【2021 全國科學探究競賽-這樣教我就懂】

題目名稱：機器視覺~應用 OpenCV 車牌辨識探究

一、摘要：

本研究主要使用 OpenCV 透過自行訓練並建立 Haar 特徵分類器「偵測車牌」模型，先偵測框選車牌，接著將用光學辨識軟體 OCR 辨識出車牌號碼。一、訓練建立 Haar 特徵分類器「偵測車牌」模型時需依：1.調整檔案架構。2.加入正樣本。3.加入負樣本。4.正樣本標記。5.微調標記框。6.微調寬高比。7.擴增樣本數量。8.打包向量檔。9.進行訓練。二、用自行訓練的 Haar 特徵分類器模型，可順利框選並正確擷取「車牌影像」。三、車牌影像若經過擷取，以輪廓偵測予以提取個別字母再重組，同時把雜點、分隔符號及畸零地一併去除，重組成另外一張更清晰的車牌，再用 OCR 對擷取「車牌影像」進行辨識更能夠提高辨識率。OCR 辨識比較適合在特定的有完善控制的場合。生活上除了車牌號碼可以透過影像辨識來管理，其他如通行證、身分證等，亦或是更複雜的文件上有文字、數字及特殊符號都可以透過人工智慧的影像辨識來處理，讓人們的生活更加便利，也更加安全。

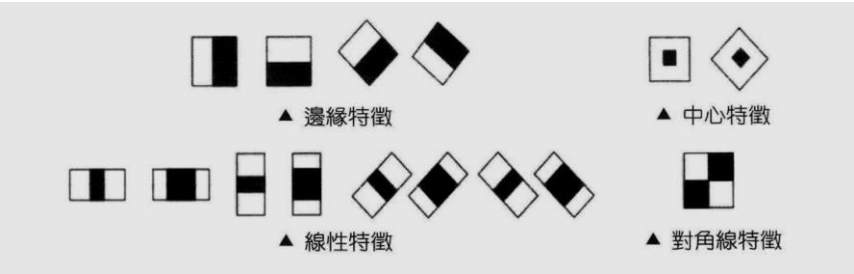
二、探究題目與動機

OpenCV 是影像辨識的超級巨星，使用 OpenCV 提供的 Haar 特徵分類器人臉模型，可以輕鬆偵測人臉部的位置並將影像框選擷取下來。那它還可以偵測其他物件嗎？若要偵測特定物件，就要自行訓練並建立該物件的 Haar 特徵分類器模型，再使用自行建立的 Haar 特徵分類器模型來偵測物件。

目前許多停車場已使用自動車牌辨識系統經營以節省人力成本，我們希望一步一步地探究影像辨識的技術並模擬建立停車場自動車牌辨識系統，從無到有自行訓練並建立 Haar 特徵分類器「偵測車牌」模型，先偵測框選車牌，接著將用光學辨識軟體 OCR 辨識出車牌號碼。

背景知識：

Haar 特徵是用簡化的數據來表示一張圖片內容的局部特徵。它是用一個矩形區域表示，並可進行旋轉、平移、縮放等：其中邊緣特徵有 4 種，中心特徵有 2 種，線性特徵有 8 種，對角線特徵有 1 種，共有 15 種特徵。Haar 特徵值反映了圖像的明暗度變化情況。例如：臉部的一些特徵能由矩形特徵簡單描述，眼睛的部分符合中心特徵，眉毛的部分符合邊緣特徵。Haar 特徵值越大，表示此區域越符合 15 種當中特定的 Haar 特徵。



訓練時，需框選出要偵測的圖形區域，系統就依照框選區域的 Haar 特徵分類器模型，提供的圖形數量愈多且形態愈多元，建立的模型準確度就會越高。使用 Haar 特徵分類器模型，系統會從要偵測的圖片左上角產生一個檢測矩形，檢查矩形內的圖形是否符合 Haar 特徵分類器模型特徵。接著將此矩形向右移動檢測，到最右方後再移到左側下方檢測，直到圖片移到右下角為止，這樣就可以檢測整張圖片。

三、探究目的與假設

一、 以偵測圖片中「車牌號碼」為目標，自行訓練並建立 Haar 特徵分類器模型。

二、 用自行訓練建立的 Haar 特徵分類器模型，框選並擷取「車牌影像」。

三、 用 OCR 對擷取「車牌影像」進行辨識，並檢驗車牌號碼的辨識正確性。

四、探究方法與驗證步驟

Haar 特徵分類器可以幫我們偵測某特定物件是否存在圖片中，並可得知該物件在圖片中的位置。這個特定物件可以是人臉、交通標誌、動物等，需根據特定的 Haar 特徵模型檔。而現在我們就要自己建立偵測車牌號碼的 Haar 特徵模型檔。建立完成 Haar 特徵分類器模型後，便可使用 OpenCV 讀入模型，而對未知的車牌進行偵測，並觀察偵測車牌號碼的正確率。

