FACE RECOGNITION SMART DOOR LOCK SECURITY SYSTEM USING HAAR CASCADE ALGORITHM AND LOCAL BINARY PATTERNS ALGORITHM

MAH KIM FONG

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

FACE RECOGNITION SMART DOOR LOCK SECURITY SYSTEM USING HAAR CASCADE ALGORITHM AND LOCAL BINARY PATTERNS ALGORITHM

MAH KIM FONG

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

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2021

DECLARATION

I hereby declare that this project report entitled

FACE RECOGNITION SMART DOOR LOCK SECURITY SYSTEM USING HAAR CASCADE ALGORITHM AND LOCAL BINARY PATTERNS ALGORITHM

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

STUDENT : _____ Date : 8.9.2021
(MAH KIM FONG)

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 Networking)] with Honours.

SUPERVISOR : _____ Date : 8.9.2021
(TS. DR. NORHARYATI HARUM)

DEDICATION

This project is dedicated to my beloved parents, Mr. Mah Ang Lai, and Mrs. Lily Yoo who have been giving me support and encourage me throughout this project. I would like to thank for my supervisor, Ts. Dr. Norharyati Harum for giving me guidance to complete this project. Finally, I would like to thank to my friends who providing me a lot of comments and suggestions when I faced some problems.

ACKNOWLEDGEMENTS

I would like to thank my supervisor, Ts. Dr. Norharyati Harum for giving assistant to complete this project successfully.

I would also like to thank my beloved parents who have been giving me support and motivation throughout my project. They motivate and encourage me throughout my projects.

I also like to special thanks to all my friends and lecturers from Faculty of Information and Communication Technology (FTMK), UTeM for their knowledge sharing through this project until I finished this project.

ABSTRACT

This project is about the smart door lock security system using face recognition approach using Haar Cascade Algorithm and Local Binary Patterns (LBP) Algorithm. Different kind of smart door lock has been launched to the market in the recent years but it still needs a lot of improvement from the security aspect. The place is insecure since everyone can access the simple door lock only with door locker or padlock, or smart door lock using other approach such as fingerprint, RFID, and password easily. Therefore, face recognition approach is needed to make it more secure and convenience. The face recognition smart door lock security system will detect and recognize the face appear to the door using Haar Cascade Algorithm and Local Binary Patterns (LBP) Algorithm on Raspberry Pi. Only the person with the face who matched with the dataset in face database can unlock and access the door. Else, a notification will be sent to the owner through Telegram with Wi-Fi to alert owner about the stranger.

ABSTRAK

Projek ini adalah mengenai sistem keselamatan kunci pintu pintar menggunakan pendekatan pengecaman wajah menggunakan Algoritma Haar Cascade Algorithm dan Local Binary Patterns (LBP). Jenis kunci pintu pintar yang berbeza telah dilancarkan ke pasaran dalam beberapa tahun kebelakangan ini tetapi ia masih memerlukan banyak peningkatan dari aspek keselamatan. Sesuatu tempat akan jadi tidak selamat kerana semua orang dapat mengakses kunci pintu yang mudah hanya dengan loker pintu atau kunci, atau kunci pintu pintar menggunakan pendekatan lain seperti cap jari, RFID, dan kata laluan dengan mudah. Oleh itu, pendekatan pengecaman wajah diperlukan untuk menjadikannya lebih selamat dan selesa. Sistem keselamatan kunci pintu pintar pengenalan wajah akan mengesan dan mengenali wajah yang muncul di pintu menggunakan Algoritma Haar Cascade dan Algoritma Corak Binari Tempatan (LBP) pada Raspberry Pi. Hanya orang dengan wajah yang sepadan dengan set data dalam pangkalan data wajah yang dapat membuka kunci dan mengakses pintu. Jika tidak, pemberitahuan akan dikirimkan kepada pemilik melalui Telegram dengan Wi-Fi untuk memberi tahu pemilik mengenai orang asing yang dikesan.

TABLE OF CONTENTS

PAGE

DECLARATION ..... **II**

DEDICATION ..... **III**

ACKNOWLEDGEMENTS ..... **IV**

ABSTRACT ..... **V**

ABSTRAK ..... **VI**

TABLE OF CONTENTS ..... **VII**

LIST OF TABLES ..... **XII**

LIST OF FIGURES ..... **XIII**

LIST OF ABBREVIATIONS ..... **XVI**

CHAPTER 1: INTRODUCTION **1**

1.1 Introduction 1

1.2 Problem Background 2

1.3 Problem Statement (PS) 4

1.4 Project Question (PQ) 5

1.5 Project Objective (PO) 5

1.6 Proposed System 6

1.7 Significant and Contribution of the Projects 7

1.8 Conclusion 8

CHAPTER 2: LITERATURE REVIEW 9

2.1 Introduction 9

2.2 Smart Door Lock System Using Other Approaches 9

2.3 Face Recognition Smart Door Lock System Using Different Algorithm 11

2.4 Technology Pillar for Face Recognition Smart Door Lock Security System 13

2.4.1 Internet of Things (IoT) 13

2.4.2 Machine Learning 14

2.4.3 Connectivity 14

2.5 Tools Integrating Modules for Face Recognition Smart Door Lock Security System 15

2.5.1 Raspberry Pi 15

2.5.2 Open Source Computer Vision Library (OpenCV) 15

2.5.2.1 Haar Cascade Algorithm 16

2.5.2.2 Local Binary Patterns (LBP) Algorithm 16

2.5.3 Python 17

2.5.4 Telegram 17

2.6 Related Work / Previous Work 17

2.7 Critical Review of Current Problem and Justification 23

2.8 Proposed Solution/Further Project 27

2.9 Conclusion 29

CHAPTER 3: METHODOLOGY 30

3.1 Introduction 30

3.2 Project Methodology 30

3.2.1 Planning and Requirement Analysis 31
3.2.2 Quick Design of System Architecture 32
3.2.3 Prototype Development 33
3.2.4 Customer Evaluation 33
3.2.5 Prototype Rectification 33
3.2.6 Engineer Product 33
3.3 Project Schedule and Milestones 34
3.4 Conclusions 37
CHAPTER 4: ANALYSIS AND DESIGN 38
4.1 Introduction 38
4.2 Problem Analysis 38
4.3 Requirement Analysis 39
4.3.1 Data Requirement 39
4.3.2 Functional Requirement 40
4.3.3 Hardware and Software Requirement 40
4.3.3.1 Hardware Requirement 40
4.3.3.2 Software Requirement 50
4.4 High-Level Design 53
4.4.1 System Architecture, Physical Design 54
4.4.2 User Interface Design 54
4.5 Conclusion 61
CHAPTER 5: IMPLEMENTATION 63
5.1 Introduction 63
5.2 Hardware and Software Development Environment Setup 63

5.2.1 Hardware Setup 63
5.2.2 Software Setup 65
5.3 Software Configuration Setup 72
5.3.1 Face image capture configuration 72
5.3.2 Face recognition configuration 75
5.3.3 Solenoid door lock configuration 76
5.3.4 Telegram configuration 77
5.4 Conclusion 78
CHAPTER 6: TESTING 79
6.1 Introduction 79
6.2 Test Plan 79
6.2.1 Test Organization 79
6.2.2 Test Environment 80
6.2.3 Test Schedule 80
6.3 Test Strategy 80
6.4 Test Design 81
6.4.1 Test Description 81
6.4.1.1 Functionality Test 81
6.4.1.2 Performance Test 83
6.5 Test Results and Analysis 87
6.5.1 Functionality Test 87
6.5.1.1 Face Detection and Recognition Test 87
6.5.1.2 Telegram Notification Test 88
6.5.1.3 Solenoid Door Lock Test 89

6.5.2 Performance Test 90
6.5.2.1 System Usability Scale (SUS) 94
6.5.3 False Test 97
6.5.3.1 Face from Picture 98
6.5.3.2 No Internet Connection 99
6.6 Conclusion 100
CHAPTER 7: PROJECT CONCLUSION 102
7.1 Introduction 102
7.2 Project Summarization 102
7.2.1 Project Objective 102
7.2.2 Project Strength and Weakness 103
7.2.2.1 Project Strength 103
7.2.2.2 Project Weakness 104
7.3 Project Contribution 105
7.4 Project Limitation 105
7.5 Future Work 106
7.6 Conclusion 106
REFERENCES 107

LIST OF TABLES

PAGE

Table 1.1: Problem Statement **5**
Table 1.2: Problem Question **5**
Table 1.3: Project Objective **5**
Table 2.1: Summarization of Smart Door Lock System Using Other Approach **10**
Table 2.2: Summarization of Face Recognition Approach Using Other Algorithm **12**
Table 2.3: Discussion of related work **18**
Table 2.4: Summarization of on Algorithm Applied in face Recognition System **22**
Table 2.5: Problem and solution based on the previous work. **24**
Table 3.1: Milestone of Face Recognition Smart Door Lock Security System Using Haar Cascade Algorithm and LBP Algorithm **34**
Table 3.2: Gantt Chart of Face Recognition Smart Door Lock Security System Using Haar Cascade Algorithm and LBP Algorithm **37**
Table 4.1: Summarization of hardware and usage **49**
Table 6.1: Face Detection and Recognition Test **81**
Table 6.2: Telegram Notification Test **82**
Table 6.3: Solenoid Door Lock Test **82**
Table 6.4: General guideline on the interpretation of SUS score **86**
Table 6.5: Responses of SUS **94**
Table 6.6: SUS Raw Score and Final Score **95**
Table 7.1: Project Summarization **102**
Table 7.2: Comparison of the price with other smart door lock systems **104**

LIST OF FIGURES

PAGE

Figure 1.1: Concept Diagram of Face Recognition Smart Door Lock Security System Using Haar Cascade Algorithm and LBP Algorithm 7
Figure 2.1: Overview Diagram for Face Recognition Smart Door Lock Security System 11
Figure 2.2: Types of features in Haar Cascade classifier algorithm 16
Figure 2.3: Concept Diagram of Face Recognition Smart Door Lock Security System Using Haar Cascade Algorithm and LBP Algorithm 27
Figure 3.1: Prototyping Model 31
Figure 4.1: Block Diagram of Face Recognition Smart Door Lock Security System Using Haar Cascade Algorithm and LBP Algorithm 39
Figure 4.2: Context Diagram of Face Recognition Smart Door Lock Security System Using Haar Cascade Algorithm and LBP Algorithm 40
Figure 4.3: 22" LED Monitor S22F350FHE Monitor (Samsung) 41
Figure 4.4: Raspberry Pi 4 Computer Model B 2GB RAM 41
Figure 4.5: Official RPi 15W (5V/3A) PSU USB C UK Plug-White 42
Figure 4.6: Micro SD Card 16GB 43
Figure 4.7: Micro HDMI to HDMI cable 43
Figure 4.8: 12VDC Solenoid Door Lock 44
Figure 4.9: 3x18650 Battery Holder 45
Figure 4.10: 3.7V 2000mAh Li-ion Battery 45
Figure 4.11: 1 CH Active H/L 5V Optocoupler Relay Module 46
Figure 4.12: CH-2 Quick Wiring Terminal Press Type 47
Figure 4.13: Maker pHAT 47
Figure 4.14: Logitech Webcam C270 48

Figure 4.15: Male to Female Jumper Wires **49**
Figure 4.16: Raspberry Pi OS (Imager) **50**
Figure 4.17: Python IDE **51**
Figure 4.18: OpenCV **52**
Figure 4.19: Telegram **53**
Figure 4.20: Physical Design of Face Recognition Smart Door Lock Security System Using Haar Cascade Algorithm and LBP Algorithm **54**
Figure 4.21: Flow Chart of Face Recognition Smart Door Lock Security System Using Haar Cascade Algorithm and LBP Algorithm **55**
Figure 4.22: Flow Chart of Stages of Haar Cascade Algorithm **56**
Figure 4.23: Flow Chart of LBP Algorithm **57**
Figure 5.1: Installation of hardware **65**
Figure 5.2: Download Raspberry Pi OS **66**
Figure 5.3: Installing Raspberry Pi Imager **66**
Figure 5.4: Welcome Message of Raspberry Pi OS **67**
Figure 5.5: Country, Language and Time zone Setting **67**
Figure 5.6: Wi-Fi Selection **68**
Figure 5.7: Enter Wi-Fi Password **68**
Figure 5.8: Setup Complete Message **69**
Figure 5.9: Face database created **73**
Figure 5.10: Create Telegram Bot **78**
Figure 6.1: SUS questionnaire (General Information) **84**
Figure 6.2: SUS questionnaire (Item 1 to 3) **85**
Figure 6.3: SUS questionnaire (Item 4 to 6) **85**
Figure 6.4: SUS questionnaire (Item 7 to 10) **86**
Figure 6.5: Face captured matched with the dataset (face recognized). **87**
Figure 6.6: Face captured not matched with the dataset (face not recognized) . **88**
Figure 6.7: Telegram Notification about stranger **89**
Figure 6.8: The solenoid door lock is unlocked. **90**
Figure 6.9: The solenoid door lock is locked. **90**
Figure 6.10: Multiple datasets added. **91**
Figure 6.11: The face of respondent (Koh Wei Kiat) is recognized. **91**
Figure 6.12: The face of respondent (Lily Yoo) is recognized. **92**
Figure 6.13: The face of respondent (Sam) is recognized. **92**

Figure 6.14: The face of respondent (Tee Pock Yuan) is recognized. 93
Figure 6.15: The face of respondent (Mandy) is recognized. 93
Figure 6.16: The face of respondent (Mah Kim Yao) is recognized. 94
Figure 6.17: Picture of user is tested. 98
Figure 6.18: Picture of stranger is tested. 98
Figure 6.19: Notification received when stranger detected. 99

LIST OF ABBREVIATIONS

Wi-Fi	**-**	**Wireless Fidelity**
RFID	**-**	**Radio Frequency Identification**
FYP	**-**	**Final Year Project**
OpenCV	**-**	**Open Source Computer Vision Library**
LBP	**-**	**Local Binary Patterns**
IoT	**-**	**Internet of Things**
IEEE	**-**	**Institute of Electrical and Electronics Engineers**
SDLC	**-**	**Software Development Life Cycle**
V	**-**	**Volt**
LED	**-**	**Light Emitting Diode**
TV	**-**	**Television**
GUI	**-**	**Graphical User Interface**
GPIO	**-**	**General Purpose Input/Output**
USB	**-**	**Universal Serial Bus**
HDMI	**-**	**High-Definition Multimedia Interface**
DC	**-**	**Direct Current**
OS	**-**	**Operating System**
IDE	**-**	**Integrated Development Environment**
SUS	**-**	**System Usability Scale**

CHAPTER 1: INTRODUCTION

1.1 Introduction

Nowadays, there are various kind of smart home security devices exist on the market and sometimes it can be overwhelming. One of these devices is smart door lock. A smart door lock is a replacement for simple door lock with only door locker or padlock. It is a door locks without using the physical key. It needs Bluetooth or Wi-Fi connection for it to works. There are many types of smart door lock system implemented using different approach available on the market such as fingerprint, password, face recognition or RFID based door lock (Al-Tuma, K. A. H (2019)). Different kinds of smart door lock provide its user with different advantages. In this project, face recognition approach is used to develop a face recognition smart door lock security system.

Our face is the unique biometric characteristic to an individual to recognize a person. With the advancement of technology, many systems are implemented with the face recognition approach. To identify a person, the face captured are matched with the face image stored in the database. The face recognition approach has widely used in various field such as forensic and security. Nowadays, the simple door lock with only door locker or padlock, or smart door lock using other approach such as fingerprint, RFID and password that install on the premise such as home or office still need a lot of improvement from the aspect of security and convenience of the owner. This is because the door lock is not secure enough to protect the user and it will also cause many inconveniences for the owner in their life indirectly. Therefore, the face

recognition smart door lock security system is primarily driven by the growing need for effective solutions in increasing the security and convenience of the user.

OpenCV is a free and open source software library for programming functions especially for machine learning, image processing, and computer vision. It is originally created at Intel by Gary Bradsky, and later maintained by Willow Garage with Gary Bradsky and Vadim Pisarevsky (Kulhary, R (2021); OpenCV (2021)). It was built so that the computer vision applications have a common infrastructure and provide an acceleration to the use of machine perception in the commercial products. OpenCV consists of C++, Python, Java and MATLAB interfaces and it supports most of the common operating system such as Windows, Linux, Android, and Mac OS. It also provides many algorithms in the library such as Haar Cascade Algorithm and LBP Algorithm in OpenCV (Emami, S., Suciu, V.(2012)). The development of OpenCV has enabled computer vision field to keep growing rapidly. This is because OpenCV has prepared the computer vision and machine learning infrastructure that was previously unavailable in many research labs for many people no matter students or professionals. This will increase their productivity in computer vision field since they are able to develop different project and do the research mainly on real-time vision more efficiently.

The demand and application of face recognition smart door lock security system keeps growing rapidly due to the outbreak of COVID-19. This is because the pandemic has surged the demand for touch-free smart door lock security system (Fortune Business Insights 2021; Transparency Market Research 2020). By installing or implementing face recognition smart door lock security system, it provides many conveniences to the user. Instead of worrying about the security aspect and inconveniences caused by the simple door lock with only door locker or padlock, or smart door lock using other approach such as fingerprint, RFID, and password, installing a face recognition door lock system would be the effective solution.

1.2 Problem Background

Nowadays, many people pursue for high efficiency and quality life. They keep finding the solution to make their life better. The implementation of face recognition

smart door lock security system using Haar Cascade Algorithm and LBP Algorithm in their premise is one of the examples of the solution. Development of face recognition door lock system not only able to improve the quality of life of the user, but it also makes the premises or places of the user more secure. Although we are used to the simple door lock only with door locker or padlock, or smart door lock using other approach such as fingerprint, RFID, and password, but these approaches have many disadvantages.

First, security is always the top concern for every individual. A secure and safe premises or home is the necessity of every individual especially those who are working or outstation most of the time. Due to the simplicity of the door lock, the risk for home invasion crimes to be occurred will be increased because the premise owner or the homeowner left their home unattended most of the time. The place is insecure since no matter owner or stranger can access the simple door lock only with door locker or padlock, or smart door lock using other approach such as fingerprint, RFID, and password easily. For example, if owner install the simple door lock, other people such as burglar can break into the house easily because the door lock are installed without user authentication and authorization or using weak authentication method. This will put the owner or the property of the owner in potentially dangerous situation.

In addition, the door lock with physical key makes the user's life inconvenient. As we know, the physical key can be unique keys for a door lock and for different door locks, it has different keys. The user must grab a bunch of keys when they want to go out. This will cause the user to face the possibility of the physical key being lost, misplaced, or stolen. They will be unable to unlock the door because they must find the key, get a new key, or even replace the whole locking mechanism. This will waste their time and thus decrease their efficiency in their daily life. Moreover, they must worry about their place situation when they are working or outstation during daytime, and their own safety during the night time because burglary and robbery seemed to peak when nobody at home or when the homeowner has slept. They will be unable to enjoy life events and even sleep peacefully. As a result, their quality of life will be affected.

Next, most of door lock system especially simple door lock are unable to notify user when stranger or intruder detected. The user is unable to keep track of surroundings of their premise and allocate the stranger that step in front of the door. When there are intruders try to access into their place, even though intruders are detected from the system, they do not know about it since no notification are sent to notify the user. This will cause them unable to protect themselves or protect their properties because they are not notified about the situation occurred and cannot act immediately.

1.3 Problem Statement (PS)

The place is insecure since no matter owner or stranger can access the simple door lock only with door locker or padlock, or smart door lock using other approach such as fingerprint, RFID, and password easily. If owner install the simple door lock only with door locker or padlock, or smart door lock using other approach such as fingerprint, RFID and password, other people such as burglar can break into the house easily because the door lock are installed without user authentication and authorization or using weak authentication method. This will put the owner or the property of the owner in potentially dangerous situation. In addition, the door lock with physical key makes the user's life inconvenient. The user must grab a bunch of keys when they want to go out. This will cause the user to face the possibility of the physical key being lost, misplaced, or stolen. They are unable to unlock the door because they must find the key, get a new key, or even replace the whole locking mechanism. It is getting worst if the lost key was found by other people because this will increase the possibility that the intruders to break into their house. Moreover, most of door lock system will not notify user when stranger or intruder detected. The user is unable to keep track of surroundings of their premise and allocate the stranger that step in front of the door. When there are intruders try to access into their place, even though intruders are detected from the system, they do not know about it since no notification are sent to notify the user.

Table 1.1: Problem Statement

PS	Problem Statement
PS1	The place is insecure since no matter owner or stranger can access the simple door lock only with door locker or padlock, or smart door lock using other approach such as fingerprint, RFID and password easily.

1.4 Project Question (PQ)

There are many questions that can relate with face recognition smart door lock security system. The project objectives will answer these questions. These questions are related to the problem statement as stated Table 1.1.

Table 1.2: Problem Question

PS	PQ	Problem Question
PS1	PQ1	How to make the place more secure?
	PQ2	What is the approach used to detect the stranger or intruders?
	PQ3	How to notify user when stranger or intruder detected?

1.5 Project Objective (PO)

This project will develop a face recognition smart door lock system using Haar Cascade Algorithm and LBP Algorithm to detect stranger or intruders and notify user through Telegram when strangers or intruders detected.

Table 1.3: Project Objective

PS	PQ	PO	Project Objective
PS1	PQ1	PO1	Develop a face recognition smart door lock security system.

	PQ2	PO2	Detect the strangers or intruders through face recognition approach which using Haar Cascade Algorithm and LBP Algorithm
	PQ3	PO3	Notify user through Telegram when strangers or intruders detected.

1.6 Proposed System

For this project, a face recognition smart door lock security system will be implemented using Haar Cascade Algorithm and LBP Algorithm. First, the face image will be captured and save as data sets. Secondly, to enable machine learning, the image will be trained to the Haar Cascade Algorithm so that the system can detect the face and save it to the face database. Then, when the face is captured by the webcam, the system will try to compare and match the face detected from webcam with the image in face database using LBP Algorithm. As the result, if the face detected from the webcam is matched with the face image in database, the solenoid door lock will be unlocked. Otherwise, a notification to inform the user about the existence of stranger will be sent through Telegram. The face detection process is using Haar Cascade Algorithm while face recognition process is using LBP Algorithm.

