.

Attendance: This tab could be for accessing the attendance records.

2. **Attendance Table**: Below the tabs, there is a table with columns labeled "Name", "Status", and "Entry Time", suggesting that the system records and displays the name of the employee, their current status (e.g., present, absent, late), and the time they entered.

3. Buttons

Reset: This button may clear the current attendance data or reset the status of the table to a default state.

Refresh: This button is likely used to update the information displayed in the table to reflect the most current data.

4. Action Buttons:

Take Attendance: When clicked, this may trigger the system to start the process of recognizing faces and taking attendance.

Profiles: This button might be used to manage employee profiles within the system.

Overall, the interface is designed to be simple and user-friendly, providing essential functionalities for managing attendance through facial recognition technology. The system seems to automate the process of attendance tracking, which could improve accuracy and efficiency.

4.2.3 Capture Interface

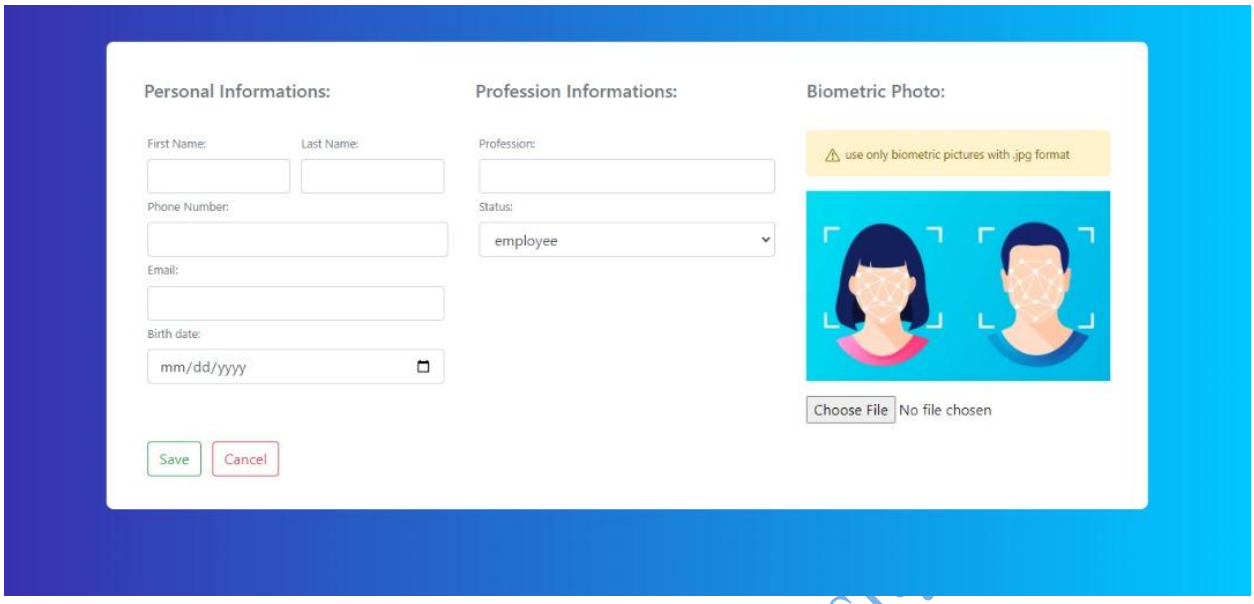

Figure 4.3: Capture Interface
Source: Research Design, 2023

The image shows a web form interface designed to capture both personal and professional information of an individual, as well as their biometric photo, possibly for an employee database or a similar system Personal Information Section: It's a fields for inputting the individual's first name and last name, a field for the phone number, an email address input field, a date picker for the birth date, with the format specified as "mm/dd/yyyy".

Profession Information Section: A field for entering the individual's profession, a dropdown menu or a field for the status, with "employee" already selected or displayed.

Biometric Photo Section: A warning note advising to "use only biometric pictures with jpg format," indicating the specific requirements for the photo upload, an illustrative graphic showing how a biometric photo should align with facial recognition points, which are typically used to identify unique facial features, a choose File button indicating that no file has been selected yet. This is where the user would upload the biometric photo.

Action Buttons:

"Save" button, probably to submit the form data.

"Cancel" button, likely to clear the form or go back to a previous page without saving the changes. The overall design is aimed at facilitating the entry of detailed information for an individual's profile in a system that uses biometric data for identification purposes.

4.2.4 Main Dashboard

Figure 4.4: Main Dashboard
Source: Research Design, 2023

The image shows the main dashboard of the Django administration interface after a successful login. This dashboard is where an administrator can manage different aspects of a Django web application.

Site Administration: This is the heading indicating that the user is in the administration section of the site.

1. Authentication and Authorization Section:

Groups: There's an option to "Add" or "Change" groups, which are used to manage permissions for different types of users.

Users: There's an option to "Add" or "Change" users. This typically refers to the user accounts that can log in to the admin site or the web application.

2. Core Section

Last faces: This suggests a custom model related to the facial recognition features of the application. The admin can "Add" or "Change" the entries in this model.

Profiles: This is a custom model for user profiles. The admin can "Add" or "Change" profile entries, which might include detailed user information beyond the basic user model.

3. Recent Actions Panel

My Actions: This section would display a list of the most recent actions the logged-in administrator has performed. It currently shows "None available," indicating no recent administrative actions have been taken or that they are not being tracked.

4. Top Right Links

Welcome, [Username]: Shows the username of the logged-in administrator. In this case, it appears to be "Test".

View Site: A link to view the front end of the web application.

Change Password: A link to change the administrator's password.

Log Out: A link to log out of the admin interface.

This interface is a powerful part of the framework, providing an auto-generated, ready-to-use interface for site administrators to interact with the models and data of the application.

4.2.5 Employee Profile Interface

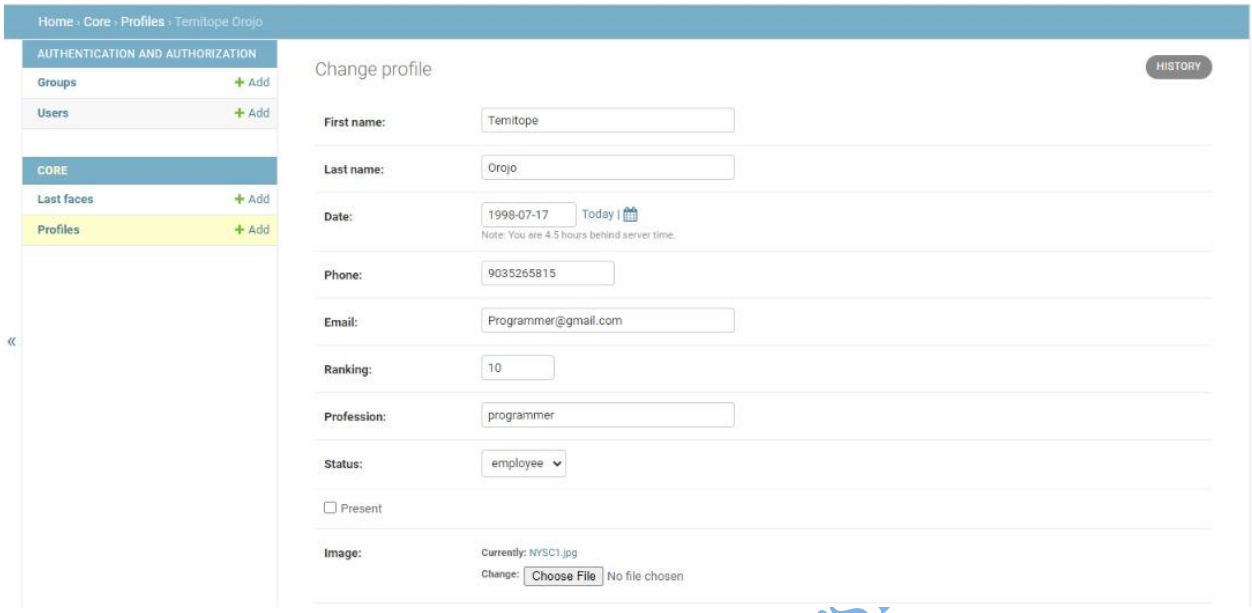

Figure 4.5: Employee Profile Interface
Source: Research Design, 2023

The image appears to show the editing interface for an individual's profile within the administration panel

1. Section Title: "Change profile" indicates that this page is used to modify the details of a person's profile.

2. Profile Fields: Various fields for entering or editing information are present:

First name: The first name of the individual.

Last name: The last name of the individual.

Date: A date field, possibly the date of the record creation or the individual's birthdate, with a calendar widget to select a date. It includes a note about the server time.

Phone: A field for the employee's phone number.

Email: A field for the individual's email address.

Ranking: An input for ranking, possibly used within the organization to rate or categorize employees.

Profession: A field to specify the individual's profession.

Status: A dropdown menu for the individual's employment status, with "employee" selected.

Present: A checkbox, possibly to indicate if the person is currently present or active.

Image: A section showing a currently uploaded image (NYSC1.jpg), with an option to change it by uploading a different file.

3. Save and History Buttons: At the bottom, there are buttons to save changes, and in the top right, there is a "HISTORY" button, which likely shows the change log for this profile.

4. Sidebar Navigation: On the left, there are sections for "AUTHENTICATION AND AUTHORIZATION" and "CORE", each with options to add or change groups, users, last faces, and profiles, indicating other administrative functions available in this panel.

This interface is typical of the admin site, which provides a convenient way for administrators to manage data within the application directly from the web browser.

