FACE RECOGNITION BASED DOOR LOCKING WITH TWO-FACTOR AUTHENTICATION USING OPENCV

MUHAMMAD ARIF AZHARI BIN HALIM

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

FACE RECOGNITION BASED DOOR LOCKING WITH TWO-FACTOR AUTHENTICATION USING OPENCV

MUHAMMAD ARIF AZHARI BIN HALIM

This report is submitted in partial fulfillment of the requirements for the Bachelor of Computer Science (Computer Security) with Honours.

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2021

DECLARATION

I hereby declare that this project report entitled

FACE RECOGNITION BASED DOOR LOCKING WITH TWO-FACTOR AUTHENTICATION USING OPENCV

is written by me and is my own effort and that no part has been plagiarized without citations.

STUDENT : _____ Date : 8/9/2021
([MUHAMMAD ARIF AZHARI BIN HALIM])

I hereby declare that I have read this project report and found this project report is sufficient in term of the scope and quality for the award of Bachelor of Computer Science (Computer Security) with Honours.

SUPERVISOR : _____ Date : 8/9/2021
([TS. DR MOHD FAIRUZ ISKANDAR BIN OTHMAN])

DEDICATION

Special thank you to my family especially my father, Mr. Halim Bin Dokoh and my mother, Mrs Salmi binti Othman for always given me great support, encouragement and advice. Also thanks to all my friends that always understand and give moral support. For my supervisor, Ts. Dr. Mohd Fairuz Iskandar Bin Othman who provided a lot of guidance and encouragement throughout this project, thank you for all your hard work to guide me through this project. Lastly, thanks to my fellow friends that assisted to complete this project and report.

ACKNOWLEDGEMENTS

First of all, I would like to say thank you very much to our Almighty God, Allah, who has always been with me when I was happy and struggling and for giving me a functioning body, strength and a functioning intellect with which to enjoy life and learn, and especially working and finish this project.

I would like to express my heartfelt gratitude to my supervisor, Ts. Dr. Mohd Fairuz Iskandar Othman for his generous support to lead me during the preparation and growth of this project. His willingness to give his time so generously has been greatly appreciated and given a ton of guidance and encouragement to complete this project.

I would also like to express my gratitude to my precious family and friends for their continued support, encouragement and love for me, which keep me inspired and dedicated during this work. Also being there for me since the beginning of my degree programme and assisting me during my journey through study.

Finally, I would like to thank the University of Technology Malaysia Melaka (UTeM) which provides a platform for the development and improvement of my computer science (computer security) skills and not to forget the Faculty of Information Technology (FTMK) which has enhanced the talent of its students with its facilities.

ABSTRACT

This project develops a face recognition-based door locking system with two-factor authentication using OpenCV. It uses Raspberry PI 4 as the microcontroller. Face recognition-based door locking has been around for many years, but most of them are just face recognition without any added security features, and they are costly. The design of this project is based on human face recognition and the sending of a One-Time Password (OTP) using the Twilio service. It will recognize the person that goes to the front of the door. Only people who match the faces stored in the dataset and inputs the correct OTP will have access to unlock the door. Twilio service and image processing algorithm Local Binary Pattern Histogram (LBPH) have been adopted for use in this system. Servo motor operates as a mechanism to access the door. LBPH take a short time to recognize the face. Additionally, if an unknown face is detected, it will log this instance into a "Fail" file and accompanying CSV sheet.

ABSTRAK

Projek ini mengembangkan sistem penguncian pintu berasaskan pengecaman wajah dengan pengesahan dua faktor menggunakan OpenCV. Ia menggunakan Raspberry PI 4 sebagai pengawal mikro. Penguncian pintu berdasarkan pengecaman wajah telah wujud selama bertahun-tahun, tetapi kebanyakannya hanya pengecaman wajah tanpa ciri keselamatan tambahan, dan harganya mahal. Reka bentuk projek ini berdasarkan pengenalan wajah manusia dan pengiriman Kata Laluan Sekali-Waktu (OTP) menggunakan perkhidmatan Twilio. Ia akan mengenali orang yang menuju ke depan pintu. Hanya orang yang sepadan dengan wajah yang disimpan dalam set data dan memasukkan OTP yang betul akan mempunyai akses untuk membuka kunci pintu. Perkhidmatan Twilio dan algoritma pemprosesan gambar Histogram Pola Binari Tempatan (LBPH) telah diadopsi untuk digunakan dalam sistem ini. Servo motor berfungsi sebagai mekanisme untuk mengakses pintu. LBPH mengambil masa yang singkat untuk mengenali wajah. Selain itu, jika wajah yang tidak dikenali dikesan, ia akan memasukkan contoh ini ke dalam fail "Fail" dan lembaran CSV yang disertakan.

TABLE OF CONTENTS

PAGE

LIST OF TABLES

PAGE

LIST OF FIGURES

PAGE

CHAPTER 1: INTRODUCTION

1.1 Project background

The development of Information Technology (IT) has led the rapid change in the human lifestyle. Modern advances in electronic and communication technology have made the application of computers, networking, and mobile devices implemented pervasively in our daily lives. These changes have catalyzed the development of the Internet of Things (IoT) which includes smart home technologies. IoT-based face recognition has been used in a variety of applications such as unmanned aerial vehicles, smart classrooms, home security, smart houses, smart surveillance and many more.

Nowadays, a home protected with only a door lock is not enough to protect the house and family. Previously, people relied on the traditional method to protect their homes. To get access, the typical security system relies on the usage of external items such as a key, password, or ID card. Biometrics was then introduced as a promising security solution. For individual recognition, biometrics is a unique and quantifiable parameter. Biometric technology is regarded as one of the most secure verification systems available, providing a higher level of security than traditional methods. Apart from fingerprint characteristics, face recognition is the most well-known biometric technology method. This is due to the fact that the face has more characteristics and features, which gives it more solidity. Furthermore, it is regarded as particularly secure because the face cannot be stolen, borrowed or forged to enter a house. Face recognition technology has piqued many scientists' interest, and it is increasingly displacing existing biometric security frameworks. Image matching is another term for face recognition. It's a fast-growing field that's on the verge of replacing the old ways. Face recognition is a more stable biometric identification technology than others because it uses the human face, which results in high accuracy, low false recognition rate, and minor changes in people's lives. Face recognition is a technique that can

identify and verify people. According to (Syafeeza, Khalil-Hani, Liew & Bakhteri, 2014), face recognition is defined as the process of identifying, distinguishing, and processing a person's face before comparing it to photographs kept in a database to authenticate who they are. Face recognition has grown in importance as a method of identifying users.

In addition, this project is to develop secure door locking using face recognition by implemented two-factor authentication using raspberry PI to make it easier without depending on the use of keys.

1.2 Problem Statements

The problem with the face recognition based door locking is the system is not secured enough to protect the household. People can bypass the face recognition by using picture or print of the face of the owner of the house and mostly face recognition system is very expensive in the market. As a result, the system must be improved in order to improve security and efficiency. The problem statements were encapsulated in Table 1.1

Table 1.1: Summary of Problem Statement

PS	Problem Statement
PS_1	Because of its high price, the face recognition system on the market is still unpopular for domestic use
PS_2	Owner being careless by accidentally lost the key and the home security with only door locker is not enough to protect the household
PS_3	Face recognition can be bypass without presence of the owner by using picture or print of the owner's face.

1.3 Project Question

The existing system will need to evolve in order to address the shortcomings of the previous system, as described in Table 1.1. The suggested technology will eliminate the use of keys to gain access to the house. The questions in implementing face recognition-based door locking are shown in Table 1.2.

Table 1.2: Project Question

PQ	Project Question
PQ_1	How much the face recognition security system costs?
PQ_2	Is depending on the keys is old method and unsecure?
PQ_3	How secure is the face recognition based door locking?

1.4 Project Objective

There are three objectives of this project. Table 1.3 shows the summary of the project objectives for this project

Table 1.3: Project Objective

PO	Project Objective
PO_1	To design and develop a face recognition system which is high efficiency means that without wasting materials, energy, low cost and easy to operate.
PO_2	To make a system a keyless or without depending on the keys.
PO_3	To develop a secured door locking with two-factor authentication.

1.5 Project Scope

The stage of project preparation during which basic project priorities, deliverables, assignments, costs, and deadlines are set and recorded is known as project scope. The project scope would define the intended user, as well as the hardware and software that will be used in this project.

1.5.1 Users Scope

The owner or residents of the house who require a secure face recognition based door locking system with two-factor authentication are the target users of this project. The user can input their faces into database and recognize it when in front of the door to unlock the door.

1.5.2 System Scope

1.5.2.1 Hardware

i) Raspberry PI 4

Raspberry PI is a credit-card sized computer manufactured and designed in the United Kingdom by the Raspberry PI foundation with the intention of teaching basic computer science to school students and every other person interested in computer hardware programming and DIY-Do-it Yourself projects.

ii) USB Webcam

A simple USB powered webcam is used to recognize people faces.

iii) Servo Motor

A servo motor is a self-contained electrical device, that rotate parts of a machine with high efficiency and with great precision.

iv) Jumper Wires

An electrical wire (or a collection of them in a cable) with a connector or pin at either end (or occasionally without – simply "tinned") that is typically used to link the components of a breadboard or other prototype.

1.5.2.2 Software

i) Raspbian Operating System (OS)

Raspberry Pi OS has been specifically designed for the Raspberry Pi range of small single-board computers powered by ARM CPUs. Raspberry Pi OS's desktop environment is a modified LXDE with the Openbox stacking window manager and a unique look.

ii) Thonny Python IDE

Thonny is an integrated development environment for Python that is designed for beginners. It supports different ways of stepping through the code, step-by-step expression evaluation, detailed visualization of the call stack and a mode for explaining the concepts of references and heap

iii) VNC

Virtual Network Computing (VNC) is a graphical desktop-sharing system that uses the Remote Frame Buffer protocol to remotely control another computer. It transmits the keyboard and mouse input from one computer to another, relaying the graphical-screen updates, over a network

1.6 Project Contribution

The project contribution describes the expected performance of this project. This section is sometimes referred to as the project's goal. Table 1.4 shows the summary of project contribution

Table 1.4: Summary of the Project Contribution

PC	Project Contribution
PC_1	To capture image of the use and store into dataset
PC_2	Sending an OTP to the user’s number phone when recognized face

1.7 Report Organisation

This section is provided for the description of the report organization. Overall the report consists of seven (7) chapters:

Chapter 1: Introduction

This chapter consists of the project background, problem statement, project question, project objective, project scope, and project contribution.

Chapter 2: Literature Review

In this chapter presents a research about face recognition based door lock and related work

Chapter 3: Project Methodology

Chapter 4: Analysis and Design

Chapter 5: Implementation

Chapter 6: Testing

Chapter 7: Project Conclusion

1.8 Conclusion

In conclusion, this chapter has given an explanation and a better understanding on the objectives of the project, regarding how it would benefit in the Cyber Security field in the future. Next chapter will be focusing on finding of face recognition based door locking. The discussion about related works will be addressed also in the next chapter.

2. LITERATURE REVIEW

2.1 Introduction

According to Jain, A.K and Li, S.Z (2011), the last few decades, face recognition has become one of the most important applications of biometrics-based authentication systems. Face recognition is a type of recognition task pattern in which a face is classified as known or unknown after being compared to photos of a known person in a dataset. In addition, face recognition is difficult due to the diversity of information generated by random variation across persons, as well as systematic variances produced by a variety of factors such as lighting and attitude.

Face recognition technology can be used for a variety of purposes, including law enforcement and inspections, access control, information security, smart cards, and more. It has a wide range of applications, including surveillance, credit cards, password security, and so on.

2.2 Related Work

There are various project efforts pertaining to facial recognition in security systems, according to the research findings. Author (Sahani, Nanda, Sahu & Pattnaik, 2015) wrote a research paper titled "Web-based online embedded door access control and home security system based on facial recognition." After analysing their product, they can determine its strengths and weaknesses. Their product's strength is that it is built on the ZigBee wireless network technology. To operate the door accessibility, the ZigBee module is combined with an electromagnetic door lock module. The suggested system is built with wireless access control in mind, thus the lock module may be readily added if necessary. When a stranger face is spotted, the residence owner is notified by email and SMS.

However, it may still be improved upon in terms of its flaws. With a snapshot or print of the owner's face, facial recognition can be bypassed. The system can be improving with add on password authentication, face recognition or fingerprint authentication.

