

Face Detection and Recognition

A
Project Report
Submitted for the partial fulfillment
of B.Tech. Degree
in
COMPUTER SCIENCE & ENGINEERING
by

Pranjul Pal 1805210035
Yash Jaiswal 1805210064
Ziyauddin 1805210067

Under the supervision of

Dr. Manik Chandra
Ms. Mudita Sharan



Department of Computer Science and Engineering
Institute of Engineering and Technology

Dr. A.P.J. Abdul Kalam Technical University, Lucknow, Uttar Pradesh.

July, 2022

Table of Content

● Declaration.....	3
● Certificate.....	4
● Acknowledgement.....	5
● Abstract.....	6
● List of Figures.....	7
1. Introduction.....	8
2. Literature Survey.....	9
3. Motivation.....	11
4. Face Detectors and Recognizers.....	12
a. Face Detectors.....	12
b. Face Recognizers.....	13
5. Facial Landmarks on Human Faces.....	17
6. Methodology.....	19
a. Import Required modules.....	20
b. Camera Setup.....	20
c. Face Detection and Creating Dataset.....	21
d. Training Dataset.....	23
e. Face Recognition.....	25
f. Flow of Working Project.....	27
7. Results.....	29
a. Face Detection Results.....	29
b. Face Recognition Results.....	30
8. Conclusion.....	32
9. References.....	33

Declaration

We hereby declare that this submission is our own work and that, to the best of our belief and knowledge, it contains no material previously published or written by another person or material which to a substantial error has been accepted for the award of any degree or diploma of university or other institute of higher learning, except where the acknowledgement has been made in the text. The project has not been submitted by us at any other institute for the requirement of any other degree.

Submitted by: -

Date: 25-05-2022

(1) Name: **Pranjal Pal**

Roll No.: 1805210035

Branch: C.S.E.

Signature:

(2) Name: **Yash Jaiswal**

Roll No.: 1805210064

Branch: C.S.E.

Signature:

(3) Name: **Ziyauddin**

Roll No.: 1805210067

Branch: C.S.E.

Signature:

Certificate

This is to certify that the project report entitled **Face Detection and Recognition** presented by **Pranjul Pal, Yash Jaiswal and Ziyauddin** in the partial fulfillment for the award of Bachelor of Technology in Computer Science and Engineering, is a record of work carried out by them under my supervision and guidance at the Department of Computer Science and Engineering at Institute of Engineering and Technology, Lucknow.

It is also certified that this project has not been submitted at any other Institute for the award of any other degrees to the best of my knowledge.

Dr. Manik Chandra

Department of Computer Science and Engineering
Institute of Engineering and Technology, Lucknow

Ms. Mudita Sharan

Department of Computer Science and Engineering
Institute of Engineering and Technology, Lucknow

Acknowledgement

We would like to express our sincere gratitude and appreciation to all those who gave us the opportunity to complete this project. First, we wish to express our sincere gratitude to our supervisors, Dr. Manik Chandra and Ms. Mudita Sharan, for their insightful comments, helpful information and practical advice that have helped us tremendously at all times in my research and writing of this project. Without their support and guidance, this project would not have been possible. We would like to express gratitude to all our friends who motivated us and helped us at each step in completing this project work. We would also like to express our sincere regards to all the Authors of all the references and other literary work referred to in this project work.

Pranjul Pal

Roll No.: 1805210035

Branch: C.S.E.

Yash Jaiswal

Roll No.: 1805210064

Branch: C.S.E.

Ziyauddin

Roll No.: 1805210067

Branch: C.S.E.

Abstract

In this project we have worked on face detection and recognition problems of an individual person. Our project is capable of detecting a human face and storing images of detected faces for training our model. In this task we will utilize the haar cascade classifier and LBPH calculations. By the utilization of LBPH we will get Local Binary Pattern Histogram of a picture which will be utilized to perceive and order pictures. When a face is identified in the real-time video, we will utilize the put away LBPH highlights of people to remember them in this picture or camera. In the event that the individual has previously showed up in the camera or has their information in the dataset then we will dole out them a name. Furthermore, we will likewise give the likelihood level of the picture matched from the dataset. We have additionally added django to our model and added styling in the site page. Thus, our entire work depends on the utilization of picture handling with the assistance of openCV and some utilization of Django.

List of Figures

Fig 4.1 HaarCascade Classifier.....	12
Fig 4.2 Eigenfaces Face Recognition.....	13
Fig 4.3 Fisherfaces Face Recognition.....	15
Fig 4.4 Local Binary Pattern Histogram.....	16
Fig 5.1 68 Point Landmark.....	18
Fig 6.1 Flow Chart.....	19
Fig 6.2 Dataset.....	22
Fig 6.3 HomePage.....	27
Fig 6.4 Train Model Result Page.....	27
Fig 6.5 Create Dataset Page.....	28
Fig 6.6 Face Recognition Page.....	28
Fig 7.1 Single Human Face Detection.....	29
Fig 7.2 Multi People Face Detection.....	30
Fig 7.3 Single Human Face Recognition.....	30
Fig 7.4 Multi People Face Recognition.....	31
Fig 7.5 Multi People Face Recognition.....	31

Chapter 1

Introduction

Technology has advanced a lot in the remaining few decades. Previously there have been no safety cameras, there have been solely safety guards who guard these areas. Once the protection cameras got here into existence, it grew to be convenient to locate humans passing inside the vary of CCTV cameras by way of absolutely looking out via the movies recorded. The face is one of the best methods to distinguish the man or woman identification of every other. Face awareness is a private identification device that makes use of the non-public traits of a individual to pick out the person's identity.

Below are the three steps for the face recognition which are as follows:

Data Gathering: Gather numerous face pictures of the people you need to perceive.

Training of Face Recognizer: Store the information of face and the names of particular individual to the face recognizer.

Face Recognition: Check on the new human countenances whether the prepared model remembers them.

OpenCV as of now has a face recognizer module, just our undertaking is to take care of it with the exact face information.

In general, in this task, we will figure out on the issue of face discovery and acknowledgment of a person. Our undertaking will be fit for distinguishing the human whose information we have gathered as of now. We will utilize Image Processing with the assistance of OpenCV, utilizing Python language.

Chapter 2

Literature Survey

Probably the earliest investigations on face acknowledgment were finished by Darwin and Galton. Darwin's work incorporates examination of the different looks because of changed close to home states, while Galton concentrated on facial profiles.