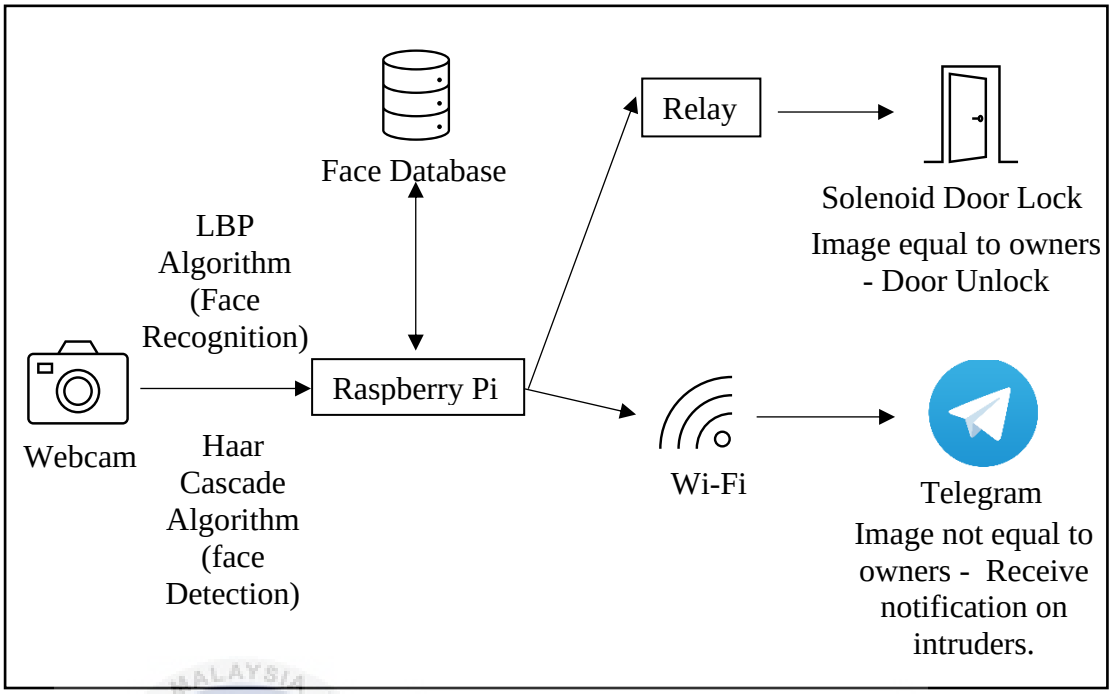


Figure 1.1: Concept Diagram of Face Recognition Smart Door Lock Security System Using Haar Cascade Algorithm and LBP Algorithm

1.7 Significant and Contribution of the Projects

By exploring and implementing this Face Recognition Smart Door Lock Security System Using Haar Cascade Algorithm and LBP Algorithm, I can understand more about the face recognition approach, smart door lock system, Raspberry Pi, and algorithm used by face detection and face recognition such as Haar Cascade Algorithm and LBP Algorithm. I can apply all the knowledge in this area in the future.

In addition, the development of a secure and convenient Face Recognition Smart Door Lock Security System Using Haar Cascade Algorithm and LBP) Algorithm enables users to secure their premises using face recognition approach. This is because only the authenticated and authorized user will be allowed to unlock the solenoid door lock. Moreover, this face recognition smart door lock security system makes user's life more convenient. Users of this system no need to no need to bring a bunch of keys when they go out because they just need to approach to the door and pause for a few second to unlock the door effortless. They also do not need to worry about problem

such as lost of key or the key being stolen since to unlock the door, only face characteristics are required. Furthermore, user of this system will get notify of any strangers or intruders regardless of the time and location. This will enable user to keep track and allocate the stranger that appear in front of the door.

1.8 Conclusion

As the technology development continues to keep advancing, the need for solutions in increasing the security and convenience of the user is growing rapidly. This has led to the development of face recognition smart door lock security system. The development of face recognition smart door lock security system has made consumer's premises more secure and make their life more convenient. In this project, a secure and convenient Face Recognition Smart Door Lock Security System Using Haar Cascade Algorithm and LBP Algorithm will be developed. It can be implemented in a real environment in the future to improve the security aspect of user's premise thereby enhance the quality of the user's life. The next chapter will discuss about the literature review about previous research.

CHAPTER 2: LITERATURE REVIEW

2.1 Introduction

This chapter will discuss about the literature review of the project. The first part of literature review will be some explanation about the face recognition smart door lock security system. Then, this chapter will introduce some technology pillar for face recognition smart door lock security system such as face recognition approach, IoT, machine learning, and wireless connectivity. The tools integrating modules for face recognition door lock security system such as Raspberry Pi, Haar Cascade Algorithm and LBP Algorithm provided by OpenCV, Python, and Telegram are also discussed in this chapter. After that, this some previous research related to this project are studied to identify gaps in the particular field of knowledge, and further look into during current research to prove the latest theories that will provide valuable information to the field. The critical review of current problem and justification are also discussed and justified with some solution. Lastly, Face Recognition Smart Door Lock Security System Using Haar Cascade Algorithm and LBP Algorithm is proposed as the solution.

2.2 Smart Door Lock System Using Other Approaches

Some research shows that there are many smart door lock systems implemented using different approaches.

Table 2.1: Summarization of Smart Door Lock System Using Other Approach

Author/ Year	Approach used by Smart Door Lock System
Prabhakar, A. Y., Oza. S. K., Shrivastava. N., Srivastava. P. and Wadhwa. G. (2019)	Password
Bhat, A. K. and Kini, S. P. (2018)	Password
Tamunowari, M. (2018)	Password
Okpara, C.R. and Ononiwu, (2017)	Password
Rahman, M. M., Ali, M. S. and Akther, M. S. (2018)	Password
Verma, G. K. and Pawan, T. (2010)	RFID
Suhail, R. (2020)	RFID
Sarma, M., Gogoi, A., Saikia, R. and Bora, D.J. (2020)	Fingerprint
Hossain, M. K. (2019)	Fingerprint

2.3 Face Recognition Smart Door Lock System Using Different Algorithm

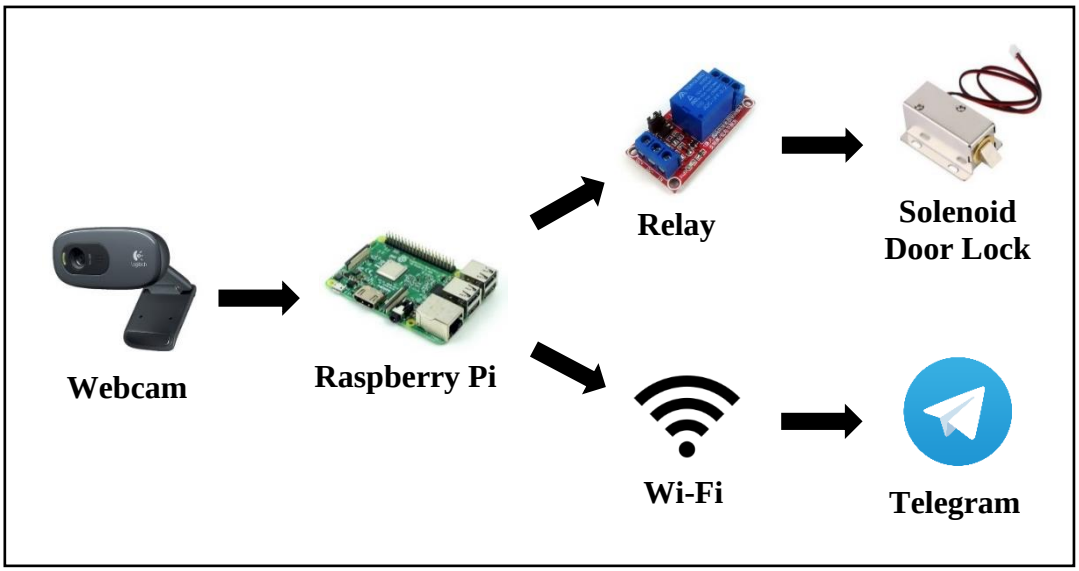


Figure 2.1: Overview Diagram for Face Recognition Smart Door Lock Security System

A face recognition smart door lock security system is one of the solutions for smart security aspect that using face recognition technology by implementing devices such as camera. It is smart door lock system that can identify and verify human identification using face recognition approach and send notification to the user. The face recognition approach compares or matches the detected faces against the faces in face database to identify and verify the human identification. The initial step for face recognition smart door lock security system is face detection. The system will capture the image of the user using webcam. Then the image will be processed and trained with some algorithm so that the system is able to identify the face of the user (Gunawan, T. S., Gani, M. H. H., Rahman, F. D. A. and Kartiwi, M.(2017)). Next step is face recognition. When the webcam detects the face of the person that appear in front of the door, the face recognition smart door lock security system will capture the face detected and compare it to the face database. If the faces detected are recognized by the system, the system will send command to the relay to unlock the solenoid door lock. Otherwise, the door lock will keep locked and a notification will be sent to the user. The face recognition smart door lock security system ensure that only authenticated and authorized user can access the premises. It improves the security aspect of the premises and brings a lot of convenience to the user (Najmurrokhman, A., Kusnandar, K., Krama, A. B., Djamal, E. C. and Rahim, R. (2018)).

There are some papers discussed about various kind of algorithm used in face recognition technology.

Table 2.2: Summarization of Face Recognition Approach Using Other Algorithm

Author/ Year	Algorithm	Description
Faisal, F., Hossain, S. A. (2019)	• Viola-Jones Face Detection Method and AdaBoost Algorithm • Principal Component Analysis and Eigenfaces Method	• First, the Viola-Jones face detection algorithm is used to convert the input image into a new image representation called an integral image that allows a very fast feature evaluation. The used features are comparable to Haar-basis functions. Then, AdaBoost, a machine learning boosting algorithm is used to select a small number of important features by constructing a classifier. By using a weighted combination of weak classifiers, a strong classifier can be constructed. Last, the cascaded classifier is used to determine whether a given sub-window classifier is a face or not. • The Principal Component Analysis (PCA) method is used to extract the relevant features of facial images. PCA use Eigenfaces to recognize the face. Eigenfaces are principal components of the distribution of faces, or equivalently, the Eigen vectors of the covariance matrix of the set of the training images, where an image with N by N pixels is considered as a point in N2 dimensional space.

Zedan, M. J. M., Al-Jbaar, M. (2020)	Convolutional Neural Network (CNN)	CNN is used to determines the faces and facial landmark points, extracts distinguishing features from the appropriate face area, and classifies the facial expressions provided by the real-time camera. Furthermore, the histogram of directional gradients features is extracted from the effective facial patches instead of the whole face making the suggested system robust with the scale and pose changes.
Anggo, M., Arapu, L. (2018)	Fisherfaces	The fisherfaces algorithm is used in image recognition based on the reduction of face space dimension using Principal Component Analysis (PCA) method, then apply Fisher's Linear Discriminant (FDL) method or also known as Linear Discriminant Analysis (LDA) method to extract the feature of image characteristic. In this research, fisherfaces is used for image recognition while minimum euclidean is used for identification or matching face image.

2.4 Technology Pillar for Face Recognition Smart Door Lock Security System

There are some important technologies and components applied in the Face Recognition Smart Door Lock Security System. For example, IoT technology, machine learning, wireless connectivity, and tools integrating modules.

2.4.1 Internet of Things (IoT)

IoT is a huge network that links various gadgets together. These gadgets gather data, stock data, and transmit data over a network without requiring human-to-human or human-to-computer interaction (Radzi, S. A., Alif, M. K.

M. F., Athirah, Y. N., Jaafar, A. S., Norihan, A. H. and Saleha, M. S. (2020)). In recent years, many research and studies were carried out and this has led to the implementation of IoT for various purposes included the implementation of a face recognition smart door lock security system. It has become a crucial part of the face recognition smart door lock security system. The main objective of IoT is to leverage the power of communication or internet connectivity and computation by using the existing network or some custom wireless or wired network to the pre-existing or daily use real-world things. IoT has improved the traditional door lock by embedding image sensors and actuators such as camera into the door lock to enhance the intelligence of the door lock system.

2.4.2 Machine Learning

Nowadays, the machine learning field is evolving continuously, and it is developed in the face recognition smart door lock security system. It is a subset of artificial intelligence. It can learn by themselves from the experience and make their own predictions and decisions. Machine learning enables the face recognition smart door lock security system to learn from the data such as the image of face that captured from the webcam by train the data and act or make decisions based on the data. The whole processes are with minimal human intervention. It improved the quality of the user's life by making their life easier.

2.4.3 Connectivity

In this project, a wireless networking technology that is Wi-Fi is used to allows the connection and communication of the devices such as computers (laptops, desktops, and Raspberry Pi), mobile devices (smart phones), and other equipment (webcam, relay module and solenoid door lock) through the Internet by using IEEE 802.11 standards. To enable the internet connectivity, the user connects to the wireless router by accessing the Wi-Fi.

2.5 Tools Integrating Modules for Face Recognition Smart Door Lock Security System

The tools integrating modules used in this Face Recognition Smart Door Lock Security System are Raspberry Pi, OpenCV, Python and Telegram. The detail of these tools integrating modules are discussed at below.

2.5.1 Raspberry Pi

A United Kingdom charity, Raspberry Pi Foundation has made a series of single-board computers for the purpose of making computing education easier. It is a tiny and affordable computer that enable us to learn programming, build projects using hardware, implement home automation, implement Kubernetes clusters, edge computing, and even apply them in industrial applications. Raspberry Pi require Raspberry Pi OS to operate. The Raspberry Pi OS runs Linux, but it also provides a set of GPIO pins, allowing us to control electronic components for physical computing and explore the IoT.

2.5.2 Open Source Computer Vision Library (OpenCV)

OpenCV is a free and open source software library for programming functions especially for machine learning, image processing, and computer vision. OpenCV consists of C++, Python, Java and MATLAB interfaces and it supports most of the common operating system such as Windows, Linux, Android, and Mac OS. The development of OpenCV has enabled computer vision field to keep growing rapidly. This is because OpenCV has prepared the computer vision and machine learning infrastructure that was previously unavailable in many research labs for many people no matter students or professionals. This will increase their productivity in computer vision field since they are able to develop different project and do the research mainly on real-time vision more efficiently.

2.5.2.1 Haar Cascade Algorithm

In 2001, Paul Viola and Michael Jones has introduced Haar Cascade Algorithm which is able to detect different objects in an image or video since it can be run on real time but was mainly focus on face detection on image or video. To perform face recognition, we need to train the classifier to recognize whether the region contain face(positive image) or not contain face(negative image). And this is how Haar Cascade works. By using this algorithm, it makes our work easier as we just need to load the pre-trained classifier and detect faces in images. This algorithm is using Haar-like feature, which is a set of elementary combinations of areas with different intensity. There are three types of features: edge, linear, and central features. Each feature is computed by minus the total of pixels under the white area from the total of pixels under the black area. The result is then combined into cascading structure to show whether the region is face or not.

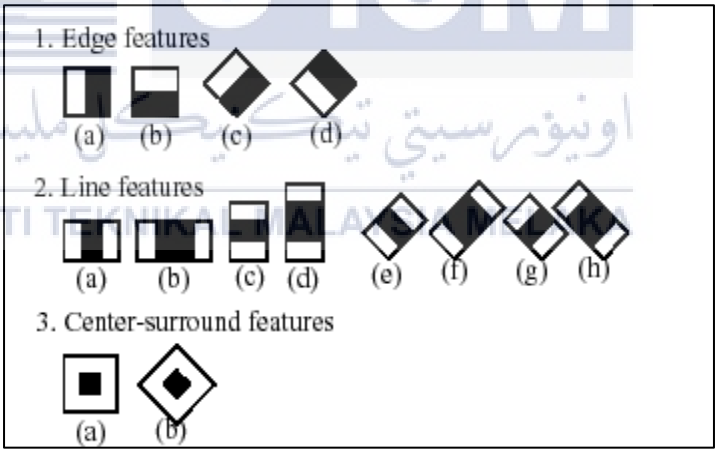


Figure 2.2: Types of features in Haar Cascade classifier algorithm

2.5.2.2 Local Binary Patterns (LBP) Algorithm

LBP Algorithm describe the neighborhood of image elements using binary codes. This algorithm is normally applied to identify the local properties and the characteristics of individual parts of the image. The LBP detects microstructures such as edges, lines, spots, flat areas,

which can be estimated by the histogram. Each pixel of the image or video is compared with its neighboring pixels and composed as the relative values.

2.5.3 Python

Python is one of the popular programming languages which can be used on a server-side to develop web application, used with software to develop workflows, connect to database system to read, and edit files, used to process big data, and perform complex mathematics. and applied for rapid prototyping, or for production-ready software development. Python is a good choice among all the programming language because it can be run on various operating system such as Windows, Mac OS, Linux, Raspberry Pi, etc. It enables developer to write the program easier by using the syntax that similar with English. Python programs execute on an interpreter system, and thus the prototyping process can be faster. Furthermore, Python can be written in text editor, or in an IDE such as Thonny, Eclipse, Netbeans and etc if larger collections of Python files need to be managed.

2.5.4 Telegram

Telegram is free open online instant messaging application which can use at computer and smartphone. It is supported by different platform such as Windows, Mac OS, iOS, and Android. It enables features such as end-to-end encrypted voice and video calling, file sharing etc. In addition, it also consists of third-party application such as Telegram Bot. Developer can create their own Telegram Bot using BotFather that benefit their invention. They can interact with their bot by giving them command or inline request. The bot will react and give respond to the developer based on the command given.

2.6 Related Work / Previous Work

In this section, the research related to this project is discussed.

Table 2.3: Discussion of related work

Author/ Year	Description
Faisal, F., Hossain, S. A. (2019)	First, the Viola-Jones face detection algorithm is used to convert the input image into a new image representation called an integral image that allows a very fast feature evaluation. The used features are comparable to Haar-basis functions. Then, AdaBoost, a machine learning boosting algorithm is used to select a small number of important features by constructing a classifier. By using a weighted combination of weak classifiers, a strong classifier can be constructed. Last, the cascaded classifier is used to determine whether a given sub-window classifier is a face or not. The Principal Component Analysis (PCA) method is used to extract the relevant features of facial images. PCA use Eigenfaces to recognize the face. Eigenfaces are principal components of the distribution of faces, or equivalently, the Eigen vectors of the covariance matrix of the set of the training images, where an image with N by N pixels is considered as a point in N2 dimensional space.
Zedan, M. J. M., Al-Jbaar, M. (2020)	Convolutional Neural Network (CNN) is used to determines the faces and facial landmark points, extracts distinguishing features from the appropriate face area, and classifies the facial expressions provided by the real-time camera. Furthermore, the histogram of directional gradients features is extracted from the effective facial patches instead of the whole face making the suggested system robust with the scale and pose changes.
Anggo, M., Arapu, L. (2018)	The fisherfaces algorithm is used in image recognition based on the reduction of face space dimension using Principal Component Analysis (PCA) method, then apply Fisher's Linear Discriminant (FDL) method or also known as Linear Discriminant Analysis (LDA) method to extract the feature of

	image characteristic. In this research, fisherfaces is used for image recognition while minimum euclidean is used for identification or matching face image.
Gunawan, T. S., Gani, M. H. H., Rahman, F. D. A. and Kartiwi, M.(2017)	The research proposed a face recognition security system using Raspberry Pi which can be connected to the smart home system. First, the image will be input during the pre-processing phase. Then, the Fisherfaces and Eigenfaces algorithm is applied for feature extraction. The output of face recognition algorithm is connected to the relay circuit and will control the magnetic lock. This research shows the accuracy of this system is about 90%. It also proposed a hierarchical image processing approach which can be include in future research to reduce the training or computation time while improving the accuracy of face recognition.
Prabhakar, A. Y., Oza. S. K., Shrivastava. N., Srivastava. P. and Wadhwa. G. (2019)	The project proposed a password based door lock system using Arduino UNO which aim to secure the place. To unlock the door, the user needs to enter the password saved in the database. If the password entered matched the password in database, the servo motor will be deflected, and the door will be unlocked for 10 seconds. If the password entered is incorrect, the buzzer will be sounded to indicate that the password is invalid. The status of the door lock will be sent through SMS to the smartphone by GSM modem. If user attempted 3 wrong passwords, the lock will be locked and show message "NO MORE TRIALS ACCESS DENIED" to stop user from entering the password. At this time, to unlock the door, the user needs to send a pre-defined code to the sim that connected in GSM modem through SMS by using the smartphone.
Bhat, A. K. and Kini, S. P. (2018)	The research proposed an electronically controlled door locking system using Arduino and IoT. The users have to insert the unique password, the system will determine which user accessed the door lock and send the data to the cloud. If

	the password is matched, the door will be unlocked. By default, after the password was matched, the door will be unlocked for 2 minutes. If user attempted password wrongly for 4 times, the system will be blocked and has to be reset by the primary user.
Tamunowari, M. (2018)	The research designed and constructed an electronic door access key which consists of three part that is input, control and output. In order to gain access of the door, the correct password have to be pressed on the 4x4 hexadecimal keypad. Then, Arduino UNO (ATmega-328) will be used for the control purpose such as power, pins, LED indicator, and etc. As the output, the lock will be unlocked is the password is correct else the buzzer will be activated for 5 second if the password is wrong. The LCD will display the message to show whether the access is granted or denied.
Okpara, C.R. and Ononiwu, (2017)	The paper proposed an automated door lock that implement using Arduino UNO R3 microcontroller. This system is to enable the lecturer to control the door lock during the lecture period. First of all, the user needs to press the password. If the password pressed is correct, the green LED will be blinked, and the mechanical door will be unlocked. Otherwise, the user needs to re-enter the correct password. After the door has unlocked for 30 seconds, the red LED will be blinked, and the mechanical door will be closed.
Rahman, M. M., Ali, M. S. and Akther, M. S. (2018)	The research proposed a system that implements a password protected electronic lock. This system is to improve the security aspect and ensure the cost efficiency in the implementation of password protected electronic lock. The keypad will be used to receive the input (password). The PIC 18F4552microcontroller will be used as a controlling unit. The 16x2 LCD will be used as the output device. To unlock the door, a correct password has to be inserted. Only

	authorized person that have fixed security code along with the old password can change the password.
Verma, G. K. and Pawan, T. (2010)	The study proposed a digital security system that contains door lock system using passive type of RFID technology. The system database consists of all the data or information regarding the user. A new user needs to register to the system and their information will be stored in RFID tag. The registered user appears to the entry point and insert their RFID tag into the reader. The system verifies whether they are registered user or intruders. After successfully authentication and authorization, the door is unlocked for the user to enter and close automatically after a specified time interval. The check-in information such as date, time, user is stored in the database. A log is also generated by the system according to check-in information.
Suhail, R. (2020)	In this paper, a RFID based door access control using Arduino is implemented. When the power supply switched on, the system turns on the white LED lights automatically. The RFID tag need to be insert into the RFID reader. The RFID reader reads the data in the tag and detects the data in the database. If the data in the tag matched the data in database, the door opens, the green LED lights up and the buzzer sounds. Otherwise, the door remains closed, red LED lights up and the buzzer sounds. By using the Arduino UNO board, the model works more efficiently.
Sarma, M., Gogoi, A., Saikia, R. and Bora, D.J. (2020)	This paper focus on the fingerprint based door access system that develop using Arduino. The system is implemented with the features which increase the security level. User scan their fingerprint using the fingerprint scanner. If the fingerprint is recognized, the information of the user will be recorded, and the servo motor is activated to unlock the hook lock connected to it. A welcome message will be displayed. If fingerprint is not recognized, user is required to scan again and if the

	attempt is over 3 times, the system will store the unrecognized fingerprint and the photograph of the unauthorized user. A message will be sent to the owner and the error message will be displayed to the user
Hossain, M. K. (2019)	This project proposed a door lock system embedded with a fingerprint interface. First, the fingerprint sensor will be installed in the door panel, facing outer side of the door. Only people inside the door can control the system. Some latches will be fixed inside the door panel, so that the thickness of the door can support the latch's strength by dividing the force among them to prevent intruders break into the room. The fingerprint sensor will scan the fingerprint of the user and send it to the microcontroller to match with its records. If the fingerprint matches with the fingerprint inside the memory of the microcontroller, the microcontroller will lock or unlock the latch, according to its current state. If the fingerprint is unrecognize, the buzzer will activate, and the user will have to scan their fingerprint again. If wrong fingerprints are attempt more than five times, the system will send a message to the owner to alert them about the situation. The system will also enter into a secure state where it will continue to sound the buzzer to alert the neighbours and grab their attention. The system will be reset once a recognized fingerprint is matched.

