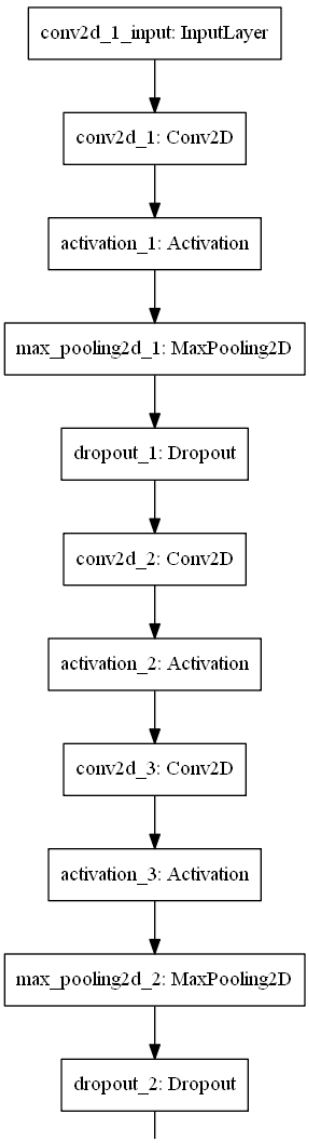
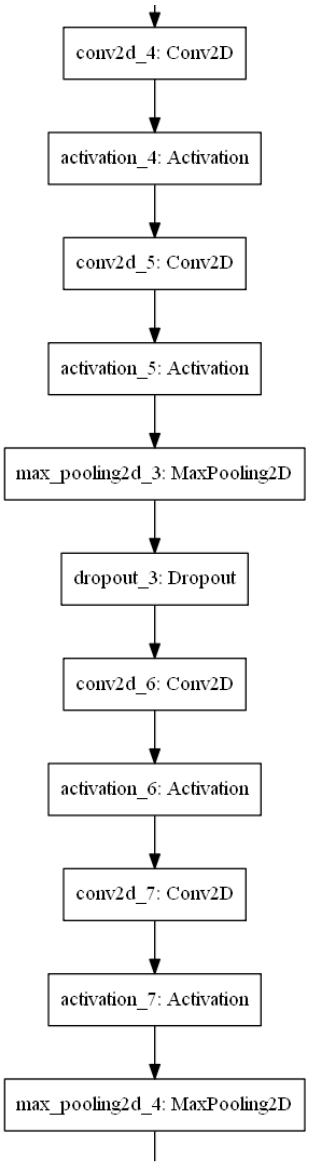
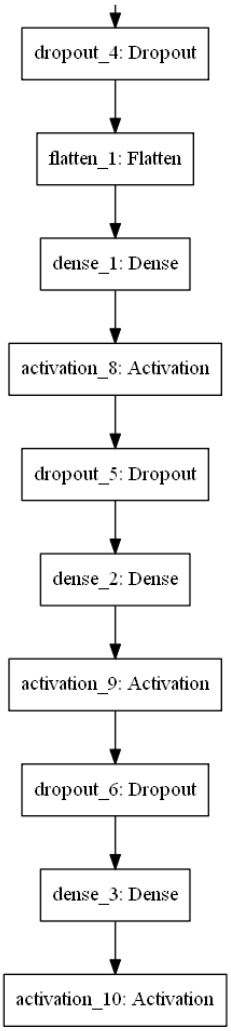
學號：R05921077 系級： 電機碩一 姓名：陳立杰

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：

Model structure:



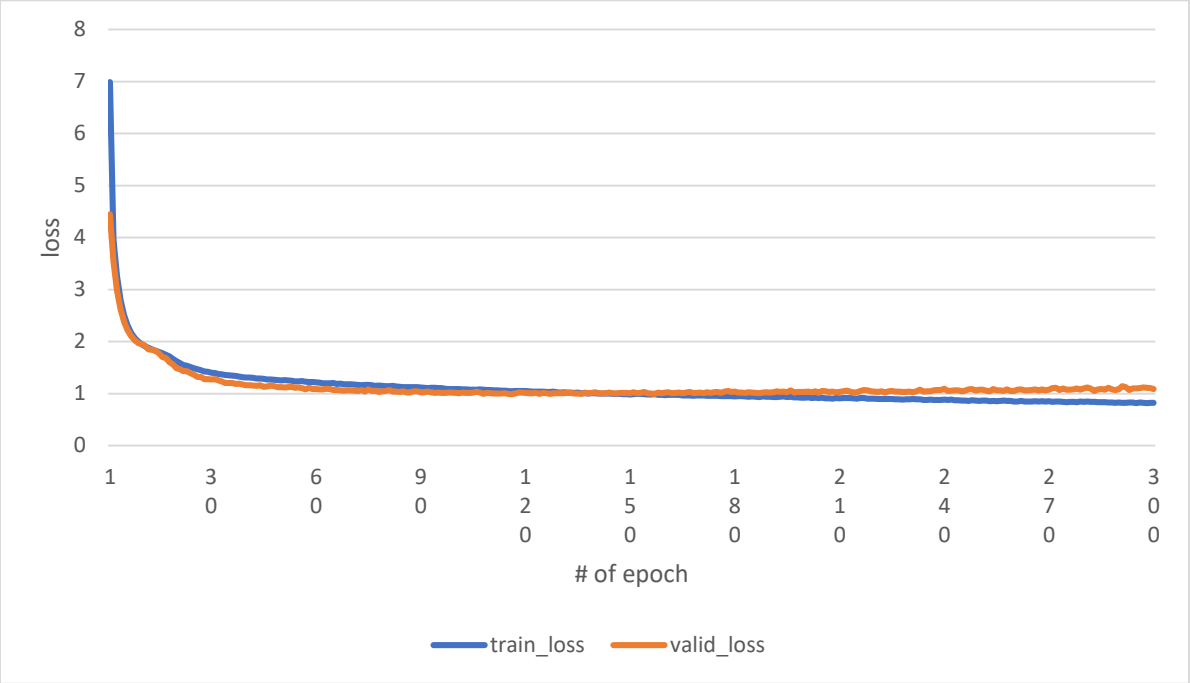




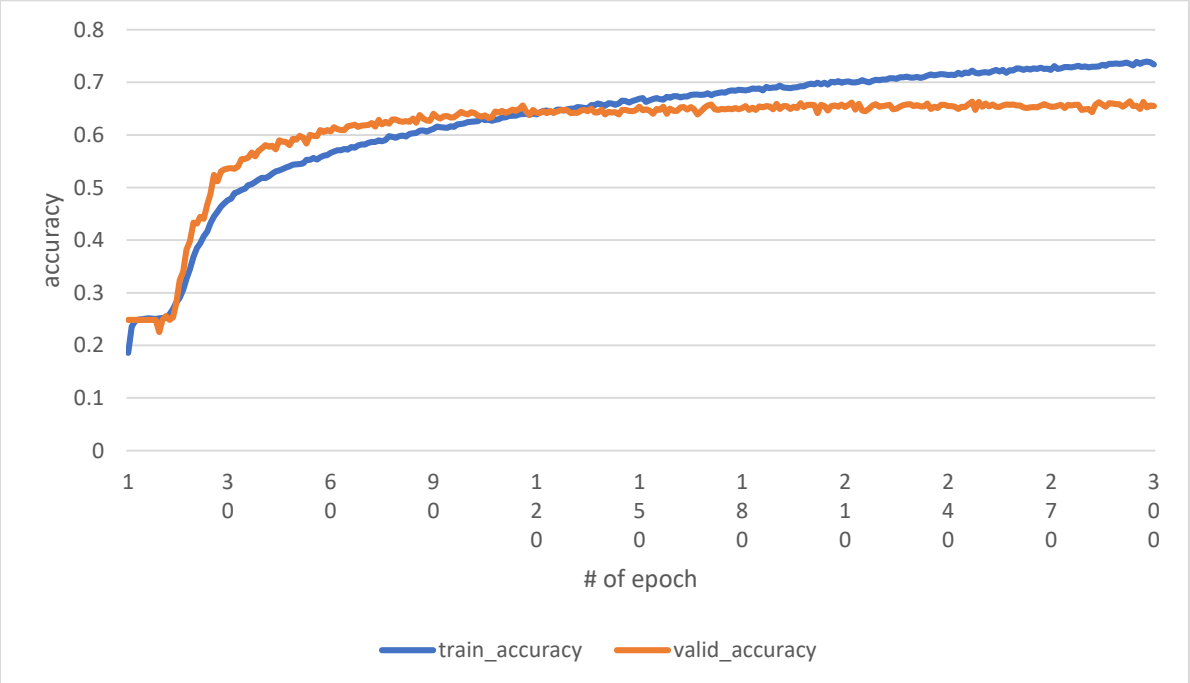
其中除了最後一個 layer 的 Activation function 是 softmax 其他全部都是 relu，dropout rate 都是 0.5，filter size 都是 3*3，max pooing size 都是 2*2， convolution layer 的 padding 都為 same，cond2_1,2,3,4,5,6,7 分別有 32,64,64,128,128,256,256 個 filter，dense_1,2,3 分別有 128,128,7 個 neural，總參數量是 1,474,887。

Training procedure and Accuracy:

Input date 的部分我有把所有圖片做水平翻轉，所以 data 數會是原本的兩倍，這樣大概可以增加 0.01 左右的準確度，其中最後 1%也就是 5742 筆資料會拿來做 validation。Optimizer = adam、loss = categorical_crossentropy，在 train 的時候 dense_1 跟 dense_2 有做 L2 regularization with lambda=0.01，batch size= 1000，共跑了 300 個 epoch。



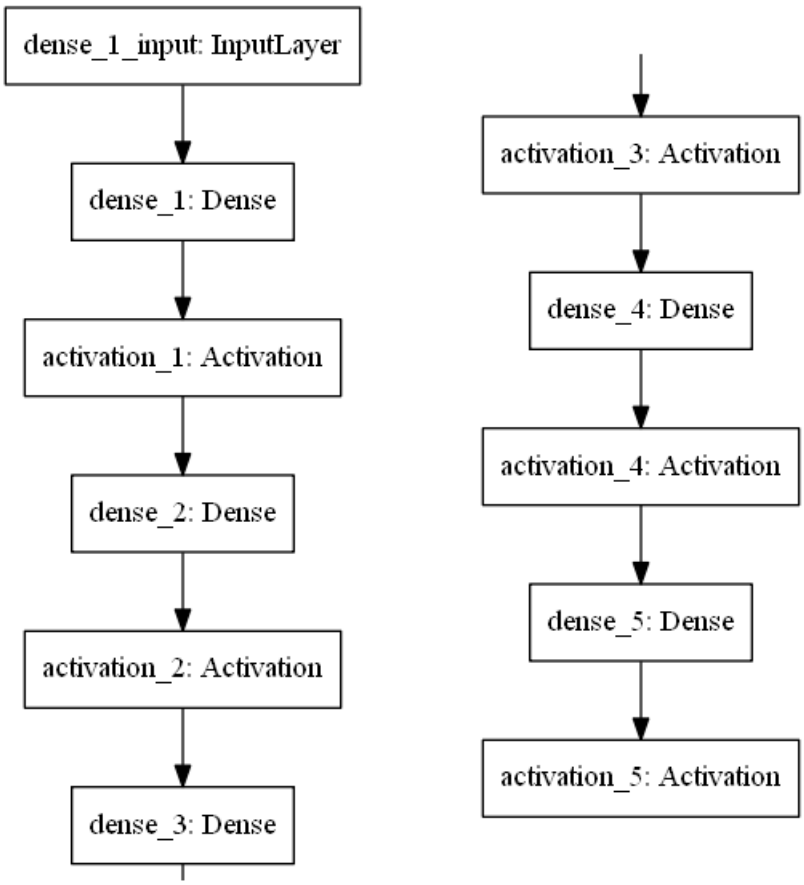
從上面這張圖可以看出 validation 一開始的 loss 會比較低，因為 training 這邊 drop 掉很多的 neural，但大概在 140 個 epoch 時 valid_loss 就不會再繼續下降，甚至有些微上升，而 train_loss 這邊則是會持續緩慢下降變得比 valid_loss 還低。



從上面這張表也可以看到一開始 validation 的 accuracy 比較高，training 跟 validation 這邊大概在 120 epoch 左右交叉，有趣的是，剛剛在分析 loss 時 valid_loss 大概在 140 epoch 後開始緩慢上升，但在 valid_accuracy 這邊 140 epoch 後 valid_accuarry 反而也是緩慢上降而不是下降，大概平均可以從 0.64 上升到 0.65 左右就到了 local minimum。最後在 validation set 最多可以達到 0.662 左右的 accuracy，在 public set 則可以達到 0.65729。

