

FACE DETECTION SYSTEM

MINI PROJECT REPORT-IT323

**BACHELOR OF TECHNOLOGY
IN
COMPUTER SCIENCE & ENGINEERING**

By

Unmilan Baruah

Roll Number: 180103024

UNDER THE GUIDANCE OF

Dr. Tanuja Das

Assitant Professor



**DEPARTMENT OF INFORMATION TECHNOLOGY
GAUHATI UNIVERSITY
GUWAHATI, INDIA
JUNE –2017**



GAUHATI UNIVERSITY
DEPARTMENT OF INFORMATION TECHNOLOGY
GopinathBordoloi Nagar, Jalukbari Guwahati-781014

DECLARATION

I *Unmilan Baruah*, Roll No: **180103024**, a B.Tech. student of the department of Information Technology, Gauhati University hereby declares that I have compiled this report reflecting all my works during the semester long full time mini project as part of my BTech curriculum.

I declare that I have included the descriptions etc. of my project work, and nothing has been copied/replicated from other's work. The facts, figures, analysis, results, claims etc. depicted in my thesis are all related to my full time project work.

I also declare that the same report or any substantial portion of this report has not been submitted anywhere else as part of any requirements for any degree/diploma etc.

Unmilan Baruah

30/09/2021

Computer Science and Engineering



GAUHATI UNIVERSITY
DEPARTMENT OF INFORMATION TECHNOLOGY
GopinathBordoloi Nagar, Jalukbari Guwahati-781014

Date:

CERTIFICATE

This is to certify that **Unmilan Baruah** bearing Roll No: **180103024** has carried out the project work "*Face Detection*" under my supervision and has compiled this report reflecting the candidate's work in the semester long project. The candidate did this project full time during the whole semester under my supervision, and the analysis, results, claims etc. are all related to his/her studies and works during the semester.

I recommend submission of this project report for the 6th semester examination of Bachelor of Technology in Computer Science & Engineering of Gauhati University.

Dr. Tanuja Das
Assistant Professor, Dept. of IT, GU

ACKNOWLEDGEMENT

I take this opportunity to express my sincere and heartfelt gratitude to Dr. Tanuja Das, Assistant Professor, Department of Information Technology, Gauhati University, Assam for her guidance and supervision throughout our project work. I also express my gratitude towards the fraternity of the Department of Information Technology of Gauhati University, Assam

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. Now a days, this is mostly done using *face_recognition* library. It uses face encodings for the face found in the images/videos. A face encoding is basically a way to represent the face using a set of *128* computer-generated measurements.

This project uses *face_recognition* and *OpenCV* libraries of python to detect faces in a video camera, represent the faces using face encodings, which done by the “*face_encoding*” function of the *face_recognition* library, present in the dataset (here, I used a folder containing images as a dataset) of images as well as for the faces in the video camera. Then, encodings of the faces in the video as well as in the dataset are compared.

If any of the encoding matches, then this means that the face is recognized and “*Welcome To Home*” dialog box appears. Otherwise, a red rectangle is drawn around the face in the video camera along with text as “*Intruder*”.

TABLE OF CONTENTS

Chapter 1: Introduction (Page no. 8)

- Overview (Page no. 8)
- Objective (Page no. 8)

Chapter 2: Modules Used (Page no. 9)

Chapter 3: Requirements (Page no. 10)

Chapter 4: Steps Implemented (Page no. 11)

Chapter 4: Project Code (Page no. 12-15)

- detect.py (Page no. 12-13)
- detection.py (Page no. 14)
- welcome.py (Page no. 15)

Chapter 5: Results (Page no. 16-17)

Chapter 6: Conclusion (Page no. 18)

Chapter 7: References (Page no. 19)

LIST OF FIGURES

Figure 1.....	16
Figure 2.....	16
Figure 3.....	17

INTRODUCTION

Overview:

The *Face Detection System* is a biometric security system project which uses the characteristics/data of particular face(s), represent the gathered characteristics/data in 128 computer-generated *face encodings*. Finally, these *encodings* are used to compare with dataset of images (dataset, in the project is a folder name “*admin*”, containing images) and thus, efficiently recognizes an “*admin*” or an “*intruder*”.