Table 2.4: Summarization of on Algorithm Applied in face Recognition System

Author/ Year	Microcontroller	Approach	Algorithm
Faisal, F., Hossain, S. A. (2019)	Raspberry Pi, Arduino UNO	Face Recognition	Viola-Jones face detection algorithm, Eigenfaces
Zedan, M. J. M., Al-Jbaar, M. (2020)	Raspberry Pi	Face Recognition	Convolutional Neural Network (CNN)

Anggo, M., Arapu, L. (2018)	Raspberry Pi	Face Recognition	Fisherfaces
Gunawan, T. S., Gani, M. H. H., Rahman, F. D. A. and Kartiwi, M.(2017)	Raspberry Pi	Face Recognition	Fisherfaces and Eigenfaces
Prabhakar, A. Y., Oza. S. K., Shrivastava. N., Srivastava. P. and Wadhwa. G. (2019)	Arduino UNO	Password	Not stated
Bhat, A. K. and Kini, S. P. (2018)	Arduino Mega	Password	Not stated
Tamunowari, M. (2018)	Arduino UNO	Password	Not stated
Okpara, C.R. and Ononiwu, (2017)	Arduino UNO	Password	Not stated
Rahman, M. M., Ali, M. S. and Akther, M. S. (2018)	Peripheral Interface Controller (PIC) 18F4552	Password	Not stated
Verma, G. K. and Pawan, T. (2010)	Not stated	RFID	Not stated
Suhail, R. (2020)	Arduino UNO	RFID	Not stated
Sarma, M., Gogoi, A., Saikia, R. and Bora, D.J. (2020)	Arduino UNO	Fingerprint	Not stated
Hossain, M. K. (2019)	ATMEGA 328P	Fingerprint	Not stated

2.7 Critical Review of Current Problem and Justification

This section discussed and justified the problem of the existed research. The solution is proposed to solve the problem.

Table 2.5: Problem and solution based on the previous work.

Face Recognition Smart Door Lock Security System Using Haar Cascade Algorithm and Local Binary Patterns (LBP) Algorithm			
Author/ Year	**Problem**	**Justification**	**Solution**
Gunawan, T. S., Gani, M. H. H., Rahman, F. D. A. and Kartiwi, M.(2017).	The face recognition security system which can be connected to the smart home system is implemented without sending any notification to the user if strangers or intruders detected.	Although smart home enable homeowner to control and monitor the house, it is not secure enough since the user unable to get alert on any important event or unusual activity. This would cause them to miss the right timing to take action immediately if any events occurred. For example, if the intruders break into the house, the properties of the homeowner even homeowner themselves will be in dangerous situation because they are unalert of the circumstance.	To improve the proposed system, a notification mechanism needs to be developed in the system to notify or alert the homeowner if the system detects the stranger.
Prabhakar, A. Y., Oza. S. K., Shrivastava. N., Srivastava. P. and Wadhwa. G. (2019).	In these papers, the door lock systems proposed are using a weak authentication	The intruders are able to attack the password-protected systems easily with different approach, the most common approach is through password guessing. This	To solve this problem, smart door lock security system using face recognition approach should

Bhat, A. K. and Kini, S. P. (2018) Tamunowari, M. (2018) Okpara, C.R. and Ononiwu, (2017) Rahman, M. M., Ali, M. S. and Akther, M. S. (2018)	method which is password.	is because people normally set their password using their name, birthday and etc. Intruders can guess their password and access the door lock system without any difficulties. In addition, the intruders or burglars can be their neighbour or stranger that saw the password of the user coincidence. They can remember the password and get into the premises unnoticeable.	be implemented to increase the difficulties for strangers or intruders to get into the house.
Verma, G. K. and Pawan, T. (2010) Suhail, R. (2020)	The door lock systems implemented are using RFID technology have scanning issues and it is not convenient for some reason.	RFID reader can scan most of the non-metallic material, but the metallic material and liquid can affect the signal. As a result, user need to consume more time to access the door lock system in order to solve this problem. In addition, tag collision and reader collision may occur if multiple tags scanned at the simultaneously or interference of multiple	Face recognition approach can be deployed in the system to make user's life more convenient. This is because those scanning issues will not occur if face recognition approach is used. Furthermore, face recognition smart door lock system only requires face

		reader's signal. Moreover, user unable to unlock the door if the token or the key card are lost, misplaced or stolen.	characteristic to unlock the door.
Sarma, M., Gogoi, A., Saikia, R. and Bora, D.J. (2020) Hossain, M. K. (2019)	The fingerprint based smart door lock systems developed are not hygienic, inefficient, and uncomfortable.	As the user need to press their finger to the sensor or the scanner to scan their fingerprint, the user will leave the germs or bacteria on the scanner or sensor. In addition, the fingerprint will not work if the users are wearing hand gloves, or their fingers are dirty or wet. Moreover, different people have different finger size, and the scanner size cannot fit everybody's finger. This cause user feels uncomfortable to use the scanner.	To mitigate these problem, face recognition smart door lock security system can be developed because face recognition is definitely a better option as it is more hygienic, efficient and easy to use.

2.8 Proposed Solution/Further Project

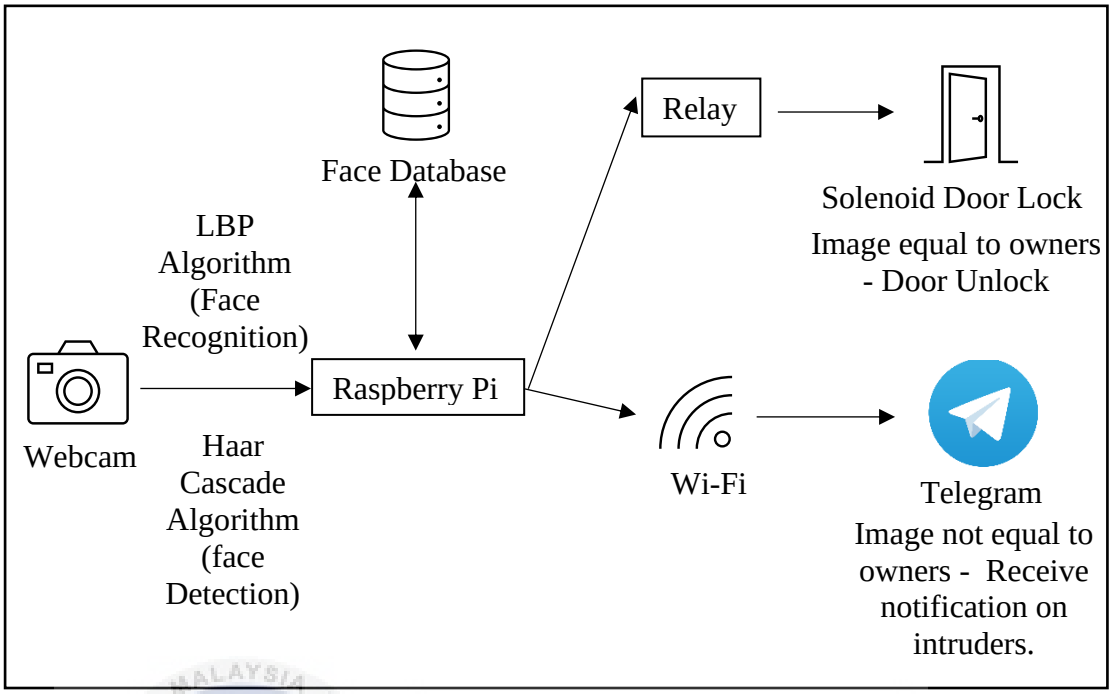


Figure 2.3: Concept Diagram of Face Recognition Smart Door Lock Security System Using Haar Cascade Algorithm and LBP Algorithm

There are a lot of problem that will occur in previous work such as no notification function, weak authentication method, scanning issue, inconvenience, unhygienic, inefficient, and uncomfortable for user. All these problems can be mitigated by develop a face recognition smart door lock security system. According to Gunawan, T. S., Gani, M. H. H., Rahman, F. D. A. and Kartiwi, M.(2017), a notification function will be implemented in the face recognition smart door lock security system. In Prabhakar, A. Y., Oza. S. K., Shrivastava. N., Srivastava. P. and Wadhwa. G. (2019), Bhat, A. K. and Kini, S. P. (2018), Tamunowari, M. (2018), Okpara, C.R. and Ononiwu, (2017), and Rahman, M. M., Ali, M. S. and Akther, M. S. (2018), password authentication method can be replaced by implementing the system using face recognition method. The scanning issue and inconvenience proposed by Verma, G. K. and Pawan, T. (2010) and Suhail, R. (2020) will not occur if the smart door lock security system using face recognition is developed. Using face recognition is definitely more hygienic compared to fingerprint in research by Sarma, M., Gogoi, A., Saikia, R. and Bora, D.J. (2020) and Hossain, M. K. (2019) because user no need to have any contact with the devices.

The proposed system is a face recognition smart door lock security system using Haar Cascade Algorithm and LBP Algorithm. It is a smart door lock security system that can identify and verify human identification using face recognition approach and send notification to the user through Telegram. First, the face recognition smart door lock security system will capture the image of the user using webcam. Then the image will be processed and trained by Haar Cascade Algorithm provided by OpenCV on Raspberry Pi so that the system can detect the face of the user to identify and verify the human identification. Then, the image will be saved in the face database. After face detection part implemented, when a person appears in front of the door, the system will capture the face of that person and try to match with the dataset in face database using and LBP Algorithm. If the face of that person matched with the face dataset in face database of the system, the system will send command to the relay to unlock the solenoid door lock. Otherwise, the solenoid door lock will keep locked and a notification will be sent through Telegram to notify premise owner that there is stranger appear in front of their door. The face recognition smart door lock security system ensure that only authenticated and authorized user can access the premises.

2.9 Conclusion

In this chapter, the literature review discussed about the existing research to produce better research. The demand and application of face recognition keeps growing rapidly. Therefore, in this project, a face recognition smart door lock security system is implemented by applying Haar Cascade Algorithm and LBP Algorithm. By installing or implementing this system in the premises, it improves the security aspect and provides many conveniences to the user in their life. It plays an important role in advancement of technology and society by improving the security and quality of the user's life. The next chapter will focus on methodology applied in this project.

CHAPTER 3: METHODOLOGY

3.1 Introduction

This chapter will discuss about the methodology that will be used in this project. Methodology is the concrete procedures used to determine, choose, and analyze the information about a topic. It defines a collection of methods and processes for accomplishing the project. It usually contains the steps or activities for each phase of project cycle to make sure the project is following the schedule.

The first topic of methodology is the project methodology. Methodology will discuss about the SDLC that applied in this project to implement a practical system that satisfies or exceeds customer requirements, accomplish the task within the stipulated time and cost estimation. In addition, the project schedule and milestone will be tabulated in the table as the guideline for the project timeline. Lastly, the overall content of this chapter will be concluded and summarized in conclusion.

3.2 Project Methodology

Project methodology is important in a project as it will be used as the guideline for the project throughout the process of implementation of the system. It is used to ensure that the system is developed in a proper manner and flow so that a practical system can be delivered. There are many types of SDLC. In this project, prototyping model is applied as the project methodology. Prototyping model refers to the activity included in developing, testing, and refining the prototype until the complete system are implemented. It helps the developers to understand the customer requirements. The stages involved in the prototyping model are Planning and Requirement Analysis,

Quick Design of Product Architecture, Prototype Development, Customer Evaluation, Prototype Rectification and Engineer Product. Each stage will be discussed in detail in below section.

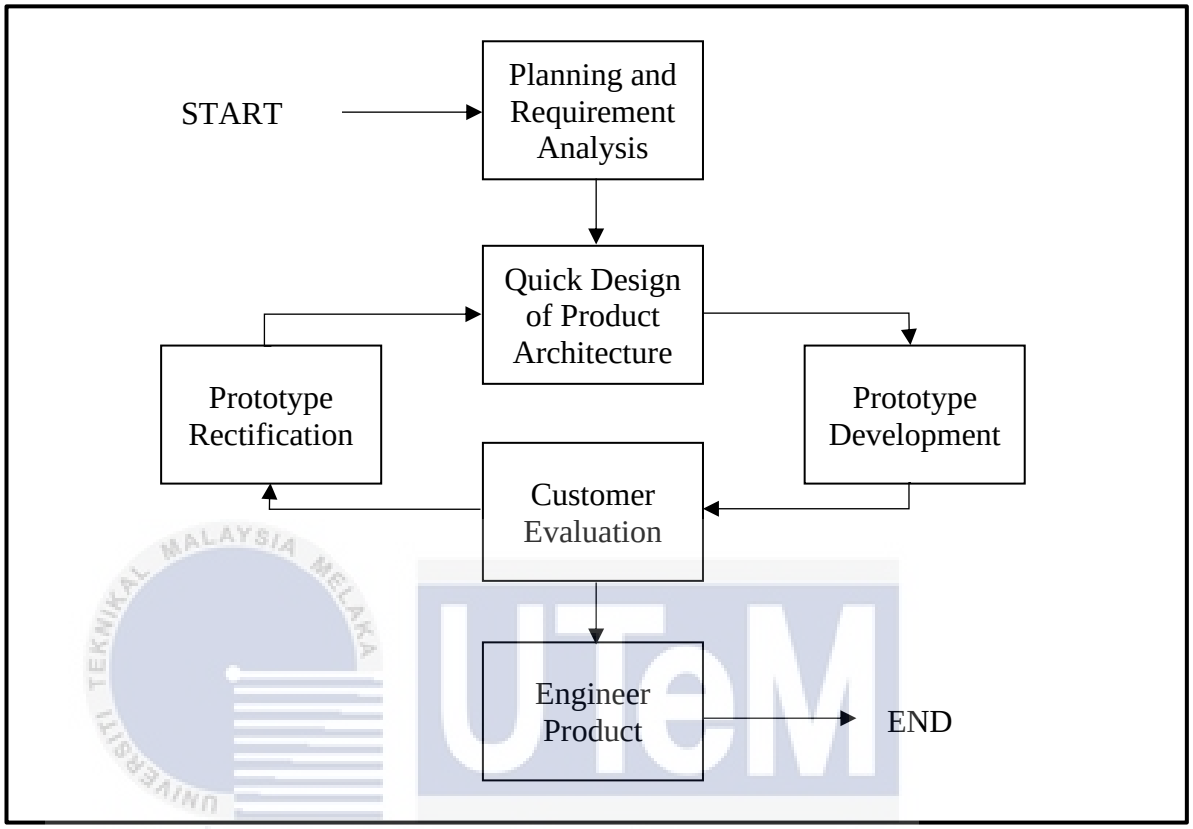


Figure 3.1: Prototyping Model

3.2.1 Planning and Requirement Analysis

In this stage, the hardware and software requirement of this project is identified and listed at below.

a. Hardware Requirement

- 22" LED Monitor S22F350FHE Monitor (Samsung)
- Raspberry Pi 4 Computer Model B 2GB RAM
- Official RPi 15W (5V/3A) PSU USB C UK Plug-White
- Micro SD Card 16GB

- Micro HDMI to HDMI cable
- 12VDC Solenoid Door Lock
- 3x18650 Battery holder
- 3x3.7V 2000mAh Li-ion Battery
- 1 CH Active H/L 5V OptoCoupler Relay Module
- CH-2 Quick Wiring Terminal Press Type
- Maker pHAT
- Logitech Webcam C270
- Male to Female Jumper Wires

b. Software requirement

- Raspberry Pi OS (Imager)
- Python IDE
- OpenCV (Haar Cascade Algorithm and LBP Algorithm)
- Telegram

3.2.2 Quick Design of System Architecture

In this stage, the simple design of the system is produced. However, it is not the final design of the system. It is an important stage as it clearly defines how the system works and provide a brief idea of the system with a diagram. The physical network diagram of the system is designed based on Figure 2.3.

3.2.3 Prototype Development

In this stage, a prototype of the system is produced based on the quick design. It precedes a complete idea of how the final system looks like so that we can make improvement on the prototype and produced the most practical system.

3.2.4 Customer Evaluation

In this stage, the prototype produced is presented to the evaluator. It enables us to identify the strengths and the weaknesses of the proposed prototype. The evaluator provided their comments and suggestions on the prototype and the developer can make improvement based on these comments and suggestions in the next stage. The developer can use the comment and suggestion of the evaluator to conduct a functionality test and performance test later. The functionality test is to ensure the system works well. The performance test is to evaluate the attributes such as stability and reliability of the system that works under the load. The SUS will be used to assess the usability of the system.

3.2.5 Prototype Rectification

In this stage, if the evaluator is not satisfied with the presented prototype, the developer is required to draft a new design and produce the next prototype starting to refine the previous prototype based on the comments and suggestions provided. This stage is repeated until all the requirements of the evaluator are met. Once the evaluator is satisfied with the proposed prototype, the developer can proceed to the next stage which is Engineer Product.

3.2.6 Engineer Product

In this stage, the complete system is implemented according to the final prototype. To make sure the system operates well and prevent failures, the system needs to be tested and carry out the maintenance regularly.

3.3 Project Schedule and Milestones

To track the progress of the project and make sure the project completed on time, the project schedule and milestone which detailed the project timeline is constructed. It marked every progress with the start date and end date. A Gantt Chart for FYP 1 and FYP 2 are constructed.

Project Milestone:

Table 3.1: Milestone of Face Recognition Smart Door Lock Security System Using Haar Cascade Algorithm and LBP Algorithm

Activity	Responsibility	Start Date	End Date
Planning the requirements	Student	Week 0 (15/3/2021 - 21/3/2021)	Week 0 (15/3/2021 - 21/3/2021)
Analyse the requirement	Student	Week 0 (15/3/2021 - 21/3/2021)	Week 0 (15/3/2021 - 21/3/2021)
Design the system	Student	Week 1 (22/3/2021 - 28/3/2021)	Week 1 (22/3/2021 - 28/3/2021)
Present the designed system to supervisor	Student	Week 1 (22/3/2021 - 28/3/2021)	Week 1 (22/3/2021 - 28/3/2021)

Prepare the hardware required	Student	Week 2 (29/3/2021 - 4/4/2021)	Week 2 (29/3/2021 - 4/4/2021)
Install the Raspberry Pi OS	Student	Week 2 (29/3/2021 - 4/4/2021)	Week 2 (29/3/2021 - 4/4/2021)
Implement the prototype	Student	Week 2 (29/3/2021 - 4/4/2021)	Week 6 (26/4/2021 – 2/5/2021)
Testing the prototype	Student	Week 6 (26/4/2021 – 2/5/2021)	Week 6 (26/4/2021 – 2/5/2021)
Progress evaluation 1	Student and supervisor	Week 7 (3/5/2021 – 9/5/2021)	Week 7 (3/5/2021 – 9/5/2021)
Supervisor evaluation on the prototype	Student and supervisor	Week 7 (3/5/2021 – 9/5/2021)	Week 7 (3/5/2021 – 9/5/2021)
Draft the new design based on evaluation	Student	Week 8 (10/5/2021 – 16/5/2021)	Week 8 (10/5/2021 – 16/5/2021)

Redesign the system in detail	Student	Week 9 (17/5/2021 – 23/5/2021)	Week 9 (17/5/2021 – 23/5/2021)
Refine the prototype	Student	Week 10 (24/5/2021 – 30/5/2021)	Week 10 (24/5/2021 – 30/5/2021)
Progress evaluation 2	Student and supervisor	Week 11 (31/5/2021 – 6/6/2021)	Week 11 (31/5/2021 – 6/6/2021)
Supervisor evaluation on the rectified prototype	Student and supervisor	Week 11 (31/5/2021 – 6/6/2021)	Week 11 (31/5/2021 – 6/6/2021)
Implementing the full project	Student	Week 12 (7/6/2021 – 13/6/2021)	Week 13 (14/6/2021 – 20/6/2021)
Testing and maintenance of the full project	Student	Week 13 (14/6/2021 – 20/6/2021)	Week 13 (14/6/2021 – 20/6/2021)
Supervisor and evaluator evaluation	Student, supervisor, evaluator	Week 14 (21/6/2021 – 27/6/2021)	Week 14 (21/6/2021 – 27/6/2021)

Project Gantt Chart:

Table 3.2: Gantt Chart of Face Recognition Smart Door Lock Security System Using Haar Cascade Algorithm and LBP Algorithm

Week / Stage	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Planning and Requirement Analysis															
Quick Design of System Architecture															
Prototype Development															
Customer Evaluation															
Prototype Rectification															
Engineer Product															

3.4 Conclusions

In conclusion, this chapter overall described about the methodology used in the project. It is an importance guideline for the project throughout the process of implementation of the system. It is used to ensure that the system is developed in a proper manner and flow so that a practical system can be delivered. In next chapter, the analysis and design of this project will be discussed briefly.

