

靜宜大學資訊工程學系

專題報告書

專題名稱：影像辨識深度學習)

實驗室名稱：主顧 512

指導老師：林耀鈴

專題學生：

資工四 A 410754370 唐永信

資工四 A 410754257 廖裕乾

資工四 A 410715978 何冠霖

資工四 A 410781123 梁哲綸

中華民國 110 年 12 月 8 日

緒論

一、摘要

近年來資訊領域快速發展。特別是近 10 年人工智慧深度學習的快速發展，在電腦視覺領域有了更多元的應用，在影像辨識領域臉部方面也被廣泛應用，例如：臉部辨識、deepfakes、美顏相機等等。

我們先透過 Dlib 技術，即時偵測臉部特徵來偵測臉部特徵點，再將偵測到的臉部特徵描繪起來，作為生成的條件制約，然後再將要生成的人物影片，例如：明星、政治人物等等把影片轉成圖像作為訓練的資料集，再合成臉部與訓練，最後透過攝像頭及時同步使用者與影片人物的表情動作。

二、專題研究動機

因為疫情的關係大部分的人喜歡選擇室內的休閒娛樂，而室內的活動基本不外乎就是看電影、玩遊戲或者是視訊，影視產業的興起造就許多人選擇在家過週末，而影視產業的產能也就處於供不應求的狀態。當我們發現場景和人物並非我們所認識屬於這個世界的場景時，大概就能知道就是特效的功勞，而我們做的就類似於將我們需要特殊人物的表情去訓練，再由任何人去模擬他的行為和表情，這樣不用化妝的形式也能讓一個虛擬人物活生生的出現在舞台，減少了需要一步一步刻畫人物表情的時間。

而在現實生活中 NVIDIA 這家在計算機圖形處理數一數二的公司，在 2021 召開了一場 GTC 線上峰會時使用了虛擬替身來演講，當大家看完這場發佈會時也並未察覺奇怪的地方，直到廠商公布其中有 10 秒左右為 AI 為虛擬人物時，其過程完全看不出來，而我們所希望能做到的就是用 AI 辦一場不露臉的替身演講。

三、關鍵字

影像辨識、Dlib、卷基神經網路(CNN)、U-net 架構、Tensorflow、Keras

四、系統設計

這次的訓練主要將臉部偵測做精確化，利用少量且重要的特徵，如人臉五官特徵（眼睛或嘴巴... 等等）達到理想的辨識率。分析比較不同取樣特徵的辨識結果，找出理想的人臉特徵，比較以往直接機器辨識的作法，此次將人臉鎖定在同一個地方，以減少人臉遠近大小不一以及臉部置中問題。系統用到 U-Net 的架構，U-Net 是 Autoencoder 演算法的延伸，使輸出的尺寸越變越小，之後，解碼器再由這些變小的特徵，重建成與原圖一樣大小的新圖像，而 U-net 可以用來標示更準確的物體位置，提供詳細的每畫素反饋給生成器，同時保持合成影象的全域性一致性。這樣在重建的過程就比較不會遺失重要資訊了。

五、 開發工具及使用環境

本專題於 Windows 10 環境開發

顯示卡 geforce gtx 1050

python 3.7.11

tensorflow 2.4.1

keras 2.3.1

cuda toolkit=10.1

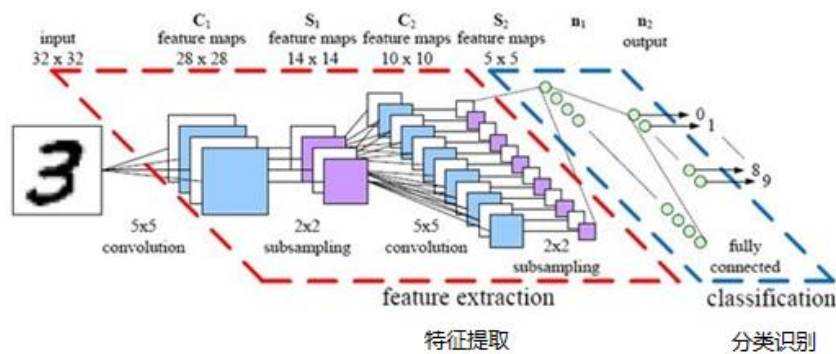
cudnn=7.6.5

dlib 19.21.1(py3.7)

