

# Model

## Generator

Hair color -> Embedding(8) + Gaussian Noise (std = 0.1)

Eye color -> Embedding(8) + Gaussian Noise (std = 0.1)

Concat (Noise, hair, eyes) -> Deconv Block \* 6 (filters = 1024, 512, 256, 128, 64, 3)

Deconv Block:

Conv2DTranspose (kernel = 4, strides = 2, padding= same)

Batch Normalization (momentum=0.5)

LeakyReLU (alpha = 0.2)

Notes:

Deconv 1: strides = 1

Deconv 5: kernel = 3, strides = 1

Deconv 6: no batch normalization

## Discriminator

Conv Block \* 4 (kernel = 64, 128, 256, 512)

Dense(1) -> Real Fake

Dense(12) -> Hair color

Dense(11) -> Eye Color

Conv Block:

Conv2D (kernel = 4, strides=2, padding= same)

Batch Normalization (momentum=0.5)

LeakyReLU (alpha = 0.2)

Notes:

Conv 1: no batch normalization

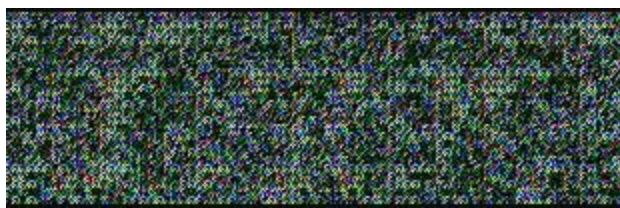
# Experiments and Improvements

## Model

Generator Filter 變多

一開始應該是看錯code, filter 都 < 64

Filter 變多後就可以產生圖片了



VS



DCGAN -> ACGAN

加了 label 進 generator, discriminator 加 auxiliary classifier 就可以控制輸出了



VS



Generator 最後一層不加 batch normalization

Generator 學的速度變快, 減少格子 + 太多白色



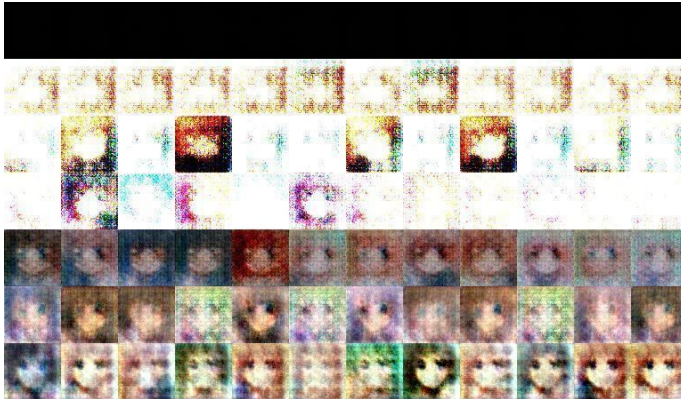
VS



Generator embedding noise std 0.05 -> 0.1

好像不是很明顯

Generator 最後一層 leaky relu -> tanh  
一開始比較不會過曝



Embedding Dimension  
4 -> 8,  
noise input 128 -> 256  
畫出來的圖多樣性比較高



Generator oversize 再 subsample

最後輸出 128x128 再 conv2d 回 64  
不穩定

## Training

Real / Fake 用 soft label

0 -> [0, 0.2]

1 -> [0.8, 1]

聽說有用就加了

Update ratio

Discriminator 用 真的 train 1 個 batch

Discriminator 用 假的 train 1 個 batch

Generator train N 個 batch

N 1- 4 好像都差不多

### Weighted loss

Auxiliary classifier loss real/fake 跟 classification 之間的權重  
real/fake 比較高 比較快會畫圖  
Classification 比較高 畫的圖比較符合label

### Extra training data

只用 label 剛好眼睛斜法各一種的  
-> 所有 image , mask 掉不能用的 loss  
影響不大, 應該是因為這只影響 discriminator, 不影響 generator生的圖