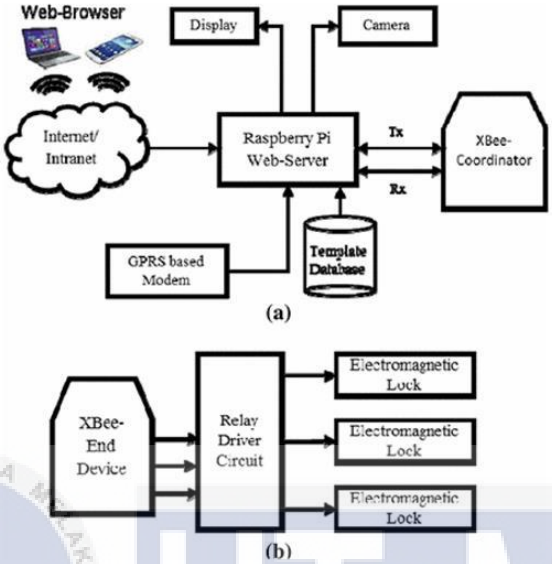


Figure 2.1 Outline of proposed system configuration layout; (a) wireless information unit, (b) wireless control unit

Authors (Hteik Htar Lwin, Aung Soe Khaing & Hla Myo Tun), proposed a system that is done by MATLAB program on personal computer (PC) for automatic face recognition and detection. The door access system is controlled by a microcontroller based on data provided from a PC. After establishing that the individual is authenticated, the door is promptly opened. The door automatically closes after 2 seconds. 2 seconds, on the other hand, is not enough time for a person to enter. So, need to set real-time conditions longer than 2 seconds.

This system used Viola Jones Algorithm for face detection. This system has limitations since this algorithm can only detect frontal view of the face correctly and it is not very secured because it is likely to be easy to bypassed and easy to open the door.

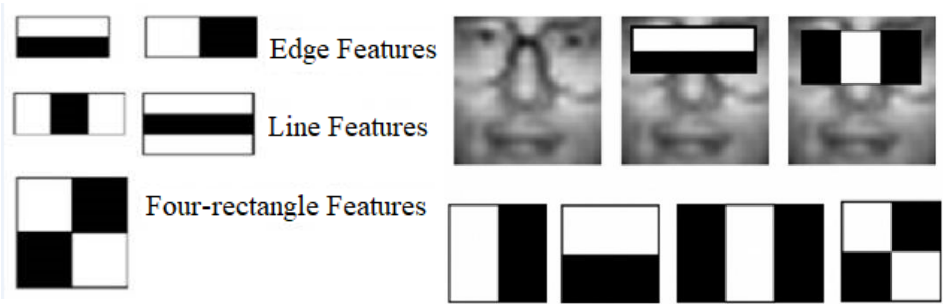


Figure 2.2 Viola Jones Algorithm method for face detection

The authors (Patil, Ranavare, Ballal & Kotekar) and (Yedulapuram, Arabelli, Mahender & Sidhardha, 2020) developed a system that uses face recognition on a Raspberry Pi board and sends an alert to the authorized person via a GSM module. But there is no authentication function because face recognition can be bypassed, as mentioned previously above. However, it can be improved by implementing a two-factor authentication such as a password or a one-time password (OTP).

The author (Khan, 2021) proposed a "Face Recognition Door Lock using Raspberry Pi with AWS Recognition", with all face recognition parts being performed on an AWS cloud by using Raspberry Pi. The author (Saputra & Surantha, 2021) developed a "Smart and real-time door lock system for an elderly user based on face recognition" that detects movement using a PIR sensor for face recognition to unlock the door. Meanwhile, the author (Prasade, Nalavade & Pathak, 2018) developed a "Face Recognition Based Door Locking System" that uses a PIR sensor to detect motion where the buzzer will start beeping and send a notification if the user is not recognized.

From the description above, it is clear that the previous solutions presented by the researchers before this lack an additional authentication mechanism to make the system more secure. Therefore, this project will fill the gap by implementing two-factor authentication by sending a one-time password (OTP) by using the Twilio service to increase levels of security.

2.2.1 Face recognition technology (Domain)

Identity fraud and theft are growing more widespread, and they have become severe issues. Traditional methods of personal identification require an external element, such as a key, security password, RFID card, or ID card, to get access to a private item or enter public area. The usage of a password is required for several processes, such as withdrawing money from banks. Any other parking in a private area would necessitate a parking penalty. In some households, the house key is quite important. This technique, however, has a number of flaws, including the possibility of losing the key or forgetting the password. It may be tough to get back on track if this occurs. Biometric methods are gradually supplanting this tactic as a possible answer to such difficulties. This technique required the use of special hardware such as fingerprint scanners, palm print scanners, and DNA analyzers to acquire information for the vast majority of biometric applications, and the target objects had to touch the hardware to obtain information.

Face recognition technology has recently piqued the interest of a large number of researchers, and it is gradually displacing traditional biometric security frameworks. Image matching is another term for face recognition. It's a rapidly evolving field that's on its way to supplanting the old method. Face recognition, unlike other biometric identification methods, is more stable since it uses the human face, which results in high accuracy, low false recognition rate, and no change in people's lives. As a result, this technology is quite useful for a variety of applications, such as unlocking a house door using face recognition.

Table 2.1 Biometric technology comparisons

Method	Accuracy	Cost	Equipment	Acceptability
Face	M	L	Camera	M
Hand	M-L	M	Scanner	M
Fingerprint	H	M	Scanner	M

Retina	H	H	Camera	L
DNA	H	H	Test Equip.	L
Voice	M	M	Microphone	H
Signature	L	M	Optic Pen	H
Password	H	L	Keypad	L

2.2.2 Cascade Classifier

According to S. Soo (2014), the cascade classifier is made up of a series of stages, each of which has a list of weak learners. By sliding a window over the image, the system recognizes the things in question. Each stage of the classifier identifies the precise region indicated by the current location of the window as positive or negative, with positive indicating the presence of an object and negative indicating the absence of the specified object in the image

If the labelling yields a negative result, then the classification of this specific region is hereby complete and the location of the window is moved to the next location. If the labelling gives a positive result, then the region moves of to the next stage of classification.

When all of the phases, including the last one, give a result, the classifier returns a positive judgement, indicating that the object is found in the image. A true positive indicates that the object in question is present in the image and has been labelled as such by the classifier - a positive result. A false positive occurs when the labelling process incorrectly determines that the object is there in the image when it is not. A false negative occurs when the classifier is unable to detect the actual object from the image, whereas a true negative occurs when the classifier properly classifies a non-object as not being the object in question.

Each level of the cascade must have a low false negative rate in order to function properly, because if the actual object is classified as a non-object, the classification of that branch stops, with no way to fix the error. However, each level can have a reasonably high false positive rate since, even if the n-th stage incorrectly classifies the non-object as the object, this error can be corrected in the n+1-th and following stages of the classifier.

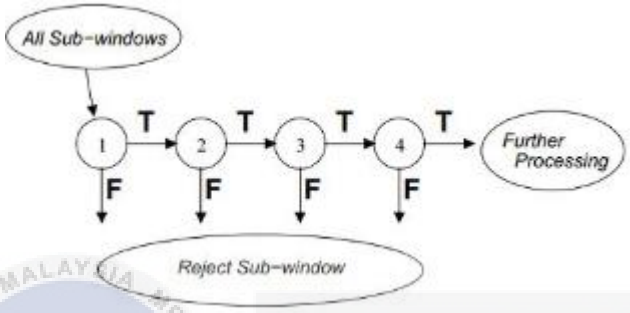


Figure 2.3 Stages of the cascade classifier

2.2.3 Introduction of Haar-like features

According to V. Jones (2001), a Haar-like feature evaluates neighbouring rectangular sections in a detection window at a specified point, sums the pixel intensities in each region, and calculates the difference between these sums. This distinction is then utilised to categorise image subsections.

The detection of human faces is an example of this. The areas surrounding the eyes are usually darker than the portions on the cheeks. A set of two adjacent rectangular sections above the eye and cheek regions, for example, is an example of a Haar-like feature for face detection.

2.3 Critical review of current problem and justification

Figure 2.4 Door lock using keypad password and RFID

A keypad passcode controls the mechanism knob on the older security door lock system. The user must be able to remember the password. The downsides of this method are that it is difficult for older and younger people to remember the password combination, and with today's technology, a burglar can easily crack it. Another device

that has been developed is RFID cards, use to identify user identity easily. It works like a key but in the shape of a card, and can unlock a door in a fraction of a second. With today's technology, it's also simple for others to make a clone of your RFID card.

Figure 2.5 Door lock using fingerprint

The use of fingerprints to verify a person's identity is also common. As a result, it's been used as a security door lock as well. Everyone, with the exception of a few disabled people and a small fraction of the world's population, has their own unique fingerprint. Nonetheless, a fingerprint cannot be used to identify a broader group of people.

2.4 Proposed Solution

The proposed project system's approach is discovered by studying the fact finding of current problem. If the user's concern continues, the Face Recognition Based Door Locking with 2FA is recommended for implementation. This project uses Raspberry PI as the microcontroller and will be implemented or install at the front door to recognize the owner. Twilio service is used to send a One Time Password (OTP) to the owner to verify the authentication. If the faces are not recognized, it will be log into Fail file and csv sheet.

2.5 Conclusion

In conclusion, a literature study is a compulsory chapter and a key component in developing a project concept. It assists in identifying the system's existing capabilities as well as receiving particular information about how to use the system. The research and analysis would make the improvement process more straightforward and feasible. In the next chapter, project methodology, how it would be carried out and the project milestones will be discussing.

3. PROJECT METHADOLOGY

3.1 Introduction

This chapter focuses on project creation. When someone approaches the door and turns to face the camera, the faces of that person are captured from the camera and saved in the system. Following that, the retrieved face is pre-processed. The face is also matched against an existing dataset. If he/she is a known user, the system will send a OTP to the phone number of the user that stored in the database and will verify the OTP either correct or not to open the door.

3.2 Methodology

The methodology that will be used in the project is in stages, with phases more commonly known as The Waterfall Model. This approach requires implementation in the order in which it is necessary to complete one phase that will move on to the next phase. Here's a description of The Waterfall Model.

This approach is chosen because every development is easy to measure because the development of the system is done from one level to another. In addition, each level will state and record the progress of the process on the system. The following below is an example of The Waterfall Model that will be used throughout the course of this project.

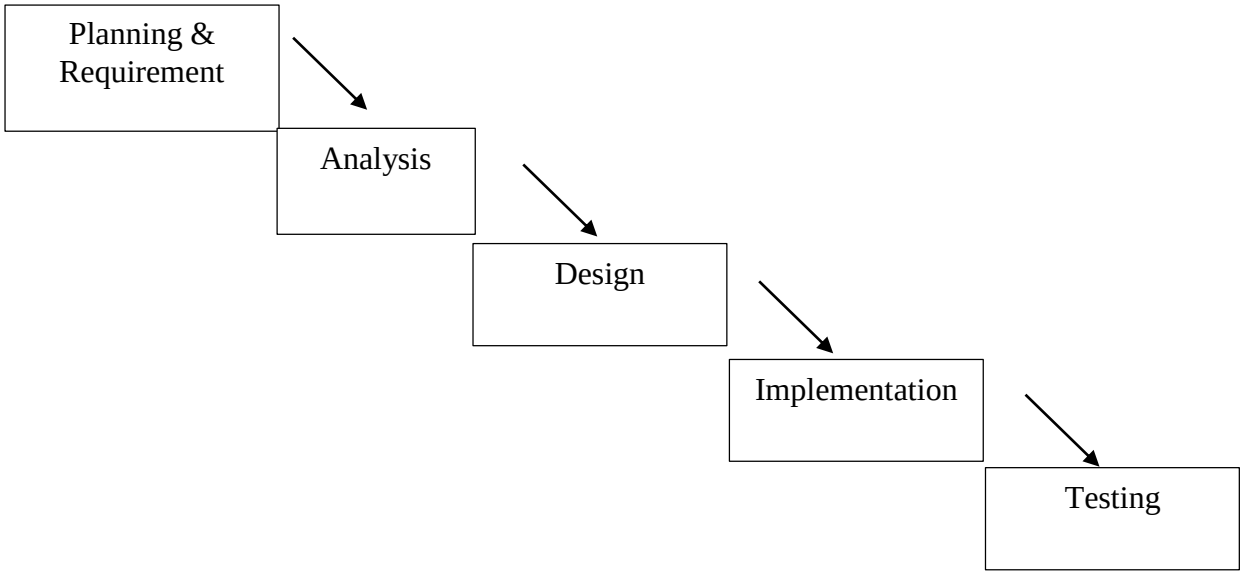


Figure 3.1 Waterfall model

3.2.1 Storing user data

About a 70 photos were captured to obtain facial data of one user. These photographs have also been cropped to a suitable size, with just significant aspects such as the forehead, chin, and eyes being used as an example. For the sampling process, all background details that aren't a legitimate face have been removed.

