工具：Jupyter，内核为Python3.12

题目：使用openCV基于Haar级联分类器的方法，对一张图片进行人脸识别。将人脸用红色矩形框出，再用绿色矩形框出

我们首先导入opencv库和NumPy函数库

1. import cv2
2. import numpy as np

然后我们创建一个人脸识别器对象，并加载我们训练好的识别器（详情请见识别器训练说明）

1. recogizer=cv2.face.LBPHFaceRecognizer\_create()
2. recogizer.read("D:/trainer/trainer.yml")

接着我们定义一个名为FaceDetection的函数

1. def FaceDetection(img):

在函数内部，我们先用cv2.cvtColor函数将图片转化为灰度图像，以便减少计算量

1. gary = cv2.cvtColor(img,cv2.COLOR\_BGR2GRAY)

紧接着我们用cv2.CascadeClassifier函数导入Haar级联分类器，加载至FaceDetect这个变量中（注意修改地址）

1. FaceDetect = cv2.CascadeClassifier("C:/Users/33086/Downloads/opencv/sources/data/haarcascades\_cuda/haarcascade\_frontalface\_alt2.xml")

然后我们使用detectMultiScale函数来检测人脸，导入灰度图像，其他参数我们则保持默认

1. face = face\_detect.detectMultiScale(gary)

接着我们用cv2.rectangle函数在图片上画矩形，由于题目要求我们再框出个绿色矩形，我们这里将绿色矩形扩大以和红色矩形区分开来

1. **for** x,y,w,h in face:
2. cv2.rectangle(img,(x,y),(x+w,y+h),color=(0,0,255),thickness=1)
3. cv2.rectangle(img,(x-10,y-10),(x+w+10,y+h+10),color=(0,255,0),thickness=1)

然后我们用predict函数评估置信度

1. ids,confidence=recogizer.predict(gary[y:y+h,x:x+w])

然后我们根据置信评分来判断是否匹配，如果是就在矩形框上打出姓名

1. **if** confidence > 80:
2. cv2.putText(img, 'kiki', (x + 10, y - 10), cv2.FONT\_HERSHEY\_SIMPLEX, 0.75, (0, 255, 0), 1)

最后我们返回图像

1. cv2.imshow('result', img)

完整的函数为：

1. def FaceDetection(img):
2. gary = cv2.cvtColor(img,cv2.COLOR\_BGR2GRAY)
3. face\_detect = cv2.CascadeClassifier("C:/Users/33086/Downloads/opencv/sources/data/haarcascades\_cuda/haarcascade\_frontalface\_alt2.xml")
4. face = face\_detect.detectMultiScale(gary)
5. **for** x,y,w,h in face:
6. cv2.rectangle(img,(x,y),(x+w,y+h),color=(0,0,255),thickness=1)
7. ids,confidence=recogizer.predict(gary[y:y+h,x:x+w])#confidence为置信评分
8. **if** confidence > 80:
9. cv2.putText(img, 'kiki', (x + 10, y - 10), cv2.FONT\_HERSHEY\_SIMPLEX, 0.75, (0, 255, 0), 1)
10. cv2.imshow('result', img)

最后我们导入需要识别的图片（注意修改地址），然后调用FaceDetection函数

1. img=cv2.imread("D:/[@SVUV8T3)P]O3[T6)G7\_PE.png")
2. FaceDetection(img)
3. cv2.waitKey(0)
4. cv2.destroyAllWindows()

完整的代码为：

1. import cv2
2. import os
3. import numpy as np
4. recogizer=cv2.face.LBPHFaceRecognizer\_create()#创建一个人脸识别器对象
5. recogizer.read("D:/trainer/trainer.yml")#加载识别器
6. def FaceDetection(img):
7. gary = cv2.cvtColor(img,cv2.COLOR\_BGR2GRAY)
8. face\_detect = cv2.CascadeClassifier("C:/Users/33086/Downloads/opencv/sources/data/haarcascades\_cuda/haarcascade\_frontalface\_alt2.xml")
9. face = face\_detect.detectMultiScale(gary)
10. **for** x,y,w,h in face:
11. cv2.rectangle(img,(x,y),(x+w,y+h),color=(0,0,255),thickness=1)
12. ids,confidence=recogizer.predict(gary[y:y+h,x:x+w])#confidence为置信评分
13. **if** confidence > 80:
14. cv2.putText(img, 'kiki', (x + 10, y - 10), cv2.FONT\_HERSHEY\_SIMPLEX, 0.75, (0, 255, 0), 1)
15. cv2.imshow('result', img)
16. img=cv2.imread("D:/[@SVUV8T3)P]O3[T6)G7\_PE.png")
17. FaceDetection(img)
18. cv2.waitKey(0)
19. cv2.destroyAllWindows()