

A Smart Face Recognizing Offline Lock

BY

**Afnanul Hassan
ID: 161-15-7213**

**Zaria Fahmida
ID: 161-15-7167**

**Proloy Sen
ID: 161-15-7549**

This Report Presented in Partial Fulfillment of the Requirements for the
Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

Prof. Dr. Syed Akhter Hossain
Head
Department of CSE
Daffodil International University



DAFFODIL INTERNATIONAL UNIVERSITY
DHAKA, BANGLADESH
DECEMBER 2019

APPROVAL

This Project titled **A Smart Face Recognizing Offline Lock**, submitted by Afnanul Hassan, ID No: 161-15-7213, Zaria Fahmida, ID No: 161-15-7167, Proloy Sen, ID No: 161-15-7549 to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 7th December, 2019.

BOARD OF EXAMINERS



Dr. Syed Akhter Hossain
Professor and Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Chairman



Md. Zahid Hasan
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner



Sadekur Rahman
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner



Dr. Dewan Md. Farid
Associate Professor
Department of Computer Science and Engineering
United International University

External Examiner

DECLARATION

We hereby notify that, this project has been done by us under the supervision of **Prof. Dr. Syed Akhter Hossain, Head, Department of CSE** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:



Prof. Dr. Syed Akhter Hossain
Head
Department of CSE
Daffodil International University

Submitted by:



Afnanul Hassan
ID: -161-15-7213
Department of CSE
Daffodil International University



Zaria Fahmida
ID: -161-15-7167
Department of CSE
Daffodil International University



Proloy Sen
ID: -161-15-7549
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

First we express our heartiest thanks and gratefulness to almighty God for His divine blessing makes us possible to complete the final year project/internship successfully.

We really grateful and wish our profound our indebtedness to **Prof. Dr. Syed Akhter Hossain, Head**, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of “*Face Detection and Recognition*” to carry out this project. His endless patience ,scholarly guidance ,continual encouragement , constant and energetic supervision, constructive criticism , valuable advice ,reading many inferior draft and correcting them at all stage have made it possible to complete this project.

We would like to express our heartiest gratitude to all the faculty member of Department of CSE, for their kind help to finish our project and also to other staff of CSE department of Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

ABSTRACT

In the 21st century digital lifestyle in a smart city, security is one of the core ingredients in order to ensure the digital continuity. For ensuring money, premises and personal belongings safe and secured from unauthorized person this project is conceptualized. This proposed project titled "**A Smart Face Recognizing Offline Lock**" a system developed for protecting assets using digital system. Although this system works even without Internet that means a user can use it through offline by an android application. The system will furnish a complete security solution and making disabled to access for unauthorized people. Prohibited access is strictly controlled through the design of this system. The system stores images of users and only authorized person will be granted by this system. A user can register his image through an android application to unlock his locked device. The proposed system is implemented using Python programming language with OpenCV. As the realm of this research model, image detection are classified, and 94% accuracy are achieved which can obviously secure our principal things.

TABLE OF CONTENTS

CONTENTS	PAGE
Approval	ii
Declaration	iii
Acknowledgements	iv
Abstract	v
CHAPTER 1: INTRODUCTION	1-3
1.1 Introduction	1
1.2 Motivation	1
1.3 Objectives	2
1.4 Expected Outcome	2
1.5 Report Layout	3
CHAPTER 2: BACKGROUND STUDY	4-7
2.1 Introduction	4
2.2 Literature Review	5
2.3 Challenges	7

CHAPTER 3: METHODOLOGY	8-18
3.1 Introduction	8
3.2 Methodology	8
CHAPTER 4: DESIGN SPECIFICATION	21-25
4.1 Front-end Design	21
4.2 Back-end Design	22
4.3 Experimental Result	22
4.4 Implementation Requirements	25
CHAPTER 5: IMPLEMENTING AND TESTING	26-29
5.1 User Registration with face detection:	26
5.2 Training Data	27
5.3 Testing Implementation	28
5.4 Test Results and Reports	28
CHAPTER 6: CONCLUSION AND FUTURE SCOPE	29
6.1 Discussion and Conclusion	29
6.2 Future Scope	29
References	30

LIST OF FIGURES

FIGURES	PAGE
Figure 3.1: Registrar user using android app	9
Figure 3.2: Face is enough to open the lock	9
Figure 3.3: Finding region of interest (ROI)	12
Figure 3.4: Flow Chart of creating dataset.	13
Figure 3.5: Applying LBP Operation	15
Figure 3.6: Visualization of circular LBP in large scale	16
Figure 3.7: Extracting Histograms	17
Figure 3.8: Flow Chart of whole face detection process	18
Figure 3.9: Unlock Process of lock	20
Figure 4.1: Android Interface to registrar new user	21
Figure 4.2: Dataset	22
Figure 4.3: New User Registration	23
Figure 4.4: Detect ROI to get face data	23
Figure 4.5: Trained Dataset	24
Figure 4.6: Recognized the face to unlock	25

Figure 4.7: System worked based on face	25
Figure 5.1: User registration Interface	26
Figure 5.2: Detected face to unlock	27
Figure 5.3: The matrix of .yml file after training	27

CHAPTER 1

INTRODUCTION

1.1 Introduction

According to “Numbeo”[12] the percentage of home robbed and things stolen is 70.5% of the total crime in last ten years. And this is escalating day by day.

Even with the utilization of locks, the crime, robberies get happened because of the very fact that such locks were simply breakable. So, there's a necessity to create alternative quite locks that cannot be simply broken.

Several researchers also proposed completely different sorts of digital door locks, automatic countersign based mostly door locks, code based door locks etc.

So we wanted to use a smart lock for home security which will be unlocked by user's image detection. Because face-detection algorithms navel on the detection of frontal human face, it's parallel to image detection within which the image of an individual is matched bit by bit. The image matches the image stores in the information. Any change to the facial features of the data, will rule out the system. So that without any authorized person nobody can enter.

1.2 Motivation

According to Bangladesh Police only in year of 2018 approximate a number of 6123 cases have been filed for robbery and theft in houses, offices or in any other places. Beside that many people even didn't filed any complain to Police. Security is one of the principal need of any person. So we thought that we could reduce this problem by using face detection method. Face-detection is used massively for any security purpose now a days. Our proposed method is capable of preventing unauthorized entry.

As for the purpose, we are extremely motivated to establish a system to secure home security and recognize exotic persons to prevent crime.

1.3 Objectives

The main objectives of this project is to secure security of Home, Offices or any places.

- To prevent externals from entering secure places.
- To identified the criminals who wanted to enter secure places illegally.
- To make security easier.
- To construct a smart security system.
- To make a secured country.

1.4 Expected Outcomes

In this digital era criminal are also using digital techniques to do crime. So it's difficult to secure any place with basic gadgets. For that we wanted to use modern technology. By using Face-detection we could solve this problems. Some aspects come from our projects are given below-

- This projects can be applied at any places like house, office, or any institutions.
- This system will keep places secure.
- This model will prevent any unauthorized person from entering any secure places.
- Only authorized persons will get access to enter.
- It will reduces stole, robbery at home or any other places.

1.5 Report Layout

Chapter 1: Introduction- The introduction, motivation, objectives and expected outcome of the project work we have done is written here and the report layout.

Chapter 2: Background- The following chapter is all about the background situation of our work that we have done. We also provided the literature review of project, Relative studies and tests of the system.

Chapter 3: Project methodology – This chapter has discussed about the method we used to construct the system. This sector has the methods and steps, data collection procedure, some numerical analysis of the cited system.

Chapter 4: Experimental results and discussion: - This chapter is discussed all the experimental result that has been attained by the proposed system is discussed along with the presentation analysis and a summary of the result is covered. Beside that we submitted the system design in this chapter.

Chapter 5: Conclusions- In This chapter contains the conclusion part and the ideas of implication of further study on this topic.

CHAPTER 2

Background Study

2.1 Introduction

In this chapter we will discuss on various research work. A lot of work has been done regarding this topic. The concept of locks and keys were originate almost 4000 years ago. The conventional door lock uses the key which was very easy to access any unauthorized person by using a duplicate key. In addition, if the owner loses the key they will not be able to open the door and need to wait more time for coming the technician. This is a process that took a long time. To solve against this baulk, some security systems have been developed with the advanced technology such as face detection, fingerprint based door locking system, optimized door locking and unlocking using IOT, automatic door locking and unlocking etc. Offline Smart Door Lock based on face detection is an android based project where we proposed a system by using OpenCV and Python language. In this method door will be unlocked by identifying and recognized images of any actual authorized person.