Be that as it may, the primary genuine attempts to help semi-robotized facial cognizance structures started in the last part of the Sixties and mid Seventies and had been founded absolutely on mathematical data. Here, milestones were put on photos finding the significant facial elements, like eyes, ears, noses, and mouth corners. Relative distances and points were processed from these milestones to a typical reference point and contrasted with reference information. In 1971 a framework was made of 21 emotional markers, for example, hair tone and lip thickness. These markers showed especially hard to mechanize because of the emotional idea of various of the assessments actually made absolutely manually.

A superior execution to perceive faces was finished by Fischler et al. (1973) and afterward by Yuille et al. (1992). This approach estimated the facial highlights utilizing layouts of single facial elements and planned these onto a worldwide format.

In synopsis, the majority of the created procedures during the primary phases of facial acknowledgment zeroed in on the programmed recognition of individual facial elements. The best benefits of these mathematical component based techniques are the obtuseness toward light and the instinctive comprehension of the extricated highlights. Regardless, without a doubt these days facial incorporate area and assessment methodologies are not strong adequate for the mathematical component based affirmation of a defy and mathematical properties alone are inadequate for stand up to affirmation.

Because of this drawback of mathematical element based affirmation, the technique has consistently been given up and an effort has been made in examining sweeping variety based procedures, which has given way improved results.

All encompassing variety based methods adjust a bunch of various countenances to get a correspondence between pixel powers, a closest neighbor classifier can be utilized to characterize new faces when the new picture is first adjusted to the arrangement of currently adjusted pictures.

By the presence of the Eigenfaces strategy, a measurable learning approach, this coarse technique was remarkably upgraded. Rather than straightforwardly contrasting the pixel forces of the different facial pictures, the element of the information powers was first decreased by a Principal Component Analysis (PCA) in the Eigenface method. Eigenfaces are an essential part of a large number of the picture based facial acknowledgment plans utilized today. One of the ongoing procedures is Fisherfaces. This method is broadly utilized and alluded to. It consolidates the Eigenfaces with Fisher Linear Discriminant Analysis (FLDA) to get better division of the appearances on the record of orientation. In Fisherfaces, the element of the info power vectors is decreased by PCA and afterward FLDA is applied to get an ideal projection for partition of the appearances from changed people. After the improvement of the Fisherface method, many related strategies have been proposed.

These new procedures target giving a shockingly better projection to the partition of appearances from changed people. They attempt to fortify the vigor in adapting to contrasts in enlightenment or picture present. Procedures like Kernel Fisherfaces, Laplacianfaces, or discriminative normal vectors can be found among these methodologies.

Chapter 3

Motivation

Face cognizance has currently acquired blooming interest and activity from the scientific neighborhood as properly as from the generic public. The pastime from the customary public is often due to the latest activities of terror round the world, which has elevated the demand for beneficial safety systems. Facial consciousness purposes are a long way from restrained to safety structures as described above.

To assemble these one-of-a-kind applications, specific and sturdy automatic facial cognizance techniques and methods are needed. However, these methods and techniques are presently no longer handy or solely accessible in incredibly complex, high priced setups.

The subject matter of this thesis is to assist in solving the tough mission of strong face awareness in an easy setup. Such an answer would be of first-rate scientific importance and would be beneficial to the public in general.

The targets of this postulation will be:

- To examine and sum up the course of facial acknowledgment.
- To check right now accessible facial acknowledgment out procedures.
- To plan and foster a vigorous facial acknowledgment calculation.

The calculation ought to be usable in a straightforward and effectively versatile arrangement. This infers a solitary camera arrangement, ideally a webcam, and no utilization of specific hardware. Other than these hypothetical goals, a proof-of-idea execution of the created strategy will be done.

Chapter 4

Face Detectors and Recognizers

4.1 Face Detectors

Object Detection the utilization of Haar highlight based overflow classifiers is an enormous item recognition strategy proposed through Paul Viola and Michael Jones in their paper, "Fast Object Detection the use of a Boosted Cascade of Simple Features" in 2001. It is a PC acquiring information on based absolutely procedure, where a fountain trademark is gifted from a great deal of top notch and unfortunate pictures. Seeing articles in various images is then utilized.

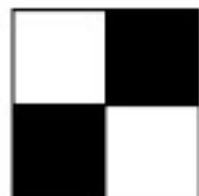
Here we will work with face location. At first, the calculation needs a great deal of pleasant pictures (pictures of countenances) and horrendous depictions (pictures other than faces) to show the classifier. Then we need to remove components from it. For this, Haar focuses demonstrated in the under picture are utilized. They are very much like our convolutional bit. Each capacity is a solitary cost got by deduction of the amount of pixels in the white locale from the amount of pixels in the dark district.



(a) Edge Features



(b) Line Features



(c) Four-rectangle features

4.2 Face Recognizers

OpenCV has three implicit face recognizers and is prepared to use any of them by fundamentally redesigning and changing a solitary line of code. Under are the names of these three face recognizers and their OpenCV calls which are extensively used in Confront Discovery and Recognition.

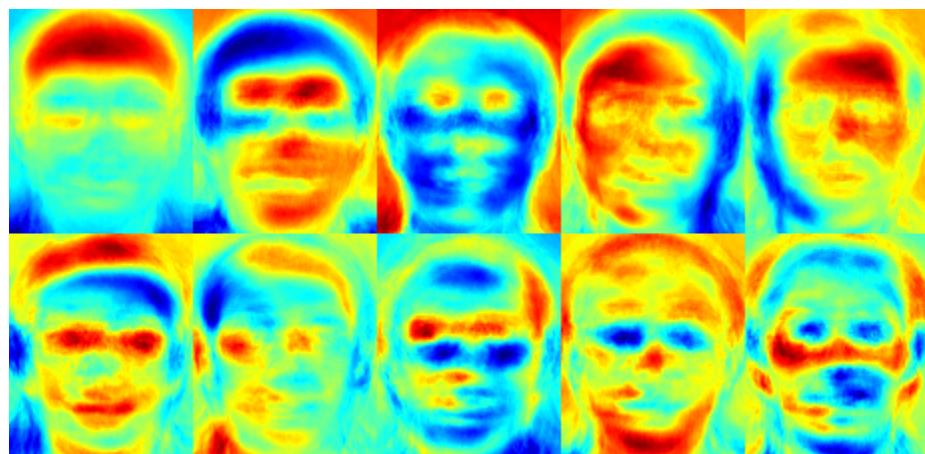
FisherFaces Face Recognizer

EigenFaces Face Recognizer

Nearby Binary Patterns Histogram Face Recognizer

EigenFaces Face Recognizer:

EigenFaces face recognizer shows up at all the training pix of the multitude of people as an aggregate and attempts to separate the viewpoints which are imperative and gainful (the perspectives that hold onto the most difference/change) and disposes of the unwinding of the perspectives which are presently not valuable. This way it no longer solely extracts the necessary factors from the coaching information however additionally saves reminiscence and time with the aid of discarding or eliminating the much less essential components. Important and beneficial extracted factors are known as primary components.



