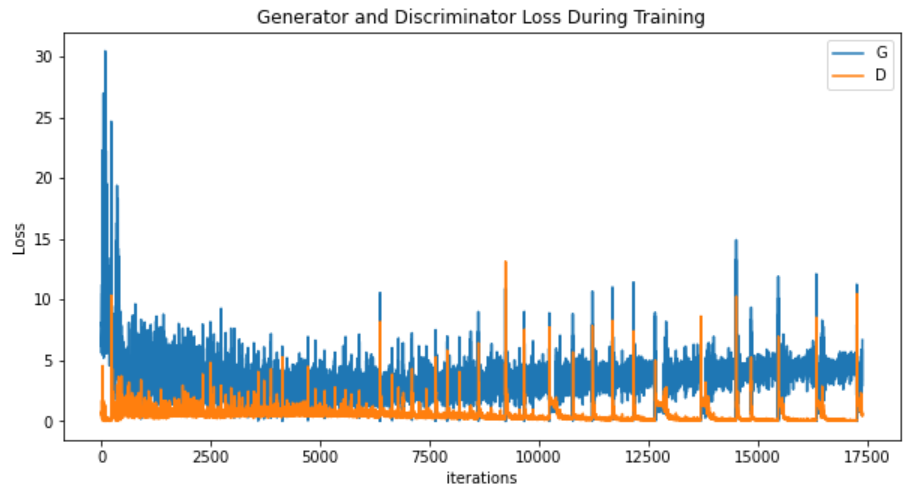
20210523 IMVFX hw2 Report

0716207 呂思函

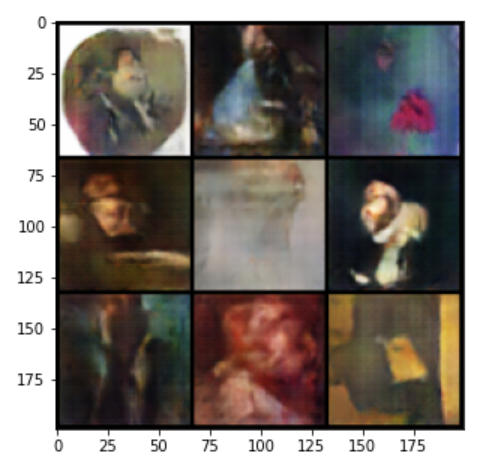
* 1-1 Train a DCGAN model and generate the images (image size 64\*64)



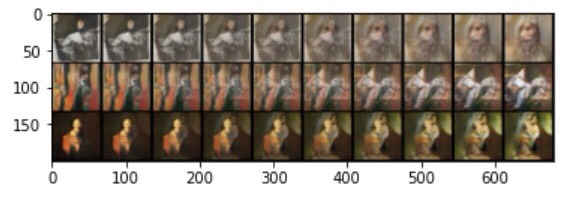
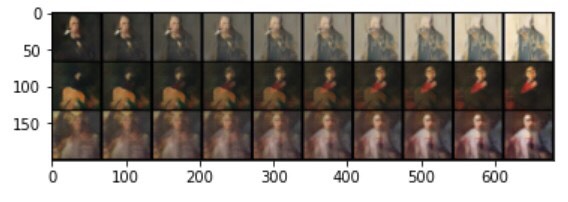
* 1-2 Plot the loss value of discriminator and generator versus training iteration



* 1-3 Store generate 3\*3 grid images in report

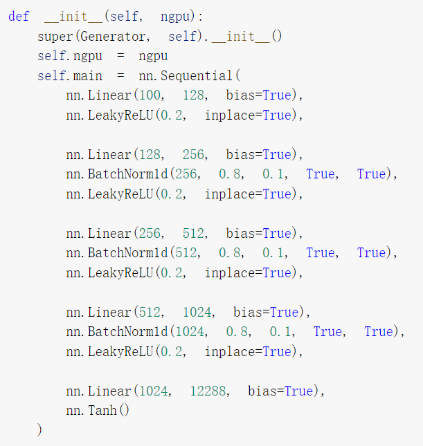


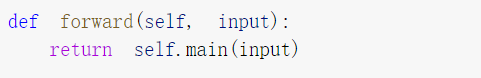
* 1-4 Explore latent space : Interpolation 3 pairs of z vectors and plot a 3\*10 image. Paste the image in report and discuss observation

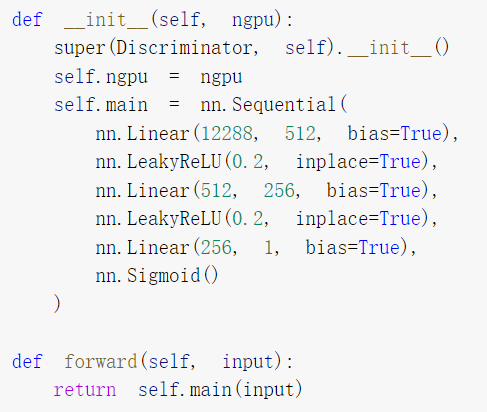


Interpolation works by using known data to estimate values at unknown points. Image interpolation works in two directions, and tries to achieve a best approximation of a pixel's intensity based on the values at surrounding pixels.

