MINI PROJECT REPORT
(2020)

Securing data using Face Recognition System

Institute of Engineering & Technology

Team Members

Tanuj Johal
(171500356)
Anushka Soni
(171500059)

Supervised By
Mr. Neeraj Khanna
Technical Trainer
Department of Computer Engineering & Applications

Department of computer Engineering and Applications
GLA University, Mathura
Chaumuha, Mathura – 281406

CERTIFICATE

Certified that this is a bonafide record of the project work titled

SECURING DATA USING FACE RECOGNITION SYSTEM

done by

Tanuj Johal (171500356)

Anushka Soni (171500059)

of V semester Computer Science & Engineering in the year 2019

in partial fulfillment of the requirements for the award of Degree of

Bachelor of Technology

in Computer Science & Engineering of GLA University, Mathura.

Mr. Neeraj Khanna
Technical Trainer
(Project Mentor)

ACKNOWLEDGEMENT

It is our pleasure to be indebted to various people, who directly or indirectly contributed in the development of this work and who influenced our thinking, behavior and acts during the course of study.

We express our sincere gratitude to **GLA University** for providing us an opportunity to undergo this Project as the part of the curriculum. We are thankful to **Mr. Neeraj Khanna** for his support, cooperation, and motivation provided to us during the training for constant inspiration, presence and blessings.

Lastly, we would like to thank each of the teammates with whom we shared our day-to-day experience and received lots of suggestions that improved our quality of work.

We perceive as this opportunity as a big milestone in my career development. We will strive to use gained skills and knowledge in the best possible way.

Names and Signature

ANUSHKA SONI

U. Roll no.:171500059

TANUJ JOHAL

U. Roll no.:171500356

ABSTRACT

The face is one of the easiest ways to distinguish the individual identity of each other. Face recognition is a personal identification system that uses personal characteristics of a person to identify the person's identity. Human face recognition procedure basically consists of two phases, namely face detection, where this process takes place very rapidly in humans, except under conditions where the object is located at a short distance away, the next is the introduction, which recognize a face as individuals. Stage is then replicated and developed as a model for facial image recognition (face recognition) is one of the much-studied biometrics technology and developed by experts. There are two kinds of methods that are currently popular in developed face recognition pattern namely, Eigenface method and Fisherface method. Facial image recognition Eigenface method is based on the reduction of face-dimensional space using Principal Component Analysis (PCA) for facial features. The main purpose of the use of PCA on face recognition using Eigen faceswas formed (face space) by finding the eigenvector corresponding to the largest eigenvalue of the face image. The area of this project is securing data using face recognition system. As the case of data breaches are increasing day by day, the best way to secure the data is by locking i with entity wich cant be easily replicate decreasing the chance of data leaking. In this project a user can lock a folder by making his/her face as the paswword for the folder.

CONTENTS

1. INTRODUCTION

1.1 Project Description

The project is based on the face recognition and unlocking system. In this project we are going to capture the picture of the client and then program will unlock the folder using the client photo as a password. In this project the client has an advantage to secure his/her folder using the face as a password which will be used to unlock the folder. We will be having multiple modules to encounter this task. There will be the frontend of the folder unlocking system which have two option in the starting. The first one is Register in this option the program will save the 100 photo of the client in the database and map it with his/her name that client will give at the time of capturing the photo. After pushing the data into the Database our model will be ready to recognize the particular photo and will tell you the name of the client according to the photo. Now, After setting up all the data into the database. You will have the option to setup the password on the folder. At the time of setting up the password on the folder. You will give the your face as an input and it will generate the database of the your photo accordingly. When you will try to unlock the folder you need to show your face it will recognize your face using the machine the algorithm CNN and after validating it with our database it will unlock your folder. The another thing that we have included in this project is that beside doing locking and unlocking you can also recognize your face. For that we have another option as well that is recognize which will try to recognize your picture if your picture exist in the database and it also be mapped according to your name. So, like Face recognition project it will also tell client his/her name as well.

1.2 About Face Recognition and Face Detection System

Face recognition is the task of identifying an already detected object as a known or unknown face.Often the problem of face recognition is confused with the problem of face detectionFaceRecognition on the other hand is to decide if the "face" is someone known, or unknown, using for this purpose a database of faces in order to validate this input face.

1.2.1 Face Recognition

DIFFERENT APPROACHES OF FACE RECOGNITION:

There are two predominant approaches to the face recognition problem: **Geometric** (feature based) and **photometric** (view based). As researcher interest in face recognition continued, many different algorithms were developed, three of which have been well studied in face recognition literature.

Recognition algorithms can be divided into two main approaches:

1. **Geometric :** Is based on geometrical relationship between facial landmarks, or in other words the spatial configuration of facial features. That means that the main geometrical features of the face such as the eyes, nose and mouth are first located and then faces are classified on the basis of various geometrical distances and angles between features.