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：

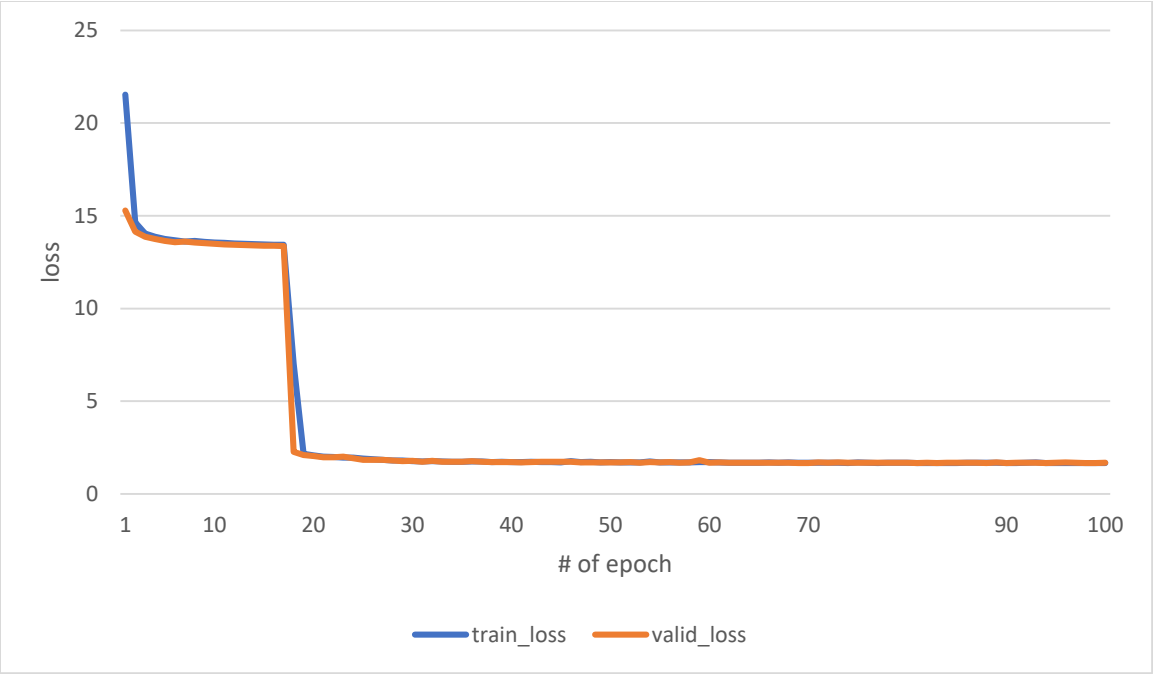
Model structure:



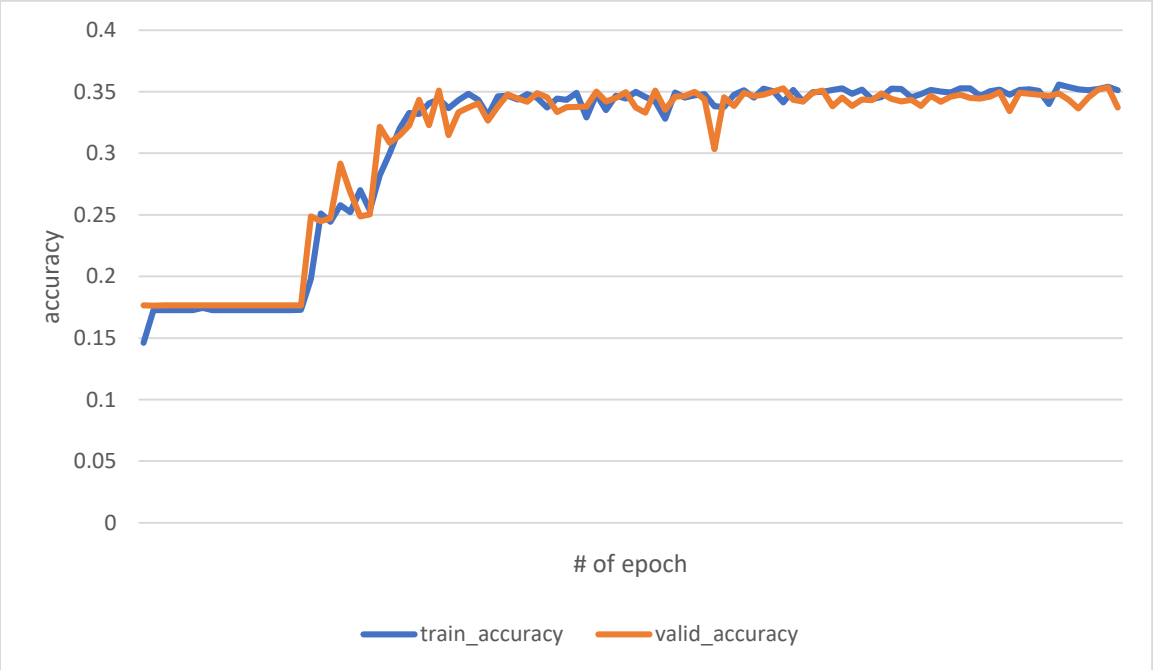
其中除了最後一個 layer 的 Activation function 是 softmax 其他全部都是 relu，dense_1,2,3,4 都是 415 個 neural，dense_5 有 7 個 neural，總參數量是 1,477,407。

Training procedure and Accuracy:

Input date 的部份跟 CNN 一樣把所有圖片做水平翻轉資料量兩倍， Optimizer = adam、loss = categorical_crossentropy，參數量和先前也差不多，共 5 個 layer，我有試過 3 跟 7 個 layer 參數量差不多但都幾乎 train 不起來。前 4 個 layer 都有做 L2 regularization with lambda= 0.01，如果沒加 regularization 也是 train 不起來，dropout 的部分都沒有加，因為連 training set 都 train 不大起來。Batch size=1000，總共跑了 100 個 epoch。