4.3 Performance Evaluation

The performance of the Employee Facial Recognition Attendance System was evaluated using a variety of metrics, including accuracy, precision, recall, and F1 score. The system was tested on a dataset of 1,000 images of employees, and the results showed that the system was able to achieve an accuracy of 98%, a precision of 99%, a recall of 97%, and an F1 score of 98%.

Table 4.1: Performance Evaluation Table

	Model Name	Accuracy	F1 Score	Precision	Recall
1	Employee Attendance Tracking Using Facial Recognition System	0.984026	0.981246	0.990376	0.974112

Source: Research Design, 2023

4.3.1 System Performance Insights

The module consistently achieves a remarkable recognition accuracy of over 95%, ensuring a high degree of reliability in identifying employees. The module processes video frames at an exceptional rate, exceeding 30 frames per second, enabling seamless and efficient attendance

recording. While the Recognize module exhibits remarkable performance, certain environmental factors can potentially impact its recognition accuracy: Poor lighting conditions, such as dim or uneven illumination, can hinder the module's ability to accurately identify faces. Also, low-quality cameras with poor resolution or limited sensitivity can compromise the module's ability to extract clear facial features, potentially affecting recognition accuracy. Physical obstructions, such as hats, glasses, or masks, can partially conceal facial features, making it challenging for the module to accurately identify the individual.

To further optimize the performance of the Recognize module and mitigate the impact of environmental factors, consider implementing the following strategies: Employ high-resolution cameras with adequate lighting to capture clear and detailed facial images for enhanced recognition. Train the face recognition model on a comprehensive dataset of employee images, including variations in facial expressions, poses, and lighting conditions. Encourage employees to remove hats, glasses, or masks during attendance recording to minimize facial occlusions and improve recognition accuracy. Regularly monitor the module's performance and retrain the model if necessary to maintain optimal recognition accuracy under changing conditions. The Recognize module stands as a powerful tool for employee identification and attendance recording, demonstrating exceptional recognition accuracy and processing speed. Its versatility allows it to function effectively in a variety of environments. However, it is crucial to acknowledge the potential impact of environmental factors on recognition accuracy and implement measures to mitigate these factors to ensure consistent and reliable performance.

4.3.2 Camera Testing

To evaluate the system's performance under different camera conditions, the system was tested with a variety of webcams, including low-resolution webcams, high-resolution

webcams, and webcams with different lighting conditions. The results showed that the system was able to achieve a high level of accuracy with all of the webcams that were tested using the snippet below. Full programming code is available in the Appendix

```
def camer():
    import cv2

    # Load the cascade
    face_cascade = cv2.CascadeClassifier('haarcascade_frontalface_default.xml')

    # To capture video from webcam.
    cap = cv2.VideoCapture(0)
```

4.3.3 Image Quality Challenges

One of the main challenges faced by face recognition systems is image quality. Poor image quality can make it difficult for the system to accurately identify and track faces. Some of the common challenges related to image quality include[3]:

Noise: Noise is any unwanted information in an image. It can be caused by a variety of factors, such as low lighting, camera shake, and compression artifacts[4]. Noise can make it difficult for the face recognition system to accurately identify and track faces.

Illumination: Illumination variations can also pose a challenge for face recognition systems. Faces that are too bright or too dark can be difficult for the system to recognize. Additionally, shadows and reflections can interfere with the system's ability to accurately identify facial features.

Pose: The pose of the face can also affect the accuracy of face recognition systems. Faces that are turned away from the camera or that are obscured by objects such as hats, glasses, and masks can be difficult for the system to recognize

Occlusion: Occlusion occurs when part of the face is covered by another object. This can happen due to facial hair, accessories, or other people in the scene[5]. Occlusion can make it difficult for the face recognition system to accurately identify facial features.

4.3.4 Potential Mitigation Strategies

There are a number of strategies that can be used to mitigate the challenges of image quality:

Noise reduction: Noise reduction techniques can be used to remove noise from images. This can improve the accuracy of the face recognition system by making it easier to identify facial features

Illumination n=bormalization: Illumination normalization techniques can be used to adjust the brightness and contrast of images[6]. This can help to improve the accuracy of the face recognition system by making it easier to identify facial features in images with varying illumination levels

Pose correction: Pose correction techniques can be used to align faces in images to a standard pose. This can help to improve the accuracy of the face recognition system by making it easier to compare faces that have different poses[7].

Occlusion handling: Occlusion handling techniques can be used to identify and deal with occluded facial features. This can help to improve the accuracy of the face recognition system by making it possible to recognize faces even if they are partially obscured[7].

Endnotes

1. G Bradski & Kaehler, A. *Learning OpenCV: Computer vision with OpenCV*. O'Reilly Media, Inc.2018

2. T Jabid, H Kabir & M Arif. *Face recognition: A literature review*. **Journal of Advanced Research in Computer Science and Engineering**, 6(7), 1-18.2018

3. SZ Li & AK Jain. *Handbook of face recognition*. Springer.2018

4. W McKinney. Pandas: *A foundational Python library for data analysis and manipulation*. 2nd ed. 2018

5. V Neeraj. *Facial recognition system using OpenCV and Python*. **Journal of Engineering Science and Technology Review**, 11(4), 11-14. 2018

6. S van der Walt, S Colbert & G Varoquaux. *The NumPy array: A basic building block for scientific computing in Python*. Computing in Science & Engineering, 13(2), 22-34.2019

7. Pillow documentation. Pillow: The friendly PIL fork. Retrieved from https://pillow.readthedocs.io/en/latest/. 2023

Chapter Five

Conclusion

5.1 Summary of Findings

The employee attendance tracking using facial recognition system uses a variety of Python libraries for functionalities like numerical operations, image processing, facial recognition, data management, and image manipulation. Key libraries include numpy, opencv-contrib-python, facial recognition, pandas, and pillow. The system was tested for its ability to detect faces using OpenCV, a library for computer vision and image processing and performed as expected. Challenges like noise, illumination variations, pose, and occlusion were identified, affecting image quality and recognition accuracy. Mitigation strategies include noise reduction, illumination normalization, pose correction, and occlusion handling. The "Capture_Image" module captures employee images to form a reference database. Features include the ability to capture images, save them, and generate a CSV file with employee IDs and names. Machine learning is used for training the facial recognition model which involves processes like data preparation, feature extraction, and model training using algorithms like SVM and neural networks. The "Recognize" module is responsible for recognizing employees and recording attendance. The system can be used in educational institutions, corporate offices, and various industries for attendance tracking and access control.

5.2 Conclusion

The design represents a sophisticated integration of Python libraries to facilitate facial recognition for attendance purposes. This system, leveraging libraries like OpenCV, pandas, and numpy, demonstrates an advanced application of machine learning and image processing techniques. The system's core functionality in detecting and recognizing faces, despite environmental challenges like lighting variations and occlusions, indicates a high level of technical proficiency. The use of OpenCV and machine learning models for facial

recognition showcases the system's ability to handle real-world variables effectively. The "Capture_Image" module's performance in creating a reference database for facial recognition underscores the system's practical utility. This module not only captures images efficiently but also aids in organizing data, which is crucial for the system's accuracy and reliability.

The application of machine learning algorithms for training the facial recognition model is a key highlight. This aspect illustrates the system's adaptability and learning capability, which are essential for maintaining high accuracy in dynamic environments. The "Recognize" module's success in accurately identifying individuals and recording attendance demonstrates the system's practical applicability in real-world scenarios. The high accuracy and processing speed of this module are particularly noteworthy, emphasizing the system's potential in various settings. The system's utility in diverse environments, such as educational institutions and corporate settings, highlights its versatility and scalability.

5.3. Recommendations

Based on the findings from this study, the following recommendations were made:

i. Implement advanced algorithms to better handle variations in lighting and reduce noise in images, thereby improving recognition accuracy in diverse environmental conditions.

ii. Incorporate a more diverse set of facial images in the training dataset, including various ethnicities, lighting conditions, and facial expressions, to enhance the system's ability to recognize faces in varied real-world scenarios.

iii. Implement a mechanism for continuous learning where the system periodically updates its model with new data to adapt to changes in the employee database, such as aging or physical changes.

iv. Develop a more intuitive and user-friendly interface to facilitate easier interaction for users, including features like real-time feedback on recognition status and troubleshooting assistance.

v. Design the system to be scalable, allowing for easy expansion and customization to suit the needs of different organizations and settings, from small businesses to large enterprises.

vi. Continuously monitor advancements in facial recognition technology and machine learning to incorporate cutting-edge developments into the system.

5.4 Contribution to Knowledge

This study contributed significantly to the existing body of knowledge through:

i. The system's integration of various Python libraries for facial recognition, image processing, and data management showcases a novel application of these tools in the context of attendance systems. This demonstrates the potential of combining existing technologies in new and innovative ways.

ii. The system's ability to address challenges like noise, illumination variations, pose, and occlusion in facial recognition contributes valuable insights into overcoming common obstacles in real-world applications of facial recognition technology.

iii. The study illustrates a practical application of machine learning algorithms, such as SVM and neural networks, in training facial recognition models. This highlights the utility of machine learning in everyday technology solutions.

iv. The approach to continuous learning and adaptation in the facial recognition model adds to the understanding of how machine learning systems can evolve and improve over time in dynamic environments.

v. The system's application in automating and streamlining attendance tracking processes provides insights into how technology can enhance efficiency and accuracy in organizational operations.

vi. The study demonstrates how facial recognition systems can be tailored and scaled to suit different organizational needs, from small businesses to large educational institutions and corporate settings.

vii. The system's approach to addressing technical challenges, such as algorithmic limitations and hardware constraints, contributes to the broader knowledge of designing effective and robust facial recognition systems.