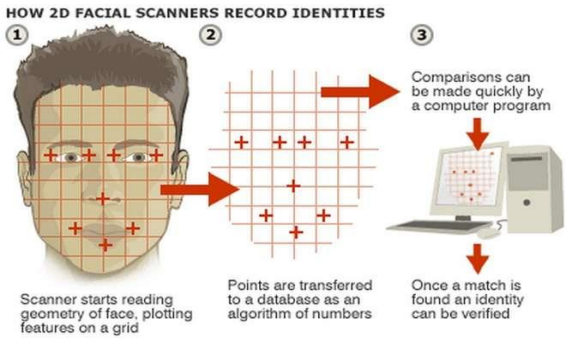

Fig 1.1 Geometric Facial Recognition

2. Photometric stereo : Used to recover the shape of an object from a number of images taken under different lighting conditions. The shape of the recovered object is defined by a gradient map, which is made up of an array of surface normals (Zhao and Chellappa, 2006) (Figure 1.2.2)

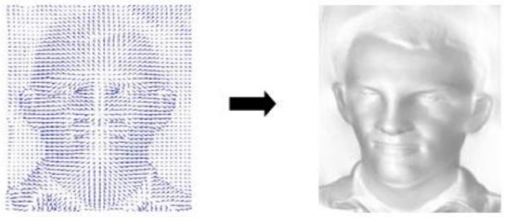

Fig 1.2 Photometric stereo image

Popular recognition algorithms include:

1. Principal Component Analysis using Eigenfaces, (PCA)
2. Linear Discriminate Analysis,
3. Elastic Bunch Graph Matching using the Fisherface algorithm

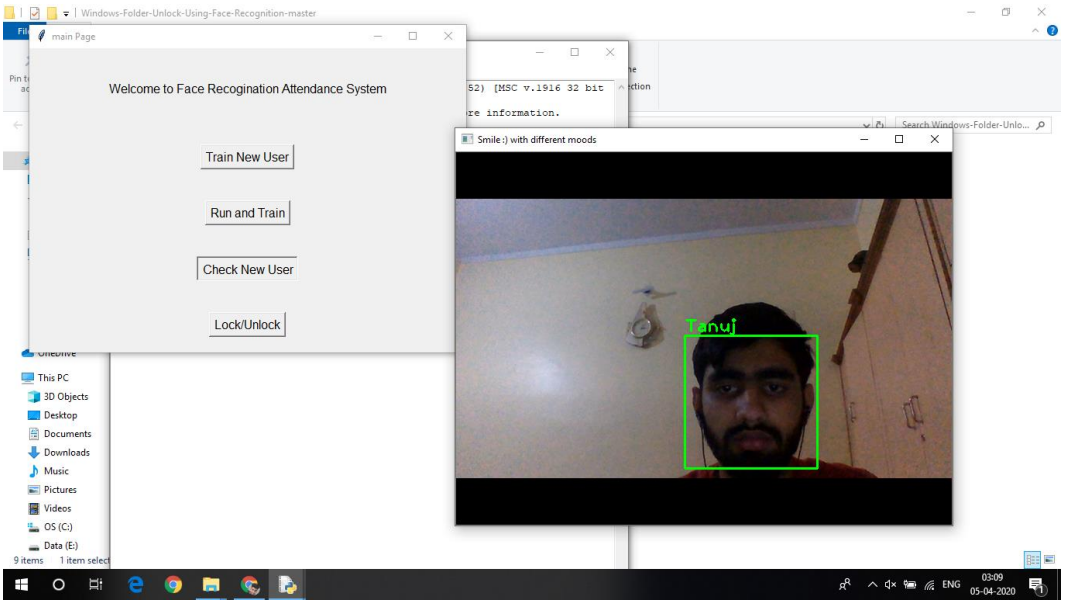

Fig 1.4 (Screenshot of Project doing Face Recognition)

1.2.2 Face Detection

Face detection involves separating image windows into two classes; one containing faces (tarning the background (clutter). It is difficult because although commonalities exist between faces, they can vary considerably in terms of age, skin colour and facial expression. The problem is further complicated by differing lighting conditions, image qualities and geometries, as well as the possibility of partial occlusion and disguise. An ideal face detector would therefore be able to detect the presence of any face under any set of lighting conditions, upon any background. The face detection task can be broken down into two steps. The first step is a classification task that takes some arbitrary image as input and outputs a binary value of yes or no, indicating whether there are any faces present in the image. The second step is the face localization task that aims to take an image as input and output the location of any face or faces within that image as some bounding box with (x, y, width, height).

Theory or face detection classifiers

A computer program that decides whether an image is a positive image (face image) or negative image (non-face image) is called a classifier. A classifier is trained on hundreds of thousands of face and non-face images to learn how to classify a new image correctly.

OpenCV provides us with two pre-trained and ready to be used for face detection classifiers:

1. Haar Classifier
2. LBP Classifier

Both of these classifiers process images in gray scales, basically because we don't need color information to decide if a picture has a face or not. As these are pre-trained in OpenCV, their learned knowledge files also come bundled with OpenCV opencv/data/.

To run a classifier, we need to load the knowledge files first, as if it had no knowledge.

Each file starts with the name of the classifier it belongs to. For example, a Haar cascade classifier starts off as haarcascade_frontalface_alt.xml.

These are the two types of classifiers we will be using to analyze Casper.

a) HAAR Classifier

The Haar Classifier is a machine learning based approach, an algorithm created by Paul Viola and Michael Jones; which are trained from many many positive images (with faces) and negatives images (without faces).

It starts by extracting Haar features from each image as shown by the windows below:

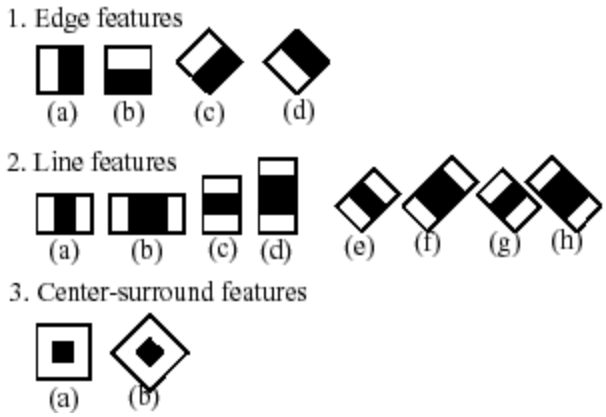

Fig 1.3 Extracting HAAR feature

Each window is placed on the picture to calculate a single feature. This feature is a single value obtained by subtracting the sum of pixels under the white part of the window from the sum of the pixels under the black part of the window.

