

學號：R06945003 系級：生醫電資碩一 姓名：林鈺盛

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：NO)

CNN model 架構：

Layer (type)	Output Shape	Param #			
conv2d_1 (Conv2D)	(None, 32, 48, 48)	320	activation_6 (Activation)	(None, 64, 24, 48)	0
batch_normalization_1 (Batch Normalization)	(None, 32, 48, 48)	128	max_pooling2d_2 (MaxPooling2D)	(None, 32, 12, 48)	0
activation_1 (Activation)	(None, 32, 48, 48)	0	conv2d_7 (Conv2D)	(None, 128, 12, 48)	36992
conv2d_2 (Conv2D)	(None, 32, 48, 48)	9248	batch_normalization_7 (Batch Normalization)	(None, 128, 12, 48)	512
batch_normalization_2 (Batch Normalization)	(None, 32, 48, 48)	128	activation_7 (Activation)	(None, 128, 12, 48)	0
activation_2 (Activation)	(None, 32, 48, 48)	0	conv2d_8 (Conv2D)	(None, 128, 12, 48)	147584
conv2d_3 (Conv2D)	(None, 32, 48, 48)	9248	batch_normalization_8 (Batch Normalization)	(None, 128, 12, 48)	512
batch_normalization_3 (Batch Normalization)	(None, 32, 48, 48)	128	activation_8 (Activation)	(None, 128, 12, 48)	0
activation_3 (Activation)	(None, 32, 48, 48)	0	conv2d_9 (Conv2D)	(None, 128, 12, 48)	147584
max_pooling2d_1 (MaxPooling2D)	(None, 16, 24, 48)	0	batch_normalization_9 (Batch Normalization)	(None, 128, 12, 48)	512
conv2d_4 (Conv2D)	(None, 64, 24, 48)	9280	activation_9 (Activation)	(None, 128, 12, 48)	0
batch_normalization_4 (Batch Normalization)	(None, 64, 24, 48)	256	max_pooling2d_3 (MaxPooling2D)	(None, 64, 6, 48)	0
activation_4 (Activation)	(None, 64, 24, 48)	0	flatten_1 (Flatten)	(None, 18432)	0
conv2d_5 (Conv2D)	(None, 64, 24, 48)	36928	dense_1 (Dense)	(None, 1024)	18875392
batch_normalization_5 (Batch Normalization)	(None, 64, 24, 48)	256	batch_normalization_10 (Batch Normalization)	(None, 1024)	4096
activation_5 (Activation)	(None, 64, 24, 48)	0	activation_10 (Activation)	(None, 1024)	0
conv2d_6 (Conv2D)	(None, 64, 24, 48)	36928	dropout_1 (Dropout)	(None, 1024)	0
batch_normalization_6 (Batch Normalization)	(None, 64, 24, 48)	256	dense_2 (Dense)	(None, 7)	7175
			Total params: 19,323,463		
			Trainable params: 19,320,071		
			Non-trainable params: 3,392		

訓練過程：

Epochs:100

Batch size:128

Optimizer: Adam

Augmentation:

```
train_datagen = ImageDataGenerator(  
    featurewise_center=True,  
    featurewise_std_normalization=True,  
    rotation_range=10,  
    width_shift_range=0.1,  
    height_shift_range=0.1,  
    horizontal_flip=True,  
    data_format='channels_first')
```

我的 CNN model 是用全部的 training data 來訓練，沒有切 validation，因為當時覺得 val\_acc 都沒有好的 performance，test data 在 kaggle 上得到的準確率 0.65。

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 : NO)

DNN model 架構：

Layer (type)	Output Shape	Param #
=====	=====	=====
dense_1 (Dense)	(None, 2048)	4720640
dense_2 (Dense)	(None, 2048)	4196352
dense_3 (Dense)	(None, 2048)	4196352
dense_4 (Dense)	(None, 1024)	2098176
dropout_1 (Dropout)	(None, 1024)	0
batch_normalization_1 (Batch Normalization)	(None, 1024)	4096
dense_5 (Dense)	(None, 1024)	1049600
dense_6 (Dense)	(None, 1024)	1049600
batch_normalization_2 (Batch Normalization)	(None, 1024)	4096
dense_7 (Dense)	(None, 512)	524800
dense_8 (Dense)	(None, 512)	262656
batch_normalization_3 (Batch Normalization)	(None, 512)	2048
dense_9 (Dense)	(None, 256)	131328
dense_10 (Dense)	(None, 128)	32896
dropout_2 (Dropout)	(None, 128)	0
dense_11 (Dense)	(None, 7)	903
=====	=====	=====
Total params: 18,273,543		
Trainable params: 18,268,423		
Non-trainable params: 5,120		
=====		

