

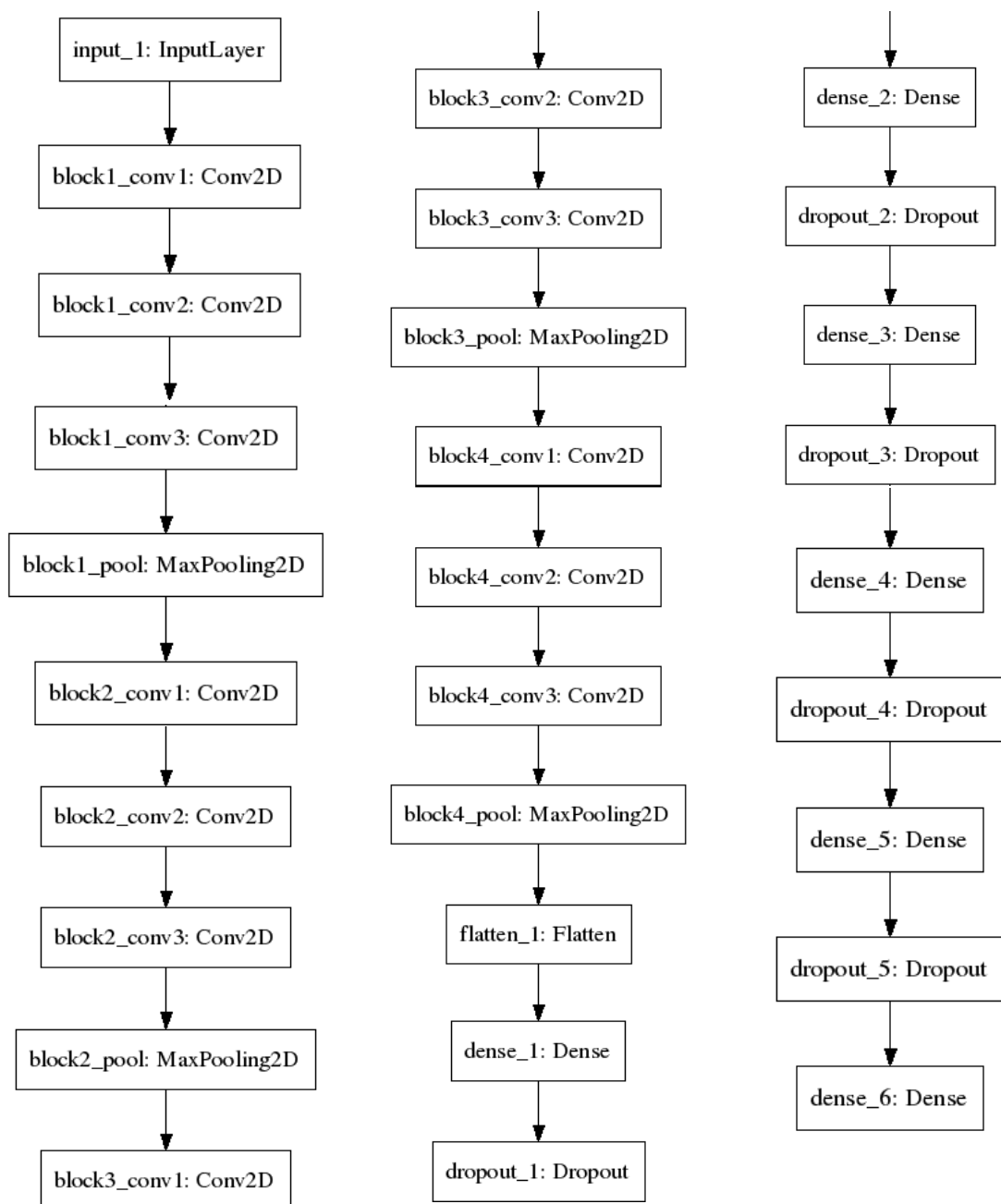
Machine Learning HW3

學號：R05943005 系級：電子所碩一 姓名：呂丞勛

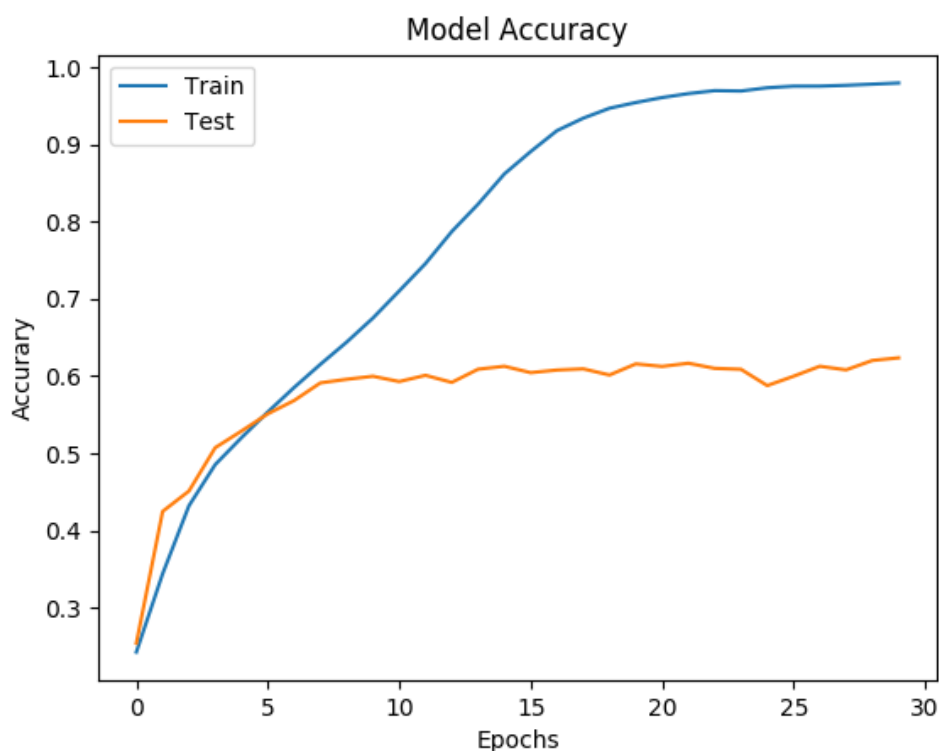
1. (1%) 請說明你實作的 CNN model，其模型架構、訓練過程和準確率為何？

答：

模型架構：convolutional layer 順序為 conv2D (size: 3x3x32) x3 + maxpooling + conv2D (size: 3x3x64) x3 + maxpooling + conv2D (size: 3x3x128) x3 + maxpooling + conv2D (size: 3x3x512) x3 + maxpooling + 5x fully connected layer (size: 2048, 1024, 512, 256, 128) + output layer



