

車牌辨識系統

一、資料前處理

01. 處理正樣本影像

A. 匯入模塊函數

【01】	<pre>import os, shutil, glob, PIL from time import sleep from PIL import Image</pre>
------	--

B. 建立 emptydir 函數，功用為建立資料夾，若原本就存在則做刪除再建立

【02】	<pre>def emptydir(dirname): if os.path.isdir(dirname): shutil.rmtree(dirname) sleep(2) os.mkdir(dirname)</pre>
------	--

- 1. os.path.isdir(path): 回傳布林值，表示該位置(path)資料夾是否存在
- 2. shutil.rmtree(path): 依序刪除該資料夾中的檔案
- 3. sleep(): 延遲秒數
- 4. os.mkdir(): 創建資料夾

C. 建立 dirResize 函數，功用為輸建資料夾與影像位置，將影像改大小 (300,225) 儲存至 emptydir 所建立的資料夾中

【03】	<pre>def dirResize(src, dst): myfiles = glob.glob(src+'/*.jpg') emptydir(dst) for i, f in enumerate(myfiles): img = Image.open(f) imgnew = img.resize((300,225), PIL.Image.ANTIALIAS) imgnew.save(dst+'/'+'resize'+str('{:0>3d}').format(i+1)+'.bmp')</pre>
------	--

- 5. glob.glob(path): 返回該位置的文件名稱 (只包括當前資料夾的文件，不包括子資料夾裡的文件)
- 6. Image.open(path): 讀取該位置的影像
- 7. img.resize((大小), PIL.Image.ANTIALIAS): 更改 img 的影像大小，並用插值方式縮放
- 8. img.save(path): 儲存 img 的影像至該位置

D. 使用 dirResize 函數來將 carPlate_sr 資料夾內的影像更改大小，並儲存至 carPlate 資料夾中

【04】	<pre>dirResize('carPlate_sr', 'carPlate')</pre>
------	---

02. 處理負樣本影像

A. 以同樣方式將負樣本影像作一樣的前處理，但是考慮正樣本車牌部分幾乎皆為黑白，為了提高訓練時的難易度，並將負樣本轉為灰階，並且將訓練負樣本的影像大小（500,376）比正樣本大

【05】

```
myfiles = glob.glob("carNegative_sr/*.jpg")
emptydir('carNegative')
for i, f in enumerate(myfiles):
    img = Image.open(f)
    imgnew = img.resize((500, 375), PIL.Image.ANTIALIAS)
    imgnew = img_new.convert('L')
    imgnew.save('carNegative/'+ 'negGray'+str('{:0>3d}').format(i+1)+'.bmp')
```

9. `img.convert('L')`: 將該 `img` 的影像轉為灰階

10. `{:0>3d}` 表示 3d 只顯示三位整數。`:0>`表示原數字在左邊沒滿三位數，則用 0 補起來。`:0<` 則是補右邊

03. 建立負樣本的標註檔案

A. 新建負樣本的標註 txt 檔案，檔案內容包含影像位置與標註數量及位置，由於負樣本都沒有車牌，所以自行建立

【06】

```
fp = open('Haar-Training-master/Haar-Training-master/training/negative/bg.txt', 'w')
files = glob.glob('Haar-Training-master/Haar-Training-master/training/negative/*.bmp')
text = ""
for file in files:
    basename = os.path.basename(file)
    filename = 'negative/' + basename
    text += filename + "\n"
print(text)
fp.write(text)
fp.close()
```

11. `file_obj = open("path.txt", mode='w')`: 開啟 `path` 的檔案，`mode` 常用的有：

- (1) `r` 為預設，只開啟檔案供讀取
- (2) `w` 開啟檔案供寫入，如果原先檔案有內容將被覆蓋
- (3) `a` 開啟檔案供寫入，如果原先檔案有內容，新寫入資料將附加在後面
- (4) `x` 開啟一個新的檔案供寫入，如果鎖開啟的檔案已經存在會產生錯誤

`file_obj.write("字串")`: 將字串寫入檔案

`file_obj.close()`: 不使用檔案時，將檔案正常關閉

12. `os.path.basename(path)`: 用於去掉資料夾的路徑，只返回文件名。（若要用於去文件名，但保留路徑，則使用 `os.path.dirname(path)`）

04. 建立正樣本的標註檔案

- A. 由於 04 與 05 這部分需要使用到正樣本影像已經標註好的檔案，因此要先去下載 OpenHaar 分類器，自行標註。可先跳到下個單元（二、安裝與訓練車牌號碼 Haar 特徵分類器模型）
- B. 安裝好 openCV 的 harr 後，開啟 Haar-Training-master\training\positive\objectmarker.exe 進行檔案標註
- 按下空白鍵就可將框選資料記錄下來
 - 按下 Enter 鍵完成這張影像標記
- C. 標注完成後，會在 Haar-Training-master\training\positive 的資料夾中，產生 info.txt 的標注檔案，它的資料結構為影像位置、標註數量及位置

05. 調整正樣本的影像與標註檔案

- A. Haar 分類器會根據訓練時的寬高比來框選物件，由於舊式車牌為六碼而新式車牌為七碼，因此有機率會考慮舊式車牌來框選物件，導致新式車牌只截取到部分車牌號碼。因此需要調整寬高比，根據計算新式車牌的寬高比為 1:3.8。以下程式要把寬高比小於 3.8 的影像調整至 3.8:

【07】

```
fp = open('Haar-Training-master/Haar-Training-master/training/positive/info.txt', 'r')
lines = fp.readlines()
rettext = ''

for line in lines:
    data = line.split(' ')
    rettext += data[0] + ' ' + data[1] + ' '

    for i in range(int(n)):
        x = float(data[2+i*4])
        y = float(data[3+i*4])
        w = float(data[4+i*4])
        h = float(data[5+i*4])

        if (w/h) < 3.8:
            newW = h * 3.8
            x -= int((newW - w) / 2)

            if x<=0:
                x=0

            w = int(newW)

            rettext = rettext+str(int(x))+' '+data[3+i*4]+' '+str(int(w))+' '+data[5+i*4]

fp.close()

fp = open('Haar-Training-master/Haar-Training-master/training/positive/info.txt', 'w')
fp.write(rettext)
fp.close()
```

13. str.split(str="分割符號", num=數值): 將 str 使用分割符號進行分割，分割 num+1 個。若沒有設置 num 數值，則字串採用全部分割



06. 影像增量

A. 由於正樣本影像太少（不均勻類別）會使得訓練的模型預測能力很差。增量的方法很多，本教材我們使用裁減的方法進行增量，我們針對四個角落各別移除邊緣長寬 10%來產生新影像，以左上角為例：移除上方 30 像素左方 22 像素，再將移除後的新影像放大至 300x225 像素（需要設立條件式防止裁減的位置將車牌分割掉）。除了影像進行增量外，標注檔案也要進行更新

【08】

```
path = 'Haar-Training-master/Haar-Training-master/training/positive/'
fp = open(path + 'info.txt', 'r')
lines = fp.readlines()
count = len(glob.glob("carPlate/*.bmp"))

rettext = ''
for line in lines:
    data = line.split(' ')
    img = Image.open(path + data[0])

    x = int(data[2])
    y = int(data[3])
    w = int(data[4])
    h = int(data[5])

    reduceW = 30 #減少的的寬度
    reduceH = int(reduceW*0.75) #減少的的高度
    multi = float(300/(300-reduceW)) #原圖與新圖比例
    neww = int(w*multi) #新圖的寬
    newh = int(h*multi) #新圖的高

    #移除左上角圖
    if (x-reduceW)>5 and (y-reduceH)>5: #左上角有空間才移除左上角
        count += 1 #編號加1,此數值會做為檔名用
        newimg = img.crop((reduceW, reduceH, 300, 225)) #擷取圖形
        newimg = newimg.resize((300, 225), Image.ANTIALIAS) #放大圖形
        newimg.save(path + 'rawdata/bmpraw{:0>3d}.bmp'.format(count), 'bmp') #存檔
        newx = int((x-reduceW)*multi-reduceW*(multi-1)/2) #新圖X坐標
        newy = int((y-reduceH)*multi-reduceH*(multi-1)/2) #新圖Y坐標
        rettext = rettext+'rawdata/bmpraw{:0>3d}.bmp'.format(count)+' '+str(newx)+' '+str(newy)+' '+str(neww)+' '+str(newh)+'\n' #記錄新影像資料

    #移除右上角圖
    if (x+w)<(300-reduceW-5) and y>(reduceW+5):
        count += 1
        newimg = img.crop((0, reduceH, (300-reduceW), 225))
```

```

newimg = newimg.resize((300, 225), Image.ANTIALIAS)
newimg.save(path + 'rawdata/bmp{0>3d}.bmp'.format(count), 'bmp')
newx = int(x*multi)
newy = int((y-reduceH)*multi)
rettext = rettext+'rawdata/bmp{0>3d}.bmp'.format(count)+' '+'1'+ ' '+str(new
x)+' '+'str(newy)+' '+'str(neww)+' '+'str(newh)+'\n'

#移除左下角圖
if (x-reduceW)>5 and (y+h)<(225-reduceH-5):
    count += 1
    newimg = img.crop((reduceW, 0, 300, 225-reduceH))
    newimg = newimg.resize((300, 225), Image.ANTIALIAS)
    newimg.save(path + 'rawdata/bmp{0>3d}.bmp'.format(count), 'bmp')
    newx = int((x-reduceW)*multi)
    newy = int(y*multi)
    rettext = rettext+'rawdata/bmp{0>3d}.bmp'.format(count)+' '+'1'+ ' '+str(new
x)+' '+'str(newy)+' '+'str(neww)+' '+'str(newh)+'\n'

#移除右下角圖
if (x+w)<(300-reduceW-5) and (y+h)<(225-reduceH-5):
    count += 1
    newimg = img.crop((0, 0, (300-reduceW), 225-reduceH))
    newimg = newimg.resize((300, 225), Image.ANTIALIAS)
    newimg.save(path + 'rawdata/bmp{0>3d}.bmp'.format(count), 'bmp')
    newx = int(x*multi)
    newy = int(y*multi)
    rettext = rettext+'rawdata/bmp{0>3d}.bmp'.format(count)+' '+'1'+ ' '+str(new
x)+' '+'str(newy)+' '+'str(neww)+' '+'str(newh)+'\n'

fp.close()

fpmake = open(path + 'Info.txt', 'a') #以新增資料方式開啟檔案
fpmake.write(rettext) #寫入檔案
fpmake.close()

```



14. `file_obj.readlines()`:讀取整個檔案所有行，儲存在一個列表(list)變數中，每行作為一個元素，但讀取大檔案會比較佔記憶體
15. `img.crop((x1, y1, x2, y2))`: 將 `img` 的影像進行裁減，將左上角與右下角裁減為(x1, y1)與(x2, y2)