學號：R08921A01　系級：電機碩一　姓名：陳允中

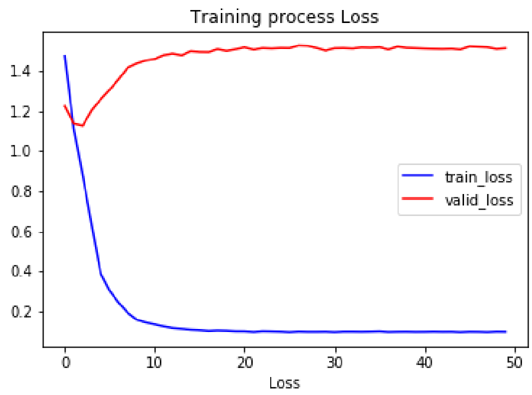
1. (1%) 請說明這次使用的 model 架構，包含各層維度及連接方式。

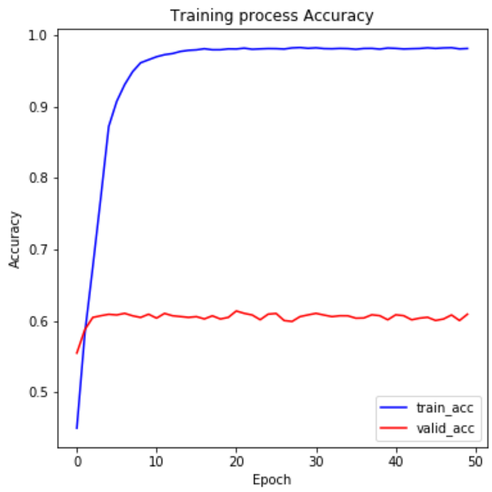
使用 Resnet18 Pretrained Model，把最後一層 output 改為 7 並在前面加 dropout

Conv2d(3, 64, kernel_size=(7, 7), stride=(2, 2), padding=(3, 3), bias=False)
BatchNorm2d(64, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
ReLU(inplace=True)
MaxPool2d(kernel_size=3, stride=2, padding=1, dilation=1, ceil_mode=False)
Conv2d(64, 64, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1), bias=False)
BatchNorm2d(64, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
ReLU(inplace=True)
Conv2d(64, 64, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1), bias=False)
BatchNorm2d(64, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
Conv2d(64, 64, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1), bias=False)
BatchNorm2d(64, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
ReLU(inplace=True)
Conv2d(64, 64, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1), bias=False)
BatchNorm2d(64, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
Conv2d(64, 128, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1), bias=False)
BatchNorm2d(128, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
ReLU(inplace=True)
Conv2d(128, 128, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1), bias=False)
BatchNorm2d(128, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
Conv2d(64, 128, kernel_size=(1, 1), stride=(2, 2), bias=False)
BatchNorm2d(128, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
Conv2d(128, 128, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1), bias=False)
BatchNorm2d(128, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
ReLU(inplace=True)
Conv2d(128, 128, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1), bias=False)
BatchNorm2d(128, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
Conv2d(128, 256, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1), bias=False)
BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
ReLU(inplace=True)
Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1), bias=False)
BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
Conv2d(128, 256, kernel_size=(1, 1), stride=(2, 2), bias=False)
BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1), bias=False)
BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
ReLU(inplace=True)

Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1), bias=False)
BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
Conv2d(256, 512, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1), bias=False)
BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
ReLU(inplace=True)
Conv2d(512, 512, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1), bias=False)
BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
Conv2d(256, 512, kernel_size=(1, 1), stride=(2, 2), bias=False)
BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
Conv2d(512, 512, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1), bias=False)
BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
ReLU(inplace=True)
Conv2d(512, 512, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1), bias=False)
BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
AdaptiveAvgPool2d(output_size=(1, 1))
Dropout(p=0.5)
Linear(in_features=512, out_features=7, bias=True)