Facial recognition system is a technology that can identify or approve a person from a digital image or video frame. There are multiple ways within which biometric identification systems work, however generally, they work by examination elite facial expression from given image with faces at intervals an information. it's additionally delineate as a Biometric computing primarily based application which will unambiguously establish an individual by analyzing patterns supported the person's facial textures and form.

In this system we used OpenCV to recognize the images. Normally OpenCV (Open supply computer Vision Library) is associate open supply computer vision and machine learning package library. OpenCV was engineered to produce a standard infrastructure for computer vision applications and to accelerate the employment of machine perception within the business product. It is an open source media. So everyone able to use it.

For face recognition the main two parts are

- Face Detection.
- Face Identifying.

At first we detect the face in front of our camera by using OpenCV. Then the image will be tested by LPHG algorithm. And it will be matched with the registered image. If the match is completed successfully door will be unlocked easily.

2.2 Literature Review

Onoyan AO et al. [1] developed biometric based door lock system by using ATMEGA 328 ARDUINO microcontroller, fingerprint scanner R305. This system consisted into two parts fingerprint inscriptions and fingerprint resemble. At first, it stores the verified fingerprint information by scanning and then when a person puts his/her fingerprint on the sensor, the sensor captures and estimates with the verified fingerprint library in microcontroller. By using C language the microcontroller was programmed. If the fingerprint matches with the registered fingerprint it will send a signal to incite the door lock process and the door will open for 5 seconds then it closes automatically. Initially LCD displayed the message for a user “insert fingerprint on the sensor” and after completing above process if it matches the text converted into “access granted”. If the fingerprint doesn’t matches with the registered fingerprint the door will be obstructed and LCD will show “Access Denied” also the buzzer will be actuated. The note is that LCD displays the name with specific fingerprint.

Adarsh V Patil et al. [2] manifested an android based smart door locking system. This paper proposed the cost effective solution with the advanced technology. This system provides a complete security for Business organization, home, school and office etc. Arduino Uno, android, smartphones are used to complete this lock security system. It works on pre-determined password concept. By using Arduino Uno, this system accepts the indication from the smartphone. It controls the electric door strike. Then the Bluetooth module HC-05 is used for communicating between the smartphone and Arduino Uno. The system is also multimode functionalities. A user can login from an android smart phone then entered the password from the keypad. If the password is correct the door is opened by the motor. After entering the door using GSM SIM 800 a user can send message to others user. Apart from these, there are many flexibilities like a

user can reset the password if he/she forgets the password or if anyone feels insecure about his password. Recovery password will be available after forgetting the registered email address. In future the system can also implement the face recognition, voice conversation process for better security.

Anuradha. R.S et al. [3] had proposed an optimized locking and unlocking a system by using wireless system. It is an application for smart home automation system for ensuring our money, data, premises and personal belonging safe and secured from an unauthorized person. They developed their proposal by using Arduino Uno, DC motor, Wi-Fi. The ideal purpose of this systems is a user can lock and unlock a door within Wi-Fi range from inside or outside a house. Create an Android application consisting of two buttons, on and off. The system sends a signal to door from a computer or Tablet or a mobile devices by using wireless system. Wireless LAN module fixed in Arduino Uno board. When a user clicks a button, the door locked, otherwise it is unlocked. The interesting part of the work is if a user forget to lock the door they can lock/unlock it from the first floor or any other floor within the available rang

Agbo David O et al. [4] implemented a door locking system by using android app and the Bluetooth module. The app can be installed on any android smart phone. There were two buttons “open door” and “close door”. The app was connected with the Bluetooth module HC05 that shows the current activity about the door is open or close. The Arduino Uno controls the locking unlocking process. When a user entire the password the application requests a permission to turn on the Bluetooth. From the android smartphone the Bluetooth receives the commands and passes to the microcontroller. If the password is correct the door will open otherwise it will be closed.

Janaki Venukumar et al. [5] Security is the most vital issue nowadays for us when we are out of our households. In these days trustable solution for the problem is not yet properly identified. Here presented an automated locking system where Arduino plays a vital role to process the whole thing. Arduino is basically a microcontroller board. It is a simple tool. The ability of sensing, monitoring, storing and controlling had made it unique. Arduino Uno can be perfectly used to control access. The project shows a keyless system for operating lock using a pre-defined password.

Lia Kamelia et al. [6] Present days are the era of living in a smart technologies. So that, the term “smart home” gives a wide thinking to us of living in new technologies. Windows based PC are highly used to do these types of projects. We have worked hard to make the smart home feeling just using the device of our pocket, the mobile phone where Bluetooth will be enough to communicate with devices.

2.3 Challenges

We faced some problem with implementing our project.

1. If we run our project in a low brightness environment, it will not work correctly. Therefore, we need a pleasant atmosphere with proper brightness.
2. We need a high quality-based camera to implement this project. So, it is a very cost-effective problem.
3. If anyone registers with an image, unique id and name, the lock will be open.

CHAPTER 3

RESEARCH METHODOLOGY

3.1 Introduction

The method of completing the project is going to be written here. We have made our research on face detecting systems and Embedded Systems. We have embedded some devices including Camera and installed face detecting system on it. Beside of using hardware parts we also have an android app to give user easily usable interface to registrar.

In previous chapter we discussed about how our system will work. We have also discussed some others work also. This chapter will have discussed about the system procedure what we are used to develop this system. Before starting to develop this system, we studied about some related works of our system. We found out the appropriate algorithm we have to use, which algorithm will be suitable for this system. We have used Local Binary Patters histogram (LBPH) for identifying the face with Haar cascade classifier for detecting the human face. We also have used Raspberry Pi to unlock the door.

3.2 Methodology

The system will start with registering users face with information. For every lock the 1st user will be able to registrar without hassle. Thus, the lock will be paired with 1st user. Then if any family member needs to do registrar s/he have to give any previous user's unique information like ID with name. This additional procedure will help the lock to be registered with any new stranger person. Without this system any user could be able to come to near the lock registrar himself with the lock to open using his face. This could be a real disaster.

After completing the registration process throughout Bluetooth, the data will be saved. And lock itself will capture some pictures of the user.

Our written code will train the device with newly captured pictures. Thus, the lock will be ready to work with. To train our lock we have used python programming language. Opencv which is an object detecting library written by intel has many build-in methods to

train our pictures. While the training is done a .yml file will be created that have all necessary data to detect a face.

After completing all of these where user just need to give his/her data using android, our lock will be ready to work. Whenever the lock will find out the registered face, it will be unlocked itself. After staying unlocked for 7 second it will be ready for being locked again.

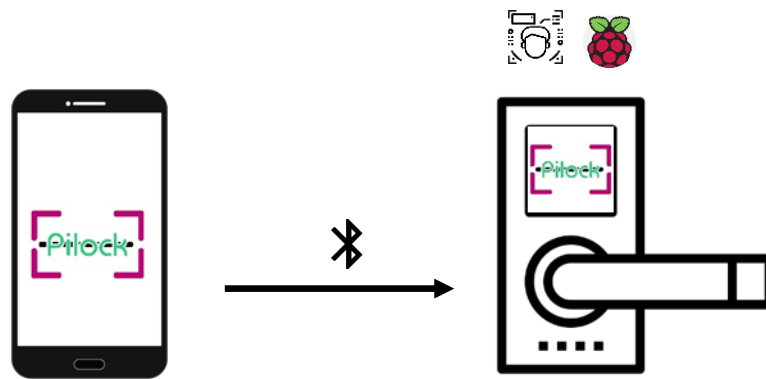


Figure 3.1: Registrar user using android app

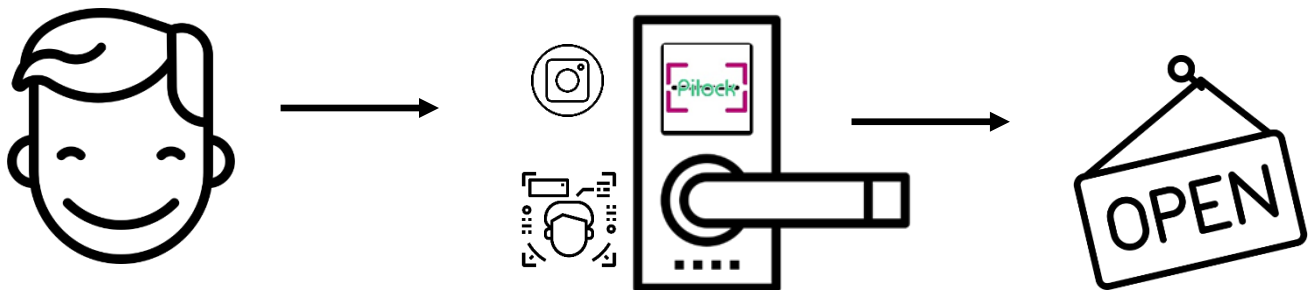


Figure 3.2: Face is enough to open the lock

Now the methodology is going to be elaborately described.