CHAPTER 4: ANALYSIS AND DESIGN

4.1 Introduction

Project design is a phase that a system is designed to meet the system requirements that have identified in the previous chapter. It is an early phase of the project where the key features, structure, criteria for success and major deliverables are all planned out.

In this chapter, the analysis and design of this project are discussed briefly. The problem analysis, requirement analysis, high-level design is explained in detail. Firstly, the problem analysis discussed about the problem exist on current smart door lock system. Next, the requirement of this project such as data requirement, functional requirement, hardware requirement and software requirement are clearly explained in this chapter. Then, the high-level design such as physical design and user interface design are constructed. In addition, the flowchart of this project is drawn to show how the system works. Lastly, the overall content of this chapter will be concluded and summarized in conclusion.

4.2 Problem Analysis

Although we are used to the simple door lock only with door locker or padlock, or smart door lock using other approach such as fingerprint, RFID and etc, but these approaches have many disadvantages. The problems of these door lock system are the place is insecure since no matter owner or stranger can access the simple door lock only with door locker or padlock, or smart door lock using other approach such as fingerprint, RFID, and password easily, the door lock with physical key makes the

user's life inconvenient, and most of door lock system will not notify user when stranger or intruder detected.

4.3 Requirement Analysis

4.3.1 Data Requirement

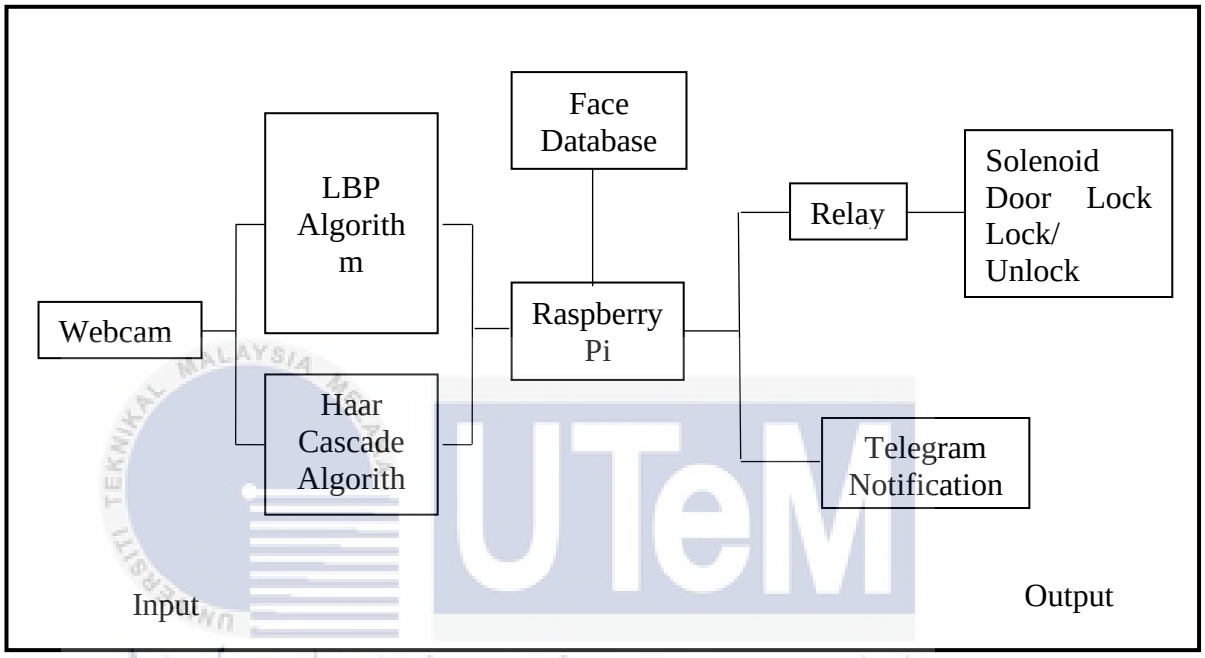


Figure 4.1: Block Diagram of Face Recognition Smart Door Lock Security System Using Haar Cascade Algorithm and LBP Algorithm

The inputs of the proposed system are the face captured or detected. The webcam is used to capture the face of the user. The image of the face captured will then train by Haar Cascade Algorithm and save into the face database. The webcam also detects the person who appear in front of the door lock system. The face detected will be compared and matched with the dataset in face database using and LBP Algorithm.

The outputs of the proposed system are the lock or unlock of solenoid door lock and Telegram notification. If the face detected is matched with the face dataset in face database of the system, the system will send command to the relay to unlock the solenoid door lock. Otherwise, the solenoid door lock will keep locked and a notification will be sent through Telegram to notify premise owner that there is stranger appear in front of their door.

4.3.2 Functional Requirement

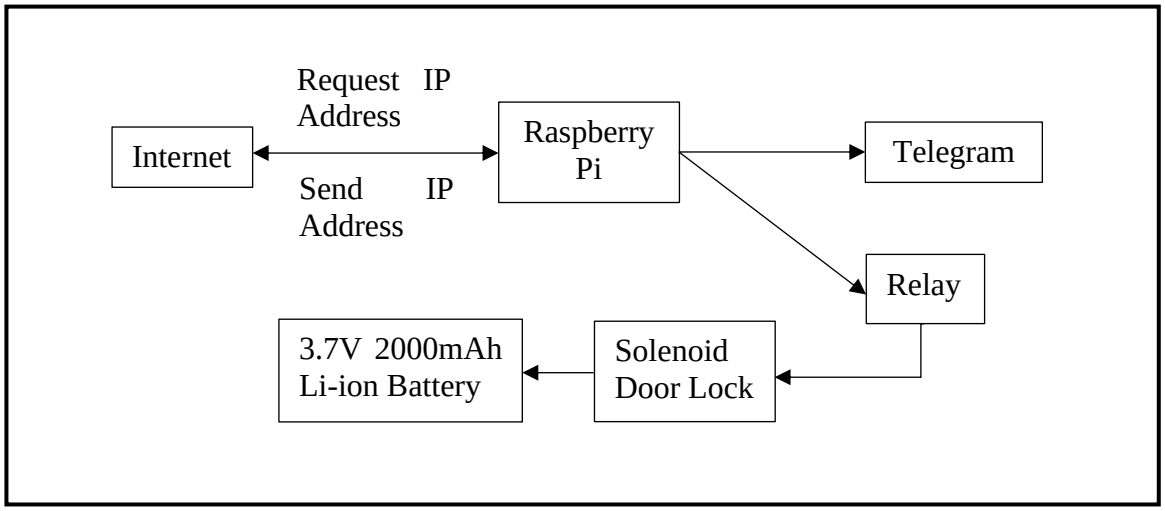


Figure 4.2: Context Diagram of Face Recognition Smart Door Lock Security System Using Haar Cascade Algorithm and LBP Algorithm

Functional requirements denote the functions or features that must implement into the system so that the system can accomplish the task. In this system, Raspberry Pi and smartphone must be connected to Internet so that the function of the face recognition smart door lock security system developed can works. If stranger is detected, a notification will be sent to the user using Telegram. This action is done by Raspberry Pi by executing the Python script. In addition, three 3.7V 2000mAh Li-ion Battery must be connected to the solenoid door lock because it is a 12V solenoid, it needs minimum 0.6A of current to drive it. When Raspberry Pi send command to the relay, the solenoid door lock will be activated minimum 0.6A of current are applied.

4.3.3 Hardware and Software Requirement

To complete this project, there are some software and hardware requirements.

4.3.3.1 Hardware Requirement

Some hardware is required to capture the image, detect the face, lock or unlock the door, and send notification through Telegram.

i. 22" LED Monitor S22F350FHE Monitor (Samsung)

Figure 4.3: 22" LED Monitor S22F350FHE Monitor (Samsung)

A TV or a computer monitor is required to view the environment or GUI of Raspberry Pi's desktop. This can make it easier for us to develop the system.

ii. Raspberry Pi 4 Computer Model B 2GB RAM

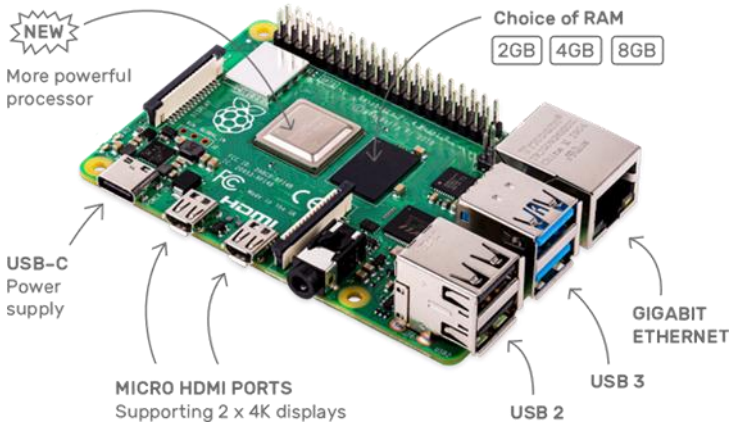


Figure 4.4: Raspberry Pi 4 Computer Model B 2GB RAM

A United Kingdom charity, Raspberry Pi Foundation has made a series of single-board computers for the purpose of making computing education easier. It is a tiny and affordable computer that enable us to learn programming, build projects using different hardware, implement home automation, implement Kubernetes clusters, edge computing, and even apply them in industrial applications. It provides a set of GPIO pins, allowing us to control electronic components for physical computing and explore the IoT.

iii. Official RPi 15W (5V/3A) PSU USB C UK Plug-White

Figure 4.5: Official RPi 15W (5V/3A) PSU USB C UK Plug-White

This is an official power supply with USB type C to power up the Raspberry Pi 4 Computer Model B board.

iv. Micro SD Card 16GB

Figure 4.6: Micro SD Card 16GB

To store the Raspberry Pi OS, configuration and files on the Raspberry Pi, a microSD card with 16GB capacity is required.

v. Micro HDMI to HDMI cable

Figure 4.7: Micro HDMI to HDMI cable

Raspberry Pi 4 consists of two micro HDMI ports. It allows us to connect the board to two different screens.

vi. 12VDC Solenoid Door Lock

Figure 4.8: 12VDC Solenoid Door Lock

Solenoids are using electromagnets, which consists of copper wire coil with an armature in the middle. This solenoid door lock has a slug with a slanted cut and a good mounting bracket. When 12VDC or 0.6A is applied, the slug is pulled into the center of the coil and unlock the door. Some commands are applied to the Raspberry Pi to control the solenoid door lock so that it can lock or unlock based on the programmed condition.

vii. 3x18650 Battery Holder

Figure 4.9: 3x18650 Battery Holder

This battery holder has bare wire leads and its end. It enables us to insert three 18650 rechargeable batteries together.

viii. 3.7V 2000mAh Li-ion Battery

Figure 4.10: 3.7V 2000mAh Li-ion Battery

The 18650 Li-ion rechargeable battery with rated voltage at 3.7V have capacity of 2000mAh. It is Lithium-Ion based and it can be recharge but a proper charger is required. It supplies electricity to the solenoid door lock. Three 3.7V 2000mAh Li-ion Battery are required as the solenoid door lock need 9-12VDC or 0.6A to works.

ix. 1 CH Active H/L 5V Optocoupler Relay Module

Figure 4.11: 1 CH Active H/L 5V Optocoupler Relay Module

This relay module is a single channel 5V activation relay, which can be used to control 250VAC at 10A. Extra isolation with opto-coupler at the front end of control, and the activation logic for relay is configurable through a mini jumper. It is used to isolate solenoid door lock (low voltage) and Raspberry Pi Model B (high voltage).

x. CH-2 Quick Wiring Terminal Press Type

Figure 4.12: CH-2 Quick Wiring Terminal Press Type

CH-2 Quick Wiring Terminal Press Type is a connector that provides a quick connecting work without using screwdriver. It is easy to use as we just need to insert the wire into the mounting hole on both ends.

xi. Maker pHAT

Figure 4.13: Maker pHAT

Maker pHAT is a well-designed board for all the beginners to use Raspberry Pi more simple and easier to accelerate the process of development. It helps us to prevent the complexity on the connection of many different devices or accessories.

xii. Logitech Webcam C270

Figure 4.14: Logitech Webcam C270

A webcam is needed in face recognition smart door lock security system to capture and detect the face. Logitech Webcam C270 is a webcam with max resolution of 720p/30fps that can be connected to Raspberry Pi using USB port and work as the camera to capture and detect the face. It is an important device in this project.

xiii. Male to Female Jumper Wires

Figure 4.15: Male to Female Jumper Wires

Male to Female Jumper wires are small wire ducts with pin protruding on one end and without pin protruding at another end. It can be used to connect device to each other on breadboards or other board. It enables quick and easy prototyping connections.

Table 4.1: Summarization of hardware and usage

Hardware	Usage
22" LED Monitor S22F350FHE Monitor (Samsung)	To view the environment or GUI of Raspberry Pi's desktop.
Raspberry Pi 4 Computer Model B 2GB RAM	To control electronic components for physical computing and explore the IoT.
Official RPi 15W (5V/3A) PSU USB C UK Plug-White	To power up the Raspberry Pi 4 Computer Model B board.
Micro SD Card 16GB	To store the Raspberry Pi OS configuration and files on the Raspberry Pi 4 Computer Model B.
Micro HDMI to HDMI cable	To connect the board to two different screens.
12VDC Solenoid Door Lock	To act as door lock and lock or unlock based on the programmed condition.

3x18650 Battery Holder	To insert three 18650 rechargeable batteries together.
3.7V 2000mAh Li-ion Battery	To supplies electricity to the solenoid door lock.
1 CH Active H/L 5V Optocoupler Relay Module	To isolate solenoid door lock (low voltage) and Raspberry Pi Model B (high voltage).
CH-2 Quick Wiring Terminal Press Type	To provides a quick connecting work without using screwdriver.
Maker pHAT	To prevent the complexity on the connection of many different devices or accessories.
Logitech Webcam C270	To capture and detect the face.
Male to Female Jumper Wires	To connect device to each other on breadboards or other board.

4.3.3.2 Software Requirement

i. Raspberry Pi OS (Imager)

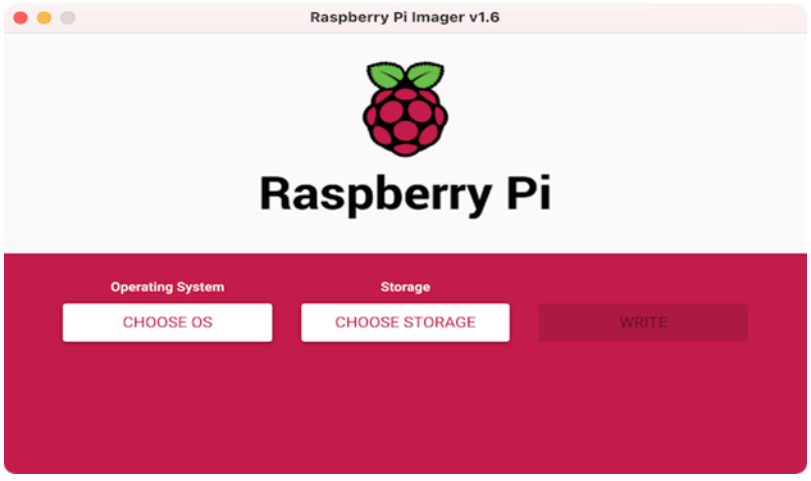


Figure 4.16: Raspberry Pi OS (Imager)

Raspberry Pi require Raspberry Pi OS to operate (previously called Raspbian). It is the official supported operating system for Raspberry Pi. Raspberry Pi Imager is one of the easy methods to install Raspberry Pi OS. The Raspberry Pi OS runs Linux. It consists of at least 35000 packages, pre-compiled software bundled that make us easier to install.

ii. Python IDE

Figure 4.17: Python IDE

Python IDE is a free and open source programming software. Python IDE is a built-in software in Raspberry Pi OS. In this project, a python IDE named Thonny Python is used to enable us to write, edit and run the Python command in order to develop a face recognition smart door lock security system using Haar Cascade Algorithm and LBP Algorithm.

iii. OpenCV (Haar Cascade Algorithm and LBP Algorithm)

Figure 4.18: OpenCV

OpenCV is a free and open source software library for programming functions especially for machine learning, image processing, and computer vision. In this project, we used Haar Cascade Algorithm and LBP Algorithm provided by OpenCV. The Haar Cascade Algorithm can detect different objects in an image or video since it can be run on real time but was mainly focus on face detection on image or video. To perform face recognition, we need to train the classifier to recognize whether the region contain face (positive image) or not contain face (negative image). The LBP Algorithm describe the neighborhood of image elements using binary codes. It detects microstructures such as edges, lines, spots, flat areas, which can be estimated by the histogram. Each pixel of the image or video is compared with its neighboring pixels and composed as the relative values.

iv. Telegram

Figure 4.19: Telegram

Telegram is free open online instant messaging application which can use at computer and smartphone. It is supported by different platform such as Windows, Mac OS, iOS, and Android. It enables features such as end-to-end encrypted voice and video calling, file sharing etc. In addition, it also consists of third-party application such as Telegram Bot. Developer can create their own Telegram Bot using BotFather that benefit their invention. They can interact with their bot by giving them command or inline request. The bot will react and give respond to the developer based on the command given.

4.4 High-Level Design

High-level design explains the architecture that used to implement face recognition smart door lock security system. The connection between the smartphone and Raspberry Pi is defined. For the devices connected to Raspberry Pi such as camera to communicate with the smartphone, wireless communication is used. Notification about the stranger detected will be sent to user and through Telegram (Telegram Bot). Proper design must be determined so that the system can work properly.

4.4.1 System Architecture, Physical Design

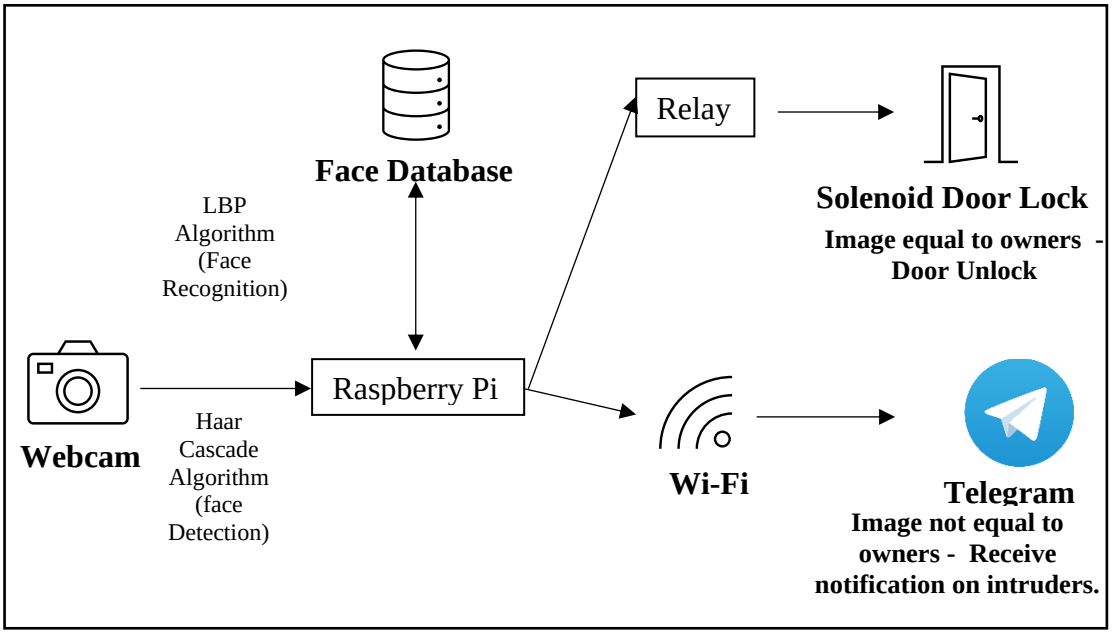


Figure 4.20: Physical Design of Face Recognition Smart Door Lock Security System Using Haar Cascade Algorithm and LBP Algorithm

4.4.2 User Interface Design

i. Flowchart

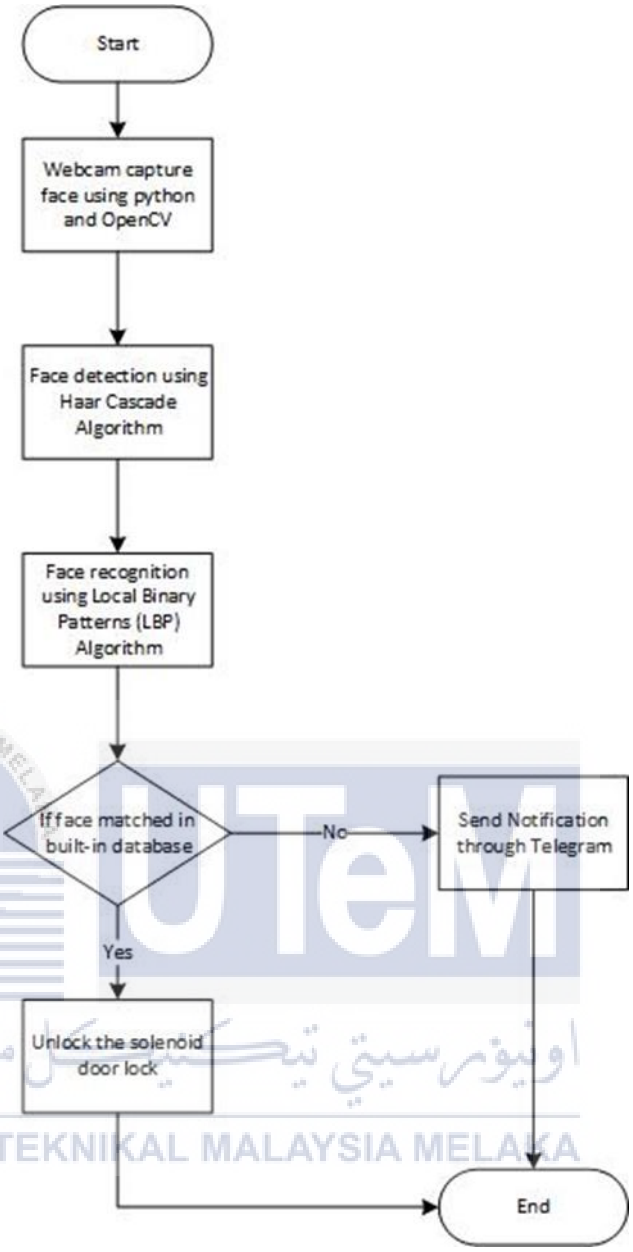


Figure 4.21: Flow Chart of Face Recognition Smart Door Lock Security System Using Haar Cascade Algorithm and LBP Algorithm