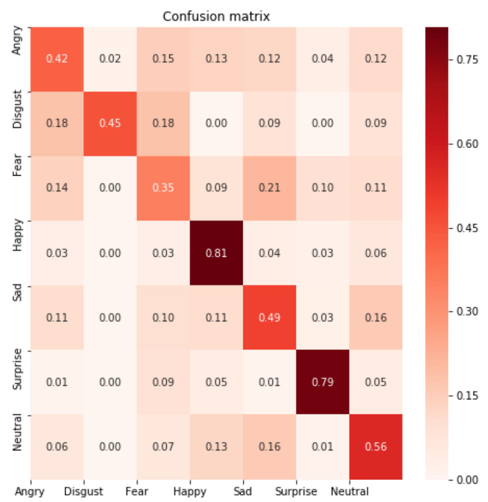
2. (1%) 請附上 model 的 training/validation history (loss and accuracy)。



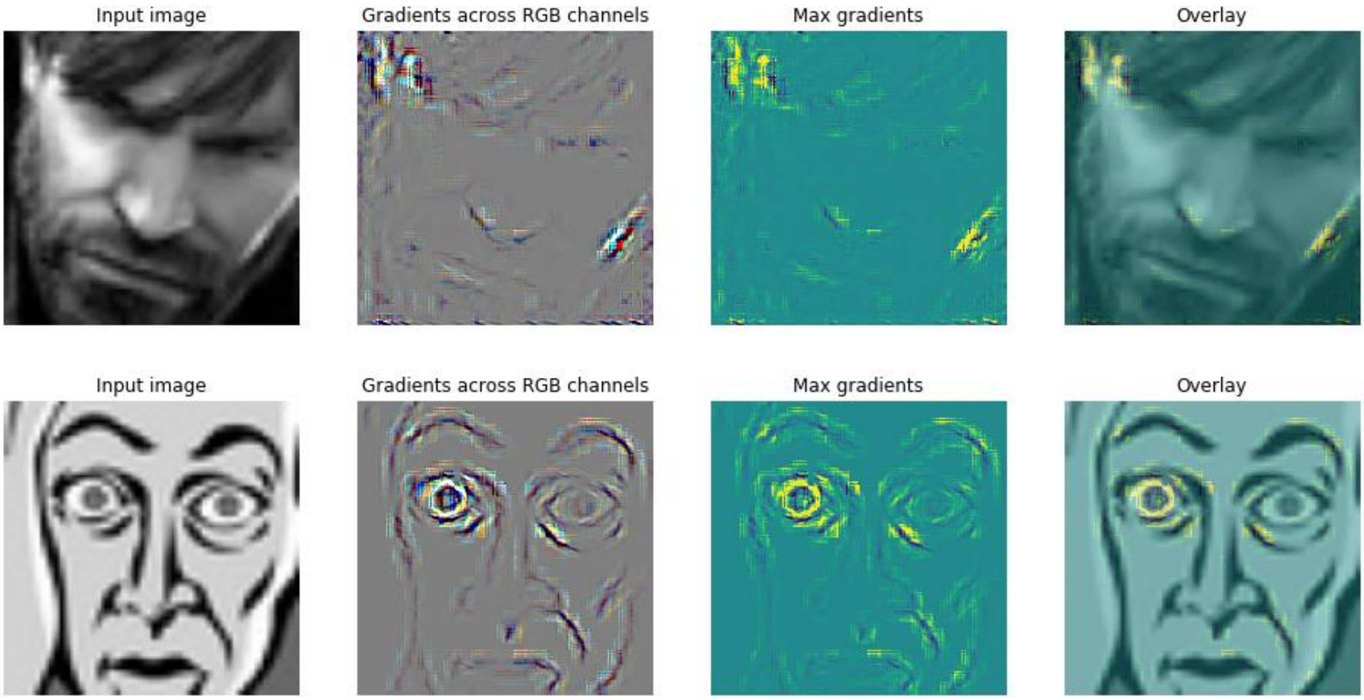


3. (1%) 畫出 confusion matrix 分析哪些類別的圖片容易使 model 搞混，並簡單說明。

Happy 和 Surprise 的特徵最明顯最容易辨認。
最容易被誤認的表情為 Fear，且最容易被誤認為 Sad。



4. (1%) 畫出 CNN model 的 saliency map，並簡單討論其現象。



上圖的照片產生 gradient 的地方是因為照片的光影造成的
下圖不是照片所以有比較明顯的邊界會產生較大的 gradient

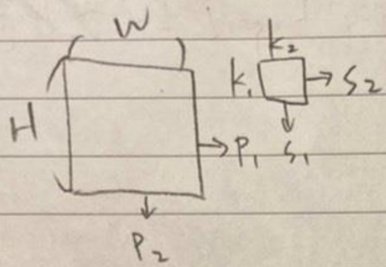
5. (1%) 畫出最後一層的 filters 最容易被哪些 feature activate。
(ref: https://reurl.cc/ZnrgYg)

