

# Deep Learning HW3\_1 Report

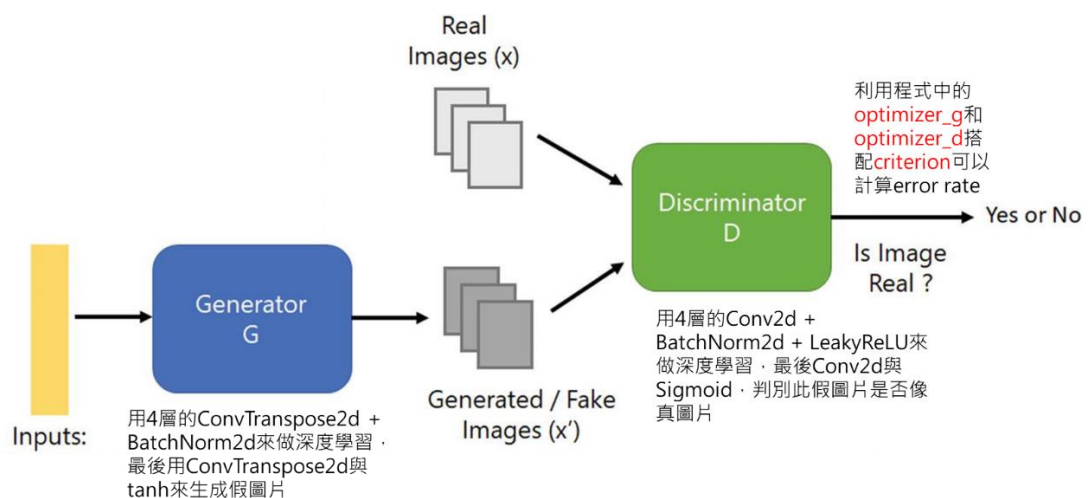
0853411 劉書維

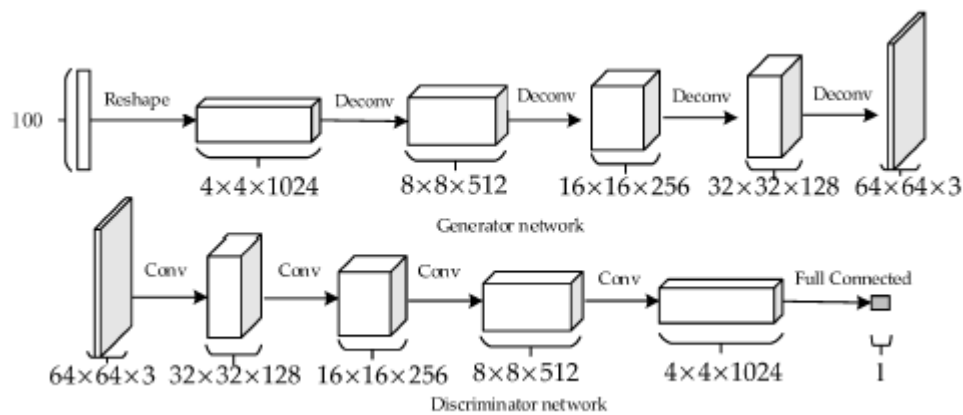
1. Describe how you preprocess the dataset (such as resize, crop, rotate and flip) and explain why.

先將 data 用 ImageFolder 來依照路徑讀入，之後用 transforms 的 Compose 來對資料前處理，依序是 resize 成相同大小 ( 64 )，然後 CenterCrop 裁剪 ( 64 )，最後將它一般化，來更顯著找到圖片的 feature，才會有較好的訓練成效，要處理的資料集再用 DataLoader 來幫助分割 batch\_size。

2. Construct a DCGAN, plot the learning curves for both generator and discriminator, and draw some samples generated from your model.