因為是讓圖片隨機生成，所以就產生兩組照片貼上來。由上方兩圖可以驗證，經過interpolation之後，圖片色塊會區分得更加清晰、人物輪廓也會比較清楚(對原本不清楚的pixel賦予顏色)

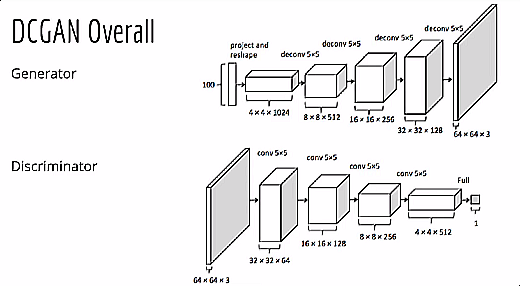
* 1-5 Change the convolution and convolutional-transpose layers in model to linear layers => GAN
  + Generator



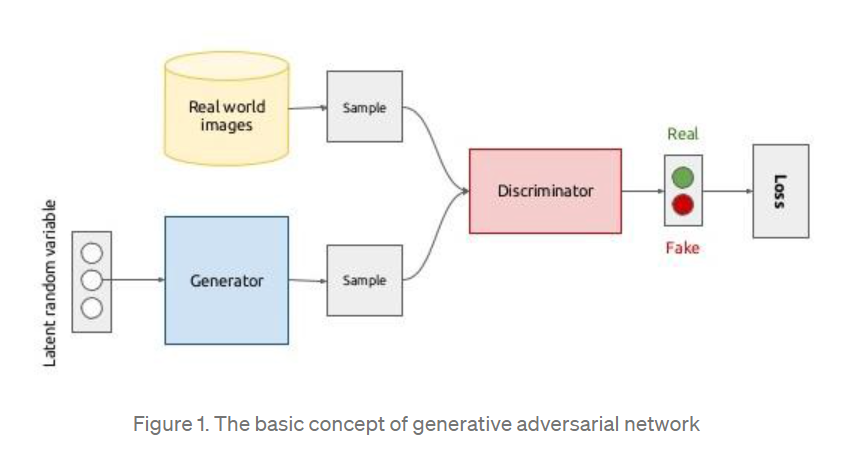
* + Discriminator
* 1-6 & 1-7 Implement and compare the generated results by DCGAN, GAN, and DCGAN+WGAN. Discuss in report.
  + DCGAN能改進GAN穩定訓練的主要原因：
    - 使用fractional-strided代替採樣層，convolution在提取圖像特徵上具有很好的作用，使用卷積層(convolution)代替全連接層(linear)。
    - generator和discriminator中幾乎每一層都使用batchnorm，並將特徵層的輸出歸到一起，不僅加速訓練，也提升了訓練的穩定性和生成結果的質量。（generator的最后一層和discriminator的第一層不加batchnorm）。
    - discriminator和generator中都採用LeakyRELU而不是RELU來防止梯度(gradient)稀疏，而generator的輸出層為Tanh。
    - 選擇Adam優化器訓練，並使用0.0002的learn rate。
* 與DCGAN不同，WGAN主要從損失函數(loss function)的角度對GAN做改進，之後的WGAN即使在全連接層上也可以得到較好的表現结果，WGAN對GAN的改進主要有：
  + - discriminator最后一層去掉sigmoid。
    - generator和discriminator的loss不取log。
    - 對更新後權重強制截斷(clip weight)到一定範圍内，使用clip\_value是0.01，並使用clamp function塞進discriminator。
* 2-1 Introduction (Brief introduction about experiment)

