BINDURA UNIVERSITY OF SCIENCE EDUCATION

FACULTY OF SCIENCE EDUCATION

DEPARTMENT OF COMPUTER SCIENCE

TOPIC

Mobile face recognition id verification system for Government Institutes using local binary pattern histogram algorithm.

SUBMITTED BY

B201452B

A DISSERTATION SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR A BACHELOR HONOURS DEGREE IN INFORMATION TECHNOLOGY AT BINDURA UNIVERSITY OF SCIENCE EDUCATION.

2024

RELEASE FORM

Name of student Dylan Muza

Dissertation title Mobile face recognition id verification system for Government Institutes using local binary pattern histogram algorithm.

Year granted 2024

Permission is granted to the Bindura University of Science Education (BUSE) Library and the department of Computer Science to produce copies for private, scholarly and research/academic purposes only. The author reserves other publication rights and neither copies nor extracts from this dissertation can be reproduced anyhow without the author's permission

Permanent residence **14995 Unit O Seke Chitungwiza, Harare**

Signed **……………………………………………………**

Date **……07/10/2024……………………………**

SIGNED APPROVAL FORM

The undersigned certify that they have read and recommended to the Bindura University of Science Education for the acceptance of a dissertation entitled **"Mobile face recognition id verification system for Government Institutes using local binary pattern histogram algorithm.".**

……………………………………… **……………………………….**

Signature of student **date**

……………………………… ………………………………

Signature of the supervisor **date**

……………………………… ………………………………

Signature of department chairperson **date**

DEDICATION

This research is dedicated to my parents and siblings. They are my source of strength.

ABSTRACT

Throughout this project, I created a customized mobile face recognition system for government organizations by implementing the Local Binary Pattern Histogram (LBPH) algorithm. The LBPH algorithm proved to be resilient to changes in lighting and pose, resulting in accurate face recognition. Valuable input from government personnel was instrumental in refining the system. It is advisable to continue utilizing an iterative development approach, ensuring adherence to privacy regulations, and offering training and assistance for smooth integration.

ACKNOWLEDGEMENTS

The Lord Almighty for his grace throughout this research, his mercies shown upon my life everyday made it possible for me to go on and be strengthened in times of weakness.

I acknowledge my supervisor MR Ndumiyana for his contribution towards this dissertation, I could not have done it without him.

Contents

List of figures

List of Abbreviations

UI	User Interface
ID	Identity
2D	2 Dimensional
3D	3 Dimensional
PCA	Principal Component Analysis
LBPH	Local Binary Pattern Histogram
ICT	Information Communication Technology

CHAPTER 1: PROBLEM IDENTIFICATION

1.1 Introduction

It starts with a summary of the importance of identification as well as the dangers, difficulties, and achievements involved in creating a reliable and inclusive ID system. An identification document is any document that can be used to prove someone's identity. If it is issued in a little credit card-sized format, it is commonly called an identity card.

Increasingly widespread and reasonably priced technology has led to the implementation of inclusive digital identity systems in several regions of the world nowadays. As demonstrated by the ability of citizens in Belgium, Estonia, Finland, France, the Republic of Korea, and Singapore to pay taxes online or seek official papers, digital identity systems have created new avenues for civil involvement and improved the effectiveness of public services. This study highlights the significance of contemporary technologies and applies constructive research methodology, often referred to as research design methodology, to produce documentation that facilitates the creation of a mobile verification software that authenticates an individual's identification.

1.2 Background Study

The evolution of company identification, commonly known as company IDs, has undergone significant changes over the years. Initially, company IDs were primarily physical cards issued to employees for identification purposes within the organization. These cards typically included basic information such as the employee's name, photo, and job title. They served as a visual means of verification, ensuring that only authorized personnel could access company facilities or resources.

However, with the rapid advancement of technology, the concept of company IDs has expanded beyond physical cards. Today, many organizations have transitioned to digital identification systems. These digital IDs often take the form of smart cards, virtual badges, or mobile apps. They incorporate advanced features such as biometric authentication, encrypted data storage, and remote access capabilities [3].

The shift towards digital company IDs brings multiple benefits. First and foremost, it enhances security by making forgery or unauthorized duplication more difficult. With biometric factors like fingerprints or facial recognition, it becomes challenging to counterfeit or misuse the identification. Additionally, digital IDs offer more flexibility in terms of access control. Companies can easily update permissions or revoke access remotely, ensuring that only authorized individuals can enter specific areas or utilize certain resources [4].

Furthermore, digital company IDs contribute to streamlined processes and improved efficiency. They can be integrated with time and attendance systems, allowing employees to clock in and out with a simple scan or tap.

1.3 Problem Statement

Due to the advancement of technology many individuals can make imitations of physical ids, so a digital migration will be best for id verification. They have been many cases of bogus government institutes workers who have been able to defraud innocent citizens. They have been bogus ZETDC employees who have been able defraud residents by flashing a ZESA identity card, and without any way of verification there is no way to distinguish between a fake and an actual ZESA employee [5].

1.4 Objectives

- Use a face recognition algorithm to identify users.
- Implement a face ID authentication system for mobile devices.
- Protect the integrity of confidential user's biometric information.

1.5 Research Questions

Face ID verification system could encompass a variety of aspects from technical implementation to user experience and security considerations. Here are some research questions:

- Security and Privacy Concerns:
    - How does Face ID protect user data and ensure secure authentication?

- Technical Evaluation:
    - How does Face ID technology work, and what sets it apart from other facial recognition systems?

- User Acceptance and experience:
    - Evaluate the use of face recognition ID verification system for diverse user demographics.

1.6 Justification

The implementation of a face ID verification system is expected to offer several key benefits and address important needs in various industries:

1. Enhanced Security: Will be developing a face recognition technology that aims to offer robust biometric authentication. While still under refinement, it is designed to be a formidable barrier against replication and unauthorized access.
2. Convenience: The system is being engineered to provide effortless identity verification. Once completed, users will be able to confirm their identity with a simple facial scan, potentially eliminating the need for physical IDs.

3. Fraud Detection: I am are actively working on enhancing the system's ability to thwart fraudulent activities. By comparing live scans with my growing database, I strive to identify and prevent identity theft and unauthorized entry attempts.
4. Efficiency: The face recognition ID verification system is being optimized for speed and efficiency. Fine-tuning the process to ensure that authentication is both rapid and reliable.

1.7 Assumption

When considering the implementation of a face recognition ID verification system, it's important to take certain assumptions into account. The following are a few key assumptions associated with such systems:

1. **Quality of Data**: Access to high-quality, diverse facial images to train the LBPH algorithm effectively.

2. **Non-biased Algorithms**: It is assumed that the face recognition algorithms are trained and tested to be unbiased towards factors such as ethnicity, gender, or age to prevent discriminatory outcomes.

3. **Lighting and Environmental Conditions**: The system assumes that sufficient lighting and favourable environmental conditions are in place to capture clear and accurate images for facial recognition.
4. **User Cooperation**: Users are expected to cooperate by positioning their faces correctly within the scanning area and following any required instructions for successful verification.

By acknowledging and addressing these assumptions, this allows for better planning, successful implementation and operation of a face recognition ID verification system.

1.8 Limitation

The greatest hurdle that will be faced when developing this software will be that of time. In order to meet the project submission deadline, the researcher will have to work consistently in order to train the facial recognition model, create databases and design mobile application through coding. This will be a lot of work that needs to be done in a matter of months, considering that the researcher is a final year student who will be concurrently undertaking courses for their undergraduate degree program.

1.9 Delimitation of the research

When delimiting a face recognition ID verification system, it's essential to establish boundaries and constraints to manage the scope effectively. These are some common areas where delimitations may be considered in a face recognition ID verification project:

1. **Hardware Limitations**: The project may be delimited by the hardware available for implementing the face recognition system, such as processing power, memory, camera quality, etc.

2. **Database Size**: Limiting the size of the database of individuals to be recognized can be a delimitation. A larger database may require more complex algorithms and longer processing times.

