REPORT FOR THE FACE RECOGNITION

As a project work for the course

PYTHON PROGRAMMING(INT 213)

Name	:Jaya Surya Immidi
Registration Number	:12008968
Name	:Kranthi Kiran Reddy
Registration Number	:12010287
Program	:CSE B.Tech
Semester	:Third
School	:School of Computer Science and Engineering
Name of University University	:Lovely Professional
Date of Submission	:19t November 2021

FACE RECOGNITION

19TH November 2021

ABSTRACT:

Face recognition technology is a biometric technology, which is based on the identification of facial features of a person. People collect the face images, and the recognition equipment automatically processes the images. The paper introduces the related researches of face recognition from different perspectives. The paper describes the development stages and the related technologies of face recognition. We introduce the research of face recognition for real conditions, and we introduce the general evaluation standards and the general databases of face recognition. We give a forward-looking view of face recognition. Face recognition has become the future development direction and has many potential application prospects.

ACKNOWLEDGEMENT:

Facial recognition is used to identify specific individuals in the form of digital image by recognizing and comparing patterns. It is a method of acknowledgement whether two images are of same person or not

Table of Contents

INTRODUCTION

Context

This project has been done due as part of our assignment for the INT213 at Lovely Professional University.

Motivations

Being extremely interested in everything related to coding of object oriented languages ,in addition a group project was a great occasion to give us the time to learn something new .the fact that we can give maximum time to the coding and algorithms. That's why we decided to conduct our project around the Machine learning.

Idea

As a first experience, we wanted to try something that is new to us and the difficultly should be more ,then we can keep our maximum efforts to solve and complete the project.The project is regarded to our daily activities and most common thing we will do with our smart phones that is "Face RECOGNITION".

TEAM MEMBERS:-

Team Leader:-

- Jaya Surya Immidi

  Contributions:-

  1.Coding(shared)
  2.GUI
  3.Report
  4.Debugging

- Kranthi Kiran Reddy

  Contributions:-

  1.Coding(shared)
  2.Report
  3.Debugging

LIBRARIES

OpenCV:

OpenCV (Open Source Computer Vision Library) is a huge open-source library for computer vision, machine learning, and image processing. OpenCV supports a wide variety of programming languages like Python, C++, Java, etc. It can process images and videos to identify objects, faces, or even the handwriting of a human .OpenCV is an open-source computer vision and machine learning software library. which is built to provide a common infrastructure for machine learning algorithms and computer vision. It has thousands of optimized algorithms which can be used different purposes like detecting and recognizing faces, identifying objects and many more. We need it to take pictures using our webcam and some manipulation needed to be done in the image.
We can install the opencv using command prompt using the text' pip install opencv-python'.all inbuilt functions will automatically downloaded in the pc.

Proposed Module

OS(Operating System):

It is possible to automatically perform many operating system tasks. The OS module in Python provides functions for creating and removing a directory (folder), fetching its contents, changing and identifying the current directory, etc.

