

## Model description (2%)

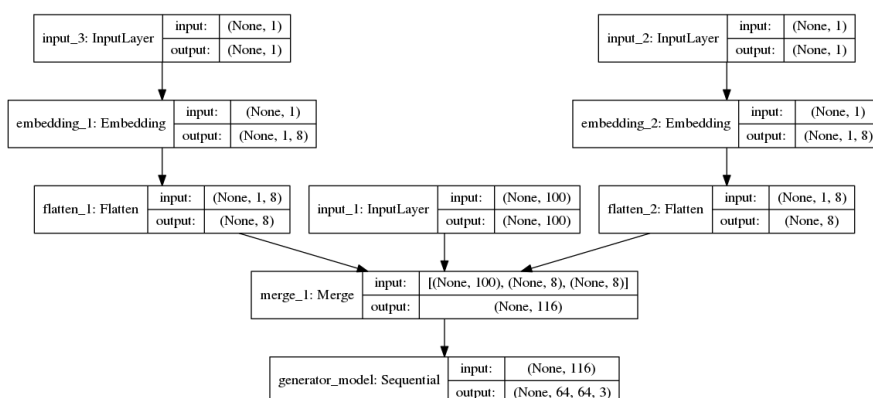
資料前處理：

先將所有沒有髮色或眼睛顏色 tag 的資料捨棄，且若有多個髮色、眼睛顏色的圖片也捨棄掉。最後直接將 12 種髮色和 11 種眼睛顏色，在 generator input 時當作 label 丟進 embedding layer。而在 discriminator 計算 loss 時轉成 one hot vector 計算。

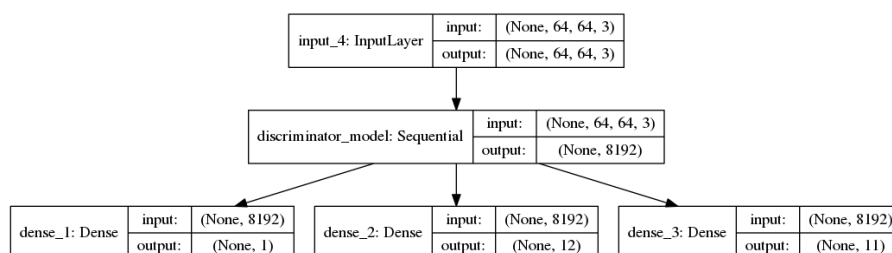
最後使用的 model 為 **ACGAN**，以用來同時產生仿真的圖片和正確的髮色和眼睛顏色 class。Generator 會 input noise 和對應的 class label。而 Discriminator 的三個 output: 第一個為和真的圖片的相似度，二三為預測的 class。

模型架構如下圖：

Generator



Discriminator



## 訓練方式

下列為每個訓練週期 discriminator 和 generator 的訓練方式

### Discriminator:

先 Sample 出 (batch\_size / 2) 大小的 real image 和 label。再用 generator 隨機 predict 出 (batch\_size / 2) 個 fake image 和 label。並對上述的 image 和 label 分別訓練一次。

