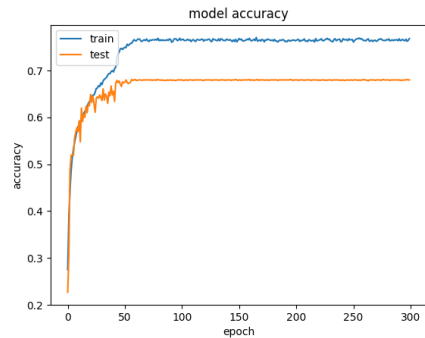


學號：R06944049 系級：網媒碩一 姓名：黃敬庭

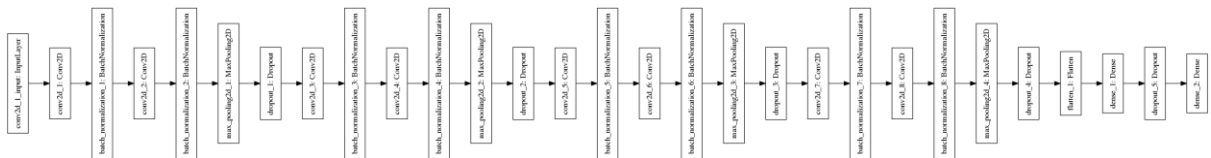
1. (1%) 請說明你實作的 CNN model，其模型架構、訓練過程和準確率為何？

(Collaborators:倪溥辰，吳政軒)

答：我使用一個 VGG-16 的 model 下去進行修改，這個架構主要由 c(conv layer) + c + p(pooling)為主架構，並且 filter(3*3)數量在每個 CCP 中不斷的變多，最終的架構是 (CCP*4)D(dense)D，我主要是將原本的 filter 變多,但將 model 變淺 (本來是設計的 input



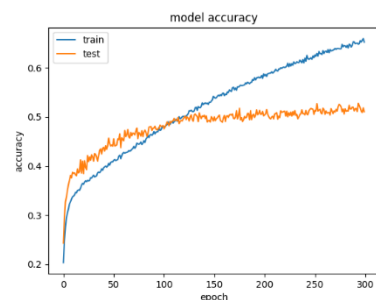
shape 為 224*224，而我們的是 48*48，無法 pool 那麼多次)，我在每個 activation layer 的後面都加入了 bn(batch normalization),然後在 CCP 塊後面與第一個D後面用上 dropout(0.25),在資料處理的部份會將 train & test data 都 normalize,用 ImageDataGenerator 生成測資(旋轉,平移,zoom in/out 等等),optimizer 使用 adam,然後切了 1/10 的資料出來做 validation data,如果經過幾圈 val_acc 沒有下降的話就將 learning rate 縮小,最後做出來的模型在 validation set 上面的 accuracy 約為 0.68,kaggle public 則為 0.694



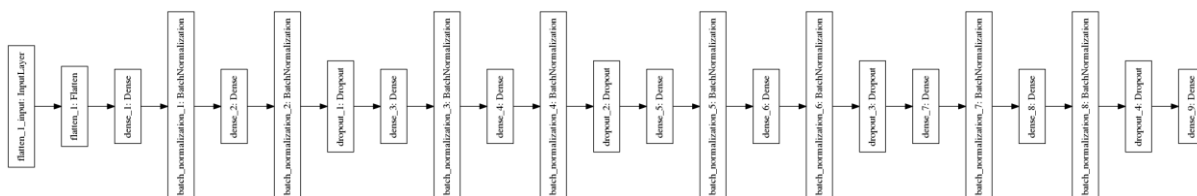
2. (1%) 承上題，請用與上述 CNN 接近的參數量，實做簡單的 DNN model。其模型架構、訓練過程和準確率為何？試與上題結果做比較，並說明你觀察到了什麼？

(Collaborators:)

答：CNN 的參數量為 4.2M 個左右,NN 則使用到了約莫 4.5M參數,我使用了 9 層的 D(最後一層為輸出),一樣會在 activation 之後接上 bn，然後在 2 個D之後會使用 dropout(0.5),其他的細節都與



CNN 的模型相同,最終 validation acc 為 0.51,相較於 NN,我覺的在圖形的辨識上，CNN 佔有更大的優勢,因為他會用不同的 filter 去對圖片上不同的區域特徵進行辨認,但在參數數量上差不多的狀況下，CNN 要跑一個 epoch 的時間也比 NN 來得多許多，但很顯然地的在這次的作業當中，CNN 會更能勝任

