

Python使用OpenCV即時人臉辨識

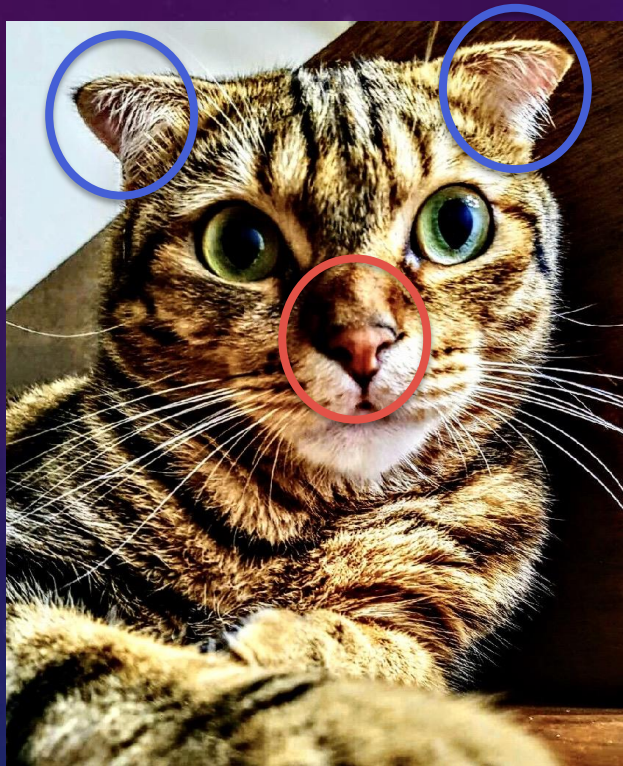
102-06.09.24.34

研究動機

近來有關新冠病毒的疫情越演越烈，而政府的應對政策也從原本的「有呼吸道症狀及出入醫療場所才須配戴口罩」，改成了「搭乘交通運輸工具及出入公共場所」都必須配戴口罩，而政府也在捷運站出入口都增設了人力控管。而利用人力控管的方式，人手上可能會有不足，若是在上下班尖峰時期，也可能有漏網之魚的可能性。因此我們想結合AI的使用，嘗試是否能做出辨識人臉的攝影系統，自動偵測到未戴口罩的乘客，並增加辨識上的精確度。

圖像辨識

我們之所以能夠辨別物體的不同，是因為我們可以透過學習該物體的特徵，例如狗與貓：

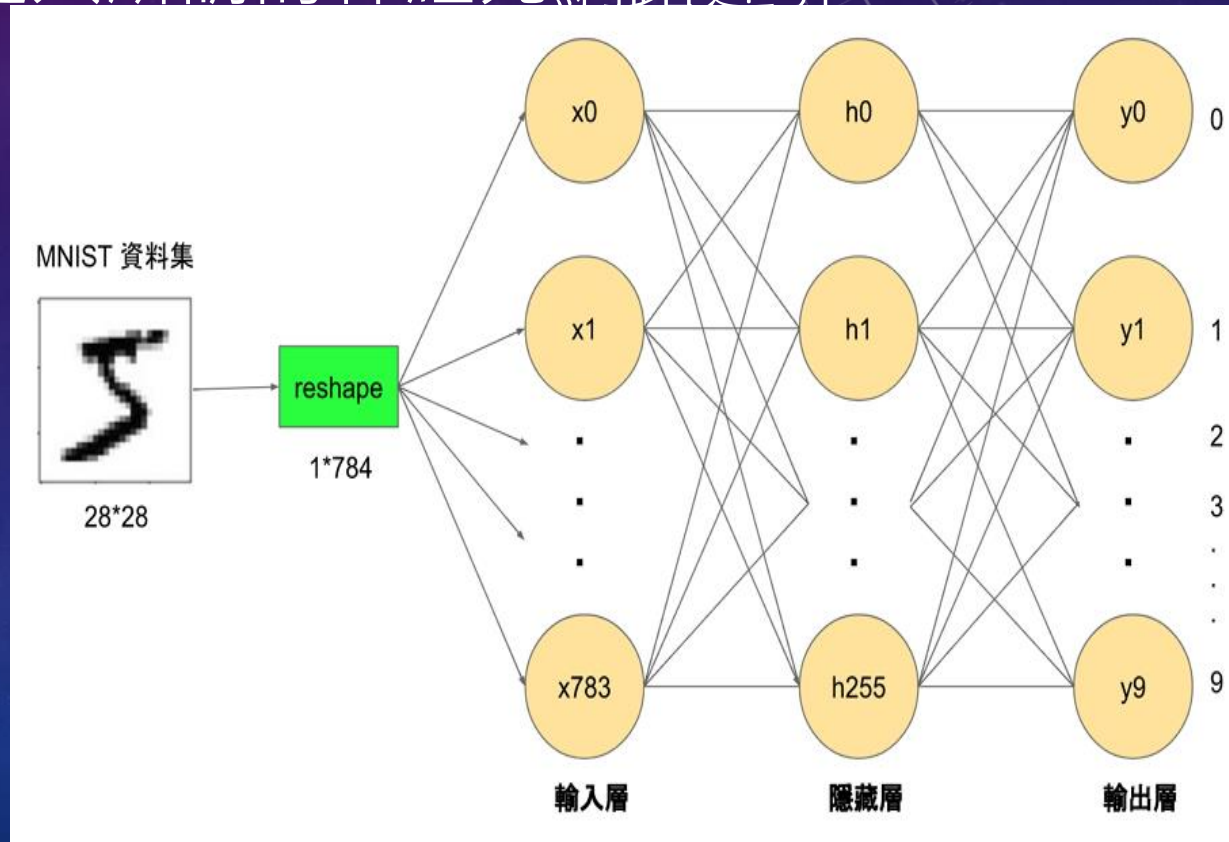
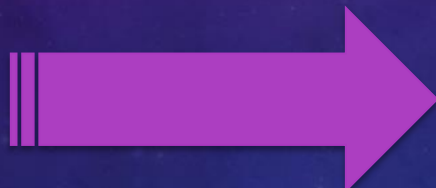