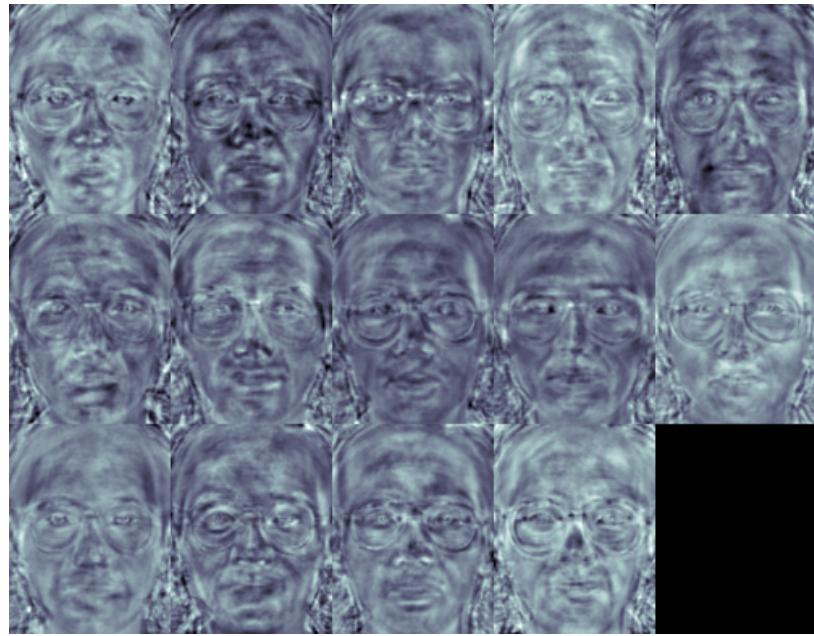
The principal factors totally imply faces and these countenances are alluded to as eigen faces and hence the title of the calculation is EigenFaces Face Recognizer. So this is in many cases how EigenFaces stand up to recognizer trains itself through removing basic parts. What's more, it furthermore has a capacity to keep current realities and it proceeds with a record of which significant issue has a place with which individual.

Later over the hour of acknowledgment, when you feed another photo or records to the calculation, it rehashes the equivalent strategy on that photograph also. It extricates the significant issue from that new picture and contrasts that issue and the posting of viewpoints it has saved at a few phase in the training periods and finds the component with the good suit and returns the singular title or mark related with that phenomenal suit part.

FisherFaces Face Recognizer:

It is a drawn out rendition of EigenFaces face recognizer. It shows up at all the preparation appearances of each and every individual when it finds significant elements from every one of them consolidated. By taking pictures, the most important factors from all of them mixed, you are now not focusing on the elements that discriminate one character from the different, however the points that characterize all the humans in the education information as a whole.

Fisherfaces calculation, then again of extricating useful angles that imply all the faces of all the people, it extricates advantageous focuses that segregate one man or lady from the others. This way focuses on one man or lady don't rule over the others and you've got the aspects that segregate one person from the others.



Prepared to see that centers removed genuinely describe faces and these countenances are implied as fisher appearances and accordingly the character of the estimation.

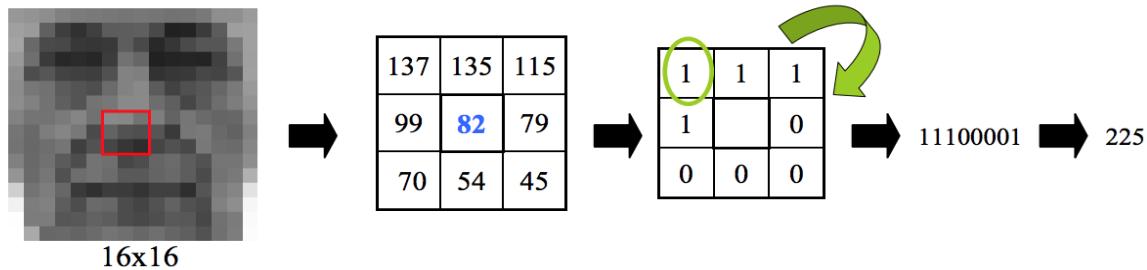
One perspective to observe here is that to be sure in Fisherfaces estimation if more than one men and women have pictures with sharp changes because of outside sources like smooth they will administer over different parts and affect comprehension precision.

Local Binary Patterns Histograms (LBPH) Face Recognizer:

We get it that Eigenfaces and Fisherfaces are each affected by using smooth and in genuine ways of life we won't guarantee amazing delicate circumstances. LBPH go up against recognizer is an improvement to beat this disadvantage.

Thought is to never again show up at the photo as a total substitute to find the close by components of a picture. LBPH calculation endeavors to find the local state of an image and it does that through assessing each pixel with its adjoining pixels.

Take a 3x3 window and go it one picture, at each cross (each close by period of a picture), assess the pixel at the center with its neighbor pixels. The neighbors with force cost considerably less than or equivalent to the center pixel are signified through 1 and others by utilizing zero Then you concentrate on these 0/1 qualities under a 3x3 window in a clockwise request and you will have a twofold example like 11100011 and this example is neighborhood to some area of the picture. You do this on the all out photo and you will have a posting of neighborhood twofold examples.



After we get a posting of neighboring paired designs, we convert each double example into a decimal reach and afterward make a histogram of these qualities.

So in the stop you will have one histogram for each face photograph in the schooling data set. That limit, on the off chance that there have been 100 pictures in the training data set, LBPH will extricate 100 histograms after schooling and save them for later acknowledgment. Keep in mind, this calculation additionally monitors which histogram has a place with which individual.

While doing acknowledgment, when somebody enters another image to that recognizer, then it will produce another histogram for that new face picture, then assess that recently created histogram with the generally produced histograms it has, find the outstanding solid histogram and return the man or lady name related with that pleasant in shape histogram.

