

1. (1%) 請說明你實作的 **CNN model**，其模型架構、訓練參數和準確率為何？

架構&參數：

4 層 convolution layer 連接 2 層 fully connected layer

第一層級第二層 convolution layer 有 64 個 filter，用 maxpooling 降維

第三及四層比照辦理，但是 filter 數目增加至 256 個

Fully connected layer 的部分每層連接 512 個神經元，以 relu 作為 activation function，最後一層則適用 softmax 吐出結果。

每一層都做 batchnormalization 並設定 Dropout 避免 overfit，dropout rate 隨著層數增加由小慢慢增大，避免 train 不起來的問題出現。

準確率：

在沒有做 augmentation 前準確率可以達到 0.61 左右；加入 augmentation 後則可以做到 0.66 左右。

2. (1%) 請嘗試 **data normalization**, **data augmentation**, 說明實行方法並且說明對準確率有什麼樣的影響？

Normalize method: 全部參數除掉 225，即將資料範圍縮至[0, 1]

實作前準確率：0.24457（根本 train 不起來）

實作後準確率：0.61234

Augmentation: 將資料做放大、旋轉、左右翻面

實作前準確率：0.61234

實作後準確率：0.66184

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]

0	0	1	2	3	4	5	6
0	240	5	22	22	53	8	73
1	8	22	2	1	4	0	2
2	56	1	155	11	85	45	78
3	8	1	7	623	13	14	45
4	48	0	27	22	245	4	120
5	10	1	17	15	4	246	14
6	23	1	12	29	49	7	372

第一行與第一列分別為不同 class 之間的 label，對角線為分類正確的資料量。

從圖中可以看出 label 4 與 label 6 之間比較容易搞混，有 120 筆在 4 中的資料被預測成 6；label 0 與 label 1 以及 label 2 與 label 6 之間也有 73 與 78 筆資料被預測

成 6。再來就是 label 2 與 label 4 之間還有 85 筆資料被預測成 4。

除了上面這幾種，並沒有出現太別誇張的狀況。

從圖中還可發現 class 1 的錯誤較少，但是將樣本相加之後發現其實他本身就沒有很多 testing data，錯誤自然會較少，若以比例來看的話，其實錯誤率並沒有與其他 class 差很多。