實作一：以偵測圖片中「車牌號碼」為目標，自行訓練並建立 Haar 特徵分類器模型

A. 收集訓練 Haar 特徵分類資料

訓練 Haar 特徵分類器模型需要「正樣本圖片」及「負樣本圖片」，訓練完後會產生模型，另外還可用一批「實測圖片」(和訓練圖片不同批)來測試模型的偵測效果。「正樣本圖片」及「實測圖片」都是包含要偵測圖形的圖片，我們要建立的偵測車牌號碼 Haar 特徵分類器模型，正樣本實測圖片就是含有車牌號碼的圖片。

1.先蒐集「正樣本圖片」，就是包含要偵測的車牌號碼的圖片。手機拍攝的圖片解析度太高，一般不適合做為訓練圖片，正樣本圖片格式必須為 BMP，所以將圖片格式轉換為 BMP，同時將圖形大小轉換為 300x225 像素。將轉換程序寫成函式可以方便整批轉換。

2.再蒐集「負樣本圖片」，負樣本圖片就是不包含車牌號碼的圖片，轉換負樣本圖片格式為灰階，圖形大小為 500x375 像素。如果能與被偵測物件相關聯的圖片更好，所以此處負樣本圖片使用各種道路圖片。訓練 Haar 特徵分類器時，負樣本圖片的尺寸需略大於正樣本圖片，且負樣本圖片格式必須為灰階的。

3.以固定的寬高比例框選物件：因為用 Haar 特徵分類器模型進行物件偵測時，需要根據訓練時的寬高比例來框選物件。若要系統知道偵測的物件是車牌號碼，我們必須框出所有正樣本圖片中車牌號碼的區域。而新式車牌的寬約為高的 3.8 倍，我們須將框選區域寬高比值小於 3.8 的圖片都調整為 3.8。

4.圖片越多訓練效果越好：圖片的收集應量多、樣多、角度多，我們收集到的圖片依次移除圖片上、下、左、右邊緣 10% 的畫素來產生並擴增新圖片，如此每張圖片可新增 4 張新的圖片。這些都需經由程式批次處理。

B. 訓練建立車牌號碼 Haar 特徵分類器模型

前面我們已準備好車牌辨識的正、負樣本圖片，使用這些圖片進行訓練，就可以建立偵測車牌號碼的 Haar 特徵分類器模型。OpenCV 官方有提供自行建立 Haar 特徵模型的方法，但過程非常繁複。眾多網路開放資源當中我們依照「https://github.com/sauhaardac/haar-training」所整理建立了訓練 Haar 特徵模型的檔案結構，及操作其撰寫的批次指令來訓練我們的模型，大幅簡化了繁複程序。主要步驟如下：

1.調整檔案架構 >> 2.加入正樣本 >> 3.加入負樣本。
>> 4.正樣本標記。 >> 5.微調標記框。 >> 6.微調寬高比。
>> 7.擴增樣本數量。 >> 8.打包向量檔。 >> 9.進行訓練

訓練的時間相當長(約 2 小時)，需要耐心等待。訓練完成後會產生<haar_carplate.xml >檔案，就是訓練完成的 Haar 特徵模型檔。

實作一 檢討與改進：

停車場自動車牌辨識系統使用時，業者可控制攝影機或相機的拍攝條件，如攝影機裝設位置、停車場光線等，以提高辨識率。我們車牌是以手機拍攝，正負樣本收集時需注意下列事項以增加訓練效果及辨識率：