六、 使用技術

(一)CNN(卷基神經網路)

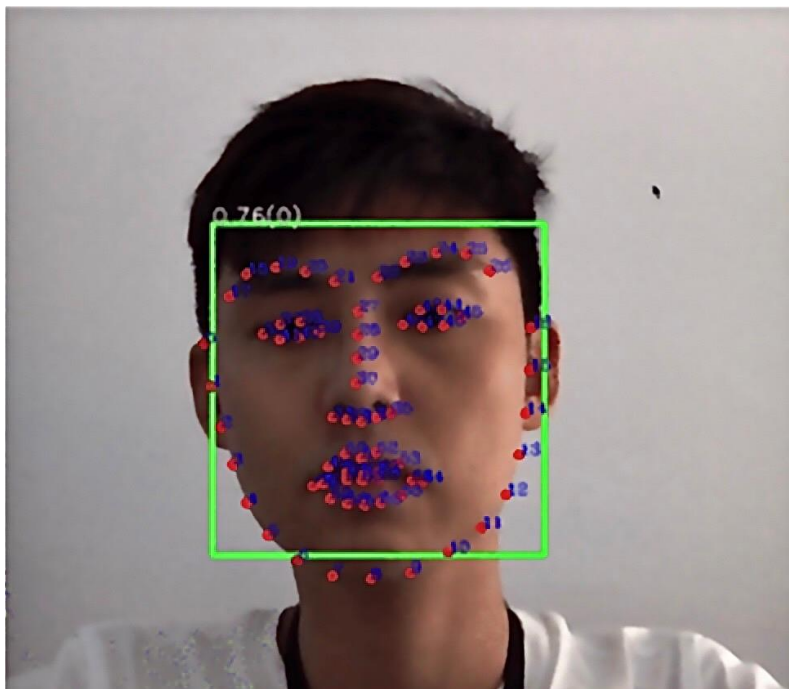
常見的深度學習模型主要有全連接網絡結構 FC、卷積神經網絡 CNN 和循環神經網絡 RNN。一般的神經網路由三個部分組成，分別是輸入層(Input Layers)、隱藏層(Hidden Layers)以及輸出層 (Output Layers)，CNN 的一大特色就是能保留圖像的位置資訊，不難想像圖片中的像素 (Pixels)與其鄰近的像素會有一定的關聯度，如果我們使用 FC 的結構來訓練圖像資訊的話，要先通過一個展開 (Flatten)的步驟，把高維的資訊拉成一條直線，如此一來就會大量失去特徵之間的空間資訊，效果當然會大打折扣。很有名的例子便是智慧圍棋軟體 AlphaGo 戰勝世界冠軍李世石，其中就使用到卷基神經網路的技術。



(二)dlib

人臉偵測的實作方法有很多種，這裡我們選擇使用 Dlib 這套機器學習函式庫所提供的人臉偵測，廣泛應用於工業界和學術界，包括機器人，嵌入式設備，行動電話和大型高性能計算環境。Dlib 的開源許可，令人們可以免費使用。

1. Dlib 可以使用 `pip install` 來安裝或者到官網下載
2. `dlib-19.6.1-cp36-cp36m-win_amd64.whl` 安裝
3. whl 文件的安裝，需要 cmd 進入終端木錄下，使用 `pip install dlib-19.8.1-cp36-cp36m-win_amd64.whl` 來實現 dlib 的安裝
4. 下載 dlib 的 68 點人臉特徵模型