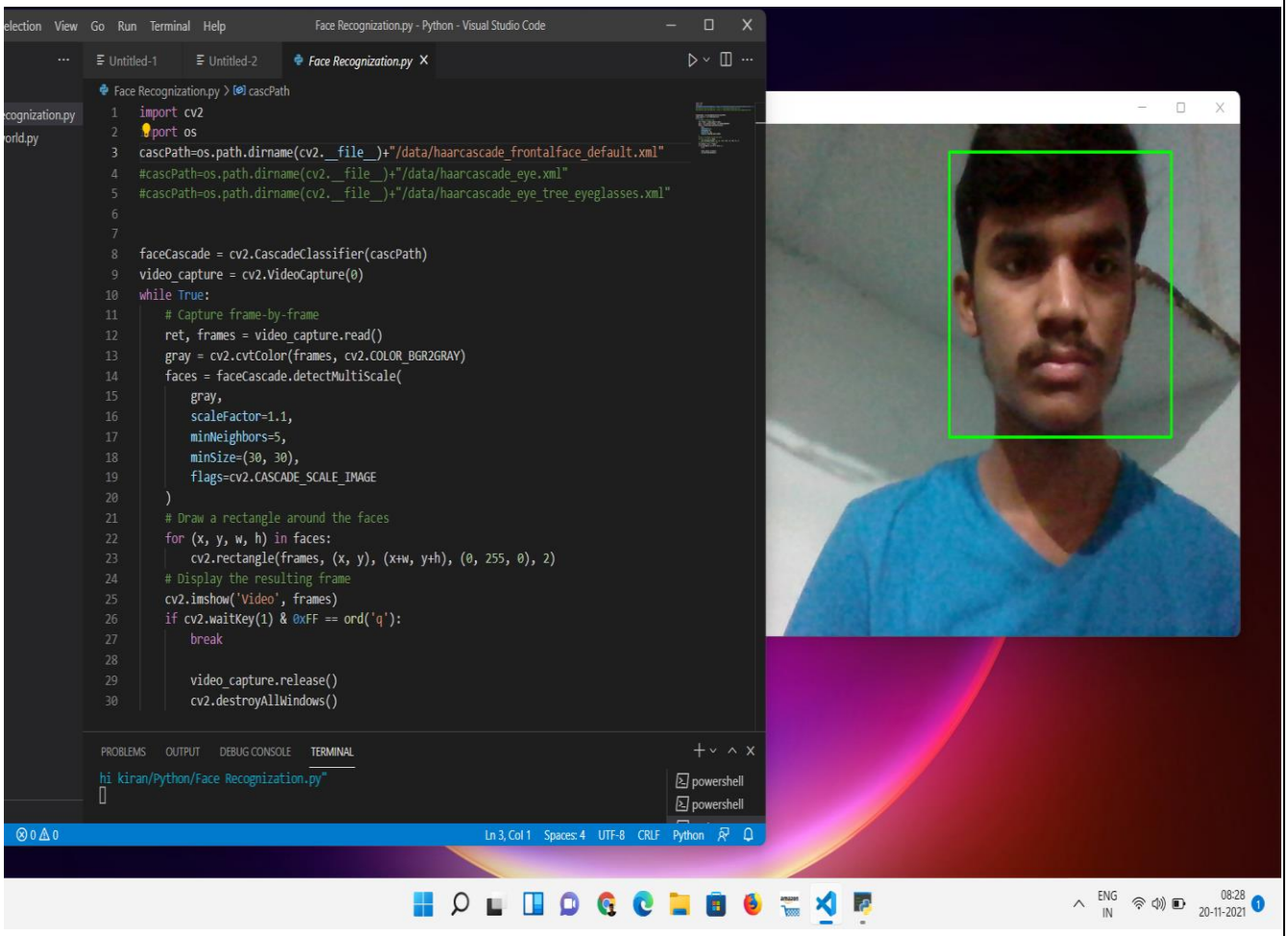
Python OS module provides the facility to establish the interaction between the user and the operating system. It offers many useful OS functions that are used to perform OS-based tasks and get related information about operating system.

You first need to import the os module to interact with the underlying operating system. So, import it using the 'import os' statement before using its functions.