有著明顯的差距

我們需要學習，才能學會分辨物體，而電腦也是，但電腦要怎麼學習？

其實電腦並不是直接看著圖，而是將圖轉成二進制 (0, 1)，例如下圖，它就是把我們寫字時的白紙當作0，黑字當作1，來判斷數字的大小、位置等訊息，再進入所謂的神經元網路運算。

```
1 000000000000000000001111000000000000
2 00000000000000000000111111000000000000
3 000000000000000000001111111000000000000
4 000000000000000000001111111000000000000
5 000000000000000000001111111000000000000
6 000000000000000000001111111000000000000
7 000000000000000000001111111000000000000
8 000000000000000000001111111000000000000
9 000000000000000000001111111000000000000
10 0000000000000000000011111110000000000000
11 0000000000000000000011111110000000000000
12 0000000000000000000011111110000000111100000
13 0000000011111110000000111111000000
14 0000000011111110000000111111000000
15 0000000011111110000000111111000000
16 00000011111110000000111111000000
17 00000011111110000000111111000000
18 00000011111111111111111111000000
19 0000111111111111111111111111000000
20 0000111111111111111111111111000000
21 0000011111111111111111111111000000
22 0000011111111111111111111111000000
23 00000111111111111111111110000000
24 0000000011111111111111111110000000
25 0000000000000000111111100000000000
26 0000000000000000111111100000000000
27 0000000000000000111111100000000000
28 0000000000000000111110000000000000
29 00000000000000000000111111000000000000
30 000000000000000000001111111000000000
31 00000000000000000000111111100000000000
32 00000000000000000000111111000000000000
```