模型的訓練過程如下圖所示 (為準確率的判斷，大約切出 20% 的 training data 當成 validation data)，validation 的準確率大概為 60%，與上傳 Kaggle 的分數 59.125% 相當接近。



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？
答：

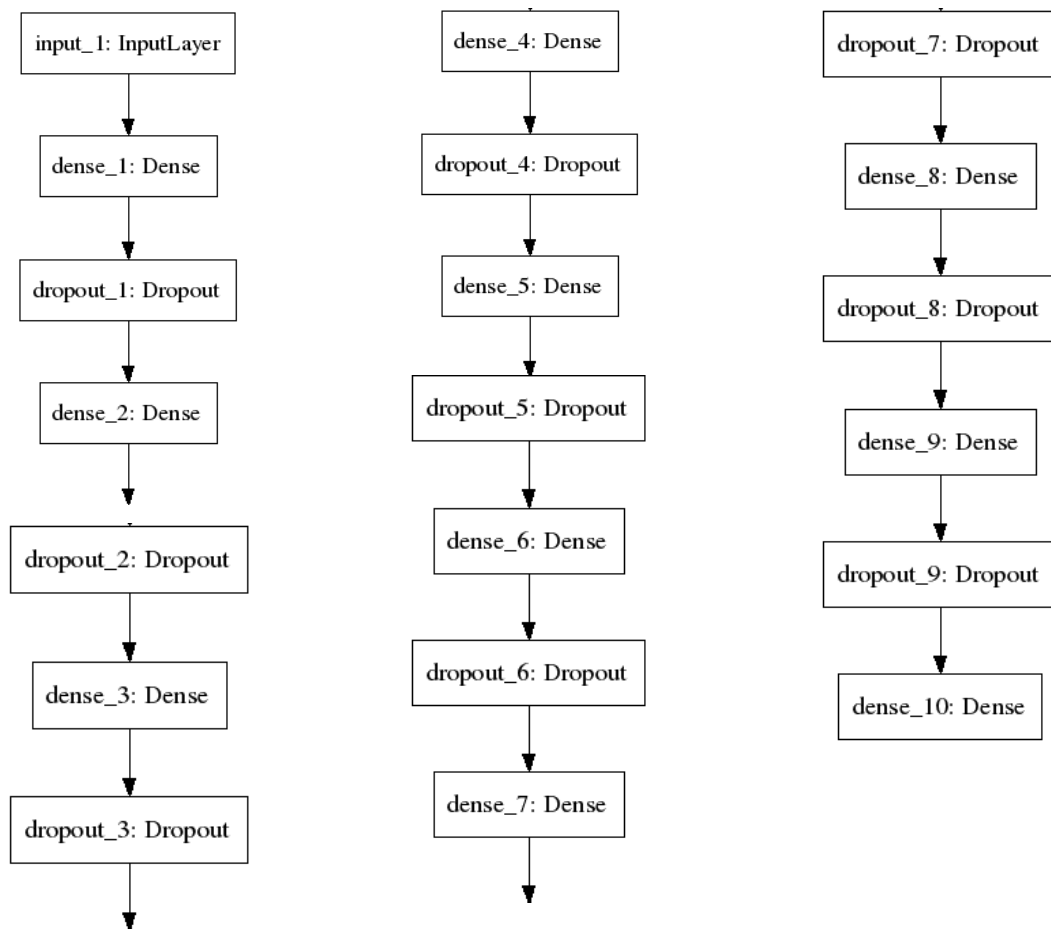
(1) 第一題中，CNN 的參數量如下圖所示

```
=====  
Total params: 18,017,479.0  
Trainable params: 18,017,479.0  
Non-trainable params: 0.0
```