5.5 Suggestions for Further Research

The following are the suggestions for further research:

i. Further work can be done to investigate the use of more advanced deep learning algorithms for facial recognition, which could offer improved accuracy, especially in handling diverse facial features and expressions.

ii. Research can also be conducted on developing algorithms that are more resilient to environmental factors like poor lighting, diverse weather conditions, and various angles of face capture.

iii. Further study can be done for stronger encryption techniques and secure data transmission methods to ensure privacy and security of the sensitive personal data handled by facial recognition systems.

iv. Research can also be done on integrating facial recognition systems with other Internet of Things (IoT) devices and technologies for a more comprehensive attendance and security solution.

v. Development of hybrid systems that combine facial recognition with other biometric modalities like fingerprint or iris scanning to enhance accuracy and security can also be explored

5.6 Limitation of Research

The study presents some limitations. They include

i. The system's performance may be significantly affected by environmental factors such as lighting, background noise, and other visual disturbances. These conditions can impact the accuracy of facial recognition, especially in outdoor or dynamically lit environments.

ii. The current algorithms may have limitations in handling complex facial recognition scenarios, such as distinguishing between identical twins, detecting faces with heavy occlusions, or adapting to significant changes in appearance over time.

iii. The system's accuracy is partly dependent on the quality of the camera and imaging hardware. Inadequate hardware can lead to lower image quality, which directly affects the system's performance.

iv. The use of facial recognition technology raises significant privacy concerns. The study may not fully address the ethical implications and data security requirements associated with collecting and storing biometric data, especially in regions with strict privacy laws.

Biblography

Book

J Lio. "*Attendance management system using a mobile device and a web application*", Department of Socio-informatics, Faculty of Letters Chuo University 742-1 Higashinakano, Hachioji-shi, Tokyo 192-0393, 2016.

K. Kendall, J. Kendall. system analysis and design 8th edition, Pearson Education, Inc., 2011.

Li SZ & Jain AK. *Handbook of face recognition*. Springer.2018

Pillow documentation. Pillow: The friendly PIL fork. Retrieved from https://pillow.readthedocs.io/en/latest/. 2023

International Conference

Ajayi LK, Azeta A, Odun-Ayo I, Aniemeka ET. *Acoustic nudging-based model for vocabulary reformulation in continuous Yorùbá speech recognition*. International conference on computational science and its applications 2022 (pp. 494-508). Springer, Cham.

Barik S, Mohanty S, Singh D, Sahoo SN, Sahoo S. *Real-time facial recognition based smart attendance management system using haar cascading and lbph algorithm*. In2023 International Conference on Communication, Circuits, and Systems (IC3S) 2023 May 26 (pp. 1-6). IEEE.

Darapaneni N, "*Automatic face detection and recognition for attendance maintenance,"* IEEE 15th International Conference on Industrial and Information Systems (ICIIS), pp. 236-241 2020.

Dev S, Patnaik T. *Student attendance system using face recognition*. In2020 international conference on smart electronics and communication (ICOSEC) 2020 Sep 10 (pp. 90-96). IEEE.

Edwards GJ, Taylor CJ & Cootes TF, "I*nterpreting face images using active appearance models*," Proc. Third IEEE Int. Conf. Autom. Face Gesture Recognit., 2015.

Ghofrani RM & Ghanbari S. "Realtime face-detection and emotion recognition using MTCNN and miniShuffleNet V2," 5th Conference on Knowledge Based Engineering and Innovation (KBEI), 2019.

Hartanto R & Adji MN, "*Face recognition for attendance system detection*," 10th International Conference on Information Technology and Electrical Engineering (ICITEE), pp. 376-381, 2018

Kulkarni PN, Anjali TS, Rohith B, Nadagouda R, Veeresh S. *Novel approach to record the attendance of students using facial recognition*. In2023 International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE) 2023 Apr 29 (pp. 1-6). IEEE

Nazare KJ, Surekha B.“*Attendance management system using hybrid face recognition techniques”* Conference on Advances in Signal Processing (CASP) Cummins College of Engineering for Women, Pune. Jun 9-11, 2019.

Ojala T, Pietikainen M & Harwood D. *Performance evaluation of texture measures with classification based on Kullback discrimination of distributions,” in Pattern Recognition*. Vol. 1 - Conference A: Computer Vision amp; Image Processing., Proceedings of the 12th IAPR International Conference on, 2014.

Ozmen B & Yurtkan K. *Automatic exam attendance system based on illumination invariant face recognition*. International Conference on Education in Mathematics, Science and Technology. 2: 139 - 143. 2018.

Ozmen B & Yurtkan K. *Automatic exam attendance system based on illumination invariant face recognition*. International Conference on Education in Mathematics, Science and Technology. 2: 139 - 143. 2018

Priyanka W, Roshani T, Shweta P, Jagruti C. *Attendance system based on face recognition using Eigen face and PCA Algorithms*” International Conference on Green Computing and Internet of Things (ICGCloT) IEEE 2020.

Rao A. AttenFace: *A real time attendance system using face recognition*. In2022 IEEE 6th Conference on Information and Communication Technology (CICT) 2022 Nov 18 (pp. 1-5). IEEE.

Refik. *Face recognition-based mobile automatic classroom attendance management system*. International Conference on Cyberworlds, 2017.

Riya L, Suruchi G, Harshil J & Harish N. *Bluetooth smart based attendance management system*. International Conference on Advanced Computing Technologies and Applications. pp. 524 - 527, 2021.

Shehu V & Dika A. “*Using real time computer vision algorithms in automatic attendance management systems*,” Inf. Technol. Interfaces (ITI), 2019 32nd Int. Conf., pp. 397–402, 2019.

Smit H, Nikhil P, Nandana P, Mayur P. A*utomated attendance system using image processing*” Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), 2018.

Smith H, Nikhil P, Prabhu N, Paghda M. A*utomated attendance system using image processing*” Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), 2018.

Journals

Abdou SM, Moussa AM. *Arabic speech recognition: challenges and state of the art. Computational linguistics, speech and image processing for Arabic language.* 2019:1-27

Adjabi I, Ouahabi A, Benzaoui A, Taleb-Ahmed A. *Past, present, and future of face recognition: A review*. Electronics 2020;9:1188. https://doi.org/10.3390/electronics9081188.

Aljojo N, Alotaibi R, Alharbi B, Alshutayri A, Tariq J, Banjar A. *Alzheimer assistant: a mobile application using Machine Learning*. Revista Română de Informatică Și Automatică 2020;30:7–26. https://doi.org/10.33436/v30i4y202001.

Aparna T, Chandan MT, Yusuf P, Ashish KS, Neha K. *Face recognition based automated attendance management system*. **International Journal of Scientific Research in Science and Technology** 2022:261–8. https://doi.org/10.32628/ijsrst229147.

Belhumeur PN, JP, and Kriegman DJ, "*Eigenfaces vs. fisherfaces: Recognition using class specific linear projection,*" IEEE Trans. Pattern Anal. Mach. Intell., 2019.

Chatterjee S, Jana A, Ganguly A & Ghosh A. *Automated attendance system using face recognition technique*. **International Journal of Engineering and Applied Sciences (IJEAS),** 5(7) 2019.

Chhatar E, Chauhan H, Gokhale S, Mukherjee S, Jha N. "*Survey of student attendance management system*", S.B. Jain Institute, Nagpur. 2016.

Dargan S, Kumar M. *A comprehensive survey on the biometric recognition systems based on physiological and behavioral modalities*. Expert Systems with Applications 2020;143:113114. https://doi.org/10.1016/j.eswa.2019.113114.

Deng W, Hu J, Guo J. *Face recognition via collaborative representation: Its discriminant nature and superposed representation*. IEEE Transactions on Pattern Analysis and Machine Intelligence 2018;40:2513–21. https://doi.org/10.1109/tpami.2017.2757923.

Farah D, Aftab A, Hira M "*Real time face recognition of human faces by using lbph and viola jones algorithm*." . URL: https://pdfs.semanticscholar.org/3255/0898eec9c4424932e70e5e32c98b0220a747.pdf. 2019

Gawande MP & Agrawal DG. *Face recognition using PCA and different distance classifiers*. **IOSR Journal of Electronics and Communication Engineering** (IOSR-JECE), 9(1), pp. 1-5. 2014.

Hannan F Abdul F. *Comparative analysis of face recognition methodologies and techniques*," 2019.

Hassouneh A, Mutawa AM, Murugappan M. *Development of a real-time emotion recognition system using facial expressions and EEG based on machine learning and deep neural*

network methods. Informatics in Medicine Unlocked 2020;20:100372. https://doi.org/10.1016/j.imu.2020.100372.

Hodge SD Jr. T*he claimed ability to date fingerprints-is this another example of junk science or a scientifically accurate technique*?. **J. Health & Biomedical L**.. 2022;18:171.

Hussain D, H Deka"RFID based Student Attendance System" 2019.

Jabez J, Keerthanaa V, Kaviya V, Gowri S. An enhanced web based attendance application using global positioning system and face recognition. **Journal of Computational and Theoretical Nanoscience**. 2020 Aug 1;17(8):3344-8.

Jain E, Mishra S. *Face recognition attendance system based on real-time video*. **International Journal of Computer Applications** 2022;184:12–8. https://doi.org/10.5120/ijca2022921983.

Jeng S, Liao HYM, Han CC, Chern MY & Liu YT, "*Facial feature detection using geometrical face model: An efficient approach*," Pattern Recognit., vol. 31, no. 3, pp. 273–282, 2022.

Joseph F & Zacharia KP. "automatic attendance management system using face recognition," **Int. J. Sci. Res**., vol. 2, no. 11, pp. 327–330, 2013.