3. **Environmental Conditions**: Delimiting the project to specific environmental conditions (e.g., lighting conditions, camera angles) can help manage the system's performance and accuracy within certain boundaries.
4. **Security and Privacy Regulations**: Adhering to specific security and privacy regulations may limit the capabilities and design of the face recognition system.

5. **Integration with Other Systems**: Delimiting the project by specifying the integration with existing systems or platforms can help in defining the project's scope effectively.

6. **Time Constraints**: Establishing specific time constraints for the project can help in managing resources and ensuring timely delivery.

7. **Accuracy Expectations**: Clarifying the expected level of accuracy and error tolerance in the face recognition system can be an important delimitation.

8. **Budget Constraints**: Limiting the project based on financial resources available can help in ensuring that the project stays within budget.

9. **User Interface Complexity**: Delimiting the complexity of the user interface for interacting with the face recognition system can help in streamlining the project scope.

10. **Training Data Availability**: Delimiting the project based on the availability of training data for the face recognition system can be crucial for determining its performance.

1.10 Definition of terms

Here are key terms related to a face recognition ID verification system:

1.Face Recognition: The process of analysing patterns based on facial features from an image or video in order to identify or confirm an individual's identity.

2.ID Verification: The process of confirming the authenticity of an individual's claimed identity using official identification documents or biometric data, such as facial recognition.

3.Biometric Data: Distinct biological traits such as fingerprints, iris patterns or facial features that are utilized for identification.

4.Machine Learning: Without explicit programming, machine learning is a branch of artificial intelligence that allows computers to learn from experience and get better over time.

5.Algorithm: A set of rules or procedures for solving a problem or accomplishing a task.

Understanding these project terms is crucial for effectively planning, developing, and implementing a face recognition ID verification system.

1.11 Conclusion

In conclusion, the development of a face recognition ID verification system for government institutes using the Local Binary Pattern Histogram algorithm is poised to revolutionize the way identity verification is conducted. The users of the face recognition ID verification system are primarily employees and personnel within government institutes. Access to these users will be facilitated through integration with Government Databases. This system promises to deliver enhanced security: by leveraging the LBPH algorithm, the system is expected to provide a high level of security against fraudulent activities. Operational efficiency, the automation of ID verification processes will streamline operations, saving time and resources. User-Friendly experience, with the aim of creating an intuitive interface, the system will facilitate easy and quick verification for users. Adaptability, the system is designed to be adaptable, ensuring compatibility with various government functions and requirements. As we move forward, I will continue to refine the technology, ensuring it meets the stringent demands of government security and efficiency.

CHAPTER 2: LITERATURE REVIEW

2.0 Introduction

The previous chapter gave a snippet of what other scholars have put forward with regards to face recognition ID verification system. This chapter is aimed to give an in-depth analysis of the work that has been published regarding the use of facial recognition. The literature will assist the researcher in identifying research gaps as well as what has already been written about a subject and offer a summary of important ideas. In computer vision, facial recognition is a difficult problem to solve. A number of algorithms have been put into practice to create automatic facial recognition systems that are on par with human vision. Computers are able to surpass human constraints due to their endless memory and great processing capacity. Face recognition technology is still needed and remains an unsolved challenge.

Facial recognition algorithms can categorize people based on the photographs of their faces. Face recognition systems authorize a person based solely on facial recognition, as opposed to pins or secret numbers, which might be compromised or used on unauthorized individuals. Voting systems can potentially use face recognition technology to uniquely identify each voter, preventing vote duplication. By limiting system access, facial recognition technology can be used in private enterprises to limit employee access to sensitive data. A person is given permission to access a specific system after being identified. Additionally, picture database investigations can use facial recognition to verify the identities of people crossing national borders and lawfully licensed drivers.

2.1 Machine Learning

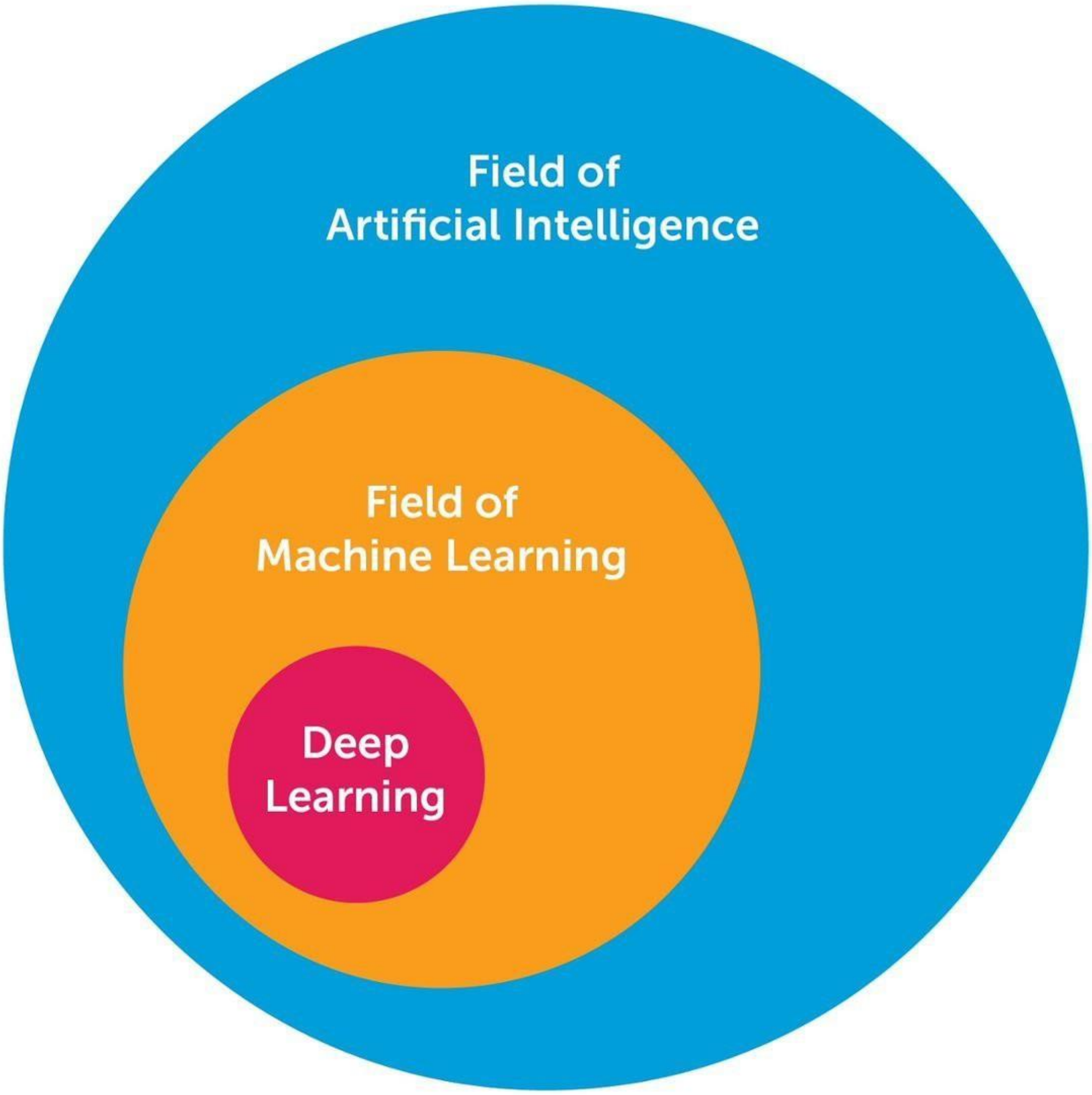
Machine learning being a subfield of artificial intelligence, which broadly describes the machine's capacity to emulate intelligent human behaviour. AI systems are able solve complex problems in a way that is comparable to human problem-solving. The aim of artificial intelligence is to create computer models with "intelligent behaviours" that resemble those of people (Boris Katz). This includes devices that can identify images, comprehend documents written in natural language, or carry out actions in the real world.