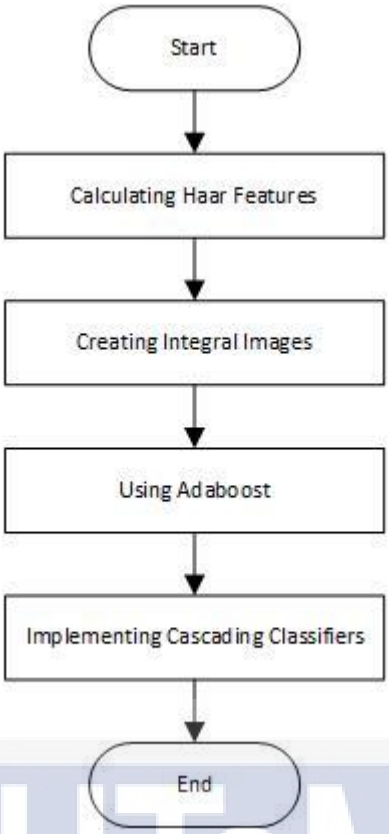


Figure 4.22: Flow Chart of Stages of Haar Cascade Algorithm

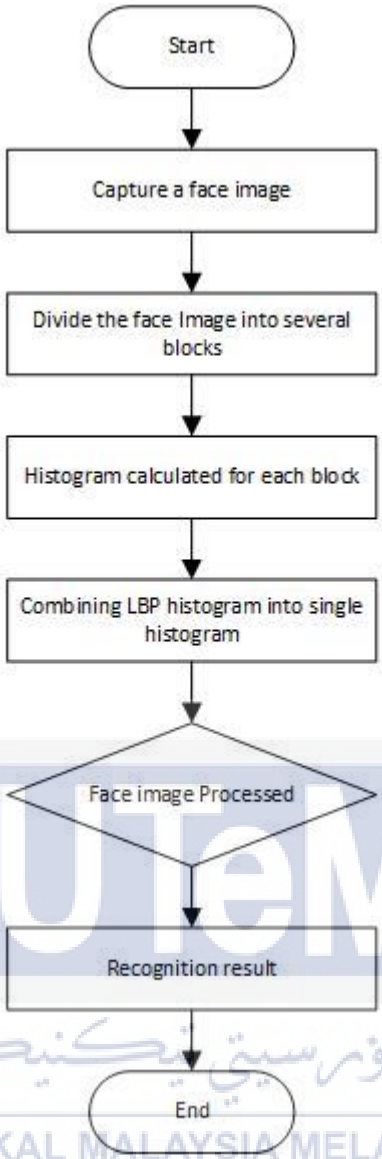


Figure 4.23: Flow Chart of LBP Algorithm

ii. Input Design

Input is the information or data sent into the system to produce output. In this project, webcam is the input which will detect and capture the face image. The image captured will be processed by the Raspberry Pi using Haar Cascade Algorithm and LBP Algorithm to detect and recognize the face.

iii. Output Design

The input sent into the system are processed by the Raspberry Pi and will produce the output. If the face image matched with dataset in face database, the system will send command to the relay to solenoid door lock will be unlocked. Otherwise, the solenoid door lock will keep locked and a notification will be sent through Telegram as the output.

iv. Algorithm

The algorithm used in face detection and face recognition is Haar Cascade Algorithm and LBP Algorithm.

a) Haar Cascade Algorithm

There are four stages in this algorithm:

i. Calculating Haar Features

First, the system will collect the Haar features from the face image. A Haar feature is essentially calculations that are performed on adjacent rectangular areas at a specific region in a detection window. The darker region in the haar feature are pixels with values 1, and the lighter region are pixels with values 0. The calculation involves the sum of the pixel intensities in darker region and the sum of the pixel intensities in lighter region, then calculating the differences between the sums. If the difference is close to 1, means that there is an edge detected by the haar feature.

ii. Creating Integral Images

If the image is too large, it will be difficult to determine these features. This can be solved by creating integral images. It accelerates the calculation of these Haar features by creates sub-rectangles and creates array references for

each of those sub-rectangles instead of computing at every pixel.

iii. Using Adaboost

To identify the best features that represent an object from the many Haar features Adaboost is used. Adaboost select the best features and trains the classifiers to use them. The ‘weak classifiers’ is combined to form a ‘strong classifier’ which enable the algorithm to detect objects. By scanning over the input image using a window and calculating the Haar features for each subsection of the image weak leaners are created. This difference is compared to a learned threshold that separates non-objects from objects. Because these are “weak classifiers,” many Haar features is needed for accuracy to form a strong classifier.

iv. Implementing Cascading Classifiers

Then, cascading classifiers are implemented to combines these weak learners into a strong learner. The cascade classifier is made up of a series of stages, where each stage is a collection of weak learners. Weak learners are trained using boosting, which allows for a highly accurate classifier from the mean prediction of all weak learners. Based on this prediction, the classifier either decides to indicate an object was found (positive) or move on to the next region (negative). Stages are designed to reject negative samples as fast as possible, because most of the windows do not contain anything of interest.

b) LBP Algorithm

The LBP Algorithm uses 4 parameters that are Radius, Neighbors, Grid X and Grid Y. Radius is used to build the circular local binary pattern and represents the radius around the central pixel. It is usually set to 1. Neighbors is the number of sample points to build the circular local binary pattern. It is usually set to 8. Grid X is the number of cells in the horizontal direction. The more cells, the finer the grid, the higher the dimensionality of the resulting feature vector. It is usually set to 8. Grid Y is the number of cells in the vertical direction. The more cells, the finer the grid, the higher the dimensionality of the resulting feature vector. It is usually set to 8.

i. Apply the LBP Operation

After capture the face image, the face image will be converted into grayscale. The image has many windows of 3x3 pixels. It can also be represented as a 3x3 matrix consists of the intensity of each pixel (0~255). The central value of these 3x3 matrices will be used as the threshold. This value will be used to define the new values from the 8 neighbors. For each neighbor of the central value (threshold), a new binary value will be set. 1 is set for values equal or higher than the threshold and 0 for values lower than the threshold. Then, the matrix will ignore the central value and left only binary values. We need to focus on each binary value from each position from the matrix line by line into a new binary value. After that, this binary value will be converted to a decimal value and set it to the central value of the matrix, which is a pixel from the original image. At the end of LBP procedure, we have a new image which represents better the characteristics of the original image.

ii. Extracting Histograms

Then, the image generated will be divided the image into multiple grids using the Grid X and Grid Y parameter and extract the histogram of each region. As we have an image in grayscale, each histogram from each grid will contain only 256 positions (0~255) representing the occurrences of each pixel intensity. To create a new and bigger histogram, each histogram will be concatenated. Supposing we have 8x8 grids, we will have 8x8x256=16,384 positions in the final histogram. The final histogram represents the characteristics of the image original image.

iii. Performing the face recognition

The algorithm is trained at the previous step. Each histogram created is used to represent each image from the training dataset. So, given an input image, the steps will be performed again for this new image and creates a histogram which represents the image. So, to find the image that matches the input image, two histograms will be compered and return the image with the closest histogram. Euclidean distance is used to compare the histograms (calculate the distance between two histograms).

$$D = \sqrt{\sum_{i=1}^{n} (hist1 - hist2)^2}$$

4.5 Conclusion

In this chapter, the analysis and design of the Face Recognition Smart Door Lock Security System Using Haar Cascade Algorithm and LBP Algorithm is the most significant section to complete this project and develop a Face Recognition Smart Door Lock Security System Using Haar Cascade Algorithm and LBP Algorithm . In problem analysis, the problem of the different door lock system is discussed. The

requirement analysis is to identify data requirement, functional requirement, hardware and software requirement of this project. The user interface design such as flowchart , input design and output design of the project are defined and will be used as the guideline to build the project. This analysis and design will be referred when building the project. The next chapter will be discussed about the implementation of the project.

CHAPTER 5: IMPLEMENTATION

5.1 Introduction

This chapter will focus on the implementation of the project. Hardware and software development environment setup will discuss about the hardware and software requirements and installation. System configuration setup will cover the configuration of the system in this project such as face capture configuration, face recognition configuration, solenoid door lock configuration and Telegram configuration. The last part of this chapter is the conclusion which will summarize this chapter.

5.2 Hardware and Software Development Environment Setup

5.2.1 Hardware Setup

a. Hardware Requirement

- 22" LED Monitor S22F350FHE Monitor (Samsung)
- Raspberry Pi 4 Computer Model B 2GB RAM
- Official RPi 15W (5V/3A) PSU USB C UK Plug-White
- Micro SD Card 16GB

- Micro HDMI to HDMI cable
- 12VDC Solenoid Door Lock
- 3x18650 Battery holder
- 3x3.7V 2000mAh Li-ion Battery
- 1 CH Active H/L 5V OptoCoupler Relay Module
- CH-2 Quick Wiring Terminal Press Type
- Maker pHAT
- Logitech Webcam C270
- Male to Female Jumper Wires

b. Hardware Installation

The hardware such as Raspberry Pi, Maker pHAT, Logitech Webcam C270 and etc are connected as in figure.

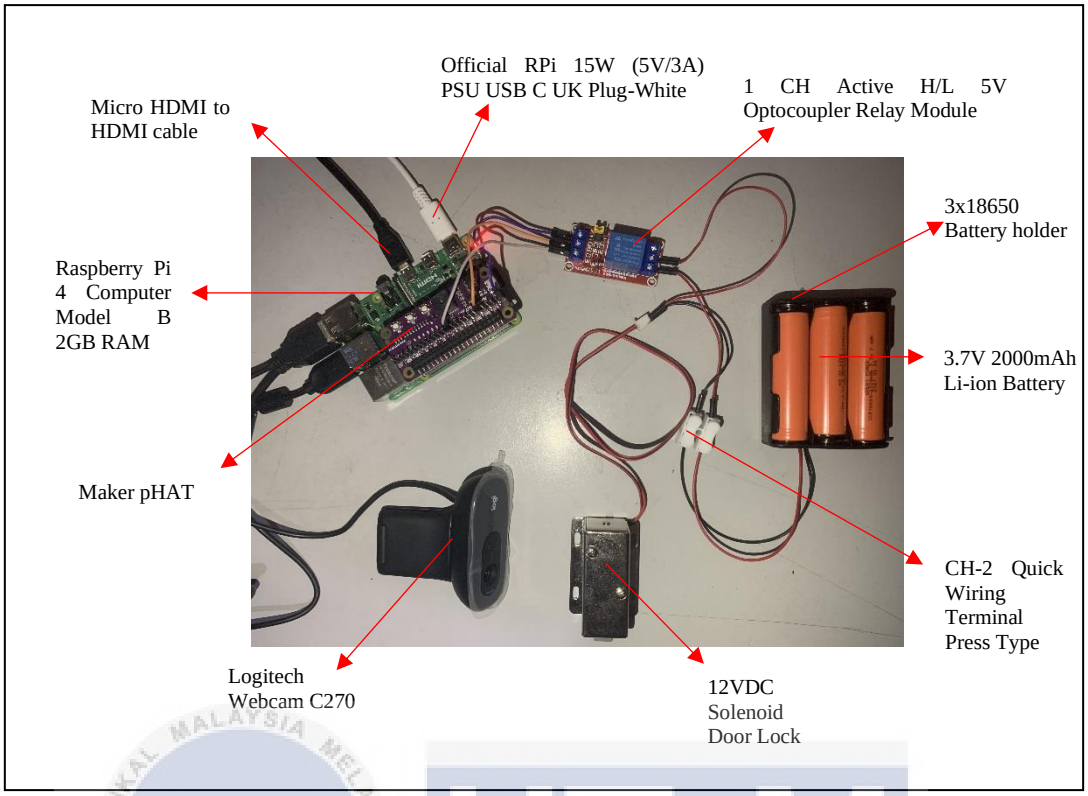


Figure 5.1: Installation of hardware

5.2.2 Software Setup

To develop Face Recognition Smart Door Lock Security System Using Haar Cascade Algorithm and LBP Algorithm, some software such as Raspberry Pi OS, Python IDE, OpenCV, and Telegram need to be installed.

a. Raspberry Pi OS

i. First, download the latest version of Raspberry Pi Imager from the Raspberry Pi website (https://www.raspberrypi.org/software/). After downloaded the Raspberry Pi Imager, insert the micro SD card into the micro SD card reader and run the Imager

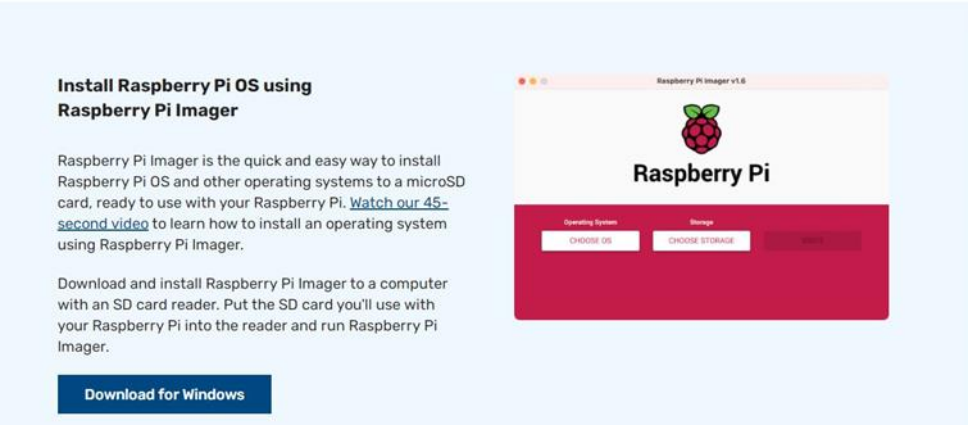


Figure 5.2: Download Raspberry Pi OS

.

ii. Choose the required OS and the micro SD card to write the image. The selections are confirmed and click 'WRITE' to begin writing data to the SD card. Then, remove the micro SD card from the card reader and insert it into the Raspberry Pi 4 Model B board.

Figure 5.3: Installing Raspberry Pi Imager

iii. The operating system when runs automatically when the power supply is turned on. A ‘Welcome to Raspberry Pi’ windows will be shown. Click ‘Next’ to get started.

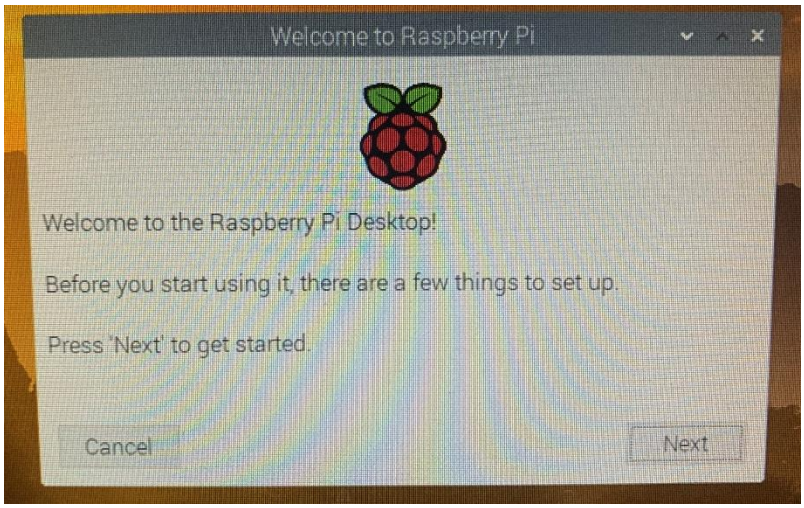


Figure 5.4: Welcome Message of Raspberry Pi OS

iv. Select the country, language, and time zone. Then, click 'Next' to proceed.

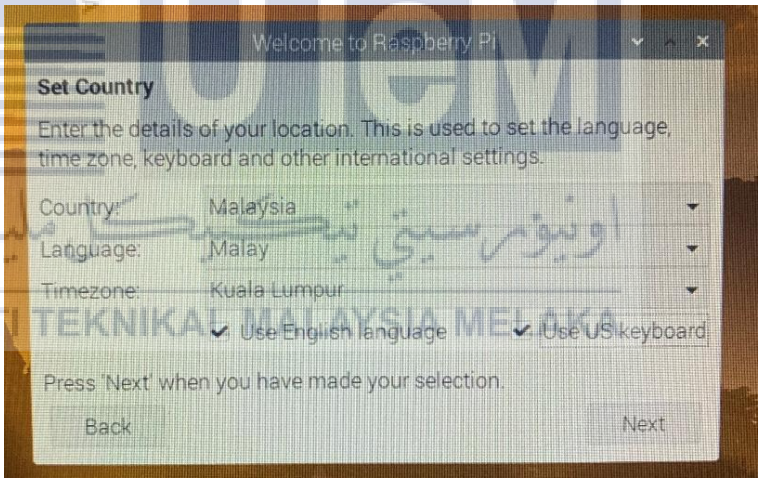


Figure 5.5: Country, Language and Time zone Setting

v. Click 'Next' until the Wi-Fi selection windows. Select the Wi-Fi to be connected and click 'Next'.

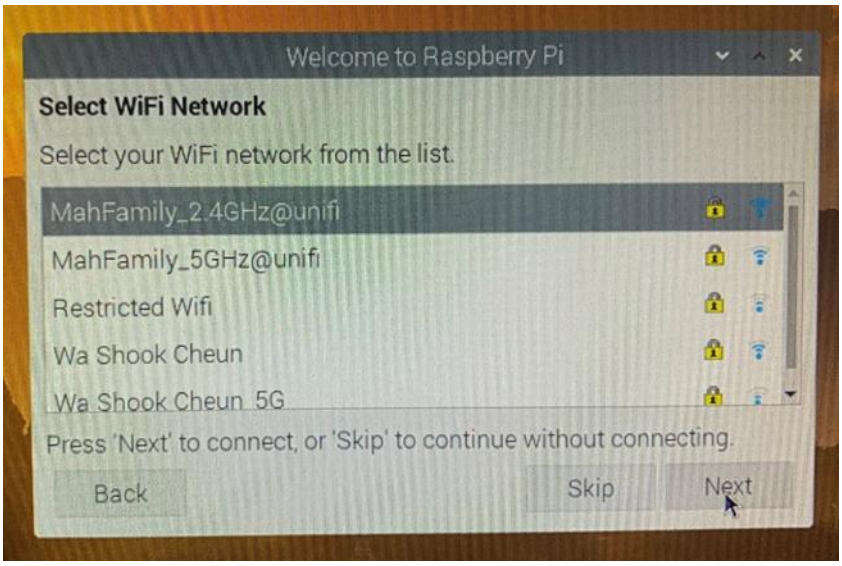


Figure 5.6: Wi-Fi Selection

vi. Insert the password of the Wi-Fi selected. Then, click 'Next'.

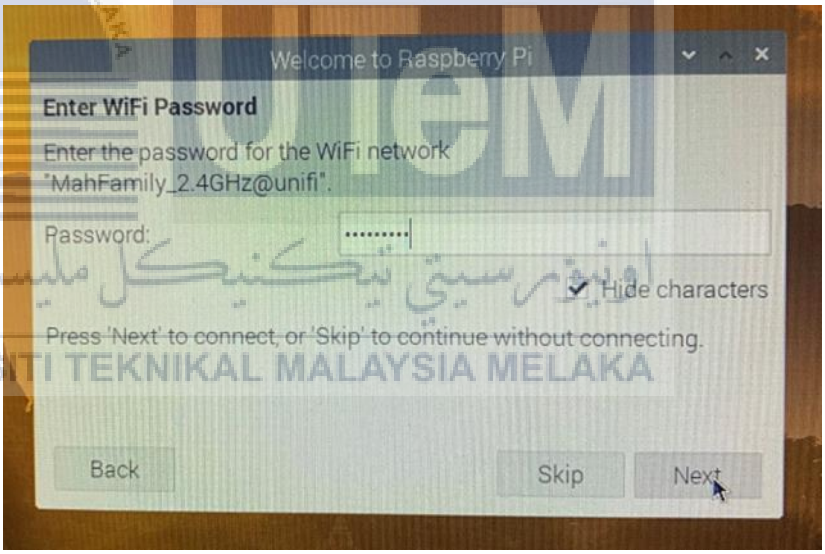


Figure 5.7: Enter Wi-Fi Password

vii. Restart the Raspberry Pi and it is ready for use.