In the end, the algorithm considers the fact that generally: most of the region in an image is a non-face region. Considering this, it’s a better idea to have a simple method to check if a window is a non-face region, and if it's not, discard it right away and don’t process it again. So we can focus mostly on the area where a face is.

b) LBP Cascade Classifier

As any other classifier, the Local Binary Patterns, or LBP in short, also needs to be trained on hundreds of images. LBP is a visual/texture descriptor, and our faces are also composed of micro visual patterns. LBP features are extracted to form a feature vector that classifies a face from a non-face

Each training image is divided into some blocks as shown in the picture below.

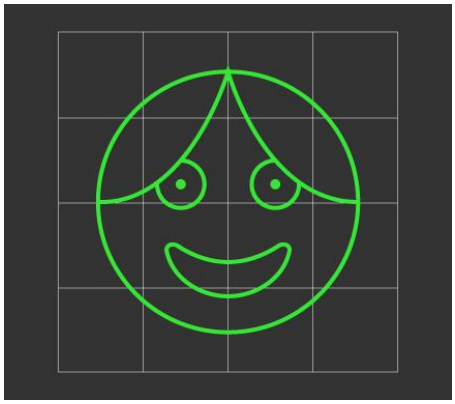

Fig 1.5 Division of training image into blocks

LBP Windows (disregard the first grader drawing)

For each block, LBP looks at 9 pixels (3×3 window) at a time, and with a particular interest in the pixel located in the center of the window.Then, it compares the central pixel value with every neighbor's pixel value under the 3×3 window. For each neighbor pixel that is greater than or equal to the center pixel, it sets its value to 1, and for the others, it sets them to 0.

After that, it reads the updated pixel values (which can be either 0 or 1) in a clockwise order and forms a binary number. Next, it converts the binary number into a decimal number, and that decimal number is the new value of the center pixel. We do this for every pixel in a block.
Each OpenCV face detection classifier has its pros and cons, but the major differences are in accuracy and speed.

So, in case more accurate detections are required, Haar classifier is the way to go. This is more suitable in technology such as security systems or high-end stalking.

But the LBP classifier is faster, therefore, should be used in mobile applications or embedded systems.

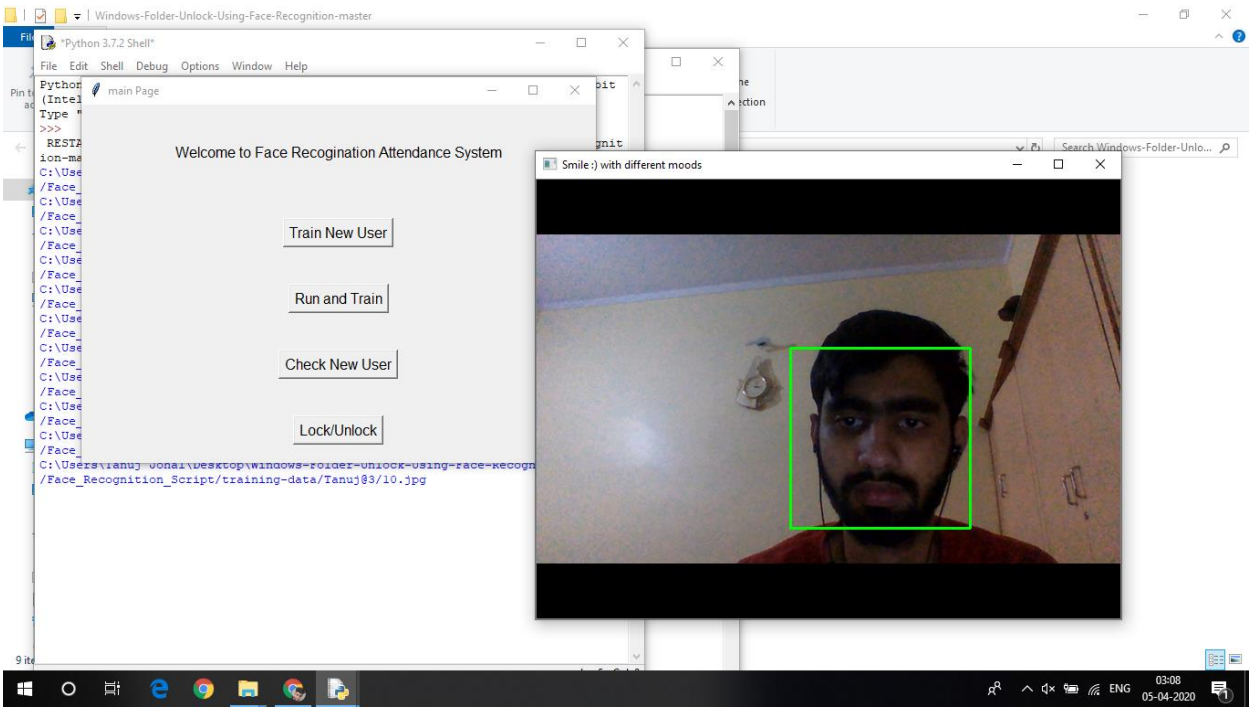

Fig 1.6 (Screenshot of Project doing Face Detection)

1.3 About Locking and Unlocking Folder

1.3.1 PowerShell

Windows PowerShell is a **command-line shell** and **scripting language** designed especially for system administration. Its analogue in Linux is called as Bash Scripting. Built on the .NET Framework, Windows PowerShell helps IT professionals to control and automate the administration of the Windows operating system and applications that run on Windows Server environment.