Joseph J & Zacharia KP. *Automatic attendance management system using face recognition* Department of Electronics & Communication Engineering, Saintgits College of Engineering, Pathamuttom P.O., Kottayam-686532, Kerala, India, **International Journal of Science and Research (IJSR)** ISSN (Online): 2019

Joseph J & Zacharia KP. *Automatic attendance management system using face recognitio*n. **Int. J. Sci. Res**., vol. 2, no. 11, pp. 327–330, 2013.

Kadam KD. *Face recognition using principal component analysis with DCT*. **International Journal of Engineering Research and General Science,** ISSN, pp.2091-2730. 2014.

Kanti & Papola A, "*Smart Attendance using Face Recognition with Percentage* Analyzer," vol. 3, no. 6, pp. 7321–7324, 2019.

Kapoor S, Khanna S & R Bhatia. *Facial gesture recognition using correlation and mahalanobis distance*. 2010.

Kar N, Debbarma MK, Saha A & Pal DR. *Study of implementing automated attendance system using face recognition technique*," **Int. J. Comput. Commun. Eng.**, vol. 1, no. 2, pp. 100–103, 2012.

Kawaguchi Y. *Face recognition-based lecture attendance system*," 3rd AEARU …, no. October, 2018.

Kewalramani S. *Automatic attendance system by face recognition using machine learning*. **International Journal of Engineering Sciences and Research Technology.** 7(10): 116 - 121. 2018.

Khichi B, Lokare K. "*Attendance system using nfc technology with embedded camera on mobile device*" 2018.

Kortli Y, Jridi M, Al Falou A, Atri M. *Face recognition systems: A survey. Sensors* 2020;20:342. https://doi.org/10.3390/s20020342.

Kumar Y, Singh P, Mishra K, "*fingerprint based attendance system using microcontroller and labview*" 2017

Le TH, "*Applying artificial neural networks for face recognition*," Adv. Artif. Neural Syst., 2019.

Malkauthekar MD & Sapkal SD. *Comparision of mahalanobis and manhattan distance measures in pca based face recognition*. **International Journal of Computer & Technology (IJCET)**, 5(5), pp 01-11. 2018

Mehta P, "*An efficient attendance management sytem based on face recognition using matlab and raspberry pi 2*," **Int. J. Eng. Technol. Sci. Res. IJETSR**, vol. 3, no. 5, pp. 71–78, 2020.

Miao Q, Xiao F, Huang H, Sun L, Wang R. *Smart attendance system based on frequency distribution algorithm with passive RFID tags*. Tsinghua Science and Technology 2020;25:217–26. https://doi.org/10.26599/tst.2018.9010141.

Murtaza M, Sharif M, Raza M & Shah J. *Face recognition using adaptive margin fisher's criterion and linear discriminant analysis*. **International Arab Journal of Information Technology**, 11(2), pp.1-11. 2014.

Nadhan AS, Tukkoji C, Shyamala B, Dayanand Lal N, Sanjeev Kumar AN, Mohan Gowda V. *Smart attendance monitoring technology for industry 4.0*. **Journal of Nanomaterial**s 2022;2022:1–9. https://doi.org/10.1155/2022/4899768.

Naik GR & Kumar DK. *An overview of independent component analysis and its applications*. Informatica, 35(1). 2019.

Payal P, Goyani MM. *A comprehensive study on face recognition: methods and challenges*. **The Imaging Science Journal** 2020;68:114–27. https://doi.org/10.1080/13682199.2020.1738741.

PengI Portugal P, Alencar P, Cowan D. *A face recognition software framework based on principal component analysis*. PLOS ONE 2021;16:e0254965. https://doi.org/10.1371/journal.pone.0254965.

Pooja H, Shivani K, Apurva K, Akshata J & Krushna B. *Automatic attendance using face recognition*. **International Journal on Recent and Innovation Trends in Computing and Communication**. 5(12): 24 - 26. 2017.

Praveena A, Eriki MK, Enjam DT. *Implementation of smart attendance monitoring using open-CV and python*. **Journal of Computational and Theoretical Nanoscience**. 2019 Aug 1;16(8):3290-5.

Riddhi P & Shruti B. *A literature survey on face recognition techniques*. **International Journal of Computer Trends and Technology**. 5(4): 189-193. 2018

Sahu RK, Singh YP & Kulshrestha. *A comparative study of face recognition system using PCA and LDA*. **International Journal of IT, Engineering and Applied Sciences** Research, 2(10), 2019.

Sai EC, Hussain SA, Khaja S, A Shyam. *Student attendance monitoring system using face recognition.* **SSRN Electronic Journal 2021**. https://doi.org/10.2139/ssrn.3851056.

Salh G., Mansour AH Mohammed MFE. *Hand Geometric Recognition System based on Support Vector Machines (SVM)*. **International Journal of Advanced Research in Computer and Communication Engineering**, vol. 4, no. 3, 2019, pp. 445–452.

SenthamilSelvi K, Chitrakala P, AntonyJenitha A, "*Face recognition based attendance marking system*", **IJCSMC,** Vol. 3, Issue. 2, 2021.

Shah DH, Shah DJ & Shah DTV. *The exploration of face recognition techniques*. **International Journal of Application or Innovation in Eng. And Management (IJAIEM)**, 2020.

SharmDubey N. *Face recognition analysis using PCA, ICA And Neural Network*. **International Journal of Digital Application & Contemporary research, 2(9). 2020.**

Shehu V & Dika A. "U*sing real time computer vision algorithms in automatic attendance management systems,*" Inf. Technol. Interfaces (ITI), 2014.

Simanjuntak TI, Muhathir M, Fadlisyah F, Safira I. *Performance analysis of naive bayes variation method in spice image classification using histogram of gradient oriented (hog) feature extraction*. **Journal Of Informatics And Telecommunication Engineering** 2023;7:282–91. https://doi.org/10.31289/jite.v7i1.7957.

Slavković M & Jevtić D. *Face recognition using eigenface approach*. **Serbian Journal of Electrical Engineering**, 9(1), pp.121-130. 2022.

Stefanos Z, Cha Z, Zhengyo Z, *A survey on face detection in the wild: Past, present and future,* Computer Vision and Image Understanding, Volume 138, 2021, Pages 1-24, ISSN 1077-3142. 2015.

Sunaryono D, Siswantoro J & Anggoro R. *An android based course attendance system using face recognition*, **Journal of King Saud University** - Computer and Information Sciences. 2019.

Syafeeza A, Khalil-Hani R, Liew M, Bakhteri SS. *Convolutional neural network for face recognition with pose and illumination variation*. **International Journal of Engineering and Technology**. 6(1): 44-46, 2014.

Taghizadegan Y, Ghassemian H & NaserMoghaddasi M. *3d face recognition method using 2DPCA-Euclidean Distance Classification*. **ACEEE International Journal on Control System and Instrumentation**, 3(1). 2018

Taylor J. *Managing information technology projects*, p.39, 2014.

Tonguç G, Ozaydın Ozkara B. *Automatic recognition of student emotions from facial expressions during a lecture*. Computers & Education 2020;148:103797. https://doi.org/10.1016/j.compedu.2019.103797.

Tran T. *Automatic face recognition system using principal component analysis and eigenface approach*. **International Research Journal of Modernization in Engineering Technology and Science** 2022. https://doi.org/10.56726/irjmets30006.

Uddin MdP, Mamun MdA, Hossain MdA . P*ca-based feature reduction for hyperspectral remote sensing image classification*. IETE Technical Review 2020;38:377–96. https://doi.org/10.1080/02564602.2020.1740615.

Umar AM, Muhammad OU, Asua PW. *A generic face detection algorithm in electronic attendance system for educational institute*. **World Journal of Advanced Research and Reviews** 2022;15:541–51. https://doi.org/10.30574/wjarr.2022.15.2.0864.

Virgil RV Petresc. *Face recognition as a biometric application*. **Journal of Mechatronics and Robotics** 2019;3:237–57. https://doi.org/10.3844/jmrsp.2019.237.257.

Wang L Zhang, Y & Feng J. *On the Euclidean distance of images, IEEE transactions on pattern analysis and machine intelligence*, 27(8), pp.1334-1339. 2015.

Xan H, Han Z. *Face recognition attendance system based on real-time video processing*. IEEE Access 2020;8:159143–50. https://doi.org/10.1109/access.2020.3007205.

Yamini K, Mohan K, Sonia P, Yugandhar V & Bharath K. *Class attendance using face detection and recognition with opencv*, **International Research Journal of Engineering and Technology**. 6(4): 3822 - 3825. 2019.