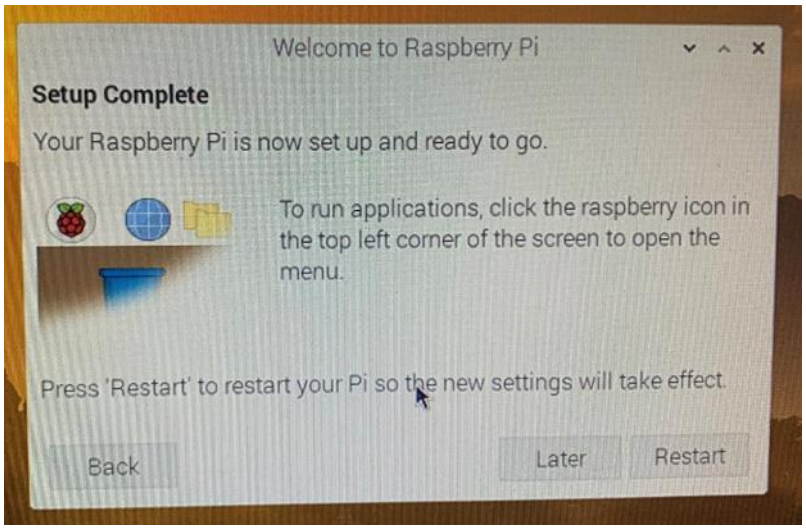


Figure 5.8: Setup Complete Message

viii. To update and upgrade the Raspberry Pi OS to the latest version, some command is used in terminal.

```
sudo apt-get update
sudo apt-get upgrade
```

b. Python IDE

i. To install Thonny as Python IDE, enter the following command in the terminal.

```
sudo apt-get install python3
sudo apt-get install python3-pip
sudo pip3 install Thonny
```

c. OpenCV

i. To install OpenCV in Raspberry Pi, enter the following command.

```
sudo apt install cmake build-essential pkg-config git
sudo apt install libjpeg-dev libtiff-dev libjasper-dev libpng-dev libwebp-dev libopenexr-dev
sudo apt install libavcodec-dev libavformat-dev libswscale-dev libv4l-dev libxvidcore-dev libx264-dev libdc1394-22-dev libgstreamer-plugins-base1.0-dev libgstreamer1.0-dev
sudo apt install libgtk-3-dev libqtgui4 libqtwebkit4 libqt4-test python3-pyqt5
sudo apt install libatlas-base-dev liblapacke-dev gfortran
sudo apt install libhdf5-dev libhdf5-103
sudo apt install python3-dev python3-pip python3-numpy
```

ii. To temporary expand the swapfile, opening dphys-swapfile for editing by using the following command.

```
sudo nano /etc/dphys-swapfile
```

iii. Once the file is open, comment out the line CONF_SWAPSIZE=100 and add CONF_SWAPSIZE=2048. Press Ctrl-X, Y and then Enter to save changes to dphys-swapfile.

iv. For the changes to take effect, restart the swapfile by entering the following command.

```
sudo systemctl restart dphys-swapfile
```

v. Resume package installations by inserting the following commands separately into the terminal.

```
git clone https://github.com/opencv/opencv.git
git clone https://github.com/opencv/opencv_contrib.git
mkdir ~/opencv/build
cd ~/opencv/build
cmake -D CMAKE_BUILD_TYPE=RELEASE \
-D CMAKE_INSTALL_PREFIX=/usr/local \
-D  OPENCV_EXTRA_MODULES_PATH=~/opencv_contrib/modules
\
-D ENABLE_NEON=ON \
-D ENABLE_VFPV3=ON \
-D BUILD_TESTS=OFF \
-D INSTALL_PYTHON_EXAMPLES=OFF \
-D OPENCV_ENABLE_NONFREE=ON \
-D CMAKE_SHARED_LINKER_FLAGS=-latomic \
-D BUILD_EXAMPLES=OFF ..
make -j$(nproc)
sudo make install
sudo ldconfig
```


vi. After OpenCV successfully installed, open swapfile again and change to its original state. Once the file is open, uncomment CONF_SWAPSIZE=100 and delete or comment out CONF_SWAPSIZE=2048. Press Ctrl-X, Y and then Enter to save

```
sudo nano /etc/dphys-swapfile
```

changes to dphys-swapfile.

vii. Restart the swapfile again by entering the following command.

```
sudo systemctl restart dphys-swapfile
```

viii. Install face_recognition.

```
sudo pip install face-recognition
```

ix. Install imutils

```
sudo pip install imutils
```

d. Telegram

i. Telepot need to be install in Raspberry Pi so that Raspberry Pi can be connected to Telegram Bot created.

```
sudo pip install telepot
```

5.3 Software Configuration Setup

5.3.1 Face image capture configuration

i. Create a folder (face database) with the name of the user in the home directory of Raspberry Pi.

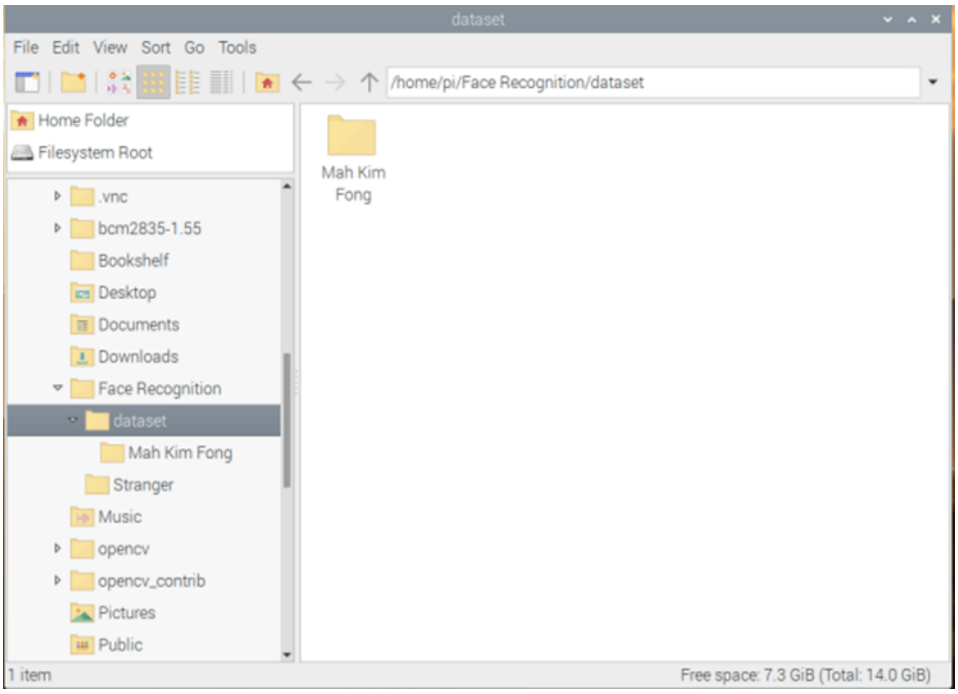


Figure 5.9: Face database created

ii. Activate the webcam and capture the face to be use as dataset.

```
import cv2
name = 'Mah Kim Fong' #replace the name if have multiple user
cam = cv2.VideoCapture(0)
cv2.namedWindow("press space to take a photo", cv2.WINDOW_NORMAL)
cv2.resizeWindow("press space to take a photo", 500, 300)
img_counter = 0
while True:
    ret, frame = cam.read()
    if not ret:
        print("failed to grab frame")
        break
    cv2.imshow("press space to take a photo", frame)
    k = cv2.waitKey(1)
    if k%256 == 27:
        # ESC pressed
        print("Escape hit, closing...")
        break
    elif k%256 == 32:
        # SPACE pressed
        img_name = "dataset/"+ name +"/image_{}.jpg".format(img_counter)
        cv2.imwrite(img_name, frame)
        print("{} written!".format(img_name))
        img_counter += 1
cam.release()
cv2.destroyAllWindows()
```

iii. Train the dataset using Haar Cascade Algorithm so that the system can detect the face.

```
# load the input image and convert it from RGB (OpenCV ordering)
        # to dlib ordering (RGB)
        image = cv2.imread(imagePath)
        rgb = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
        # detect the (x, y)-coordinates of the bounding boxes
        # corresponding to each face in the input image
        boxes = face_recognition.face_locations(rgb, model="hog")
        # compute the facial embedding for the face
        encodings = face_recognition.face_encodings(rgb, boxes)
        # loop over the encodings

        for encoding in encodings:

                # add each encoding + name to our set of known names and

                # encodings

                knownEncodings.append(encoding)

                knownNames.append(name)

# load the known faces and embeddings along with OpenCV's Haar

# cascade for face detection

print("[INFO] loading encodings + face detector...")

data = pickle.loads(open(encodingsP, "rb").read())

detector = cv2.CascadeClassifier(cascade)
```

5.3.2 Face recognition configuration

i. Use LBP Algorithm to enable the system to compare and match the face detected with the face dataset in face database.

```
# detect faces in the grayscale frame
rects = detector.detectMultiScale(gray, scaleFactor=1.1,
        minNeighbors=5, minSize=(30, 30),
        flags=cv2.CASCADE_SCALE_IMAGE)
# OpenCV returns bounding box coordinates in (x, y, w, h) order, reorder it
in (top, right, bottom, left) order
boxes = [(y, x + w, y + h, x) for (x, y, w, h) in rects]
# compute the facial embeddings for each face bounding box
encodings = face_recognition.face_encodings(rgb, boxes)
names = []
# loop over the facial embeddings
for encoding in encodings:
# attempt to match each face in the input image to our known encodings
matches = face_recognition.compare_faces(data["encodings"], encoding)
```

5.3.3 Solenoid door lock configuration

i. If the face detected matched with the face dataset in face database, the solenoid door lock will be unlocked.


```
# check to see if we have found a match
if True in matches:
        # find the indexes of all matched faces then initialize a
        # dictionary to count the total number of times each face was
        # matched
        matchedIdxs = [i for (i, b) in enumerate(matches) if b]
        counts = {}
        # to unlock the door
        GPIO.output(RELAY,GPIO.HIGH)
        prevTime = time.time()
        doorUnlock = True
        print("door unlock")
```

ii. After the solenoid door lock has unlocked for 5 seconds, the system will detect the face again. If the face detected is matched with the face dataset, the door will stay unlock. Else, the solenoid door lock will be locked.

```
#lock the door after 5 seconds
if doorUnlock == True and time.time() – prevTime > 5:
        doorUnlock = False
        GPIO.output(RELAY,GPIO.LOW)
        print("door lock")
```

5.3.4 Telegram configuration

i. If the face detected not matched with the face dataset in face database, the solenoid door lock will be locked, and a notification will be sent through Telegram.

```
telegram_bot = telepot.Bot('1843778705:AAE7K_-Ue-PBwNbjlR0tQxEtlBs7sYn08to')
#message to be displayed
message = "Alert: Stranger detected. Door Locked."
#send message
telegram_bot.sendMessage (1283580476, message)
#send picture
image = cv2.imwrite(filename = 'image.jpg', img=frame)
telegram_bot.sendPhoto (1283580476, photo = open('image.jpg', 'rb'))
```

ii. Before running the python program, a Telegram Bot need to be created. Launch Bot Father at https://telegram.me/botfather. To create a new Telegram bot, send command '/newbot'. Name the Telegram Bot as 'FYP_TELEGRAM_BOT'. Set the username of the Telegram Bot as

'FYP_TELEGRAMBOT'

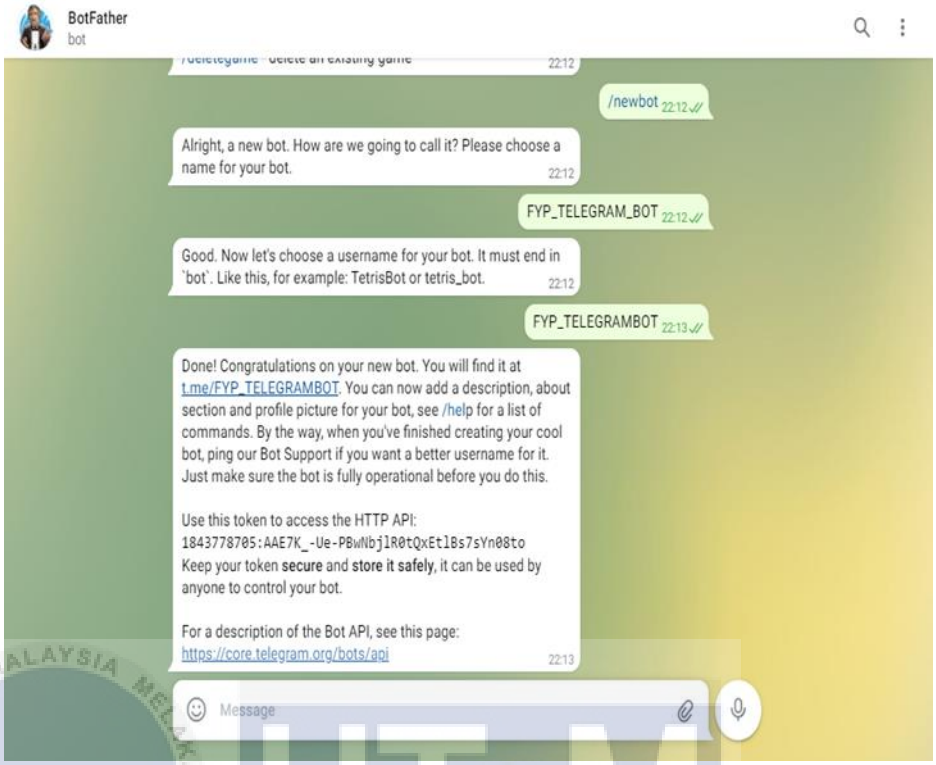


Figure 5.10: Create Telegram Bot

5.4 Conclusion

This chapter discussed about all the implementation of hardware and software in this project such as the setup of hardware and software used in this project and the configuration of those software. The next chapter will focus on the testing and analysis about the implementation of this project.

CHAPTER 6: TESTING

6.1 Introduction

In this chapter, the method used to test the system will be discussed. The purpose of this testing is to make sure that the face recognition smart door lock security system works perfectly. System usability testing such as Functionality testing and Performance testing will be conducted to test the functionalities and performance of the system such as face detection and recognition test, Telegram notification test and solenoid door lock test. In addition, System Usability Scale (SUS), a set of questionnaires will also be conducted to test the usability.

6.2 Test Plan

The basis of the testing is discussed in test plan. There are two type of test plan which is test organization and test environment. Test plan also explain about testing activities include.

6.2.1 Test Organization

This test is involved the developer and 6 respondents. Since the sample size will not affect the reliability of the result, therefore SUS can be applied even the sample sizes are very small with minimum two users (Sauro, J. (2016)). The developer will test the system according to its functionality and performance of the system with the respondents. Developer in charge for the test of the functionality of the system because only developer knows how the system work. The developer and respondents both will be responsible for

testing of the performance of the system. The developer will create multiple datasets in the face database for the respondents and let them to use the system. In addition, they are required to answer a of a questionnaire which consists of 10 items with five response options (Strongly disagree to Strongly agree) with score 1 to 5.

6.2.2 Test Environment

The structure of the testing will be determined. Test requirements are crucial and will be decided in the test environment. Testing of the project will face certain problem if the test environment did not manage appropriately. The test will be carried out using the prototype created.

6.2.3 Test Schedule

The process of testing for the functionality of each function of the system will be done by the developer for three cycles. The testing of the performance of the system will be done for one cycle for each respondent.

6.3 Test Strategy

In this part, the flow of the testing will be determined. Test strategy for this project is how the performance test and functionality test will be conducted to detect and recognize the face, and to receive the Telegram notification if fail to recognize the face. For the functionality test, first, the developer turns on the Raspberry Pi and appear in front of the webcam to enable the webcam to capture the face. If the face detected matched with the dataset, the solenoid door lock will be unlocked. Else, the solenoid door lock will keep locked and the notification will be sent through telegram to the user to inform the user about the intruder. If these function works well, the test of the system is successful. For performance test, the developer will create multiple datasets for the respondents and the respondents will try to use the system to test whether the system meet their requirement. After using the system, they are required to answer the SUS questionnaire about the usability of the system. The SUS score will be calculated to indicate the usability performance of the system.

6.4 Test Design

Test design is made by the developer to perform the testing of the project functionality and performance with the respondents.

6.4.1 Test Description

6.4.1.1 Functionality Test

Specific test such as face detection and recognition test, Telegram notification test and solenoid door lock test will be conducted in detail. Test purpose, test environment, test setup and expected result will be tabulated for each test.

Table 6.1: Face Detection and Recognition Test

Test	Face Detection and Recognition Test
Test Purpose	To ensure that the system able to detect the face capture from the webcam and recognize it if it is matched with the dataset.
Test Environment	To test this project, the webcam must be connected to Raspberry Pi. The user needs to be appeared in front of the webcam.
Test Setup	i. Turn on the Raspberry Pi and connect it to Internet connection. ii. Python code for this project is executed. iii. Once the webcam is activated, the user needs to appear in front of the webcam.

Expected Result	If the face captured matched with the dataset, a green box with the name of the person detected will be shown. If it is not match with the dataset, a green box with the word ‘unknown’ will be shown.

Table 6.2: Telegram Notification Test

Test	Telegram Notification Test
Test Purpose	To ensure that notification about the stranger or intruders is received by the user through Telegram.
Test Environment	To test this project, the smart phone of the user must connect to Internet. User must create a Telegram Bot.
Test Setup	i. Connect the smart phone to the Internet connection. ii. Log into the Telegram account and access the Telegram Bot created.
Expected Result	If the face captured not matched with the dataset, the user will receive the notification through the Telegram application along with the image of the face of the stranger or intruders detected.

Table 6.3: Solenoid Door Lock Test

Test	Solenoid Door Lock Test

Test Purpose	To ensure that solenoid door lock can lock or unlock according to the face recognized.
Test Environment	To test this project, the webcam and the solenoid door lock need to be connected to Raspberry Pi. The face detection and recognition function need to work well.
Test Setup	i. Turn on the Raspberry Pi and connect it to Internet connection. ii. Python code for this project is executed. iii. Once the webcam is activated, the user needs to appear in front of the webcam.
Expected Result	If the face captured matched with the dataset, the solenoid door lock will be unlocked. If it is not match with the dataset, the solenoid door lock will be kept locked.

6.4.1.2 Performance Test

The performance test will be conducted to test the performance of the whole system. The developer will create multiple datasets in face database and use the algorithm to train the datasets. Then, the respondents will try to use the system and answer the SUS questionnaire to assess the usability of the system. To calculate the usability score using SUS, the respondents need to rank all the question. Based on the response that they have ranked, sum up the total score for all odd-numbered questions, then subtract 5 from the total score to get value X. At the same time, sum up the total score for all even-numbered questions, then subtract that total score from 25 to get value Y. Lastly, get the new total score by sum up X and Y and multiply the new total score by 2.5 and we will get the total score out of 100. The average score of SUS is 68. The

scoring above or below 68 will enable us to conclude the overall usability of the system. The results are shown below. The sample of SUS questionnaire and general guideline on the interpretation of SUS score are shown as below:

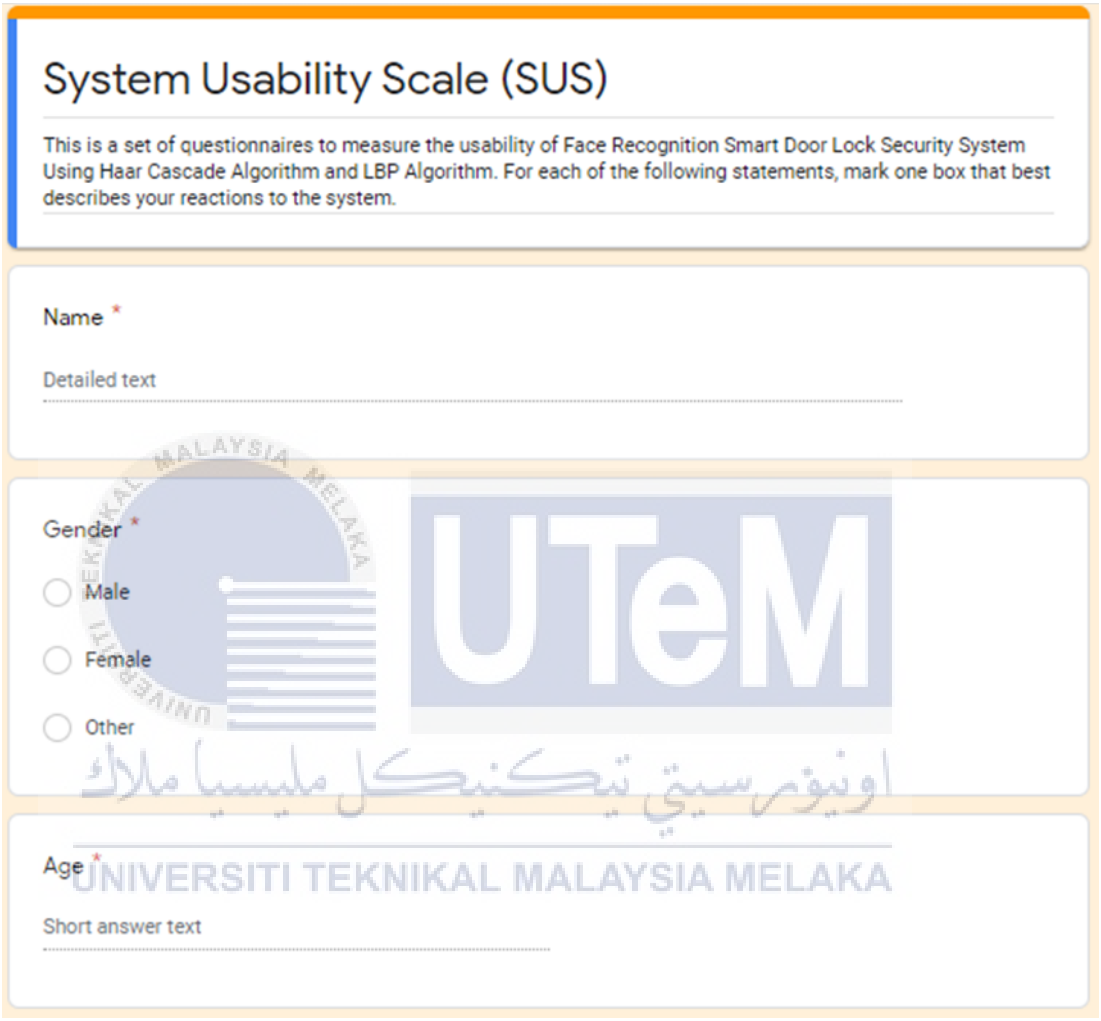


Figure 6.1: SUS questionnaire (General Information)