▲二進制化的數字

▲神經元網路

視覺庫

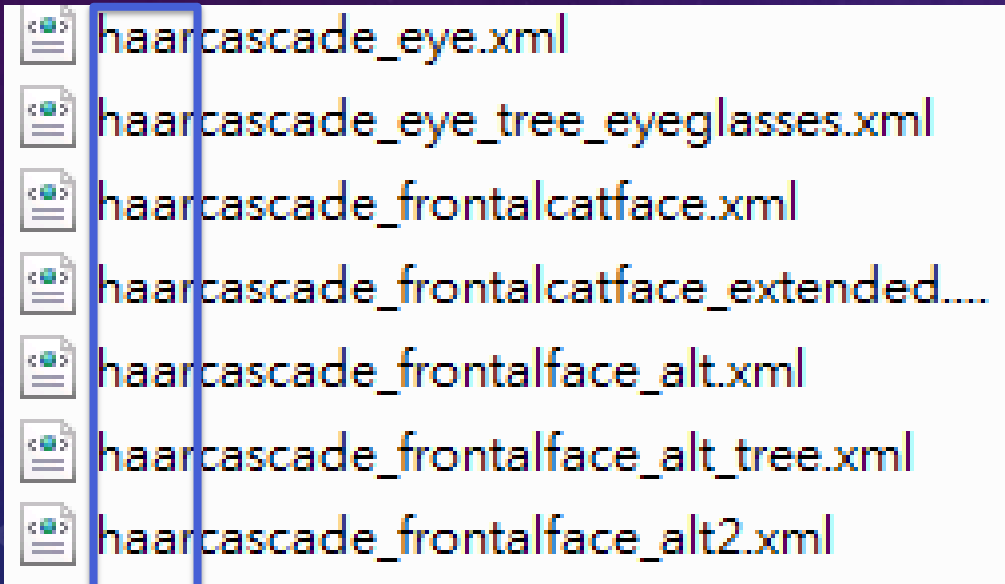
我們要讓電腦了解一個物體前，需要大量的範本來訓練它，範本愈多，辨別的精確度就愈高，少說也要數以百計才夠電腦使用，而視覺庫就是存放範本的地方，所以我們要達成電腦辨識前，需要先準備相應的視覺庫，但需求範本量大，要自己做視覺庫是個龐大的工程，所幸有已經做好的視覺庫，我們才可以輕鬆的使用，在此舉一個由intel公司所做的openCV視覺庫作為我們要用的工具。



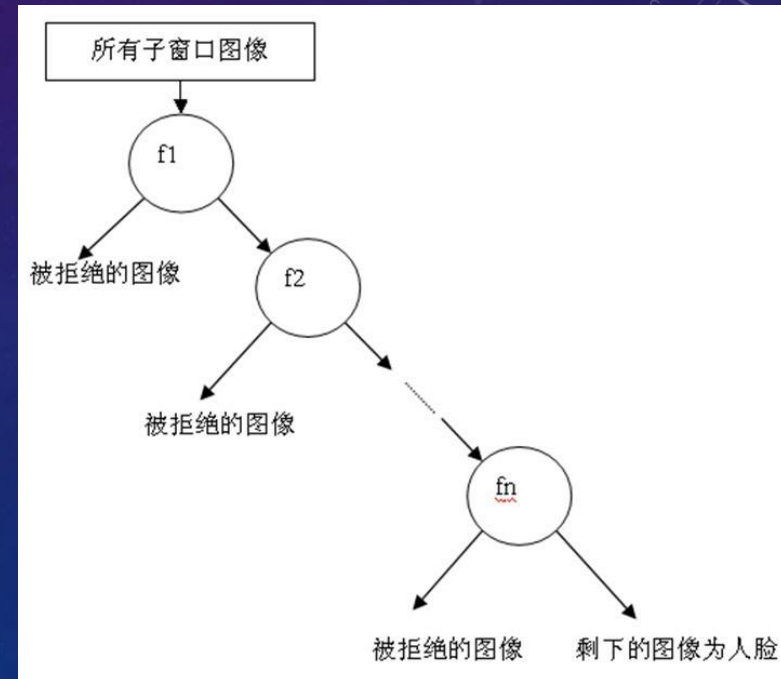
Haar分類器

我們打開openCV，會發現每一個視覺庫名稱前都會有一個Haar的英文

哈爾特徵（英語：Haar-like features）是用於物體辨識的一種數位影像特徵，是第一種即時的人臉檢測運算，最主要的優勢是它的計算非常快速。



▲電腦裡的樣子



▲算法示意圖

實作

1. 下載Opencv、opencv_contrib(openCV的分支)作為視覺庫，用以辨識有無鼻子作為判斷依據，協助我們辨認有無配戴口罩
2. 編寫Python程式語言
3. 測試程式成果

