PC-side face tracking

This chapter is similar to the previous chapter, we can optimize the Raspberry Pi face recognition function in the same way. In both experiments, the incremental PID algorithm and PCA9685 control the servo are consistent, **so the Raspberry Pi program remains the same.**

We only need to modify the PC program,as shown in the figure below.

```
from __future__ import division
import cv2
import time
import numpy as np
import socket

addr = ('192.168.1.111',7782)#Target host IP
readdr = ('192.168.1.110',7782)#Locla host IP
s = socket.socket(socket.AF_INET,socket.SOCK_DGRAM)
s.bind(readdr)

cap = cv2.VideoCapture(0)
cap.set(3, 640)
cap.set(4, 480)
face_cascade=cv2.CascadeClassifier('face.xml')

x=0;
while 1:
    ret,frame = cap.read()
    gray=cv2.cvtColor(frame,cv2.COLOR_BGR2GRAY)
    faces=face_cascade.detectMultiScale(gray)
    if len(faces)>0:
        for(x,y,w,h)in faces:
            cv2.rectangle(frame,(x,y),(x+h,y+w),(0,255,0),2)
            result=(x,y,w,h)
            x=result[0]
            y=result[1]

        print('x',x);
        data =str(x)+','+str(y)
        s.sendto(data.encode("utf-8"),addr)
    cv2.imshow("capture", frame)
    if cv2.waitKey(1)==119:
        break
cap.release()
cv2.destroyAllWindows()
s.close()
```