Objective:

The main objective of this project is to detect faces in video camera, get the encodings of the faces, compare with encodings of the dataset images and then predict whether the person in the video camera is recognized (*admin*) or not (*intruder*).

MODULES USED

For accurate as well as efficient working of the project, various Python modules/libraries are used.

They are:

1. **cv2 (OpenCV)**
2. **face_recognition**
3. **OS**
4. **PyQt5**

OpenCV: OpenCV is the huge open-source library for the computer vision, machine learning, and image processing and now it plays a major role in real-time operation which is very important in today's systems. By using it, one can process images and videos to identify objects, faces, or even handwriting of a human.

Face-recognition: Used to detect as well as recognise faces in an image or a video or in a folder full of photos.

OS: The OS module in Python provides functions for interacting with the operating system. OS comes under Python's standard utility modules. This module provides a portable way of using operating system-dependent functionality.

PyQt5: There are so many options provided by Python to develop GUI application and PyQt5 is one of them. PyQt5 is cross-platform GUI toolkit, a set of python bindings for Qt v5. One can develop an interactive desktop application with so much ease because of the tools and simplicity provided by this library.

REQUIREMENTS

Software Requirements

- Operating system : Microsoft Windows10
- Application Software : Anaconda Navigator
- Language : Python

Hardware Requirements

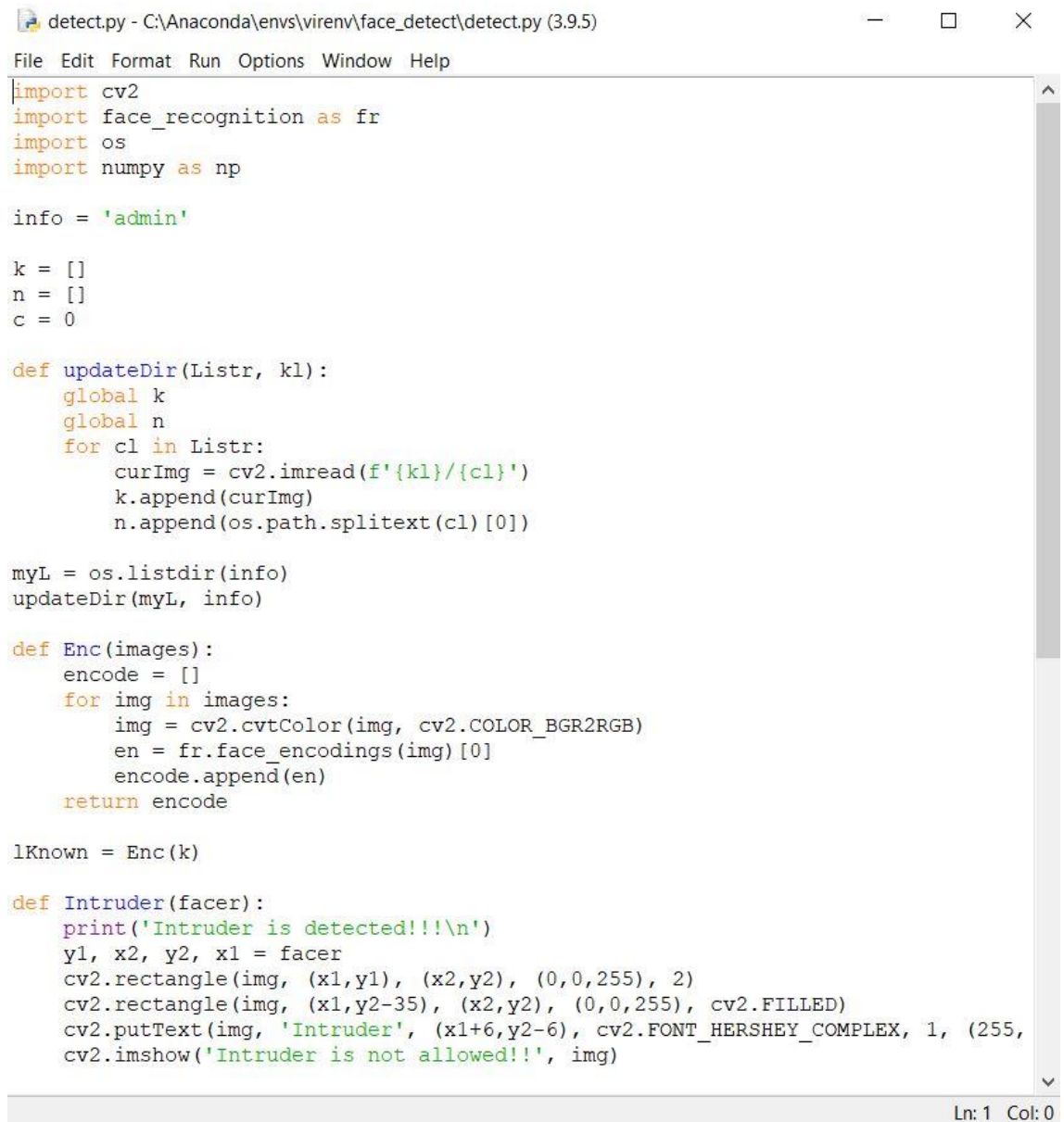
- System type: 64-bit Operating system
- RAM : 4GB
- Processor : Intel(R) Core(TM) i3-600GU CPU@2.00GHz