訓練過程：

Epochs : 80

Batch size : 128

Validation set : 20% training data

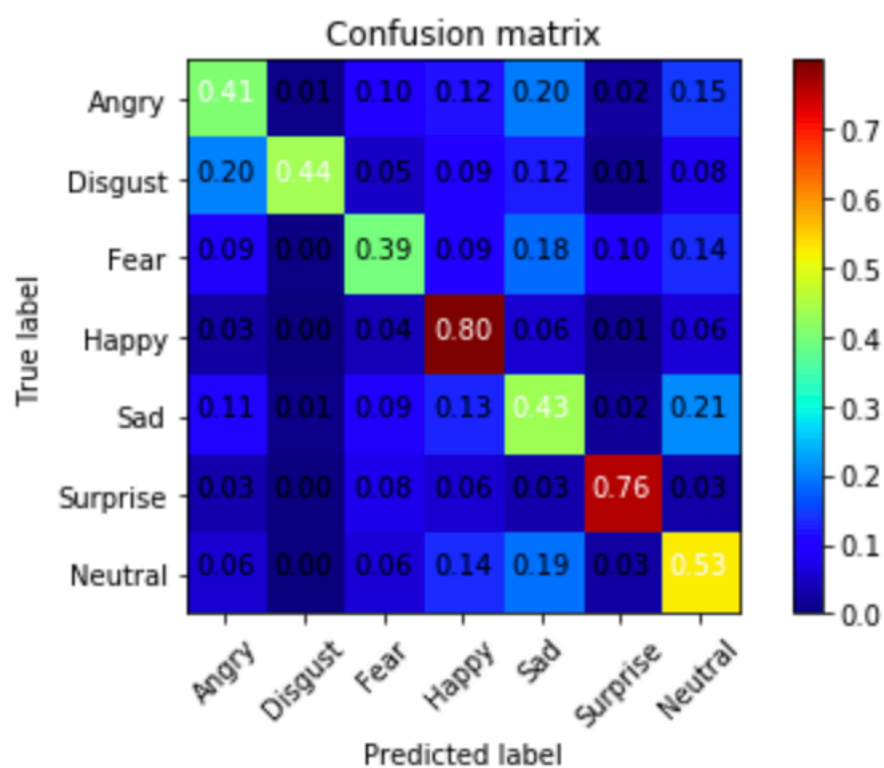
Optimizer : Adam

Accuracy : val\_acc : 0.33 kaggle : 0.32

很明顯可以觀察到用一般的 DNN 對於影像辨識的準確率非常差，相較於 CNN

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]

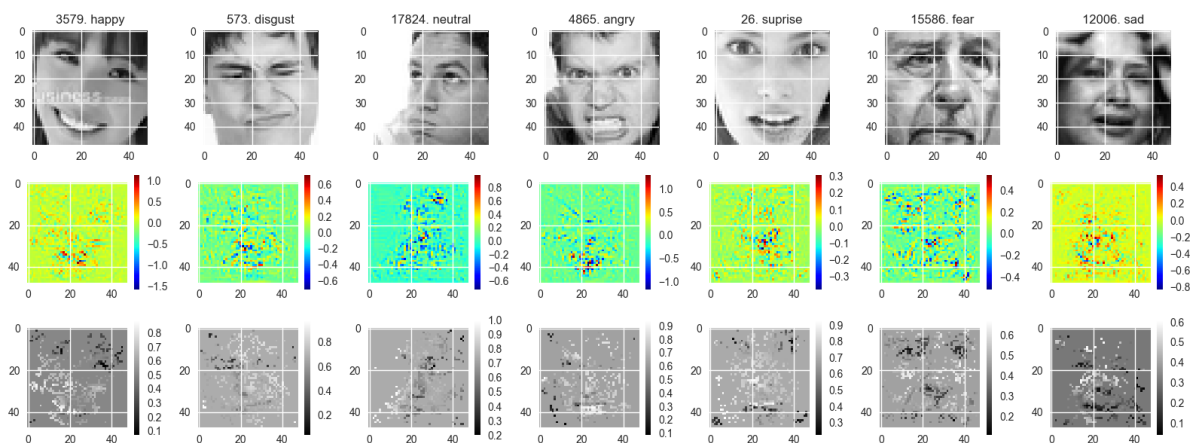
(Collaborators : NO)