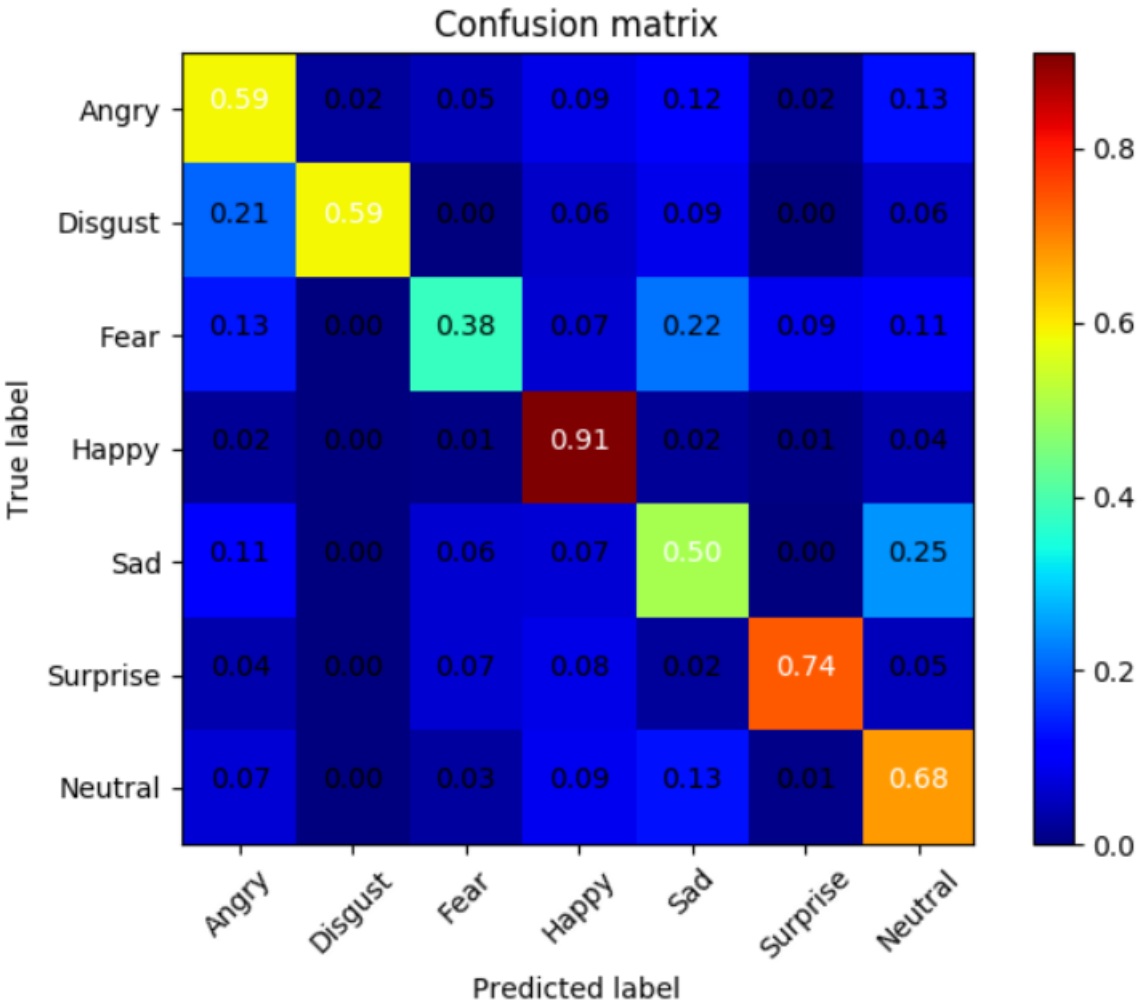
從上面這張表可觀察到在 loss 的部分因為沒有 dropout 所以 training 跟 validation 的線條前面沒有什麼差距，但 loss 不像 CNN 是穩定下降，比較特別的是用 DNN 來做居然到後面的 epoch 的時候 training 跟 validation 的 loss 幾乎完全一樣。



雖然在 loss 的部分 validation 跟 training 的線條長得差不多，但在 accuracy 差很多，尤其是在 validation 這邊的變化很大很不穩定，相較起來 CNN 的線條平滑許多。最後 train_accuracy 和 valid_accuracy 都沒辦法 train 到很高，大概都卡在 0.35 左右就不能繼續往上升了，相同的參數下，CNN 的準確度比 DNN 高多了，不過 DNN 跑的速度稍微快一些。