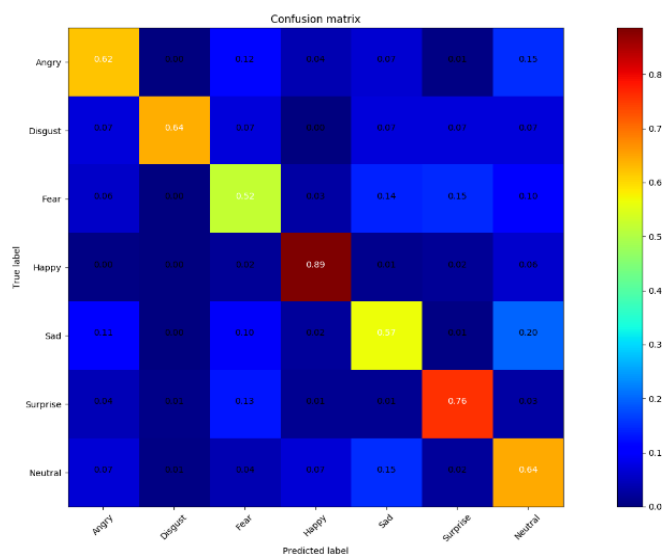


3. (1%) 觀察答錯的圖片中，哪些 class 彼此間容易用混？[繪出 confusion matrix 分析]

(Collaborators: 倪溥辰，
吳政軒)

答：依照最後畫出的

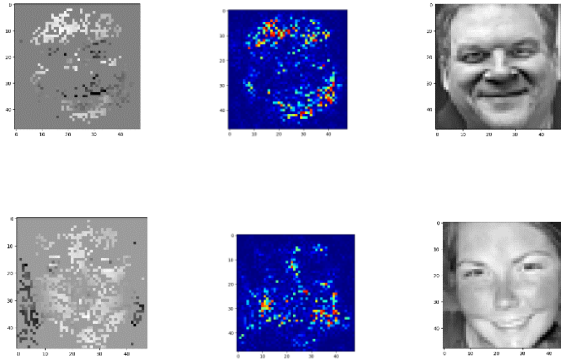
Confusion Matrix,發現我的 model 在 happy(0.89)上面答對的機率最高,在 fear (0.52)和 sad(0.57)答對的機率相對比較低,然後 sad 有 20%的機率被判定為 Neutral 算是相對比較容易混淆的,而 Disgust 最不容易混淆其他的 label,其他 label 要被誤判為 Disgust 的機率幾乎是 0



4. (1%) 從(1)(2)可以發現，使用 CNN 的確有些好處，試繪出其 saliency maps，觀察模型在做 classification 時，是 focus 在圖片的哪些部份？

(Collaborators: 倪溥辰，吳政軒)

答：

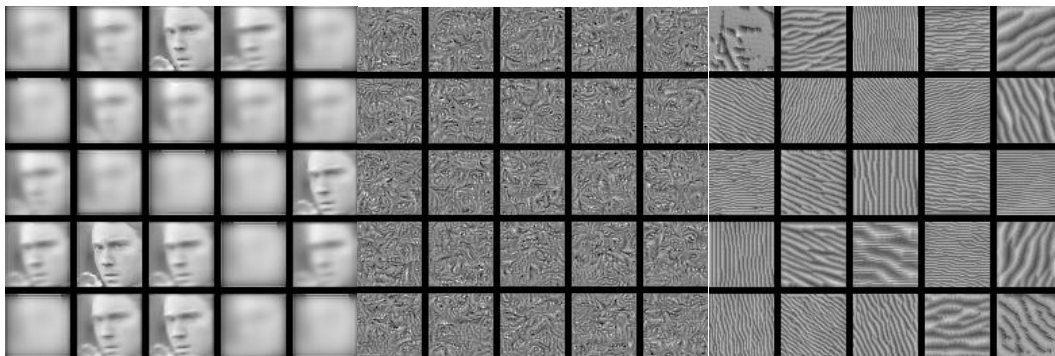


這題選用了一個標記為 happy 的 image 來顯示(因為分析 Confusion Matrix 對 happy 的正確率最高),感覺對於這個 class filter 會 focus 在臉的下半部和臉頰，眼睛好像反而並沒有那麼重要

5. (1%) 承(1)(2)，利用上課所提到的 gradient ascent 方法，觀察特定層的 filter 最容易被哪種圖片 activate。

(Collaborators:)

答：



這是對比第一個 conv, dropout, pooling 用 gradient ascent 做出來，隨機挑出來的 filter 的 visualization，conv layer 幾乎都包含整張圖，但是圖片中的材質清晰則各有所不同，dropout 因為是隨機的所以感覺幾乎像是雜訊，而 pooling layer 視乎比較多紋理，但也夾雜一些不太一樣的資訊