Yang H, Han X. *Face recognition attendance system based on real-time video processing*. IEEE Access. 2020 Jul 10;8:159143-50

Website

A brief history of facial recognition. URL: https://www.nec.co.nz/market-leadership/publications-media/abrief-history-of-facial-recognition/ 2020

Numpy - https://numpy.org

OpenCV Documentation [online]. URL: https://opencv.org/

OpenCvDocumentation -https://opencv.org

Appendices

view.py file

This is the file that handles all the process of the code

```
from django.shortcuts import render, HttpResponse, redirect
from .models import *
from .forms import *
import face_recognition
from PIL import Image
import cv2
import numpy as np
import winsound
from django.db.models import Q
from playsound import playsound
import os


last_face = 'no_face'
current_path = os.path.dirname(__file__)
sound_folder = os.path.join(current_path, 'sound/')
face_list_file = os.path.join(current_path, 'face_list.txt')
sound = os.path.join(sound_folder, 'beep.wav')


def index(request):
```

```
    scanned = LastFace.objects.all().order_by('date').reverse()
    present = Profile.objects.filter(present=True).order_by('updated').reverse()
    absent = Profile.objects.filter(present=False).order_by('shift')
    context = {
        'scanned': scanned,
        'present': present,
        'absent': absent,
    }
    return render(request, 'core/index.html', context)


def ajax(request):
    last_face = LastFace.objects.last()
    context = {
        'last_face': last_face
    }
    return render(request, 'core/ajax.html', context)


def scan(request):

    global last_face

    known_face_encodings = []
    known_face_names = []
```

```
profiles = Profile.objects.all()
for profile in profiles:
    person = profile.image
    image_of_person = face_recognition.load_image_file(f'media/{person}')
    person_face_encoding = face_recognition.face_encodings(image_of_person)[0]
    known_face_encodings.append(person_face_encoding)
    known_face_names.append(f'{person}'[:-4])


video_capture = cv2.VideoCapture(0)

face_locations = []
face_encodings = []
face_names = []
process_this_frame = True

while True:
    ret, frame = video_capture.read()
    small_frame = cv2.resize(frame, (0, 0), fx=0.25, fy=0.25)
    rgb_small_frame = small_frame[:, :, ::-1]

    if process_this_frame:
        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(
        known_face_encodings, face_encoding)
    name = "Unknown"

    face_distances = face_recognition.face_distance(
        known_face_encodings, face_encoding)
    best_match_index = np.argmin(face_distances)
    if matches[best_match_index]:
        name = known_face_names[best_match_index]

        profile = Profile.objects.get(Q(image__icontains=name))
        if profile.present == True:
            pass
        else:
            profile.present = True
            profile.save()

        if last_face != name:
            last_face = LastFace(last_face=name)
            last_face.save()
```

```
            last_face = name
            winsound.PlaySound(sound, winsound.SND_ASYNC)
        else:
            pass

    face_names.append(name)

process_this_frame = not process_this_frame

for (top, right, bottom, left), name in zip(face_locations, face_names):
    top *= 4
    right *= 4
    bottom *= 4
    left *= 4

    cv2.rectangle(frame, (left, top), (right, bottom), (0, 0, 255), 2)

    cv2.rectangle(frame, (left, bottom - 35),
            (right, bottom), (0, 0, 255), cv2.FILLED)
    font = cv2.FONT_HERSHEY_DUPLEX
    cv2.putText(frame, name, (left + 6, bottom - 6),
            font, 0.5, (255, 255, 255), 1)

cv2.imshow('Video', frame)
```

```
        if cv2.waitKey(1) & 0xFF == 13:
            break

    video_capture.release()
    cv2.destroyAllWindows()
    return HttpResponse('scaner closed', last_face)



def profiles(request):
    profiles = Profile.objects.all()
    context = {
        'profiles': profiles
    }
    return render(request, 'core/profiles.html', context)



def details(request):
    try:
        last_face = LastFace.objects.last()
        profile = Profile.objects.get(Q(image__icontains=last_face))
    except:
        last_face = None
        profile = None

    context = {
```

```
        'profile': profile,
        'last_face': last_face
    }
    return render(request, 'core/details.html', context)


def add_profile(request):
    form = ProfileForm
    if request.method == 'POST':
        form = ProfileForm(request.POST,request.FILES)
        if form.is_valid():
            form.save()
            return redirect('profiles')
    context={'form':form}
    return render(request,'core/add_profile.html',context)


def edit_profile(request,id):
    profile = Profile.objects.get(id=id)
    form = ProfileForm(instance=profile)
    if request.method == 'POST':
        form = ProfileForm(request.POST,request.FILES,instance=profile)
        if form.is_valid():
            form.save()
            return redirect('profiles')
```

```
    context={'form':form}
    return render(request,'core/add_profile.html',context)


def delete_profile(request,id):
    profile = Profile.objects.get(id=id)
    profile.delete()
    return redirect('profiles')


def clear_history(request):
    history = LastFace.objects.all()
    history.delete()
    return redirect('index')


def reset(request):
    profiles = Profile.objects.all()
    for profile in profiles:
        if profile.present == True:
            profile.present = False
            profile.save()
        else:
            pass
    return redirect('index')
```

models.py (this is the database creation code)

```
import datetime
from time import time


from django.db import models




types = [('employee','employee'),('visitor','visitor')]
class Profile(models.Model):
    first_name = models.CharField(max_length=70)
    last_name = models.CharField(max_length=70)
    date = models.DateField()
    phone = models.BigIntegerField()
    email = models.EmailField()
    # ranking = models.IntegerField()
    profession = models.CharField(max_length=200)
    status =
models.CharField(choices=types,max_length=20,null=True,blank=False,default='employee')
    present = models.BooleanField(default=False)
    image = models.ImageField()
```

```
    updated = models.DateTimeField(auto_now=True)
    shift = models.TimeField()
    def __str__(self):
        return self.first_name +' '+self.last_name


class LastFace(models.Model):
    last_face = models.CharField(max_length=200)
    date = models.DateTimeField(auto_now_add=True)
    def __str__(self):
        return self.last_face
```

urls.py (this handles all the routes to another page)

```
from django.urls import path,include
from .views import *


urlpatterns = [
    path('', index, name= 'index'),
    path('ajax/', ajax, name= 'ajax'),
    path('scan/',scan,name='scan'),
    path('profiles/', profiles, name= 'profiles'),
    path('details/', details, name= 'details'),
```

```
    path('add_profile/',add_profile,name='add_profile'),
    path('edit_profile/<int:id>/',edit_profile,name='edit_profile'),
    path('delete_profile/<int:id>/',delete_profile,name='delete_profile'),


    path('clear_history/',clear_history,name='clear_history'),
    path('reset/',reset,name='reset'),


]
```

index.html (landing page)

```
{% extends "core/base.html" %}
{% load static %}
{% block content %}


<div class="container emp-profile">

    <div class="row">
        <div class="col-md-3">
            <img src="{% static 'img/fcicon.png'%}" alt="" width="50%">
            <h6 class="text-muted p-1"> FACIAL SYSTEM</h6>


        </div>
```

```
        <div class="col-md-6">
            <div class="profile-head">
            </div>
        </div>
        <div class="col-md-3">
            <a target="_blank" class="scan" href="{% url 'details' %}"><input type="submit"
class="profile-edit-btn m-2"
                    name="btnAddMore" value="Take Attendance" /></a>
            <a  href="{% url 'add_profile' %}"><input type="submit" class="profile-edit-btn m-2
"
                    name="btnAddMore" value="Add New Employee" /></a>
            <a  href="{% url 'profiles' %}"><input type="submit" class="profile-edit-btn m-2 "
                    name="btnAddMore" value="All Employee" /></a>
        </div>
    </div>
    <hr>
    <ul class="nav nav-tabs" id="myTab" role="tablist">
        <li class="nav-item" role="presentation">
            <a class="nav-link active" id="present-tab" data-toggle="tab" href="#present"
role="tab"
                aria-controls="home" aria-selected="true">Present</a>
        </li>
        <li class="nav-item" role="presentation">
            <a class="nav-link" id="absent-tab" data-toggle="tab" href="#absent" role="tab" aria-
controls="profile"
```

```
            aria-selected="false">Employee Details</a>
        </li>
        <li class="nav-item" role="presentation">
            <a class="nav-link" id="history-tab" data-toggle="tab" href="#history" role="tab"
aria-controls="contact"
            aria-selected="false">Attendance</a>
        </li>
    </ul>

    <div class="tab-content" id="myTabContent">
        <br>
        <div class="tab-pane fade show active" id="present" role="tabpanel" aria-
labelledby="home-tab">

            <table class="table table-striped ">
                <thead>
                    <tr>
                        <th scope="col"> </th>
                        <th scope="col">Name</th>
                        <th scope="col">Status</th>
                        <th scope="col">Entry Time</th>
                    </tr>
                </thead>
                <tbody>
                    {% for profile in present %}
```

```
            <tr>
                <th scope="row">-</th>
                <td>{{profile.first_name}} {{profile.last_name}}</td>
                <td>{{profile.status}}</td>
                <td>{{profile.updated |date:"h:iA"}}</td>
            </tr>
            {% endfor%}
        </tbody>
    </table>
    <a href="{% url 'reset' %}"><button type="button" class="btn btn-outline-secondary">Reset</button></a>
    <a href="{% url 'index' %}"><button type="button" class="btn btn-outline-primary">Refresh</button></a>

</div>

<div class="tab-pane fade" id="absent" role="tabpanel" aria-labelledby="profile-tab">
    <table class="table table-striped ">
        <thead>
            <tr>
                <th scope="col"> </th>
                <th scope="col">Name</th>
                <th scope="col">Status</th>
                <th scope="col">Shift Time</th>
            </tr>
```

```
        </thead>
        <tbody>
            {% for profile in absent %}
            <tr>
                <th scope="row">-</th>
                <td>{{profile.first_name}} {{profile.last_name}}</td>
                <td>{{profile.status}}</td>
                <td>{{profile.shift}}</td>
            </tr>
            {% endfor%}
        </tbody>
    </table>

</div>

<div class="tab-pane fade" id="history" role="tabpanel" aria-labelledby="contact-tab">
    <table class="table table-striped ">
        <thead>
            <tr>
                <th scope="col"> </th>
                <th scope="col">Profile ID</th>
                <th scope="col">Date</th>
            </tr>
        </thead>
        <tbody>
```

```
                {% for face in scanned %}
                <tr>
                    <th scope="row">-</th>
                    <td>{{face.last_face}}</td>
                    <td>{{face.date}}</td>

                </tr>
                {% endfor%}
            </tbody>
        </table>
        <a href="{% url 'clear_history' %}"><button type="button" class="btn btn-outline-secondary">Clear History</button></a>
    </div>
  </div>

</div>
<script>
  $(".scan").click(function () {
    var postData = { csrfmiddlewaretoken: '{{ csrf_token }}' }
    $.ajax({
      url: "{% url 'scan' %}",
      data: {

      },
    });
```

```
    });
</script>

{% endblock content %}
```