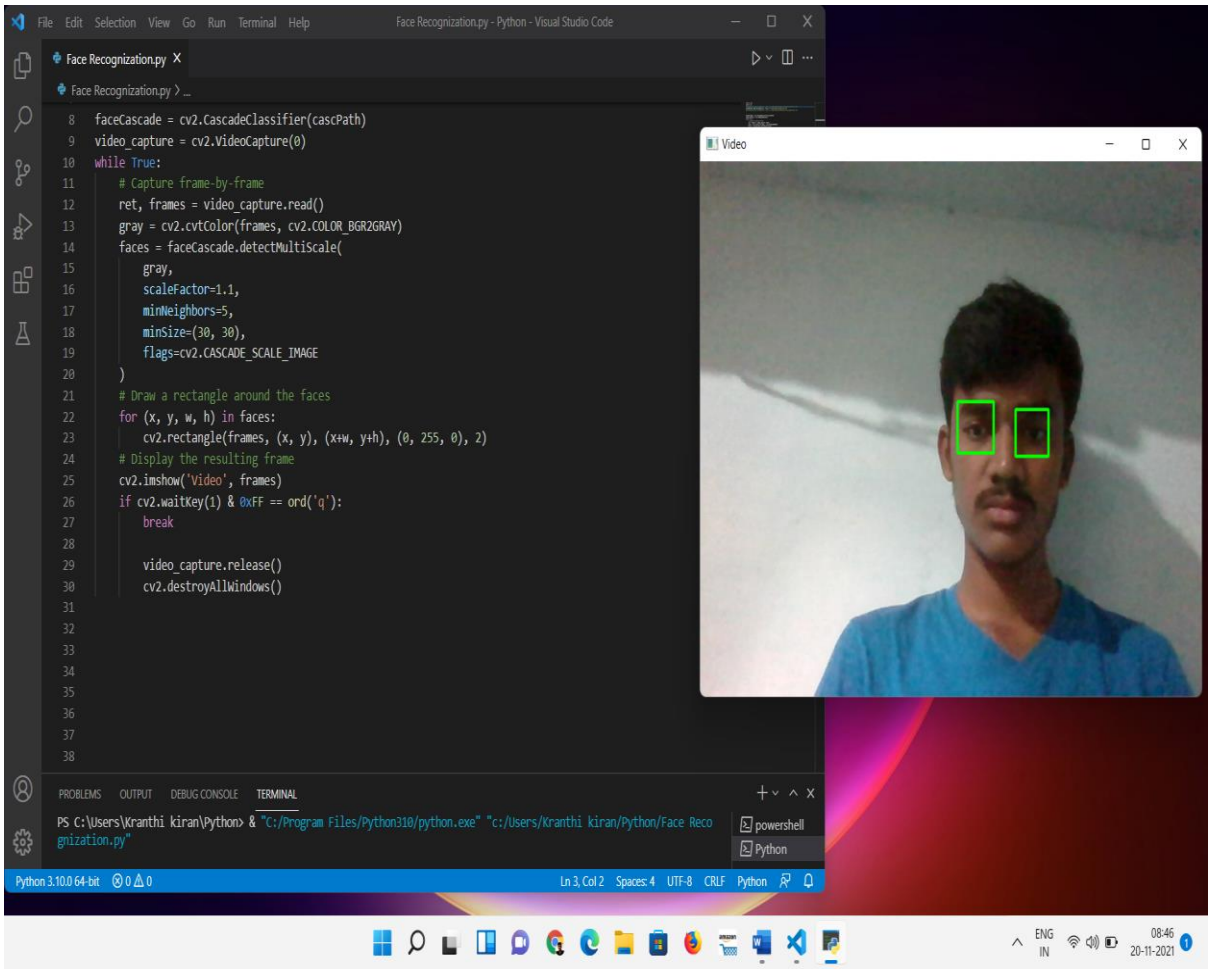
SCREENSHOTS:-

Student 1:

Recognition of Face:

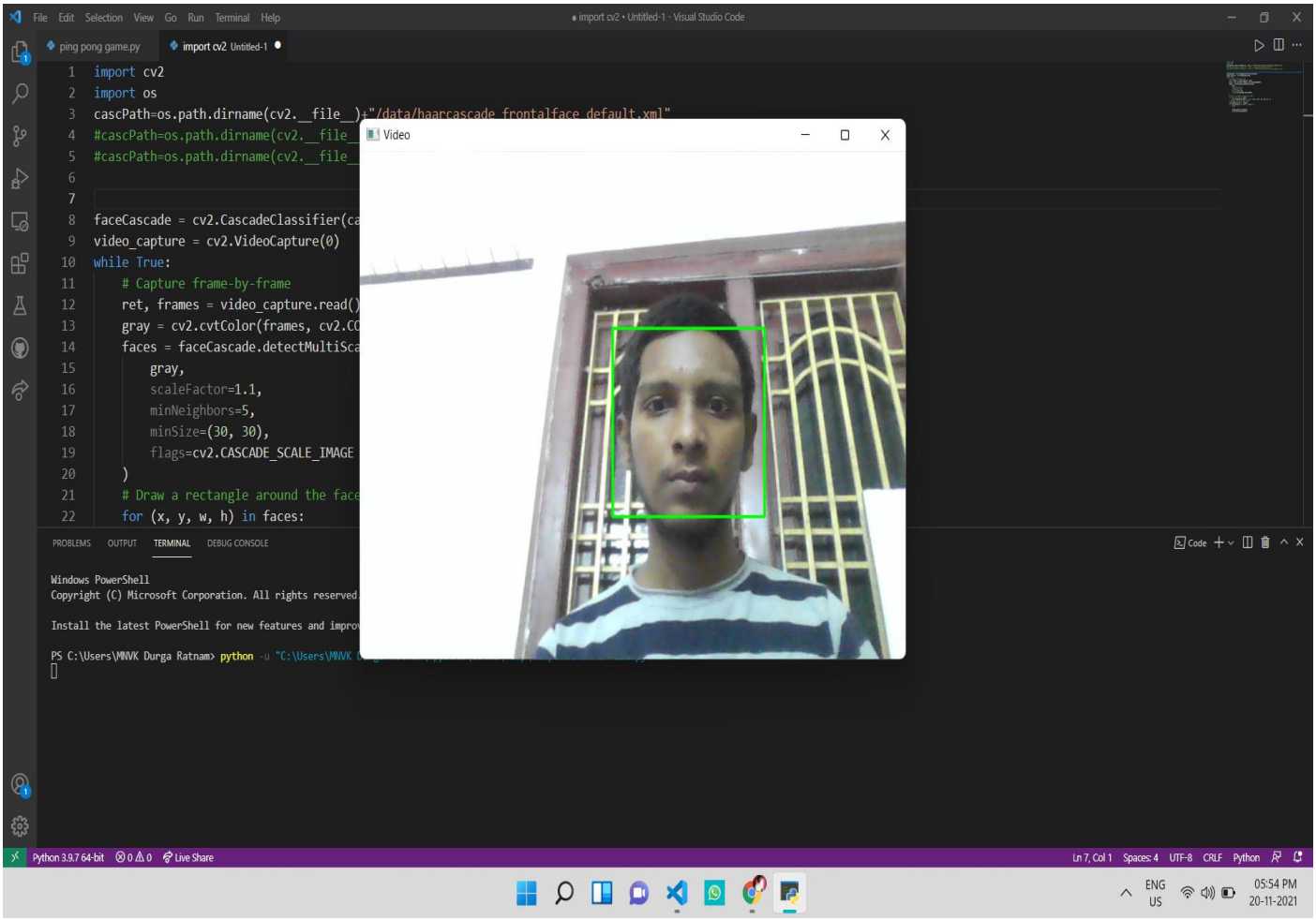


Recognition of Eyes:



Student 2:

Face Recognition:



CODING IN VS CODE

```
import cv2
import os
cascPath=os.path.dirname(cv2.__file__)+"/data/haarcascade_frontalface_default.
xml"
#cascPath=os.path.dirname(cv2.__file__)+"/data/haarcascade_eye.xml"
#cascPath=os.path.dirname(cv2.__file__)+"/data/haarcascade_eye_tree_eyeglasses
.xml"


faceCascade = cv2.CascadeClassifier(cascPath)
video_capture = cv2.VideoCapture(0)
while True:
    # Capture frame-by-frame
    ret, frames = video_capture.read()
    gray = cv2.cvtColor(frames, cv2.COLOR_BGR2GRAY)
    faces = faceCascade.detectMultiScale(
        gray,
        scaleFactor=1.1,
        minNeighbors=5,
        minSize=(30, 30),
        flags=cv2.CASCADE_SCALE_IMAGE
    )
    # Draw a rectangle around the faces
    for (x, y, w, h) in faces:
        cv2.rectangle(frames, (x, y), (x+w, y+h), (0, 255, 0), 2)
    # Display the resulting frame
    cv2.imshow('Video', frames)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

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

Haar Cascade:

Haar Cascade is basically a classifier which is used to detect the objects for which it has been trained for, from the source. The result is an XML file which stores the trained result. If said simply the Haar Cascade is trained by superimposing the positive image over a set of negative images. The training requires a high spec system and a good internet connection and thousands of training images that is why it is carried out in the server.

For increasing the efficiency of the results they use high-quality images and increase the number of stages for which the classifier is trained. We need haar cascade frontal face recognizer to detect the face from our webcam.

In this project haarcascade is used in 3 different positions on the face one is “haarcascade_frontalface_default”, which draw a rectangular box on the face and the other is the “haarcascade_eye” ,which draw a box on the two eyes and the last one is “haarcascade_eye_tree_eyeglasses” which draw a box on the eye glasses if present . These data is stored in cv2 library .

TEST-CASES

A test case is a set of conditions or variables under which a tester will determine whether a system under test satisfies requirements or works correctly. The process of developing test cases can also help find problems in the requirements or design of an application.

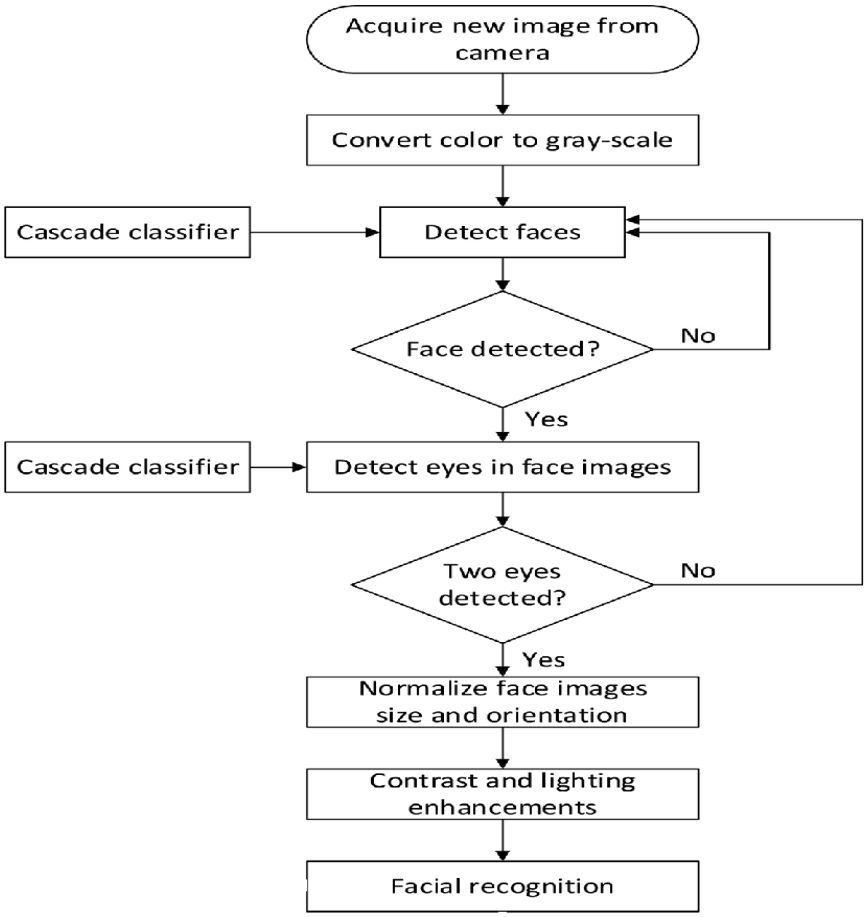
S.no.	Testing Name	description	Expected results	Actual result	remarks	output
1.	Debugging	Checking for the errors in the code	displaying of the errors if present	Displaying some syntax errors	After debbugging we correct the code	passed
2.	Program Initiating	Running the whole code using vs code	Program run	failed	Failed because of the technical error	Failed
3.	Program Initiating	Running the whole code using vs code	Program run	failed	Failed because there is no open cv in the system	Failed
4.	Program Initiating	Running the whole code using vs code	Program run	passed	Corrected the errors and installed libraries	Passed

METHODOLOGY

Below are the methodology and descriptions of the applications used for data gathering, face detection,training and face recognition.

FACE DETECTION:

First stage was creating a face detection system using Haar-cascades. Although, training is required for creating new Haar-cascades, OpenCV has a robust set of Haar-cascades that was used for the project.
Using face-cascades alone caused random objects to be identified and eye cascades were incorporated to obtain stable face detection. The flowchart of the detection system can be seen in the below diagram.



Face and eye classifier objects are created using classifier class in OpenCV through the cv2.CascadeClassifier() and loading the respective XML files. A camera object is created using the cv2.VideoCapture() to capture images. By using the CascadeClassifier.detectMultiScale() object of various sizes are matched and location is returned. Using the location data, the face is cropped for further verification. Eye cascade is used to verify there are two eyes in the cropped face. If satisfied a marker is placed around the face to illustrate a face is detected in the location.

For this project three algorithms are implemented independently. These are Eigenface, Fisherface and Linear binary pattern histograms respectively. All three can be implemented using OpenCV libraries.

There are three stages for the face recognition as follows:

1. Collecting images IDs
2. Extracting unique features, classifying them and storing in XML files
3. Matching features of an input image to the features in the saved XML files and predict identity

CONCLUSION

This project describes the mini-project for visual perception and autonomy module. Next, it explains the technologies used in the project and the methodology used. Finally, it shows the results, discuss the challenges and how they were resolved followed by a discussion. Using Haar-cascades for face detection worked extremely well even when subjects wore spectacles.

Face recognition technology has come a long way in the last twenty years. Today, machines are able to automatically verify identity information for secure transactions, for surveillance and security tasks, and for access control to buildings etc. These applications usually work in controlled environments and recognition algorithms can take advantage of the environmental constraints to obtain high recognition accuracy. However, next generation face recognition systems are going to have widespread application in smart environments -- where computers and machines are more like helpful assistants

A real life example of this project is a system to identify known troublemakers in a mall or a supermarket to provide the owner a warning to keep him alert or for automatic attendance taking in a class.

The main working principle of the project is that, the video captured data is converted into image to detect and recognize it. Further the recognized image of the student is provided with attendance, else the system marks the database as absent.