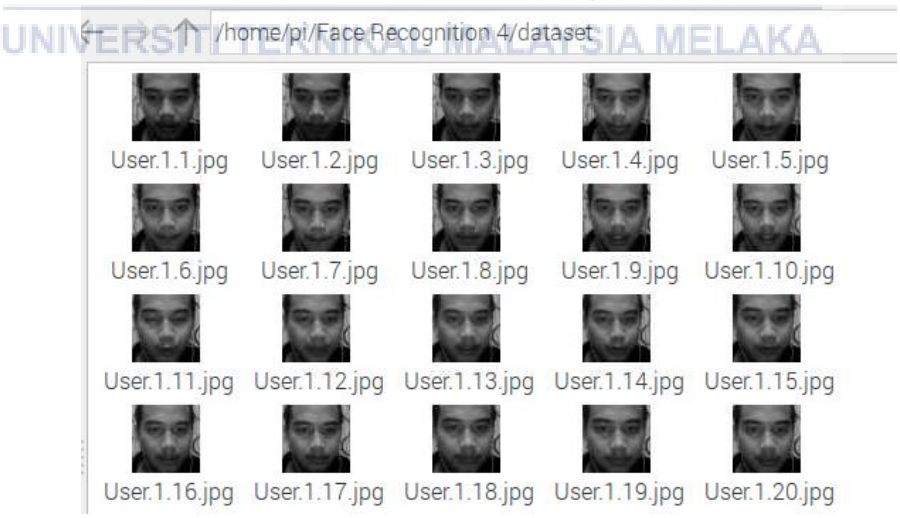


Figure 3.2 User dataset

3.2.2 Training the user's face that stored in dataset

In order to make raspberry PI learn user's face, LBPH Face Recognizer is used. LBPH is a popular face recognition algorithm that identifies pixels in an image by generating threshold values for neighbouring pixels and producing a binary number as a result. We take a 3X3 matrix from the top left corner of each user photograph. We classify each pixel value by using the centre pixel of the matrix as the threshold. The new pixel value of the centre pixel of the 3X3 matrix is obtained by converting this new binary value to decimal. This matrix-generation method is done to the entire image. After that, a histogram is created for each image using the pixel values generated before. Each histogram reflects the characteristics of a single image.

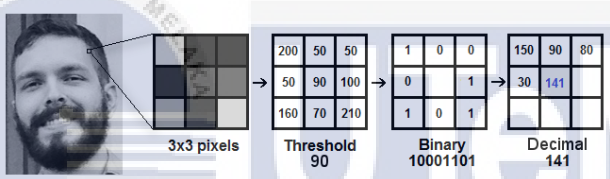


Figure 3.3 Conversion of grayscale image to decimal

3.2.3 Face detection from video input

In order to capture photo frame of user from a live video, we need to used USB Webcam. After getting a live video input from USB Webcam, haar cascade frontal face classifier is used to extract the region of interest from the video stream. Haar Cascade classifier is an effective object detection approach which was proposed by Paul Viola and Michael Jones in their paper, "Rapid Object Detection using a Boosted Cascade of Simple Features" in 2001.

3.2.4 Face Recognition

Following the training of the LBPH algorithm with real faces, the trained model is now given a new unknown image as input for recognition. This new face is the result of the previous stage of collecting faces from a video feed. The same LBPH algorithm is used to create a histogram for this fresh image. Now, we compare the histograms of the trained faces model and the histogram for a new unknown face to see if they match. For comparison, we'll use the Euclidean distance formula.

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

Figure 3.4 Euclidean distance

We can get a confidence value from the Euclidean distance. The lower the confidence number, the more likely it is that the user is genuine. It's because the confidence values reveal how closely the two histograms are related.

3.2.5 Sent OTP and give permission to unlock the door

We have set the confidence value to ___ for this system. If the confidence value is more than that, the system will send an OTP to the mobile number of the know user's face and if the user input the correct OTP, it will trigger the general purpose input output (GPIO) pin of raspberry PI to the relay module and trigger the servo motor to open it.

3.2.6 Updating dataset of known faces

We will regularly update the data set acquired for the user in order to accommodate subsequent changes in the user's face, such as growing a beard or wearing eye glasses. Using time as a limitation, we'll collect several fresh faces of the user throughout time in order to deal with change.

3.3 Project Milestone

A project milestone is a reference point that may be used to track the project's progress and signifies the project's major activity. Project milestones can be developed and effectively coordinated to guarantee that all project operations are completed within the project timetable, ensuring that the project runs smoothly.

3.3.1 Flowchart

A flowchart is a type of diagram representing an algorithm, workflow or process, showing the steps as boxes of different types, and their order by connecting them with arrows. This diagrammatic representation illustrates a model of solution to a given issue. Flowcharts are used in the analysis, design, documentation or management of various fields of a method or system. Below is represent the overall Flow Cart for this project from beginning to the end:

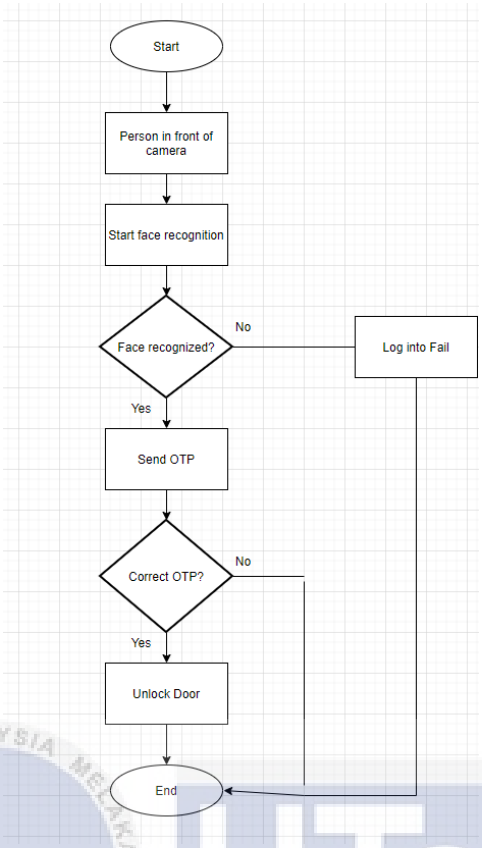


Figure 3.5 Flowchart design

3.3.2 Gantt Chart

Gantt chart is a timeline has been set for each activity being implemented that can be found in all the project. If an activity contained in the project is late, it will affect the lasting future and the set costs to increase over the rest of the activities involved in this project.

Table 3.1 Gantt Chart

Activities	Time Period by Week														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Proposal															
Planning & Requirement															
Analysis															
Design															
Implementation															
Testing															
Documentation															

3.4 Conclusion

To conclude, methodology plays a significant part in assessing a project's progress. It will be implemented in the event of a project that does not have a methodology, which will have problems in its implementation. It helps explain all of the steps involved in achieving the goal. All phases and their respective approaches and methods are discussed in this chapter.

CHAPTER 4: ANALYSIS AND DESIGN

4.1 Introduction

After the analysis phase, the Rapid Development process continues with research and design. It's a technique for gathering and analyzing data, finding issues, and deconstructing a system into its component elements. It explains how the plan will be carried out. The proposal must be rigorously planned during the design phase to ensure that the framework fits all of the requirements.

4.2 Problem Analysis

The main problem for some Face Recognition Based Door Locking is it does not have any two-factor authentication, so that can be bypass easily to unlock the door. Therefore, this project will be implemented a two-factor authentication for more secured.

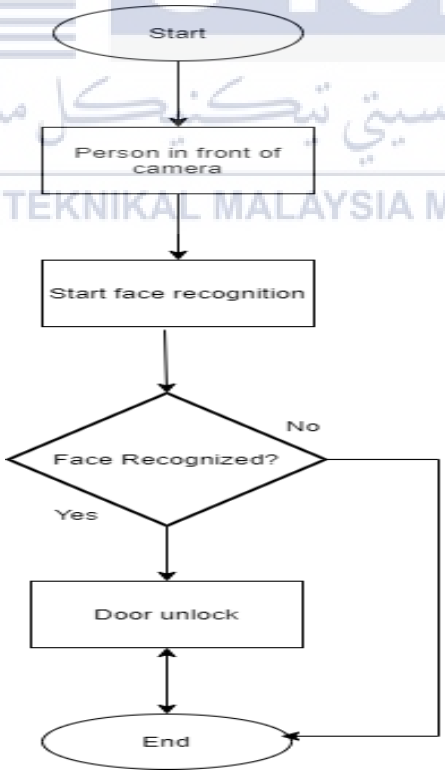


Figure 4.1 Research Flowchart

To create this project, need to use Twilio service to send an OTP to mobile number that stored in database after the system recognized the face.

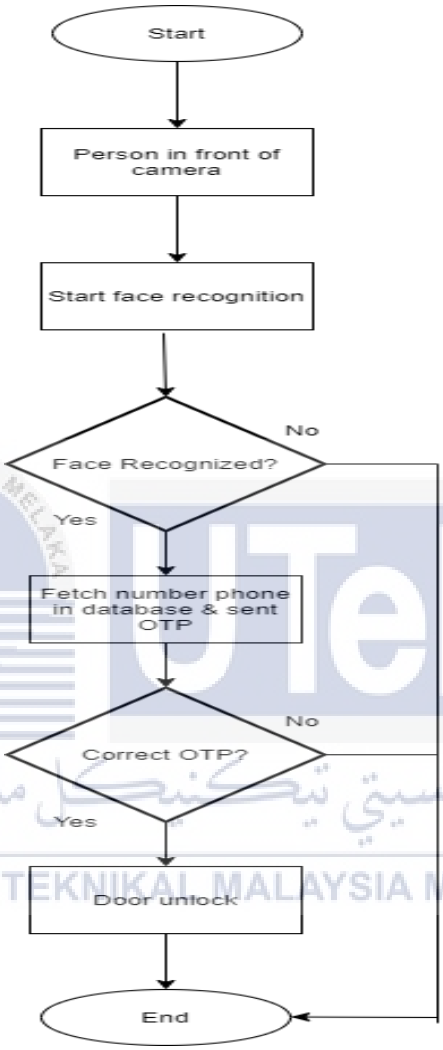


Figure 4.2 Flowchart Design (To unlock the door)

4.3 Requirement Analysis

Requirement analysis, often known as requirement engineering, is the process of defining user expectations for new or revised software. It's also known as requirements collecting or requirements capture in software engineering.

4.3.1 Data Requirement

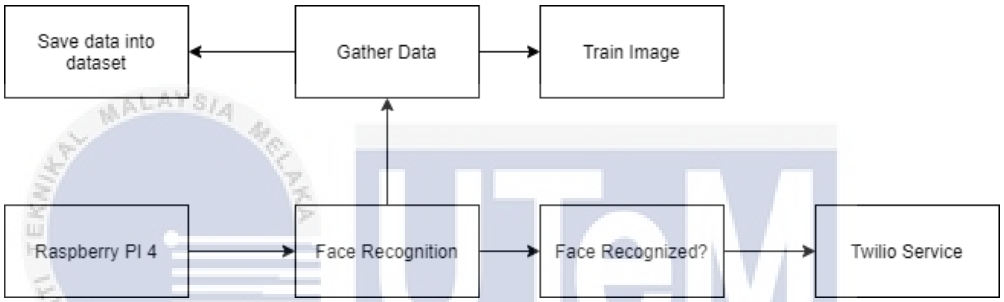


Figure 4.3 Data Flow

The data flow for Face Recognition Based Door Locking with 2FA is shown in figure __ . The raspberry PI will have attached with USB Webcam. Gather data is used to gather the faces of the user and info like name and number phone and save into dataset and database. After that need to train the image that captured before. Then the face recognition will recognize the faces and will send an OTP to user's phone number using Twilio service.

4.3.2 Requirement

4.3.2.1 Hardware Requirement

1. Raspberry PI 4

   The Raspberry PI is a credit-card sized computer created by the Raspberry PI Foundation in the United Kingdom with the goal of teaching basic computer science to schoolchildren and anybody else interested in computer hardware programming and DIY-Do-It-Yourself projects.