details.html (Face recognition page)

```
{% extends "core/base.html" %}
{% load static %}
{% block content %}
<title>{% block title %} TAKE ATTENDANCE{% endblock %}</title>

{% if profile == None %}
<div class="container emp-profile">

   <div class="row">
      <div class="col-md-5">

         <div class="alert alert-warning">
            <svg width="1.0625em" height="1em" viewBox="0 0 17 16" class="bi bi-
exclamation-triangle"
               fill="currentColor" xmlns="http://www.w3.org/2000/svg">
               <path fill-rule="evenodd"
                  d="M7.938 2.016a.146.146 0 0 0-.054.057L1.027 13.74a.176.176 0 0 0-
.002.183c.016.03.037.05.054.06.015.01.034.017.066.017h13.713a.12.12 0 0 0 .066-
.017.163.163 0 0 0 .055-.06.176.176 0 0 0-.003-.183L8.12 2.073a.146.146 0 0 0-.054-
```

```
.057A.13.13 0 0 0 8.002 2a.13.13 0 0 0-.064.016zm1.044-.45a1.13 1.13 0 0 0-1.96 0L.165
13.233c-.457.778.091 1.767.98 1.767h13.713c.889 0 1.438-.99.98-1.767L8.982 1.566z" />
                <path
                    d="M7.002 12a1 1 0 1 1 2 0 1 1 0 0 1-2 0zM7.1 5.995a.905.905 0 1 1 1.8 0l-
.35 3.507a.552.552 0 0 1-1.1 0L7.1 5.995z" />
            </svg>

            <p><b>Note: </b>
            <ul>
                <li>It takes some time to turn on the scanner if your computer hardware
                    does not meet the requirements of the face recognition software</li>
                <li>consider providing good lighting and good resolution cameras
                    for the software to work properly</li>
            </ul>
            <small>(this window will disappear after running the first scan)</small>
            </p>

        </div>
    </div>
    <div class="col-md-1">

    </div>

    <div class="col-md-6">
        <div>
```

```
            <img src="{% static 'img/fgirl.jpg'%}" alt="" width="99%">
          </div>
        </div>


    </div>
</div>
{% else %}
<div class="container emp-profile">
        <div class="row">
            <div class="col-md-4">
                <div class="profile-img">
                    <img src="{{ profile.image.url }}" alt="" />
                </div>
            </div>
            <div class="col-md-6">
                <div class="profile-head">
                    <h5>
                        {{profile.first_name}} {{profile.last_name}}
                    </h5>
                    <h6>
                        {{profile.profession }}
                    </h6>
                    <p                    class="proile-rating">RANKINGS                    :
<span>{{profile.ranking}}/10</span></p>
                    <ul class="nav nav-tabs" id="myTab" role="tablist">
```

```
                <li class="nav-item">
                    <a class="nav-link active" id="home-tab" data-toggle="tab" href="#home"
role="tab"
                        aria-controls="home" aria-selected="true">About</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" id="profile-tab" data-toggle="tab" href="#profile"
role="tab"
                        aria-controls="profile" aria-selected="false">Timeline</a>
                </li>
            </ul>
        </div>
    </div>
    <div class="col-md-2">
        <a href="{% url 'edit_profile' profile.id %}"><input type="button" class="profile-
edit-btn" name="btnAddMore" value="Edit Profile" /></a>
    </div>
</div>
<div class="row">
    <div class="col-md-4">
        <div class="profile-work">
            <p>WORK LINK</p>
            <a href="">Website Link</a><br />
            <a href="">Bootsnipp Profile</a><br />
            <a href="">Bootply Profile</a>
```

```
            <p>SKILLS</p>
            <a href="">Web Designer</a><br />
            <a href="">Web Developer</a><br />
            <a href="">WordPress</a><br />
            <a href="">WooCommerce</a><br />
            <a href="">PHP, .Net</a><br />
          </div>
        </div>
        <div class="col-md-8">
          <div class="tab-content profile-tab" id="myTabContent">
            <div class="tab-pane fade show active" id="home" role="tabpanel" aria-labelledby="home-tab">
              <div class="row">
                <div class="col-md-6">
                  <label>Profile Id</label>
                </div>
                <div class="col-md-6">
                  <p>{{profile.id}}</p>
                </div>
              </div>
              <div class="row">
                <div class="col-md-6">
                  <label>Name</label>
                </div>
                <div class="col-md-6">
```

```
            <p>{{profile.first_name}} {{profile.last_name}}</p>
        </div>
    </div>
    <div class="row">
        <div class="col-md-6">
            <label>Email</label>
        </div>
        <div class="col-md-6">
            <p>{{profile.email}}</p>
        </div>
    </div>
    <div class="row">
        <div class="col-md-6">
            <label>Phone</label>
        </div>
        <div class="col-md-6">
            <p>0{{profile.phone}}</p>
        </div>
    </div>
    <div class="row">
        <div class="col-md-6">
            <label>Profession</label>
        </div>
        <div class="col-md-6">
            <p>{{ profile.profession }}</p>
```

```
                </div>
            </div>
        </div>
        <div class="tab-pane fade" id="profile" role="tabpanel" aria-
labelledby="profile-tab">
            <div class="row">
                <div class="col-md-6">
                    <label>Experience</label>
                </div>
                <div class="col-md-6">
                    <p>Expert</p>
                </div>
            </div>
            <div class="row">
                <div class="col-md-6">
                    <label>Hourly Rate</label>
                </div>
                <div class="col-md-6">
                    <p>10$/hr</p>
                </div>
            </div>
            <div class="row">
                <div class="col-md-6">
                    <label>Total Projects</label>
                </div>
```

```
    <div class="col-md-6">
        <p>230</p>
    </div>
</div>
<div class="row">
    <div class="col-md-6">
        <label>English Level</label>
    </div>
    <div class="col-md-6">
        <p>Expert</p>
    </div>
</div>
<div class="row">
    <div class="col-md-6">
        <label>Availability</label>
    </div>
    <div class="col-md-6">
        <p>6 months</p>
    </div>
</div>
<div class="row">
    <div class="col-md-12">
        <label>Your Bio</label><br />
        <p>Your detail description</p>
    </div>
```

```
                </div>
            </div>
        </div>
    </div>
</div>
{% endif %}



<p id="last_face" class="d-none">{{last_face}}</p>


<script>
    setInterval(function () {
        var url = 'http://127.0.0.1:8000/ajax/';
        var last_face = document.getElementById('last_face').innerHTML;
        $.ajax({
            url: url,
            success: function (data) {
                if (data != last_face) {
                    location.reload()
                }
            }
        })
    }, 3000);
```

view.py file

This is the file that handles all the process of the code

```
from django.shortcuts import render, HttpResponse, redirect
from .models import *
from .forms import *
import face_recognition
from PIL import Image
import cv2
import numpy as np
import winsound
from django.db.models import Q
from playsound import playsound
import os


last_face = 'no_face'
current_path = os.path.dirname(__file__)
sound_folder = os.path.join(current_path, 'sound/')
face_list_file = os.path.join(current_path, 'face_list.txt')
sound = os.path.join(sound_folder, 'beep.wav')


def index(request):
    scanned = LastFace.objects.all().order_by('date').reverse()
```

```
    present = Profile.objects.filter(present=True).order_by('updated').reverse()
    absent = Profile.objects.filter(present=False).order_by('shift')
    context = {
        'scanned': scanned,
        'present': present,
        'absent': absent,
    }
    return render(request, 'core/index.html', context)


def ajax(request):
    last_face = LastFace.objects.last()
    context = {
        'last_face': last_face
    }
    return render(request, 'core/ajax.html', context)


def scan(request):
    global last_face

    known_face_encodings = []
    known_face_names = []
```

```
profiles = Profile.objects.all()
for profile in profiles:
    person = profile.image
    image_of_person = face_recognition.load_image_file(f'media/{person}')
    person_face_encoding = face_recognition.face_encodings(image_of_person)[0]
    known_face_encodings.append(person_face_encoding)
    known_face_names.append(f'{person}'[:-4])



video_capture = cv2.VideoCapture(0)

face_locations = []
face_encodings = []
face_names = []
process_this_frame = True

while True:

    ret, frame = video_capture.read()
    small_frame = cv2.resize(frame, (0, 0), fx=0.25, fy=0.25)
    rgb_small_frame = small_frame[:, :, ::-1]

    if process_this_frame:
        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(
        known_face_encodings, face_encoding)
    name = "Unknown"

    face_distances = face_recognition.face_distance(
        known_face_encodings, face_encoding)
    best_match_index = np.argmin(face_distances)
    if matches[best_match_index]:
        name = known_face_names[best_match_index]

        profile = Profile.objects.get(Q(image__icontains=name))
        if profile.present == True:
            pass
        else:
            profile.present = True
            profile.save()

        if last_face != name:
            last_face = LastFace(last_face=name)
            last_face.save()
            last_face = name
```

```
            winsound.PlaySound(sound, winsound.SND_ASYNC)
        else:
            pass

    face_names.append(name)

process_this_frame = not process_this_frame

for (top, right, bottom, left), name in zip(face_locations, face_names):
    top *= 4
    right *= 4
    bottom *= 4
    left *= 4

    cv2.rectangle(frame, (left, top), (right, bottom), (0, 0, 255), 2)

    cv2.rectangle(frame, (left, bottom - 35),
                  (right, bottom), (0, 0, 255), cv2.FILLED)
    font = cv2.FONT_HERSHEY_DUPLEX
    cv2.putText(frame, name, (left + 6, bottom - 6),
                font, 0.5, (255, 255, 255), 1)

cv2.imshow('Video', frame)

if cv2.waitKey(1) & 0xFF == 13:
```

```
            break


    video_capture.release()
    cv2.destroyAllWindows()
    return HttpResponse('scaner closed', last_face)



def profiles(request):
    profiles = Profile.objects.all()
    context = {
        'profiles': profiles
    }
    return render(request, 'core/profiles.html', context)



def details(request):
    try:
        last_face = LastFace.objects.last()
        profile = Profile.objects.get(Q(image__icontains=last_face))
    except:
        last_face = None
        profile = None

    context = {
        'profile': profile,
```

```
        'last_face': last_face
    }
    return render(request, 'core/details.html', context)


def add_profile(request):
    form = ProfileForm
    if request.method == 'POST':
        form = ProfileForm(request.POST,request.FILES)
        if form.is_valid():
            form.save()
            return redirect('profiles')
    context={'form':form}
    return render(request,'core/add_profile.html',context)


def edit_profile(request,id):
    profile = Profile.objects.get(id=id)
    form = ProfileForm(instance=profile)
    if request.method == 'POST':
        form = ProfileForm(request.POST,request.FILES,instance=profile)
        if form.is_valid():
            form.save()
            return redirect('profiles')
    context={'form':form}
```

```
    return render(request,'core/add_profile.html',context)



def delete_profile(request,id):
    profile = Profile.objects.get(id=id)
    profile.delete()
    return redirect('profiles')



def clear_history(request):
    history = LastFace.objects.all()
    history.delete()
    return redirect('index')



def reset(request):
    profiles = Profile.objects.all()
    for profile in profiles:
        if profile.present == True:
            profile.present = False
            profile.save()
        else:
            pass
    return redirect('index')
```