In our system, it will work with 5 main steps which are given below

1. Face Detection & Data set create
2. Train Data Set
3. Face Recognition
4. Send the data through Bluetooth to Raspberry Pi
5. Unlocked the lock.

- 1. Face Detection and Data set create:** Face recognition technology is used in many face recognition applications. Visual scene can be created where the location of human and their faces with the process of psychology. Here we need to detect face in real-time.

Now Some of the important topics are described in detail:

- **OpenCV:** Opencv is an Open Source Computer Vision Library. This is a computer vision and ML (Machine Learning) software library. The library is open source[7]. Opencv is licensed by BSD.
- **What Do OpenCV Do:** OpenCV has about 2500 algorithms. A huge number of built-in methods help user in various ways.
 - Detect faces and recognize faces,
 - Object can be identified,
 - Human actions can be classified from video,
 - camera movements can be tracked,
 - moving objects can be followed,
 - 3D models of any objects can be extracted,
 - 3D point clouds can be produced by using stereo cameras,
 - to produce a high-resolution image OpenCV can stitch images together of an entire scene,
 - similar images can be identified from any database,
 - red eyes from images occupied using flash can be easily detached,
 - follow eye activities,
 - recognize background and found markers to overlay it with augmented reality etc.
- **NumPy:** In Python NumPy is the central package for technical computing. It is a Python library that offers a multidimensional array object [8], It is a Python library that offers a multidimensional array object, various resulting objects, and an assortment of routines for fast processes on arrays, including mathematical,

logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random simulation and much more.

the *ndarray* object is the core of the NumPy package. This summarizes *n*-dimensional arrays of homogeneous data types, with many processes being performed in compiled code for performance. There are some important variances between NumPy arrays and the standard Python sequences:

- NumPy arrays work with a fixed length, Python lists (which can grow dynamically) is quite unlike with it. A new array can be created by changing the size of a ndarray and also let it delete the original.
 - The NumPy array must-have elements of the same data type, which can affect the operation if it is not, and, therefore, will have the same size in memory. Exception: you can have arrays of objects (Python, including NumPy), which allows you to use arrays of elements of different sizes.
 - NumPy arrays facilitate complex mathematical and other types of operations with large amounts of data. Typically, such operations are executed more efficiently and with less code than is possible using Python's built-in sequences.
- **Cascade Classifier:** Firstly, the classifier is trained with the help of several hundred samples of views of a particular object, called positive examples that are the same size, and negative - arbitrary images of the same size.

After learning the classification, it can be applied to any area of interest in the input image. An area object can be displayed and otherwise display "0" in the Search window classifier can be moved around the image to search for an object in the entire image, and each position can be checked using the classifier. The hierarchy is designed in such a way that it can easily find points of interest in various forms, which is more efficient than resizing the image itself. Thus, the scanning method must be performed several times at different scales in order to find an object of unknown size in the image.

The word "cascade" in the name of the classifier means that the resulting classifier consists of several simpler classifiers (steps), which are subsequently applied to the area of interest until at some stage the candidate is rejected or all steps are completed. The word "elevated" means that the classifiers at each stage of the cascade are complex in themselves, and they are built from basic classifiers using one of four different methods of raising (weighted voting). Currently supported

are Discrete Adaboost, Real Adaboost, Gentle Adaboost, and Logitboost. The basic classifiers are classifiers of the decision tree with at least 2 sheets. Haar-like features are input to the main classifiers and are calculated as described below. The current algorithm uses the following haar-like functions:

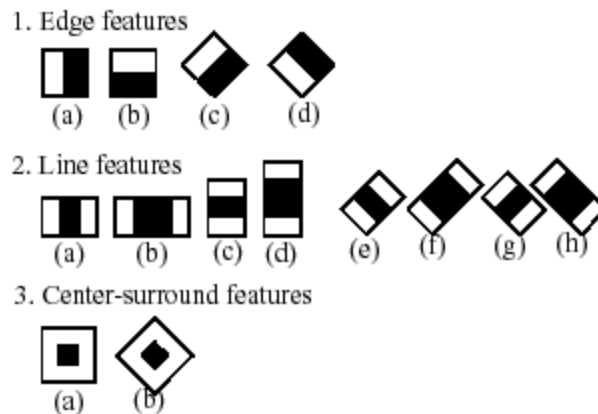


Figure 3.3: Finding region of interest (ROI)

The characteristic used in a specific classification is characterized by its shape (1a, 2b, etc.), its location in the region of interest and its scale (this scale is not the same as used in step 1), although these two scales are manifold). For example, in the third line (2c), the answer is calculated as the difference between the sum of the pixels of the image below the rectangle covering the entire function (two white bands and the black band in the middle) and the sum of the pixels of the image under the black band multiplied by 3 to fill the field size differences. Was done the sum of the values of the pixels in a rectangular region is calculated quickly using the integral image.-

- **Haarcascade_frontalface_default.xml:** this is the trained classifiers for detecting objects of a particular type[10][11].
- **Converting into Gray-Scale:** Various reasons for converting RGB image into grayscale image is,
 1. Noise signal. For many image processing applications, color information does not help us identify important edges or other features. There are exceptions. Color information can be detected if the color is marked on a grayscale image or if we need to detect known color objects (orange fruit in front of the green

leaf), we can consider the word if color is not needed. At first, the "scale" of grayscale is a bit flexible, but you get used to it.

2. Code complexity. If you want to find edges based on brightness and data, you need to do more if extra color information is not useful for the application of interest, that extra work (and additional debugging, extra pain in software support, etc.) is not.
3. To learn image processing, you should first understand gray processing and how to apply it to multi-channel processing, not start with full-color imaging and miss all the important insights that can be learned from single-channel processing.
4. Difficult to visualize. In gray images, the concept of aquatic algorithms is very conceptual because we can think of two spatial and one-dimensional dimensions as a three-dimensional image of mountains, valleys, water basins, veins, and so on. There are many algorithms that have intuitive "physical" explanations that help us think about our problems. This kind of visualization is much more powerful in RGB, HSI, laboratories and other color space because there are additional dimensions that standard human brains cannot easily imagine. Sure, we might think of it as a "red peak" but what does that mountain (x, y, h, s, i) look like in space? Second, one description is that any color variable is thought of as a high-intensity image, but this goes back to our gray-scale image processing.
5. Color is a complex thing. People understand colors and recognize colors with captivating ease. If you want to distinguish colors, (a) you want to keep track of results and ensure the best results, light, brightness, color camera calibration, and Control other factors. The following is a long-term business trip on a topic you want to resolve, or (c) We can give you black-and-white comfort, but at least it seems to be the solution.

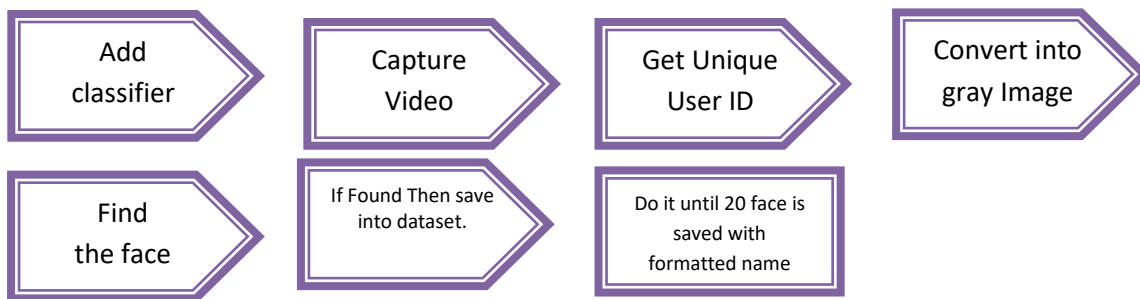


Figure 3.4: Flow Chart of creating dataset.