希望能使用不同的GAN來生成油畫，分別有DCGAN, GAN, WGAN。

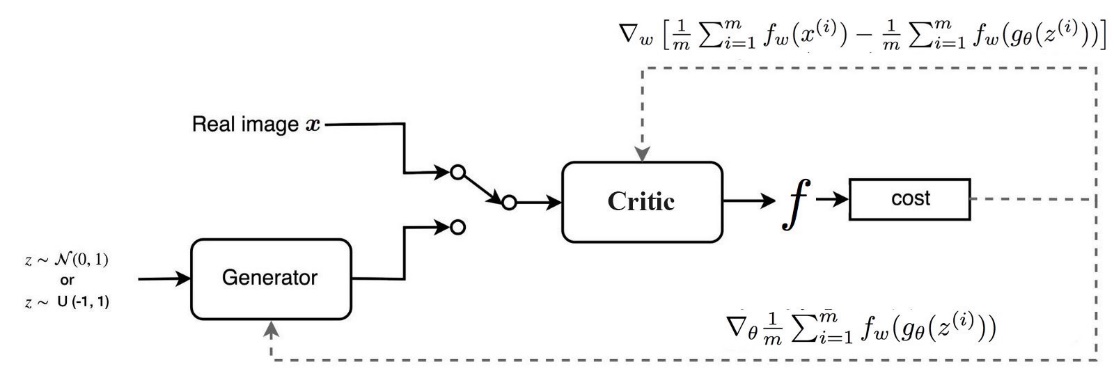
* 2-2 Network structure
  + DCGAN



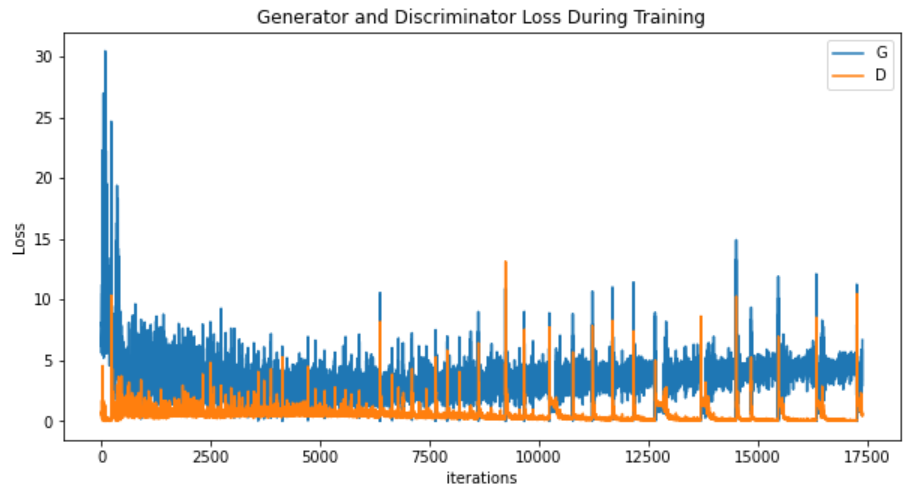
* + GAN



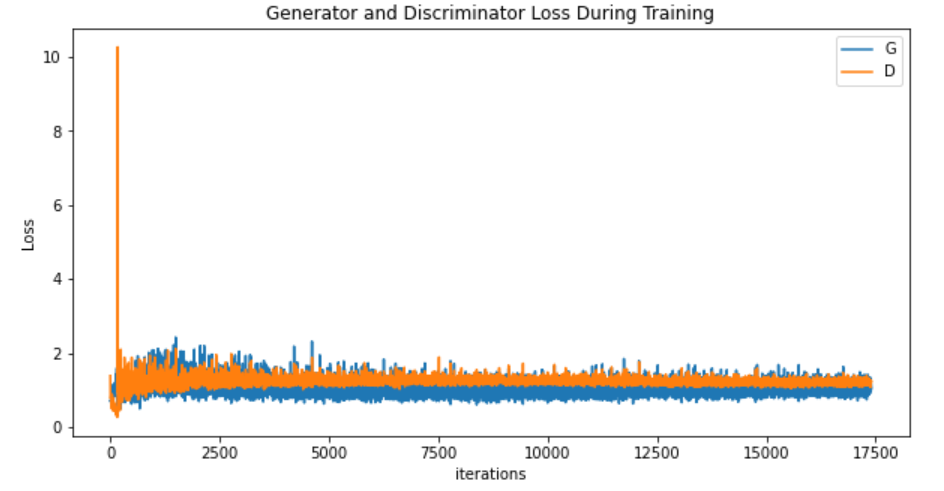
* + WGAN



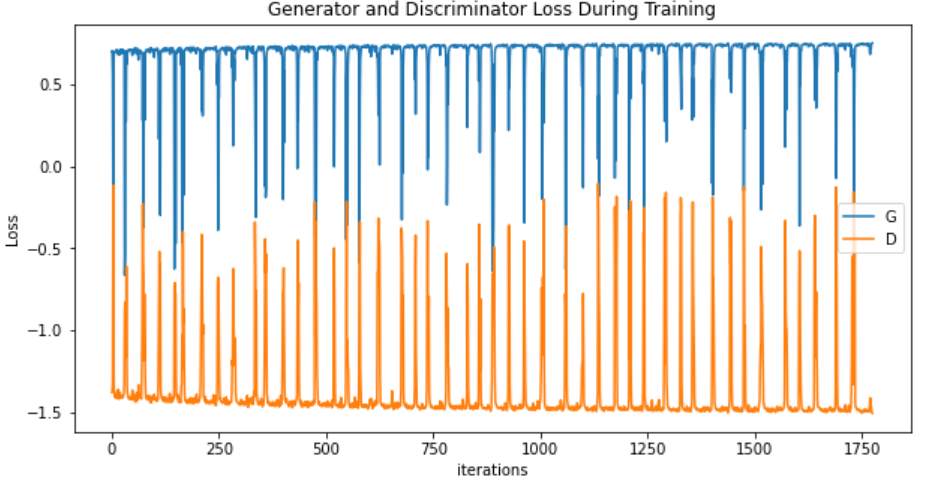
* 2-3 Experiment result