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：下面這圖表是我做出來的 confusion matrix:



從這個 confusion matrix 中可以看到大部分的表情猜測分佈都蠻正常的，但在 fear 這個 class 分佈比較沒那麼準確，他猜測的正確率只有 38%非常低，而且跟猜測 sad 這個 class 的比例有些接近有 22%，代表 fear 這個 class 很容易跟 sad 搞混猜錯。有趣的是在 sad 這個 class 並沒有這個現象發生，sad 猜成 fear 的比例只有 6%，代表 fear 這個 class 的集合可能有蠻多圖片跟 sad 相似，但這些圖片只占 sad 裡圖片的一小部份。

下方列出 fear 這個 class 挑出幾張圖片猜測的 distributions:

[0.128913 0.002805 0.247318 0.012802 0.258703 0.025681 0.323774]

[0.468328 0.008440 0.296922 0.014612 0.158803 0.008246 0.044647]

[0.070723 0.001426 0.418189 0.020905 0.420144 0.008868 0.069742]

[0.105425 0.022907 0.710739 0.002698 0.153301 0.003485 0.001446]

[0.104237 0.002137 0.523610 0.008565 0.312766 0.007666 0.041017]

[0.148756 0.014702 0.297128 0.052065 0.129628 0.279537 0.078184]

所有的 fear class 的 validation set 之 distribution 的平均是:

[0.14781371 0.00611387 0.36290521 0.0665379   0.20754447 0.09830372 0.11078112]

從這些數據可以觀察到，確實 fear 跟 sad 這兩個 class 的機率幾乎都是最高的，其中第一個 distribution 的圖片是:

這張圖看起來好像是難過但又有點恐懼的感覺，所以它猜的機率在這兩個 class 幾乎是相等的，但事實上猜最高的機率是 neutral，我猜可能是因為沒有一個特定的特徵所以後來就分配到中立那邊。

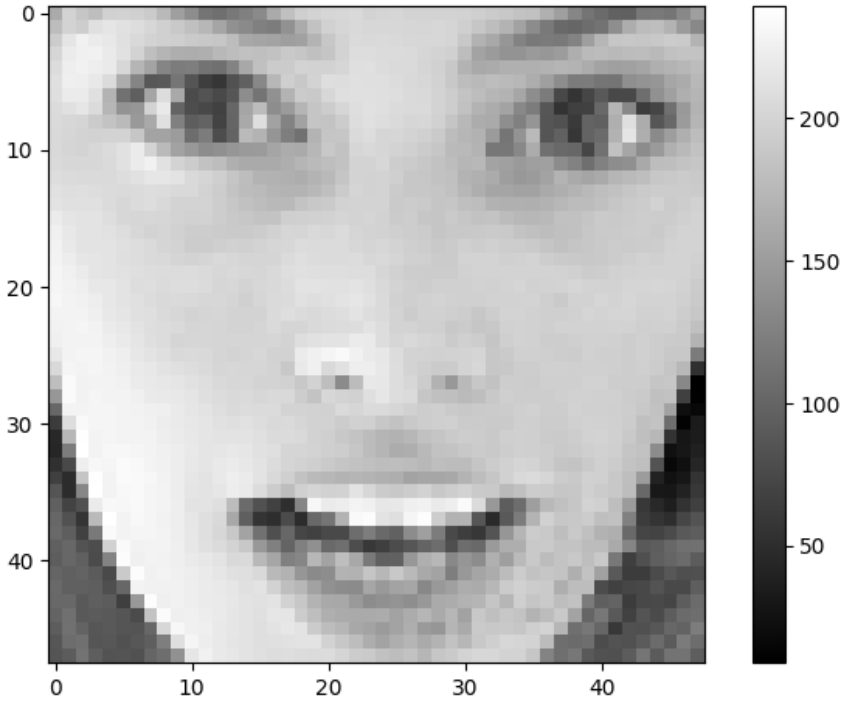
而所有的 sad class 的 validation set 之 distribution 的平均是:

[0.1392387   0.00378177 0.14761563 0.06864173 0.41665545 0.00670677   0.21735994]