1. **Train Dataset:** Face detection procedure cannot be completed without training the system according to the dataset. The data set is combined with the Name, ID, and image of the user. The name and ID will be collected from the Android App and the pictures will be collected by the camera itself. Combining all of these together a .yml file will be ready to work. When the file will be created, the camera will detect and recognize face according to the training file of .yml extension.

Now Some of the important topics are described in detail.

- **OS:** OS is a python library that's full abbreviation is Operating System. This python library works to walk through the file of the device. In our project, we have saved the dataset and tanning file different locations. So that we need to find and work with the files. OS module has a lot of built-in methods and objects that can help us to walk through and work with various file of PC
- **PIL:** PIL is a python library that's full abbreviation is Python Image Library. The latest version of PIL is widely known as Pillow. This library is also widely used for python programing to do image processing activities. This library has a lot of built-in methods and object that can make easier for programmers to edit, manipulate a control, change the images in various ways. The library is capable to work in Windows, Linux, macOS perfectly. And we will use it in our Linux OS of Raspberry Pi.
- **LBPH Face Recognizer:** The LBPH Face Recognizer is a mostly used face detection algorithm. Here LBPH means Local Binary Patterns Histograms[9]. It works with the numpy array which is a ndarray created from the grayscale image if input image. Now the algorithm is going to be discussed.

Local Binary Pattern (LBP) is a simple but highly efficient operator that labels each image pixel with a pixel edge and considers the result as a binary number.

Now that we know a little more about face recognition and the LBPH, let's go further and see the steps of the algorithm:

1. **Parameters:** the LBPH usually requires 4 parameters to work properly:

- **Radius:** A circular Local Binary Pattern (LBP) uses the Radius and the radius represents around the pixel located in middle. It is set to 1.
- **Neighbors:** The number of sample points that make up a local binary pattern in around. Remember The more sample points you include, the higher the perceived cost. Is usually set to 8.
- **Grid X:** Number of horizontal cells. The more cells, the finer the grid, and the higher the resultant vector dimension. Is usually set to 8.
- **Grid Y:** Number of cells in the vertical direction. The more cells, the finer the grid, and the higher the resultant vector dimension. Is usually set to 8. Don't worry about the parameters right now, you will understand them after reading the next steps.

2. Training the Algorithm: The algorithm needed to be trained. To do, we need to use a dataset which is containing the images of the faces of users. The images need to be of the people we want to recognize. We also need to set an ID (it must be a unique integer number for individual person) for each image, so the algorithm will use this information to recognize an input image and give you an output. Images of the same person must have the same ID. With the training set already constructed, let's see the LBPH computational steps.

3. Applying the LBP operation: The first scheming step for LBPH is to create an intermediate image that better describes the original image by enhancing facial features. To that end, the algorithm uses the idea of a sliding window based on the parameter **radius** and its **neighbor**.

The image below shows this procedure:

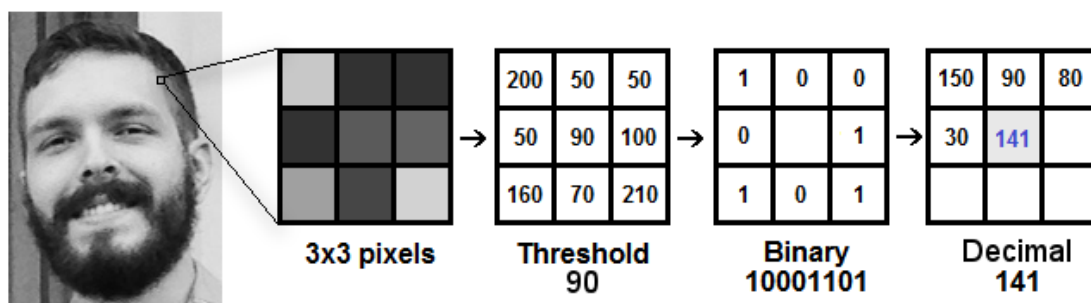


Figure 3.5: Applying LBP Operation

Based on the image above, let's break it into several small steps so we can understand it easily:

- Suppose we have a facial image in grayscale.
- We can get part of this image as a window of 3x3 pixels.
- A 3x3 matrix can help us to represent it. The matrix will contain the intensity of each pixel (0~255).
- The central value is saved in the matrix will be used as the threshold.
- From the 8 neighbors, the values will be used to define the new binary values.
- For each neighbor of the central value (threshold), we must have a neighbor number, we will convert them into a binary value. We customary **1 if the neighboring values equal or higher than the threshold** and **0 if the neighboring values lower than the threshold**.
- Now, the finally created matrix will comprehend only binary values (the central value will be ignored). We will concatenate each binary value from each position of the matrix line by line into a new binary value (e.g. 10100101). Other tactics to concatenate the binary values like clockwise direction can be used, but the final resulted matrix will be the same.
- Next, change this binary value into decimal value and set it to the matrix median. This is truly a pixel in the inventive image.
- Finally, in this LBP method, an original image is created that better represents the features of the inventive image.
- **Note:** The method we are using named LBP (Local Binary Pattern) method has been protracted to use a large number of radii and neighbors and usually it is called circular LBP.

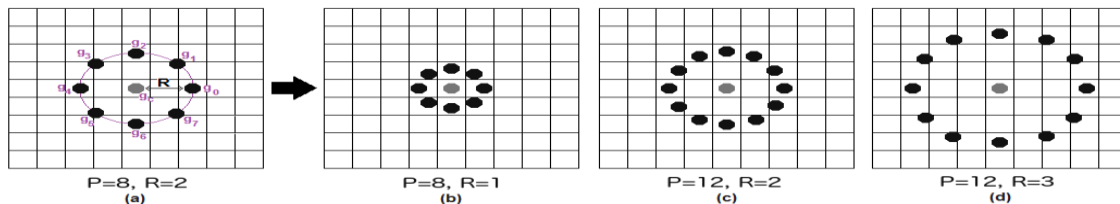


Figure 3.6: Visualization of circular LBP in large scale

Using **bilinear interpolation**, it can be done. If some data point is among the pixels, it uses the values from the 4 closest pixels (2x2). To guess the value of the new data point, this technique will help much.

4. Mining the Histograms: In this case, we will be using the image generated in the final step, we can split the image into several grids using the grid X and grid Y parameters, like the image given below.

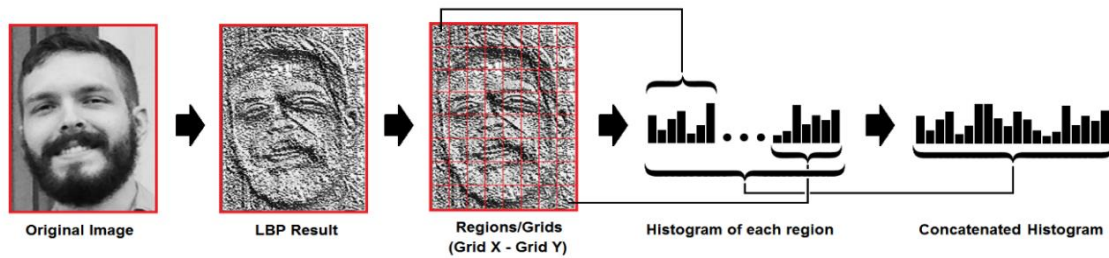


Figure 3.7: Extracting Histograms

Based on the image above, we can extract the histogram of each region as follows:

- Because there is a grayscale image, each histogram (from each grid) represents only the presence of the intensity of each pixel in positions 256 (0-255).
- Next, you'll need to combine each histogram to create a new larger histogram. Given an 8x8 grid, the final histogram has $8 \times 8 \times 256 = 16,384$ positions. The final histogram signifies the key image topographies of the image. The LBPH algorithm basically works like this.

5. Performing the face recognition: This step is most significant because the algorithm is now trained. Each histogram formed is mainly worked to signify each image in the training dataset. Therefore, given an input image, steps will be taken to create a histogram that displays the final image again.

