

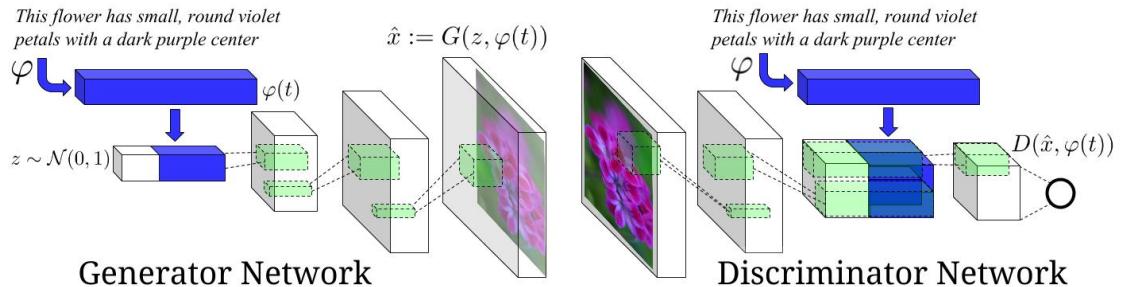
ADLxMLDS HW4 report

Text to Image Generation

姓名：李宇哲 系級：電信所碩一 學號：R06942074

1. Introduction

首先，按照 HW4 投影片的指示，需要 build 一個 Conditional GAN network (如下圖)，然後每一筆 image data 都是 $96 * 96 * 3$ 的維度，以及對應的 Tags。不過在做 conditional GAN 之前需要將 tags 用 skip-thought 的工具轉換成對應的 vector，通常是 4800 維(2400 Uni + 2400 Bi)，最後再將 4800 綴的 vector 利用一層 DNN 將維然後和 noise(我用 100 綴度) 接在一起送進 generator，至於 discriminator 在最後面也會跟將維過的 skip-thought 接在一起。



上圖為原始 Conditional GAN with skip thought

然而，經過初步實驗後發現如果把 tags 用 skip-thought 到 4800 綴的話，很難準確調整 tags 去使 generator 產出要的圖片，因為 HW4 的 conditional 其實很簡單，就是一個髮色配一個眼睛：

[color hair 12]: orange, white, aqua, gray, green, red, purple, pink, blue, black, brown, blonde, [color eyes 11]: gray, black, orange, pink, yellow, aqua, purple, green, brown, red, blue (共 $11 * 12 = 132$ 種組合)

所以我最後決定不使用 skip-thought，直接把 conditional 變成 one hot 的 vector 送進 model。

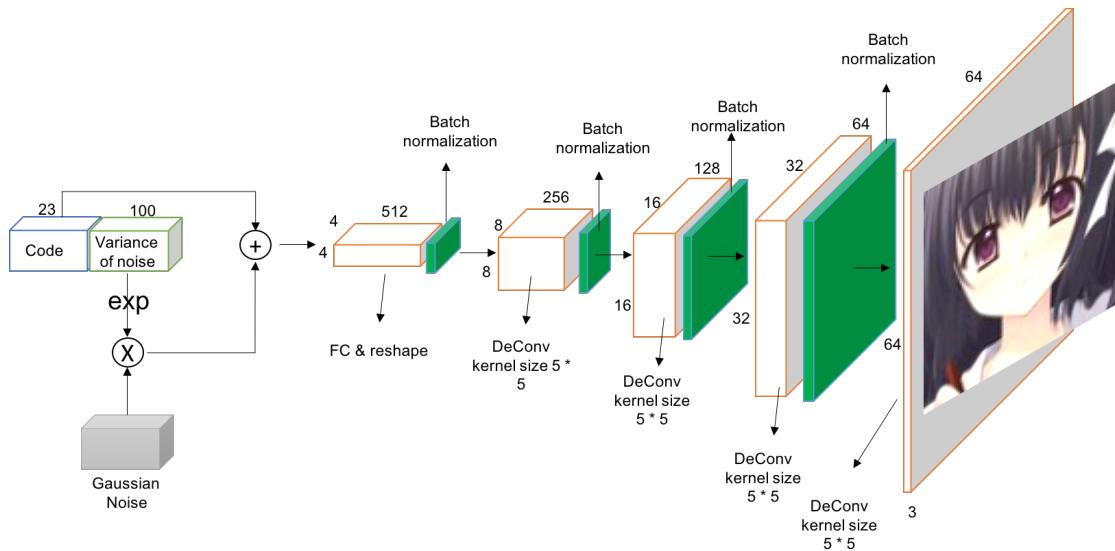
2. Model Description

首先我們了解 GAN 的 Basic objective function 如下，基本上 Generator 和 Discriminator 共用一個 objective function：

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim P_{\text{data}}} [\log D(x)] + \mathbb{E}_{z \sim \text{noise}} [\log (1 - D(G(z)))]$$