2. USB Webcam

   A simple USB powered webcam is used to recognize people faces

Figure 4.5 USB Webcam

3. Servo Motor

A servo motor is a self-contained electrical device that rotates machine parts with high precision and efficiency.

Figure 4.6 Servo Motor

4. Jumper wires

An electrical wire (or a collection of them in a cable) with a connector or pin at either end (or occasionally without – simply "tinned") that is typically used to link the components of a breadboard or other prototype.

Figure 4.7 Jumper wires

5. Laptop

To run VNC and able to connected to raspberry PI and monitor from laptop

Figure 4. 8 Laptop ASUS

4.3.2.2 Software Requirements

1. Raspbian Operating System (OS)

Raspberry Pi OS has been specifically designed for the Raspberry Pi range of small single-board computers powered by ARM CPUs. Raspberry Pi OS's desktop environment is a modified LXDE with the Openbox stacking window manager and a unique look.

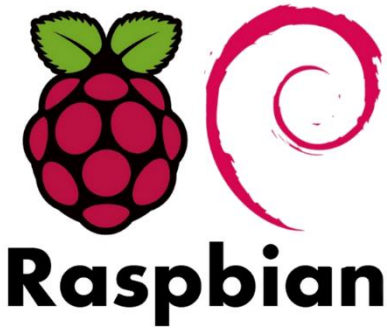


Figure 4.9 Raspbian OS

2. Thonny Python IDE

Thonny is an integrated development environment for Python that is designed for beginners. It supports different ways of stepping through the code, step-by-step expression evaluation, detailed visualization of the call stack and a mode for explaining the concepts of references and heap.

Figure 4.10 Tonny Python IDE

3. VNC

Virtual Network Computing (VNC) is a graphical desktop-sharing system that uses the Remote Frame Buffer protocol to remotely control another computer. It transmits the keyboard and mouse input from one computer to another, relaying the graphical-screen updates, over a network.

Figure 4.11 VNC

4.4 High-Level Design

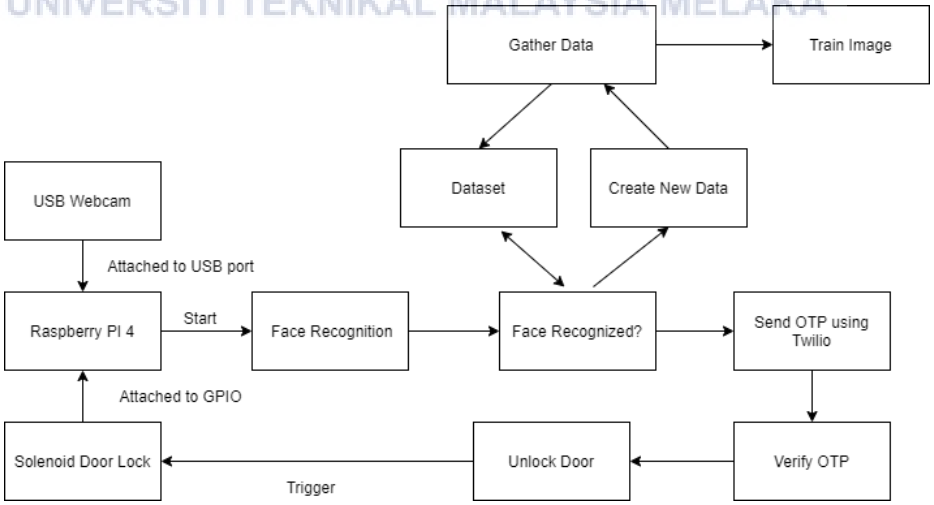


Figure 4.12 High-Level Design (HLD)

Figure shows the High-Level Design Diagram for this project. The architecture that would be utilized to construct a system is described in high-level design (HLD). The architectural diagram depicts a comprehensive system, indicating the primary components that would be produced for the product as well as their interactions.

4.4.1 System Architecture

A system architecture is a conceptual model that defines a system's structure, behaviour, and other aspects. A formal description and representation of a system arranged in a way that facilitates reasoning about the system's structures and behaviours is known as an architecture description.

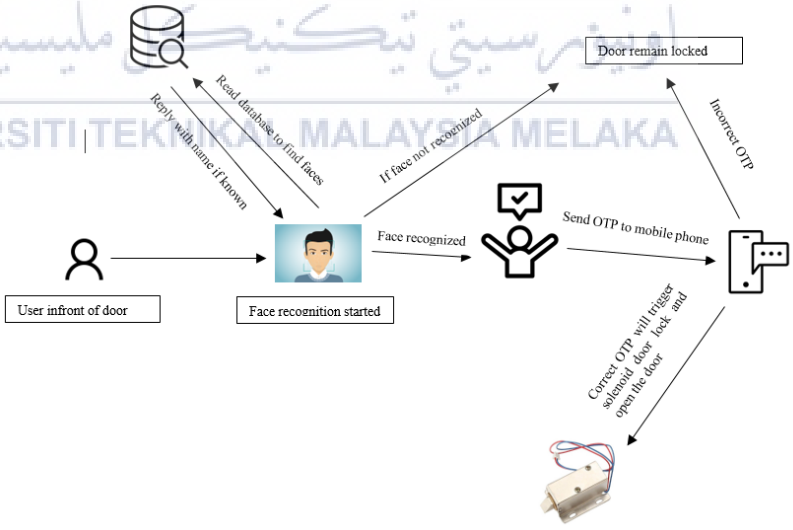


Figure 4.13 System Architecture

Connect the servo motor wire to the Raspberry Pi GPIO, orange wire of servo motor to pin 11 (GPIO 17) of Raspberry Pi, red to pin 4 (5v) and brown to pin 6 (GND)

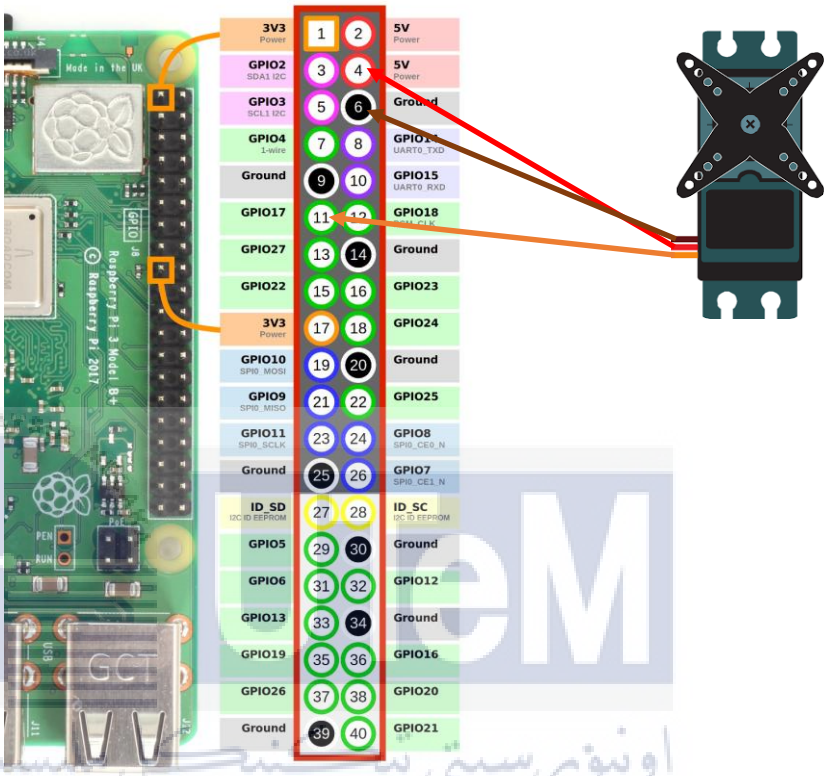


Figure 4.14 Circuit Diagram

4.4.2 User Interface Design

This project uses Tkinter to make a Graphical User Interface (GUI) to ease the operation. This interface provides the user with a fundamental understanding of how the system's flow operates. Figure below show the interface and the function.

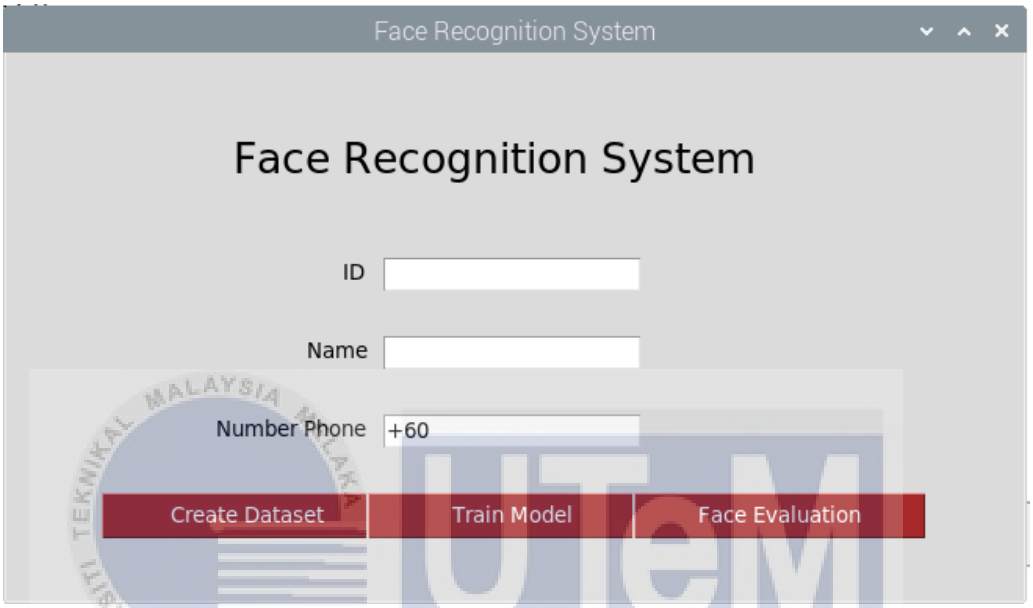
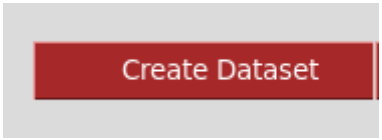
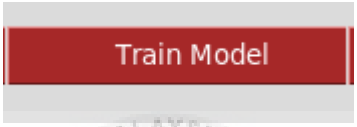
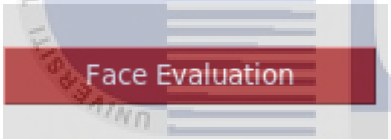


Figure 4.15 Interface of Face Recognition System

Table 4.1 Button Function

Button	Function
	• Input or update details of the user into database • Capture image and insert into dataset
	• Important step to do after capture an image, to ensure that the system recognizes the face images stored in the dataset
	• To test or evaluate whether it is accurate or not

4.5 Conclusion

The analysis and design are two of the most important parts of this project. To ensure that the hardware and software work effectively, all of the project's criteria must be created and researched beforehand. It is regarded a pre-implementation stage in this chapter, and it also incorporates the entire system flow for a better understanding prior to implementation. The following chapter is Implementation, which will go through how the project will be implemented and the results that may be expected.

CHAPTER 5: IMPLEMENTATION

5.1 Introduction

The chapter tells about the steps taken to take image of the user and insert into the dataset by using face recognition based on the proposed method. Also, this chapter will briefly explain the collection of how face recognition can send an OTP after recognized a face.

5.2 Software Development Environment Setup

5.2.1 Software Development Environment of Entire Project

In this part, several types of software are used such as Thonny Python IDE, Rasp OS and VNC.

5.2.2 Hardware Development Environment of Entire Project

In this part, several types of hardware are needed. The Raspberry Pi, wire jumper, servo motor, laptop and webcam.

5.3 Software Configuration Management

5.3.1 Configuration Environment Setup

In this part, we will briefly explain how to configure the entire environment in this project.