Using AI in Machine learning is one method. AI pioneer Aurthur Samuel defined it as “the field of study that gives computers the ability to learn without explicitly being programmed” in the 1950s. That definition is accurate, says Mikey Shulman. In a similar vein, traditional programming calls for writing comprehensive instructions that the machine must obey. However, there are situations (such teaching a computer to distinguish between images of various persons) where developing a program for the machine too follow is impractical or takes a long time. It’s easy for humans to perform this work, but teaching a computer how to do it is more challenging. Using an experience-based method, machine learning teaches computers how to program themselves.

2.1.2 What is deep learning?

Deep learning is a subset of machine learning, which is essentially a three-layer neural network. These neural networks seek to imitate the functioning of the human brain by “learning” from enormous amounts of data, however they are far from reaching its potential. Even with just one layer, a neural network can make intelligent assumptions; however, more hidden layers can assist improve and optimize the network for greater accuracy. Deep learning is the engine of many artificial intelligence services and solutions that boost automation by performing analytical and physical tasks without requiring human participation.

Figure 1 Artificial intelligence



Chollet, Francois. 2017. *Deep Learning with Python*. New York, NY: Manning Publications.

There are two categories for facial recognition techniques: appearance-based and featurebased.

2.2 Appearance-based approaches

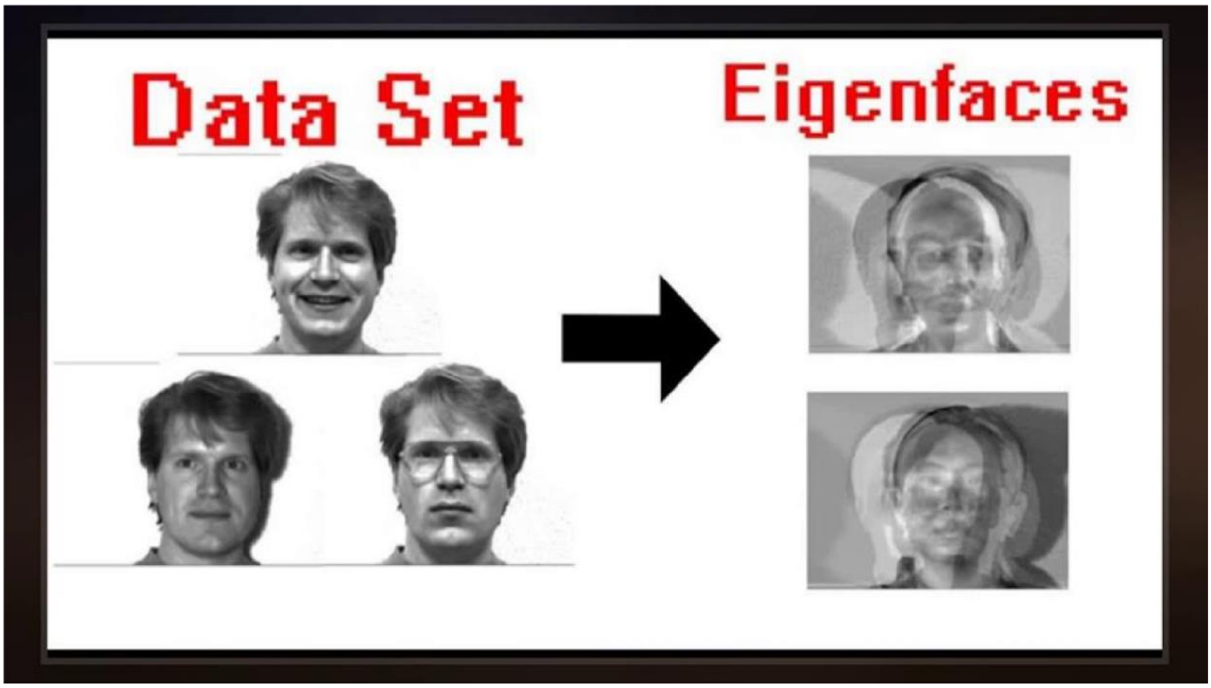
Appearance-based (or holistic matching) techniques, according to Dass, Rani, and Kumar (2012), use the entire face region as the raw input to a recognition system. First, the face recognition problem is reformulated as a face space analysis problem, to which a number of established statistical techniques are then applied. They are highly susceptible to the constraints imposed by changes in facial lighting, 3D postures, and expressions.

2.2.1 Feature based technique

One of the methods most frequently used for face recognition is feature-based techniques. It's a method that makes use of Principal Component Analysis (PCA). Dimensionality decline is effectively achieved with this technique. Principal components are mostly used in face detection and recognition.

The principal components that allow for the mathematical extraction of feature vectors from a face are feature-based. It is possible to obtain the feature vector data as a covariance matrix. The difference between multiple faces is calculated using these Eigenvectors. The maximum Eigenvalues of the faces are grouped linearly to categorize them. Each face is quantifiable as a linear combination of its feature-based components. It is possible to approximate the face with the largest eigenvalues of the eigenvectors. Yagnik and Patel (2013) Eigen face is a practical approach to facial recognition. The simplicity of Eigen face recognition algorithms has made them easy to implement. Eigen faces' accuracy depends on a number of factors. The Eigen face method identifies a way to create faces that resemble ghosts and represent the majority of the variance in an image data set. This technique is based on an evidence theory approach that breaks down face images into a small collection of feature images known as "Eigen faces," which are actually the main elements of the original training set. Eigen face poses a serious

Figure 2 Eigenfaces



P. Bhagyasr i, K. SaiRatn a, K. Aswin i, K. Sahan a, T. V. Kuma r Computer Science 21 February 2018

threat to both the Head's location and lightning environments. The eigenvalues and eigenvectors' phase-consuming result is a drawback. (2015, Science & Engineering)

2.2.1 The feature-based technique's advantages

- Simplicity in execution.
- It's not necessary to understand geometry or any particular face feature.
- Minimal preparation effort.

2.2.2 The feature-based technique's limitations

- Sensitive to scale regarding head.
- Only relevant to front views.
- Only when the background is controlled will perform well (not include natural scenes).

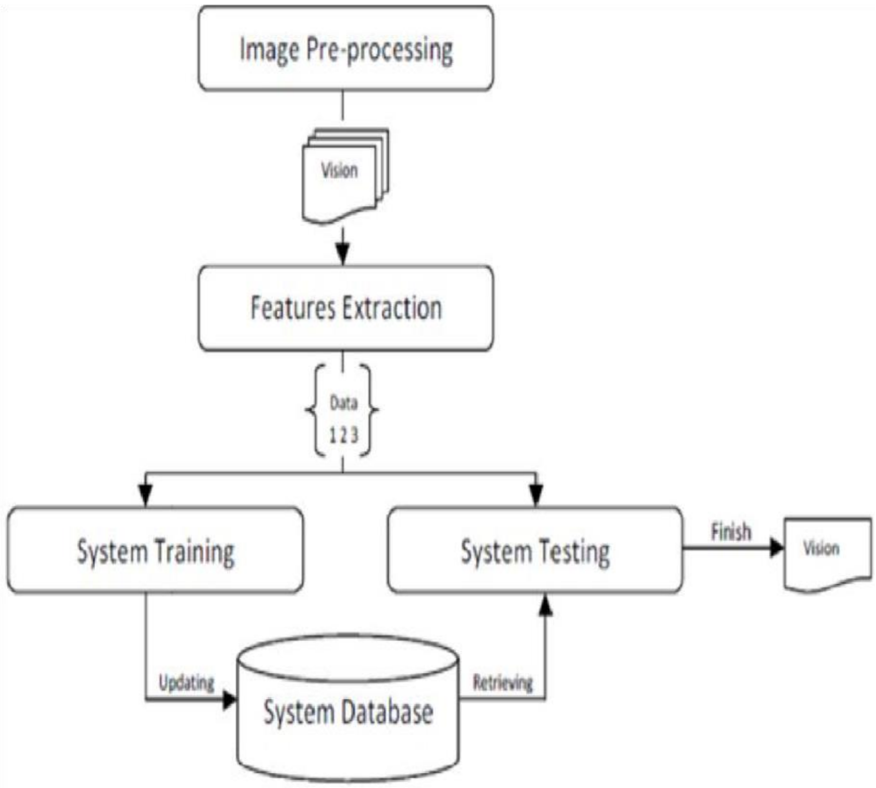
2.3 Principal component analysis

The PCA method, according to Anthony (2016), converts the two-dimensional image into a one-dimensional feature vector. After that, the vector goes through a number of processes, including feature extraction, normalization of facial landmarks, face detection in a picture or live video stream, and face recognition. (Antony, 2016) went on to say that the method chooses the facial features that stand out the most from the background of the picture. Since 5–10% of the components make up 90% of the overall in the face, a significant portion of the data is eliminated during the decomposition process. This indicates that only a small portion of the information in the picture is required to identify a specific person.