  + 2-3.1 Plot of loss function and hyperparameters
    - DCGAN

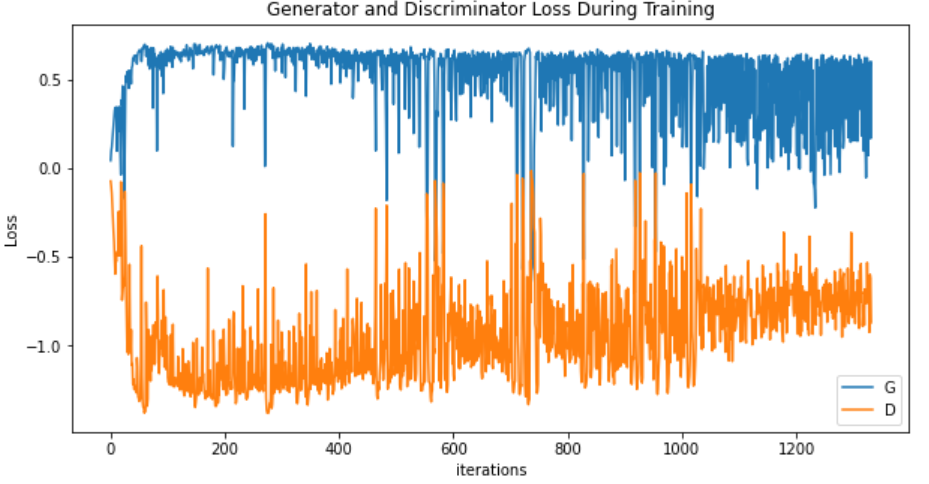


* + - GAN



* + - WGAN





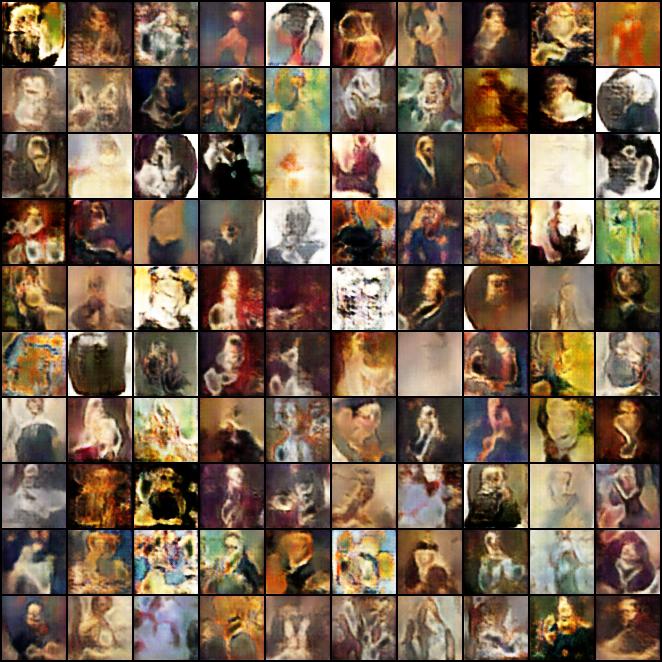
* + 2-3.2 Discussion about the result

前面討論過三種不同的做法了，所以這邊就單純分析一下plot loss的神秘狀況，其實三種狀況圖都看起來蠻可疑的 :