5.3.1.1 Configuration of SD Card

i. Open Raspberry Pi Imager

Figure 5.1 Raspberry Pi Imager

ii. Choose storage 'Mass Storage Device USB Device- 15.8 GB'

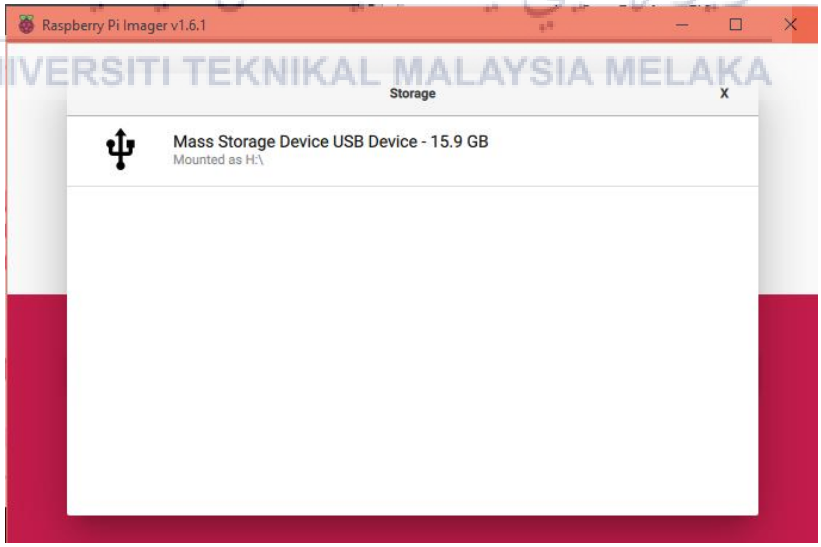


Figure 5.2 Choose Storage on Raspberry Pi Imager

iii. Choose Operating System ‘Raspberry Pi OS (32-bit)’

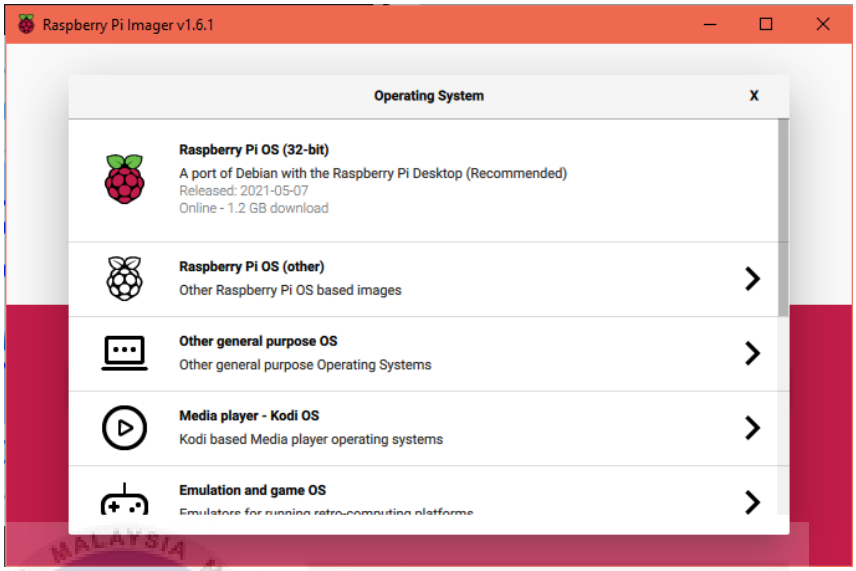


Figure 5.3 Choose Operating System on Raspberry Pi Imager

iv. Click ‘YES’ to write the SD Card

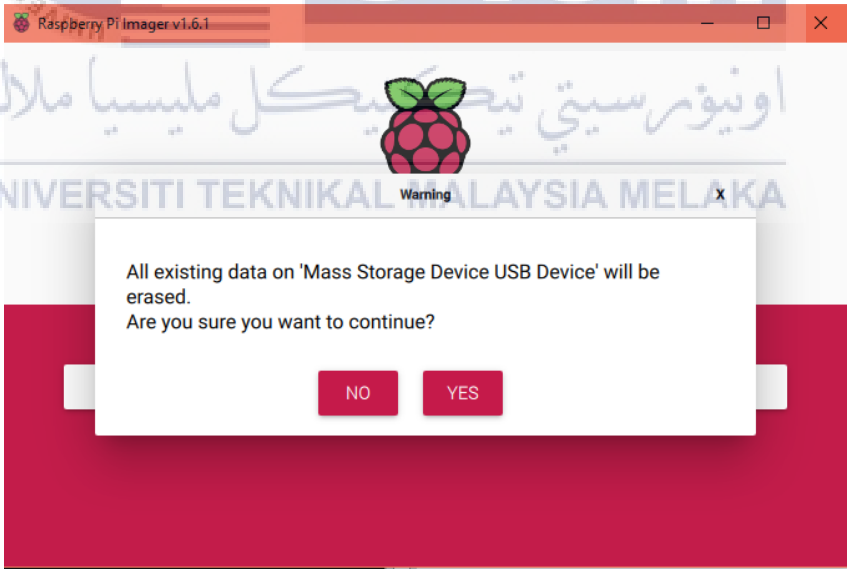


Figure 5.4 Write SD Card

5.3.1.2 Enable VNC on Rasp OS

i. Open Raspberry Pi Configuration

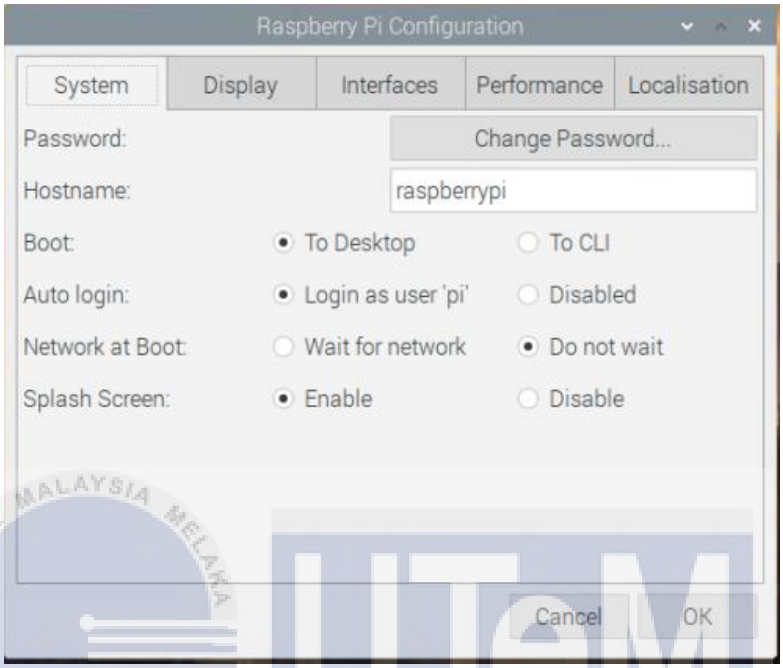


Figure 5.5 Raspberry Pi Configuration

ii. Click 'Interfaces' on Raspberry Pi Configuration to enable VNC

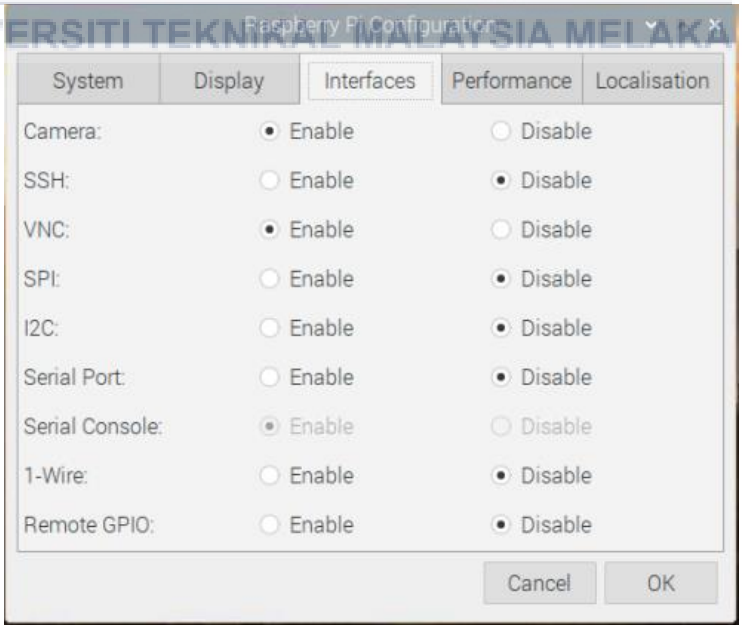


Figure 5.6 Interface of Raspberry Pi Configuration

5.3.1.3 Install OpenCV

i. Open Terminal

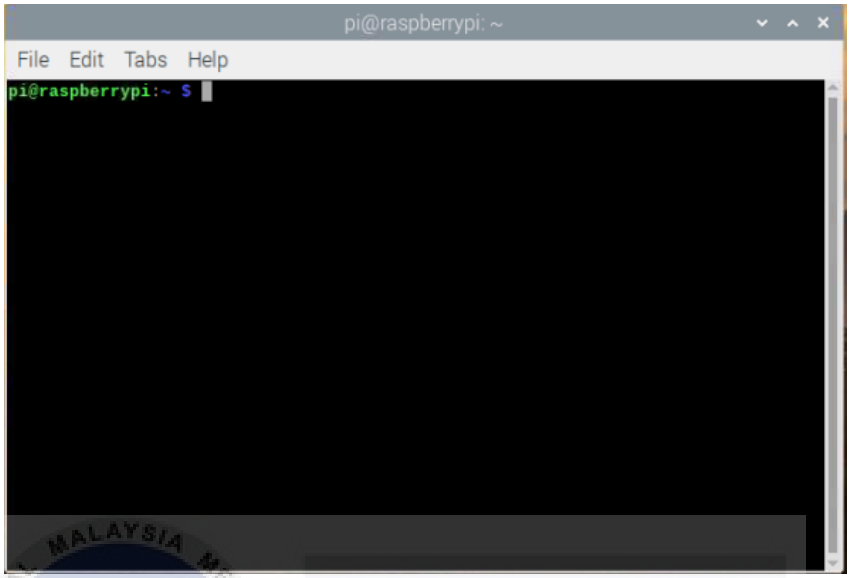


Figure 5.7 Terminal on Raspberry Pi

ii. Update Raspbian to the latest version

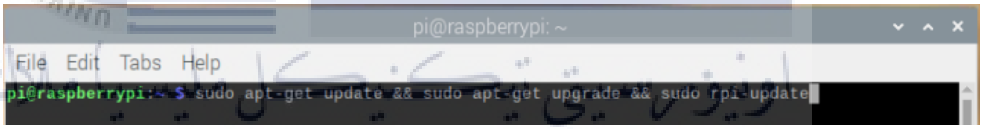


Figure 5.8 Update Raspbian

iii. Open swapfile

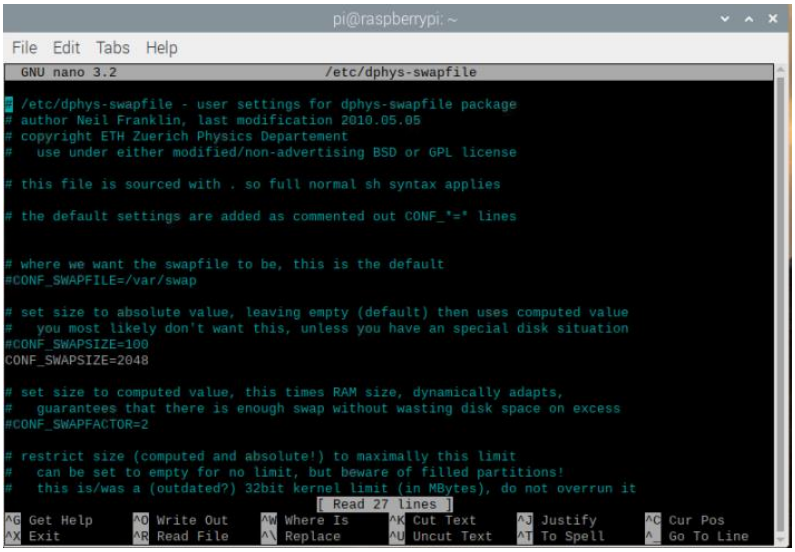


Figure 5.9 Swapfile

iv. Edit the variable ‘CONF_SWAPSIZE = 100’ to ‘CONF_SWAPSIZE = 2048’

```
# set size to absolute value, leaving empty (default) then uses computed value
#   you most likely don't want this, unless you have an special disk situation
#CONF_SWAPSIZE=100
CONF_SWAPSIZE=2048
```

Figure 5.10 Edit variable of CONF_SWAPSIZE

v. Install tools and libraries for openCV

1.

```
pi@raspberrypi:~ $ sudo apt-get install build-essential cmake pkg-config
Reading package lists... Done
Building dependency tree
Reading state information... Done
build-essential is already the newest version (12.6).
pkg-config is already the newest version (0.29-6).
cmake is already the newest version (3.16.3-3~bpo10+1).
The following package was automatically installed and is no longer required:
  python-colorzero
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 91 not upgraded.
```

