
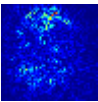


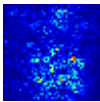


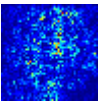


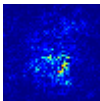
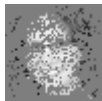

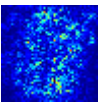


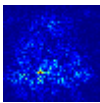


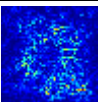

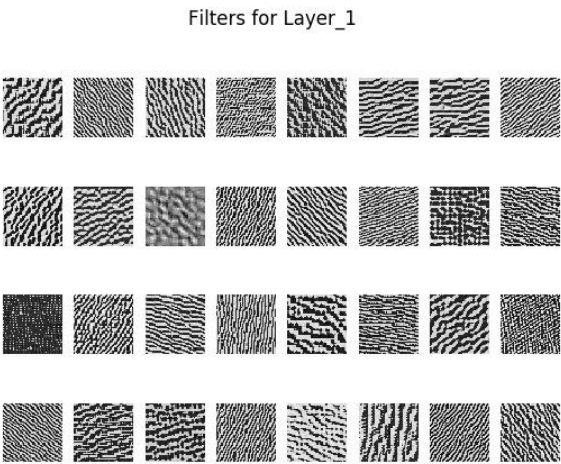


1. (2%) 從作業三可以發現，使用 CNN 的確有些好處，試繪出其 saliency maps，觀察模型在做 classification 時，是 focus 在圖片的哪些部份？

class	image	saliency map	mask	class	image	saliency map	mask
0				1			
2				3			
4				5			
6							

由上表可觀察到，CNN model 在分類時，能夠確實地將重點放在人臉尤其是五官的位置。

2. (3%) 承(1) 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 與觀察 filter 的 output。

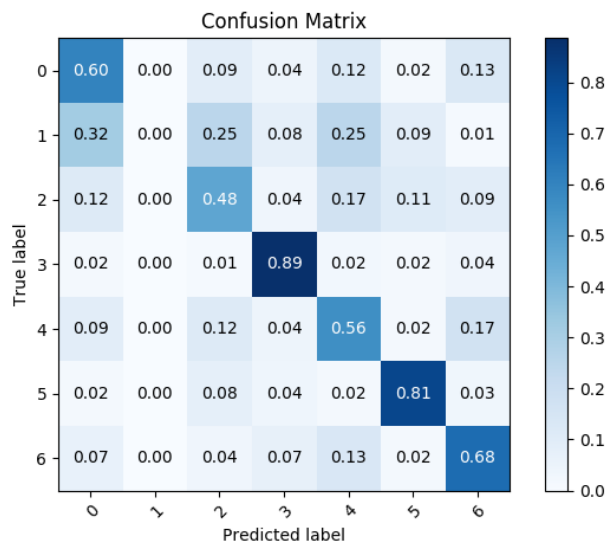


Filters for Layer_1



3. (3%) 請使用 Lime 套件分析你的模型對於各種表情的判斷方式，並解釋為何你的模型在某些 label 表現得特別好 (可以搭配作業三的 Confusion Matrix)。

class	0	1	2	3	4	5	6
image							






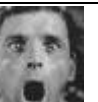

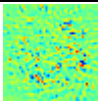
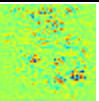
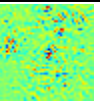
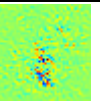
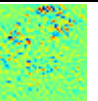
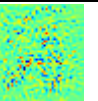
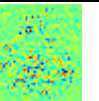


由 lime 的分析結果與 confusion matrix 可以觀察到，在大多數的 class 中，臉部 (尤其是五官) 對於 model 分類都是有幫助的。從 class 1 的 lime 結果來看，model 的確沒有很好地學習到相關的特徵，因此導致 confusion matrix 中 class 1

的分類準確度 0 的奇特現象（本次作業中使用的 class 1 圖片取自 training set，confusion matrix 則是利用 validation set 計算得來的）。此外，從 class 3 及 class 5 的 lime 結果可以發現，這兩個 class 的有利區塊的分布較為正常地貼合人臉（例如 class 5 的結果，張大的嘴巴有利於判斷驚訝的表情），因此表現較佳。

4. (2%) [自由發揮] 請同學自行搜尋或參考上課曾提及的內容，實作任一種方式來觀察 CNN 模型的訓練，並說明你的實作方法及呈現 visualization 的結果。

Smooth Grad 是一種產生 saliency map 的方法：將一張影像複製多次，並對每一張影像都加入隨機的高斯雜訊後求 saliency map，最後再對這些 maps 求平均得到最終的 saliency map。根據 reference，這樣的作法能夠降低 saliency map 的雜訊。實作後的結果如下所示：

class	0	1	2	3	4	5	6
image							
result							

Reference: <https://github.com/utkuozbulak/pytorch-cnn-visualizations>