STEPS IMPLEMENTED

- Firstly; cv2(OpenCV), os, face_recognition, PyQt5, libraries are imported.
- Next, faces are detected in the images present in the dataset folder named “*admin*”, these faces are converted into “*face encodings*” using *face_encodings* function of *face_recognition* module.
- Then, input is taken as a video capture from the webcam. In input, the face are detected using *face_locations* function of *face_recognition* module and then converted into “*face encodings*”.
- Later on, these *encodings* of input as well as dataset images are compared using *compare_faces* function of *face_recognition* module. *compare_faces* function returns only *True* or *False*.
- The *face_distance* function of *face_recognition* module is used to find the difference between the *encodings* of input and of the dataset images and *argmin* function of *numpy* module is used to find the minimum of the difference.
- After getting the least difference between anyone of the dataset image and the input, the *encodings* of both of them are compared using *compare_faces* function.
- If the output is *True*, then the face in the input is recognised as an “*Admin*”. Else, the system will alarmed for an intruder with a *red rectangle* around the face in the video input.

PROJECT CODE

1. detect.py:



```
detect.py - C:\Anaconda\envs\viren\face_detect\detect.py (3.9.5)
File Edit Format Run Options Window Help

import cv2
import face_recognition as fr
import os
import numpy as np

info = 'admin'

k = []
n = []
c = 0

def updateDir(Listr, kl):
    global k
    global n
    for cl in Listr:
        curImg = cv2.imread(f'{kl}/{cl}')
        k.append(curImg)
        n.append(os.path.splitext(cl)[0])

myL = os.listdir(info)
updateDir(myL, info)

def Enc(images):
    encode = []
    for img in images:
        img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
        en = fr.face_encodings(img)[0]
        encode.append(en)
    return encode

lKnown = Enc(k)

def Intruder(facer):
    print('Intruder is detected!!!\n')
    y1, x2, y2, x1 = facer
    cv2.rectangle(img, (x1,y1), (x2,y2), (0,0,255), 2)
    cv2.rectangle(img, (x1,y2-35), (x2,y2), (0,0,255), cv2.FILLED)
    cv2.putText(img, 'Intruder', (x1+6,y2-6), cv2.FONT_HERSHEY_COMPLEX, 1, (255,
    cv2.imshow('Intruder is not allowed!!', img)
```

Ln: 1 Col: 0

PROJECT CODE

```
video_capture = cv2.VideoCapture(0)

def Found():
    print('Admin found')
    video_capture.release()
    cv2.destroyAllWindows()
    os.system('python welcome.py')

while c==0:
    _, img = video_capture.read()
    img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
    iloc = fr.face_locations(img)
    ien = fr.face_encodings(img, iloc)
    for eF, fL in zip(ien, iloc):
        mat = fr.compare_faces(lKnown, eF)
        Dis = fr.face_distance(lKnown, eF)
        index = np.argmin(Dis)
        if mat[index]:
            c = 1
            Found()
        else:
            c = 0
            Intruder(fL)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

video_capture.release()
cv2.destroyAllWindows()
```

