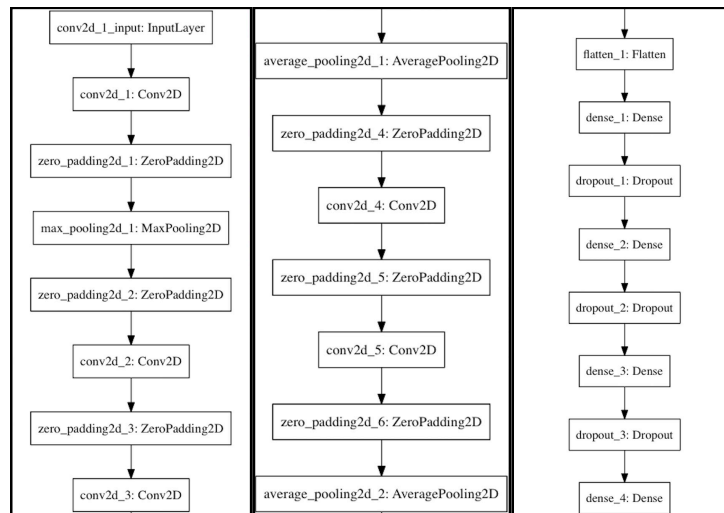


學號：R06942128 系級：電信碩一 姓名：許祐銘

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？



Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 44, 44, 32)	832
zero_padding2d_1 (ZeroPadding2D)	(None, 46, 46, 32)	0
max_pooling2d_1 (MaxPooling2D)	(None, 21, 21, 32)	0
dropout_1 (Dropout)	(None, 21, 21, 32)	0
conv2d_2 (Conv2D)	(None, 19, 19, 64)	18496
zero_padding2d_2 (ZeroPadding2D)	(None, 21, 21, 64)	0
dropout_2 (Dropout)	(None, 21, 21, 64)	0
conv2d_3 (Conv2D)	(None, 19, 19, 128)	73856
average_pooling2d_1 (AveragePooling2D)	(None, 9, 9, 128)	0
zero_padding2d_3 (ZeroPadding2D)	(None, 11, 11, 128)	0
dropout_3 (Dropout)	(None, 11, 11, 128)	0
conv2d_4 (Conv2D)	(None, 9, 9, 256)	295168
zero_padding2d_4 (ZeroPadding2D)	(None, 11, 11, 256)	0
average_pooling2d_2 (AveragePooling2D)	(None, 5, 5, 256)	0
dropout_4 (Dropout)	(None, 5, 5, 256)	0
conv2d_5 (Conv2D)	(None, 3, 3, 512)	1180160
zero_padding2d_5 (ZeroPadding2D)	(None, 5, 5, 512)	0
average_pooling2d_3 (AveragePooling2D)	(None, 2, 2, 512)	0
dropout_5 (Dropout)	(None, 2, 2, 512)	0
flatten_1 (Flatten)	(None, 2048)	0
dense_1 (Dense)	(None, 512)	1049088
dropout_6 (Dropout)	(None, 512)	0
dense_2 (Dense)	(None, 512)	262656
dropout_7 (Dropout)	(None, 512)	0
dense_3 (Dense)	(None, 512)	262656
dropout_8 (Dropout)	(None, 512)	0
dense_4 (Dense)	(None, 7)	3591
Total params: 3,146,503		
Trainable params: 3,146,503		
Non-trainable params: 0		

Batch Size = 256, epochs = 200, Acc on train data = 52.84 %, Acc on valid data = 53.16%