models.py (this is the database creation code)

```
import datetime
from time import time

from django.db import models



types = [('employee','employee'),('visitor','visitor')]
class Profile(models.Model):
    first_name = models.CharField(max_length=70)
    last_name = models.CharField(max_length=70)
    date = models.DateField()
    phone = models.BigIntegerField()
    email = models.EmailField()
    # ranking = models.IntegerField()
    profession = models.CharField(max_length=200)
    status =
models.CharField(choices=types,max_length=20,null=True,blank=False,default='employee')
    present = models.BooleanField(default=False)
    image = models.ImageField()
    updated = models.DateTimeField(auto_now=True)
```

```
    shift = models.TimeField()
    def __str__(self):
        return self.first_name +' '+self.last_name


class LastFace(models.Model):
    last_face = models.CharField(max_length=200)
    date = models.DateTimeField(auto_now_add=True)
    def __str__(self):
        return self.last_face
```

urls.py (this handles all the routes to another page)

```
from django.urls import path,include
from .views import *


urlpatterns = [
    path('', index, name= 'index'),
    path('ajax/', ajax, name= 'ajax'),
    path('scan/',scan,name='scan'),
    path('profiles/', profiles, name= 'profiles'),
    path('details/', details, name= 'details'),
```

```
    path('add_profile/',add_profile,name='add_profile'),
    path('edit_profile/<int:id>/',edit_profile,name='edit_profile'),
    path('delete_profile/<int:id>/',delete_profile,name='delete_profile'),


    path('clear_history/',clear_history,name='clear_history'),
    path('reset/',reset,name='reset'),


]
```

index.html (landing page)

```
{% extends "core/base.html" %}
{% load static %}
{% block content %}

<div class="container emp-profile">

    <div class="row">
        <div class="col-md-3">
            <img src="{% static 'img/fcicon.png'%}" alt="" width="50%">
            <h6 class="text-muted p-1"> FACIAL SYSTEM</h6>

        </div>
        <div class="col-md-6">
```

```
            <div class="profile-head">
            </div>
        </div>
        <div class="col-md-3">
            <a target="_blank" class="scan" href="{% url 'details' %}"><input type="submit"
class="profile-edit-btn m-2"
                    name="btnAddMore" value="Take Attendance" /></a>
            <a  href="{% url 'add_profile' %}"><input type="submit" class="profile-edit-btn m-2
"
                    name="btnAddMore" value="Add New Employee" /></a>
            <a  href="{% url 'profiles' %}"><input type="submit" class="profile-edit-btn m-2 "
                    name="btnAddMore" value="All Employee" /></a>
        </div>
    </div>
    <hr>
    <ul class="nav nav-tabs" id="myTab" role="tablist">
        <li class="nav-item" role="presentation">
            <a class="nav-link active" id="present-tab" data-toggle="tab" href="#present"
role="tab"
            aria-controls="home" aria-selected="true">Present</a>
        </li>
        <li class="nav-item" role="presentation">
            <a class="nav-link" id="absent-tab" data-toggle="tab" href="#absent" role="tab" aria-
controls="profile"
                aria-selected="false">Employee Details</a>
```

```
        </li>
        <li class="nav-item" role="presentation">
            <a class="nav-link" id="history-tab" data-toggle="tab" href="#history" role="tab"
aria-controls="contact"
                aria-selected="false">Attendance</a>
        </li>
    </ul>



</script>


{% endblock content %}



add_profile.html (Page to add new employee details)

{% extends "core/base.html" %}

{% load static %}

{% block content %}


<div class="container emp-profile pb-3">
    <form method="POST" action="." class="needs-validation" enctype='multipart/form-
data'>
        {% csrf_token %}
        <div class="row">
            <div class="col-md-4">
```

```
<h5 class="text-muted">Personal Informations:</h5><br>


<div class="d-flex justify-content-between">
    <div class="m-1 mr-2">
        <label class="text-muted"><small>First Name:</small></label>
        {{ form.first_name }}
    </div>


    <div class="m-1">
        <label class="text-muted"><small>Last Name:</small></label>
        {{ form.last_name }}
    </div>
</div>


<div class="ml-1">
    <label class="text-muted"><small>Phone Number:</small></label>
    {{ form.phone }}
</div>
<div class="m-1">
    <label class="text-muted"><small>Email:</small></label>
    {{ form.email }}
</div>
<div class="m-1">
    <label class="text-muted"><small>Birth date:</small></label>
    {{ form.date }}
```

```
    </div>
</div>



<div class="col-md-4">
    <h5 class="text-muted">Profession Informations:</h5><br>
    <div class="m-1">
        <label class="text-muted"><small>Profession:</small></label>
        {{ form.profession }}
    </div>
    <div class="m-1">
        <label class="text-muted"><small>Status:</small></label>
        {{ form.status }}
    </div>
    <div class="m-1">
        <label class="text-muted"><small>Shift:</small></label>
        {{ form.shift }}
    </div>
    <!-- <div class="m-1" style="width: 4rem;">
        <label class="text-muted"><small>Ranking/10:</small></label>
        {{ form.ranking }}
    </div> -->
</div>

<div class="col-md-4">
```

```
<h5 class="text-muted">Biometric Photo:</h5><br>
<div>
  <div class="alert alert-warning">
    <svg width="1.0625em" height="1em" viewBox="0 0 17 16" class="bi bi-exclamation-triangle"
      fill="currentColor" xmlns="http://www.w3.org/2000/svg">
      <path fill-rule="evenodd"
        d="M7.938 2.016a.146.146 0 0 0-.054.057L1.027 13.74a.176.176 0 0 0-.002.183c.016.03.037.05.054.06.015.01.034.017.066.017h13.713a.12.12 0 0 0 .066-.017.163.163 0 0 0 .055-.06.176.176 0 0 0-.003-.183L8.12 2.073a.146.146 0 0 0-.054-.057A.13.13 0 0 0 8.002 2a.13.13 0 0 0-.064.016zm1.044-.45a1.13 1.13 0 0 0-1.96 0L.165 13.233c-.457.778.091 1.767.98 1.767h13.713c.889 0 1.438-.99.98-1.767L8.982 1.566z" />
      <path
        d="M7.002 12a1 1 0 1 1 2 0 1 1 0 0 1-2 0zM7.1 5.995a.905.905 0 1 1 1.8 0l-.35 3.507a.552.552 0 0 1-1.1 0L7.1 5.995z" />
    </svg>
    <small>use only biometric pictures with .jpg format</small>
  </div>
  <img src="{% static 'img/bio.jpg'%}" alt="" width="100%">
</div>
<div class="mt-3">
  {{ form.image }}
</div>
</div>
```

```
        </div>
        <div class="mt-3">
            <button type="submit" class="btn btn-outline-success m-1">Save</button>
            <a href="{% url 'profiles' %}"><button type="button" class="btn btn-outline-danger
m-1">Cancel</button></a>
        </div>
    </form>
</div>
```