Windows PowerShell commands, called **cmdlets**, let you manage the computers from the command line. Windows PowerShell providers let you access data stores, such as the Registry and Certificate Store, as easily as you access the file system.

In addition, Windows PowerShell has a rich expression parser and a fully developed scripting language. So, in simple words you can complete all the tasks that you do with GUI and much more.

a) PowerShell ISE

The Windows PowerShell **Integrated Scripting Environment** (ISE) is a host application for Windows PowerShell. In Windows PowerShell ISE, you can run commands and write, test, and debug scripts in a single Windows-based graphic user interface with multiline editing, tab completion, syntax colouring, selective execution, context-sensitive help, and support for right-to-left languages.

You can use menu items and keyboard shortcuts to perform many of the same tasks that you would perform in the Windows PowerShell console. For example, when you debug a script in the Windows PowerShell ISE, to set a line breakpoint in a script, right-click the line of code, and then click **Toggle Breakpoint**.

1.3.2 Batch File

A batch file is a text file that contains a sequence of commands for a computer operating system. It's called a batch file because it batches (bundles or packages) into a single file a set of commands that would otherwise have to be presented to the system interactively from a keyboard one at a time. A batch file is usually created for command sequences for which a user has a repeated need. Commonly needed batch files are often delivered as part of an operating system. You initiate the sequence of commands in the batch file by simply entering the name of the batch file on a command line.

In the Disk Operating System (DOS), a batch file has the file name extension ".BAT". (The best known DOS batch file is the AUTOEXEC.BAT file that initializes DOS when you start the system.) In UNIX-based operating systems, a batch file is called a shell script. In IBM's mainframe VM operating systems, it's called an EXEC.

We have created a .bat file for locking and unlocking the folder when we will click to the locker.bat file script will work and then it will read the script that we have written in there that script is given below. We have use subprocesses to open the .bat file.

a) Locker.bat Script

```
@ECHO OFF
title Folder Locker
if EXIST "Control Panel.{21EC2020-3AEA-1069-A2DD-08002B30309D}" goto UNLOCK
if NOT EXIST Locker goto MDLOCKER
:CONFIRM
echo Are you sure u want to Lock the folder(Y/N)
set/p "cho=>"
if %cho%==Y goto LOCK
if %cho%==y goto LOCK
if %cho%==n goto END
if %cho%==N goto END
echo Invalid choice.
goto CONFIRM
:LOCK
ren Locker "Control Panel.{21EC2020-3AEA-1069-A2DD-08002B30309D}"
attrib +h +s "Control Panel.{21EC2020-3AEA-1069-A2DD-08002B30309D}"
echo Folder locked
```

```
goto End
:UNLOCK
echo Enter password to Unlock folder
SET PASS=tanuj
python "%CD%\Face_Recognition_Script\Face recognition.py" %PASS%> Output
SET /p MYVAR=<Output
if %MYVAR%==%PASS% goto DONE
for /f "tokens=*" %%a in (Output) do (
  if %%a==%PASS% goto DONE
)
goto FAIL
:DONE
attrib -h -s "Control Panel.{21EC2020-3AEA-1069-A2DD-08002B30309D}"
ren "Control Panel.{21EC2020-3AEA-1069-A2DD-08002B30309D}" Locker
echo Folder Unlocked successfully
pause
goto End
:FAIL
pause
echo Invalid password
goto End
:MDLOCKER
md Locker
echo Locker created successfully
goto End
:End
```

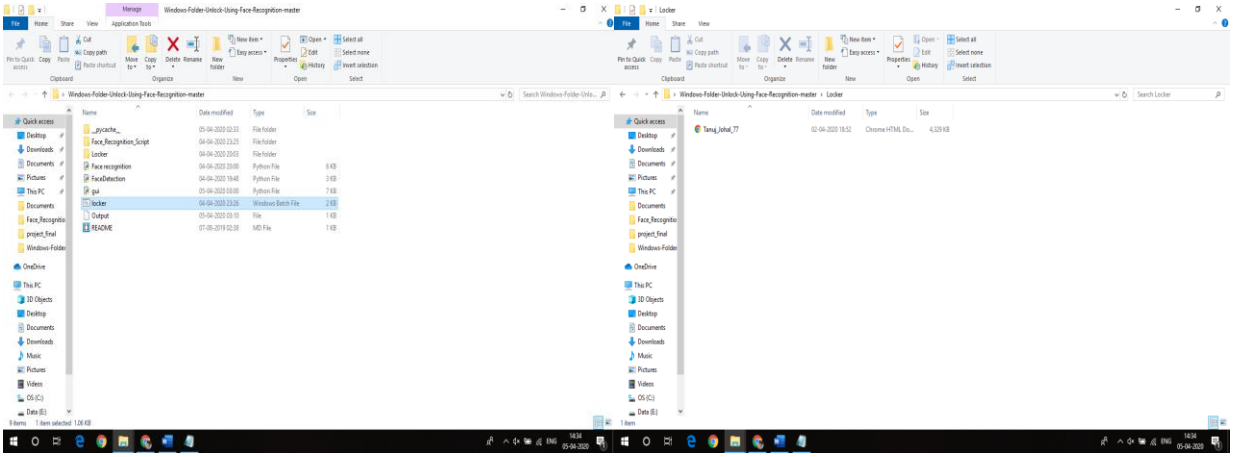

Fig 1.7 (Screenshot of Locker Folder)

After triggered the locker.bat script Locker folder will get disappeared.