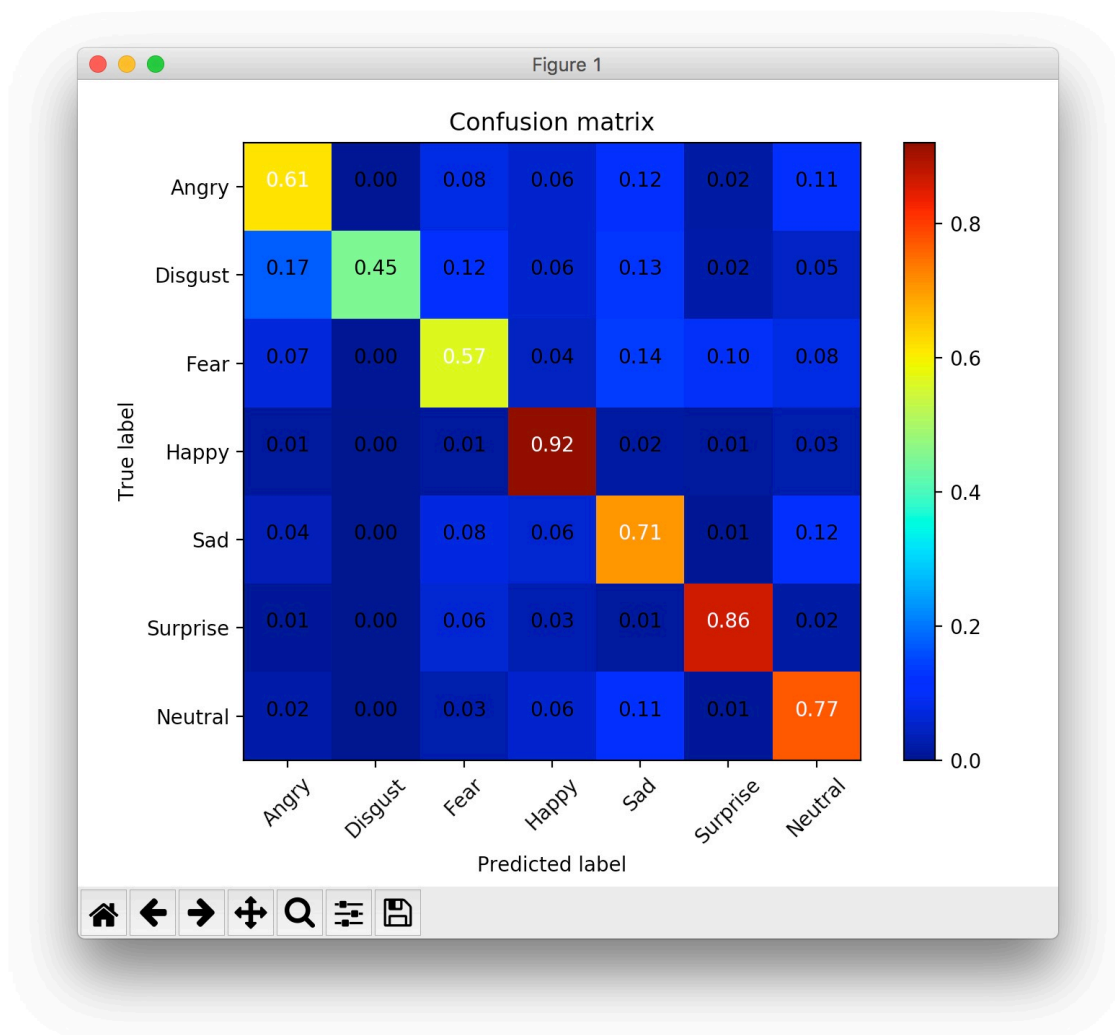
訓練過程：一直到 epoch 63 時，train & valid 的 accuracy 一直卡在 25%左右，而後

就一直上升，但是跑完 200 epoch 時似乎還沒收斂，但是因為我的mac實在 train不

動，HTC deepQ 平台只要我參數太多，就會有 ran out of memory的情況，太晚開始用

deepQ平台，不得已只好先跑這樣...

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？
3. (1%) 觀察答錯的圖片中，哪些 class 彼此間容易用混？

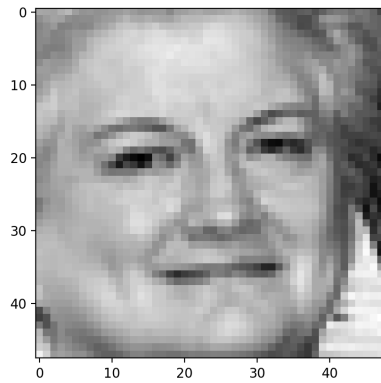


看得出來，正確率最低的是 Disgust，很容易辨識成 Angry、Fear 和 Sad。

而正確率最高的則是 Happy 和 Surprise，我猜應該是這兩個 class 的臉部表情較為明

顯，所以比較不容易辨識錯誤。

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。