2.

```
pi@raspberrypi:~ $ sudo apt-get install libjpeg-dev libtiff5-dev libjasper-dev libpng12-dev
Reading package lists... Done
Building dependency tree
Reading state information... Done
libjasper-dev is already the newest version (1.900.1-debian1-2.4+deb8u1).
libjpeg-dev is already the newest version (1:1.5.2-2+deb10u1).
libtiff5-dev is already the newest version (4.1.0+git191117-2~deb10u2).
The following packages were automatically installed and are no longer required:
  gir1.2-atspi-2.0 gir1.2-gtk-2.0 gir1.2-harfbuzz-0.0 icu-devtools libatk-bridge2.0-dev
  libatk1.0-dev libatspi2.0-dev libblkid-dev libcairo-script-interpreter2 libdbus-1-dev
  libegl-dev libegl1-mesa-dev libepoxy-dev libffi-dev libfribidi-dev libgl-dev libgl1-mesa-dev
  libglib2.0-dev libglib2.0-dev-bin libglx-dev libgraphite2-dev libharfbuzz-dev
  libharfbuzz-gobject0 libicu-dev liblzo2-2 libmount-dev libpcre16-3 libpcre3-dev libpcre32-3
  libpcrecpp0v5 libpixman-1-dev libpng-tools libselinux1-dev libsepol1-dev libwayland-bin
  libwayland-dev libxcb-render0-dev libxcb-shm0-dev libxcomposite-dev libxcursor-dev
  libxdamage-dev libxext-dev libxfixes-dev libxi-dev libxinerama-dev libxkbcommon-dev
  libxml2-utils libxrandr-dev libxrender-dev libxtst-dev pango1.0-tools python-colorzero
  uuid-dev wayland-protocols x11proto-composite-dev x11proto-damage-dev x11proto-fixes-dev
  x11proto-randr-dev x11proto-record-dev x11proto-xext-dev x11proto-xinerama-dev
Use 'sudo apt autoremove' to remove them.
The following packages will be REMOVED:
  libcairo2-dev libfontconfig1-dev libfreetype6-dev libgdk-pixbuf2.0-dev libgtk-3-dev
  libgtk2.0-dev libpango1.0-dev libpng-dev libxft-dev
The following NEW packages will be installed:
  libpng12-dev
0 upgraded, 1 newly installed, 9 to remove and 91 not upgraded.
Need to get 0 B/233 kB of archives.
After this operation, 35.2 MB disk space will be freed.
Do you want to continue? [Y/n] y
(Reading database ... 171606 files and directories currently installed.)
Removing libgtk2.0-dev:armhf (2.24.32-3+rpt1) ...
Removing libgtk-3-dev:armhf (3.24.5-1+rpt2) ...
Removing libpango1.0-dev:armhf (1.42.4-8~deb10u1) ...
Removing libxft-dev:armhf (2.3.2-2) ...
Removing libgdk-pixbuf2.0-dev:armhf (2.38.1+dfsg-1) ...
```

Figure 5.11 Install tools and libraries for OpenCV

vi. Check for OpenCV

```
pi@raspberrypi:~ $ python3
Python 3.7.3 (default, Jan 22 2021, 20:04:44)
[GCC 8.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
>>> import cv2
>>> cv2._version_
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
AttributeError: module 'cv2' has no attribute '_version_'
>>> cv2.__version__
'4.1.0'
```

Figure 5.12 Check for OpenCV

5.3.1.4 Insert or update details into SQLite3 database

i. First, must import sqlite3 and os.path. Make sure that database is in the same folder.

```
import sqlite3
import os.path
```

Figure 5.13 Import sqlite3 and os.path

ii. Need to fill the details if not, alert box will appear

```
def collect_data():
    if(ID.get()=="" or Name.get()=="" or Num_Phone.get()==""):
        messagebox.showinfo('Alert','Please fill the box')
```


Figure 5.14 Fill the details

iii. If the user input ID that existed, it will update the details.

```
else:
    face_id=ID.get()
    name=Name.get()
    num_phone=Num_Phone.get()
    conn = sqlite3.connect(db_path)
    cmd = "SELECT * FROM User WHERE ID =" + (face_id)
    cursor = conn.execute(cmd)
    isRecordExist = 0
    for row in cursor:
      isRecordExist = 1
    if(isRecordExist == 1):
      messagebox.showinfo('Result','User exist! Updating info..')
      cursor.execute('UPDATE User SET Num_Phone =? WHERE ID = ?',(num_phone,face_id))
      messagebox.showinfo('Result','Successfully updated number phone!')
```

Figure 5.15 User Update

iv. Stored the details into database

```
else:
  cursor.execute('INSERT INTO User (ID,Name,Num_Phone) VALUES(?,?,?)',(face_id,name,num_phone,))
  messagebox.showinfo('Result','Successfully input data into database!')
```

Figure 5.16 User Insert

v. Table in database

User		CREATE TABLE
ID	INTEGER	`ID` INTEGER N
Name	TEXT	`Name` TEXT
Num_Phone	TEXT	`Num_Phone` T

Figure 5.17 Table Information

5.3.1.5 Capture faces and insert into dataset

i. Set video width and height and use the cascade as detector

```
cam = cv2.VideoCapture(0)
cam.set(3, 640) # set video width
cam.set(4, 480) # set video height

face_detector = cv2.CascadeClassifier('Cascades/haarcascade_frontalface_default.xml')
```

Figure 5.18 Set Video Width and Height

ii. Change the captured image into gray and create a rectangle to detect a face.

```
while(True):
 ret, img = cam.read()
 img = cv2.flip(img, 1) # flip video image vertically
 gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
 faces = face_detector.detectMultiScale(gray, 1.3, 5)

 for (x,y,w,h) in faces:
     cv2.rectangle(img, (x,y), (x+w,y+h), (255,0,0), 2)
     count += 1
```

Figure 5.19 Turn image into gray and create a rectangle

iii. Save the captured image into dataset folder and it will take 70 images of a user's face.

```
        # Save the captured image into the datasets folder
        cv2.imwrite("dataset/User." + str(face_id) + '.' + str(count) + ".jpg", gray[y:y+h,x:x+w])

    cv2.imshow('Face Recognition', img)

    k = cv2.waitKey(100) & 0xff # Press 'ESC' for exiting video
    if k == 27:
        break
    elif count >= 70: # Take 70 face sample and stop video
        break
```

Figure 5.20 Save captured image into dataset

5.3.1.6 Train the face that stored in dataset

i. Train the faces and it will produce a file name trainer.yml

```
def getImagesAndLabels(path):
    imagePaths = [os.path.join(path,f) for f in os.listdir(path)]
    faceSamples=[]
    ids = []
    for imagePath in imagePaths:
        PIL_img = Image.open(imagePath).convert('L')
        img_numpy = np.array(PIL_img,'uint8')
        id = int(os.path.split(imagePath)[-1].split(".")[1])
        faces = detector.detectMultiScale(img_numpy)
        for (x,y,w,h) in faces:
            faceSamples.append(img_numpy[y:y+h,x:x+w])
            ids.append(id)
    return faceSamples,ids

faces,ids = getImagesAndLabels(path)
recognizer.train(faces, np.array(ids))

# Save the model into trainer/trainer.yml
recognizer.save('trainer/trainer.yml')
```


Figure 5.21 Train the faces

5.3.1.7 Face Evaluation

i. If the face is recognized, it will show the name that recognized.

```
    for(x,y,w,h) in faces:
        cv2.rectangle(img, (x,y), (x+w,y+h), (0,255,0), 2)
        id, confidence = recognizer.predict(gray[y:y+h,x:x+w])

        # If confidence is less then 100 ==> "0" : perfect match
        if (confidence < 100):
            id = names[id]
            confidence = "  {0}%".format(round(100 - confidence))
        else:
            id = "unknown"
            confidence = "  {0}%".format(round(100 - confidence))

        cv2.putText(img, str(id), (x+5,y-5), font, 1, (255,255,255), 2)
```

Figure 5.22 Face Evaluation

5.3.1.8 face_lock.py is to open the door

i. If the face is recognized or not, it will return a Boolean, ID and name. Then, if recognized, it will prompt either want to send an OTP or not. Other than that, the door still remains locked.

```
if (confidence < 100):
    id1 = names[id]
    confidence = "  {0}%".format(round(100 - confidence))
    cnt=cnt+1;
    if cnt%5==0:
        faceRec=True
        photoFile,dt_string= trigger("Success")
        write_to_csv(photoFile,dt_string,str(id1))
        return faceRec,faceNotRec,id,id1
else:
    id1 = "unknown"
    confidence = "  {0}%".format(round(100 - confidence))
    cnt=cnt+1;
    if cnt%5==0:
        faceNotRec=True
        photoFile,dt_string=trigger("Fail")
        write_to_csv(photoFile,dt_string,str(id1))
        return faceRec,faceNotRec,id,id1


return faceRec,faceNotRec,id,id1
print("\nExiting face recognition....\n")
cam.release()
cv2.destroyAllWindows()
#doorQuest = input('Do you want to open the door?[y/n]')
#if doorQuest == 'y':
isDoorOpen,faceNotRec,id,id1 = recognizeFace()
if isDoorOpen==True:
    sentOTP = input(id1 + " is recognized! Sent OTP? [y/n] : ")
    if sentOTP == 'y' or sentOTP == 'Y':
        fetchNumPhone(id)

    else:
        print("\nDoor locked\n")
        sys.exit()

if faceNotRec==True:

    print('Face is unknown! Door locked!\n')
```

Figure 5.23 face_lock.py

5.3.1.9 Fetch phone number from the ID of the user

i. Fetch number phone

```
def fetchNumPhone(id):
    conn = sqlite3.connect(db_path)
    cmd = "SELECT * FROM User WHERE ID =" + str(id)
    cursor = conn.execute(cmd)
    profile = None
    for row in cursor:
        profile = row
        num_phone = profile[2]
        sendOTP(num_phone)
```

Figure 5.24 Fetch number phone

5.3.1.10 Send an OTP

i. Send an OTP

```
otp = random.randint(1000,9999)
otp = str(otp)

def sendOTP(num_phone):
    print("\nSending your OTP to " +num_phone+ "...")
    print("Success!\n")

    auth_sid = "[redacted]"
    auth_token = "[redacted]"
    client = Client(auth_sid,auth_token)

    message = client.messages.create(
        body = "Your OTP number to open the door is " +str(otp),
        from_ = "[redacted]",
        to = num_phone
        )

    checkOTP(otp)
```


Figure 5.25 Send an OTP

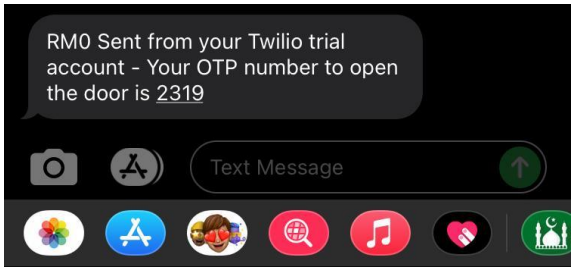


Figure 5.26 Example of received OTP

5.3.1.11 Check the OTP

i. If the OTP is correct,

```
def checkOTP(otp):
    chkOTP = input("Please enter your OTP : ")
    try:
        if chkOTP == otp:
            print("Successfull! Door is unlock!\n")

            #to unlock the door
            servo1.start(0)
            time.sleep(1)

            # Define variable duty
            duty = 2

            # Turn back to 90 degrees
            servo1.ChangeDutyCycle(7)
            time.sleep(10)

            #turn back to 0 degrees
            print ("Door is lock after 10s")
            servo1.ChangeDutyCycle(2)

            time.sleep(0.5)
            servo1.ChangeDutyCycle(0)

            #Clean things up at the end
            servo1.stop()
            GPIO.cleanup()
```

Figure 5.27 OTP Correct

ii. If the OTP is incorrect,

```
        ...
        else:
            print("\nWrong OTP! Door is locked!\n")

    except Exception:
        print("Invalid OTP")
```