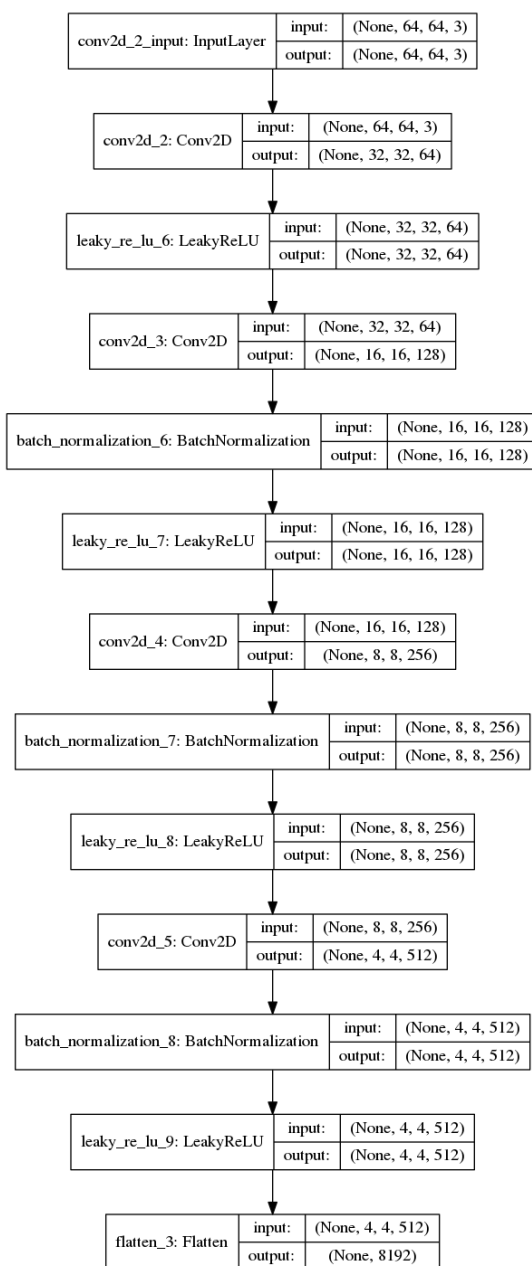
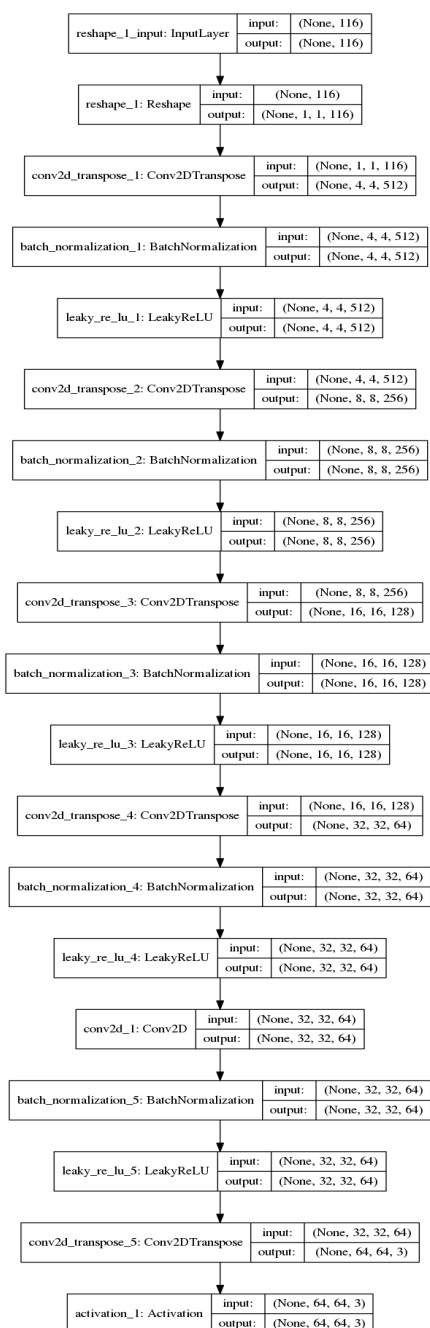
```
dis_metrics_real = discriminator.train_on_batch(real data)
dis_metrics_fake = discriminator.train_on_batch(generated data)
```

## Generator:

將圖片的 input label (hairs, eyes)都丟到 embedding layer 裡成 8 維的 vector 並 concat 起來成 16 維。最後此 feature vector 再與 latent size 為 100 維的 noise concat 起來成 116 維的 vector 丟進 generator model 裡訓練。此時會將 discriminator trainable 設成 false, 並對整個 GAN 模型 sample batch\_size 大小的 data train\_on\_batch 一次。

上述步驟接完成即為完成一個訓練週期，並重複下去。（最後選擇的 batch\_size = 128）

而圖中的 generator\_model, discriminator\_model :Sequential 的內部結構圖如下。  
(左為 generator /右為 discriminator)



## How do you improve your performance (2%)

### 1. 訓練更多的 steps

通常在兩萬左右即可獲得清楚且不崩會的臉部影像了。但是若訓練更多的 steps（在 model 不壞掉的前提下），通常可以獲得更細緻的圖像，頭髮會有更多細節。

	
18000 steps	51000 steps

### 2. 將髮色和眼睛眼色 embedding 後的 vector 從 multiply 改成 concat

最原先的 model 會將 embedding vector 的結果相乘，但是這樣得到結果就是最後 generate 髮色和眼睛時常是錯誤的。改成 concat 後就顯著改善了這個問題。

	
Blue hair aqua eyes (multiply)	Blue hair aqua eyes (concat)

3. 將 generator 和 discriminator 在每一個 time step 中訓練的總 data 大小改成一樣 discriminator 會在真假資料各 train 一次，故我將 discriminator 分別 train 的 batch size 除 2，讓他和 generator 看到的總 image 數一樣。這樣可以避免一個現象也就是最後 generator 全部都 generate 幾乎同樣的圖片（同一髮色）。可能是原先 discriminator 太強導致的現象。

	
沒減半 (red hair different eyes)	有減半(red hair different eyes)

## Experiment settings and observation (2%)

1. 若不做 ACGAN 而是單存的 DCGAN 產生圖片（不管 class），可以很快的就得到相當清楚的圖片，長的也蠻不錯的。


DCGAN 9500 steps

2. 若始終 generate 的圖片樣子不太清楚、模糊或是崩壞，可以在 discriminator 的 loss 那邊加上不同的 weight。給予第一個 loss(真實度) 較高的 weight 而 class(髮色眼睛顏色) 較低的 weight。這樣可以比較迅速的獲得清楚和比較好看的頭像。以將生成圖片相似度的 loss 的 weight 設成 1.4，hair , eyes class 的 loss weight 設成 0.8 的結果。

	
沒有加 loss weight 9500 steps	有加 loss weight[1.4, 0.8, 0.8] 9500 steps

### 3. 不同的 batch size 不太會影響結果

試過 batch\_size = 64, 128 ,256，最後的結果接鄉差不遠。