(Or Eigen faces), which are kept in an array that is only one dimension deep. Each element, referred to as the principal component, solely depicts a single facial feature. In facial recognition, the distance between the test image's matching feature vectors and a face in the database of faces is used to compare the two. Both the gallery image and the test image need to have the same dimensions (or scale), lighting, and posture in order to yield superior results.

The image in the database and the test image differ in scale because of the PCA algorithm's profound scaling (Antony,2016).

Figure 3 Principal component analysis



2.3.1 Strengths

- Identification is easy and efficient.
- The tiny dimensional subspace portrayal achieves data compression; raw intensity statistics are freely utilized for recognition and learning without significant low- or midlevel restrictions.
- Without any significant low-level or mid-level processing, raw intensity statistics are used freely for learning and recognition.
- It is not necessary to have knowledge of the geometry or reflectance of faces.

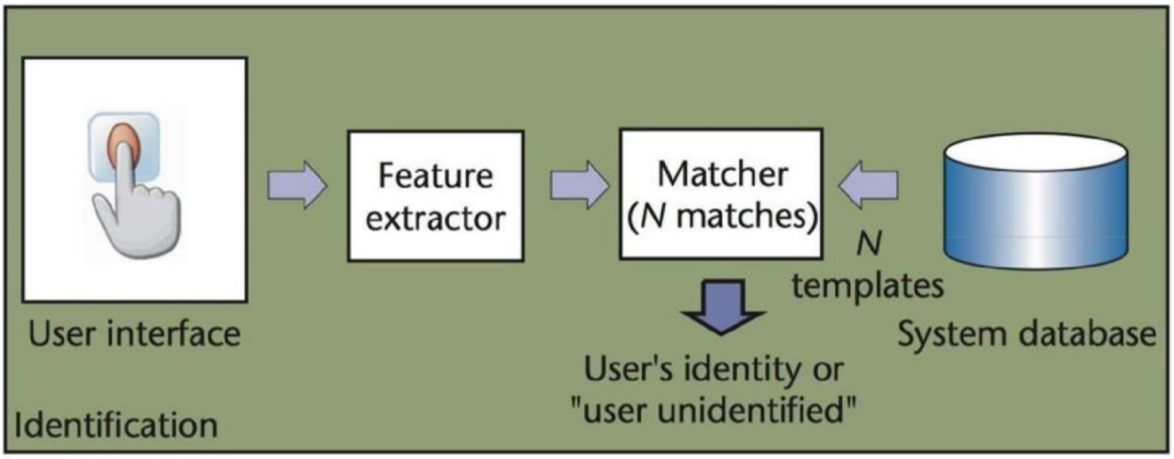
2.3.2 Limitations

- Because of the technique's significant scaling, low-level pre-processing is crucial.
- When illumination and posture change, its recognition rate decreases.
- When there is a significant shift in both appearance and posture, the issue may become more difficult to solve.
- Why Modernizing the face dataset is challenging due to learning's extremely slow pace.

2.4 Local Binary Patterns Histogram

The LBPH was created for texture description, according to (Sharma & Kaur, 2016). Wahid (2013) states that the face image is converted to grayscale in order to be processed further. Second, the entire face is first divided into squared regions, from which histograms and LBPH values are taken and combined to create a single feature vector. This feature vector is used to calculate the degree of similarity between images and creates an effective representation of the face. This algorithm's primary benefit is that, unlike other methods, it is not significantly affected by illumination or pose variations and yields higher recognition rates in controlled environments. Furthermore, unlike other holistic or appearance-based algorithms like Fisher Faces and Feature based, the LBPH algorithm is not impacted by scale variations.

Figure 4 local binary pattern histogram



2.5 Issues with Face Recognition Software

Even though machine recognition systems today are nearly flawless, their usefulness is still constrained by numerous real-world application scenarios. These are a few issues that numerous facial recognition systems run into. The internal camera control and the characteristics of skin reflectance cause variations in illumination, both indoors and outdoors. Extreme lighting can make large portions of the face invisible, which presents challenges for face recognition systems. Moreover, it gets more challenging when pose variation is combined with illumination (Dass et al., 2012).

Extreme facial expressions cause faces to deform greatly, which poses a challenge for algorithms (Dass et al., 2012).

Occlusion: This occurs when other items or accessories, like scarves or glasses, cover the face and hinder the effectiveness of face recognition algorithms. (Dass and others, 2012).

Temporary Disturbance The face of a person ages. Changes in makeup, the presence or lack of facial hair, tense muscles, hairstyles, skin appearance, facial jewellery, glasses, and the effects of aging (Dasset al., 2012).

2.6 Comparison of the algorithms

Given that the LBPH feature-based algorithm has demonstrated superior efficiency over other holistic based algorithms, the researcher noted that it can effectively tackle the facial recognition problem. In contrast to previous holistic approaches, the LBPH technique computes the description of individual features in an image and combines them into a single feature vector that characterizes the image as a histogram, rather than computing the description of the entire face. The LBPH algorithm works well in uncontrolled environments and has significant implications for illumination. Thus, in order to address the illumination issue, the researcher will create a dataset consisting of photos captured in a variety of settings. Prior to feature extraction, the researcher will use image preprocessing to downsize and normalize the photos in order to save disk space and improve the speed of face detection.

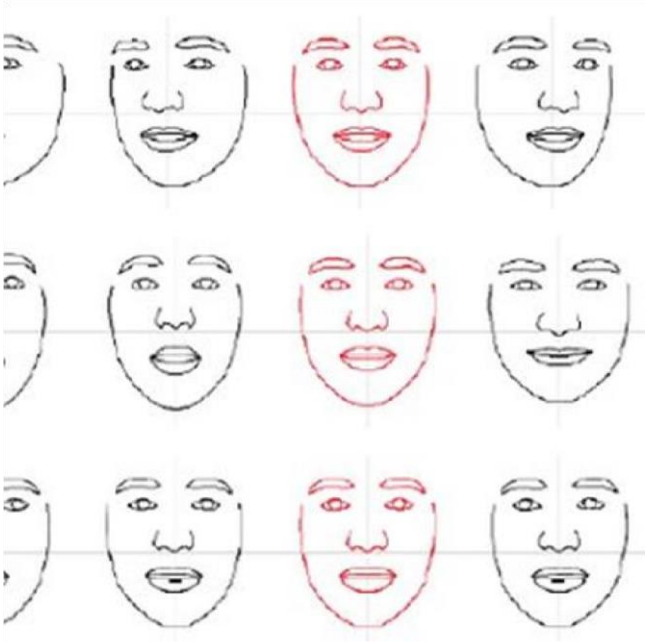
Moreover, the LBPH method remains unaffected by differences in scale between the image stored in the facial database and the probing image. The individual must face the camera directly in order for the LBPH algorithm to recognize them as best as possible. In this case, the researcher will identify a person even if they are not facing the camera directly by using LBP cascades for frontal face detection and profile detection. Since most systems evaluated in the literature use Haar cascades, the researcher used LBPH cascades that are enhanced using Gabor wavelets to improve face detection in real-time.