首先，我們要讓電腦有辦法將人臉部分用綠框圈住

```
In [1]: import cv2
cap=cv2.VideoCapture(0)
while(True):
    ret,frame=cap.read()

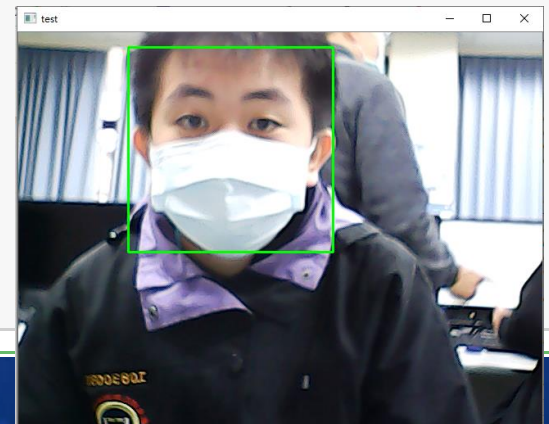
    face_cascade=cv2.CascadeClassifier("haarcascade_frontalface_default.xml")
    frame=cv2.flip(frame,1) #水平翻轉畫面
    gray=cv2.cvtColor(frame,cv2.COLOR_BGR2GRAY) #轉成灰階
    faces=face_cascade.detectMultiScale(gray,scaleFactor=1.2,minNeighbors=5,minSize=(30,30))
    for (x,y,w,h)in faces:
        cv2.rectangle(frame,(x,y),(x+w,y+h),(0,255,0),2) #臉畫綠框
    cv2.imshow("test",frame)
    if cv2.waitKey(1)&0xFF==ord("q"):
        break
cap.release()
cv2.destroyAllWindows()
```



```
In [1]: import cv2
cap=cv2.VideoCapture(0)
while(True):
    ret,frame=cap.read()

    face_cascade=cv2.CascadeClassifier("haarcascade_frontalface_default.xml")
    frame=cv2.flip(frame,1) #水平翻轉畫面
    gray=cv2.cvtColor(frame,cv2.COLOR_BGR2GRAY) #轉成灰階
    faces=face_cascade.detectMultiScale(gray,scaleFactor=1.2,minNeighbors=5,minSize=(30,30))
    for (x,y,w,h)in faces:
        cv2.rectangle(frame,(x,y),(x+w,y+h),(0,255,0),2) #臉畫綠框
    cv2.imshow("test",frame)
    if cv2.waitKey(1)&0xFF==ord("q"):
        break
cap.release()
cv2.destroyAllWindows()
```

▼配戴口罩時，臉變綠框



然後以鼻子作為判斷有無口罩的依據

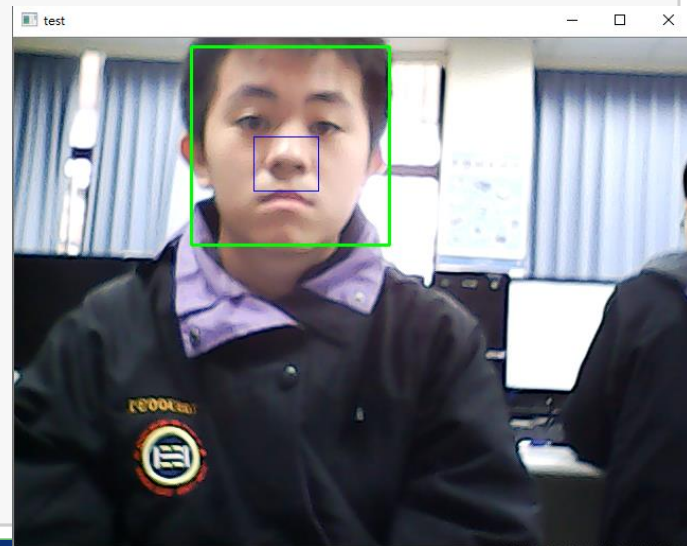
```
In [2]: import cv2
cap=cv2.VideoCapture(0)
while(True):
    ret,frame=cap.read()

    face_cascade=cv2.CascadeClassifier("haarcascade_frontalface_default.xml")
    nose_cascade =cv2.CascadeClassifier("haarcascade_mcs_nose.xml")
    frame=cv2.flip(frame,1) #水平翻轉畫面
    gray=cv2.cvtColor(frame,cv2.COLOR_BGR2GRAY) #轉成灰階
    faces=face_cascade.detectMultiScale(gray,scaleFactor=1.1,minNeighbors=5,minSize=(30,30))

    nose = nose_cascade.detectMultiScale(gray,scaleFactor=1.5,minNeighbors=5,minSize=(30,30))
    for (nx, ny, nw, nh) in nose:
        cv2.rectangle(frame,(nx,ny),(nx+nw,ny+nh),(255,0,0),1) #鼻子畫藍框

    for (x,y,w,h)in faces:
        cv2.rectangle(frame,(x,y),(x+w,y+h),(0,255,0),2) #臉畫綠框

    cv2.imshow("test",frame)
    if cv2.waitKey(1)&0xFF==ord("q"):
        break
cap.release()
cv2.destroyAllWindows()
|
```

