**Machine Learning Final Report**

第27組

0310515 葉尚畇

0310527 李韋辰

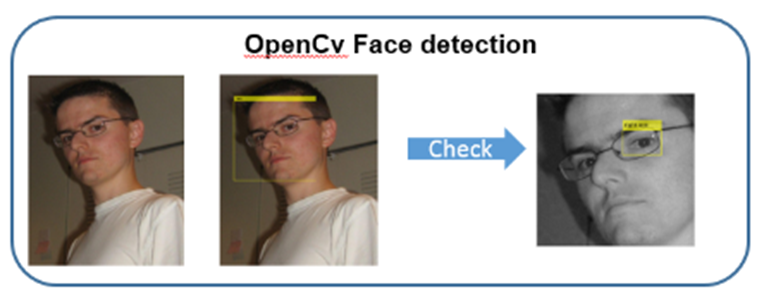
0310519 陳柏諺

在這次的期末報告中，分別對於影像輸入和轉換的前端端處理、初始所預期的訓練方式Neural Network+Random forest 和最後所使用的Convolutional Neural Network(CNN)進行詳細的闡述和參數的設定。因為一開始所預期的訓練方式為Neural Network+Random forest ，但是效果不如預期的理想，所以後來改用CNN的方式來完成，並且得到較佳化的結果。

* **前端處理**：

1. 利用OpenCV 的 face detector 找出臉部位置，若臉裡面偵測的到眼睛則確

定找到的為正確的人臉，並轉成灰階圖。

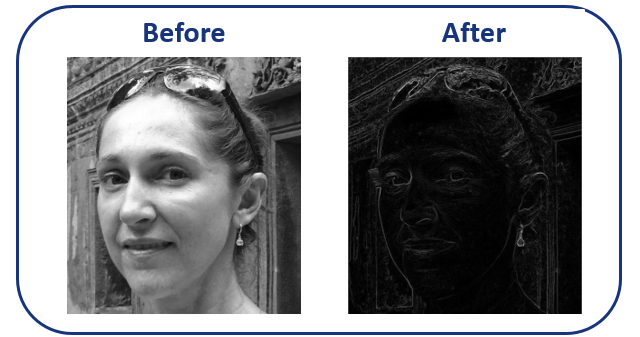


2. 對找的到人臉的圖使用Data Augmentation，平衡每一種資料的數量。

3. 再找一次人臉，其人臉輸出以臉跟眼睛的位置做判斷，擴大輸出的範圍。



4. 利用OpenCV 的 sobel function 取出臉的輪廓，期待降低背景雜訊干擾。



5. 將圖片縮為一定大小，讓後端處理。

* **後端處理**：
* **Neural Network+Random forest**

原本預期的做法**Neural Network+Random forest:**

**輸入的資料為灰階值0~255 的經過sobel filiter的matrix(60x60)，將資料調整灰階值從原本的0~255 同除上255進行歸一化，避免再neural network 中產生過大的數值。另外由於輸入的維度過高，導致在neural network 和weight 相乘後數值過大無法訓練，因此在weight 中同除上輸入資料量的維度(3600)，進行訓練。**

**訓練過程:**

1.以單層的 Neural Network 取出其特徵

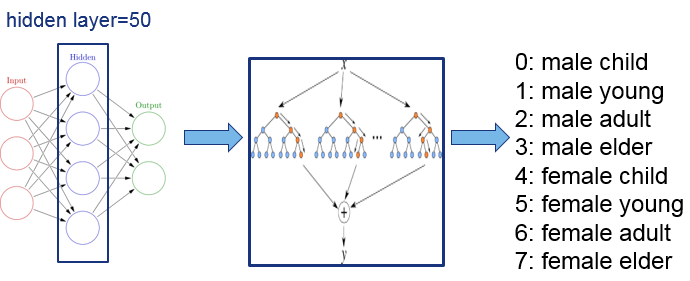
2.將 Hidden layer 當成特徵向量以 Random forest 的方式算出結果。