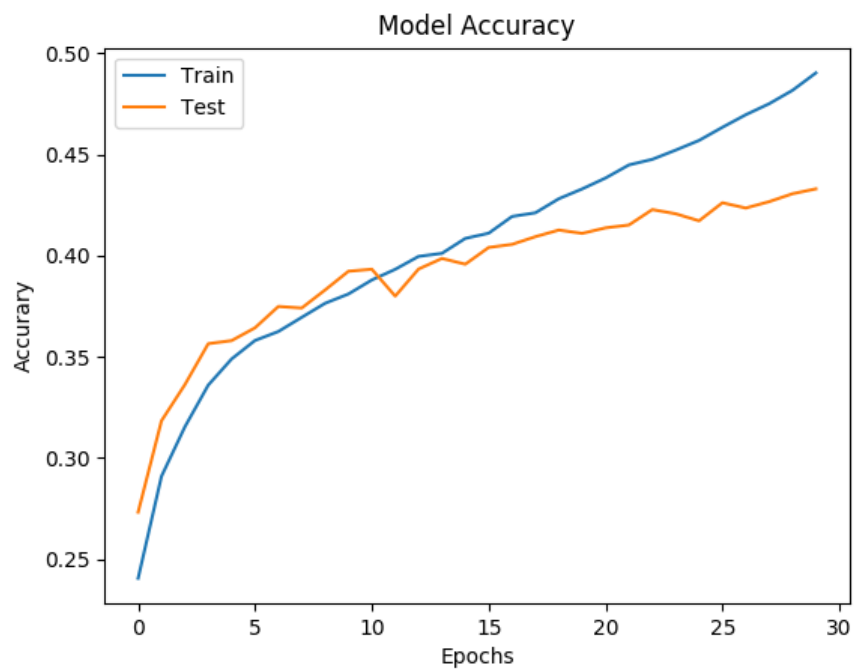
(2) 因此採取 DNN 的參數如下

```
=====  
Total params: 18,000,647.0  
Trainable params: 18,000,647.0  
Non-trainable params: 0.0
```

結構為下所示



Training processing

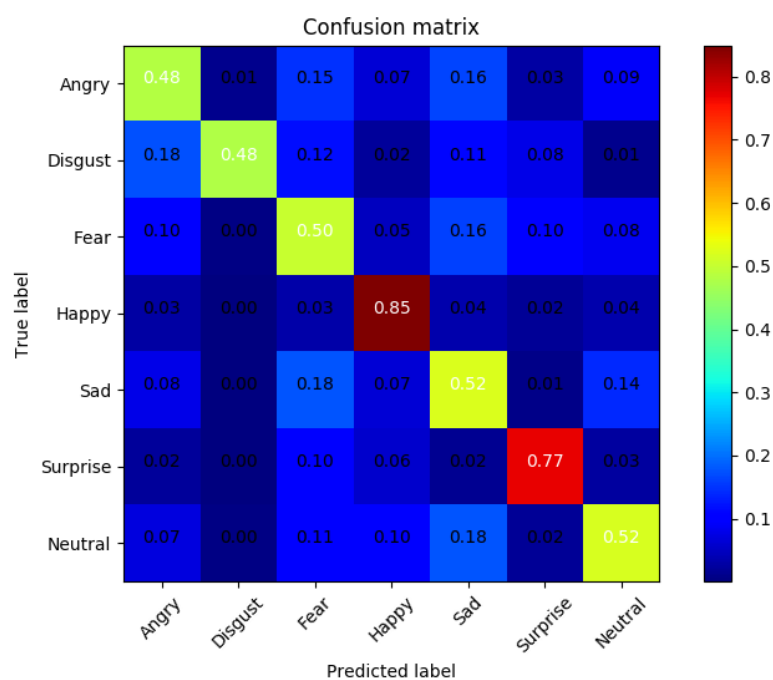


可以看出，若使用 DNN 的模型，反而效果沒有比較好，training accuracy 只有 49%，validation accuracy 也只有 43%，因此在圖片上，CNN 有較好的表現。