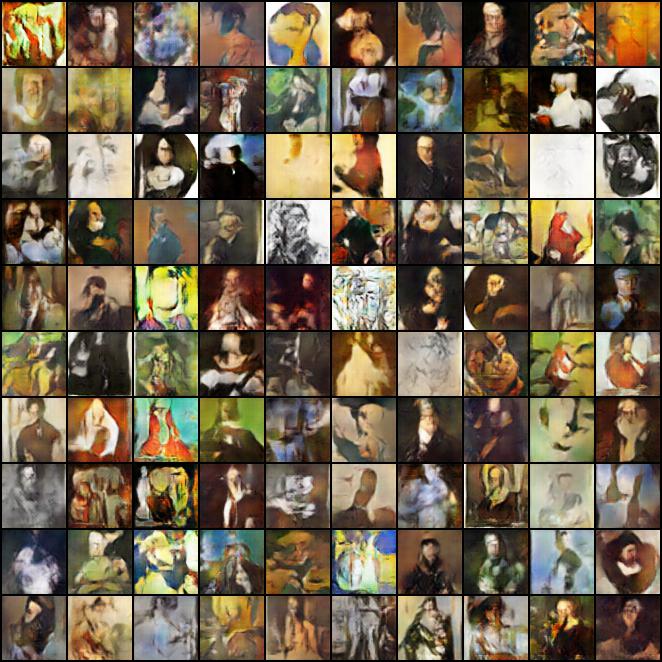
1. 首先第一張DCGAN netG loss較netD loss還多、但振幅有趨緩；netD loss雖然穩定少於netG loss，但振幅卻趨於劇烈，然而隨著iterations增加，從plot loss來看可以發現有一定的週期會讓netD和netG震盪關係相似，或許這種同步就是導致DCGAN有好結果的原因。
2. 第二張的GAN除了最初netD loss有劇烈變化之外，隨著iterations增加，之後的netG和netD振幅都比較均勻、震盪範圍野蠻相近，但是這樣的話可能就會比較難區分real和fake image(電腦會覺得real和fake是很像的東西)。
3. 第三張WGAN的狀況就比較極端，而且每次train的plot loss都差異挺大，所以上面附上兩張WGAN的loss。

netG loss和netD loss同樣變化劇烈，但是netG loss的值會比netD loss大得多，卻可以發現其實震盪也蠻同步的(波形其實蠻對齊的)，所以可能才讓WGAN的結果還蠻不錯的。

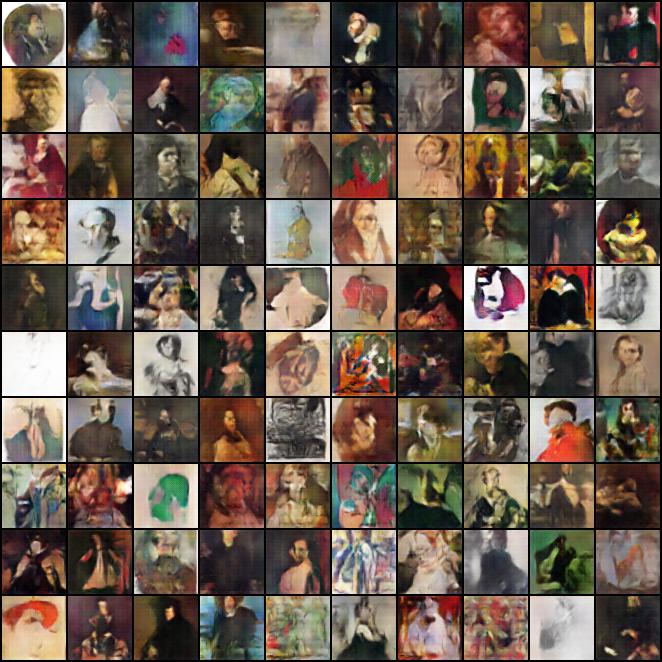
* + 2-3.3 Generate image series of experiment (put images generated in different epochs here)
    - DCGAN(epoch\_num=200)
      * Epoch50