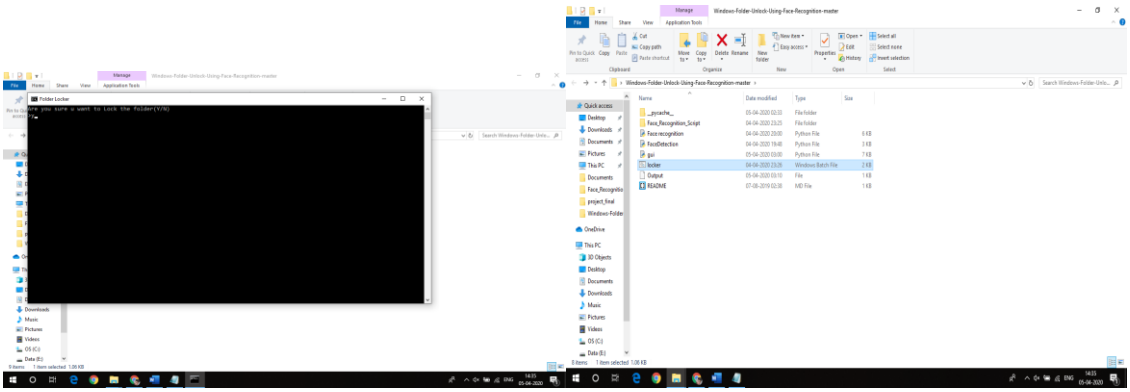

Fig 1.8 (Screenshot of disappearing Locker folder after triggered shell script locker.bat)

2. Graphical User Interface

The graphical user interface, developed in the late 1970s by the Xerox Palo Alto research laboratory and deployed commercially in Apple's Macintosh and Microsoft's Windows operating systems, was designed as a response to the problem of inefficient usability in early, text-based command-line interfaces for the average user.

Graphical user interfaces would become the standard of user-centred design in software application programming, providing users the capability to intuitively operate computers and other electronic devices through the direct manipulation of graphical icons such as buttons, scroll bars, windows, tabs, menus, cursors, and the mouse pointing device. Many modern graphical user interfaces feature touchscreen and voice-command interaction capabilities.

2.1 GUI in Python:

Python provides various options for developing graphical user interfaces (GUIs). Most important are listed below.

- **Tkinter** − Tkinter is the Python interface to the Tk GUI toolkit shipped with Python. We would look this option in this chapter.
- **wxPython** − This is an open-source Python interface for wxWindows http://wxpython.org.
- **JPython** − JPython is a Python port for Java which gives Python scripts seamless access to Java class libraries on the local machine http://www.jython.org.

There are many other interfaces available, which you can find them on the net.

2.2 Tkinter Programming:

Tkinter is the standard GUI library for Python. Python when combined with Tkinter provides a fast and easy way to create GUI applications. Tkinter provides a powerful object-oriented interface to the Tk GUI toolkit.

Creating a GUI application using Tkinter is an easy task. All you need to do is perform the following steps –

- Import the *Tkinter* module.
- Create the GUI application main window.
- Add one or more of the above-mentioned widgets to the GUI application.
- Enter the main event loop to take action against each event triggered by the user.

2.3 WxPython Programming:

wxPython is a Python wrapper for **wxWidgets** (which is written in C++), a popular cross-platform GUI toolkit. Developed by Robin Dunn along with Harri Pasanen, wxPython is implemented as a Python extension module.

Just like wxWidgets, wxPython is also a free software. It can be downloaded from the official website http://wxpython.org. Binaries and source code for many operating system platforms are available for download on this site.

Principal modules in wxPython API include a core module. It consists of **wxObject** class, which is the base for all classes in the API. Control module contains all the widgets used in GUI application development. For example, wx.Button, wx.StaticText (analogous to a label), wx.TextCtrl (editable text control), etc.

wxPython API has GDI (Graphics Device Interface) module. It is a set of classes used for drawing on widgets. Classes like font, color, brush, etc. are a part of it. All the container window classes are defined in Windows module.

Official website of wxPython also hosts Project Phoenix – a new implementation of wxPython for Python 3.*. It focuses on improving speed, maintainability, and extensibility. The project began in 2012 and is still in beta stage.

3. METHODOLOGY

Below are step how the flow of project will go to unlock folder using face recogniton system:

1. Interfacing of camera module to capture live face image.
2. Creating a database of authorized person.
3. Capturing current face, saving it and comparing with database in real-time.
4. Interface relay as output module depend on wether to unlock folder on given key or not.

The project will be done in two phase. In Phase 1 the face will be detected and database of user will be created and stored using face detection system and in Phase 2 the data stord will be retieved and compared with real time frame and will predict the name and will unlock the folsder.