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]

答：

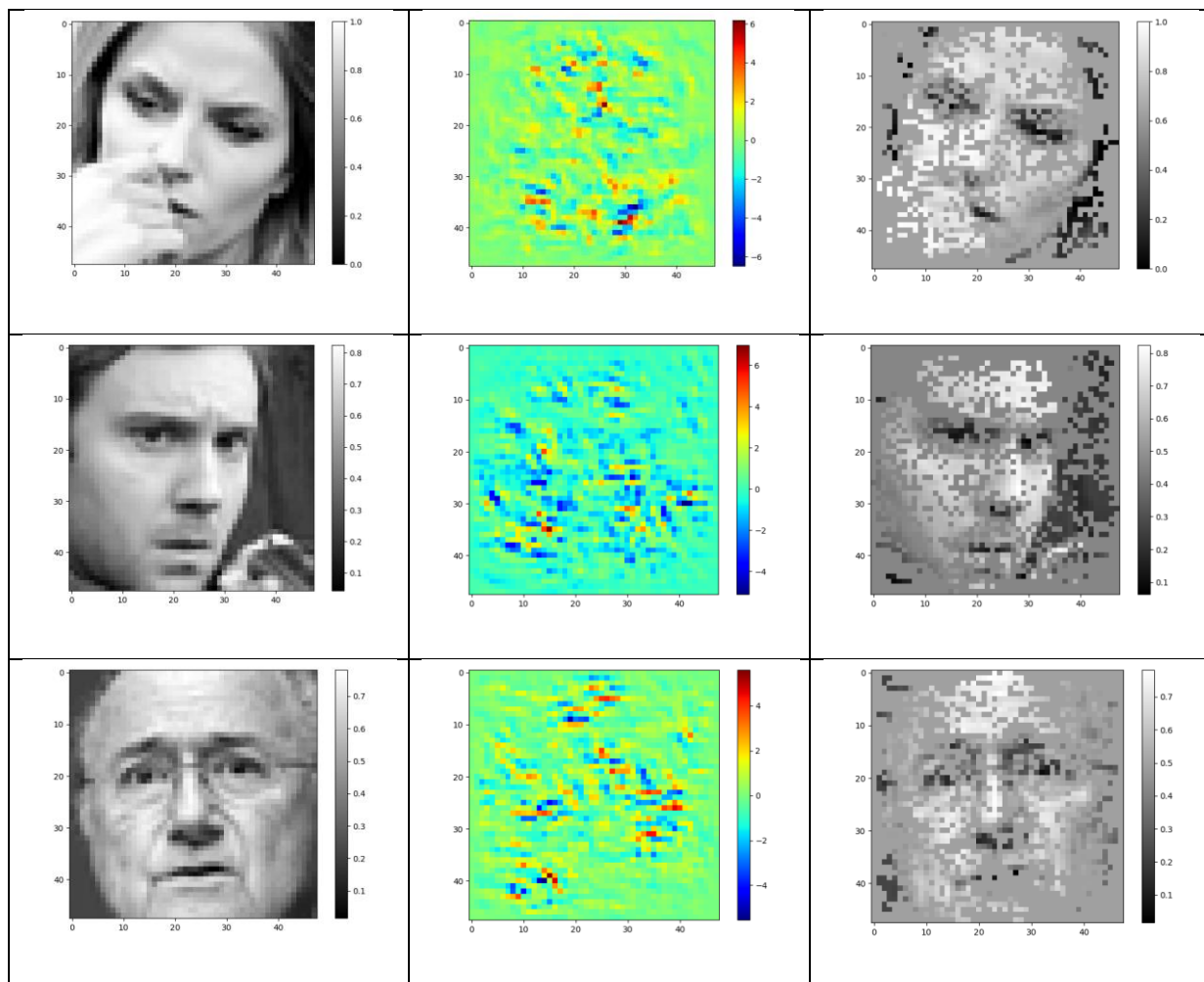
透過以下 confusion matrix 的分布來看，可以看出大部分的圖片都可以被分到對的種類(對角線的 probability 最高)，而其中，angry 與 disgust 的圖片在七個種類中最容易被分成其他類，happy 和 surprise 則最容易分對。其中最容易分錯的 classes 為(disgust, angry)，(fear, sad)，(surprise, fear)，(neutral, sad)，