我的 model 的架構如下

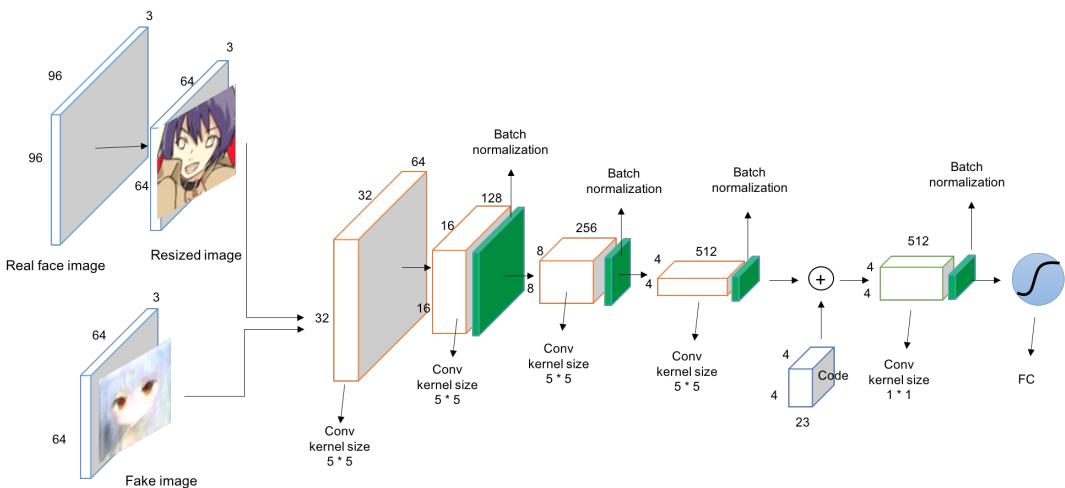
Generator:



說明：

1. 首先把 one hot code 跟 100 維的 noise 加在一起，然後把 123 綴的向量用 fully connected network 輸出成 $4 \times 4 \times 512$ 綴的 data
2. 然後分別經過 3 層的 De Convolution layer 和 Batch normalization，最後在用一層 De Convolution 輸出成圖片

Discriminator:



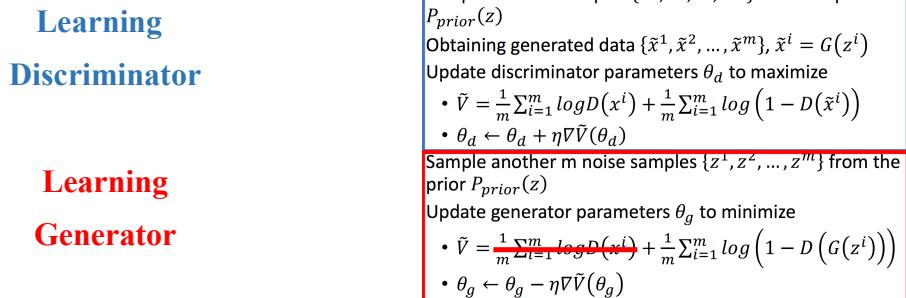
說明：

1. 首先把 real dataset resize 成 $64 \times 64 \times 3$ 的圖片維度
2. 然後分別把 Real image 和 Fake image 塞進 discriminator 的第一層 Convolution，注意的是第一層並沒有用 batch normalization

3. 然後再經過 3 層 kernel size $5 * 5$ 的 Convolution layer 和 Batch normalization，然後把結果與 code 接成 $4 * 4 * (512+23)$ 的維度
4. 最後再過一層 kernel size $1*1$ 的 Convolution 和一層 dense 出一個機率值

3. Methodology and improvement

Training 的過程如下圖所示：



上面藍色部分為 Learning Discriminator，下方紅色部分為 learning Generator，通常 learning 的更新比例要看 dataset 的狀況，由於這次的 dataset 為動漫人物，所以可以發現 discriminator 很容易駕馭 generator，所以我會將 generator 的更新次數變成 discriminator 的兩倍。

不過這其實只是一般的 GAN 更新的方式，為了要讓 model 能夠注意 code (condition)的部分

我參考了 InfoGAN 和 AC GAN 的論文，讓 $G(z, c)$ 可以跟 c 有高度的相關：

$$\min_G \max_D V_I(D, G) = V(D, G) - \lambda I(c; G(z, c))$$

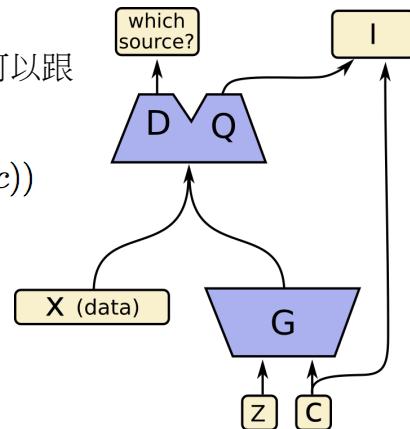
但是 I 這項其實很難去 approximate 它，所以我讓 model 學一個 Q network 去 approximate Posterior。

最後 objective function 就會如下：

$$\min_{G, Q} \max_D V_{\text{InfoGAN}}(D, G, Q) = V(D, G) - \lambda L_I(G, Q)$$

通常 Q network 和 Discriminator 會 share 前面幾個 layer 的 weights。因此 generator 就有辦法在 minimize loss 的時候產出跟 condition 有關的圖片。