Ln: 1 Col: 0

PROJECT CODE

2. detection.py:

```
detection.py - C:\Anaconda\envs\virenv\face_detect\detection.py (3.9.5)
File Edit Format Run Options Window Help
from PyQt5 import QtCore, QtGui, QtWidgets
import os

class Ui_FaceDetect(object):
    def setupUi(self, FaceDetect):
        FaceDetect.setObjectName("FaceDetect")
        FaceDetect.resize(387, 80)
        self.Detect = QtWidgets.QPushButton(FaceDetect)
        self.Detect.setGeometry(QtCore.QRect(10, 20, 371, 28))
        self.Detect.setObjectName("Detect")

        self.Detect.clicked.connect(self.c1)

        self.retranslateUi(FaceDetect)
        QtCore.QMetaObject.connectSlotsByName(FaceDetect)

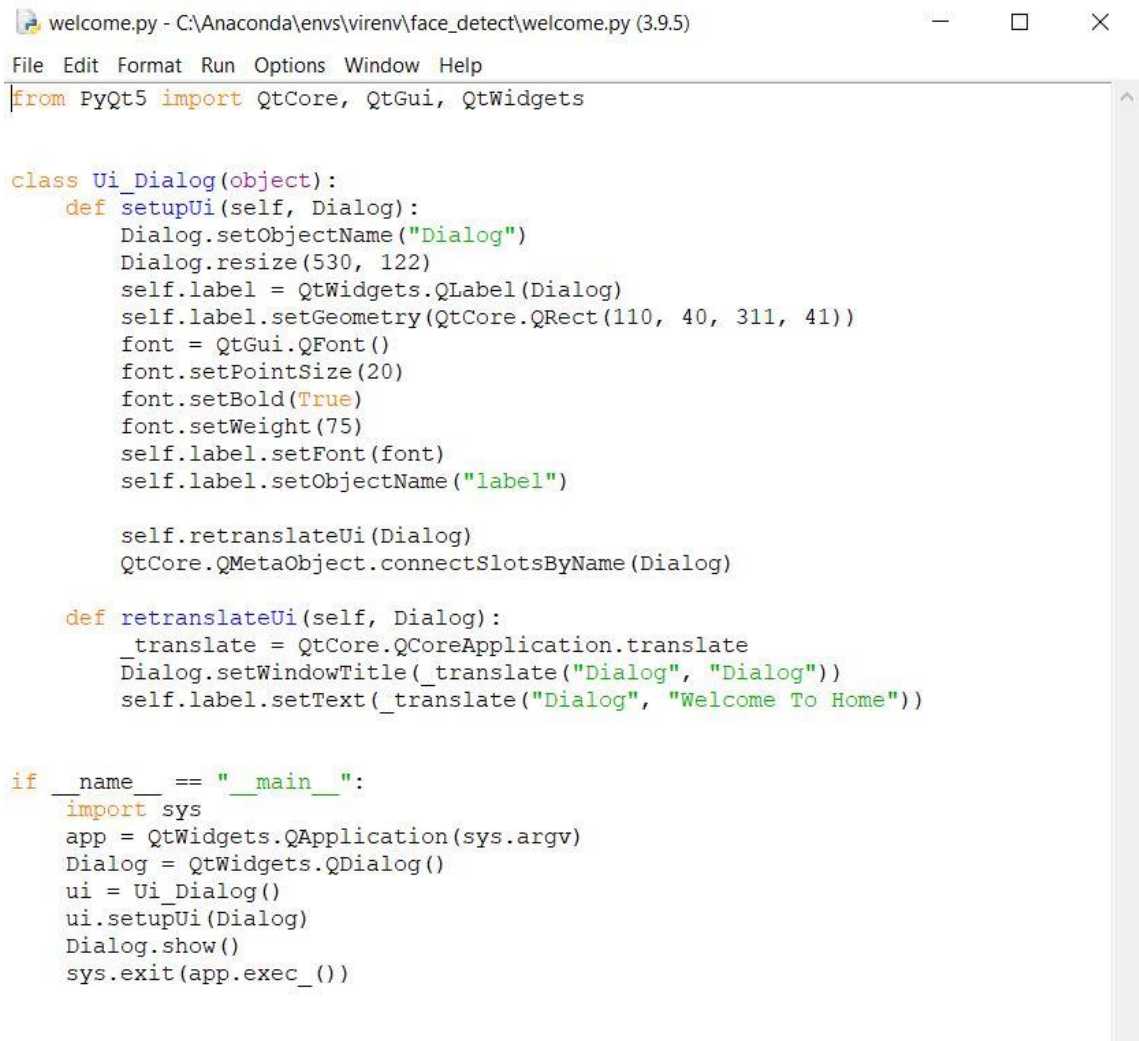
    def retranslateUi(self, FaceDetect):
        _translate = QtCore.QCoreApplication.translate
        FaceDetect.setWindowTitle(_translate("FaceDetect", "Security System"))
        self.Detect.setText(_translate("FaceDetect", "Activate Detector"))

    def c1(self):
        os.system('python detect.py')

if __name__ == "__main__":
    import sys
    app = QtWidgets.QApplication(sys.argv)
    FaceDetect = QtWidgets.QDialog()
    ui = Ui_FaceDetect()
    ui.setupUi(FaceDetect)
    FaceDetect.show()
    sys.exit(app.exec_())
```