Chapter 5

Facial Landmarks on Human Face

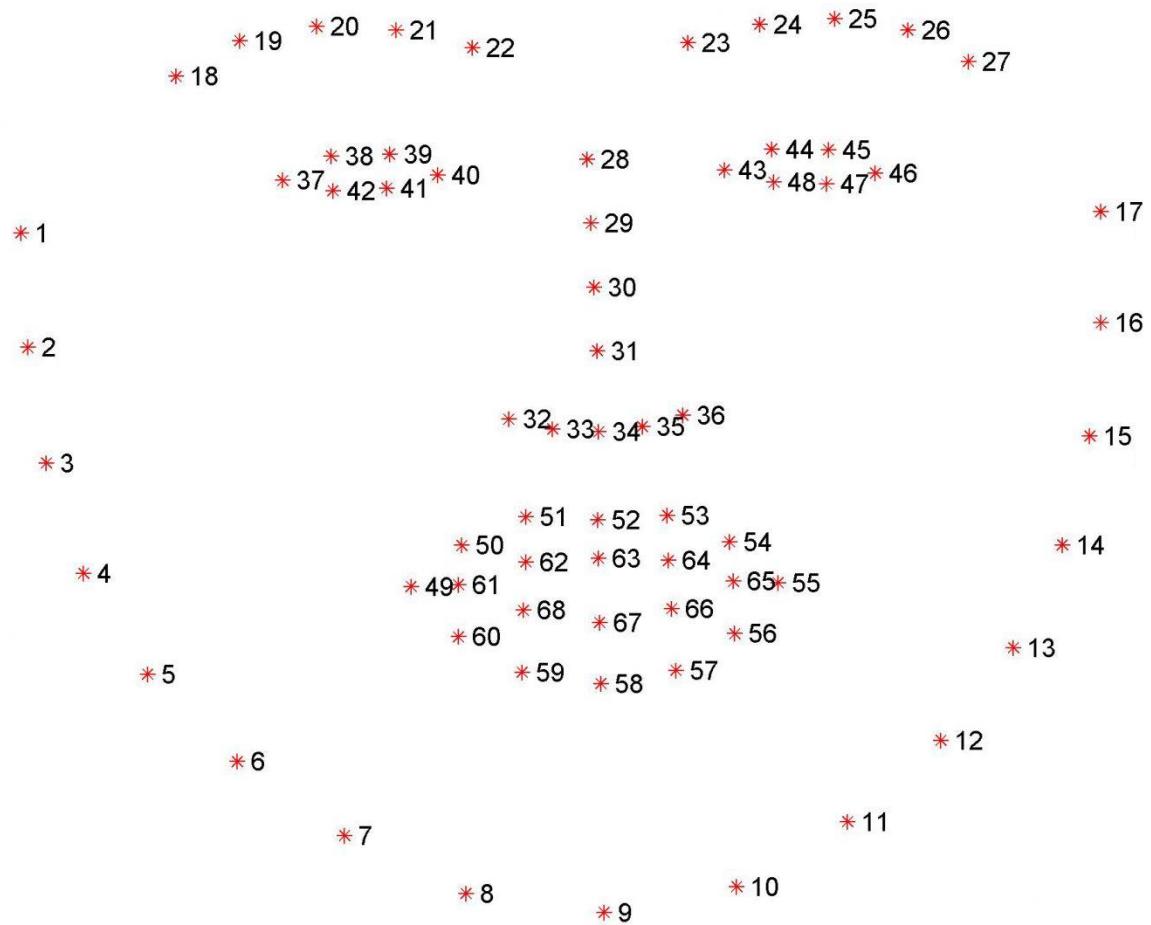
Understanding the human face now not solely helps in facial attention however finds purposes in facial morphing, head pose detection, and digital makeovers. If you are an ordinary person of social media apps like Instagram or Snapchat, have you questioned how the filters in shape flawlessly for every face? Though each and every face on the planet is unique, these filters appear to magically align on your nose, lips, and eyes. These filters or face-swapping purposes make use of facial landmarks. These landmarks are essentially factors that are intended to assist with the identification of the distance between eyes, role of the nose, measurement of the lips, etc. In the context of facial landmarks, our aim is to realize essential facial constructions on the face the use of form prediction methods.

Popular types of landmark detectors:

The Dlib library is the most famous library for detecting landmarks in the face. There are two sorts of detectors in this library.

5 point landmark detector: To make matters quicker than the sixty eight factor detector, dlib added the 5 factor detector which assigns two factors for the corners of the left eye, two factors for the proper eye, and one factor for the nose. This detector is most often used for the alignment of faces.

68-point landmark detectors: This pre-trained landmark detector identifies sixty eight factors ((x,y) coordinates) in a human face. These factors localize the area round the eyes, eyebrows, nose, mouth, chin, and jaw.



Representation of Face parts -

Point 01 - Point 17: Outline of face lower part

Point 18 - Point 22: Left Eyebrow Outline

Point 23 - Point 27: Right Eyebrow Outline

Point 28 - Point 36: Nose Outline

Point 37 - Point 42: Left Eye Outline

Point 43 - Point 48: Right Eye Outline

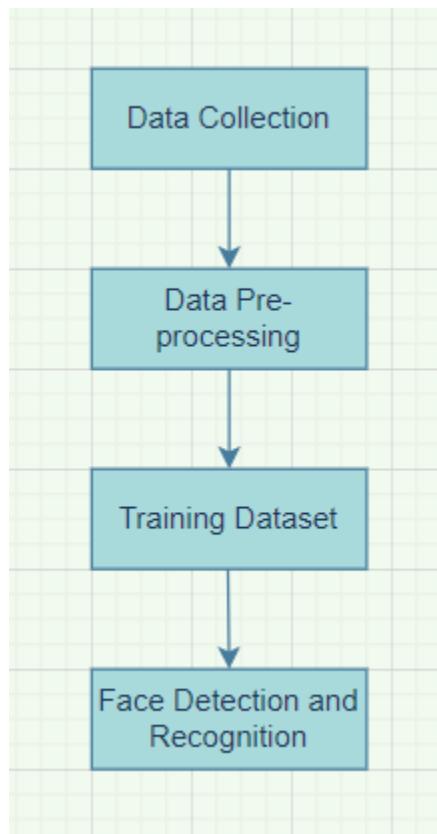
Point 49 - Point 68: Lips Outline

Chapter 6

Methodology

In this segment, we will talk about the various stages through which the framework goes through prior to giving the result to the client. This gives an outline about how the framework functions and how the result is given.

The flow chart shown is the overall mapping of our project. The flow chart gives us ideas about the flow of our project and the stages involved in it. The program works in three stages: the face detection and gathering stage, the data training stage, and the face recognition stage.