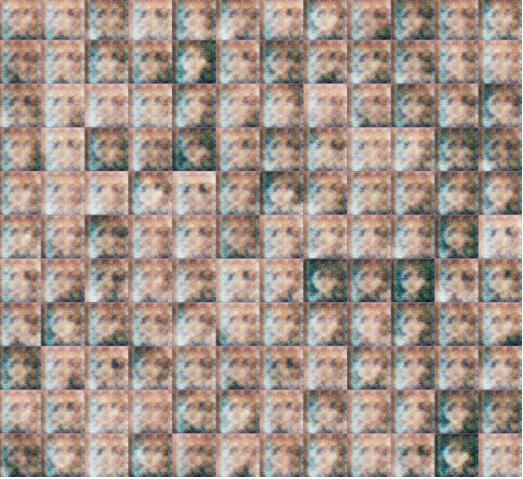
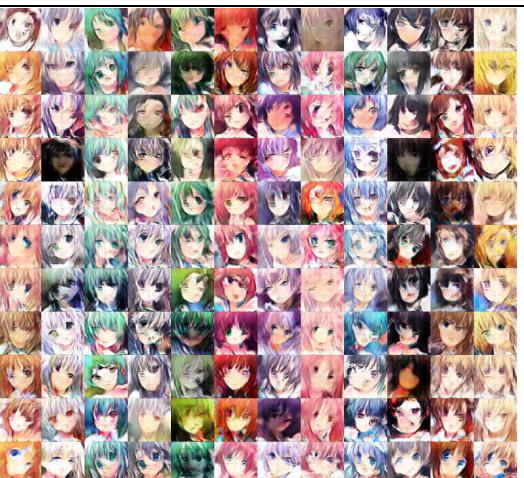
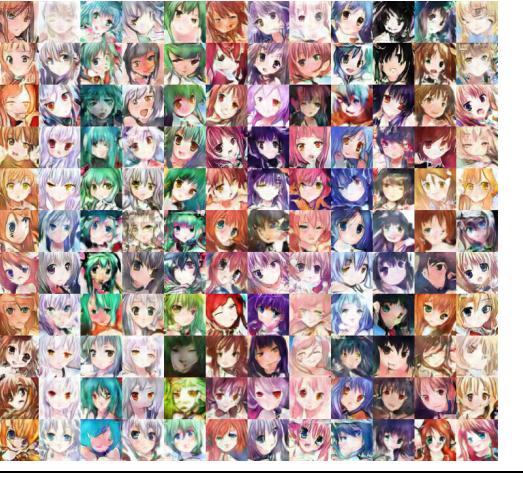
參考論文：Conditional Image Synthesis with Auxiliary Classifier GANs Ref: <https://arxiv.org/pdf/1610.09585.pdf>



4. Experimental settings and observations

Training real dataset: (Only from TA's) Number 33431, Size: $96 * 96 * 3$

batch size = 64, Optimizer: Adam, Learning rate = 0.0002, momentum = 0.5

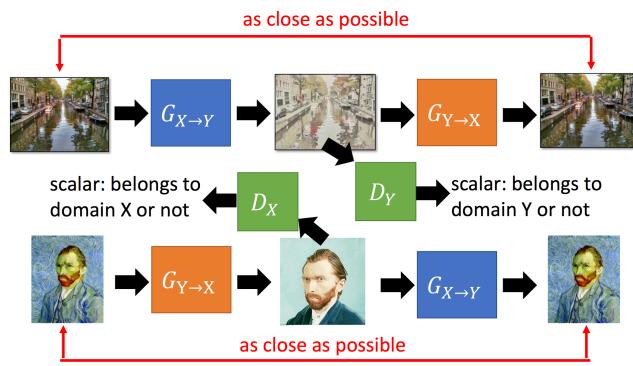
Epoch: 1	Epoch: 10
	
Epoch: 20	Epoch: 30
	
Epoch: 100	Epoch: 170
	

上面的每一行代表一個髮色，每一列代表一個眼睛顏色，可以發現髮色比眼睛的顏色好 train。從觀察可以發現，從第 30 個 epoch 基本上就已經很完整，然後到 100 epoch 以上圖片的品質都相當的不錯，不過有時候 GAN 不一定能更夠

一直的維持平衡，像是在 epoch 170 的時候就已經壞掉，然後把 loss 拿出來看可以發現 Generator 的 loss 飆到 7~8 甚至 1X。

5. (Bonus) Style Transfer:

Bonus 的部分我參考了老師上課的講義，實作了 cycle GAN



Dataset X: 這次作業的 dataset 共有 33431, Size: 96 * 96 * 3



Dataset Y: CelebA



Experimental settings:

Generator: Unet, Discriminator: Basic D

Learning rate for D = 2e-4, learning rate for G = 2e-4, batch size = 4, Optimizer: Adam

Experimental result:

X -> Y 把動畫人物轉成真人	Y->X 把真人轉成動畫人物

從上面可以發現，GAN 的產生真假能力可能還是有待加強，不過 cycle GAN style transfer 的能力還是蠻顯著的，尤其從動畫到真人可以發現真的非常成功。