**Basics**

 Here, we are using the Python programming language for detecting faces in images and videos. It requires the installation of additional packages such as OpenCV. OpenCV stands for Open Source Computer Vision. In OpenCV,all the images are converted to or from numpy arrays. This makes it easier to integrate it with other librabries that uses numpy.

 After installing OpenCV, we need to install some specific librabries.

 Such as,

o dlib

o face\_recognition

 We also need a working webcam. .it includes some basic performance tweaks to make things to run faster.

o process each video frame at ¼ resolution

o only detect faces from every other frame.

 We will use Haar-cascade for face detection.

 For detect the face from the video,we need a **VideoCapture** object.We will learn about this further.

**Face Detection**

**Procedure:**

* Take an image then create a haar cascade classifier. It will contain the features of the face from the image. With the help of that code can determine the face.
* OpenCV will read the image and the features file. It will convert the image into numpy array. So it will search for the rows and column values of the face.
* Numpy ndarray(The face rectangle co-ordinates).It will display the image with the rectangular face box.

These are the main syntax , we use for the face detection from the image using webcam.

import cv2

#create a cascade classifier object

face\_classifier = cv2.CascadeClassifier("D:\EC\SEM 7\TP - TERM PROJECT\haarcascade\_frontalface\_default.xml")

#reading the image as it is

img = cv2.imread("D:\EC\SEM 7\TP - TERM PROJECT\first.jpg",0)

#reading the image as gray scale image

gray = cv2.cvtColor(img,cv2.COLOR\_BGR2GRAY)

#SEARCH the co-ordinates of the image

faces = face\_classifier.detectMultiScale(gray,1.3,5)

#scalefactor = 1.3

#minNeighbors = 5

#rectangular face box

for(x,y,w,h) in faces:

cv2.rectangle(img, (x,y),(x+w,y+h),(0,255,255),2)

print(type(faces))

print(faces)

**Using OpenCV open Webcam:**

import cv2

frameWidth = 640

frameHeight = 480

cap = cv2.VideoCapture(0)

cap.set(3,frameWidth)

cap.set(4,frameHeight)

while True:

sucess,img = cap.read()

cv2.imshow("Video",img)

if cv2.waitKey(1) & 0xFF == ord('q'):

break

**Main Code for the face detection through the webcam**

* With the help of this following code, computer stored or collect the faces(samples) through the webcam which we want to detect.

import cv2

import numpy as np

face\_classifier = cv2.CascadeClassifier("D:\EC\SEM 7\TP - TERM PROJECT\haarcascade\_frontalface\_default.xml")

def face\_extractor(img):

gray = cv2.cvtColor(img,cv2.COLOR\_BGR2GRAY)

faces = face\_classifier.detectMultiScale(gray,1.3,5)

if faces is():

return None

for(x,y,w,h) in faces:

cropped\_face = img[y:y+h, x:x+w]

return cropped\_face

cap = cv2.VideoCapture(0)

count = 0

while True:

ret, frame = cap.read()

if face\_extractor(frame) is not None:

count+=1

face = cv2.resize(face\_extractor(frame),(200,200))

face = cv2.cvtColor(face, cv2.COLOR\_BGR2GRAY)

file\_name\_path = 'D:\EC\SEM 7\TP - TERM PROJECT\First'+str(count)+'.jpg'

cv2.imwrite(file\_name\_path,face)

cv2.putText(face,str(count),(50,50),cv2.FONT\_HERSHEY\_COMPLEX,1,(0,255,0),2)

cv2.imshow('Face Cropper',face)

else:

print("Face not found")

pass

if cv2.waitKey(1)==13 or count==100:

break

cap.release()

cv2.destroyAllWindows()

print('Collecting samples complete!!!')