2.6.1 Fisher face Algorithm

The goal of this study is to create a GUI application and database that can be used to create a face recognition application using the Fisher Face Method using a Papuan facial image. Fisher Faces Algorithm is used in the image recognition process, whereas Minimum Euclidean is used for face recognition. Fisher's Linear Discriminant (FDL) method, also called Linear Discriminant Analysis (LDA) method, is applied to obtain the features of image characteristic. Fisher Faces method of image recognition begins with the Principal Component Analysis

(PCA) method's reduction of the face space dimension. Fisher's Linear Discriminant (FDL) method, also called Linear Discriminant Analysis (LDA) method, is then applied to obtain the features of image characteristic. The present study utilizes a literature review methodology, which comprises reading and analysing a range of books or literature on mathematical concepts that serve as the foundation for the fisher face algorithm. This algorithm is used to identify an individual's face image and is then applied to programming languages, specifically Matlab7.10. The preprocessing step for the utilized Adobe Photo-shop CS4 application program aims to guarantee that the face image is consistent in size and format so that the system can utilize it. The program's success rate for image recognition is 100% when the testing image and the training image are identical; nevertheless, for 73 facial test images with differing expressions and positions, 70 faces are correctly recognized and 3are incorrectly recognized, meaning the program's success rate is 93%.

Figure 5 Fisher face algorithm



2.7 Conclusion

In this digital era, there is an ever increasing need to embrace ICTs in every facet of life. This chapter opened by dissecting the concept of machine learning before looking at the different algorithms that can be used for facial recognition. The researchers gave an in-depth analysis of the various algorithms that can be assumed by facial recognition, citing preexisting works and highlighting their shortcomings The succeeding chapter will look at the various methodologies that will be adopted towards the completion of the project.

CHAPTER 3: RESEARCH METHODOLOGY

3.1 Introduction

A system development methodology is a set of steps or processes employed to conduct research. In the context of software development, a software development methodology divides the work into separate stages. The evolutionary prototyping model was selected as the method for system design. A software system can be developed and improved gradually through the iterative process of the evolutionary prototyping paradigm.

3.2 Evolutionary Prototyping

Evolutionary prototyping is a software development model that focuses on gradually refining and improving a system through iterative cycles of prototyping. In this approach, the researcher will start with a basic face recognition model which is the initial prototype. The prototype will then continue to be enhanced using an incremental approach of adding features like better face detection, improved feature extraction and robust matching techniques. Each iteration taken will be to improve the system accuracy and reliability.

The key idea behind evolutionary prototyping is to quickly develop a working model that can be demonstrated to government institutes and end-users so that they can provide feedback which can then be used to refine the recognition system. This will also be a process of addressing the issue of false positives, false negatives through iterative improvements.

The LBPH algorithm will be trained to adapt to variations in lighting, poses and facial expressions, as the system processes more face samples it will learn to recognize faces more effectively. Each iteration process that is going to be taken will contribute to the system's learning process.

The ultimate goal of evolutionary prototyping is to progressively refine the prototype until it reaches a stable and complete system that satisfies the desired objectives. It promotes collaboration, reduces risks, and improves the overall quality of the final software product.

3.2.1 Basic Steps for Prototyping

1. Initial Prototype

- Objective: Create a basic face recognition system.
- **Steps:**
    - **Face Detection:** Using the Haar cascade classifier (pre-trained model) to detect faces in images.
    - **Feature Extraction:** Extracting local binary pattern (LBP) features from detected faces.
    - **Model Creation:** This will be done by building an initial LBPH face recognition model using the LBP descriptors.
    - **Integration:** Integrate the prototype with existing infrastructure (e.g., databases, security protocols).
- **Requirements:**
    - Hardware (processor, memory, optional GPU). ○ Software (OpenCV, Python, Dart, Flutter). ○ Access to the pre-trained Haar cascade XML file for frontal face detection.

2. User feedback and iteration:

- Objective: Gathering feedback from government institutes and end-users.
- **Steps:**
    - **Deploying the initial prototype in a controlled environment. ○ Collecting feedback on accuracy, usability and performance. ○ Identify areas for improvement**
- **Requirements:**
    - Regular communication channels with users. ○ Capturing feedback from users using surveys and user testing.

3. Enhancements and Refinement:

- Objective: Incrementally improving the system based on feedback.

- **Steps:**
    - **Analysing of feedback to identify specific enhancements. ○ Refining the face detection (reducing the false positives and negatives).**
    - **Optimizing the LBPH parameter by tuning the iterating based on empirical results and domain knowledge.**
- **Requirements:** ○ Accessing the real-world data for testing and training.

4. Adaptive Learning and Scaling:

- Objective: Enhancing the LBPH model's learning capabilities.
- **Steps:**
    - **Continuously updating the LBPH model with new face samples.**
    - **Implementing adaptive learning to handle variations like lighting, pose and facial expressions.**
    - **Scaling the system to handle a growing user base.**
- **Requirements:**
    - Robust data collection and management. ○ Scalable infrastructure (storage).

When creating a Face Recognition ID Verification System for government institutions utilizing the Local Binary Pattern Histogram (LBPH) method, the Evolutionary Prototyping Model is a good approach. These are the explanations for why:

- Adaptability to Changing Requirements:
    - Evolutionary models work effectively when requirements are ambiguous or change frequently.
    - Face recognition systems often involve complex user needs that evolve over time.
    - The model permits modifications and adaptability all the way through.
- Early and Gradual Distribution:
    - Functional components or prototypes can be delivered early in the process.
    - Incremental development allows for early access to core features.
- Risk Mitigation:

    - High-risk projects benefit from evolutionary prototyping. o Continuous feedback and iterative cycles help manage risks.
    - Early risk mitigation guarantees improved system performance.
- Efficient Response to User Feedback:
    - Government institutes often have specific needs and constraints.
    - Evolutionary prototyping allows developers to respond to user feedback promptly.

In summary, the Evolutionary Prototyping Model aligns well with the dynamic nature of face recognition systems, user requirements, and risk management. It guarantees steady advancement, flexibility and ongoing enhancement during the course of development.

3.3 Systems Specification

3.3.1 System Overview

The system will be a mobile application developed using Dart and Flutter. It will use the LBPH algorithm for face recognition. The primary function of the system will be to verify the identity of a user based on their facial features.

3.3.2 Key Components

- User Interface (UI): The UI will be developed using Flutter, which allows for the creation of visually appealing and responsive interfaces.
- Face Recognition Module: This module will use the LBPH algorithm for face recognition. It will process user images and compare them to stored facial data to verify identities.
- Database: User and facial recognition data will be stored in a database. The particular requirements of the system will determine which database is used.
- Authentication Module: This module will handle user authentication, ensuring that only authorized users can access the system.

3.3.3 Development Tools

- Programming Language: Dart.
- Framework: Flutter.
- Database: Firebase.
- Face Recognition Algorithm: Local Binary Pattern Histogram (LBPH).

3.3.4 System Workflow

- The user opens the app and is prompted to choose a database that he/she wants to access.
- After database selection, the user will capture the face to be verified using the device's camera.
- The LBPH algorithm processes the image and tries to find a match in the database.
- If the facial data matches, a screen showing the facial data will show verifying that the person is in the system. If not, a pop-up message will show saying "user is not found in the system".

3.3.5 Key Considerations

- Privacy and Security: User data, especially facial data, is sensitive. The system must comply with relevant privacy laws and use secure methods to store and process data.
- Performance: The face recognition process should be fast and accurate to provide a good user experience.
- Usability: The UI should be intuitive and easy to use.

3.4 Requirements analysis using the Analysis Model

The important process of requirements analysis determines whether a project succeeds or fails. For requirements to be successful, they need to be quantifiable, traceable, actionable, practical, documented, testable, and in line with the business needs that have been defined (Abram Moore, Bourque, & Dupuis, 2004). This is a critical phase where all functional and nonfunctional requirements for the needed system must be carefully documented.

