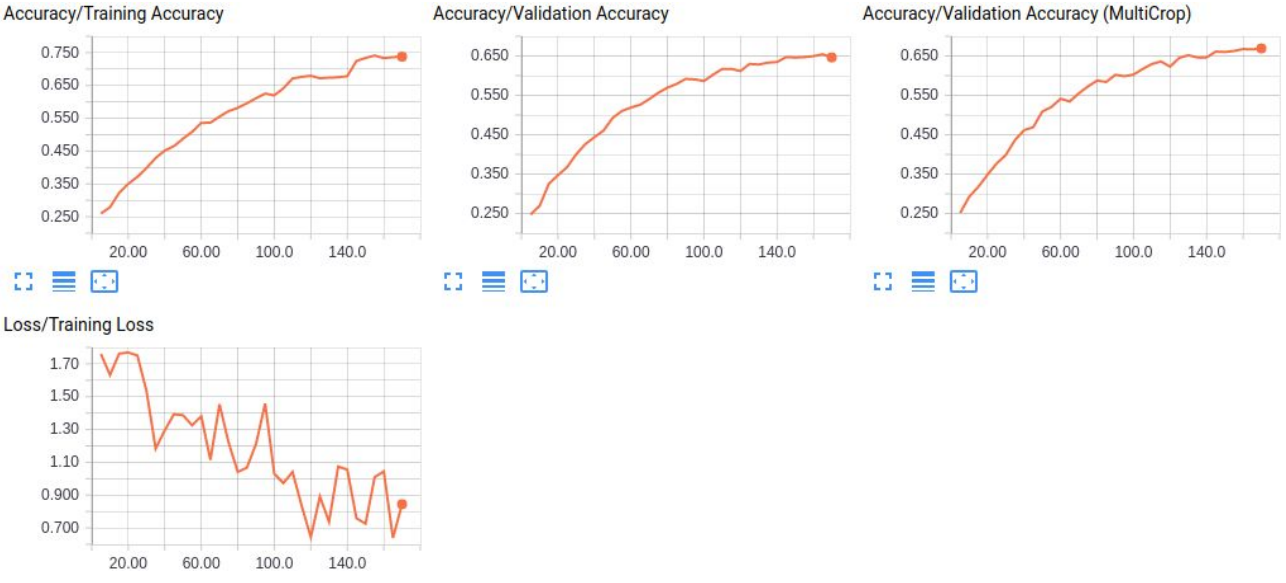
學號：R06922128 系級： 資工碩一 姓名：楊碩碉

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？

```
conv1 = conv(image, [3, 3, 1, 64])
conv1_2 = conv(conv1, [3, 3, 64, 64])
pool1 = pool(conv1_2)

conv2 = conv(pool1, [3, 3, 64, 128])
conv2_2 = conv(conv2, [3, 3, 128, 128])
pool2 = pool(conv2_2)

conv3 = conv(pool2, [3, 3, 128, 256])
conv3_2 = conv(conv3, [3, 3, 256, 256])
conv3_3 = conv(conv3_2, [3, 3, 256, 256])
pool3 = pool(conv3_3)

conv4 = conv(pool3, [3, 3, 256, 512])
conv4_2 = conv(conv4, [3, 3, 512, 512])
conv4_3 = conv(conv4_2, [3, 3, 512, 512])
pool4 = pool(conv4_3)

conv5 = conv(pool4, [3, 3, 512, 512])
conv5_2 = conv(conv5, [3, 3, 512, 512])
conv5_3 = conv(conv5_2, [3, 3, 512, 512])
pool5 = pool(conv5_3)

flat = tf.reshape(pool5, [-1, 2*2*512])

keep_prob = tf.placeholder(tf.float32)
fc1 = fc(flat, [2*2*512, 4096], act=tf.nn.relu)
fc1 = tf.nn.dropout(fc1, keep_prob)
fc2 = fc(fc1, [4096, 4096], act=tf.nn.relu)
fc2 = tf.nn.dropout(fc2, keep_prob)
fc3 = fc(fc2, [4096, 7])
```