3.1 PHASE 1 :

1. Saving the face portion
2. Take frame
3. Detect face
4. Extract the face
5. Resize the extracted image
6. Save the extracted image

Finally the faces of person will be saved in prove folder with the name of the persons

those faces will then be used to recognize face

3.2 PHASE 2:

1. get faces from training folder
2. compute model
3. get real time frame
4. detect face from reat-time
5. match that face with stored database
6. predict name according to the database stored
7. compare the key of locked folder
8. unlock the folder

4.Testing

Testing is a group of techniques to determine the correctness of the application under the predefined script but, testing cannot find all the defect of application. The main intent of testing is to detect failures of the application so that failures can be discovered and corrected. It does not demonstrate that a product functions properly under all conditions but only that it is not working in some specific conditions.

4.1 Unit Testing:

Unit testing involves the testing of each unit or individual component of the software application. It is the first level of software testing. The aim behind unit testing is to validate unit component with its performance. A unit is a single testable part of a software system and tested during the development phase of the application software. This testing aims to test the correctness of isolated code. A unit component is an individual function or code of the application. White box testing approach used for unit testing and usually done by the developers.

4.2 Integrated Testing:

Integration testing is the second level of the software testing process comes after unit testing. In this testing, units or individual components of the software are tested in a group. The focus of the integration testing level is to expose defects at the time of interaction between integrated components or units. Unit testing uses modules for testing purpose, and these modules are combined and tested in integration testing. The Software is developed with a number of software modules that are coded by different coders or programmers. The goal of integration testing is to check the correctness of communication among all the modules.