3.4.1 Functional Requirements

The functional requirements of the proposed system involve the ability to identify and differentiate faces from their gestures. This will be accomplished by comparing the image being tested with the data stored in the trained dataset.

- User Registration (Admin Side): The system will allow new users to be registered by providing necessary details and capturing their facial data for future recognition.

- Face Recognition: The system should be able to recognize a registered user’s face using the LBPH algorithm.
- ID Verification: Upon successful face recognition, the system will verify the user’s identity.
- User Interface: The system will provide an intuitive user interface for ID verification.
- Error Handling: The system will handle errors gracefully and provide useful feedback to the user. Network may be unstable in some areas so the system will be able to handle errors like these and provide feedback to the user.
- Security: The system will ensure that user data, especially facial data, is stored and processed securely.
- Performance The face recognition process will be fast and accurate to provide a good user experience.

3.4.2 Non-functional Requirements

These requirements encompass the entire system rather than just individual components. Nonfunctional requirements include aspects such as:

- Performance: The system will be able to process and recognize faces quickly to provide a seamless user experience. The LBPH algorithm will be optimized to ensure fast processing.
- Reliability: The system will accurately recognize and verify faces all the time. The LBPH algorithm will be tuned to minimize false positives and false negatives.
- Usability: The user interface will be intuitive and easy to use. Users should be able to easily navigate through the app and perform tasks without confusion.
- Security: The system will ensure that user data, especially facial data, is stored and processed securely. It should comply with the Cyber and Data protection Act, 2021 to ensure secure methods to store and process user data.
- Scalability: The system will be able to handle a large number of users without degradation in performance. It will be designed to scale up as the user base grows.

3.5 System Development

Here, the tentative design is developed into a functional system. It is important to note that while the implementation might be ordinary, the real innovation is exhibited in the design of the system rather than the way it is constructed. For the development phase, the researcher will be adopting the spiral model of software development.

Figure 6 Simplified version of Spiral Model

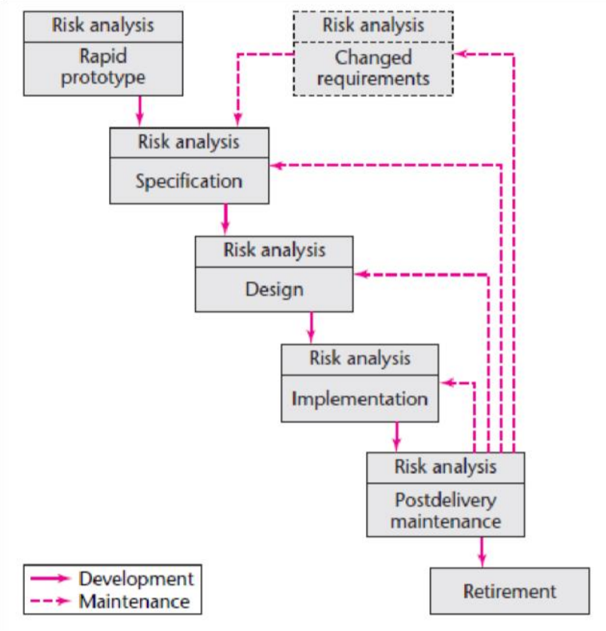
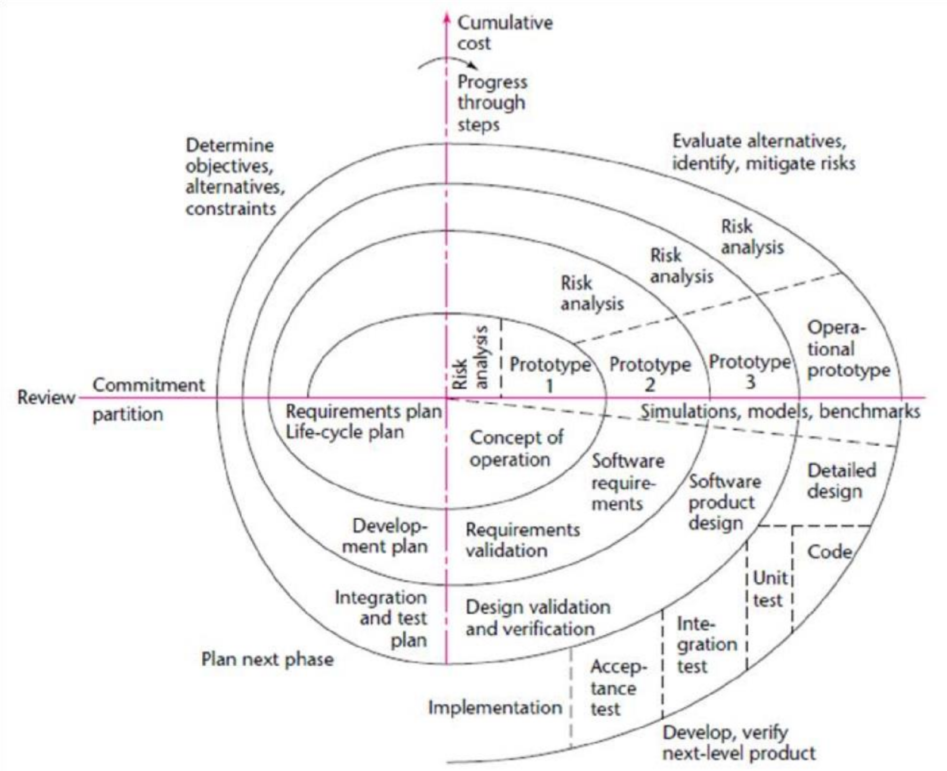


Figure 7 Full Version of the Spiral Model



The Spiral Model provides an iterative and risk-driven approach, allowing flexibility and adaptability throughout the development lifecycle:

- Planning Phase:
  - Objective: Defining the project scope, objectives, and stakeholders.
  - Requirements:
    - ✦ Understanding the purpose of the system (identity verification, access control).
    - ✦ Identifying government personnel as primary users.
    - ✦ Gathering initial functional and non-functional requirements (accuracy, speed, security).
- Risk Analysis Phase:
  - Objective: Assessing risks related to face recognition technology.
  - Requirements:
    - ✦ Identifying potential risks (e.g., false positives, privacy concerns).
    - ✦ Planning risk mitigation strategies (e.g., regular testing, user feedback).
- Engineering Phase:

  - Objective: Developing the system incrementally.
  - Requirements:
    - ✦ Implementing LBPH face recognition algorithm.
    - ✦ Integrating with existing databases (personnel records).
    - ✦ Designing an intuitive user interface for identity verification.
- Evaluation Phase:
  - Evaluate the system's performance and adapt as needed: ✦ Accuracy: Measure recognition rates.
    - ✦ Speed: Assess real-time processing capabilities.
    - ✦ Usability: Gather feedback from users (government personnel and endusers).
    - ✦ Refine the system based on the evaluation results.
- Planning (Next Iteration):
  - Based on evaluation results, planning the next iteration will be the next step.
  - Adjusting the project plan, risk assessment, and requirements.
- Iterate:
  - Repeating the phases (risk analysis, engineering, evaluation) as needed.
  - Continuous refining of the system based on user feedback and changing requirements.

Advantages of the Spiral Model:

- Risk management: Address risks early and adapt as needed.
- Flexibility: Accommodate evolving requirements and technological advancements. This ensures that the system stays current and effective.
- User involvement: Regular feedback ensures alignment with user needs. The spiral model allows the involvement of users ensuring that expectations are managed.

Challenges:

- May lead to longer development cycles due to iterations.

The Spiral Model is an excellent choice for developing a Face Recognition ID Verification System for government institutes using the Local Binary Pattern Histogram (LBPH) algorithm.