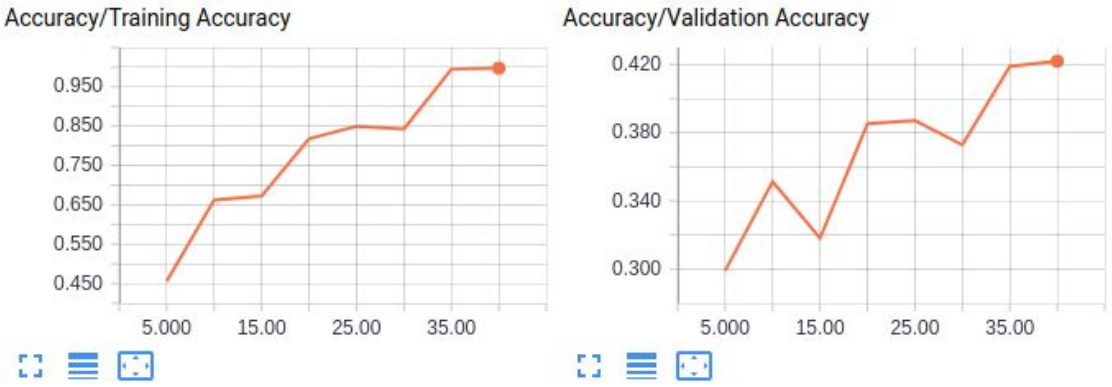


Kaggle上的成績為多個model ensemble，此為其中一個model，kaggle 上的分數為0.686

使用Adam optimizer，learning rate一開始為1e-3，並且手動調整，調整根據為validation accuracy停止上升時，把learning rate*0.1。一開始先不加dropout等到收斂時再把keep_prob調為0.5。data augmentation包含random crop成比較小張的圖片，以及旋轉、左右翻轉。圖形會有起伏是因為data augmentation是逐步加入的，而不是一開始就全部加進去，這樣會train不起來。

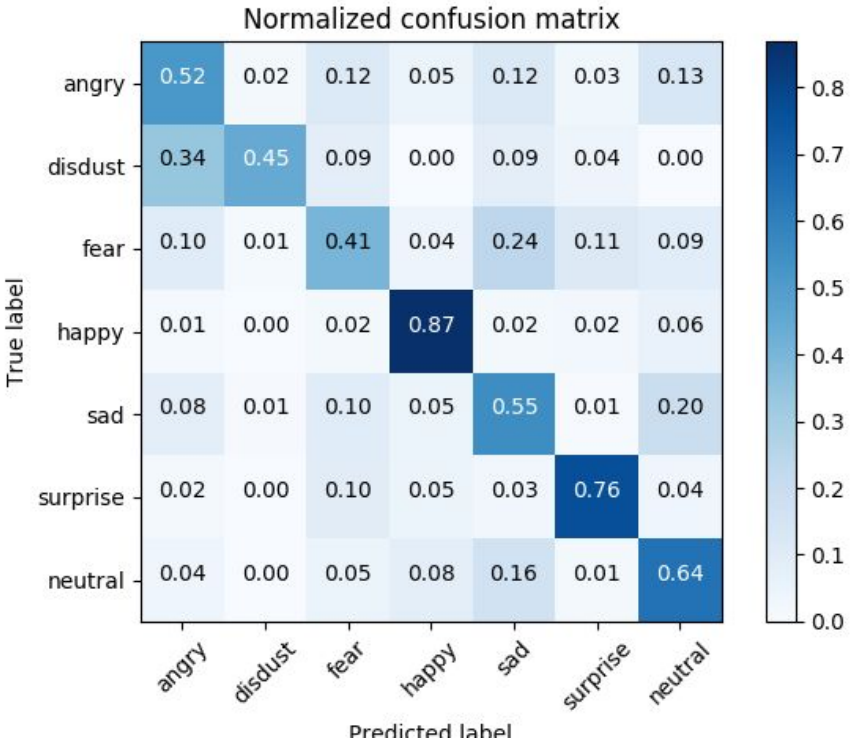
2. (1%) 承上題，請用與上述 CNN 接近的參數量，實做簡單的 DNN model。其模型架構、訓練過程和準確率為何？試與上題結果做比較，並說明你觀察到了什麼？

```
fc1 = fc(image, [48*48, 4096], act=tf.nn.relu)
fc2 = fc(fc1, [4096, 4096], act=tf.nn.relu)
fc3 = fc(fc2, [4096, 4096], act=tf.nn.relu)
fc4 = fc(fc3, [4096, 7], act=tf.nn.relu)
```