6.1 Import Required Modules:

Prior to beginning with the genuine coding part, we need to introduce the expected modules for the coding purposes. So let us incorporate them first.

cv2: This is an OpenCV module to be utilized in Python language which will be utilized for discovery and acknowledgment of countenances.

os: This Python module will be utilized to concentrate on the schooling catalogs and record names in the working framework.

numpy: This module will be utilized to change over records in Python into numpy exhibits as OpenCV face recognizers are given numpy clusters.

```
import cv2  
import os  
import numpy as np  
import pillow  
  
# pip3 install django  
# Database: sqlite3
```

6.2 Camera Setup:

```
import cv2 # This is to import opencv  
  
cam = cv2.VideoCapture(0) # This line is to start camera  
  
while(True):  
    ret, img = cam.read()  
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)  
    cv2.imshow('image', img)  
  
    k = cv2.waitKey(100) & 0xff # Press 'ESC' for exiting video  
    cam.release()  
    cv2.destroyAllWindows()
```

6.3 Face Detection and Creating Dataset:

XML preparing documents for Haar overflow are put away in FaceRecognizer/Cascades envelope. First the necessary XML classifier will be stacked. Then we will stack our entered picture in the dim scale mode. Numerous activities in OpenCV are done in grayscale.

```
face_detector = cv2.CascadeClassifier('FaceRecognizer/Cascades/haarcascade_frontalface_default.xml')

# For each person, enter one numeric face id

ret, img = self.video.read()
if img is not None:
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    faces = face_detector.detectMultiScale(gray, 1.3, 5)
    for (x, y, w, h) in faces:
        cv2.rectangle(img, (x, y), (x + w, y + h), (255, 0, 0), 2)
        cv2.imwrite("FaceRecognizer/dataset/User." + str(face_id) + '.' + str(self.count) + ".jpg",
                   gray[y:y + h, x:x + w])
        self.count += 1

    ret, jpeg = cv2.imencode('.jpg', img)
```

detectMultiScale Parameter Details:

This capacity recognizes the countenances in a given test picture and following are subtleties of its choices.

detectMultiScale: It is a typical element which recognizes the articles. Since it is being approached the face overflow, that is the very thing that it additionally recognizes. Grayscale picture is desirable over distinguish the appearances.

scaleFactor: Since certain countenances could likewise be closer to the camera, they would appear more prominent than these countenances toward the back. The scale component makes up for this.

minNeighbors: The location calculation utilizes a moving window to notice objects. This boundary characterizes the number of articles that are identified near the contemporary one sooner than it reports the face found.

Here is the pattern of our dataset:



6.4 Training Dataset:

This is the second phase of this project. In this we are going to train our model using the images collected from the different persons. Each person has given a separate id so that our LBPH Algorithm can understand a different person.

Haar Cascade and LBPH usage:

We will store all the data or images of a person in a folder named dataset and give the path of it in required files.

```
path = 'FaceRecognizer/dataset' # Path for face image database

recognizer = cv2.face.LBPHFaceRecognizer_create()
detector = cv2.CascadeClassifier("FaceRecognizer/Cascades/haarcascade_frontalface_default.xml")
```

For training the dataset we need to use the PIL library which we have already installed as [pip3 install pillow](#) and for the array we are using numpy array.

```

# function to get the images and label data

def getImagesAndLabels(path):

    imagePaths = [os.path.join(path, f) for f in os.listdir(path)]
    faceSamples = []
    ids = []

    for imagePath in imagePaths:

        PIL_img = Image.open(imagePath).convert('L') # convert it to grayscale
        img_numpy = np.array(PIL_img, 'uint8')

        id = int(os.path.split(imagePath)[-1].split(".")[1])
        faces = detector.detectMultiScale(img_numpy)

        for (x, y, w, h) in faces:
            faceSamples.append(img_numpy[y:y + h, x:x + w])
            ids.append(id)

    return faceSamples, ids

faces, ids = getImagesAndLabels(path)
recognizer.train(faces, np.array(ids))

# Save the model into trainer/trainer.yml
recognizer.save('FaceRecognizer/trainer/trainer.yml')

```

6.5 Face Recognition:

This is the final step or phase of this project. In this step we are going to use our trained model and try to recognise the person.

If the person already has their data stored in the dataset then our model will recognise that person and show their name on the top of the rectangle box and if not, then “unknown” will be shown on the top of the box.

We have also added the confidence which shows how much our face is matching with the images in the dataset.

```
font = cv2.FONT_HERSHEY_SIMPLEX

names = list(Person.objects.all().values_list('name', flat=True))

# set video width and height
self.video.set(3, 1500)
self.video.set(4, 1000)

# Define min window size to be recognized as a face
minW = 0.1 * self.video.get(3)
minH = 0.1 * self.video.get(4)

ret, img = self.video.read()
```

```

ret, img = self.video.read()

if img is not None:

    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

    faces = faceCascade.detectMultiScale(
        gray,
        scaleFactor=1.2,
        minNeighbors=5,
        minSize=(int(minW), int(minH)),
    )

    for (x, y, w, h) in faces:
        cv2.rectangle(img, (x, y), (x + w, y + h), (255, 0, 0), 2)

        id, confidence = recognizer.predict(gray[y:y + h, x:x + w])

        # Check if confidence is less them 100 ==> "0" is perfect match
        if 100 - confidence >= 55:
            id = names[id]
            confidence = " {0}%".format(round(100 - confidence))

        else:
            id = "unknown"
            confidence = " {0}%".format(round(100 - confidence))

        cv2.putText(img, str(id), (x + 5, y - 5), font, 2, (0, 255, 0), 3)
        cv2.putText(img, str(confidence), (x + 5, y + h - 5), font, 1, (255, 255,
255), 1)

```

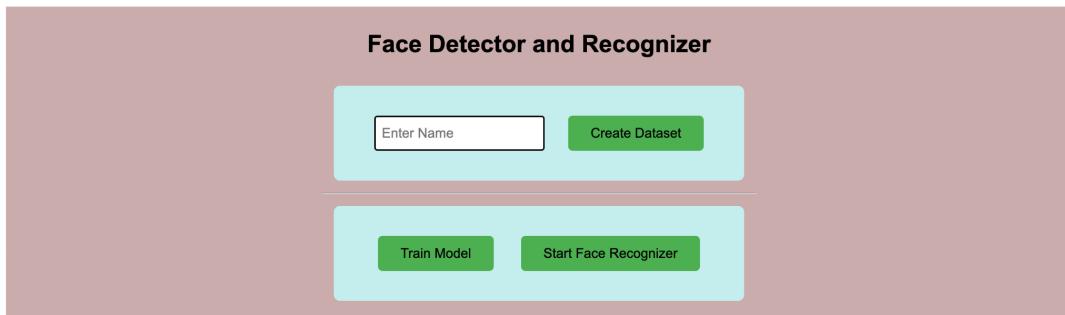
6.6 Flow of Working Project:

We have used **Django Framework** and **HTML/CSS** to integrate the frontend and backend in this project.