- Therefore, just compare the two histograms to find an image that matches the input image and return the image with the closest histogram.
- Hist Compare histograms using different methods such as Euclidean distance, chi-square, and absolute value (distance calculation between two histograms). In this example, the Euclidean distance (very well known) is based on the following formula:

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

- So, the final algorithm output is the ID from the image with the neighboring histogram. The algorithm also should return the calculated distance, which can be used as a ‘**confidence**’ dimension.
- We also can then use a threshold and the ‘confidence’ to mechanically approximation if the algorithm has correctly recognized the image. We can undertake that the algorithm has successfully recognized only if the confidence is a lower number than the threshold demarcated.

Creating .yaml file: Completing the LBPH Face Recognizer Algorithm the face type data be returned by cv2.face.LBPHFaceRecognizer_create() method. This face type value is added with ID and combines to create tanning file with .yaml extension.

2. **Face Recognition:** Face recognition is a process to understand the name of the person. In this process, the camera is turned on and start capturing video. Video is basically an infinite loop of still frames. The whole LBPH Face Recognition process and matching with .yaml keep going on for every frame. Whenever the identifiable face be matched.

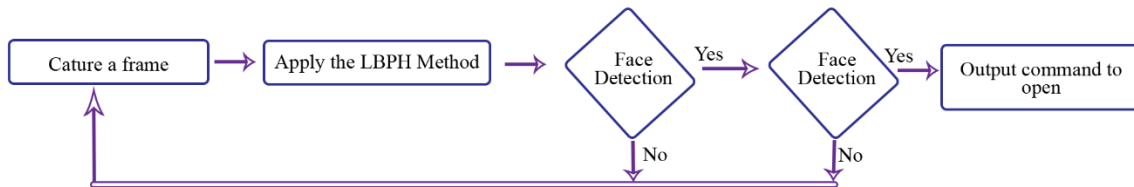


Figure 3.8: Flow Chart of whole face detection process

Difference between face detection and recognition is face detection means understanding that there is a human face in that frame. On the other hand, the term face recognition means to understand that whose face is it. Here a dataset is a must. Because without a dataset, the system can not know the name of the face. The dataset should have enough pictures of a specific face with a name. The .yaml file be created with face and ID for the individual which is unique.

3. **Send the Data Through Bluetooth to Raspberry Pi:**

Whenever we need to registrar the user with the lock, we must use the android app. The app will provide a completely user-friendly interface to input user data. The data will be loaded to the lock and the lock is built with raspberry pi. Officially, Raspberry Pi is implemented with Bluetooth. So, using Bluetooth we can be connected with android and the lock.

Throughout Bluetooth, we will be able to send user information to the raspberry pi. Then, the lock will be able to identify the face to be unlocked.

Bluetooth:

Bluetooth is a Small-range communication system that allows devices like mobile computers, phones, and peripherals to transmit data or voice wirelessly over a small distance.

Bluetooth uses something called frequency-hopping spread spectrum, which is a radio technology. frequency-hopping spread spectrum (FHSS) is a technique of transmitting radio signals by hopping or changing a carrier among many frequency channels, using some sequence that is known to both transmitter and receiver.

4. Unlock the Lock:

Whenever the lock will recognize any face from the database, the lock will output an One “1” which will provide a current to a servo motor. The servo motor will rotate a spin, the spin will help to unlock the lock.

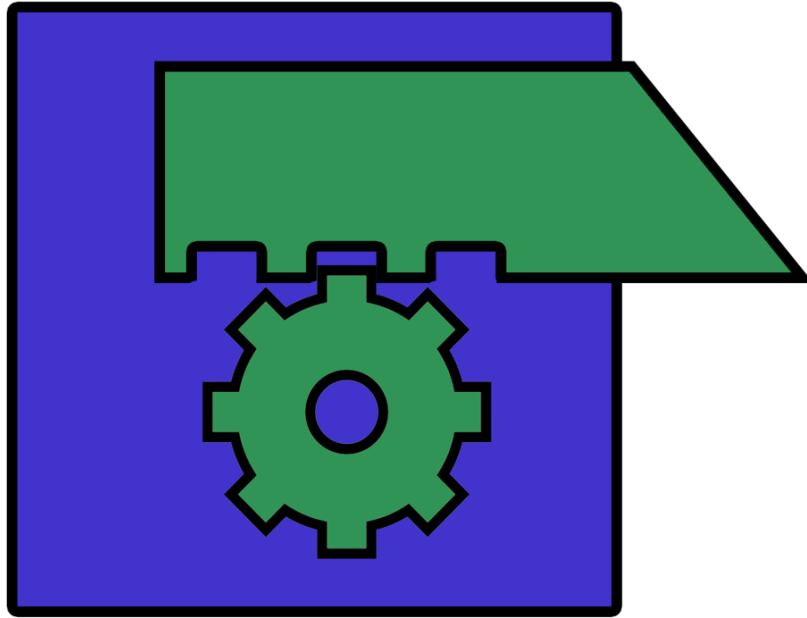


Figure: 3.9: The procedure of being unlocked

CHAPTER 4

DESIGN SPECIFICATION AND RESULT

4.1 Front-end design

The user will register his/her information by using an android application. This user id registration needs to the precise identification of the user. Don't use the same id to registration.

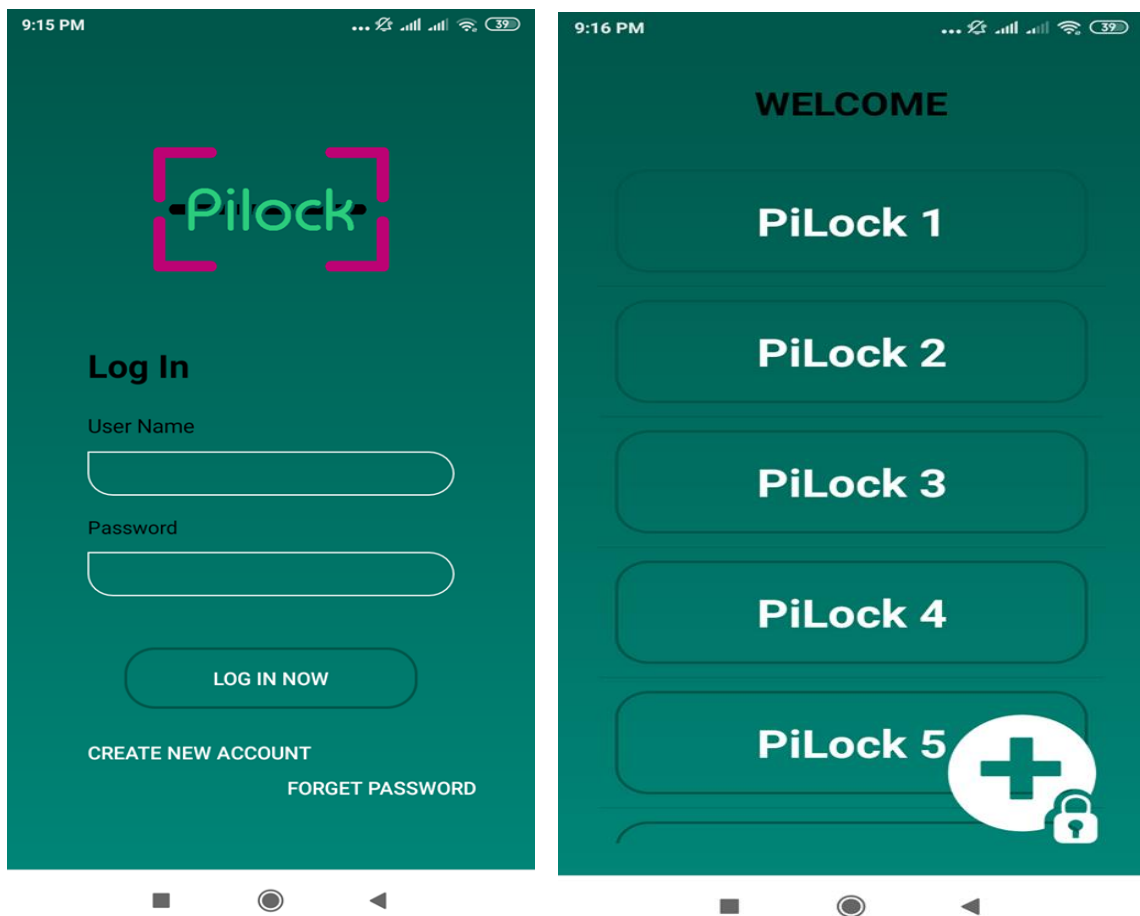


Figure 4.1: Android Interface to registrar new user

4.2 Back-End design

When users register for id our system will take an image from the video then this image will be detected by the system and stored with trained data sets.