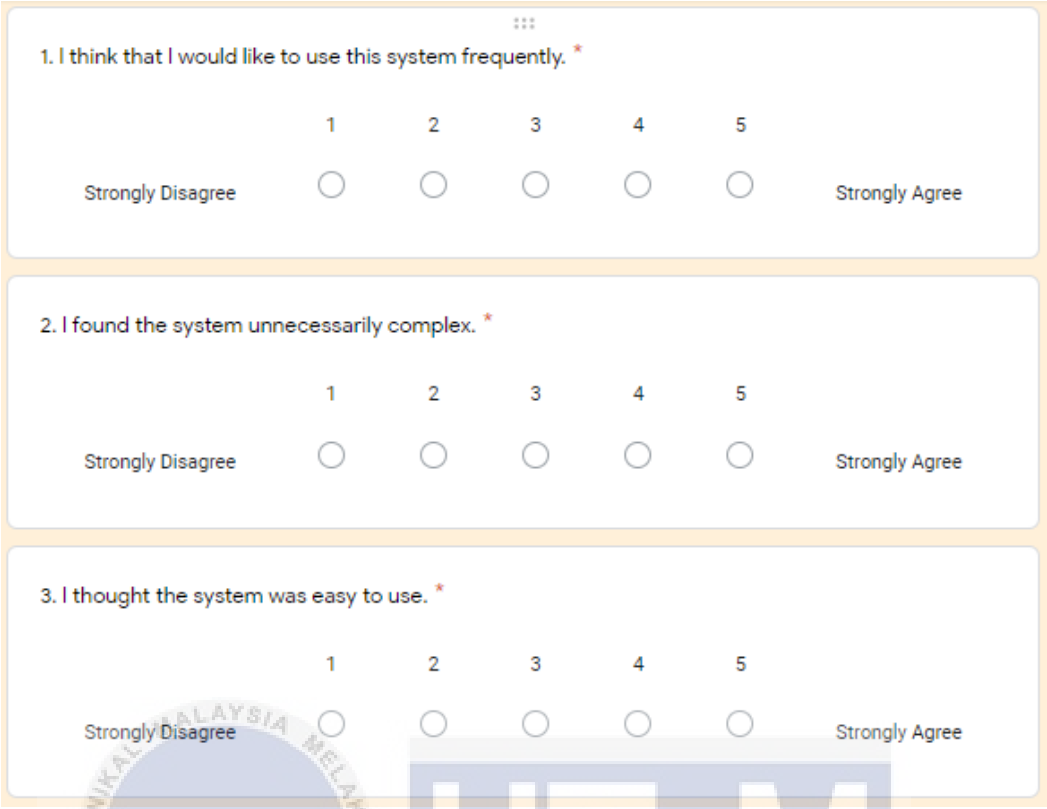


Figure 6.2: SUS questionnaire (Item 1 to 3)

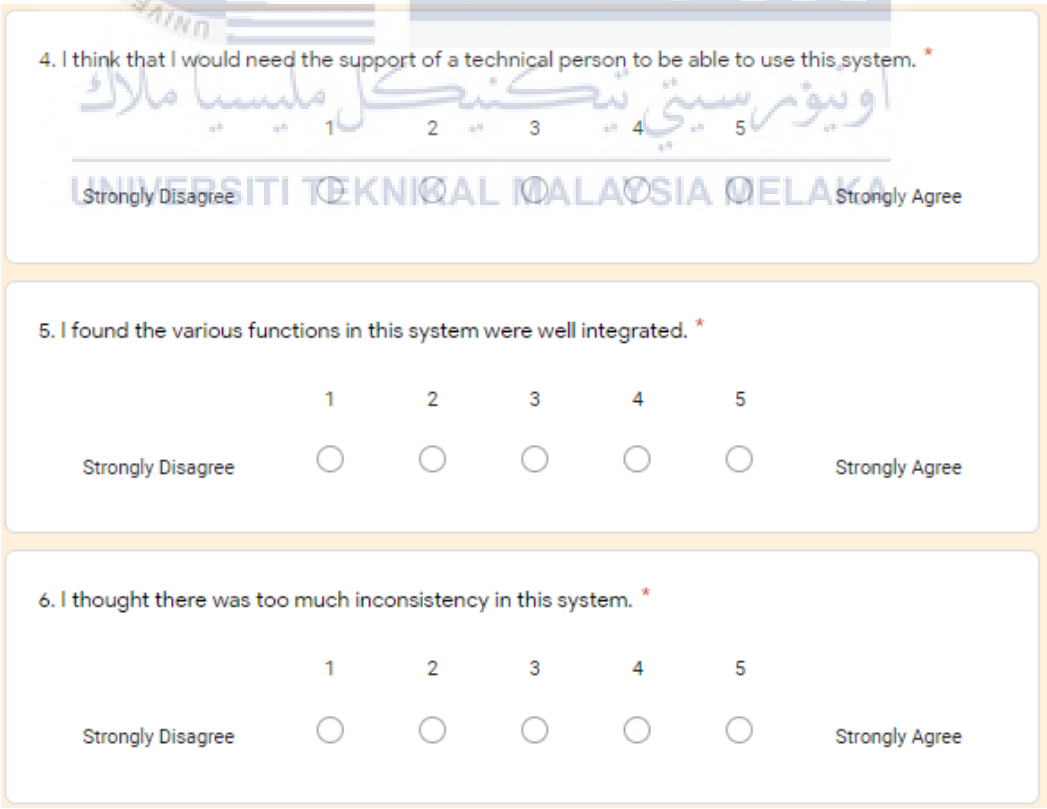


Figure 6.3: SUS questionnaire (Item 4 to 6)

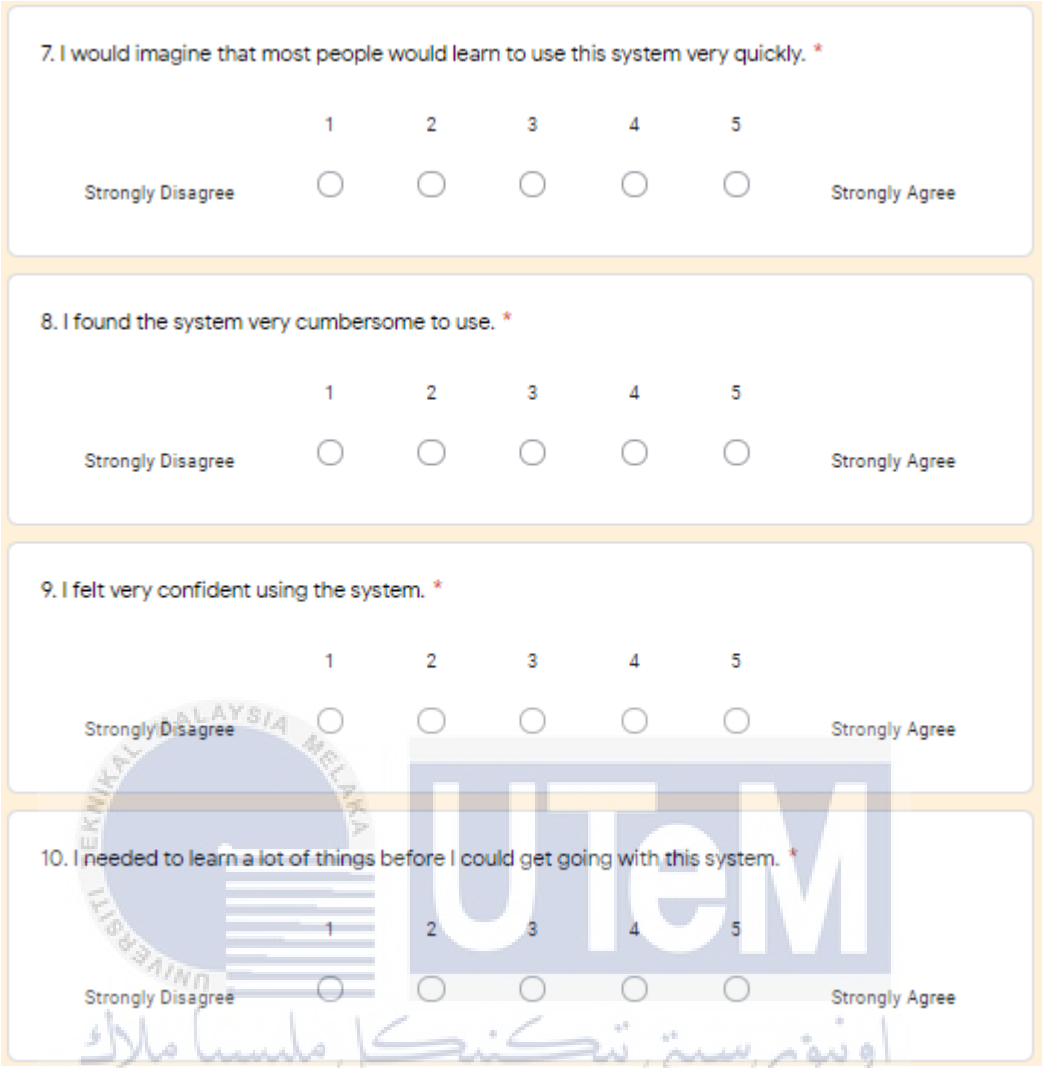


Figure 6.4: SUS questionnaire (Item 7 to 10)

Table 6.4: General guideline on the interpretation of SUS score

SUS Score	Grade	Adjective Rating
> 80.3	A	Excellent
68 – 80.3	B	Good
68	C	Okay

51 - 68	D	Poor
< 51	E	Awful

6.5 Test Results and Analysis

In this part, the result from the testing will be discussed in detail. Analysis of the result also will be covered in this part.

6.5.1 Functionality Test

Functionality test is conducted to test the functionality of the system. The test included face detection and recognition test, Telegram notification test, and solenoid door lock test.

6.5.1.1 Face Detection and Recognition Test

By referring to the table above, the Face Detection and Recognition Test is done. The result of the test is shown below.

Figure 6.5: Face captured matched with the dataset (face recognized).

Figure 6.6: Face captured not matched with the dataset (face not recognized)

6.5.1.2 Telegram Notification Test

When the face detected not matched with the dataset in face database, a notification with the image captured will be sent to user through the Telegram.

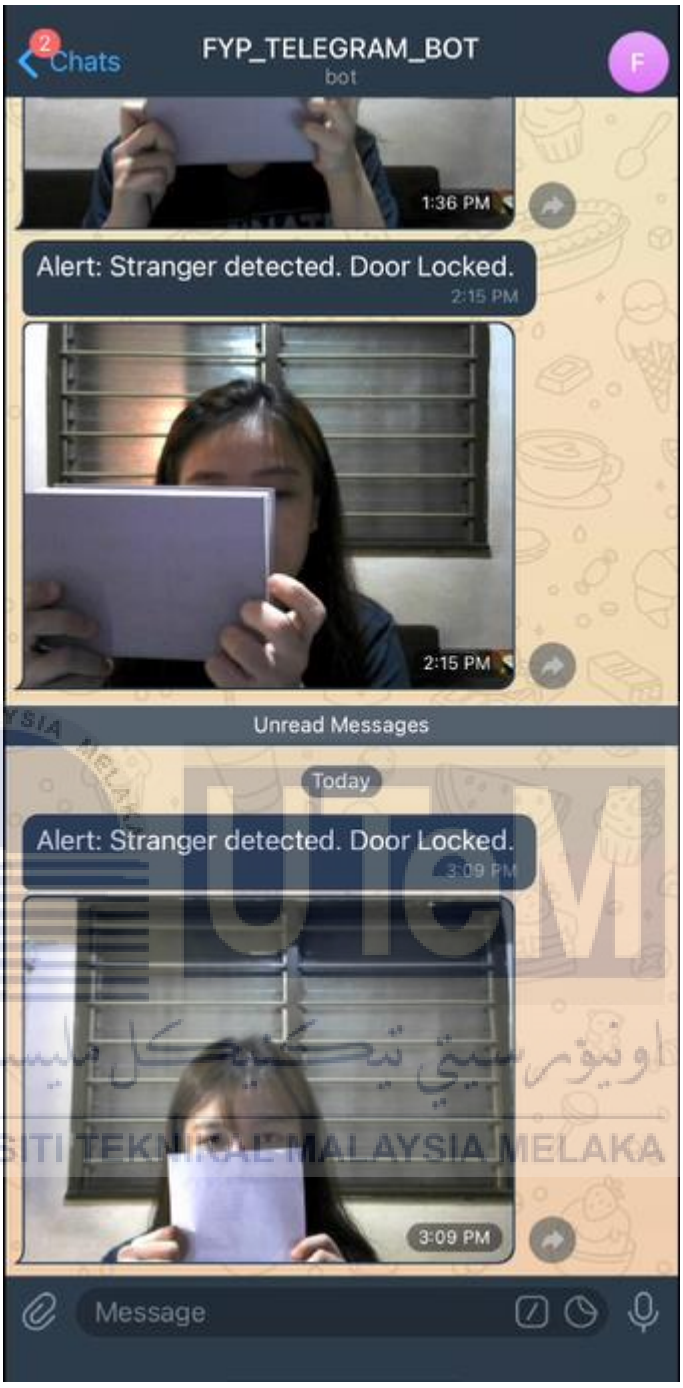


Figure 6.7: Telegram Notification about stranger

6.5.1.3 Solenoid Door Lock Test

When the face captured matched with the dataset, the solenoid door lock is unlocked. At the same time, the solenoid door lock keeps

locking when the face captured not match with the dataset. The result of the solenoid door lock test is shown below.

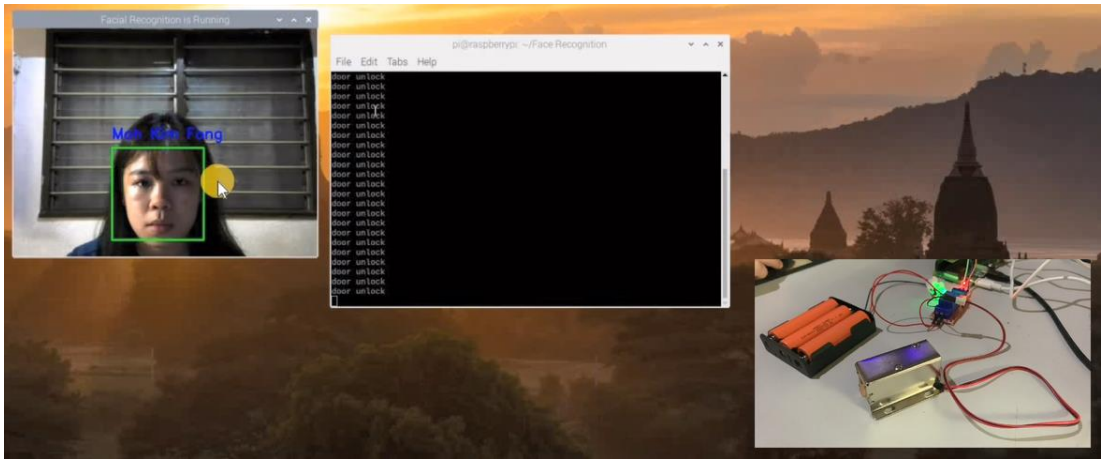


Figure 6.8: The solenoid door lock is unlocked.

Figure 6.9: The solenoid door lock is locked.

6.5.2 Performance Test

Performance test is conducted to test the performance of the whole system. Multiple datasets are created in face database and trained by the system. Then, the respondents tried to use the system and answered the SUS questionnaire to assess the usability of the system. The usability score using SUS is calculated. The results are shown below.

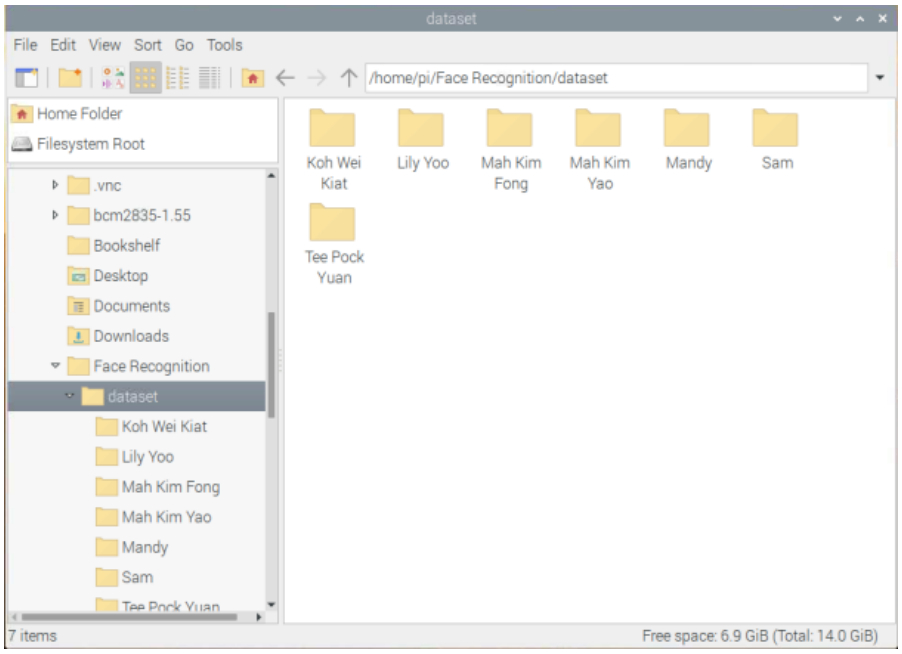


Figure 6.10: Multiple datasets added.

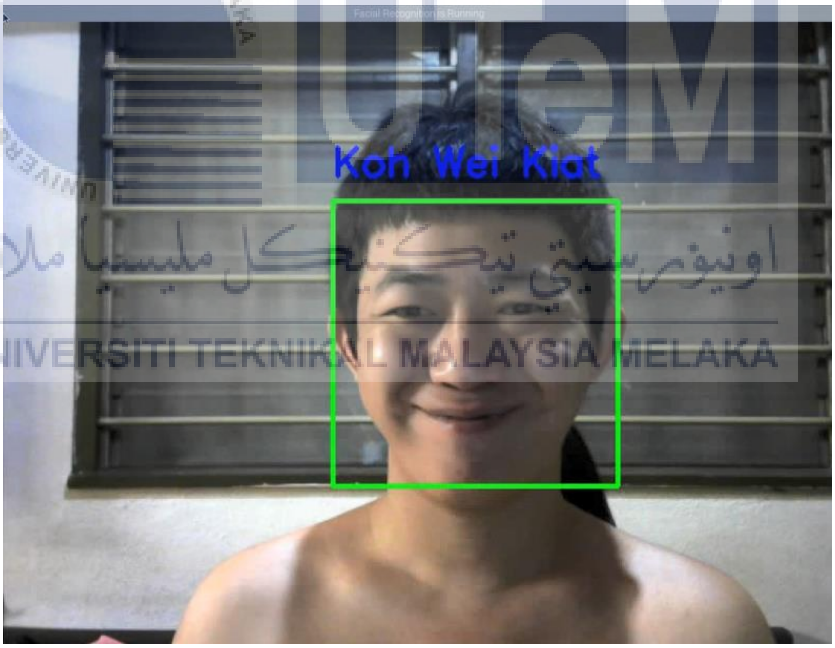


Figure 6.11: The face of respondent (Koh Wei Kiat) is recognized.

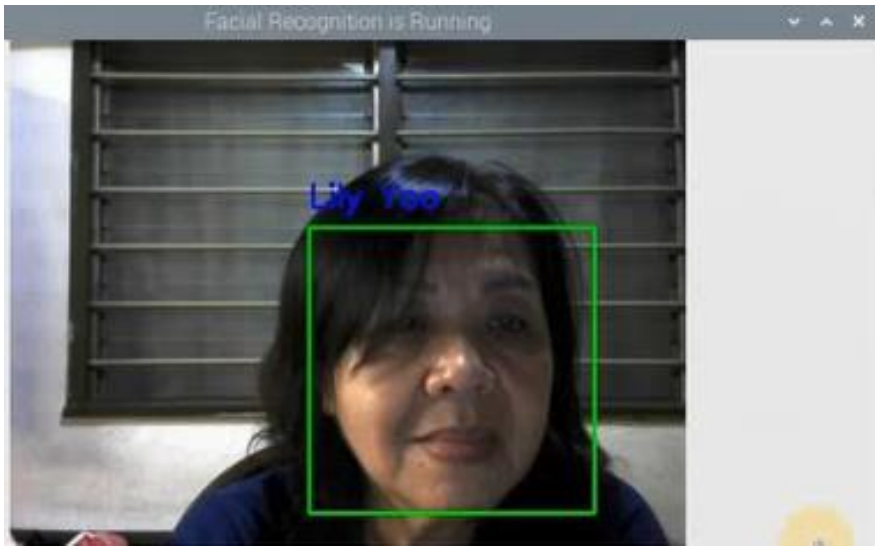


Figure 6.12: The face of respondent (Lily Yoo) is recognized.

Figure 6.13: The face of respondent (Sam) is recognized.

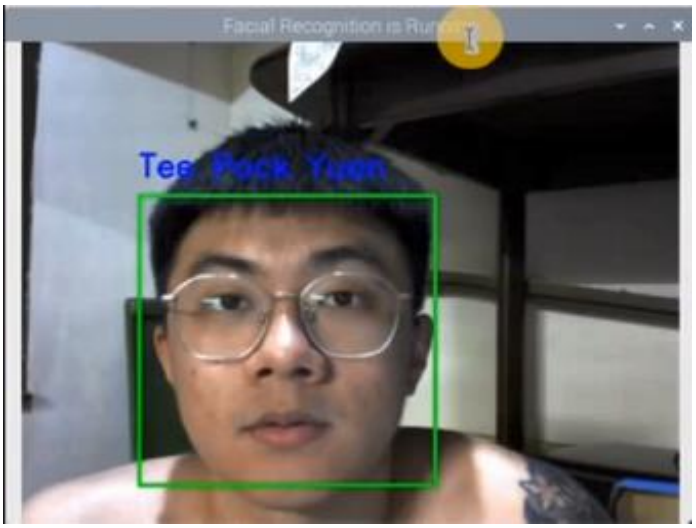


Figure 6.14: The face of respondent (Tee Pock Yuan) is recognized.

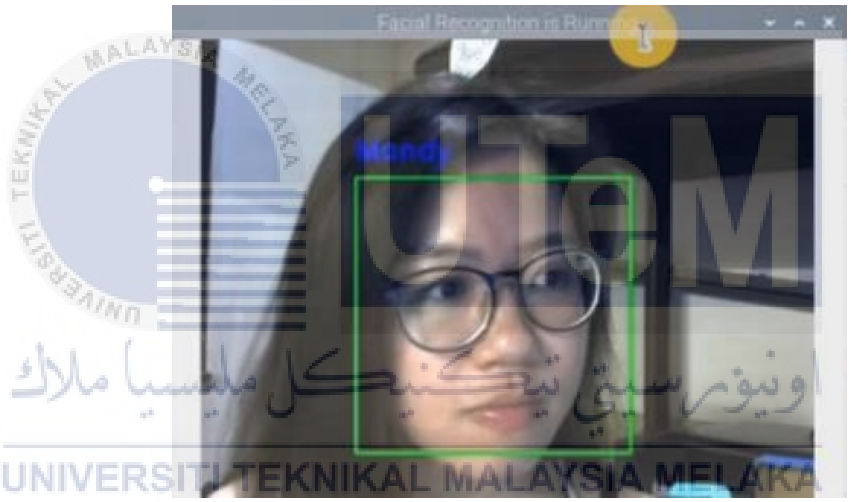


Figure 6.15: The face of respondent (Mandy) is recognized.