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？

答：

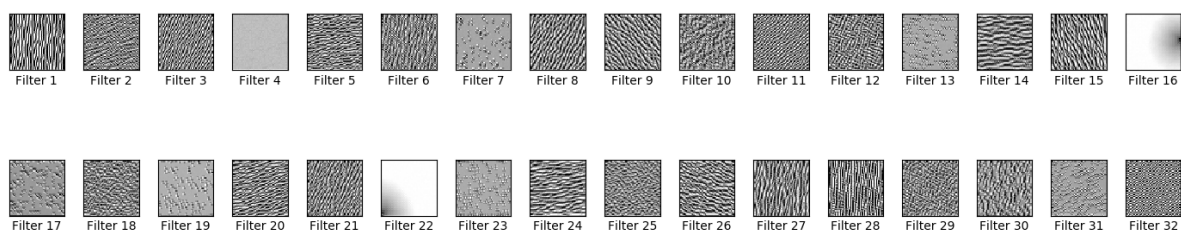
從下面的表格中，選出三個圖片來分析，可以發現模型大部分都可以準確的 focus 在各個圖片中的臉部(其中又以眼睛與鼻子的區域最為明顯)，這表示利用 CNN 可以利用 kernel 來專注於區域特徵的擷取，因此在影像上有相當好的表現。

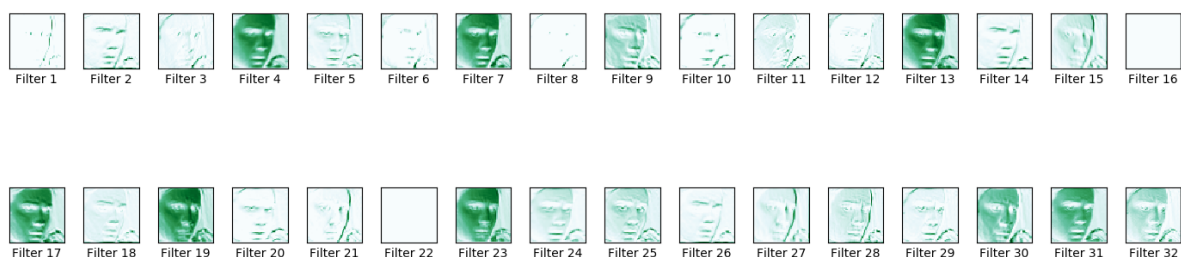


5. (1%) 承(1)(2), 利用上課所提到的 **gradient ascent** 方法, 觀察特定層的 **filter** 最容易被哪種圖片 **activate**。

答：

block1_conv1 的 **filters** 如下, 一開始的 **filters** 幾乎都是在判斷有沒有直線, 不同角度的直線特徵。而輸入第一張圖片來看結果, 大致上並沒有什麼改變, 有將臉部的特徵稍微擷取出來。





Block2_conv1 的 filters 如下，此層已經較為中段，判斷的特徵也從直線到有點曲線，或是一些點狀的結構。將第一張圖片輸入，可以發現，不同于上述前面的 convolutional layer，該層的結果比較可以 focus 在特定的區域，像是眼睛、鼻子、嘴巴、眉毛等區域，可以看出抓取的重點越縮越小，進而得到更高層級的特徵。



[Bonus] (1%) 從 training data 中移除部份 label，實做 semi-supervised learning

這題使用 testing data 當成 unlabeled data，來實作 self-training，CNN 結構如第一題，而每次對 unlabeled data 作預測時，將 prediction probability 的結果大於 0.98 的 data 作 label，在利用這些 labeled data 重 train 一次，反覆此步驟 10 iterations，得出在 Kaggle 上的預測準確率達到了 65.534%，可以看出 self-training 用在 testing data 上可以讓準確率上升。另外，透過 semi-supervised 的訓練出來的模型中，挑出 4~5 個 MODELS 來做 majority vote，又可以讓正確率在上升 2.1% 左右。

[Bonus] (1%) 在 Problem 5 中，提供了 3 個 hint，可以嘗試實作及觀察 (但也可以不限於 hint 所提到的方向，也可以自己去研究更多關於 CNN 細節的資料)，並說明你做了些什麼？ [完成 1 個: +0.4%, 完成 2 個: +0.7%, 完成 3 個: +1%]