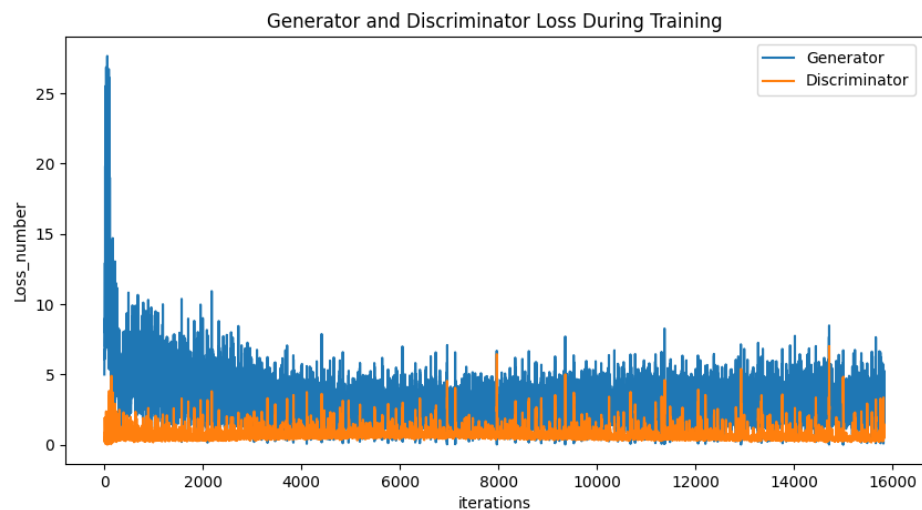
我的 DCGAN 架構如下圖：





( 上圖為助教給予的圖 )

然後各參數： $\text{batch\_size} = 128$ 、 $\text{num\_epochs} = 10$ 、 $\text{lr} = 0.0002$ 、 $\text{beta1} = 0.5$ ，Model 中的參數就如助教原先給的，未做調整。所以每次 epoch 會記錄 1583 次的 G\_loss 和 D\_loss，共做了 10 個 epoch 就會有接近 16000 筆的 error，如下圖所示：

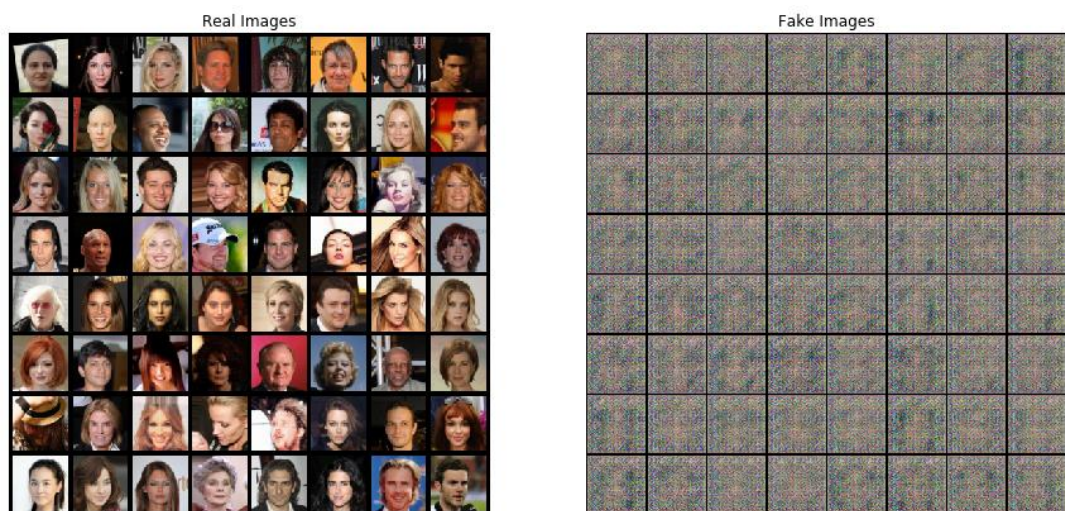


可以看出一開始的 G\_loss 非常高，但是在 2 個 epoch 後就會漸漸穩定，整體走勢持續下降，代表 model 訓練成功雖然仍有一些雜訊，生成的假圖片也不是那麼逼真。