6. (3%)Refer to math problem

Date

學號: R08921A01 系級: 電機碩一 姓名: 陳允中

Convolution :



$$W' = \frac{W + 2P_1 - k_2}{S_2} + 1$$

$$H' = \frac{H + 2P_2 - k_1}{S_1} + 1$$

$$A : \left(B, \frac{W+2P_1 - k_2}{S_2} + 1, \frac{H + 2P_2 - k_1}{S_1} + 1, \text{output channels} \right)$$

Batch Normalization :

$I : B = \{x_{1 \dots m}\}$

$O : y_i = BN_{\gamma,\beta}(x_i)$

$$\mu_B = \frac{1}{m} \sum x_i \qquad \hat{x}_i = \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \varepsilon}}$$

$$\sigma_B^2 = \frac{1}{m} \sum (x_i - \mu_B)^2$$

$$y_i = \gamma \hat{x}_i + \beta \equiv BN_{\gamma,\beta}(x_i)$$

$$\frac{\partial l}{\partial \hat{x}_i} = \frac{\partial l}{\partial y_i} \cdot \gamma$$

$$\frac{\partial l}{\partial \sigma_B^2} = \sum_{i=1}^{m} \frac{\partial l}{\partial \hat{x}_i} (x_i - \mu_B) \cdot \frac{-1}{2} (\sigma_B^2 + \varepsilon)^{-\frac{3}{2}}$$

$$\frac{\partial l}{\partial \mu_B} = \left(\sum_{i=1}^{m} \frac{\partial l}{\partial \hat{x}_i} \cdot \frac{-1}{\sqrt{\sigma_B^2 + \varepsilon}} \right) + \frac{\partial l}{\partial \sigma_B^2} \cdot \frac{-2}{m} \cdot \sum_{i=1}^{m} (x_i - \mu_B)$$

$$\frac{\partial l}{\partial x_i} = \frac{\partial l}{\partial \hat{x}_i} \cdot \frac{1}{\sqrt{\sigma_B^2 + \varepsilon}} + \frac{\partial l}{\partial \sigma_B^2} \cdot \frac{2}{m} (x_i - \mu_B) + \frac{\partial l}{\partial \mu_B} \cdot \frac{1}{m}$$

$$\frac{\partial l}{\partial \gamma} = \sum_{i=1}^{m} \frac{\partial l}{\partial y_i} \cdot \hat{x}_i$$

$$\frac{\partial l}{\partial \beta} = \sum_{i=1}^{m} \frac{\partial l}{\partial y_i}$$

Softmax and Cross Entropy:

$$\text{softmax}(z_t) = \frac{e^{z_t}}{\sum_i e^{z_i}}$$

$$\text{cross\_entropy} = L(y, \hat{y}) = -\sum_i y_i \log \hat{y}_i$$

$$\text{cross\_entropy} - L_t(y_t \hat{y}_t) = -y_t \log \hat{y}_t - (1-y_t)\log(1-\hat{y}_t)$$

$$\hat{y}_t = \text{softmax}(z_t)$$

$$\frac{\partial L_t}{\partial z_t} = \frac{\partial L_t}{\partial \hat{y}_t} \cdot \frac{\partial \hat{y}_t}{\partial z_t}$$

$$= \left(-\frac{y_t}{\hat{y}_t} + \frac{1-y_t}{1-\hat{y}_t}\right) \cdot \left(\frac{e^{z_t}}{\sum_i e^{z_i}} + e^{z_t} \cdot \frac{-1}{(\sum_i e^{z_i})^2} \cdot e^{z_t}\right)$$

$$= \left(\frac{-y_t}{\hat{y}_t} + \frac{1-y_t}{1-\hat{y}_t}\right) \cdot \hat{y}_t(1-\hat{y}_t)$$

$$= -y_t + y_t\hat{y}_t + \hat{y}_t - y_t y_t$$

$$= \hat{y}_t - y_t$$