**分成兩種不同的分類方式**

1. 將全部不同資料同時代入Neural Network進行訓練，取出Hidden layer同時

去訓練random forest**一次分八類** 。

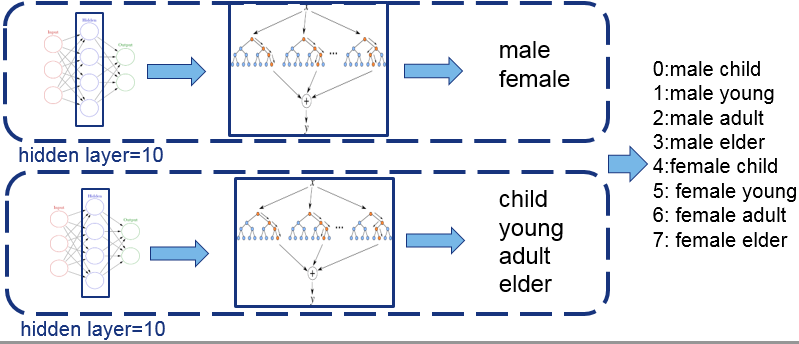
neural network 的層數設定為50層，random forest 設定1000顆的樹並且最小的葉子數為100



1. 將男女資料代入Neural Network進行訓練，取出Hidden layer去訓練r

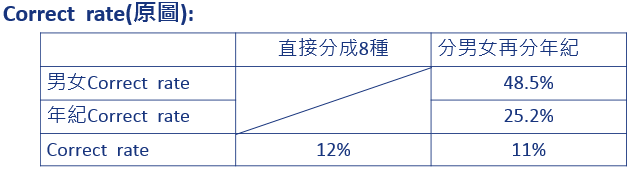
andom forest，接著再用不同年齡的資料代入Neural Network進行訓練，取出Hidden layer去訓練random forest，**每筆資料都具有男女和年紀的標籤，再進行最後的整合**。

兩種(男女&年紀)的分類neural network 的層數都設定為10層，random forest 都設定1000顆的樹並且最小的葉子數為100

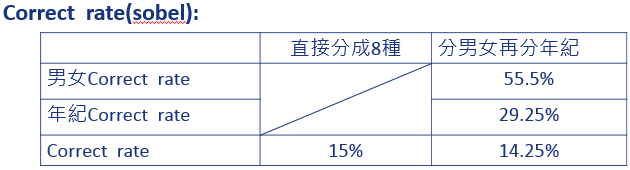


下方為對於不同的資料和分類方式所產生出來的結果，利用validation計算的正確率:

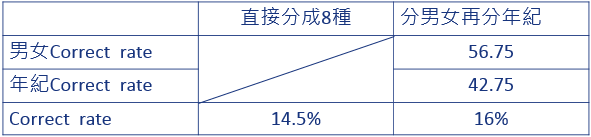
**Correct rate(原圖)**



**Correct rate(sobel)**



**Correct rate(原圖+ Data Augmentation)**



**討論**

從第一個表格和第二個表格來說，無論是直接分成8種還是分男女在分年紀，**取sobel 圖分類的正確率比直接用原圖分類的高**，所以可以推估sobel取邊緣確實可以提升正確率，而直接分成8種比分男女在分年紀得正確率來的高，可能是當男女得分錯下又在對年紀分時，建立在錯誤的基礎上，導致成效差，正確率降低。

從第一個表格和第三個表格中**，**對於利用Data Augmentation來平衡每一種資料的數量所產生的影響，可以從正確率中發現**Data Augmentation有助於分類的成效**，無論是直接分成8種還是分男女在分年紀。

藉由這樣**Neural Network+Random forest**的方法正確率十分的低，結果不如預期的理想，所以改用**Convolutional Neural Network(CNN)來訓練**

