

# ADLxMLDS hw4 Report

## B03902059 紀典佑

### 1. Model Description:

這次的 GAN 實作採用 ACGAN 的方式，髮色跟眼睛顏色的 data 是採用 one-hot 的方式丟進 model 中，圖片則 normalize 到  $[-1, 1]$ ；D 除了原本判斷圖片真偽外，還會有兩個 classifier 分別來判斷頭髮顏色和眼睛的顏色，而 G 的輸入是先將兩個 one-hot vector concatenate 在一起，之後再 embed 成 100 維另外跟一個 100 維的 noise 做 elementwise multiply，架構圖如圖 a 以及圖 b。

而 objective function 是將三種 loss 直接相加，即  $L = L_{GAN} + L_{hair} + L_{eyes}$ ，不管是否是 G 生成的圖片或者是 data 中的圖片，都有算 classifier loss。

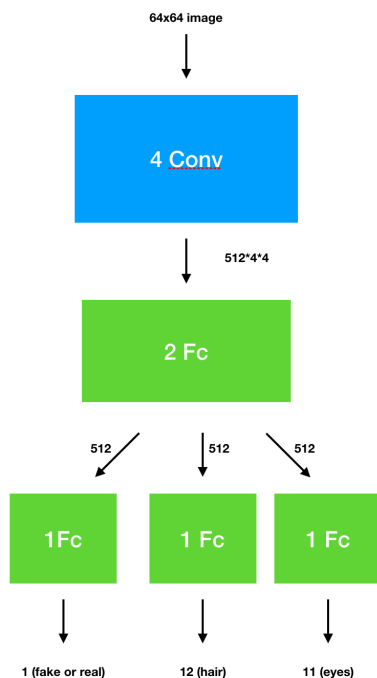


圖 a. Discriminator 架構

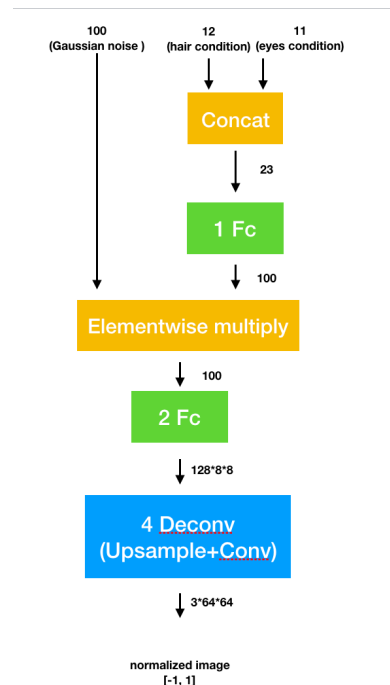


圖 b. Generator 架構

### 2. Improve performance:

一開始 training 的過程會 train 到一定的 epoch 後 D 的強度過於強大，讓 G 學習的很差，因此朝這個方向試了幾種 improvement：

- Label smooth：** Goodfellow 提出 DNN 很容易對於較具信心的 output 給予較高的機率，而造成 net 學習的錯誤，尤其是在 gan 這種 adversarial 的架構，因此 Salimans 提出了將正確的 label 從 1 改成 0.9 的 smooth 方法，但在這次的作業中做了這項修改並未改變一開始學習很差的結果。
- Add noise to input：** 這項 trick 是原本 ML 就有的技巧，加入一些 noise

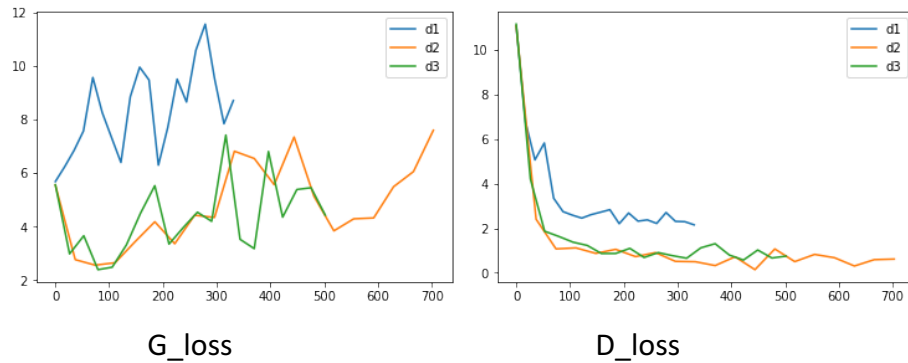
可以讓學習比較不容易 overfitting，而 D 太強其實也是一種 overfitting 的結果，但在這次作業中加了這項結果還是沒什麼改變。

c. Tune learning rate：調整了 D,G 兩者的 Lr，將 D 的 Lr 調小一點，看能不能 balance 兩者，但結果還是不如預期，只有比原本好一些。

d. Model Structure redesign：在 NIPS 2016 GAN Tutorial 提到 D 通常比 G 深且每層通常有較多的 filter 數量，因次重新調整了 model 架構，把 D 弄得更深更多 filter 數，結果大大改善了原本的效果，可見 model 得影響不容小覷。

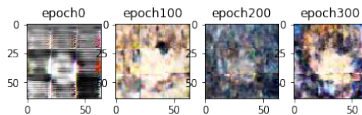
### 3. Experiment：

實驗在不同的 D model 架構上，d1, d2, d3 的不同在於 Conv 深度  $d2=d3>d1$ ，而 d2,d3 的不同則是 filter 量以及 stride 的參數不同，D\_loss 以及 G\_loss 隨著 epoch 的變化如圖：



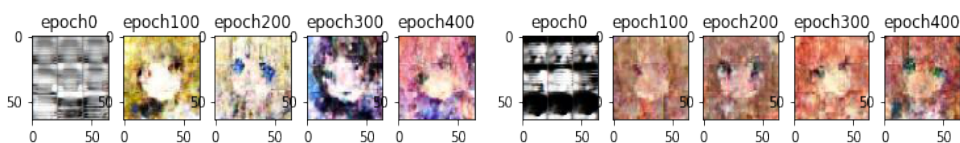
當 D 的 loss 降到一定數值後，G\_loss 會開始不太能降下去，此時差不多就到了這個 GAN 的極限，而依照圖片可以看出，比較深的 model 不管是在學習 D 或者是 G 都能達到比較低的 loss，但 filter 量以及 stride 數量並沒有像深度一樣造成那麼多的影響，以下三張圖是 training 過程中生出的圖片，給的 condition 是橘髮灰眼：

d1



d2

d3



可以看出真正有學習到 **condition** 的是 **d3**，但他的結果其實比 **d2** 模糊，而 **d1** 只學習到一些人臉的特徵，實驗數據證實了 **improvement** 時的發現，**model** 的架構對整個 **training** 的影響極大，就算只是換了 **conv** 層的 **stride** 數量也會有影響。

#### 4. 總結：

這次作業用 **ACGAN** 出來的圖片大部分都能展現出臉的部份，但是對於條件的判定就沒那麼準確，想到未來有可能的解決方法有：

- a. 調整 **condition** 在 **loss** 上面的 **weight**。
- b. 有 **attribute** 的資料做 **image augmentation** 增加 **data**。
- c. 沒有 **attribute** 的資料可以用 **ACGAN** 中的 **D** 的 **classifier**，判斷出髮色跟眼色後，變成新的資料。