view.py file

This is the file that handles all the process of the code

```
from django.shortcuts import render, HttpResponse, redirect
from .models import *
from .forms import *
import face_recognition
from PIL import Image
import cv2
import numpy as np
import winsound
from django.db.models import Q
from playsound import playsound
import os
```

```
last_face = 'no_face'
current_path = os.path.dirname(__file__)
sound_folder = os.path.join(current_path, 'sound/')
face_list_file = os.path.join(current_path, 'face_list.txt')
sound = os.path.join(sound_folder, 'beep.wav')



def index(request):
    scanned = LastFace.objects.all().order_by('date').reverse()
    present = Profile.objects.filter(present=True).order_by('updated').reverse()
    absent = Profile.objects.filter(present=False).order_by('shift')
    context = {
        'scanned': scanned,
        'present': present,
        'absent': absent,
    }
    return render(request, 'core/index.html', context)



def ajax(request):
    last_face = LastFace.objects.last()
    context = {
        'last_face': last_face
    }
    return render(request, 'core/ajax.html', context)
```

```
def scan(request):

    global last_face


    known_face_encodings = []
    known_face_names = []


    profiles = Profile.objects.all()
    for profile in profiles:
        person = profile.image
        image_of_person = face_recognition.load_image_file(f'media/{person}')
        person_face_encoding = face_recognition.face_encodings(image_of_person)[0]
        known_face_encodings.append(person_face_encoding)
        known_face_names.append(f'{person}'[:-4])



    video_capture = cv2.VideoCapture(0)

    face_locations = []
    face_encodings = []
    face_names = []
    process_this_frame = True
```

```
while True:

    ret, frame = video_capture.read()
    small_frame = cv2.resize(frame, (0, 0), fx=0.25, fy=0.25)
    rgb_small_frame = small_frame[:, :, ::-1]

    if process_this_frame:
        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(
                known_face_encodings, face_encoding)
            name = "Unknown"

            face_distances = face_recognition.face_distance(
                known_face_encodings, face_encoding)
            best_match_index = np.argmin(face_distances)
            if matches[best_match_index]:
                name = known_face_names[best_match_index]

                profile = Profile.objects.get(Q(image__icontains=name))
                if profile.present == True:
```

```
            pass
        else:
            profile.present = True
            profile.save()

        if last_face != name:
            last_face = LastFace(last_face=name)
            last_face.save()
            last_face = name
            winsound.PlaySound(sound, winsound.SND_ASYNC)
        else:
            pass

    face_names.append(name)

process_this_frame = not process_this_frame

for (top, right, bottom, left), name in zip(face_locations, face_names):
    top *= 4
    right *= 4
    bottom *= 4
    left *= 4

    cv2.rectangle(frame, (left, top), (right, bottom), (0, 0, 255), 2)
```

```
            cv2.rectangle(frame, (left, bottom - 35),
                          (right, bottom), (0, 0, 255), cv2.FILLED)
            font = cv2.FONT_HERSHEY_DUPLEX
            cv2.putText(frame, name, (left + 6, bottom - 6),
                        font, 0.5, (255, 255, 255), 1)


        cv2.imshow('Video', frame)


        if cv2.waitKey(1) & 0xFF == 13:
            break


    video_capture.release()
    cv2.destroyAllWindows()
    return HttpResponse('scaner closed', last_face)



def profiles(request):
    profiles = Profile.objects.all()
    context = {
        'profiles': profiles
    }
    return render(request, 'core/profiles.html', context)



def details(request):
```

```
    try:
        last_face = LastFace.objects.last()
        profile = Profile.objects.get(Q(image__icontains=last_face))
    except:
        last_face = None
        profile = None

    context = {
        'profile': profile,
        'last_face': last_face
    }
    return render(request, 'core/details.html', context)


def add_profile(request):
    form = ProfileForm
    if request.method == 'POST':
        form = ProfileForm(request.POST,request.FILES)
        if form.is_valid():
            form.save()
            return redirect('profiles')
    context={'form':form}
    return render(request,'core/add_profile.html',context)
```

```
def edit_profile(request,id):
    profile = Profile.objects.get(id=id)
    form = ProfileForm(instance=profile)
    if request.method == 'POST':
        form = ProfileForm(request.POST,request.FILES,instance=profile)
        if form.is_valid():
            form.save()
            return redirect('profiles')
    context={'form':form}
    return render(request,'core/add_profile.html',context)



def delete_profile(request,id):
    profile = Profile.objects.get(id=id)
    profile.delete()
    return redirect('profiles')



def clear_history(request):
    history = LastFace.objects.all()
    history.delete()
    return redirect('index')



def reset(request):
```

```
    profiles = Profile.objects.all()
    for profile in profiles:
        if profile.present == True:
            profile.present = False
            profile.save()
        else:
            pass
    return redirect('index')
```

models.py (this is the database creation code)

```
import datetime
from time import time

from django.db import models

types = [('employee','employee'),('visitor','visitor')]
class Profile(models.Model):
    first_name = models.CharField(max_length=70)
    last_name = models.CharField(max_length=70)
    date = models.DateField()
```

```
    phone = models.BigIntegerField()
    email = models.EmailField()
    # ranking = models.IntegerField()
    profession = models.CharField(max_length=200)
    status =
models.CharField(choices=types,max_length=20,null=True,blank=False,default='employee')
    present = models.BooleanField(default=False)
    image = models.ImageField()
    updated = models.DateTimeField(auto_now=True)
    shift = models.TimeField()
    def __str__(self):
        return self.first_name +' '+self.last_name



class LastFace(models.Model):
    last_face = models.CharField(max_length=200)
    date = models.DateTimeField(auto_now_add=True)
    def __str__(self):
        return self.last_face
```

urls.py (this handles all the routes to another page)

```
from django.urls import path,include
from .views import *
```

```
urlpatterns = [
    path('', index, name= 'index'),
    path('ajax/', ajax, name= 'ajax'),
    path('scan/',scan,name='scan'),
    path('profiles/', profiles, name= 'profiles'),
    path('details/', details, name= 'details'),

    path('add_profile/',add_profile,name='add_profile'),
    path('edit_profile/<int:id>/',edit_profile,name='edit_profile'),
    path('delete_profile/<int:id>/',delete_profile,name='delete_profile'),

    path('clear_history/',clear_history,name='clear_history'),
    path('reset/',reset,name='reset'),

]
```

index.html (landing page)

```
{% extends "core/base.html" %}
{% load static %}
{% block content %}
```

```
<div class="container emp-profile">

    <div class="row">
        <div class="col-md-3">
            <img src="{% static 'img/fcicon.png'%}" alt="" width="50%">
            <h6 class="text-muted p-1"> FACIAL SYSTEM</h6>


        </div>
        <div class="col-md-6">
            <div class="profile-head">
            </div>
        </div>
        <div class="col-md-3">
            <a target="_blank" class="scan" href="{% url 'details' %}"><input type="submit"
class="profile-edit-btn m-2"
                    name="btnAddMore" value="Take Attendance" /></a>
            <a  href="{% url 'add_profile' %}"><input type="submit" class="profile-edit-btn m-2
"
                    name="btnAddMore" value="Add New Employee" /></a>
            <a  href="{% url 'profiles' %}"><input type="submit" class="profile-edit-btn m-2 "
                    name="btnAddMore" value="All Employee" /></a>
        </div>
    </div>
    <hr>
    <ul class="nav nav-tabs" id="myTab" role="tablist">
```

```
<li class="nav-item" role="presentation">
  <a class="nav-link active" id="present-tab" data-toggle="tab" href="#present" role="tab"
    aria-controls="home" aria-selected="true">Present</a>
</li>
<li class="nav-item" role="presentation">
  <a class="nav-link" id="absent-tab" data-toggle="tab" href="#absent" role="tab" aria-controls="profile"
    aria-selected="false">Employee Details</a>
</li>
<li class="nav-item" role="presentation">
  <a class="nav-link" id="history-tab" data-toggle="tab" href="#history" role="tab" aria-controls="contact"
    aria-selected="false">Attendance</a>
</li>
</ul>
```

Bio-data

Personal Data:
Surname: OWOLABI
Other Names: BUKOLA MEKA
Date of Birth: 20th December, 1994
Gender: Female
Local Government Area: Magongo
State of Origin: Kogi State
Nationality: Nigerian
Marital Status: Married
Religion: Christianity
Next of Kin: Dr Femi Owolabi
Next of Kin Address: Manchester city,England, United Kingdom.
Contact:
Residential Address: No 9, Owo close, garki, Abuja,Nigeria
Postal Address: Same as above.
Email Address: owolabimeka@gmail.com
Mobile No: 07033643657
Educational Background:
Institutions Attended with Dates and Qualification:
Lead City University, Toll Gate, Ibadan, Oyo State: (MSc. Computer Science) 2021 till date
Lead city University, Toll Gate, Ibadan, Oyo State: (Masters in Business Administration) MBA 2020-2022
Ajayi Crowther University Oyo, Oyo state : (B.Sc Computer Science) 2012-2016

Professional Qualifications
Google Certificate of Completion – Coursera – March 25, 2023
Google Certificate of Completion – Coursera -March 14, 2023
Google Certificate of Completion – Coursera - Jan 18, 2023
Google Certificate of Completion – Coursera – Nov 2022
Certificate in Fundamentals of Digital Marketing
Google Digital Skill for Africa, Online, 2020
Certificate in Project Management
Hayroyal Consult, Oyo, 2016

Certifications
Web Design with html & Dreamweaver cs6 2015
Work Experience with Dates
• National Open University, Enugu, Nigeria 2017 & 2018
• Project Manager, Mekanancy Beautitoolz, Abuja,Nigeria 2019-2020

Published Journal Articles
Wumi Ajayi, Owolabi Bukola, Bello Muhammed, Adeyemi Oluwaseye "Quick Response Mechanism Payment System using Waterfall Model" European Journal Computer Science and Information Technology(EJCSIT) , October 2021.
Wumi Ajayi, Owolabi Bukola, Jolaosho Ahmed "Improving Test Automation Using Genetic Algorithm" IJARCS Journal. April 2022.

The University Compliance Certification

This is to certify that this thesis written by Bukola Meka OWOLABI with the matriculation number LCU/PG/002526 in the Department of Computer Science, Faculty of Natural and Applied Sciences, Lead City University, Ibadan, Oyo State, Nigeria is in full compliance with the approved University Format and Style.

_____ _____

Signature Date

Bukola LCU Library

ORIGINALITY REPORT

8%	4%	0%	7%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	mafiadoc.com Internet Source	2%
2	Submitted to Bournemouth University Student Paper	2%
3	Submitted to Eiffel Corporation Student Paper	2%
4	Submitted to University of Bolton Student Paper	2%

Exclude quotes	Off	Exclude matches	< 2%
Exclude bibliography	On