

# Assignment 6

Name: Kripa Karthik

Registration number: 19BEC0549

Develop a python code to detect any object using Haar cascade classifier. Objects that are detected using Haar cascade classifier are:

(1) Face, Eyes and Smile Detection

(2) Cat face Detection

1)

## PYTHON CODE

```
import cv2
import datetime

face_classifier=cv2.CascadeClassifier("haarcascade_frontalface_default.xml")
eye_classifier=cv2.CascadeClassifier("haarcascade_eye.xml")
smile_classifier=cv2.CascadeClassifier("haarcascade_smile.xml")

#It will read the first frame/image of the video
video=cv2.VideoCapture(0)

while True:

    #capture the first frame
    check,frame=video.read()

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

    #detect the faces from the video using detectMultiScale function
    faces=face_classifier.detectMultiScale(gray,1.3,5)
    eyes=eye_classifier.detectMultiScale(gray,1.3,5)
    smiles=smile_classifier.detectMultiScale(gray,1.3,5)

    print(faces)
```

```

#drawing rectangle boundaries for the detected face
for(x,y,w,h) in faces:
    cv2.rectangle(frame, (x,y), (x+w,y+h), (127,255,255), 2)
    cv2.imshow('Face detection', frame)
    cv2.putText(frame, 'Face',(x,y-20),cv2.FONT_HERSHEY_SIMPLEX,0.8,(0,255,0),2)
    picname=datetime.datetime.now().strftime("%Y-%m-%d-%H-%M")
    cv2.imwrite(picname+".jpg",frame)

#drawing rectangle boundaries for the detected eyes
for(ex,ey,ew,eh) in eyes:
    cv2.rectangle(frame, (ex,ey), (ex+ew,ey+eh), (127,0,255), 2)
    cv2.imshow('Face detection', frame)

#drawing rectangle boundaries for the detected smile
for (sx, sy, sw, sh) in smiles:
    cv2.rectangle(frame,(sx,sy),((sx+sw),(sy+sh)),(0,0,255),2)
    cv2.imshow('Smile detection', frame)

#waitKey(1)- for every 1 millisecond new frame will be captured
Key=cv2.waitKey(1)

if Key==ord('q'):
    #release the camera
    video.release()

    #destroy all windows
    cv2.destroyAllWindows()

    break

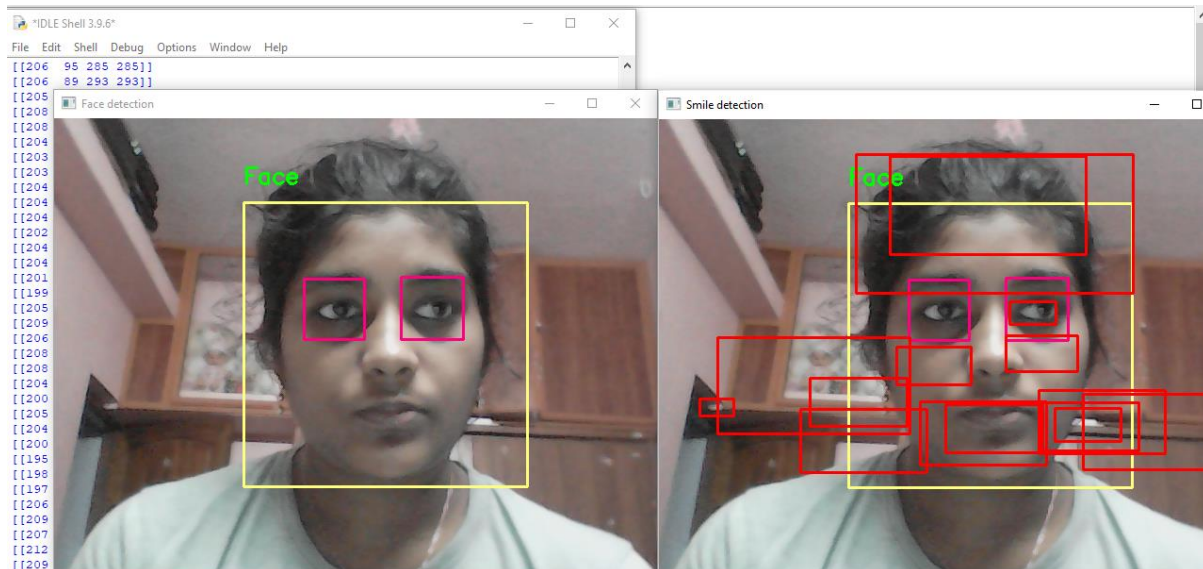
```