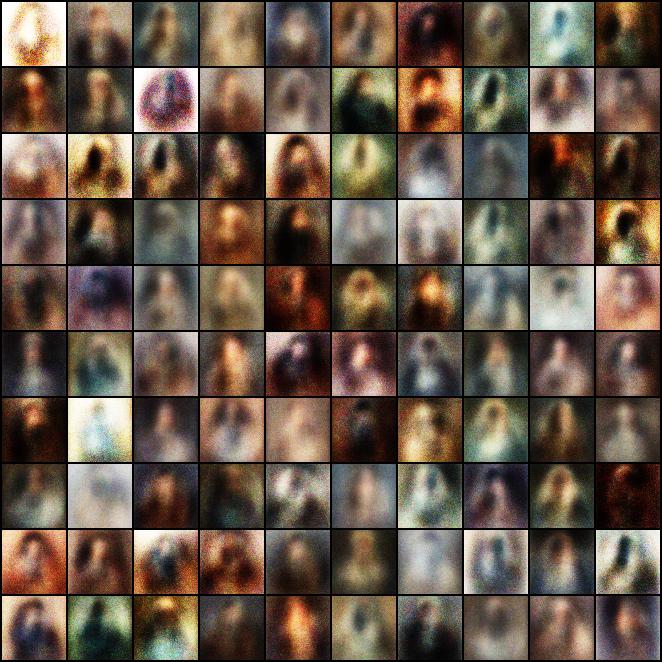
* + - * Epoch100



* + - * Result



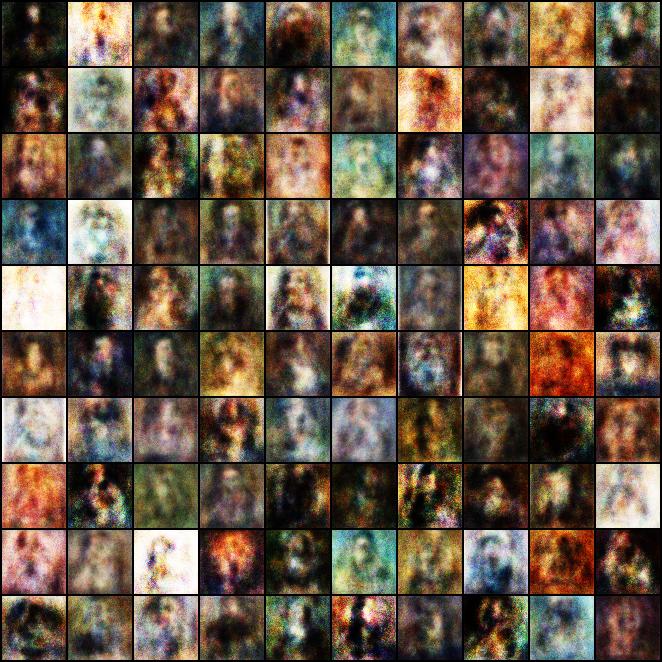
* + - GAN(epoch\_num=200)
      * Epoch50