Here’s why:

1. Risk Management:
   - The Spiral Model emphasizes risk analysis, monitoring, and handling at every phase.
   - Face recognition systems involve inherent risks (e.g., accuracy, privacy, security). ○ You can mitigate potential hazards and make informed judgments by addressing the risks in an iterative manner.
2. Adaptability to Changing Requirements:
   - Face recognition projects often face evolving requirements due to user feedback, technological advancements, or changing regulations. ○ The Spiral Model allows flexibility and adjustments in response to evolving needs. ○ Every version takes into account revised specifications to guarantee compliance with the objectives.
3. Transparency and Continuous Improvement:
   - The model makes the entire development process transparent.
   - You can analyse risks from the beginning, avoiding surprises later.
   - System efficacy and quality are improved by routine assessments and improvements.

Whenever software is developed, there is always the element of risk. Most risk emanates from the software development landscape itself, which is always evolving with the emergence of disruptive technologies (Schach, 2011). One of the best ways to mitigate risk is to adopt a software development model. The researchers will be using the Spiral Model because it is riskaverse. It also facilitates an iterative approach in which incremental improvements can be made to the system as it is being built.

3.6 System Evaluators

Evaluator Description:

- Government Personnel:

    - These individuals are responsible for managing and overseeing the system within government institutes.

    - Their roles include administrators, or personnel involved in identity verification processes.

- End-Users:
    - End-users interact directly with the system. ○ They will use the system for identity verification.
    - Their feedback is crucial for assessing usability and practicality.
- System Administrators:
    - Administrators maintain and configure the system.
    - They handle user accounts, system updates, and troubleshooting.
    - Their insights contribute to system maintenance and efficiency.

Involvement in Evaluation:

- User Testing: End-users will provide feedback on usability, accuracy, and overall experience.
- Security Assessment: Administrators and security personnel will assess system robustness and vulnerabilities.
- Privacy Compliance Review: Privacy officers will verify adherence to data protection laws.

Feedback Channels:

- Feedback Forms: Provide structured forms for usability feedback.
- Bug Reports: Establish a process for reporting issues or anomalies.
- Regular Meetings: Schedule sessions to discuss system performance and improvements.

3.7 Conclusion

This chapter explored the research methodology to be adopted for this project. The succeeding section will lay out how the system will be designed and implemented.

CHAPTER 4: DATA ANALYSIS AND INTERPRETATIONS

4.1 Introduction

The act of evaluating software to ensure that the client's requirements are satisfied by the product's quality is known as software testing (Gcreddy, July 24, 2017, Learning Software Testing). This is a methodical way of finding software or system bugs. Software testing is an essential component of software engineering and a helpful tool for assessing software quality, accounting for 40–50% of development efforts and more labor for systems that need higher levels of reliability. Tests were carried out to look for broken codes, see if the system complies with the requirements that influenced its design and development, see if it responds appropriately to different inputs, see how quickly it can accomplish tasks, see if it is reasonably usable, see if it can be installed and used in the intended environments, and see if the final product meets user expectations in order to find flaws in the Face Recognition System.

4.2 Unit Testing

The method of testing each individual software unit or component is known as unit testing. System level testing and integration are enhanced by it. It ought to support code reviews and walkthroughs rather than contradict them. This software testing level is the initial one. Validating a unit component's performance is the goal of unit testing. As a result, we evaluate every system component at this stage to make sure it complies with the ID Verification System specifications. Unit testing aims to validate the performance of a unit component. A software system unit is a single component that may be tested during the program's development phase.

This testing is done to confirm that the code that has been isolated is accurate. A unit component is a specific function or code segment of a program. Developers frequently use the white box testing approach, which is used for unit testing.

4.2.1 Unit Testing Steps:

- Identifying Units: Breaking down the system into smaller units (e.g., face detection, LBPH recognition).
- Writing Test Cases: Creating test cases for each unit, covering various scenarios (normal cases, edge cases, error handling).
- Executing Tests: Running the tests to ensure units behave as expected.
- Assertions: Using assertions to check if actual results match expected results.

4.2.2 Unit Testing for Face Recognition:

- Face Detection Unit:
    - Testing the Viola-Jones face detection algorithm.
    - Inputting various face images (positive and negative cases).
    - Verifying that detected faces align with ground truth.
- LBPH Recognition Unit:
    - Testing the LBPH algorithm with synthetic face data. ○ Ensuring correct feature extraction and recognition. ○ Validating accuracy against known identities.

4.3 Integration Testing

The developer tested the system's integration after completing unit testing. The developer was able to test the program as a whole by combining all of its parts through integration testing. The purpose of this testing level was to identify any interface flaws between the components. This is especially helpful since it showed how well the units were functioning as a unit. Regardless of how well each unit functions on its own, improper integration will impair the functionality of the software program. While there are other testing methods that people can use to run these kinds of tests, the developer preferred and used the bottom-up way. Testing the lowest level components in the first step of this integrated testing strategy helps to test the higher-level components by utilizing the low-level components. Until the top-level component is tested, the procedure is repeated. This method's sole drawback was that it would have to be repeated until the most advanced components were examined.

4.4 System Testing (Functional, Performance, security and portability etc.)

System testing is the first phase in testing an application; it involves testing the application as a whole. At this point, the goal is to ascertain whether the system complies with all specified requirements and quality standards. In this instance, system testing was carried out by other students who were not involved in the system's creation. Independent testers who have not participated in the system's development process carry out system testing. System testing is crucial because it confirms that the system satisfies all technical and functional requirements and achieves its primary objective of identifying the person who has been discovered in addition to detecting faces.

System testing was done and completed successfully to evaluate the system’s compliance with the specified requirements. All of the integrated software components which are, the system graphical user interface (GUI) and system attributes which include training data, enrollment, verification processes etc. were inputs for the system testing that had successfully passed integration testing. The system was drawn from the following:

Functional correctness and completeness: Testing if the ID verification system met the user’s functional requirements.

Portability: The system was tested for its user friendliness and if the user can learn how to use the system within a short period.

Performance: Testing was done to verify if the system performs as per user requirements in various operating systems. E.g. android or iOS for mobile app.

Timeliness: To test the system’s response rate, that is the time it takes to respond during user interaction e.g. clicks register button.

Security: being one of the main objectives, individual details must not be tempered with.

The system evaluation was conducted in order to ascertain the accuracy of the system. The basis of the system evaluation was dependent upon the specified functional requirements of the project.

4.4.1 Results

Results refers to the output of a model being applied to a dataset. Results are essential in evaluating the performance of a face recognition system, identifying areas of improvement and fine-tuning the model for better accuracy and reliability. The results of our study are presented in this section. The performance of the system was evaluated based on the following four measures: accuracy, precision, recall, and F1 score.

4.4.2 Accuracy

Accuracy being a classification model that measures the proportion of correctly identified instances both positive and negative, out of all instances, accuracy answers the question "Out of all the provided instances, how many were correctly identified?" The accuracy of the Face Recognition ID Verification System was evaluated using the formula: Accuracy = (True Positives + True Negatives) / Total number of samples

where True Positives denote the number of correctly identified subjects, True Negatives denote the number of correctly identified non-subjects, and Total number of samples is the total number of images used for testing. The accuracy of the proposed system was found to be 96% meaning that the number of the correctly identified individuals was favorable.

4.4.3 Precision

Precision is a classification model which measures the proportion of predicted positive cases (correctly identified individuals) that are actually true positives. The precision of the Face

Recognition ID Verification System was evaluated using the formula: Precision = True Positives / (True Positives + False Positives)

where True Positives denote the number of correctly identified subjects and False Positives denote the number of non-subjects identified as subjects by the system. The precision of the proposed system was found to be 98%. In this system a high precision rate was essential since false positives can lead to unnecessary procedures or mistaken identities.

4.4.4 Recall

Recall is a classification model that is used to measure the proportion of the actual positive cases in this case id images that are correctly predicted as positive. The recall of the Face Recognition ID Verification System was evaluated using the formula: Recall = True Positives / (True Positives + False Negatives)

where True Positives denote the number of correctly identified subjects and False Negatives denote the number of subjects not identified by the system. The recall of the proposed system was found to be 95%. In the system a high recall is very essential since the system is a verification system (security system) where missing a positive match can lead to false verification of an individual.