**SCREENSHOTS**

assignment6.py - C:/Users/Lenovo/Desktop/python programs IOT/face detection/assignment6.py (3.9.6)  
File Edit Format Run Options Window Help

```
import cv2
import datetime
face_classifier=cv2.CascadeClassifier("haarcascade_frontalface_default.xml")
eye_classifier=cv2.CascadeClassifier("haarcascade_eye.xml")
smile_classifier=cv2.CascadeClassifier("haarcascade_smile.xml")
#It will read the first frame/image of the video
video=cv2.VideoCapture(0)
while True:
    #capture the first frame
    check,frame=video.read()
    gray=cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    #detect the faces from the video using detectMultiScale function
    faces=face_classifier.detectMultiScale(gray,1.3,5)
    eyes=eye_classifier.detectMultiScale(gray,1.3,5)
    smiles=smile_classifier.detectMultiScale(gray,1.3,5)
    print(faces)

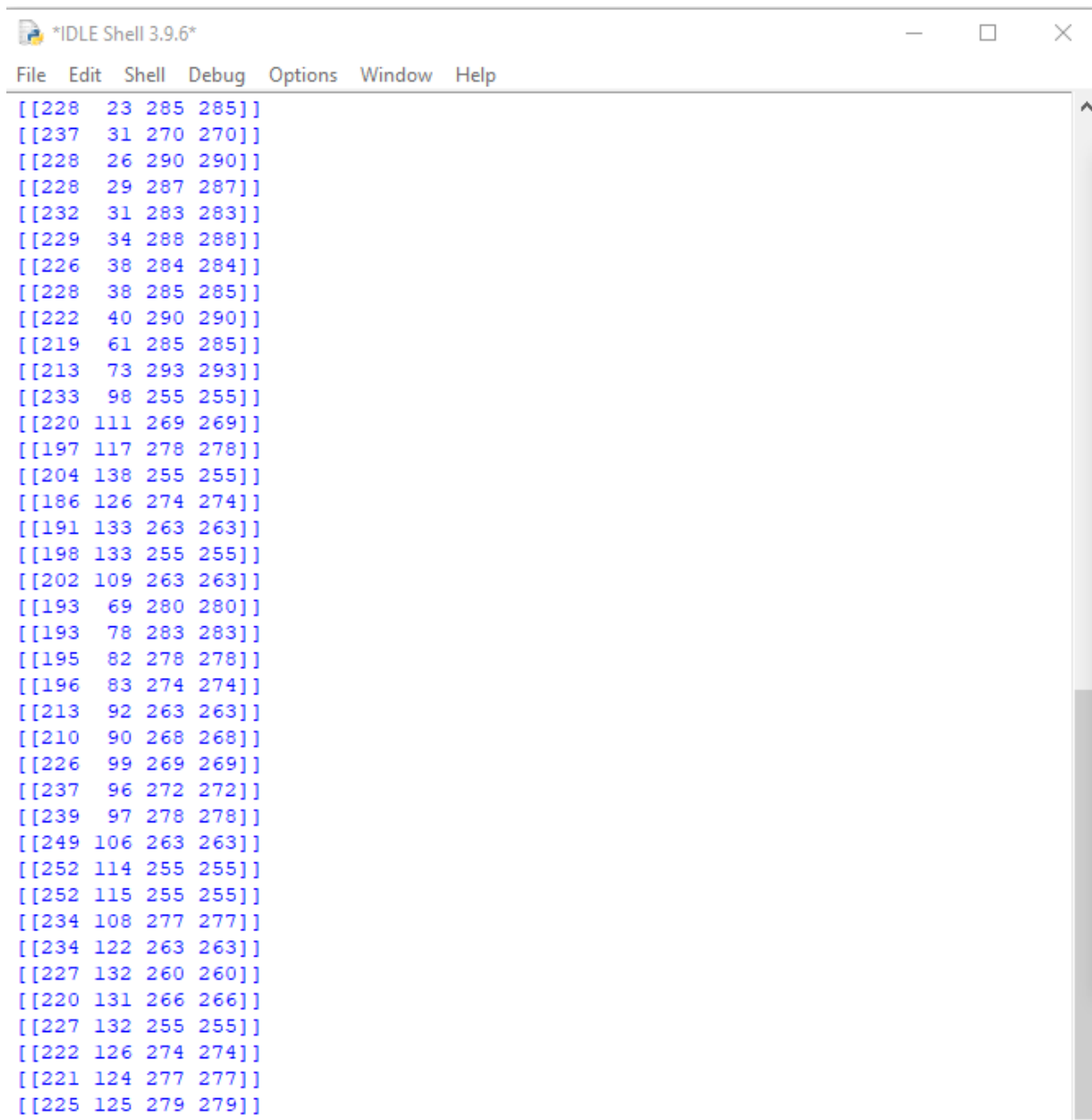
    #drawing rectangle boundaries for the detected face
    for(x,y,w,h) in faces:
        cv2.rectangle(frame, (x,y), (x+w,y+h), (127,255,255), 2)
        cv2.imshow('Face detection', frame)
        cv2.putText(frame, 'Face', (x,y-20),cv2.FONT_HERSHEY_SIMPLEX,0.8, (0,255,0),2)
        picname=datetime.datetime.now().strftime("%y-%m-%d-%H-%M")
        cv2.imwrite(picname+".jpg",frame)
    #drawing rectangle boundaries for the detected eyes
    for(ex,ey,ew,eh) in eyes:
        cv2.rectangle(frame, (ex,ey), (ex+ew,ey+eh), (127,0,255), 2)
        cv2.imshow('Face detection', frame)
    #drawing rectangle boundaries for the detected smile
    for (sx, sy, sw, sh) in smiles:
        cv2.rectangle(frame, (sx,sy), ((sx+sw), (sy+sh)), (0,0,255),2)
        cv2.imshow('Smile detection', frame)
    #waitKey(1)- for every 1 millisecond new frame will be captured
    Key=cv2.waitKey(1)
    if Key==ord('q'):
        #release the camera
        video.release()
        #destroy all windows
        cv2.destroyAllWindows()
        break
```



