

Face Recognition and Comparison

Rui Carapinha
Adrian Galeziowski
Wroclaw Univesity of Technology
Politechnika Wroclawska

23/01/2019

1 Introduction

My project is Face Recognition and Comparison, the goal of the project is for the camera to recognize the faces and compare it when other faces. The program can take pictures of the face and then we can compare the face with another images with a certain degree of certainty.

2 Software

The software I used was the following:

- Python 2.7
- OpenCV

2.1 Coding

In this part of the report I'll explain, part by part, how the code works. First, I'll start with the access to the camera:

Listing 1: Camera

```
import cv2

def show_webcam():
    cam = cv2.VideoCapture(0)
    while True:
        ret_val, img = cam.read()
        cv2.imshow('Webcam', img)
        if cv2.waitKey(1) == 27:
            break
    cam.release()
    cv2.destroyAllWindows()

def main():
    show_webcam()

main()
```

This code only access the camera and when the key "Space" is pressed the program closes.

The next step of the project was to detect the face and draw a blue rectangle around it. The code is the following:

Listing 2: Face Detection

```
import cv2
import numpy as np

faceCascade=cv2.CascadeClassifier('/home/carapinha/Desktop
/Face/haarcascade_frontalface_default.xml')
cam = cv2.VideoCapture(0)
while True:
    ret_val, img = cam.read()
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    faces = faceCascade.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)
        roi_gray = gray[y:y+h, x:x+w]
        roi_color = img[y:y+h, x:x+w]

    cv2.imshow('Webcam', img)
    if cv2.waitKey(1) == 27:
        break
cam.release()
cv2.destroyAllWindows()
```

This code detects the face using a classifier called Haar Cascade, this is an effective object detection method. The next step was to take a Snapshot only of the face:

Listing 3: Photo

```
import numpy as np
import cv2
import time

face_cascade=cv2.CascadeClassifier('haarcascade_frontalface_
_default.xml')

def TakeSnapshotAndSave():
    cap = cv2.VideoCapture(0)

    num = 0
    while num<2:
        # Capture frame-by-frame
        ret, frame = cap.read()

        # to detect faces in video
        gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
        faces = face_cascade.detectMultiScale(gray, 1.3, 5)

        for (x,y,w,h) in faces:
            cv2.rectangle(frame,(x,y),(x+w,y+h),(255,0,0),2)
            roi_gray = gray[y:y+h, x:x+w]
            roi_color = frame[y:y+h, x:x+w]

        x = 0
        y = 20
        text_color = (0,255,0)

        cv2.imwrite('opencv'+str(num)+'.jpg',frame)
        num = num+1

    # When everything done, release the capture
    cap.release()
    cv2.destroyAllWindows()

if __name__ == "__main__":
    TakeSnapshotAndSave()
```

This code takes two shots and saves them in a .jpg file, named opencv1.jpg and opencv2.jpg. The final code is:

Listing 4: Face Recognition and Comparison - Part 1

```
import cv2
import glob
import numpy as np
from itertools import izip
from PIL import Image
import sys
from scipy.misc import imread
from scipy.linalg import norm
from scipy import sum, average

def Hist():
    In1 = raw_input("1_Image_Name->")
    i1 = Image.open(In1 + ".jpg")

    In2 = raw_input("2_Image_Name->")
    i2 = Image.open(In2 + ".jpg")

    width1, height1 = i1.size
    width2, height2 = i2.size

    if width1 <= width2:
        width = width1
    else:
        width = width2

    if height1 <= height2:
        height = height1
    else:
        height = height2

    I1 = i1.resize((width,height))
    I2 = i2.resize((width,height))

    h1 = I1.histogram()
    h2 = I2.histogram()
    SumIm1 = 0.0
    SumIm2 = 0.0
    diff = 0.0
    for i in range(len(h1)):
        SumIm1 += h1[i]
        SumIm2 += h2[i]
        diff += abs(h1[i] - h2[i])
        maxSum = max(SumIm1, SumIm2)

    print "Differences(%) : ", (diff/(2*maxSum))*100
```

Listing 5: Face Recognition and Comparison - Part 2

```
def Snapshot(img, num, nm):
    cv2.imwrite(nm+".jpg",img)

def main():
    faceCascade = cv2.CascadeClassifier('/home/carapinha
/Desktop/Face/haarcascade_frontalface_default.xml')
    cam = cv2.VideoCapture(0)
    num = 0;
    while True:
        ret_val, img = cam.read()
        gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
        faces=faceCascade.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)

            roi_color = img[y:y+h, x:x+w]

        cv2.imshow('Webcam', img)
        if cv2.waitKey(1) == 27:
            break
        if cv2.waitKey(33) == ord('s'):
            Name = raw_input("Save_as_s->")
            Snapshot(roi_color, num, Name)
            print "Image_sSaved_s->" + str(num)
            num = num + 1
        if cv2.waitKey(33) == ord('a'):
            print(glob.glob('*.jpg'))
            Hist()

    cam.release()
    cv2.destroyAllWindows()

if __name__ == "__main__":
    main()
```

This code is a merged version of all the other codes and still has the ability to compare images. To compare the faces, I compare the histogram of both pictures

3 Development

In this part of the report I will explain the development I made through these months of the project. First, I splitted the project in several objectives. They were the following:

- Access the camera
- Draw Rectangle in the Face
- Take a screenshot of the Face and save it to a .jpg file

- Compare two pictures

The code I made from each one of this parts is already shown and explained above.

The last results I had for the Face Comparison were quite nice. I had 3 photos (me, my roommate and my father):



Figure 1: Me, Roommate and Father

When I compare these photos the values I had were the following:

1st Image	2nd Image	Differences (%)
Mine	Father	28.72
Mine	Roommate	83.42
Roommate	Father	82.31

Figure 2: Tablue of Difference Values

4 Conclusion

In conclusion the project was well done and interesting results were achieved. We found an interesting way with some credability to compare two faces. It was also verty interesting to make this project because I never used OpenCV and Python in a project and it was a great experience to learn this two powerful tools.