PROJECT CODE

3. welcome.py:



```
welcome.py - C:\Anaconda\envs\venv\face_detect\welcome.py (3.9.5)
File Edit Format Run Options Window Help
from PyQt5 import QtCore, QtGui, QtWidgets

class Ui_Dialog(object):
    def setupUi(self, Dialog):
        Dialog.setObjectName("Dialog")
        Dialog.resize(530, 122)
        self.label = QtWidgets.QLabel(Dialog)
        self.label.setGeometry(QtCore.QRect(110, 40, 311, 41))
        font = QtGui.QFont()
        font.setPointSize(20)
        font.setBold(True)
        font.setWeight(75)
        self.label.setFont(font)
        self.label.setObjectName("label")

        self.retranslateUi(Dialog)
        QtCore.QMetaObject.connectSlotsByName(Dialog)

    def retranslateUi(self, Dialog):
        _translate = QtCore.QCoreApplication.translate
        Dialog.setWindowTitle(_translate("Dialog", "Dialog"))
        self.label.setText(_translate("Dialog", "Welcome To Home"))

if __name__ == "__main__":
    import sys
    app = QtWidgets.QApplication(sys.argv)
    Dialog = QtWidgets.QDialog()
    ui = Ui_Dialog()
    ui.setupUi(Dialog)
    Dialog.show()
    sys.exit(app.exec_())
```