Figure 5.28 OTP Incorrect

5.3.1.12 Success and fail folder

i. If the face is recognized or not, it will store in success and fail folder. Also in csv file type too.

```
def recognizeFace():

    def trigger(folder):
        now = datetime.now()
        dt_string = now.strftime("%d-%m-%Y %H-%M-%S")
        photoFile="log/"+ folder +"/Photo"+ str(dt_string) +".jpg"
        cv2.imwrite(photoFile, img)
        print("\n Exiting face recognition....\n")
        cam.release()
        cv2.destroyAllWindows()
        return photoFile,dt_string

    def write_to_csv(photoFile,dt_string,name):
        with open("log/log_Files.csv", mode="a") as csv_readings:
            logfile_write = csv.writer(csv_readings, delimiter=",", quotechar="/", quoting=csv.QUOTE_MIN
            write_to_log = logfile_write.writerow([photoFile,dt_string,name])
        return(write_to_log)
```

Figure 5.29 Coding for log folder

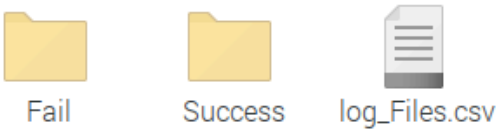


Figure 5.30 Example of log folder

5.3.1.13 Servo Motor

i. Connect the servo motor wire to the Raspberry Pi GPIO, orange wire of servo motor to pin 11 (GPIO 17) of Raspberry Pi, red to pin 4 (5v) and brown to pin 6 (GND)

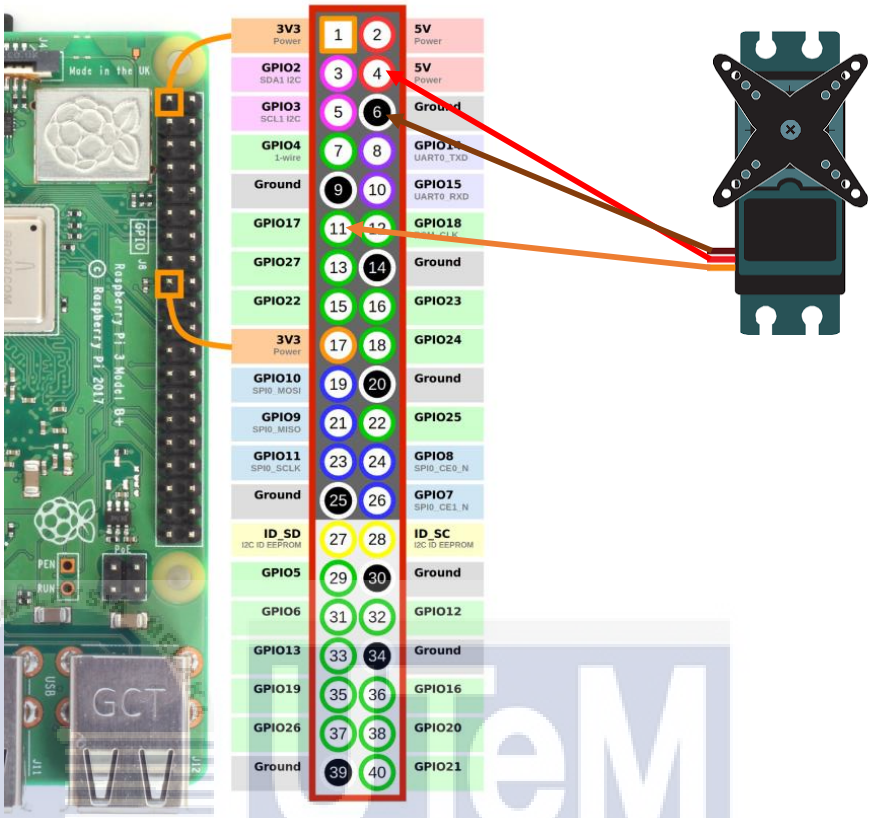


Figure 5.31 Servo motor connected to GPIO

5.3.2 Version Control Procedure

Thonny Python IDE source code version is physical items that connect an embedded system to the internet' in projects involving big data, machine learning, data analytics, wireless data networks, and cyber-physical systems, it is critical. It also Support for a large number of open-source libraries. Syntax that is both efficient and precise. Then, Integration with other programming languages is straightforward. Python has a low barrier to entry. Lastly, adaptable to a variety of operating systems and architectural style

5.4 Implementation Status

i. Component or module name

- Python
- SQLite3

ii. Description

- Python is widely used for web and software development, task automation, data analysis, and data visualization.
- SQLite is a C-language library that implements a small, fast, self-contained, high-reliability, full-featured, SQL database engine.

iii. Duration to complete

- 2 months.

iv. Date completed

- 02/9/2021

v. Size of software

- 1.08GB

5.5 Conclusion

This chapter has covered every part of installing OpenCV, configure Raspbian Pi. This chapter also demonstrates how to use the Thonny Python IDE and SQLite3 to implement the source code. Finally, the technique for version and the statues of implementation have already discussed in this chapter

CHAPTER 6: TESTING

6.1 Introduction

This chapter will go over the testing part of the approach that was outlined in chapter two. The test plan, the initial research of testing on this system, test organization, test environment, test schedule, and test technique are all included in this testing documentation. There are two sorts of testing in test design: test description and data and test outcome and analysis. This system is tested in a test environment that includes both hardware and software. Furthermore, at this testing phase, the test strategy is crucial because it directs the testing methodologies.

6.2 Test Plan

The test plan is a document that describes the approach, scope, resources, and schedule for upcoming test exercises. It distinguishes test items, highlights to be tried, testing assignments, which does each task, analyzer autonomy degree, and test situation, among other things. The tests define systems and sections, as well as the criteria to be used, the logic behind their selection, and any risks that require risk management. It's a summary of the test preparation method.

6.2.1 Test Organization

In this part, the test organization consists of user, that is owner of the house and admin. Scopes that are going tested include functional requirement and non-functional requirement on the table 6.1. Every failure and error of the outcome in the system will be recorded and fixed.

Table 6.1: Roles and Responsibilities

Roles	Responsibilities
User (Owner)	• Insert details in the form. • Insert face images into dataset. • Update number phone and images. • Received and input OTP. • Using face recognition to recognize their faces.
Admin	• Keep data.

6.2.2 Test Environment

The test environment is made up of components that aid in the execution of software and hardware tests. To uncover any condition or arrangement-related flaws, the test environment design must replicate the creation condition. The hardware components used throughout this development are listed in Table 6.2. Table 6.3 lists the programs and software that are used to set up the configuration system.

Table 6.2: Test Environment of Hardware Components

Environment Specification	Description
Model 1	Raspberry Pi 4 Model B
Storage	4GB
Model 2	1080 Webcam HD Web Camera
Model 3	Power Adapter Raspberry Pi
Model 4	SD Card
Storage	16GB

Table 6.3: Test Environment of Software Components

Environment Specification	Description
Operating System	Raspberry Pi
Development Tools	Thonny Python IDE
Database	SQLite3
Documentation	CSV

6.2.3 Test Schedule

Test schedule is a timeline for software testing that includes the testing stages as well as the start and conclusion dates and responsibilities. It should also show how the test will be assessed, followed, and confirmed. The activity, testing description, start and finish dates, and duration of the testing are all listed in Table 6.4.

Table 6.4: Schedule of Face Recognition based Door Locking for Testing

Activities	Description	Start Date	End Date	Duration
Functionality Testing	Functionality testing is defined as a type of testing that ensures that each capacity of a product application is operating in accordance with the requirements.	20/7/2021	21/8/2021	31 days

6.3 Test Strategy

Black-box testing, white-box testing, bottom-up testing, and top-down testing are the four forms of testing used by Test Strategy. Black box testing, also known as Behavioural Testing, is a software testing approach in which the tester has no knowledge of the tested objects' internal structure, design, or implementation. This type of testing is used to compare functional and non-functional requirements. White box testing, also known as Code-Based Testing or Structural Testing, is a method of software testing in which the tester selects paths using a programming language and calculates acceptable outputs from the input resources.

Furthermore, bottom-up testing examines each segment of the lower progressing system separately before moving on to the segments that rely on them. Top-down testing, on the other hand, is a coordination testing system that is used to simulate the behaviour of lower-level modules that have not yet been integrated. However, the black box testing methodologies like as equivalence testing and boundary value analysis are only covered in this project.

6.4 Test Design

To construct and write test suites for software testing, test designs are provided. The goal of the test design is to ensure that the supplied requirements are met in tandem with the client's needs and desires. Test design is divided into two parts: test description and test data.

6.4.1 Test Description

Test description explains test cases. All the tests will be listed on the test result and analysis part.

6.5 Test Results and Analysis

6.5.1 Test Insert and Update User

i. Insert user's details

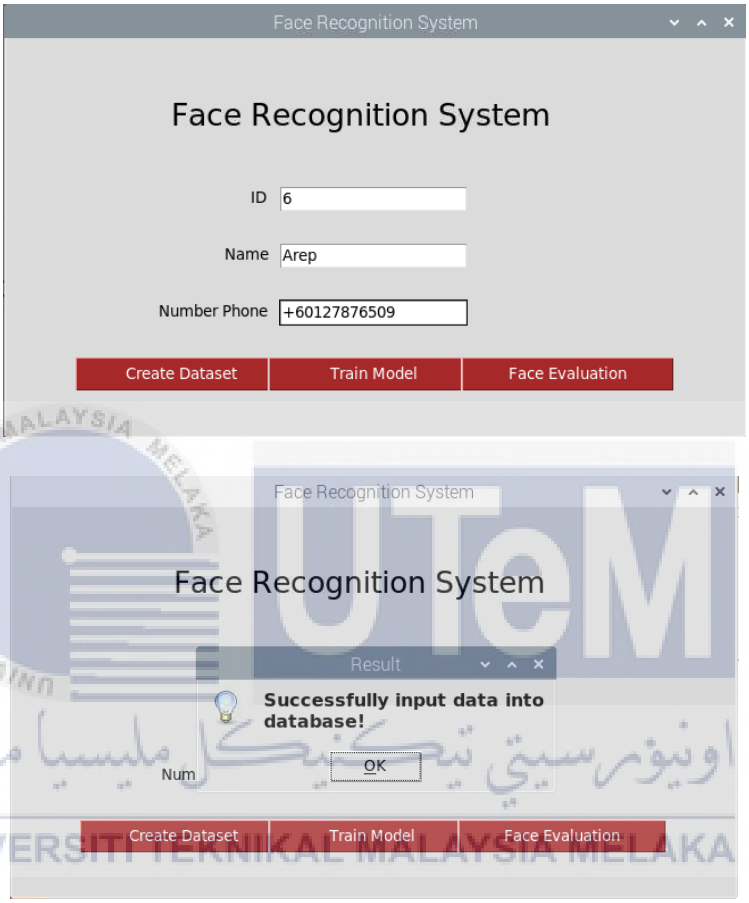
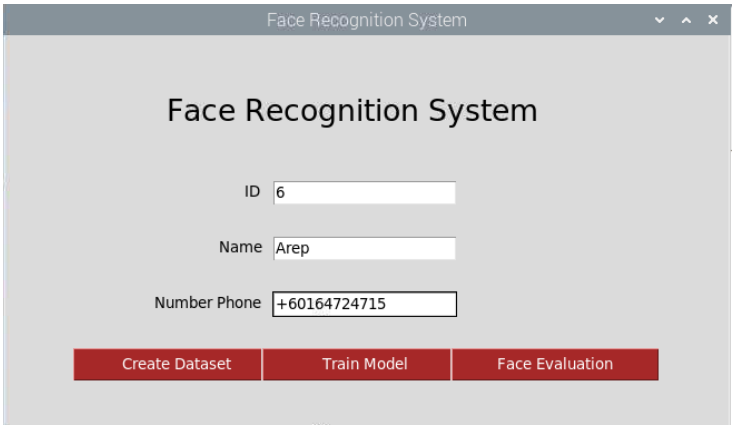
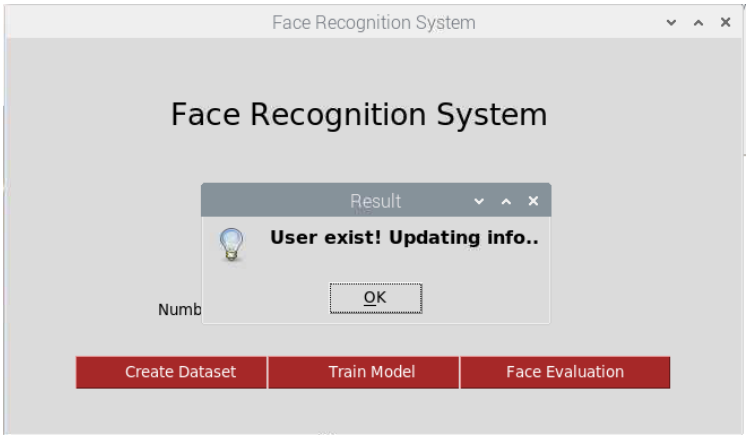