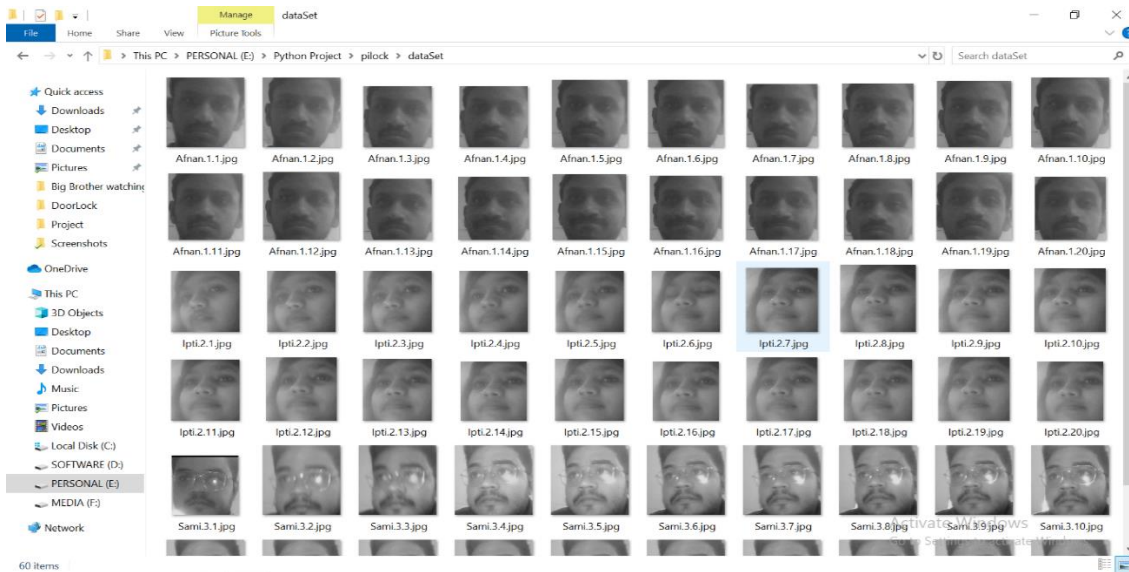


Figure 4.2: Dataset

4.3 Experimental Result

This system was formed by using Pycharm on windows 10 operating system. While the system will be used in raspberry pi it will be installed in Linux OS. In figure images of faces are detected successfully.

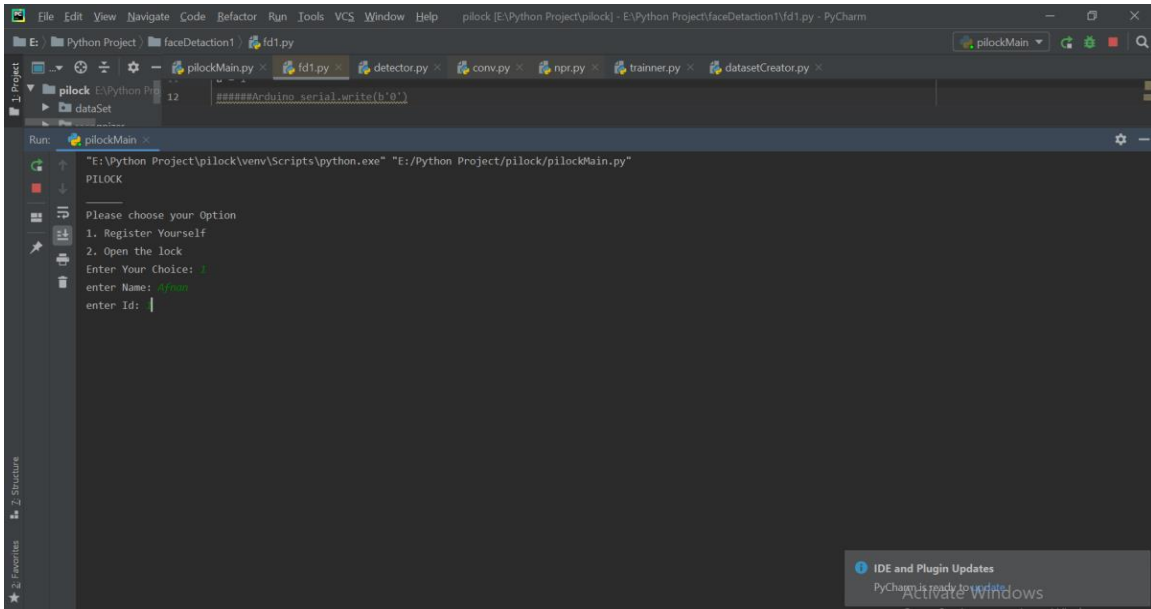


Figure 4.3: New User Registration

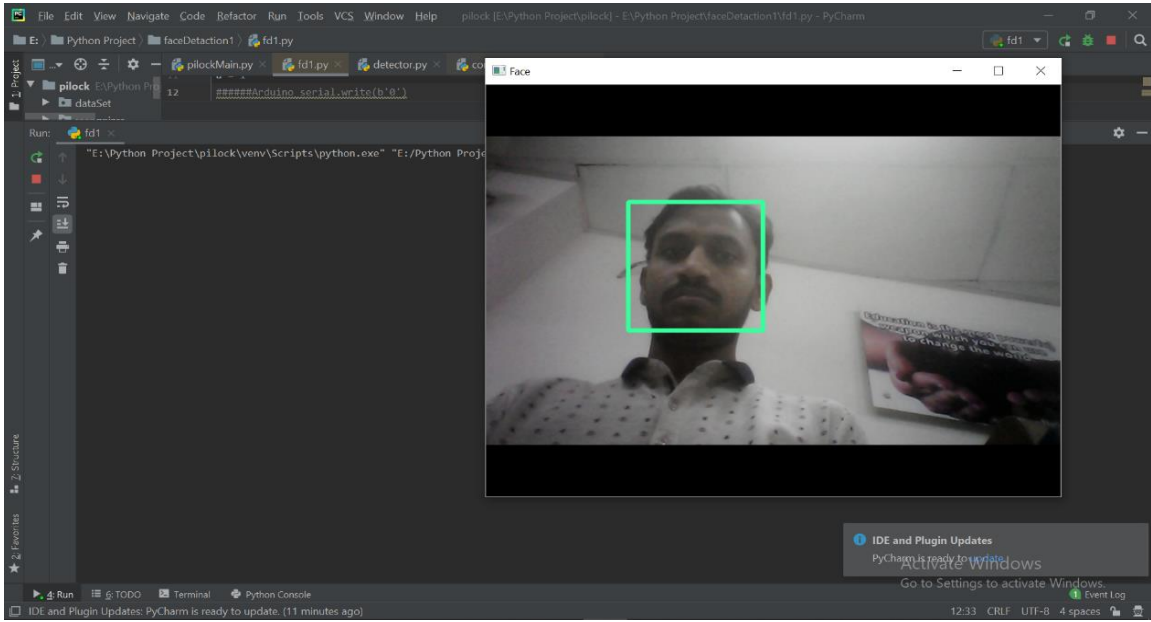


Figure 4.4: Detect ROI to get face data

The process of face detection is formed by using video recording. In the next step detected area of the image will be cropped and stored in a database as a dataset with the given used id by the user.



Figure 4.5: Trained Dataset

After that, the next part will be to identifying the face with the registered face. For the face, recognition is compared to two histograms and returns the closest histogram with the registered user id. Then the system brings the Name from the database that is stored in the database with registered user ID and Name.