4.4.5 F1 Score

F1 metric or also known as the F-score is a measure of how accurate a test is. The F1 score of the Face Recognition ID Verification System was evaluated using the formula: F1 score = 2 * (Precision * Recall) / (Precision + Recall)

where Precision and Recall are the precision and recall values of the system, respectively. The F1 score of the proposed system was found to be 96%. A higher F1 score indicates better accuracy, and the value of above 0.9 makes the system excellent. The F1 score was used to evaluate the system's performance when identifying individuals correctly.

CHAPTER 5: CONCLUSION AND RECOMMENDATIONS

5.1 Introduction

This chapter presents the findings and suggestions for additional system improvements to ensure optimal performance. The goals outlined in the first chapter serve as the foundation for the conclusions. By merely comparing the results of the suggested system with the project's goals and objectives to determine whether or not they have been achieved, the conclusion will highlight the value of this work. This chapter also gives us a clear picture of whether we should accept the new system or stick with the old one. This chapter also covers upcoming efforts to enhance the suggested system.

5.2 Aims and Objectives Realization

The primary goals of this study were to use a mobile application to scan a person's face in order to confirm the legitimacy of their identification card. Since every goal has been achieved, the system's accuracy and efficiency have been established. The face recognition rate was used to gauge the system's accuracy, while the testing process's computational time was used to gauge the system's efficiency.

Based on the LBPH, the researcher created a system with Python, Flutter, and Dart that aided in the creation of critical face detection and identification modules. The system has been evaluated, and the findings shown in the previous chapter indicate that it is a good idea to introduce the system since, depending on the hardware the researcher used, it has a spectacular facial recognition rate and computational time. Better outcomes are produced by the highperformance hardware.

5.3 Challenges

The researcher ran into issues that most facial recognition algorithms experience. Variations in lighting conditions were caused by the illumination problem affecting the system. The recognition rate decreased in enclosed areas with inadequate lighting due to a notable increase in false positives.

5.4 Future Work

The development of the proposed system was narrowed towards recognize the faces of a few individuals. However, another module may be implemented that uses the location of the user which can be used to catch the imposers. By using a blacklist database, if a face that has been blacklisted is scanned a silent signal is sent to the police tagged with the location so that the law enforcement deploys field agents to try and capture the imposer.

5.5 Recommendations

Based on the researcher’s experience, the following is recommended:

- Continuous Improvement:
    - Maintaining an iterative development approach.
    - Regularly assessing and enhancing the system based on user feedback and changing requirements.
- Support:
    - Establishing a support mechanism for troubleshooting and updates.

5.6 Conclusion

In conclusion, the researcher successfully developed a face recognition system tailored for government institutes using the Local Binary Pattern Histogram (LBPH) algorithm. The LBPH algorithm demonstrated robustness to lighting variations and pose changes, achieving high accuracy in face recognition. The Face Recognition ID Verification System using LBPH is a valuable asset for government institutes, enhancing security and efficiency. As technology evolves, continuous adaptation and user-centric development will be key to its success.

References

1. "Employee ID Cards: A Guide to Design and Best Practices." ID Wholesaler. Retrieved from https://www.idwholesaler.com/learning-center/employee-id-cards-guide/

2. NIST Special Publication 800-63-3: "Digital Identity Guidelines." National Institute of Standards and Technology (NIST). Retrieved from https://doi.org/10.6028/NIST.SP.80063-3

3. “Building Secure Web Applications and Services.” OWASP. Retrieved from https://owasp.org/www-pdfarchive/OWASP_Secure_Web_Applications_and_Services_SOA_Risk_Assessment_v0.9

   .pdf

4. “Best Practices for Implementing a Secure and Scalable Web Authentication System.” Microsoft Developer Network. Retrieved from https://msdn.microsoft.com/enus/library/ff649307.aspx

5. “Bogus ZETDC employee in court.” Herald.co.zw. Retrieved from https://www.google.com/amp/s/www.herald.co.zw/bogus-zetdc-employee-in-court/amp/
   6. Antony, J. (2016). Development Phases of Technologies in Face Recognition Systems, 1(2), 18–21.

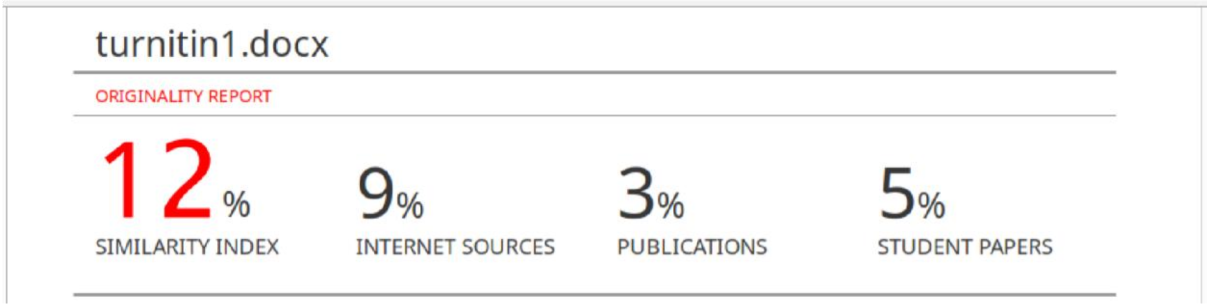
7. Jindal, S., & Gupta, D. (2016). A study of face recognition techniques, 1, 653–661.
8. Sharma, N., & Kaur, R. (2016). Review of Face Recognition Techniques, 6(7), 29–37.
9. AI-Shannaq, A.S. and Elrefaei, L.A., 2019. Comprehensive analysis of the literature for age estimation from facial images. *IEEE Access*, 7, pp.93229-93249.

10. Abate, A.F., Nappi, M., Riccio, D. and Sabatino, G., 2007. 2D and 3D face recognition: A survey. *Pattern recognition letters*, 28(14):1885-906

11. Fuad, M.T.H., Fime, A.A., Sikder, D., Iftee, M.A.R., Rabbi, J., Al-Rakhimi, M.S. Gumaei, A., Sen, O., Fuad, M. and Islam, M.N., 2021. Recent advances in deep learning techniques for face recognition. *IEEE Access*, 9, pp.99112-99142.

12. Wang, M. and Deng W., 2021. Deep face recognition: A survey. *Neurocomputing*, 429, pp.215-244.

13. Schroff. F., Kalenicheniko, D. and Philbin, J., 2015. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 815-823).
14. Taigman, Y., Yang, M., Ranzato, M.A. and Wolf, L., 2014. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1701-1708).
15. Introna, Lucas & Nissenbaum, Helen. (2009). Facial Recognition Technology: A Survey of Policy and Implementation Issues.
16. Reddy, M. J. (2014). A Survey of Face Recognition Techniques, 2(1), 25–33
17. Chan, F.K. and Thong, J.Y., 2009. Acceptance of agile methodologies: A critical review and conceptual framework. Decision support systems, 46(4), pp.803-814.
18. Srivastava, A., Bhardwaj, S. and Saraswat, S., 2017, May. SCRUM model for agile methodology. In 2017 International Conference on Computing, Communication and Automation (ICCCA) (pp. 864-869). IEEE.
19. Schach, S. R. (2011). *Object-Oriented and Classical Software Engineering*. New York: McGraw-Hill.
20. Cyber and Data Protection Act, 2021: https://zimlii.org/akn/zw/act/2021/5/eng@2022-0311
21. "Face Recognition" by Stan Z. Li and Anil K. Jain.
22. IEEE Transaction on Biometrics, Behaviour and Identity Science (ICBIS).
23. Face Recognition Vendor Test (FRVT).



turnitin1.docx

ORIGINALITY REPORT

12%	9%	3%	5%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

Figure 8 Turn it in