Figure 6.16: The face of respondent (Mah Kim Yao) is recognized.

6.5.2.1 System Usability Scale (SUS)

A questionnaire which consists of 10 items with five response options (Strongly agree to Strongly disagree) with score 1 to 5 are answered by the respondents. The results are shown below.

Table 6.5: Responses of SUS

Respondents	Gender	Age	Response/ Score given to item 1 to 10									
			1	2	3	4	5	6	7	8	9	10
Tee Pock Yuan	Male	23	5	2	5	1	4	1	5	1	4	1
Lily Yoo	Female	59	4	2	4	2	5	1	4	2	3	3
Koh Wei Kiat	Male	21	3	1	4	1	3	3	4	1	3	1
Sam	Male	21	4	1	5	1	5	1	5	1	5	1
Mandy	Female	28	5	2	4	1	5	2	4	1	4	1

Mah Kim Yao	Female	20	3	3	5	1	3	2	5	3	3	1
Total Score:			24	11	27	7	25	10	27	9	22	8

Table 6.6: SUS Raw Score and Final Score

Respondents	**SUS Raw Score**	**SUS Final Score**
Tee Pock Yuan	Total score for all odd-numbered questions = 5 + 5 + 4 + 5 + 4 = 23 X = 23 – 5 = 18 Total score for all even-numbered questions = 2 + 1 + 1 + 1 + 1 = 6 Y = 25 – 6 = 19	(18 + 19) * 2.5 = 92.5
Lily Yoo	Total score for all odd-numbered questions = 4 + 4 + 5 + 4 + 3 = 20 X = 20 – 5 = 15 Total score for all even-numbered questions = 2 + 2 + 1 + 2 + 3 = 10 Y = 25 – 10 = 15	(15 + 15) * 2.5 = 75

Koh Wei Kiat	Total score for all odd-numbered questions = 3 + 4 + 3 + 4 + 3 = 17 X = 17 – 5 = 12 Total score for all even-numbered questions = 1 + 1 + 3 + 1 + 1 = 7 Y = 25 – 7 = 18	(12 + 18) * 2.5 = 75
Sam	Total score for all odd-numbered questions = 5 + 5 + 5 + 5 + 5 = 25 X = 25 – 5 = 20 Total score for all even-numbered questions = 1 + 1 + 1 + 1 + 1 = 5 Y = 25 – 5 = 20	(20 + 20) * 2.5 = 100
Mandy	Total score for all odd-numbered questions = 4 + 4 + 5 + 4 + 4 = 21 X = 21 – 5 = 16 Total score for all even-numbered questions	(16 + 18) * 2.5 = 85

	= 2 + 1 + 2 + 1 + 1 = 7 Y = 25 – 7 = 18	
Mah Kim Yao	Total score for all odd-numbered questions = 3 + 5 + 3 + 5 + 3 = 19 X = 19 – 5 = 14 Total score for all even-numbered questions = 3 + 1 + 2 + 3 + 1 = 10 Y = 25 – 10 = 15	(14 + 15) * 2.5 = 72.5
Average:		92.5+ 75 + 75 + 100 + 85 + 72.5 = 83.33333333

The SUS score shows the usability performance of the system in many aspects such as effectiveness, efficiency, and overall ease of use. From table 6.5, the average score of SUS is 83.3333 out of 100. This means that this system is recommended but still needs a minor improvement.

6.5.3 False Test

False test such as face from picture and no internet connection are conducted.

6.5.3.1 Face from Picture

In this test, the picture contain faces will be used to test whether the system will detect the face from the picture or not. The picture of user and the picture of the stranger will be shown in front of the webcam to represent the face since the system using face recognition approach.

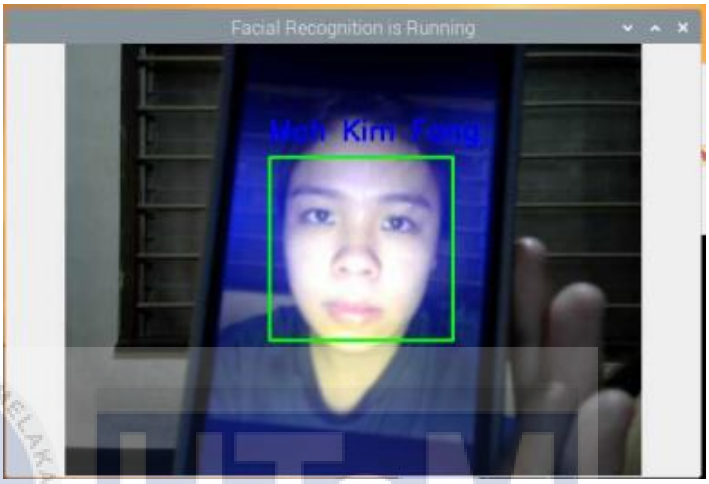


Figure 6.17: Picture of user is tested.

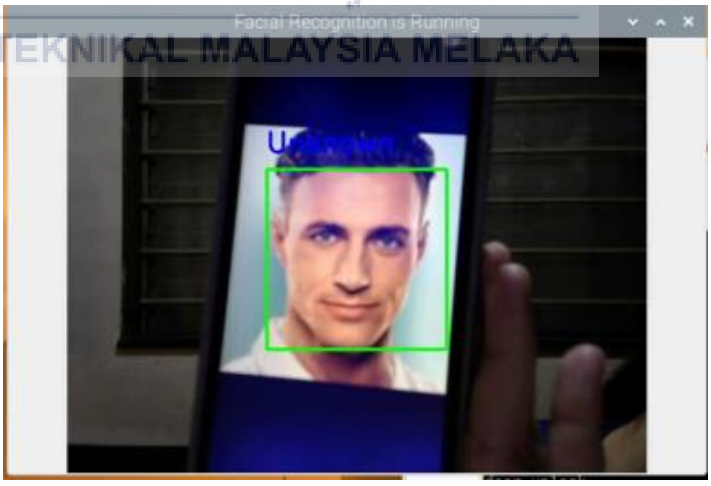


Figure 6.18: Picture of stranger is tested.

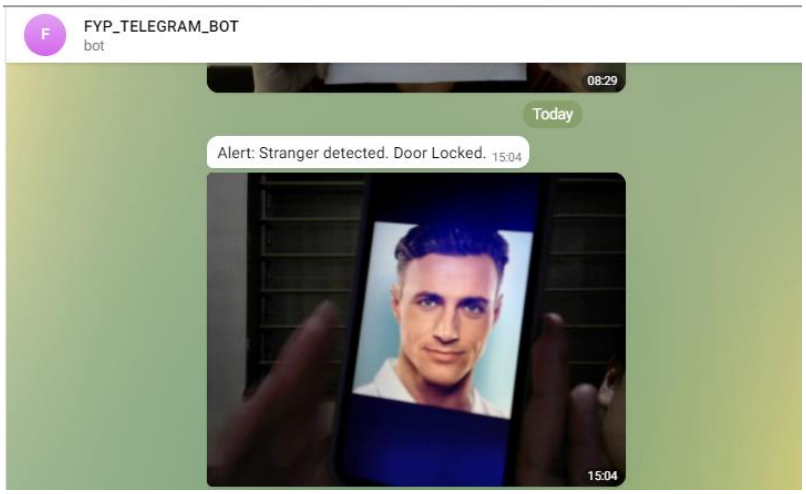
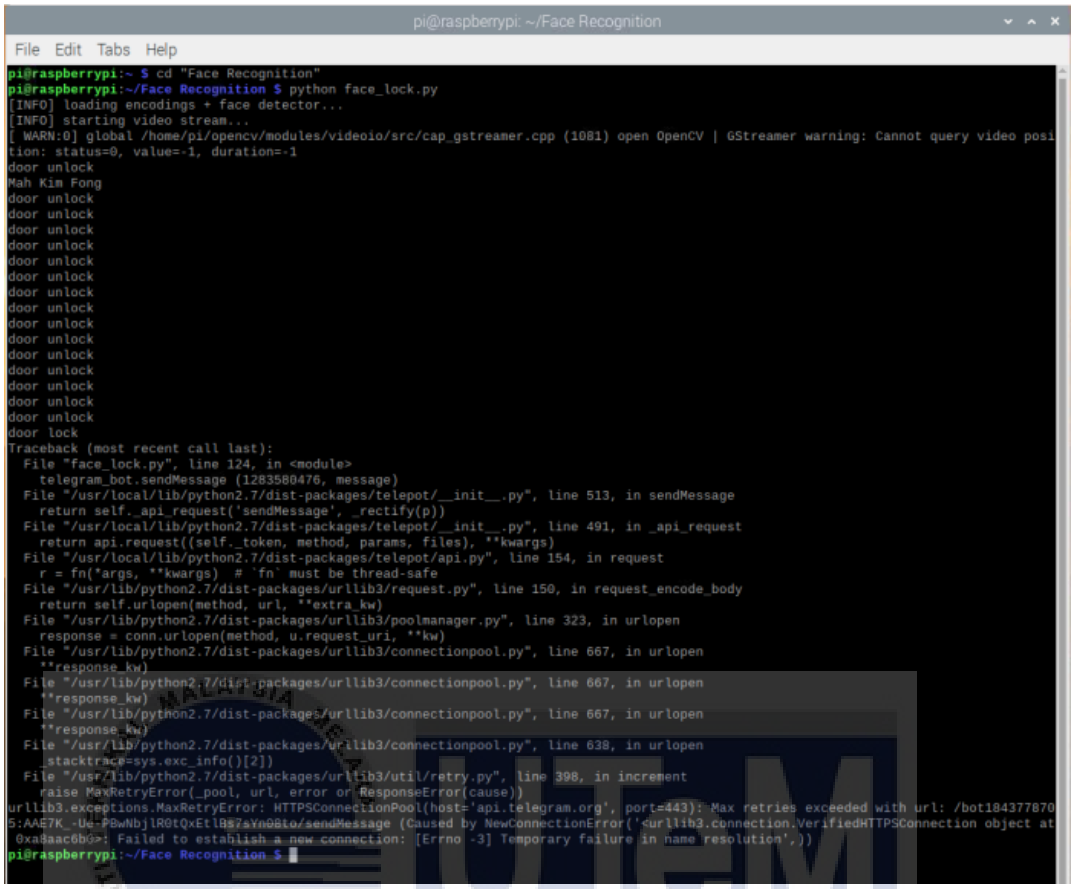


Figure 6.19: Notification received when stranger detected.

From the result above, we can conclude that the system will detect the face from the picture. When the system detected the face of the user from the picture, the solenoid door lock is unlocked. When the system detected the face of a stranger from the picture, a notification is received through the Telegram to notify the user about the existence of the strangers or intruders.

6.5.3.2 No Internet Connection

In this test, Raspberry Pi and the smart phone will be disconnected from the Internet connection to test the availability of the functions without Internet connection.



From the figure above, we can conclude that the device cannot work if there is no Internet connection. Without internet connection, the face recognition function and the solenoid door lock can work as usual. However, when stranger or intruders are detected, it is unable to send the notification to the user through Telegram. An error message will be shown. For the Raspberry Pi to connect to Telegram, an internet connection is required.

6.6 Conclusion

In this chapter, all the testing activities are carried out in detail to verify whether this project is successful or not. Test plan discussed about the planning and activities involved in the test while test strategy guides us to perform the test in a proper manner. The test design provides us the guideline in performing specific test for this project.

As for the result and analysis, we can conclude that the system is working well. It can detect the face of the person appear in front of the webcam and try to match the face captured with the dataset in face database. In the next chapter, overall chapter of this project will be summarized.

CHAPTER 7: PROJECT CONCLUSION

7.1 Introduction

In this chapter, the overall chapter of this project will be concluded. This chapter will discuss about project summarization which include project objective followed by project weakness and strength, project contribution, project limitation, future work, and conclusion of this chapter. This whole chapter is the recap of the whole project.

7.2 Project Summarization

For project summarization, project objective followed by project weakness and strength are discussed as below:

7.2.1 Project Objective

Three project objectives had been achieved in Chapter 5 (Implementation) and Chapter 6 (Testing and Analysis).

Table 7.1: Project Summarization

Objectives	Achieved
Develop a face recognition smart door lock security system.	/

Detect the strangers or intruders through face recognition approach which using Haar Cascade Algorithm and LBP Algorithm.	/
Notify user through Telegram when strangers or intruders detected.	/

7.2.2 Project Strength and Weakness

This project has some strengths and weaknesses. The strength of the project enables us to achieve the project objective and make the project more likely to be succeed. In contrast, the weakness of the project needs our improvement to it. Otherwise, it will affect the progress of the whole project.

7.2.2.1 Project Strength

In this project, some strengths are found. The first project strength is this project is affordable by most people. The equipment used in this project only cost around RM500.00 compared to other face recognition smart door lock security system such as Xiaomi DI R5 Smart Door Lock, Samsung SHP-DP609 and RS PRO Thermal Imaging Door Entry System. In addition, this system enable user to reduce electricity use and cost compares to other face recognition smart door lock security system since only small amount of electricity are required. This is because the power supply required for Raspberry Pi to operate is only 5V. Next, it makes user's life more convenient because they just need to show their face to the system to unlock the door. Moreover, this system can send notification to the user through Telegram to alert the user if strangers or intruders are detected. The image of the intruders will be captured and send together with the notification. It can prevent the massive loss of property and ensure the security of the house.

Table 7.2: Comparison of the price with other smart door lock systems

Face Recognition Smart Door Lock System	Price (RM)
Face Recognition Smart Door Lock Security System Using Haar Cascade Algorithm and LBP Algorithm	500.00
Xiaomi DI R5 Smart Door Lock	1,550.00
Samsung SHP-DP609	2,199.00
RS PRO Thermal Imaging Door Entry System.	5,340.96

7.2.2.2 Project Weakness

Although this project has many strengths, it also has some weaknesses. First, the notification function only works if there is an internet connection. When Python programming are executed and it detects strangers or intruders, a command will be sent to Telegram Bot. For user to receive the alert, internet connection is required. At the same time, if user want to access Raspberry Pi remotely, internet connection is also required. In addition, user is unable to control the face recognition smart door lock security system remotely using Telegram. If the stranger detected but user recognized them, the user still needs to unlock the door for them manually. They cannot use Telegram to control the door lock remotely. Next, the resolution of camera module used in this project is

only 720p. Therefore, the quality of the image captured is not the best and it will affect the accuracy of the face recognition. Furthermore, intruders can use the picture of the user to unlock the solenoid door lock without any notification. The user is unable to track and get notified of it.

7.3 Project Contribution

By implementing this project, it ensures the security of the premises and help to prevent the massive loss of property since strong authentication and authorization method which is face recognition approach is used. Furthermore, the notification function implemented enable users to keep track of surroundings of their premise and allocate the stranger that step in front of the door. Users can act promptly when the strangers or intruders detected. In addition, user life will be more convenient and efficient since the possibility of the physical key being lost, misplaced, or stolen are reduced. This face recognition smart door lock security system can be used by all premises owners.

7.4 Project Limitation

In this project, there are several limitations that can be identified. The first limitation is this project require strong Internet connection. Strong Internet connection is important since the Raspberry Pi is connected to Telegram to send data. Next, additional hardware such as processor coolers are required since the Raspberry Pi processor and solenoid door lock are easy to become hot. In addition, Telegram application can only be used to receive notification and unable to control the face recognition smart door lock security system. Moreover, the surrounding of the camera module must be bright. This is because the quality of the image captured are affected by the type and resolution of the camera module. Moreover, an infrared camera is required. The algorithm used in the program only able to recognize face features but it is unable to differentiate between real face of the user and face of the user in picture.

7.5 Future Work

For the future work, some improvement can be done in this project. Below are the improvements that can be done on this project.

i. The camera module can be replaced by infrared camera to provide a better quality of image and increase the accuracy. This is because infrared camera is not affected by the condition of light and is able to detect the thermal energy and heat emitted. Thus, the weakness for camera module to differentiate real face and face in picture can be solved.

ii. Enable user to control the face recognition smart door lock security system by giving command in the Telegram Bot.

iii. Integrate the sensor with other sensor such as motion sensor to activate the system only when there is a motion in front of the door.

iv. Connect the system with cloud storage to save the log files that contain the information and image of the people detected.

7.6 Conclusion

In conclusion, this project is successful since it achieves all the project objective stated in 7.2.1. This chapter is the summarization of the whole chapter regarding to this project. The strength of this project has enables us to achieve all the project objective, whereas the weaknesses found in this project must be improve so that the system implemented is more practical to be used. Project contribution and project limitation also has been identified and discussed in detail. Even the project is complete and meets all the objective, there are still some improvements that can be done in this project in the future to produce a better system.

REFERENCES

Al-Tuma, K. A. H (2019). Smart Door Lock Based on Bluetooth and IoT. Degree Thesis, Universiti Tun Hussein Onn Malaysia https://core.ac.uk/download/pdf/328819789.pdf

Kulhary, R 2021, GeeksforGeeks. accessed 20 June 2021, https://www.geeksforgeeks.org/opencv-overview/

OpenCV team 2021, OpenCV, accessed 20 June 2021, https://opencv.org/about/

Emami, S., Suciu, V.(2012). Facial Recognition using OpenCV. Journal of Mobile, Embedded and Distributed Systems. 4. https://www.researchgate.net/publication/267426877_Facial_Recognition_using_OpenCV

Fortune Business Insights 2021, accessed 20 June 2021, https://www.fortunebusinessinsights.com/face-recognition-door-lock-market-105039

Transparency Market Research 2020, accessed 20 June 2021, https://www.transparencymarketresearch.com/digital-door-lock-systems-market-2017-2025.html

Gunawan, T. S., Gani, M. H. H., Rahman, F. D. A. and Kartiwi, M.(2017). Development of Face Recognition on Raspberry Pi for Security Enhancement of Smart Home System. Indonesian Journal of Electrical Engineering and Informatics. 5. 317-325. 10.11591/ijeei.v5i4.361. https://www.researchgate.net/publication/322009772_Development_of_Face_Recognition_on_Raspberry_Pi_for_Security_Enhancement_of_Smart_Home_System

Najmurrokhman, A., Kusnandar, K., Krama, A. B., Djamal, E. C. and Rahim, R. (2018). Development of a secured room access system based on face recognition using Raspberry Pi and Android based smartphone. MATEC Web of Conferences, 197, 11008. https://doi.org/10.1051/matecconf/201819711008

Faisal, F., Hossain, S. A. (2019). Smart Security System Using Face Recognition on Raspberry Pi. 1-8. 10.1109/SKIMA47702.2019.8982466.

https://www.researchgate.net/publication/339095435_Smart_Security_System_Using_Face_Recognition_on_Raspberry_Pi

Zedan, M. J. M., Al-Jbaar, M. (2020). Face Recognition Security System Based on Convolutional Neural Networks. 29. 5177-5187. https://www.researchgate.net/publication/342492530_Face_Recognition_Security_System_Based_on_Convolutional_Neural_Networks

Anggo, M., Arapu, L. (2018). Face Recognition Using Fisherface Method. Journal of Physics: Conference Series, 1028, 012119. https://doi.org/10.1088/1742-6596/1028/1/012119

Radzi, S. A., Alif, M. K. M. F., Athirah, Y. N., Jaafar, A. S., Norihan, A. H. and Saleha, M. S. (2020). IoT based facial recognition door access control home security system using raspberry pi. International Journal of Power Electronics and Drive Systems (IJPEDS). 11. 417. 10.11591/ijpeds.v11.i1.pp417-424. https://www.researchgate.net/publication/339606773_IoT_based_facial_recognition_door_access_control_home_security_system_using_raspberry_pi

Prabhakar, A. Y., Oza. S. K., Shrivastava. N., Srivastava. P. and Wadhwa. G. (2019). Password Based Door Lock System. IRJET, 06(02), 1154-1157. Retrieved from https://www.irjet.net/archives/V6/i2/IRJET-V6I2225.pdf

Bhat, A. K. and Kini, S. P. (2018). Password Enabled Door Locking System using Arduino and IoT. IJERT. https://www.ijert.org/password-enabled-door-locking-system-using-arduino-and-iot

Tamunowari, M. (2018). Design and Construction of an Electronic Door access key. https://www.researchgate.net/publication/338690665_Design_and_Construction_of_an_Electronic_Door_access_key

Okpara, C.R. and Ononiwu, (2017). Automated Door Lock System Using Arduino. https://www.researchgate.net/publication/344256856_Automated_Door_Lock_System_Using_Arduino

Rahman, M. M., Ali, M. S. and Akther, M. S. (2018). Password Protected Electronic Lock System for Smart Home Security. 10.13140/RG.2.2.23684.63365. https://www.researchgate.net/publication/325093383_Password_Protected_Electronic_Lock_System_for_Smart_Home_Security

Verma, G. K. and Pawan, T. (2010). A Digital Security System with Door Lock System Using RFID Technology. International Journal of Computer Applications. 5. 10.5120/957-1334.

https://www.researchgate.net/publication/45602075_A_Digital_Security_System_with_Door_Lock_System_Using_RFID_Technology

Suhail, R. (2020). Automated door access based on RFID using Arduino. 10.4108/eai.27-2-2020.2303113. https://www.researchgate.net/publication/343575758_Automated_door_access_based_on_RFID_using_Arduino

Sarma, M., Gogoi, A., Saikia, R. and Bora, D.J. (2020). Fingerprint Based Door Access System using Arduino. https://www.researchgate.net/publication/343524392_Fingerprint_Based_Door_Access_System_using_Arduino

Hossain, M. K. (2019). Smart Door Lock System with Fingerprint Interface. https://www.researchgate.net/publication/336279067_Smart_Door_Lock_System_with_Fingerprint_Interface

T, W. 2021, UIUX Trend. accessed 10 August 2021, https://uiuxtrend.com/measuring-system-usability-scale-sus/

Sauro, J. (2016). Measuring Usability With The System Usability Scale (SUS). https://www.userfocus.co.uk/articles/measuring-usability-with-the-SUS.html