另外，順帶一提，一開始是使用本機端的資源來跑程式，所以只放了 1000

張圖片來做訓練，出來的圖幾乎沒有鑑別度，如下圖：



所以 DCGAN 真的也是很需要足夠硬體資源的程式。

### 3. Implementation details are addressed as follows

#### (a) Model.py :

在這個程式裡面我要架構 Discriminator Network 與 Generator

Network。

Discriminator 訓練出一個 Neural Network 可以分辨偽造出來的圖跟真實的圖，然後把 Generator 出來的圖標記為 0(fake image)，把真實的圖標記為 1，這樣的 training data 丟進我們的 Discriminator Network 做訓練，這就是每一次 Discriminator 訓練的步驟。運算是用 4 層的 Conv2d + BatchNorm2d + LeakyReLU 來做深度學習，最後 Conv2d 與 Sigmoid。

Generator 是要訓練出一個 Neural Network 可以讓 Discriminator 分辨出來的結果越接近真實(1)的結果越好。用 4 層的 ConvTranspose2d + BatchNorm2d 來做深度學習，最後用 ConvTranspose2d 與 tanh 來生成假圖片。

**(b) main.py :**

1. main(): 利用 dataloader 完成資料前處理，並建置 models, optimizers 來設置初始權重和計算，最後是建立一個 criterion 來負責計算 G\_loss 與 D\_loss。  
  
G\_Loss 是生成器產生的偽造圖輸入進鑑定器中的輸出跟 1 的 loss。  
  
D\_Loss 是我們把生成器產生的偽造圖輸入進鑑定器中的輸出跟 0 的 loss+真實的圖輸入進鑑定器中的輸出跟 1 的 loss。就可以開始 train model 了。
2. train(): 就是整個程式的核心了，必須將真實圖片先放入

Discriminator，然後更新其參數與權重。之後利用 Generator 來生成假圖片，在送入 Discriminator 中來計算 loss function，藉由 optimizer\_d、optimizer\_g 和 criterion 來計算 loss，然後反覆重複這樣的過程，最後生成一個幾可亂真 ( ? ) 的假圖片。

**(c) Visualization. Plot the following two charts.**

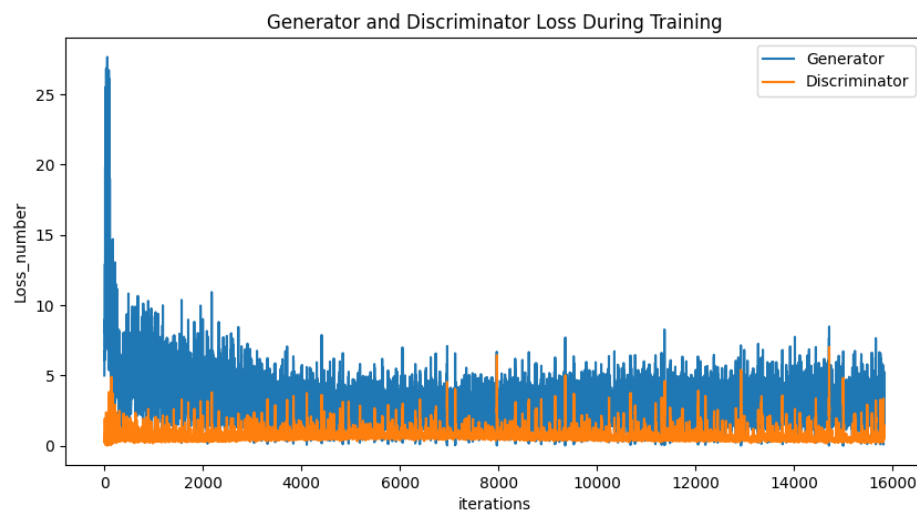
1. original samples and generated samples

可以用 real\_picture 和跑模型是紀錄的 img\_list 即可產生。



2. learning curves for generator and discriminator

藉由跑模型時記錄下來的 G\_losses 和 D\_losses，然後 plot 出來即可產生



**(d) Please do some discussion about your implementation.**

做 DCGAN 最主要的就是先要檢查自己的硬體資源是否充足，且資料是否足夠，不然很容易會製作出很奇怪的假圖片，再來就是掌握資料性質，利用合適的激勵函數，並且將圖片資料的 size 經過一層又一層的 nn 都掌握清楚，就可以開始 train 一個屬於自己的 GAN 模型。

也要注意 GAN 不適合處理離散形式的資料，比如文字與 GAN 存在訓練不穩定、梯度消失、模式崩潰的問題，所以只要這些都能避免就可以生成一個了不起的作品。