5. Working Model ScreenShots

5.1 Face Detection Code Working:

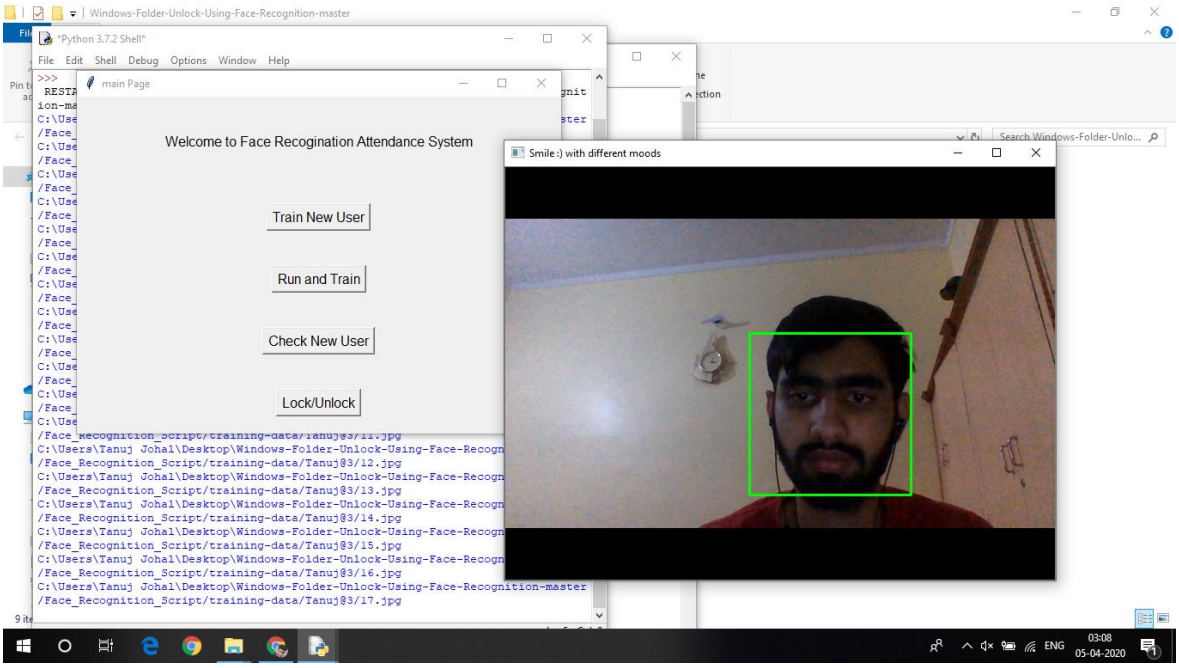

Fig 5.1

5.2 Face Recognition Code Working:

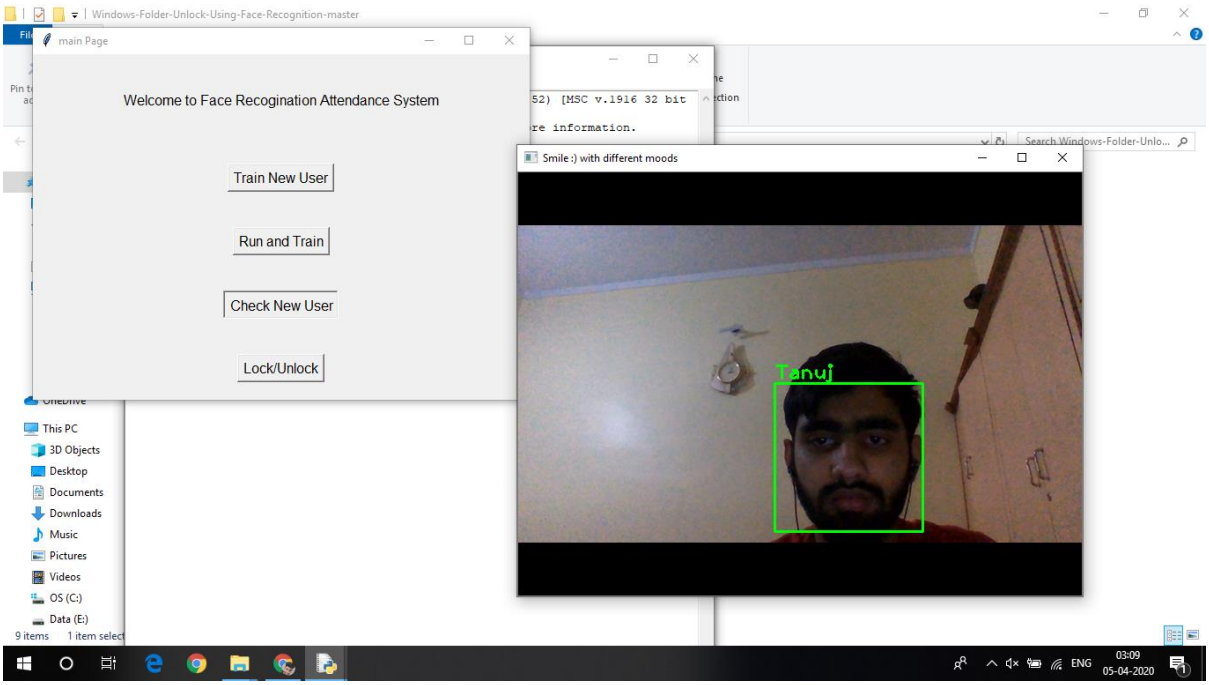

Fig 5.2

5.3 Graphical user interface

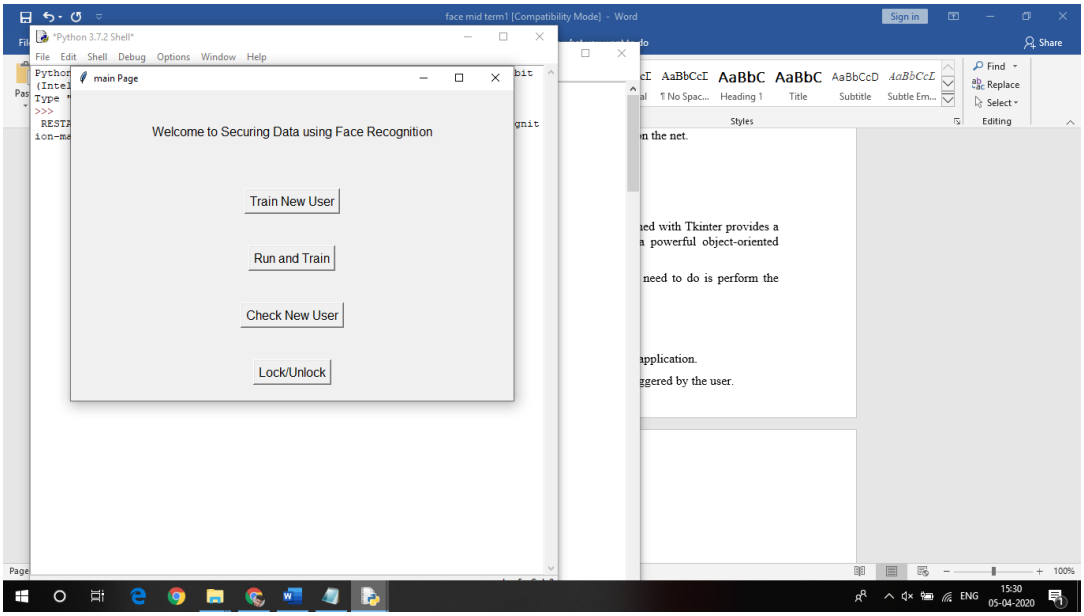

Fig 5.3

5.4 Locking Folder:

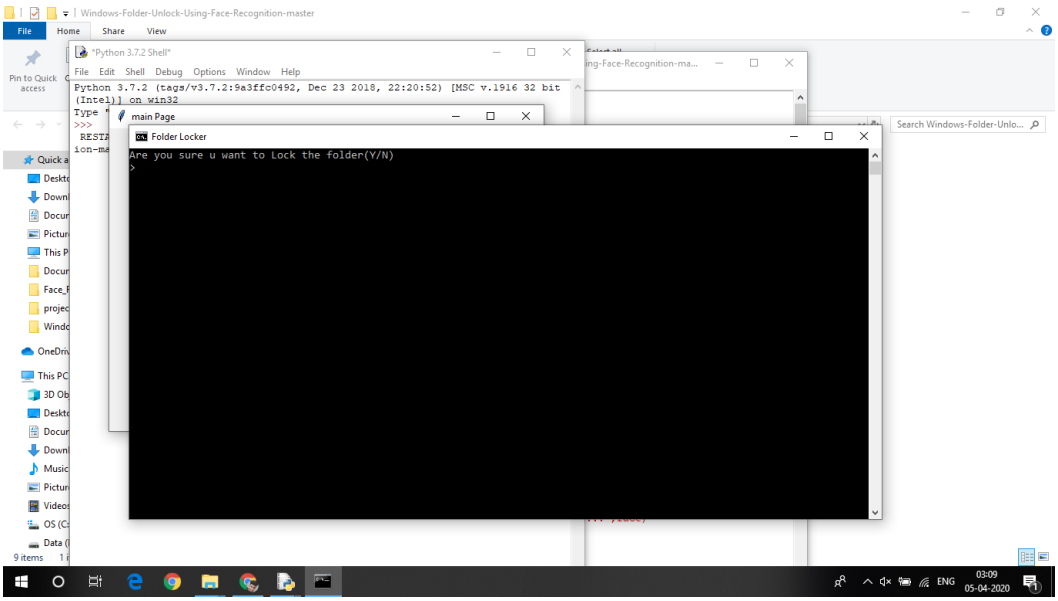

Fig 5.4

5.5 Unlocking Folder:

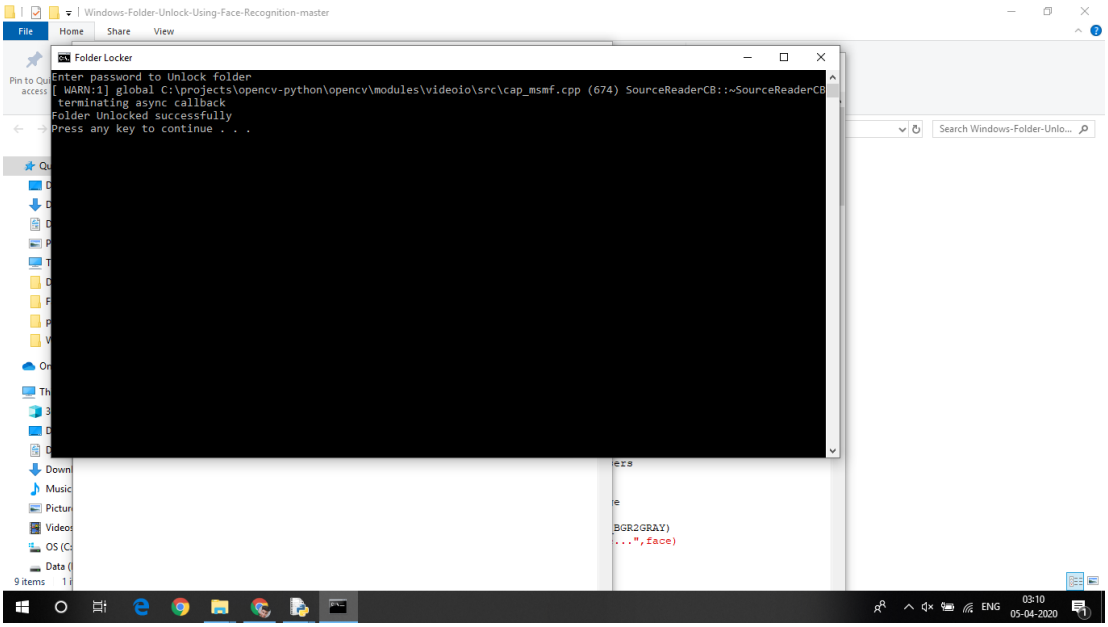

Fig 5.5

6. Contribution Detail

6.1 Tanuj Johal Contribution:

1. Work on Face Detection code that is written in python script.
2. Work on the graphical user Interface of the program. Connect all the backend code. Connect multiple Script and use subprocess to triggered folder locking .bat file.
3. Test the Code on multiple environment and analysis the Code accuracy on multiple training data set as well.
4. Record and drop the video on github reprository.

6.2 Anushka Soni Contibution:

1. Work on face recognition code that is written in python. Use open cv2 for the recognition of the face and use detection code as a module to interact with user.
2. Work on .bat file as well. Code the power shell script for locking and unlocking of the folder which uses face as a password.
3. Connect the shell script with the face recognition app.
4. Test the accuracy of the working module using different testing technique.

7. Conclusion and Future Scope

7.1 Conclusion

The approach presented here for face detection, tracking and folder unlocking decreases the computation time producing results with high accuracy. Tracking of a face in database that we have used local database system is done using KLT algorithm whereas Viola Jones is used for detecting facial features. The Face recognition code not only work in video sequences, but it has also been tested on live video using a webcam. Using this system many security and surveillance systems can be developed and required object can be traced down easily. In the coming days these algorithms can be used to detect a particular object rather than faces. Also we can use this project to lock any folder or we can use it to lock and unlock not only our folder but also we can use it for unlocking out laptop. It can increased the security of any personal data.