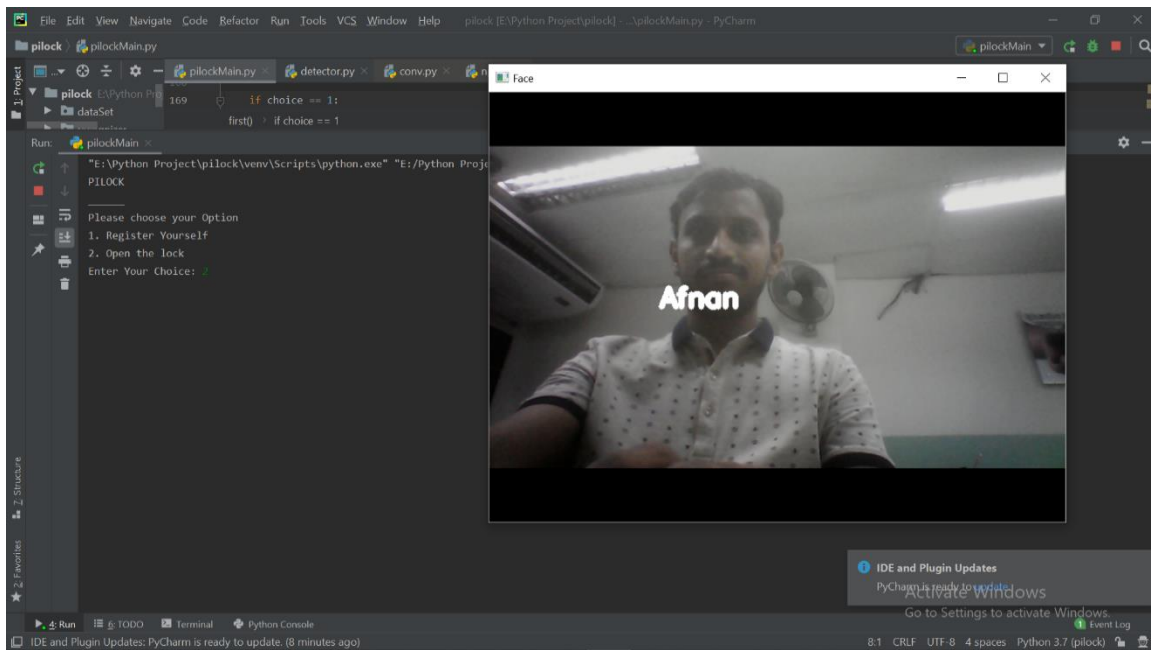


Figure 4.6: Recognized the face to unlock

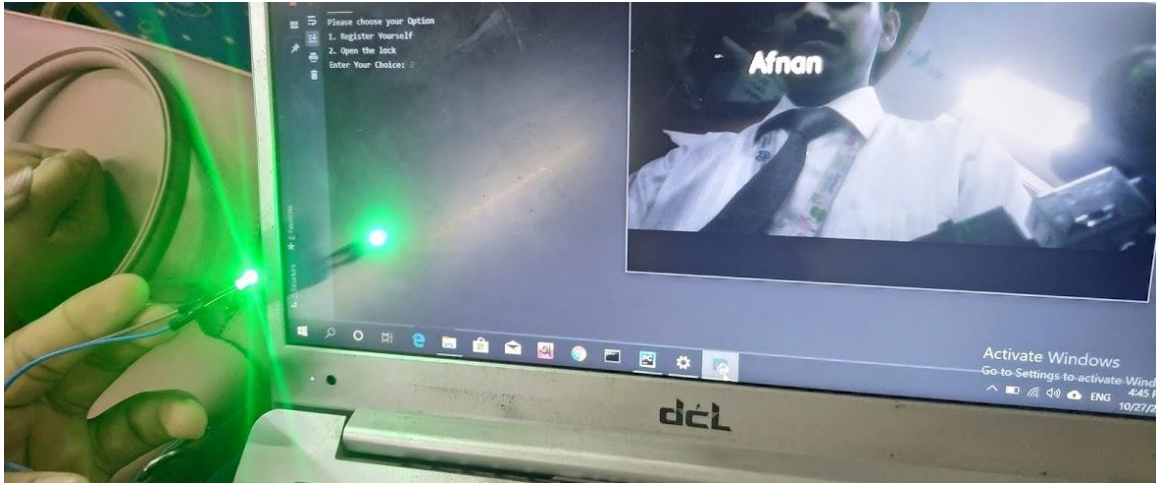


Figure 4.7: System worked based on face

4.4 Implementation Requirement

When we started to execute this system, it was a very new task to us. It was hard to make up these challenges. For execute this system we need both hardware and software tools. For execution, a camera needs to detection and identify and need a computer to implement the code.

- a. Python
- b. OpenCV
- c. Android Studio
- d. Raspberry pi
- e. Bluetooth Module

Besides, it needs some soft skills also.

CHAPTER 5

IMPLEMENTATION AND TESTING

5.1 User Registration with Face Detection:

For face recognition, at first, a user has to register his/her name with an image by using a camera. After input, the data by the user the system will detect the face in front of the camera and store the image with trained data set.

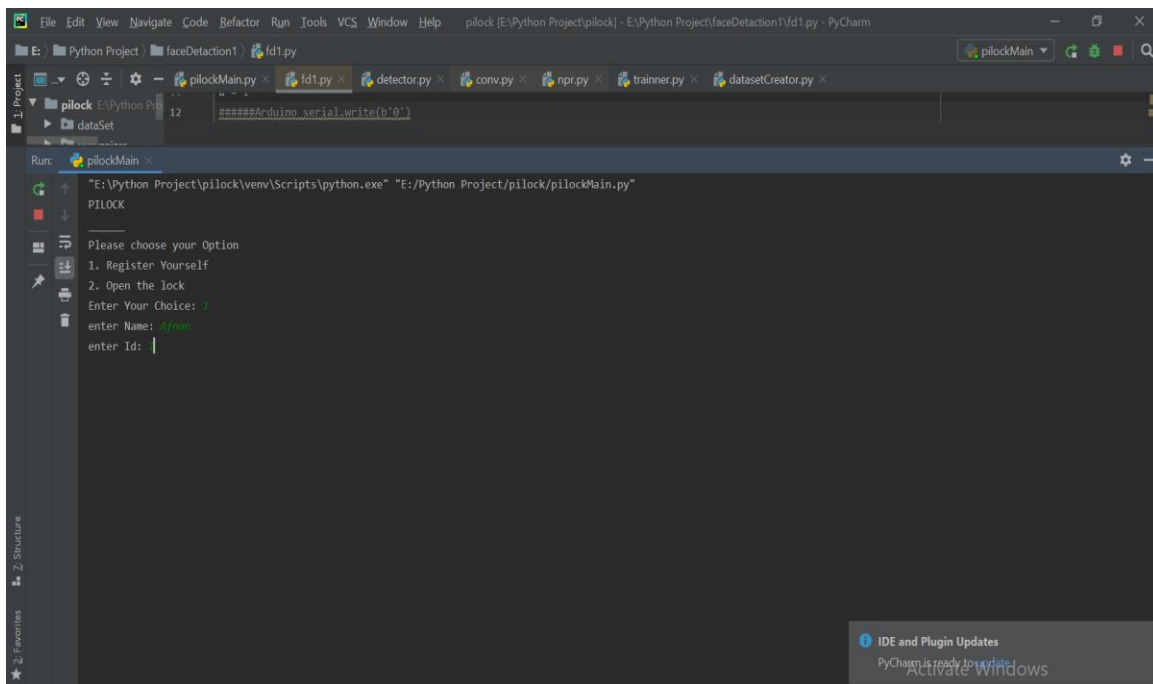


Figure 5.1: User registration Interface

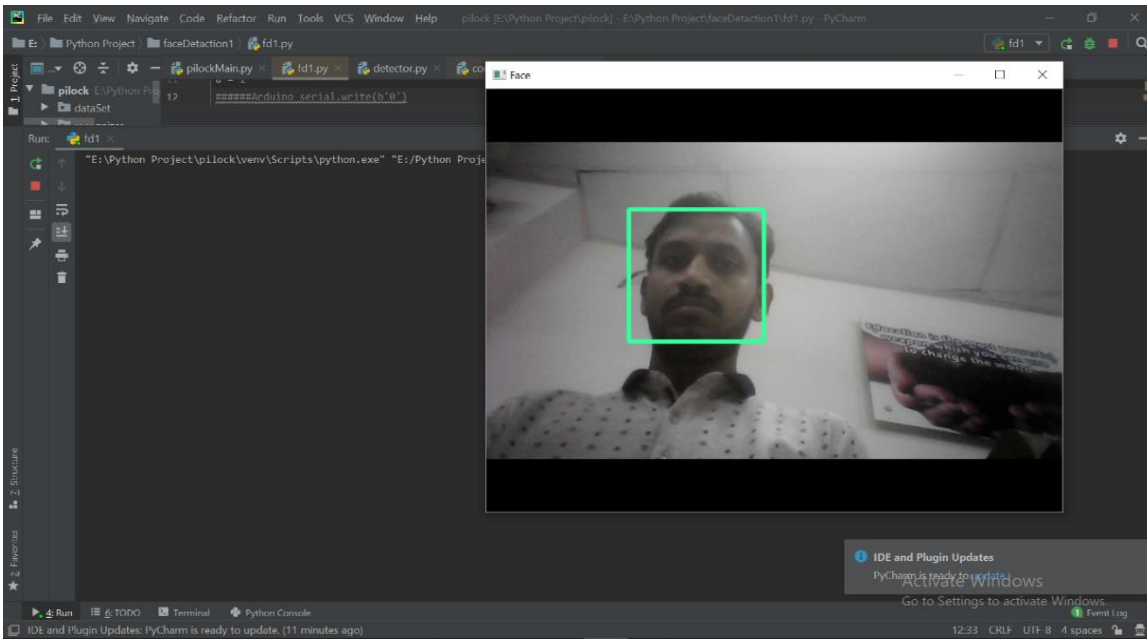


Figure 5.2: Detected face to unlock

5.2 Training Data:

The system creates a .yaml file. This file holds all the matrix of each face. These matrix are used for identifying authorized persons' face.