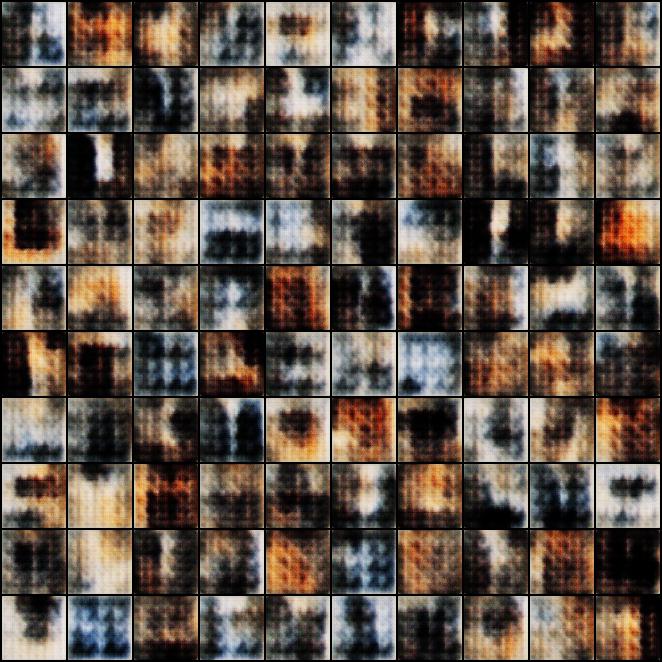
* + - * Epoch100



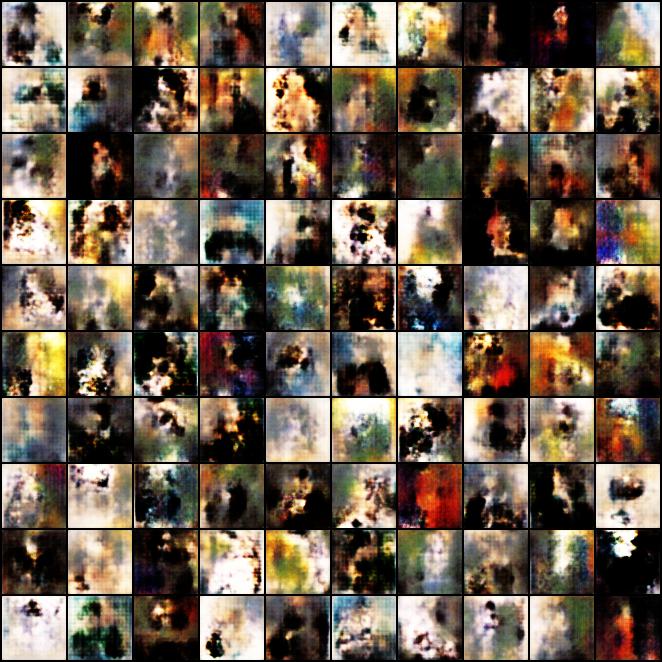
* + - * Result



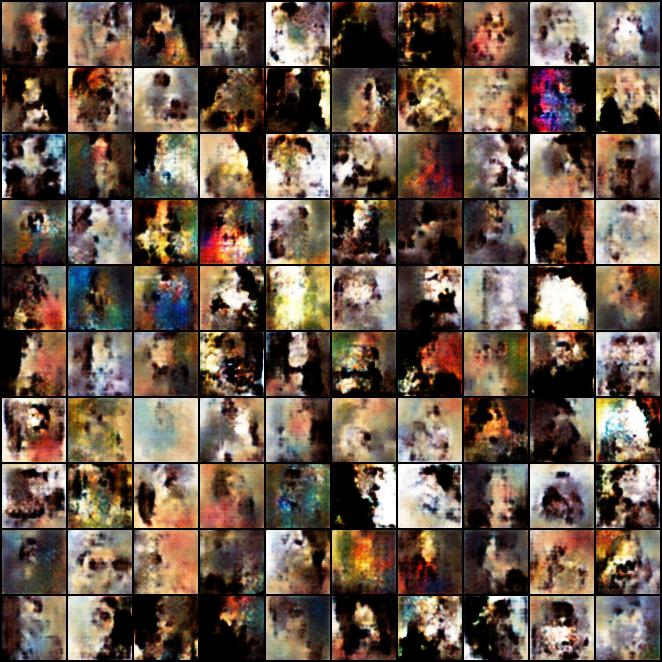
* + - WGAN(epoch\_num=74)
      * 因為colab可以用的RAM會炸掉，所以努力修code的結果最多還是只能到74，然後看前面幾個epoch感覺跟DCGAN的趨勢相近，例如明顯的噪點、色彩的馬賽克區塊、背景人物的成像等，所以推測如果RAM給力的話應該能train出好東西ˊˋ
      * Epoch10



* + - * Epoch50



* + - * Result



* 2-4 Experiment discussion
  + 2-4.1 Discussion about the reason why we use BCE Loss for training (check p.15)

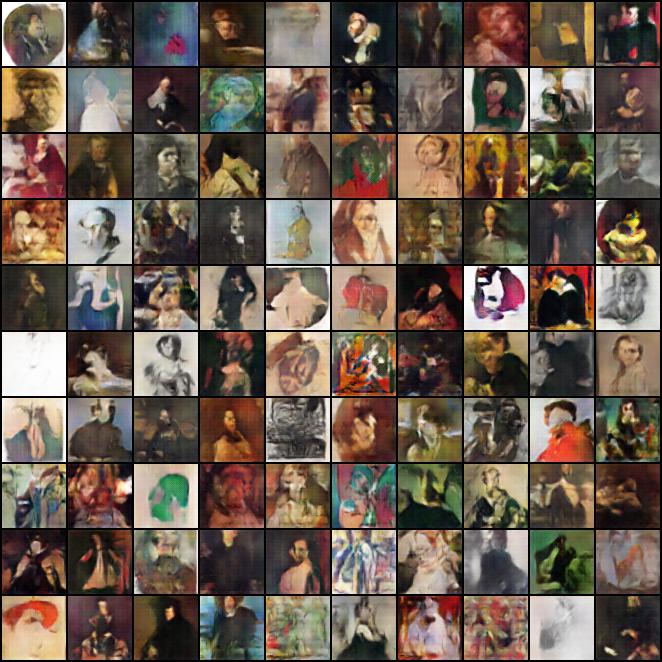
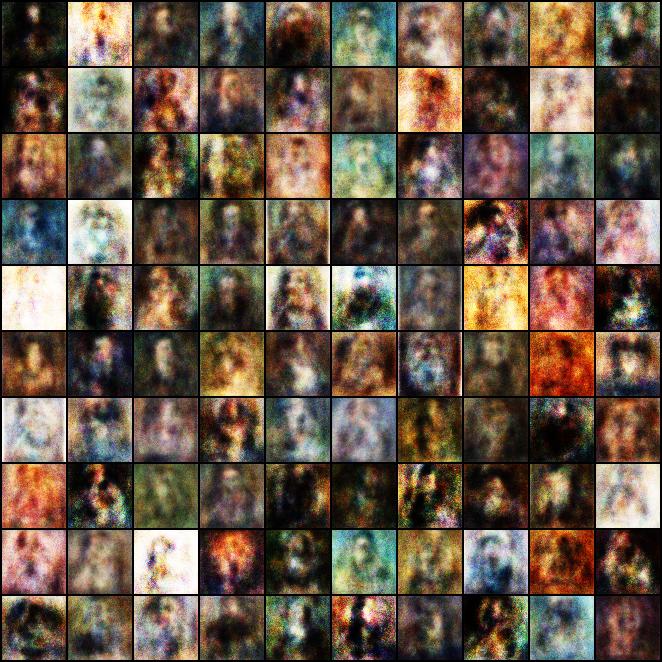
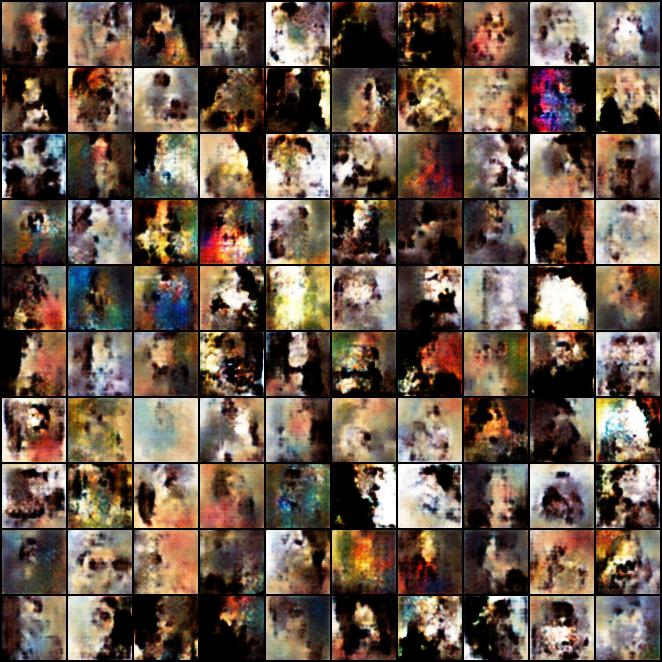
When we are developing a model for a classification problem, the object we are showing is an Apple or an Orange given we know what the particular fruit is. The model gives the output that we can evaluate how good (or bad) is the prediction.