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？



input



saliency map



input



saliency map



input



saliency map



input



saliency map



input

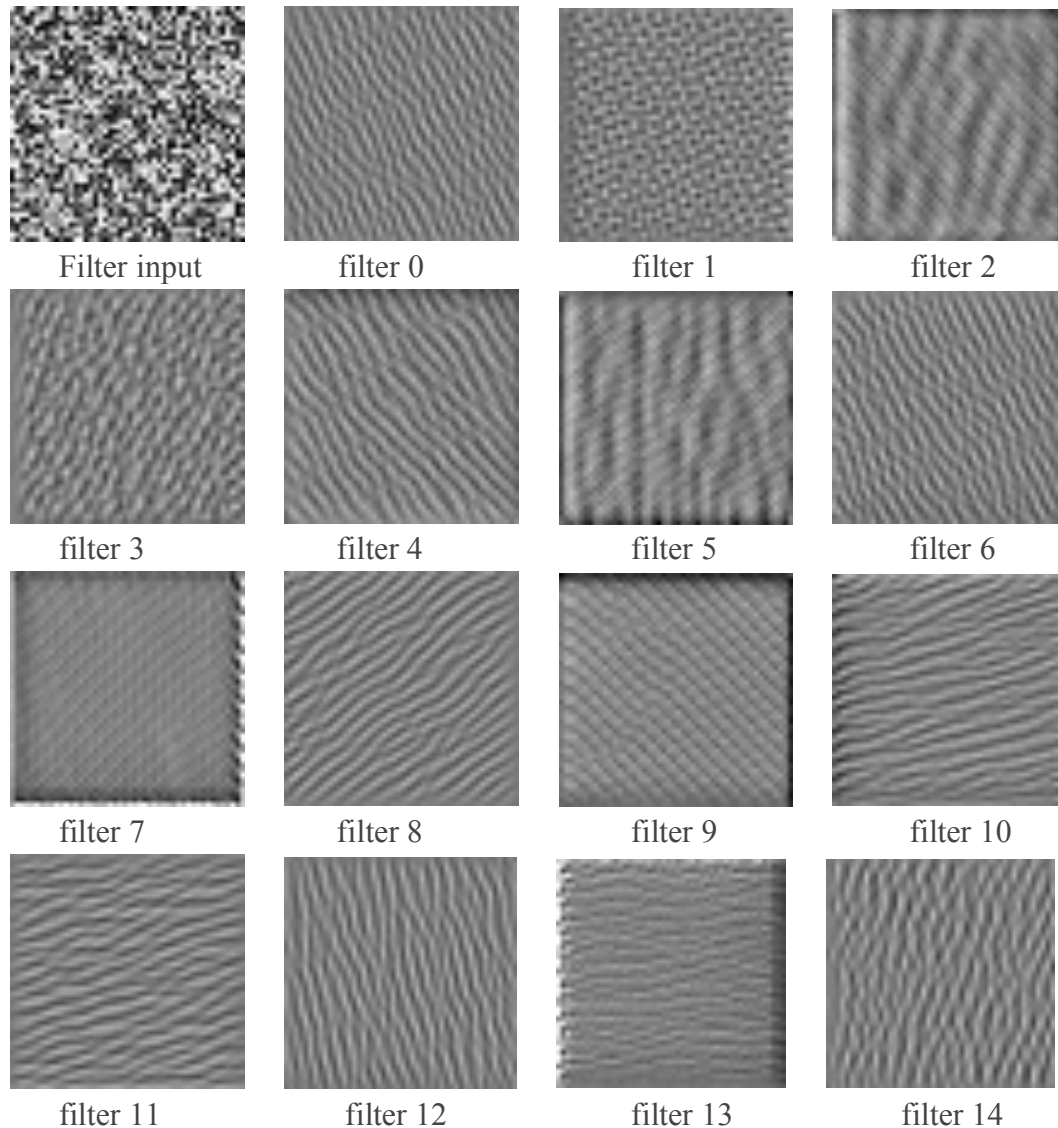


saliency map

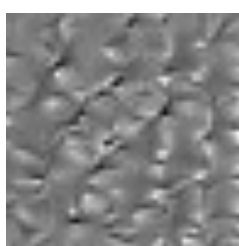
從上面的 saliency map 可以發現其實根本看不出當 CNN 在做分類的時後是著重在什麼部分，或許是因為這樣，所以分類的效果並不是很好，只有大概 7 成左右。

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**。

要找出最容易被 **activate** 的圖片首先要先給整個 **model** 一個 **input** 圖片，我的做法是隨機產生一個灰色圖片在隨機加上一些 **noise**；選擇觀察的 **filter** 為第二層及第四層 **convolutional layer** 的部分 **filter**(64 個)，最後如下



這 14 張圖片是第二層的 **convolution layer** 做出來的結果，可以看出被 **activate** 的圖片其實沒有很清晰，但是大致有個紋路存在，接下來貼出第四層的 **filter**



filter 1



filter 2



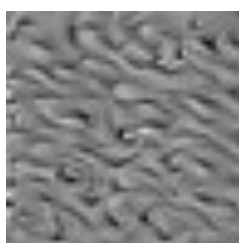
filter 3



filter 4



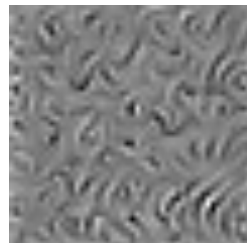
filter 5



filter 6



filter 7



filter 8



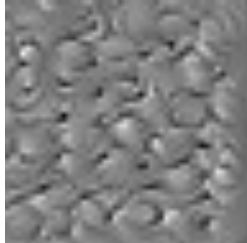
filter 9



filter 10



filter 11



filter 12

從第四層的這幾個 filter 可以看出比起第二層的 filter 他們所要找的 pattern 更明確，有就是說我的 model 會先在第一層濾出一個大略的圖片，接這在分工給下一層做更精細的分類，可見我的 CNN 是有架設成功的。