1.**新舊車牌數量均衡：**目前台灣現行車牌有兩種型式，舊式車牌為 6 碼，新式車牌為 7 碼。兩者的長寬比例不同，文字字型也不同，訓練圖片兩種車牌都要包含。

2.拍攝時和車牌平行：由於車牌的位置較低，拍攝時手機位置較高，車牌會呈現梯形變形降低辨識率，應盡量等高拍攝車牌才會呈現矩形。

3.圖片尺寸大小適中：注意拍攝距離，近拍圖片太大辨識費時，遠拍圖片太小不清楚影響辨識正確率。

4.拍攝時自然光較佳：拍攝時最好選在晴天白天戶外場合，避免在光線不足的室內停車場拍攝車牌樣本。

5.微調標記框選區域：應改由程式控制微調，減少人工作業的失誤。

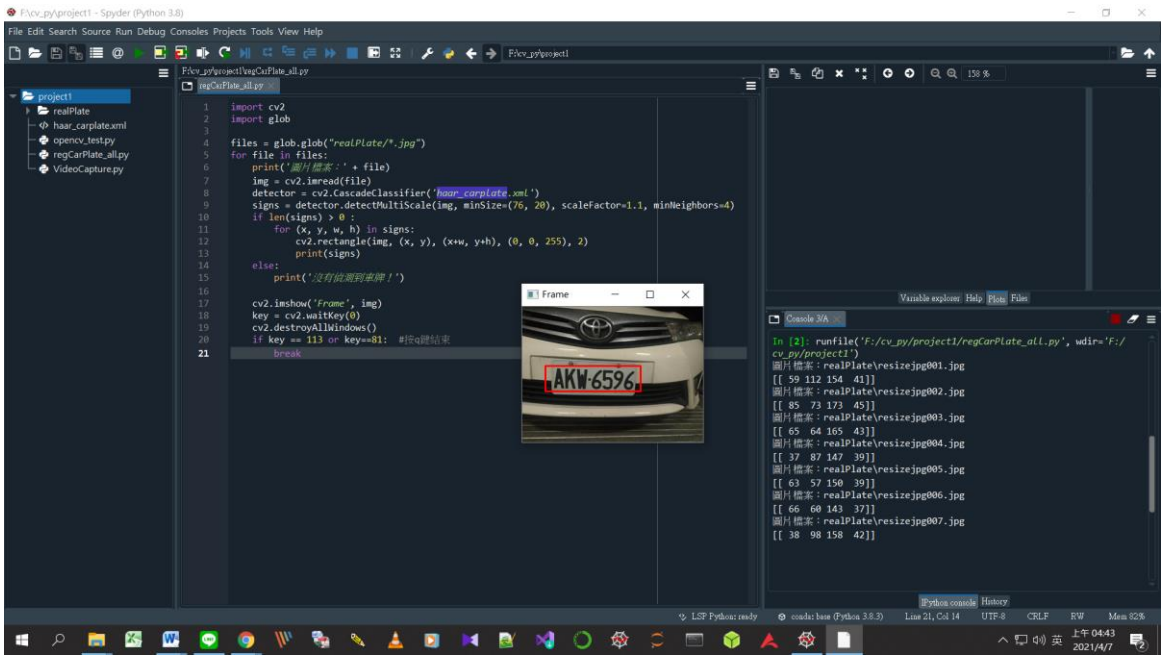
實作二：用自行訓練建立的 Haar 特徵分器模型，框選並擷取「車牌影像」

程式碼：

```
def emptydir(dirname):                  #清空資料夾
    if os.path.isdir(dirname):          #資料夾存在就刪除
        shutil.rmtree(dirname)
        sleep(2)                        #需延遲,否則會出錯
    os.mkdir(dirname)                    #建立資料夾

def dirResize(src, dst):
    myfiles = glob.glob(src + 'carPlate sr/*.JPG')   #讀取資料夾全部 jpg 檔案
    emptydir(dst)
    print(src + ' 資料夾：')
    print('開始轉換圖形尺寸！')
    for i, f in enumerate(myfiles):
        img = Image.open(f)
        img new = img.resize((12,9), PIL.Image.ANTIALIAS)  #尺寸 300x225
        outname = str("resizejpg") + str('{:0>3d}').format(i+1) + '.jpg'
        img new.save(dst + '/' + outname)
    print('轉換圖形尺寸完成！\n')

```

```
import PIL
from PIL import Image
import glob
import shutil, os
from time import sleep
dirResize('carPlate sr', 'carPlate')
dirResize('realPlate sr', 'realPlate')
import cv2
import glob

files = glob.glob("carPlate sr/*.jpg")
for file in files:
    print('圖片檔案：' + file)
    img = cv2.imread(file)
    detector = cv2.CascadeClassifier('haar carplate.xml')
    signs = detector.detectMultiScale(img, minSize=(76, 20),
scaleFactor=1.1, minNeighbors=4)
    if len(signs) > 0 :
        for (x, y, w, h) in signs:
            cv2.rectangle(img, (x, y), (x+w, y+h), (0, 0, 255), 2)
            print(signs)
    else:
        print('沒有偵測到車牌！')

    cv2.imshow('Frame', img)
    key = cv2.waitKey(0)
    cv2.destroyAllWindows()
    if key == 113 or key==81:  #按 q 鍵結束
        break

```

執行結果：



實作二 檢討與改進：

若偵測到車牌，會將車牌號碼框選出來。程式改為可按任一鍵，可連續偵測下一張圖片。實作一模型的訓練成效，會直接影響實作二的辨識框選結果。如圖所示，辨識框選的結果非常精確。