- Steps to use the project:

1. Create the dataset of your face by entering your name and clicking on **Create Dataset** button.
2. It will start the camera and will take 800 images after detecting your face and save it in *dataset* folder. Try to rotate your face to every possible angle for gathering diverse data.
3. Then, click on **Train Model**.
4. This process will take a few minutes to train the model on the newly added dataset. It will store the model *trainer.py* in *trainer* folder
5. Finally, click on **Start Face Recognizer**.
6. This process will start the camera. It will detect the face and will show the name of the person associated with that face.

- Homepage:



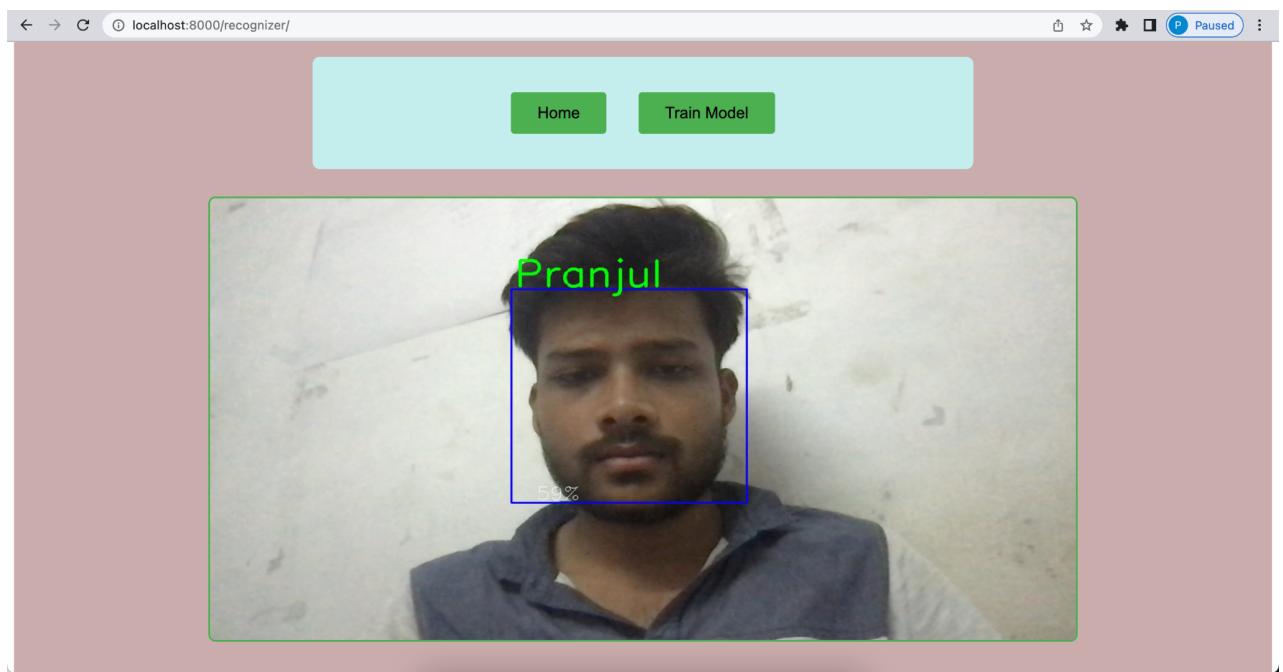
- Train Model Result Page:



- Create Dataset Page:



- Face Recognition Page:



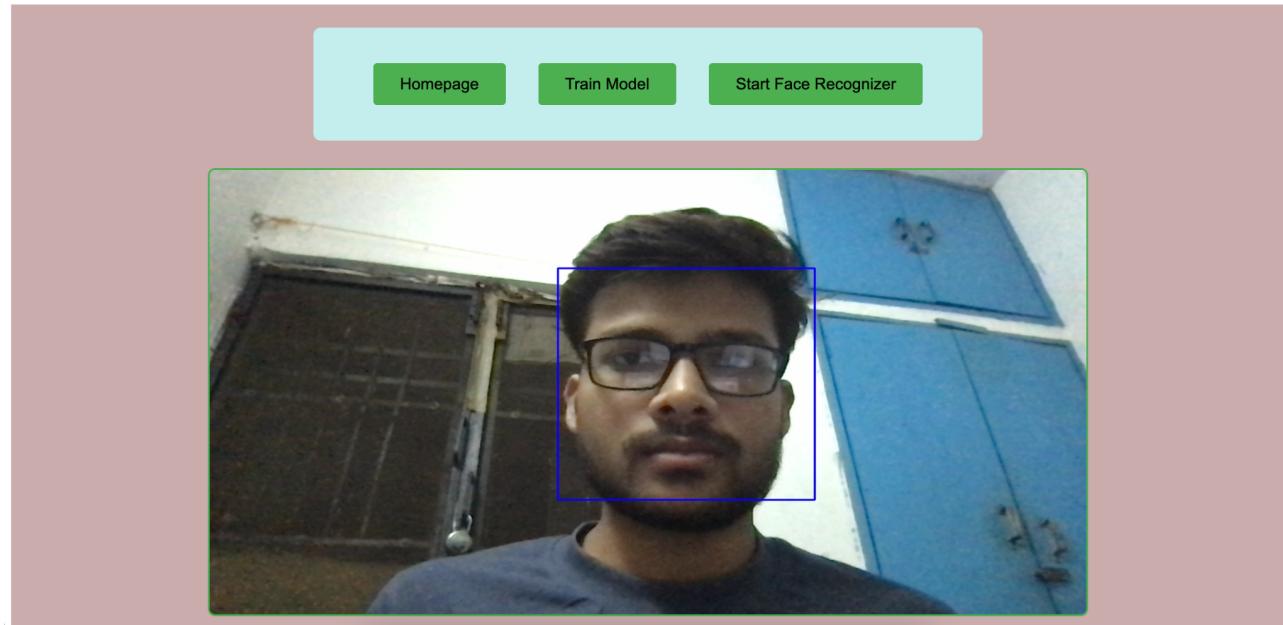
Chapter 7

Results

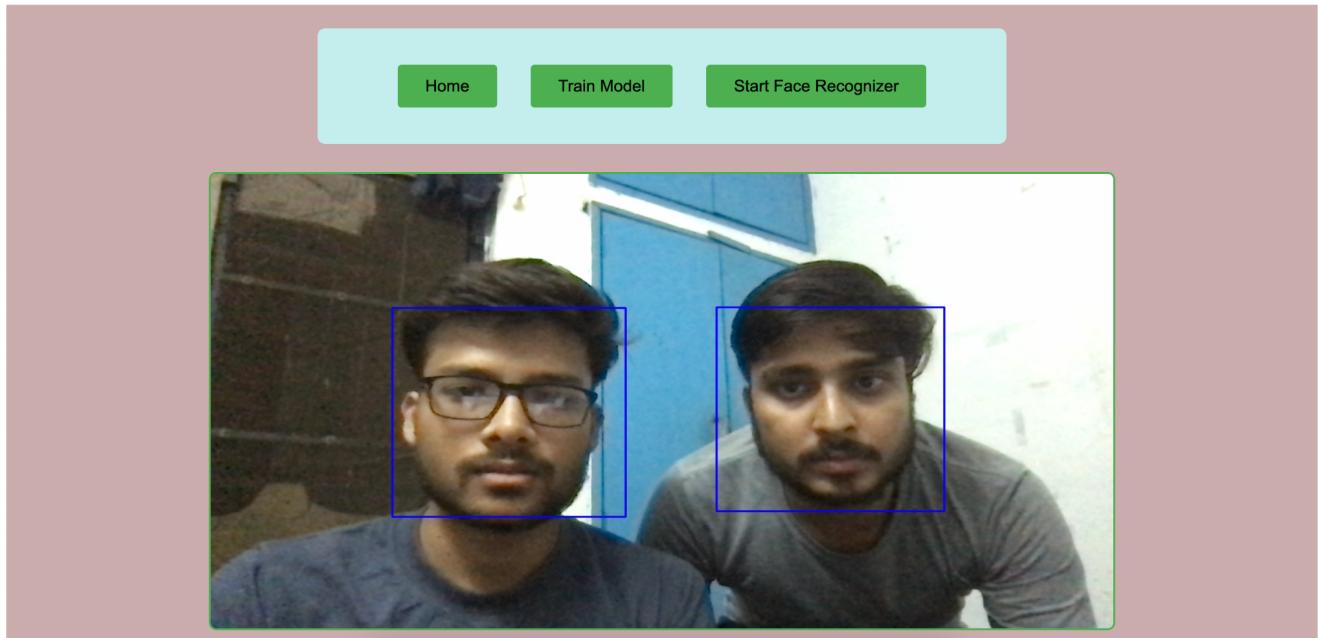
7.1 Face Detection Results:

The HAAR Cascade classifier is capable of detecting single as well as multiple faces in a single image at real-time.

- **Detecting single face:**

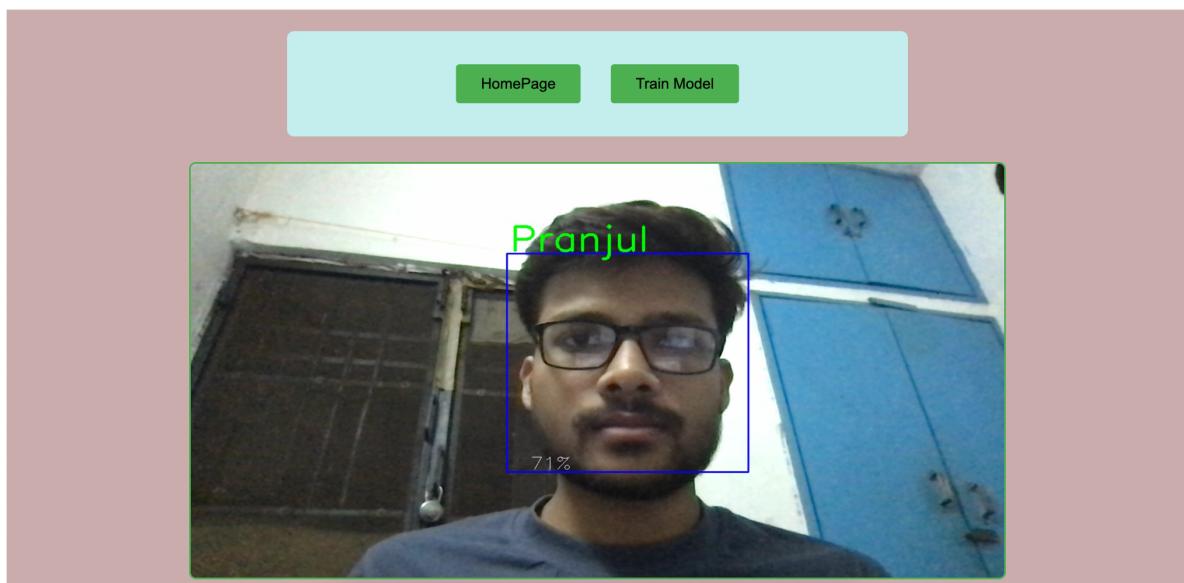


- Detecting multiple faces:

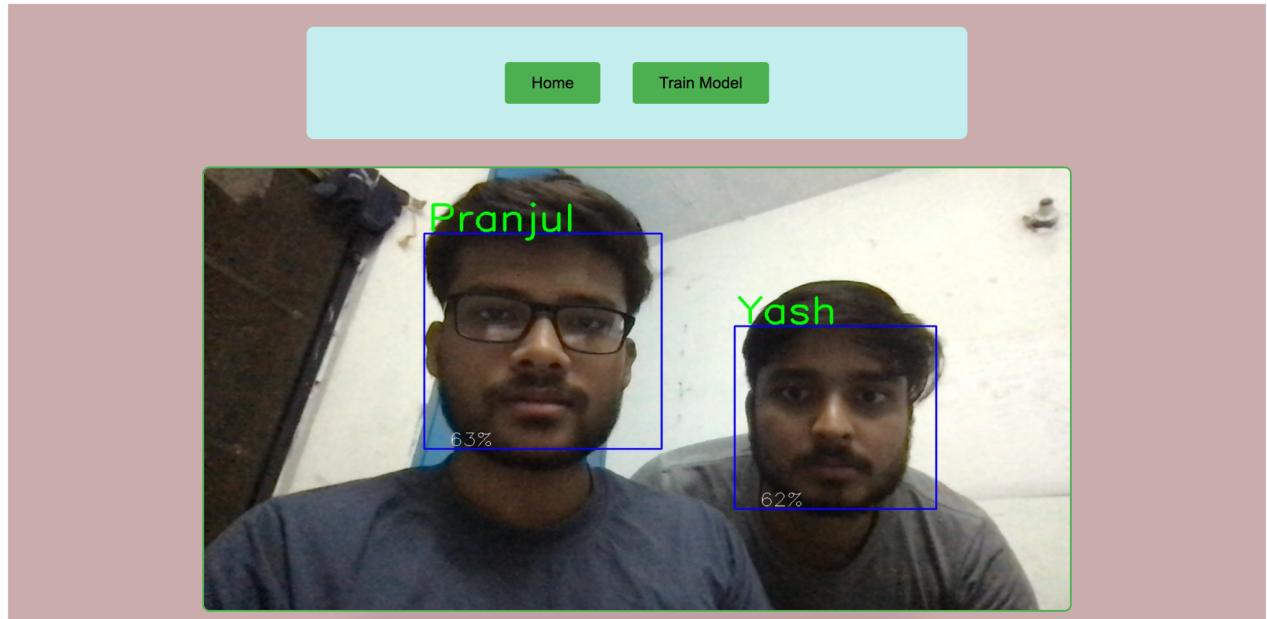


7.2 Face Recognition Results:

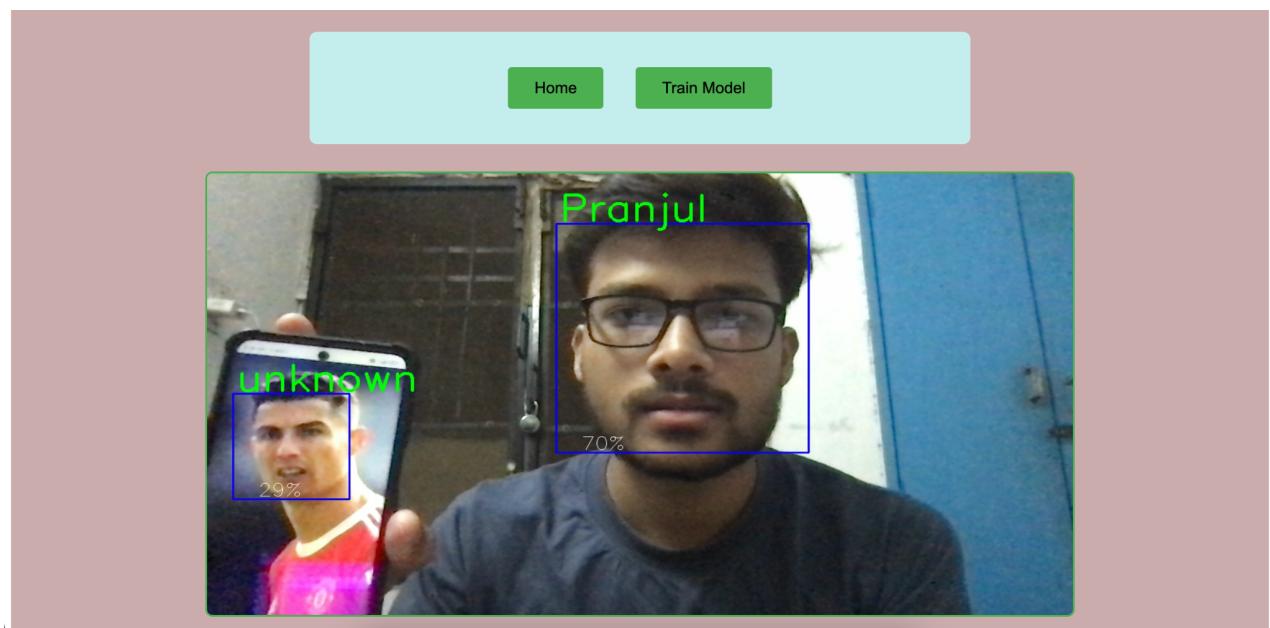
- Model recognising single person: Model is already trained at this face. The name of the person is shown with the confidence level up to 100%.



- **Model recognising multiple persons:** The model is trained for both these faces. Hence, it is showing the name of both persons.



- **Model recognising multiple persons:** Model is not trained for the face of person on the left. Hence, showing '*unknown*' for him.



Chapter 8

Conclusion

The venture basically centers around distinguishing and perceiving an individual in a camera. The program takes an ongoing video as its feedback and gives the names of individuals, alongside the exactness of the model, who are found in the video. We expect that the ongoing video is being taken in a sufficiently bright climate. The client ought to give his face information to the program previously, really at that time will the program perceive the individual.

We are intending to add the component that the actual program gains from the information video when it tracks down a renewed individual and marks it with a name. So when the following time, the individual is found in the video or any future recordings the program can perceive that individual all alone. With this component, the client doesn't have to prepare the machine however it will prepare without help from anyone else. This will make the existence of the client a lot simpler while utilizing this program. The spotlight will likewise be on decreasing the quantity of misleading up-sides and making the program hearty for any sort of climate.

Chapter 9

References

- Salihbašić, A., & Orehovački, T. (2019, May). Development of android application for gender, age and face recognition using opencv. In 2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO) (pp. 1635-1640). IEEE
- Jagtap, A. M., Kangale, V., Unune, K., & Gosavi, P. (2019, February). A Study of LBPH, Eigenface, Fisherface and Haar-like features for Face recognition using OpenCV. In 2019 International Conference on Intelligent Sustainable Systems (ICISS) (pp. 219-224). IEEE
- Balachandran, B., Saad, K. F., Patel, K., & Mekhiel, N. (2019, December). Parallel Computer for Face Recognition Using Artificial Intelligence. In 2019 14th International Conference on Computer Engineering and Systems (ICCES) (pp. 158-162). IEEE
- Adusumalli, H., Kalyani, D., Sri, R. K., Pratapteja, M., & Rao, P. P. (2021, February). Face Mask Detection Using OpenCV. In 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV) (pp. 1304-1309). IEEE
- Apoorva, P., Impana, H. C., Siri, S. L., Varshitha, M. R., & Ramesh, B. (2019, March). Automated criminal identification by face recognition using open computer vision classifiers. In 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC) (pp. 775-778). IEEE
- T Ojala, M Pietikainen and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary pattern [J]", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 7, pp. 971-987, 2002.

- B. Mandal, "Face recognition: Perspectives from the real world [C]", IEEE 2016 14th International Conference on Control Automation Robotics and Vision (ICARCV), pp. 1-5, 13-15 Nov. 2016.
- https://www.researchgate.net/publication/355886757_Face_Detection_and_Recognition_Using_OpenCV
- https://en.wikipedia.org/wiki/Facial_recognition_system
- https://docs.opencv.org/3.4/db/d28/tutorial_cascade_classifier.html
- <https://medium.com/analytics-vidhya/haar-cascades-explained-38210e57970d>
- <https://www.kite.com/python/docs/numpy>