RESULTS

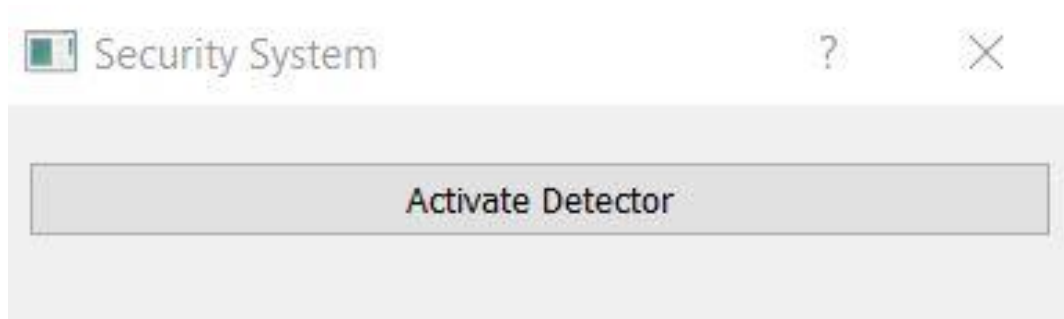


Figure 1: Face Detection System

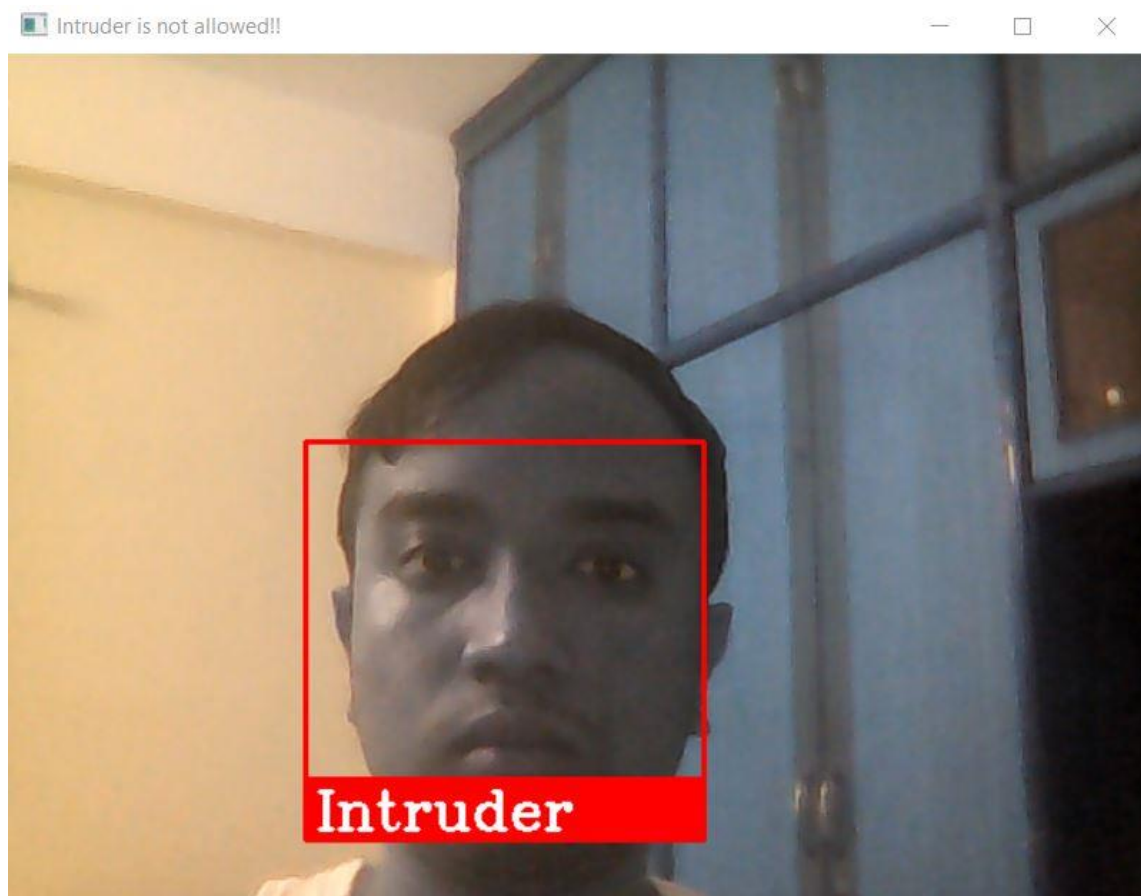


Figure 2: When the person in video camera is not recognised. (Intruder)

RESULTS



Figure 3: When the person in the video camera is recognised. (Admin)

CONCLUSION

This project uses *face recognition* concept to work as a surveillance system. The logic implemented behind this project can be used to make various applications for various *fields* like biometrics, information system, law enforcement, access control, etc.

REFERENCES

- <https://face-recognition.readthedocs.io/en/latest/readme.html>
- <https://www.geeksforgeeks.org/python-introduction-to-pyqt5/>
- <https://www.geeksforgeeks.org/os-module-python-examples/>
- <https://www.geeksforgeeks.org/opencv-overview/>