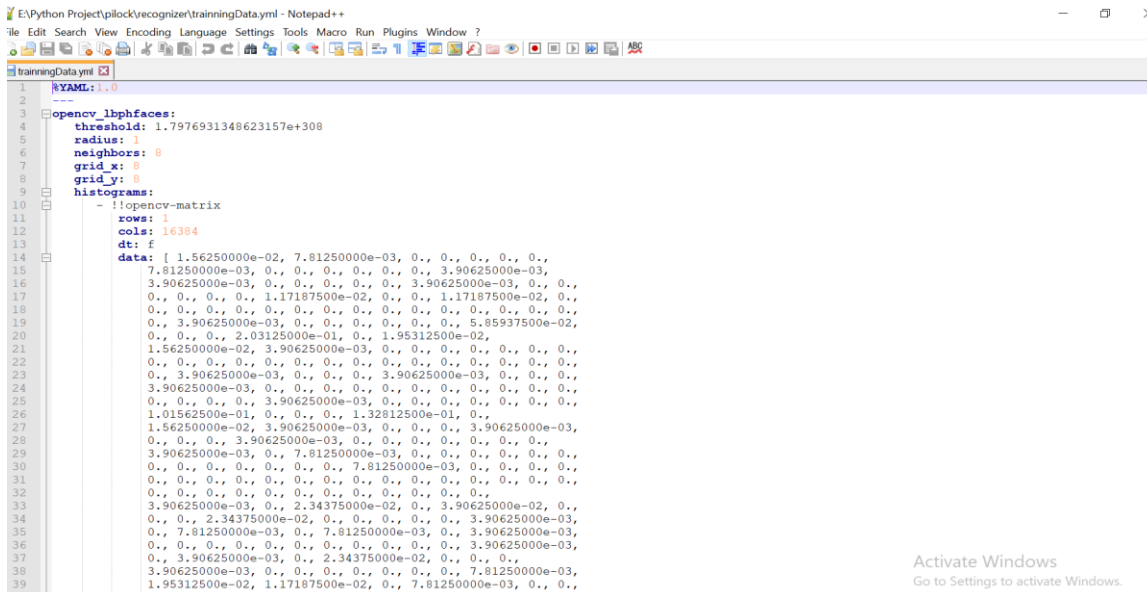


Figure 5.3: The matrix of the .yaml file after training

5.3 Testing Implementation

The testing part was an amazing experience for us. In this period we registered 50 images for each person for finding the laxness of our system. But our system gives us an excellent output. It can detect images successfully and after that, the identification result was also satisfactory.

It can

- a. Detect Face
- b. Identify face
- c. Show details of a person who is registered
- d. Add new user information by registration
- e. Unlock lock system

5.3.1 Acceptance Test

We didn't find any system which is working exactly the same as ours. Besides that, we used some face detection and recognition system but our system works much better in case of face detection and face recognition.

5.4 Test result and report

All of the tested results was up to the point. We should need to focus on the light of the images.

5.4.1 System Testing

The system testing result was very satisfactory. Due to the android application, a user can easily access this system and acceptance will be positive.

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

6.1 Conclusion

This project is ended up successfully with identifying, detecting human face and also the ability to open or close the door by an android application, python programming and also a Raspberry pi. The most interesting part of this project is that it will work even without the internet.

- Develop a mobile phone application that can be installed on any android smartphone.
- Design face identifying and detecting GUI in an android phone
- Design of a hardware Bluetooth module system.

The lock security system was designed to improve the conventional lock system to secure our data, home, business organization, school, office and any other sophisticated system. To identify and recognizing face, a user registered his/her name with an image. The door was opened automatically if the user's image matches the registered images. By following this process, this system prevents any unauthorized person to access. If any robbery issues occurred in any organization or institution this system is successfully able to show the evidence to the security management system.

6.2 Future Scopes

We have developed a system that can work successfully. But we have decided in the future, we will develop this system more efficient by following this step.

- If anyone tries to break the lock, the system will capture the images and store them.
- System will store the information of every accesses.
- Retina Scanning will make the lock more secure.

References

- [1] A. O. Onyan and K. O. Enalume, "Property Security Using a Biometric Based Door Lock System," 2018.
- [2] A. V Patil, Ch. Patgar, S. Prakash, and S. A. Kumar J, "Android Based Smart Door Locking System."
- [3] A. R. S. #1, B. R. #2, K. K. #3, K. S. #4, S. Venkatasubramanian, and B. E. Student, "Optimized Door Locking and Unlocking Using IoT for Physically Challenged People," *Int. J. Innov. Res. Comput. Commun. Eng.* (An ISO, vol. 3297, 2007.
- [4] A. David Odu, M. Chinaza Alice, and O. J. Odinya, "Low Cost Removable (Plug-In) Electronic Password-Based Door Lock," *Am. J. Eng. Res.*, no. 6, pp. 146–151, 2017.
- [5] J. Venukumar, "Arduino Based Door Access Control," *Int. J. Res. Advent Technol.*, vol. 4, no. 8, 2016.
- [6] L. Kamelia, A. Noorhassan, M. Sanjaya, and E. Mulyana, "DOOR-AUTOMATION SYSTEM USING BLUETOOTH-BASED ANDROID FOR MOBILE PHONE," vol. 9, no. 10, 2014.
- [7] S. Emami and V. P. Suci, "Facial Recognition using OpenCV," *J. Mobile, Embed. Distrib. Syst.*, vol. 4, no. 1, pp. 38–43, 2012.
- [8] R. Okuta, Y. Unno, D. Nishino, S. Hido, and C. Loomis, "CuPy: A NumPy-Compatible Library for NVIDIA GPU Calculations," *Work. ML Syst. NIPS 2017*, no. Nips, pp. 1–7, 2017.
- [9] A. Ahmed, J. Guo, F. Ali, F. Deeba, and A. Ahmed, "LBPH based improved face recognition at low resolution," 2018 *Int. Conf. Artif. Intell. Big Data, ICAIBD 2018*, no. May, pp. 144–147, 2018.
- [10] Varun Garg, Kritika Garg, Face Recognition Using Haar Cascade Classifier, *Journal of Emerging Technologies and Innovative Research (JETIR)* , December 2016, Volume 3, Issue 12.
- [11] heng Xiang, Hengliang Tan, Wienling Ye. The excellent properties of dense gird-based HOG features on face recognition compare to gabor and LBP, 2018 volume issue 99.
- [12] Crime rate of Peoples Republic Of Bangladesh is given in the following Link <https://www.numbeo.com/crime/country_result.jsp?country=Bangladesh> accessed on 6-12-2019 at 6:20 p.m.

Test

ORIGINALITY REPORT

24%
SIMILARITY INDEX

17%
INTERNET SOURCES

7%
PUBLICATIONS

23%
STUDENT PAPERS

PRIMARY SOURCES

1	towardsdatascience.com Internet Source	5%
2	Submitted to Daffodil International University Student Paper	2%
3	Submitted to Siddaganga Institute of Technology Student Paper	2%
4	Submitted to ABES Engineering College Student Paper	1%
5	docs.opencv.org Internet Source	1%
6	numpy.org Internet Source	1%
7	dspace.bracu.ac.bd Internet Source	1%
8	Submitted to Chittagong University of Engineering and Technology Student Paper	1%