Happy 以及 Surprise 的準確率最高，其餘容易搞混的有 Angry-Sad、Disgust-Angry、Fear-Sad、Sad-Neutral。(ps.因為我較好的 model 並沒有切 validation，所以用了一個約 55% accuracy 的 model 來 predict validation(20% training data) )

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 : NO)

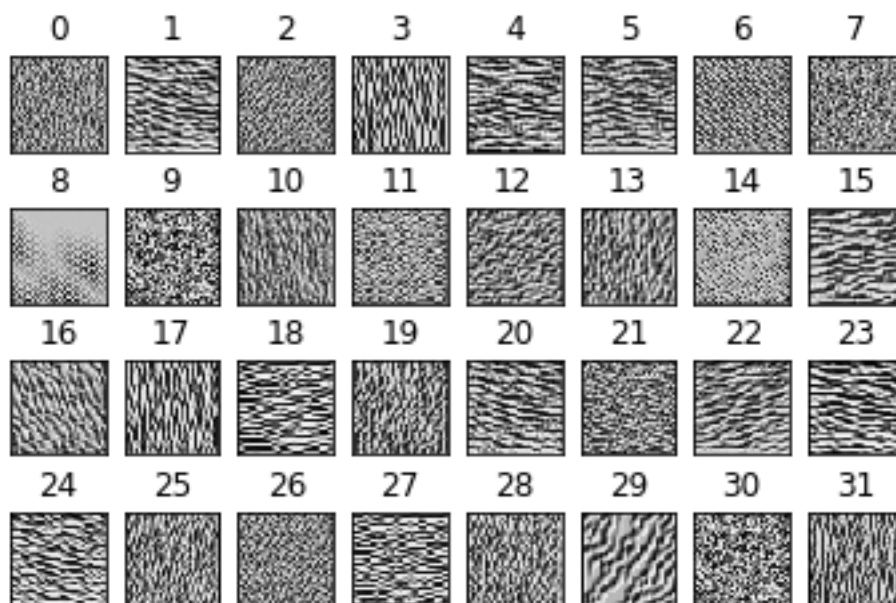


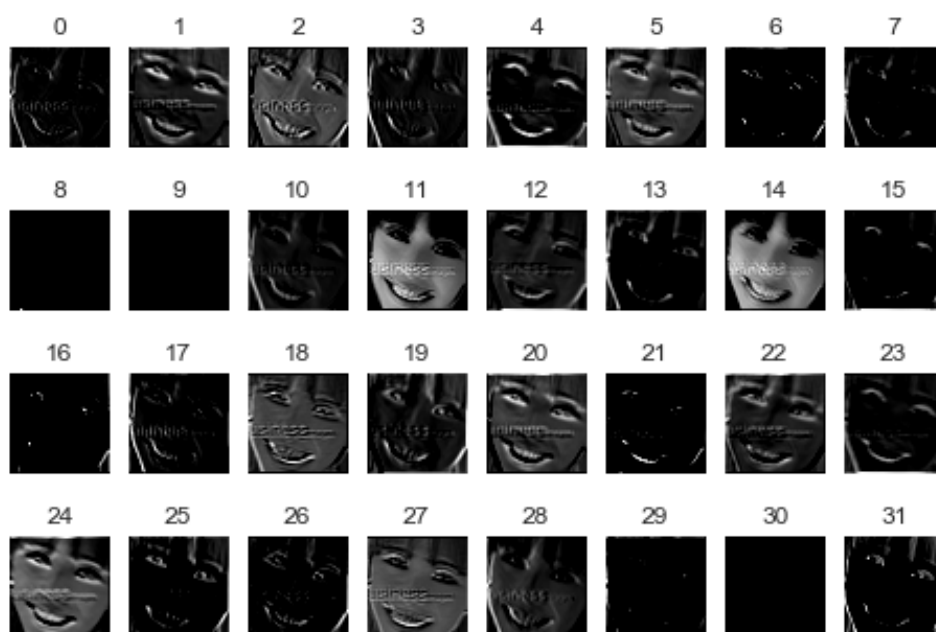
每種 class 皆挑選一張圖片來分析，可以發現 model 主要 focus 在五官的輪廓，尤其是嘴巴延伸至臉頰的部分

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 : NO)

First layer : conv2D\_1(32 filters with kernel size = 3)





選擇的原圖為上一題的 happy