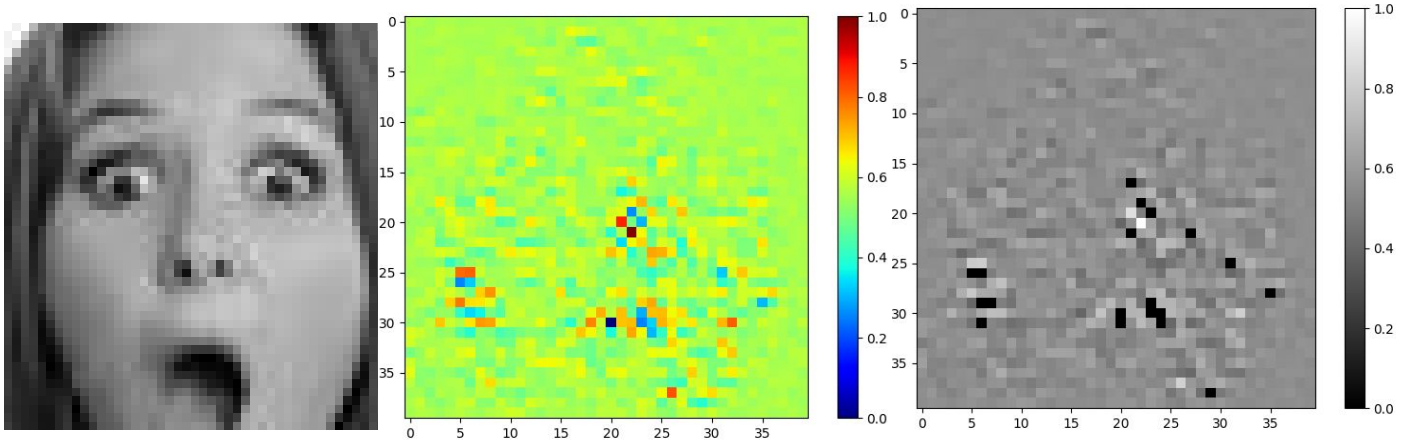
從圖上可以看到雖然在training set表現很好(但應該是因為沒有做data augmentation)，validation set卻差CNN很多。訓練過程使用adam optimizer，learning rate固定為1e-5。Kaggle上的成績為0.402。

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]



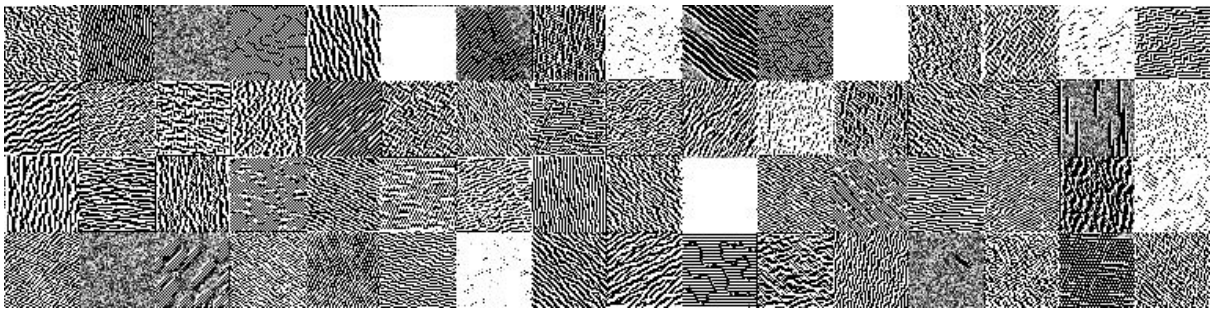
取training dat中25000張以後的圖片作為validation data。可以觀察到disgust容易被認為是angry，因此推測CNN可能認為這2類別是相似的。

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？



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。

會讓第一層CNN filter最被activate的圖片：



取training data中一張圖片經過第一層CNN的結果：