* **Idea 2-CNN**
* **前端處理**：

1. 利用OpenCV 的 face detector 找出臉部位置，若臉裡面偵測的到眼睛則確

定找到的為正確的人臉。

2. 對找的到人臉的圖使用Data Augmentation，平衡每一種資料的數量。

3. 再找一次人臉，其人臉輸出以臉跟眼睛的位置做判斷，擴大輸出的範圍。

4. 將圖片縮到 28\*28，並轉成灰階，**將灰階值規一化，並對於灰階值開根號**

**()→增加明暗之間的差距**

* **後端處理**：

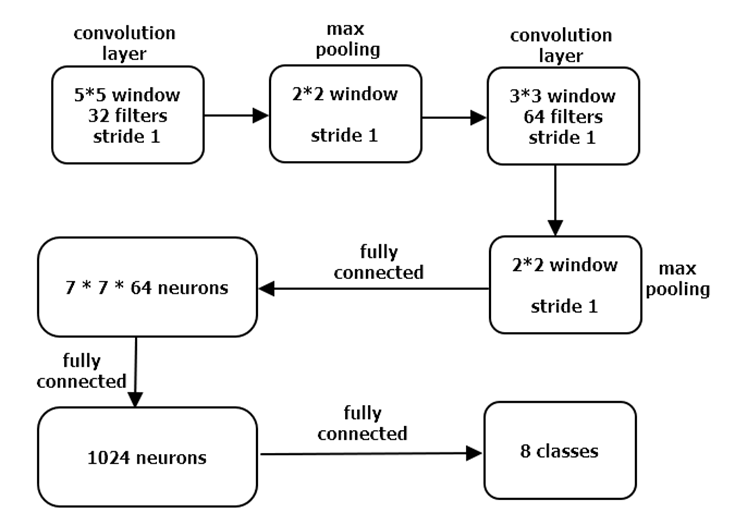
1.使用TensorFlow 建構 CNN 模型。  
 2.採用Mini batch增加準確度與運算時間 (size = 20)。

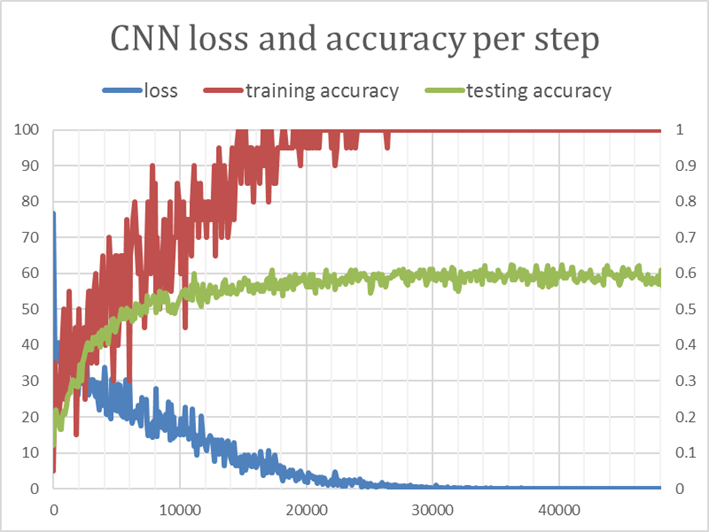
3.採用Bagging讀取Input資料。

4.使用AdamOptimizer來修正參數

5.learning rate 0.0001

6.架構：

  
 **目前準度最高約61% ，正確率遠大於初始所使用的Neural Network+Random forest**



**問題與討論:**

1.Data Augmentation 並沒有使資料量如預期的平衡，但確實使資料量平均了

一點點，將原本的資料擴增約一千左右，但是取完人臉後，資料仍然部分產生不平均。

2.我們使用偵測邊緣的工具Sobel，因把圖片轉成黑白，凸顯了邊緣的部分，但

也會使較複雜的背景被凸顯出來。再來此Filter會使臉上只要是有顏色陡峭

變化的地方都產生圖形，對整個判斷是加分還是扣分，還得再細一步研究才

能知曉。

3.CNN的參數設定難以找到最好的結果，只能不斷利用iteration的方式人工去

嘗試，耗費大量時間。

4. 臉部偵測的能力，難以上升，會使我們漏掉很多資料量。更可能使我們難以

判斷testing data。

5.在CNN的架構裡，增加資料後，正確率並無增加，反而從52%遞減到47%，

但在 Random forest的架構中，卻能從原本的11%上升到14%，可能是在cnn

架構中所產生的52%正確率是overfitting 的現象，因此當增加資料量時，正確

率下降。

6. CNN中跑太多次step後，會讓loss function跑到nah，使正確率下降，但在

此發生之前，跑到一定大小的step則能成功使test accuracy 上升。