7.2 Future Scope

1. Future work is to work on the same domain but to track a particular face in a video sequence. That is like avoiding all other faces except the face required.

2. Viola Jones Algorithm is only meant for frontal and upright movement of the faces. It doesn't work when it comes to any arbitrary movement and hence, doesn't make sense. We would try to train classifiers so that it is subtle to all sort of movements.

3. We would definitely try to find the displacement of the Eigen vectors using Taylor Series.

4. We would try to mark attendance of all the student who are sitting in the classroom using Face Recognition System.

5. We can install our model to the entrance of our Home Entry which makes our Home more protective from thief and all.

8. References

[1] IEEE Visual Surveillance; DorinComaniciu and Visvanathan Ramesh

[2] International Journal of Soft Computing and Artificial Intelligence. ; MAMATA S. KALAS

[3] IJCTA, vol. 2. Nandita Sethi and Alankrita Aggarwal.

[4] Department of Electrical Engineering, Polytechnic University, Brooklyn. Zhu Liu and Yao Wang

[5] Pulsar team, INRIA. Etienne Corvee and Francois Bremond.

[6] ARPN Journal of Engineering and Applied Sciences. 10(17): 7678-7683. Divya George and Arunkant A. Jose.

[7] Conference on Computer Vision and Pattern Recognition. Paul Viola and Michael Jones.

[8] IEEE Trans. On Pattern Analysis and Machine Intelligence. L. Stan and Z. Zhang

[9] Conference on Computer Systems and Technologies. Peter Gejgu and Martin perka

[10] International Journal of Computer Vision, 57:137154, 2004. Paul Viola and Michael Jones

[11] Mitsubishi Electric Research Lab TR2000396, (July), 2003. M Jones and P Viola