Figure 6.1 Insert user details

ii. Update user's number phone





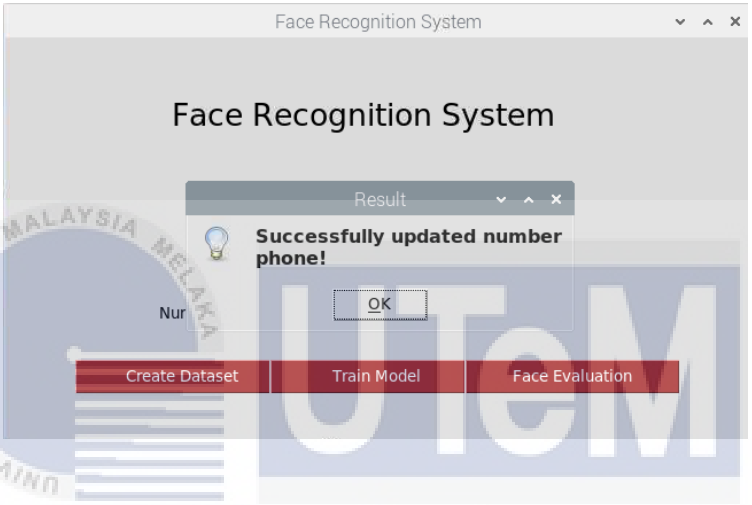


Figure 6.2 Update user's number phone

6.5.2 Test system capability to detect face

Table 6.1 shows the test case to ensure the system recognizes the person wearing headgear, spectacle, and surgical masks. Table 6.2 shows the result of the testing based on the test codes in Table 6.1.

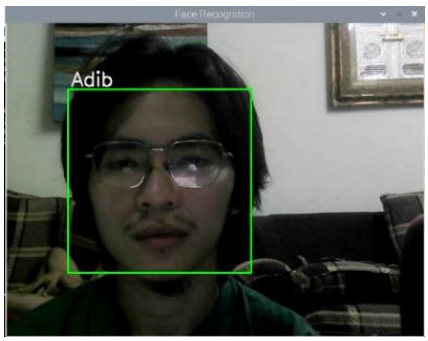


Figure 6.3 Example of test A1

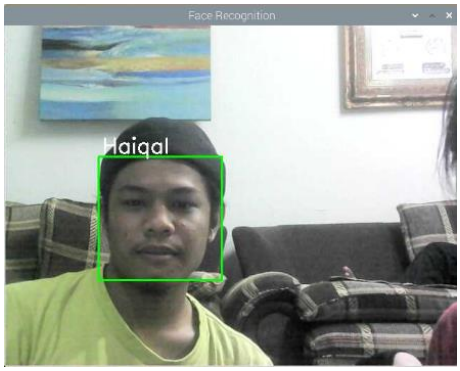


Figure 6.4 Example of test A2

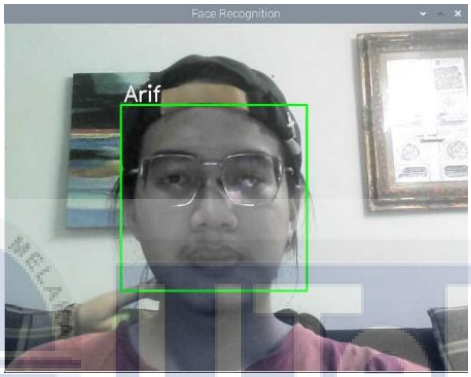


Figure 6.5 Example of test A3

Figure 6.6 Example of test A4

Table 6.5 Test Case

Test Code	Description
A1	Wearing spectacle
A2	Wearing headgear
A3	Wearing spectacle and headgear
A4	Wearing surgical mask

Table 6.6 Test Result

Participant	Test Code			
	A1	A2	A3	A4
A	Pass	Pass	Pass	Failed
B	Pass	Pass	Pass	Failed
C	Pass	Pass	Pass	Failed
D	Pass	Pass	Pass	Failed
E	Pass	Pass	Pass	Failed

6.5.3 Test of a person's face in colour and black & white on a sheet

In addition, we also tested the print of a person's face in colour and print of a person's face in black and white. Table 6.3 shows the testing results.

Figure 6.7 Example of print of user's face in colour

Figure 6.8 Example of print of user's face in black & white

Table 6.7 Test Print of Face on a Sheet

Participant	Color	Black & White
A	Pass	Pass
B	Pass	Pass
C	Pass	Pass
D	Pass	Pass
E	Pass	Pass

6.5.4 Brightness Level Test

Brightness also affects the system's capability to recognize a person. The value is between 0 to 100; a higher value indicates a brighter face image. Table 6.4 shows the testing that has been done to identify the brightness level that the system would still be able to recognize.

Table 6.8 Brightness Level Test

Test No.	Brightness Level	Result
1	100	Pass
2	90	Pass
3	80	Pass
4	70	Pass
5	60	Pass
6	50	Failed
7	40	Failed
8	30	Failed
9	20	Failed
10	10	Failed
11	0	Failed

6.5.5 LBPH vs dlib face_recognition

Moreover, in Table 6.5, we also measure the processing time taken between Local Binary Patterns Histograms (LBPH) algorithm and dlib face_recogntion algorithm to recognize the user and for Table 6.6 is time taken for both of algorithms to train the faces of 5 users.

Table 6.9 Time Taken to Recognize Face

Participant	Algorithm	
	LBPH	dlib face_recognition
A	1.43s	1.49s
B	1.40s	1.50s
C	1.45s	1.56s
D	1.58s	1.36
E	1.50s	1.54s

Table 6.10 Time Taken to Train Faces

Algorithm	Time taken
LBPH	21.96s
dlib face_recognition	43.43s

6.6 Conclusion

The strategies used to confirm and approve the system functions are described in this chapter. This chapter has also demonstrated the project's success in all testing. Furthermore, LBPH algorithm was able to recognize a person wearing spectacles and a headgear as described in Table I's test case. It can also recognize the side of a person's face with ease and fast in training the faces. However, this system is not good in low brightness situations. Thus it cannot recognize a person under low-level brightness.

CHAPTER 7: PROJECT CONCLUSION

7.1 Introduction

This chapter will be discussed about the overall performance of this system using Raspberry Pi, which will include all analysis of the system and propositions on improvements according to the strength and weakness analysis. Furthermore, contribution of project will be defined in this chapter as well.

7.2 Project Summarization

This project was developed to ensure the security of a house and its inhabitants through face recognition and authentication. This study shows that the developed two-factor authentication for face recognition offers a high level of protection without depending on keys. Also, the admin can check who failed and success through recognition process in a certain file.

7.3 Project Contribution

Contribution on this project is the household will be more secure from any danger by offered a two-factor authentication and also not to depending on keys anymore to enter the house. This is because some people carelessly lose the keys or the keys is stolen. Furthermore, such constraints will be overcome and significantly improved in this project.

7.4 Project Limitation

The main limitation of this system is that it has a weakness in detecting the faces more accurately. Due to limited resources, this system need the user to push a button to start the face recognition. Other than that, some low-quality cameras randomly drop their brightness levels and make it difficult to recognize the face.

7.5 Future Works

This system can be upgraded, by making it more accurate by using and testing other algorithms. Also, can also implement a sensor to start the face recognition process automatically and other functions such as sending a notification to the use's phone when the system recognize an unknown face. In addition, camera quality is vital in this case, because the higher the quality of the camera, the better for it to recognize or capture an image.

7.6 Conclusion

To summarize, this project was completed according to plan, and the conditions specified in the preceding section were also met. It has been of great benefit to society by providing guidance and instructions on how to resolve the mentioned concerns in accordance with the project's goals. Finally, it is hoped that the community will be able to apply this project to the fields of information technology and computer science while understanding the value of this research.

REFERENCES

Ali, W., Dustgeer, G., Awais, M., & Shah, M. A. (2017, September). IoT based smart home: Security challenges, security requirements and solutions. In 2017 23rd International Conference on Automation and Computing (ICAC) (pp. 1-6). IEEE.J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

N. H. Motlagh, M. Bagaa, T. Taleb., ‘'UAV-based iot platform: a crowd surveillance use case,’‘IEEE Commun. Mag, vol. 55, No. 2, pp. 128-134, 2017.K. Elissa, "Title of paper if known," unpublished.

C. H. Chang., "Smart classroom roll caller system with IOT architecture," in Proceedings - 2011 2nd International Conference on Innovations in Bio-InsPired Computing and Applications, IBICA 2011, pp. 356-360, 2011.Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].

J. See & S. W. Lee, ‘'An integrated vision-based architecture for home security system," IEEE Trans. Consum. Electron, vol. 53, No. 2, pp. 489-498, 2007

L. Y. Mano et al., "Exploiting IoT technologies for enhancing health smart homes through patient identification and emotion recognition," Comput. Commun, vol. 89-90, pp. 178-190, 2016.

G. Senthilkumar, K. Gopalakrishnan, V. S. Kumar., “Embedded image capturing system using Raspberry Pi system," vol. 3, No. 2, pp. 213-215, 2014.

M. R. Mulla., "Facial image based security system using PCA," pp. 548-553, 2015.

Sahani, M., Nanda, C., Sahu, A., & Pattnaik, B. (2015). Web-based online embedded door access control and home security system based on face recognition. 2015 International Conference on Circuits, Power and Computing Technologies [ICCPCT-2015], 1-6.

M. Sajjad et al., "Raspberry Pi assisted face recognition framework for enhanced law-enforcement services in smart cities," Futur. Gener. Comput. Syst., 2017.

A. R. Syafeeza, S. S. Liew, R. Bakhteri., ‘'Convolutional neural networks with fused layers applied to face recognition," Int. J. Comput. Intell. Appl., vol. 14, No. 3, 2015.

A. R. Syafeeza, M. Khalil-Hani, S. S. Liew, R. Bakhteri., "Convolutional neural network for face recognition with pose and illumination variation," Int. J. Eng. Technol., Vol. 6, No. 1, pp. 44-57, 2014.

K. Syazana-Itqan, A. R. Syafeeza, N. M. Saad, N. A. Hamid, W. H. Bin Mohd Saad., "A review of finger-vein biometrics identification approaches," Indian J. Sci. Technol., vol. 9, No. 32, 2016.

Hteik Htar Lwin, Aung Soe Khaing, Hla Myo Tun "Automatic Door Access System Using Face Recognition" International Journal of Scientific Technology Research, Issue 06, volume 4, June 2015.

Patil, A. N., Ranavare, R. B., Ballal, D. V., & Kotekar, A. Raspberry Pi based face recognition system for door unlocking. *International journal of innovative research in science and engineering vol*, (2).

Yedulapuram, S., Arabelli, R., Mahender, K., & Sidhardha, C. (2020, December). Automatic Door Lock System by Face Recognition. In IOP Conference Series: Materials Science and Engineering (Vol. 981, No. 3, p. 032036). IOP Publishing.

Khan, A. (2021). Face Recognition Door Lock Using Raspberry Pi with AWS Recognition. Case Studies for Research in Computer Science and Engineering, 77

Saputra, R., & Surantha, N. (2021). Smart and real-time door lock system for an elderly user based on face recognition. Bulletin of Electrical Engineering and Informatics, 10(3), 1345-1355.

Prasade, K., Nalavade, S., & Pathak, D. (2018). Face Recognition Based Door Locking System. International Research Journal of Engineering and Technology (IRJET).

V. Jones, "Rapid object detection using a boosted cascade of simple features," Computer Vision and Pattern Recognition, 2001.

S. Soo, Object Detection Using Haar-Cascade Classifier, Institute of Computer Science, University of Tartu, 2014.

Y. Januzaj, A. Luma, Y. Januzaj, V. Ramaj., "Real time access control based on face recognition," in International Conference on Network security & Computer Science (ICNSCS-15), pp. 7-12, 2015.

Zhao, X., & Wei, C. (2017). A real-time face recognition system based on the improved LBPH algorithm. 2017 IEEE 2nd International Conference on Signal and Image Processing (ICSIP), 72-76.

Jain, A.K. and Li, S.Z., 2011. Handbook of face recognition. New York: springer.