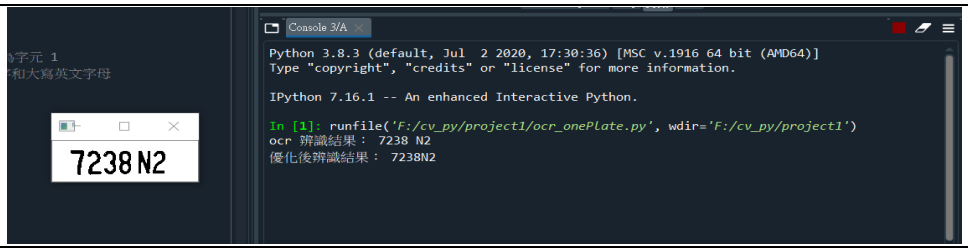
實作三：用 OCR 對擷取「車牌影像」進行辨識，並檢驗車牌號碼的辨識正確性

要將圖片中的數字或英文字母轉換為文字，可透過圖形處理模組將大部分圖片背景去除，再以 OCR (Optical Character Recognition,光學字元辨識) 讀取並轉換成文字。Tesseract 是一個很受歡迎的 OCR 程式庫，可由「http://digi.bib.uni-mannheim.de/tesseract/tesseract-ocr-w64-setup-v5.0.0-alpha.20200328.exe」下載安裝檔。<tesseract-ocr-w64-setup-v5.0.0-alpha.20200328.exe>，於安裝檔按滑鼠左鍵兩下即可開始安裝，接著連續按”下一步”便可完成安裝。預設的目錄是<C:\Program Files\Tesseract-OCR>，要在環境變數 Path 設定這個路徑。另外還要安裝 pyocr 模組就可以透過指令來操作文字辨識。

程式碼：

```python
import cv2
from PIL import Image
import sys
import pyocr
import pyocr.builders
import re

image = cv2.imread('assember.jpg')
#OCR 辨識車牌
tools = pyocr.get_available_tools()
if len(tools) == 0:
    print("No OCR tool found")
    sys.exit(1)
tool = tools[0]  #取得可用工具

result = tool.image_to_string(
    Image.open('assember.jpg'),
    builder=pyocr.builders.TextBuilder()
)
# 將 ocr 辨識結果優化
txt=result.replace("!","1") # 如果是 ! 字元，更改為字元 1
real_txt=re.findall(r'[A-Z]+|[\d]+',txt) #只取數字和大寫英文字母

#組合真正的車牌
txt_Plate=""
for char in real_txt:
    txt_Plate += char
print("ocr 辨識結果：", result)
print("優化後辨識結果：",txt_Plate)

cv2.imshow('image', image)          #顯示原始圖形
cv2.moveWindow("image",800,700)     #將視窗移到指定位置
key = cv2.waitKey(0)                #按任意鍵結束
cv2.destroyAllWindows()
```

執行結果：



實作三 檢討與改進：

如圖所示，辨識的結果非常精確。車牌影像若經過擷取，以輪廓偵測予以提取個別字母再重組，會提高辨識度。提取的同時還可以把雜點、分隔符號及畸零地一併去除，重組成另外一張更清晰的車牌，更能夠提高辨識率。

五、結論與生活應用

用 OCR 對擷取「車牌影像」進行辨識首先要能夠正確的擷取到車牌號碼；要正確的擷取到車牌號碼就取決於訓練的模型，要增加模型的準確性就要使用大量的車牌做訓練，環環相扣。另外，車牌影像太大或太小、車牌是否清晰、光線是否充足、是否反光或者有霧氣阻擋、以及拍攝角度是否造成車牌號碼影像歪斜梯形化，這都會影響辨識的結果。因此，OCR 辨識比較合在特定的場合，攝影機位置固定調整好，光線太暗能給予補光，車牌影像的大小、角度都在特定的控制之下，更能提高辨識率。

生活上除了車牌號碼可以透過影像辨識來管理，其他如通行證、身分證等，亦或是更複雜的文件上有文字、數字及特殊符號都可以透過人工智慧的影像辨識來處理，讓人們的生活更加便利，也更加安全。

參考資料

1. Python 機器學習超進化：AI 影像辨識跨界應用實戰，文淵閣工作室，碁峯資訊出版
2. Python 技術者們-2 實踐！帶你一步一腳印由初學到精通，施威銘研究室，旗標資訊出版
3. 科技大觀園：智慧影像辨識的應用，李承勳，太奇雲端股份有限公司
4. 科技大觀園：智慧輪椅機器人讓一人就醫不困難，范賢娟，臺灣師範大學科學教育研究所