Yellow color-face

Pink-eyes

Red-smile



The screenshot shows a window titled "IDLE Shell 3.9.6" with a menu bar (File, Edit, Shell, Debug, Options, Window, Help). The main area contains a list of 40 coordinate pairs, each enclosed in double brackets. The pairs are as follows:

[[228	23	285	285]]
[[237	31	270	270]]
[[228	26	290	290]]
[[228	29	287	287]]
[[232	31	283	283]]
[[229	34	288	288]]
[[226	38	284	284]]
[[228	38	285	285]]
[[222	40	290	290]]
[[219	61	285	285]]
[[213	73	293	293]]
[[233	98	255	255]]
[[220	111	269	269]]
[[197	117	278	278]]
[[204	138	255	255]]
[[186	126	274	274]]
[[191	133	263	263]]
[[198	133	255	255]]
[[202	109	263	263]]
[[193	69	280	280]]
[[193	78	283	283]]
[[195	82	278	278]]
[[196	83	274	274]]
[[213	92	263	263]]
[[210	90	268	268]]
[[226	99	269	269]]
[[237	96	272	272]]
[[239	97	278	278]]
[[249	106	263	263]]
[[252	114	255	255]]
[[252	115	255	255]]
[[234	108	277	277]]
[[234	122	263	263]]
[[227	132	260	260]]
[[220	131	266	266]]
[[227	132	255	255]]
[[222	126	274	274]]
[[221	124	277	277]]
[[225	125	279	279]]

## 2) Python code to detect Cat face using Haarcascades

### PYTHON CODE

```
import cv2
```

```
#Read video file from the folder by giving its path
```

```
video = cv2.VideoCapture(r'C:\Users\Lenovo\Desktop\python programs IOT\face  
detection')
```

```
face_cascade = cv2.CascadeClassifier('haarcascade_frontalcatface.xml')
```

**while True:**

**ret, img = video.read()**

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

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

**print(faces)**

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

**# To draw a rectangle in a face**

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

**roi\_gray = gray[y:y+h, x:x+w]**

**roi\_color = img[y:y+h, x:x+w]**

**cv2.imshow('Cat', img)**

**k = cv2.waitKey(30) & 0xff**

**if k == 27:**

**break**

**video.release()**

**cv2.destroyAllWindows()**

**SCREENSHOTS**

assignment6b.py - C:/Users/Lenovo/Desktop/python programs IOT/face detection/assignment6b.py (3.9.6)

File Edit Format Run Options Window Help

```
import cv2
#Read video file from the folder by giving its path
video = cv2.VideoCapture(r'C:\Users\Lenovo\Desktop\python programs IOT\face detection')
face_cascade = cv2.CascadeClassifier('haarcascade_frontalcatface.xml')
while True:
    ret, img = video.read()
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    faces = face_cascade.detectMultiScale(gray, 1.3, 5)
    print(faces)

    for (x,y,w,h) in faces:
        # To draw a rectangle in a face
        cv2.rectangle(img, (x,y), (x+w,y+h), (0,0,255),2)
        roi_gray = gray[y:y+h, x:x+w]
        roi_color = img[y:y+h, x:x+w]
        cv2.imshow('Cat', img)

    k = cv2.waitKey(30) & 0xff
    if k == 27:
        break
video.release()
cv2.destroyAllWindows()
```