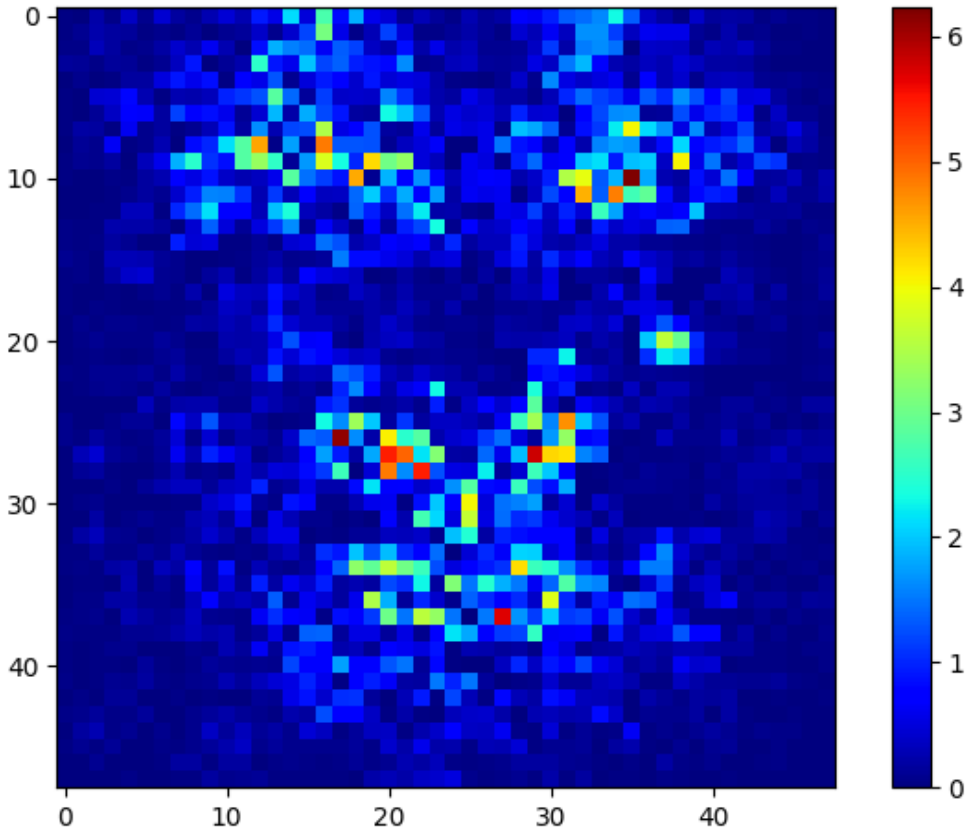
這邊猜 fear 跟 sad 的機率差很多，所搞混的機率就低很多。

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：

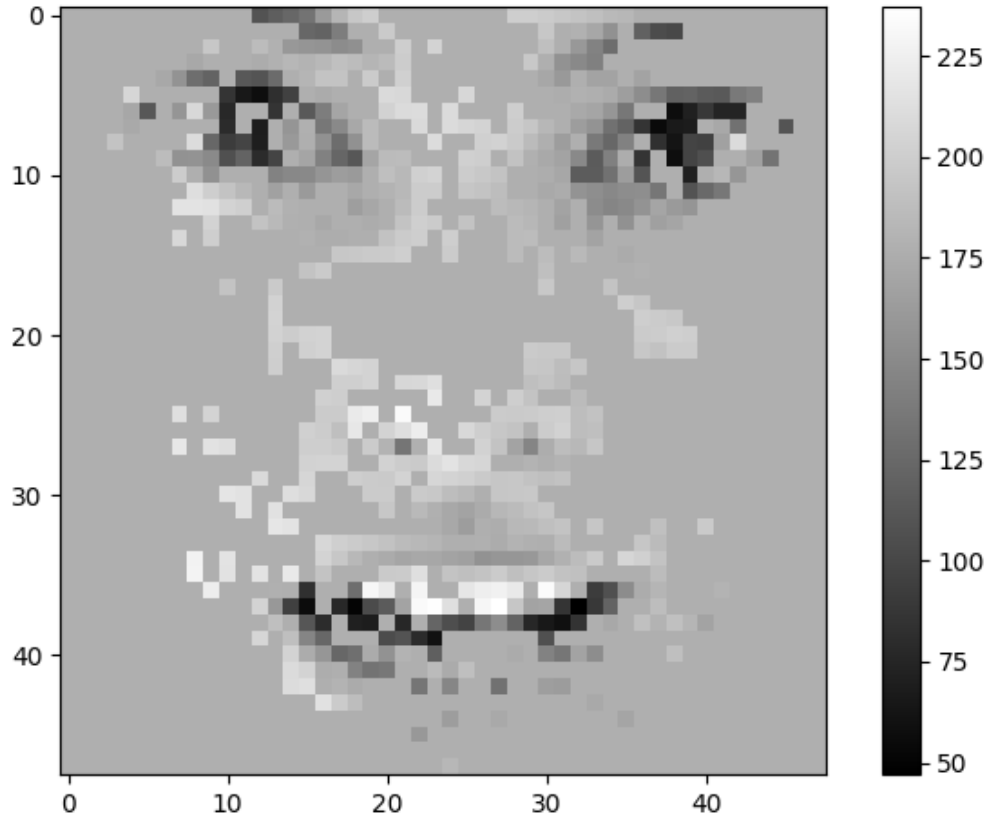
原圖:



Saliency Map:



Mask 掉 heat 小的部份:

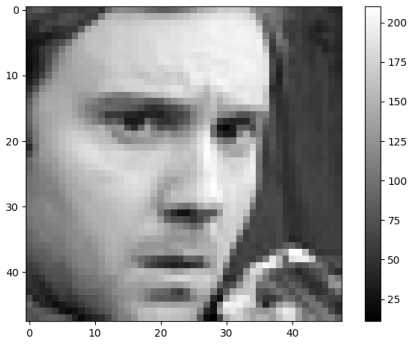


從這三張圖很明顯的可以看出主要都是 focus 在臉部的眼睛、鼻子和嘴巴這三個部位。

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：

觀察 conv2d_7 的第一個 filter

White noise:  Image: 

原圖: 

從 white noise 做出來這張圖片感覺他是要偵測某些左右方向的長條形狀，而從 image 做出這張圖片就可以觀察到，他的眼睛鼻子嘴巴都變得很寬，感覺就是要偵測臉部比較會左右延展的表情，但 activate 後看起來就會變很奸詐的樣子。

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

[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%]

1.我找了一個在 validation set 準確率只有 0.4377 的 model 在某一個 filter 做 activate 來比較:

White noise:  Image: 

從這兩張圖片幾乎看不出什麼規則或規律，很難去猜測它想辨識什麼。

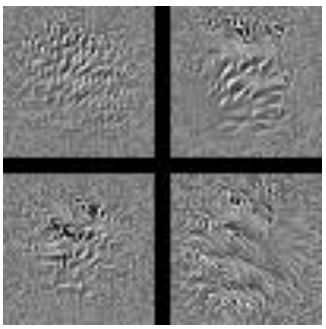
2.  這張圖是由 white noise activate 第 0 個 class 的結果，跟老師上課時做的結果差不多，看不大出來什麼圖形。

3.我把一張圖片拿來 activate 各種 class，做出來的結果還蠻有趣的，下面幾張圖對應的是生氣、厭惡、恐懼、高興、難過、驚訝、中立。

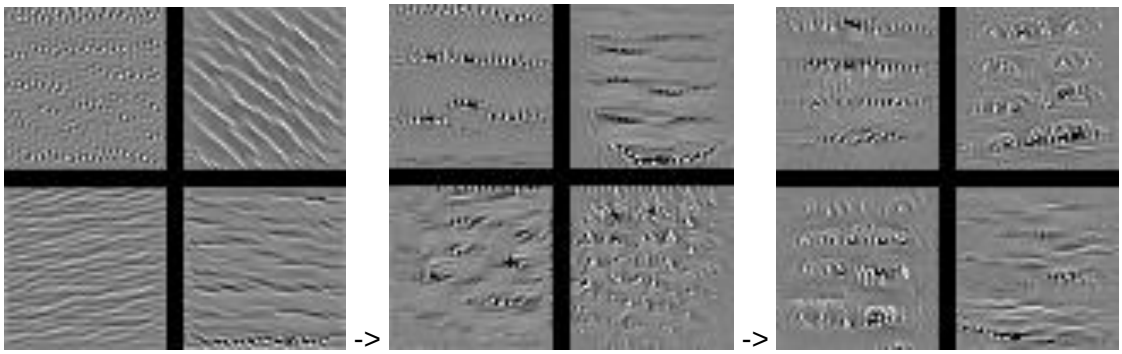
生氣的圖可以看到他眉頭深鎖很明顯是生氣的臉，厭惡跟恐懼也大概看的出來，高興我覺得是最明顯的，看他的嘴角跟眼睛都很明顯是高興，難過的圖嘴角下垂也很容易判斷，驚訝的圖嘴巴張得很開也看的出來，但中立就看不大出來了，嘴巴好像在笑，眼睛又有點嚴肅加難過的感覺。

4.Maxpooling 的重要性: 我一開始一直沒辦法過 strong baseline，後來把 maxpooling 加到四層後就成功通過了 strong baseline，後來在觀察每一層 filter 圖片時我發現每做一次 maxpooling 後他的圖就會明顯變的比較複雜，在相同的 layer 中跟只有兩層 maxpooling 的圖相較起來也複雜多了。

只有兩層 maxpooling 的最高 layer 的圖片:



四層 maxpooling 的圖片:



可以看到四層 maxpooling 的最高 layer 的 filter activate input 後所產生出來的圖(下排最右邊那張)比兩層複雜很多 ，從這三張圖就可以觀察到每次經過 maxpooling 後所產生的圖就會比原的本更加複雜。