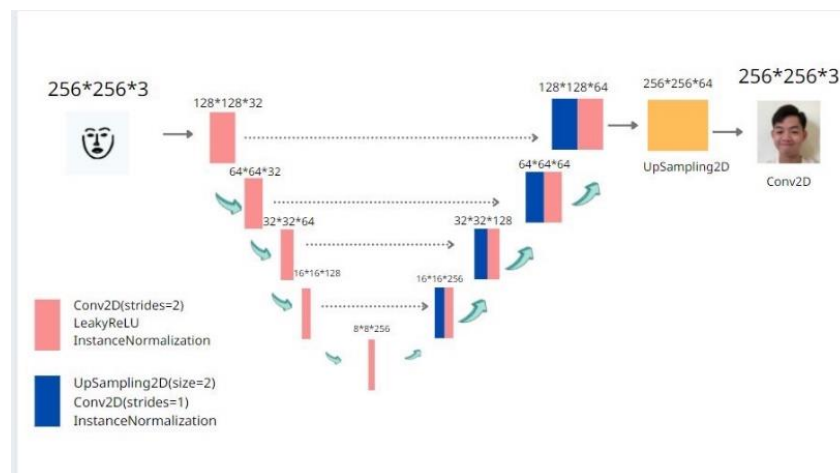


dlib 提供訓練好的模型，可以辨識出人臉的 68 的特徵點，68 特徵點包括鼻子、眼睛、眉毛，以及嘴巴等等，如上圖紅點就是偵測出人臉的 68 個特徵點。

(三)U-net 架構

U-net 架構是一種捲積神經網路的方法，可以很好的分割並降維圖片，U-net 的前半部分是特徵提取，後半部分是上取樣。在一些文獻中把這種結構叫做編碼器-解碼器結構，編碼器使用標準捲積層，解碼器則是使用反捲積和一個最終捲積層。

跳階(灰色箭頭) 保留原有的特徵，以大圖轉小圖可能沒有問題，若從小圖轉至大圖，就有失真的可能性，所以跳階可以保有原來的特徵。



(四) 圖片正規化

圖片大小正規化為了使擷取下來的圖片擁有一致的特徵，把圖片縮放成一定的大小比例，能夠讓要擷取的特徵克服放大/縮小的問題，在後續處理上能提高比對人臉的五官，使辨識度更加提升。

```

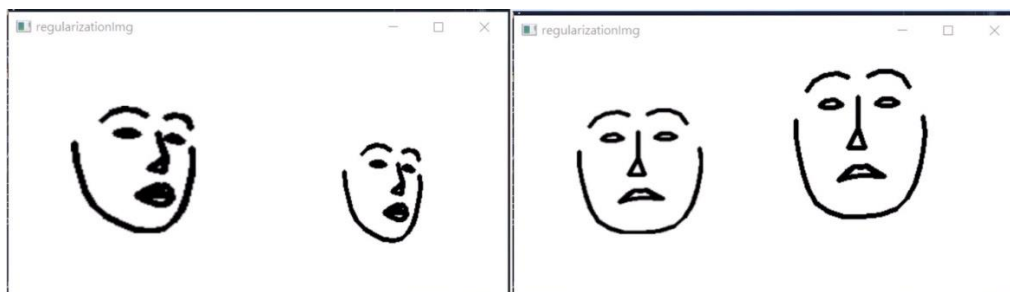
'''
-----
/ (x1,y1) /
/ (x2,y2) /
-----*
'''
'''
    原點
'''
def findBound(mask):
    x1,x2,y1,y2 = 0,0,0,0
    gray = cv2.cvtColor(mask, cv2.COLOR_BGR2GRAY)
    for i in range(gray.shape[0]):
        if np.where(gray[i,:]==0)[0].size > 1:
            x2 = i
            break
    for i in range (gray.shape[0]-1,0,-1):
        if np.where(gray[i,:]==0)[0].size > 1:
            x1 = i
            break
    for i in range(gray.shape[1]):
        if np.where(gray[:,i]==0)[0].size > 1:
            y2 = i
            break
    for i in range (gray.shape[1]-1,0,-1):
        if np.where(gray[:,i]==0)[0].size > 1:
            y1 = i
            break
    return x1,y1,x2,y2