Binary cross entropy compares each of the predicted probabilities to actual class output which can be either 0 or 1. It then calculates the score that penalizes the probabilities based on the distance from the expected value. That means how close or far from the actual value.

* + 2-4.2 Discussion about 1-4 (explain how the z vector influence images)

GAN由一個Generative network與一個Discriminative network組成。Generative network從潛在空間（latent space）中隨機取樣作為輸入，其輸出結果需要盡量模仿訓練集中的真實樣本；Discriminative network的輸入則為真實樣本或生成網路的輸出，其目的是將Generative network的輸出從真實樣本中盡可能分辨出來，而Generative network則要盡可能地欺騙Discriminative network。

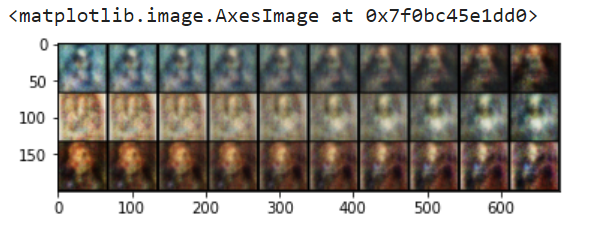
兩個網路相互對抗、不斷調整參數，最終目的是使Discriminative network無法判斷Generative network的輸出結果是否真實。

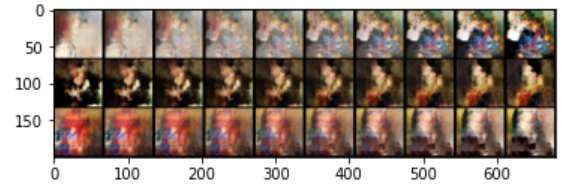
* + 2-4.3 Discussion about 1-1, 1-5, 1-6 (put results here and compare the results generated by the three different network)
    - DCGAN
    - GAN
    - WGAN
    - Discussion
      * Interpolation :

DCGAN



GAN



WGAN

從上方內插排出的結果可以看到，DCGAN的結果最強，但是卻和WGAN結果相似的會有點馬賽克感，中間的GAN是會是模糊的樣子、然後隱約會有彩色的雜訊，但是其實三種方式train出來的結果好像差不多，就是略有形體、顏色分明，但是人物衣服的輪廓尚不清晰。

* + 2-4.4 Others about the experiment you want to discuss

陰錯陽差使用了GAN+WGAN，結果效果非常的差，只能說convolution比linear的連接方式會讓成果好很多，GAN+WGAN會在很前面的epoch就顏色定型，然後就救不回來了，甚至調整其他參數(lr, loss\_D, loss\_G等)會讓結果更崩潰，例如產生許多彩色雜訊。

以下是GAN+WGAN的plot loss，結果與GAN傾向相似。