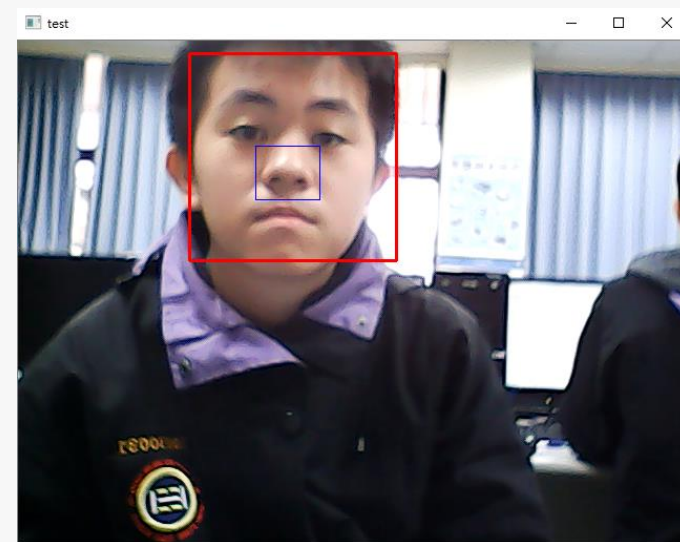


```

In [4]: import cv2
cap=cv2.VideoCapture(0)
while(True):
    ret,frame=cap.read()
    face_cascade=cv2.CascadeClassifier("haarcascade_frontalface_default.xml")
    nose_cascade =cv2.CascadeClassifier("haarcascade_mcs_nose.xml")
    frame=cv2.flip(frame,1)          #水平翻轉畫面
    gray=cv2.cvtColor(frame,cv2.COLOR_BGR2GRAY)    #轉成灰階
    faces=face_cascade.detectMultiScale(gray,scaleFactor=1.1,minNeighbors=5,minSize=(30,30))
    nose = nose_cascade.detectMultiScale(gray,scaleFactor=1.5,minNeighbors=5,minSize=(30,30))
    for (nx, ny, nw, nh) in nose:
        cv2.rectangle(frame,(nx,ny),(nx+nw,ny+nh),(255,0,0),1)
    if(len(nose)!=0):                #如果偵測到鼻子，表示沒戴口罩
        for (x,y,w,h)in faces:
            cv2.rectangle(frame,(x,y),(x+w,y+h),(0,0,255),2)    #臉畫紅框
        else:
            #否則表示有戴口罩
            for (x,y,w,h)in faces:
                cv2.rectangle(frame,(x,y),(x+w,y+h),(0,255,0),2)    #臉畫綠框
    cv2.imshow("test",frame)
    if cv2.waitKey(1)&0xFF==ord("q"):
        break
cap.release()
cv2.destroyAllWindows()

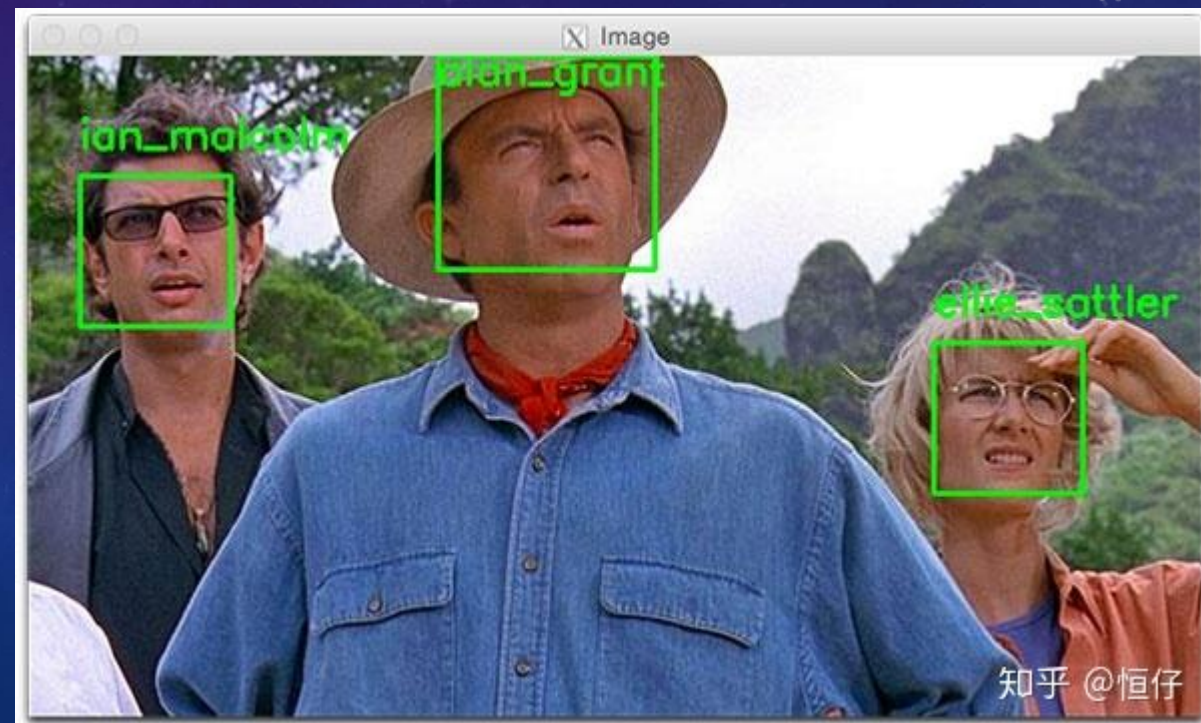
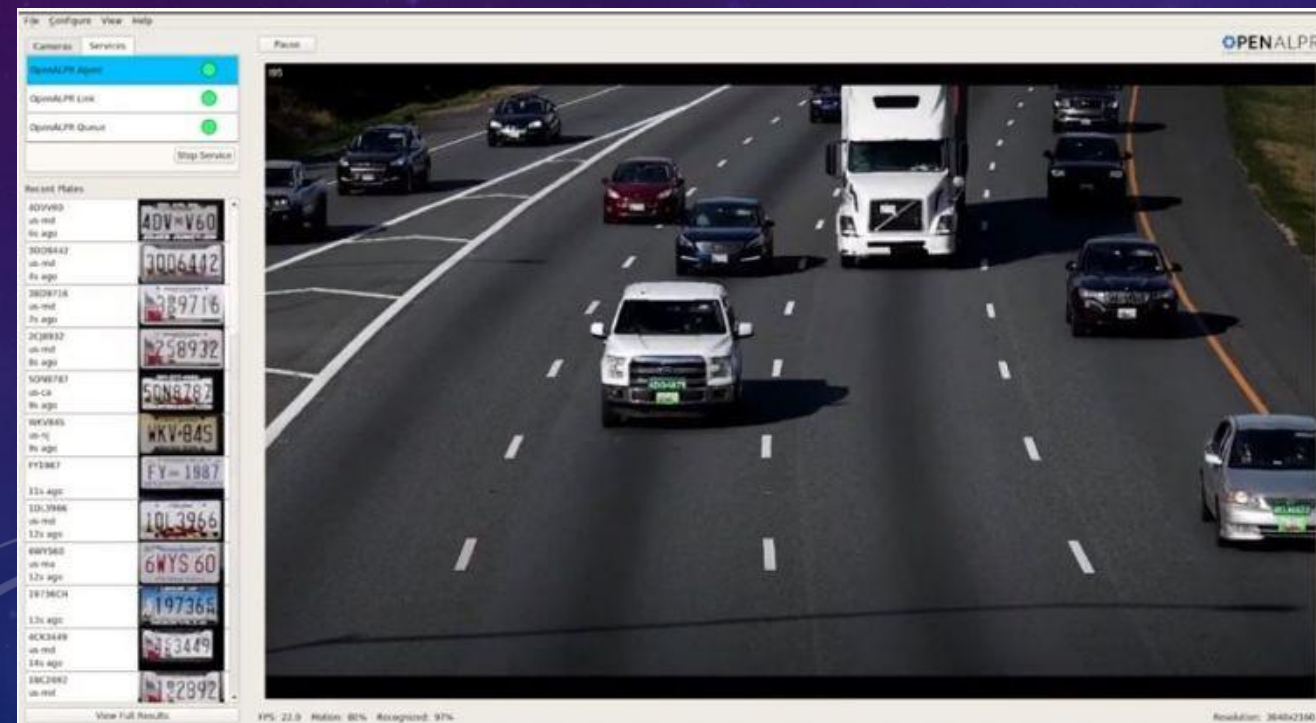
```

▼沒戴口罩時，臉變紅框



結論

透過此次的實作，將openCV與python應用，使用於有沒有配戴口罩的檢測，這次的辨識，還只是openCV的基礎應用，只要正確的訓練視覺庫，不但可以增加辨識率，還能更多方面的應用(舉凡車牌辨識，可以分辨特定人的辨識等等)，能做到很多的事情。



參考資料

<https://tpu.thinkpower.com.tw/tpu/articleDetails/950>

<https://blog.gtwang.org/programming/opencv-basic-image-read-and-write-tutorial/>

https://github.com/opencv/opencv_contrib

謝謝大家