```

```

def regularization(mask):
    white = 255
    halfSize = mask.shape[0]//2
    resize = (halfSize,halfSize)
    quarterSize = halfSize//2
    white_backGroup = np.full(mask.shape, white, np.uint8)
    x1,y1,x2,y2 = findBound(mask)
    #print((x1,y1),(x2,y2))
    if x1!=0 and x2!=0 and y1!=0 and y2!=0:
        mask_resize = cv2.resize(mask[x2:x1, y2:y1], resize , cv2.INTER_AREA)
        #將mask_resize 放入白背景正中間
        white_backGroup[halfSize-quarterSize : halfSize+quarterSize, halfSize-quarterSize : halfSize+quarterSize] = mask_resize
    return white_backGroup, mask_resize
return white_backGroup, mask

```



由左圖可看出右側輪廓無論是在較遠且偏右下的情況，或是右圖較為靠近輪廓，左側的輪廓均固定在中間且具同一大小。

七、系統訓練集

為了擁有更好更穩定的訓練資料，我們對模型使用許多種不同的嘗試。



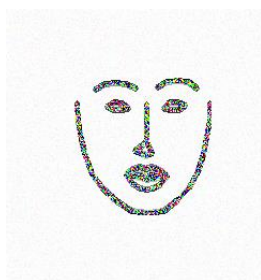
一般



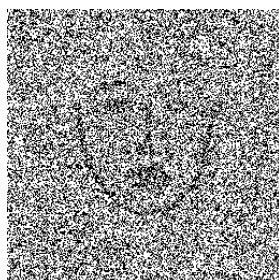
正旋轉



負旋轉



高斯雜訊



胡椒鹽雜訊



隨機挖空



輪廓加深



輪廓變淺

※隨機挖空:將一塊 30*30 像素，隨機缺一角設為空白，嘗試讓模型是否更加穩定

八、 成本分析

硬體設備:

主機板:華碩 STRIX B360-F (約 3000

CPU : Intel Core i7-10700KF (約 9000

顯卡 GPU : NVIDIA GTX 1650 4G (約 4000

RAM: DDR4 8G (約 1500

SSD:512G (約 2000

550W 電源供應器 (約 2000

720P 攝像頭 (約 500

軟體設備:(免費開源軟體成本不計)

九、 結論及未來展望

人臉辨識能應用的範圍深且廣，我們不只要讓機器識別我們的表情，還需要花大量的時間去訓練模型。接續上次的專題，我們再接觸人臉辨識的另一個部分，這次風格的轉換再次讓人臉有了新的風貌。在應用上或許只是一個供人娛樂的程式，但也許是我們從臉部識別有了更進一步的探討。

這次我們分別從網紅博恩以及同學的臉上找尋特徵，經過機器學習下成功控制臉部特徵。雖然部分臉部特徵依就有雜訊存在，但我們也確實有看出是人在動，我想這是一個很好的開始。也許在未來新聞播報和演說有可能在台上的不會是一個活生生的人，而是一個能透過 A I 合成出一個虛擬人物，有可能「它」是你喜歡的明星也說不一定。

十、 參考文獻

1. GAN 對抗式生成網路，Jakub Langr Vladimir Bok 著，哈雷譯，旗標，2020
2. Deep learning 深度學習必讀：Keras 大神帶你用 Python 實作，François Chollet 著，葉欣睿譯，旗標，2019
3. 用 TensorFlow 提早進入人工智慧的未來世界，李嘉璇著，佳魁數位發行，2017
4. Python 機器學習超進化:AI 影像辨識跨界應用實戰，鄧文淵著，基峰，2021
5. 深度學習:CNN 原理，<https://reurl.cc/vgZV5e>，Cinnamon AI Taiwan 著，2019
6. Python 影像辨識筆記(四)：使用 dlib 辨識器，<https://reurl.cc/2odpbE>，Yanwei Liu 著，2019
7. CNN 卷基神經網路，維基百科，<https://reurl.cc/DZVGve>