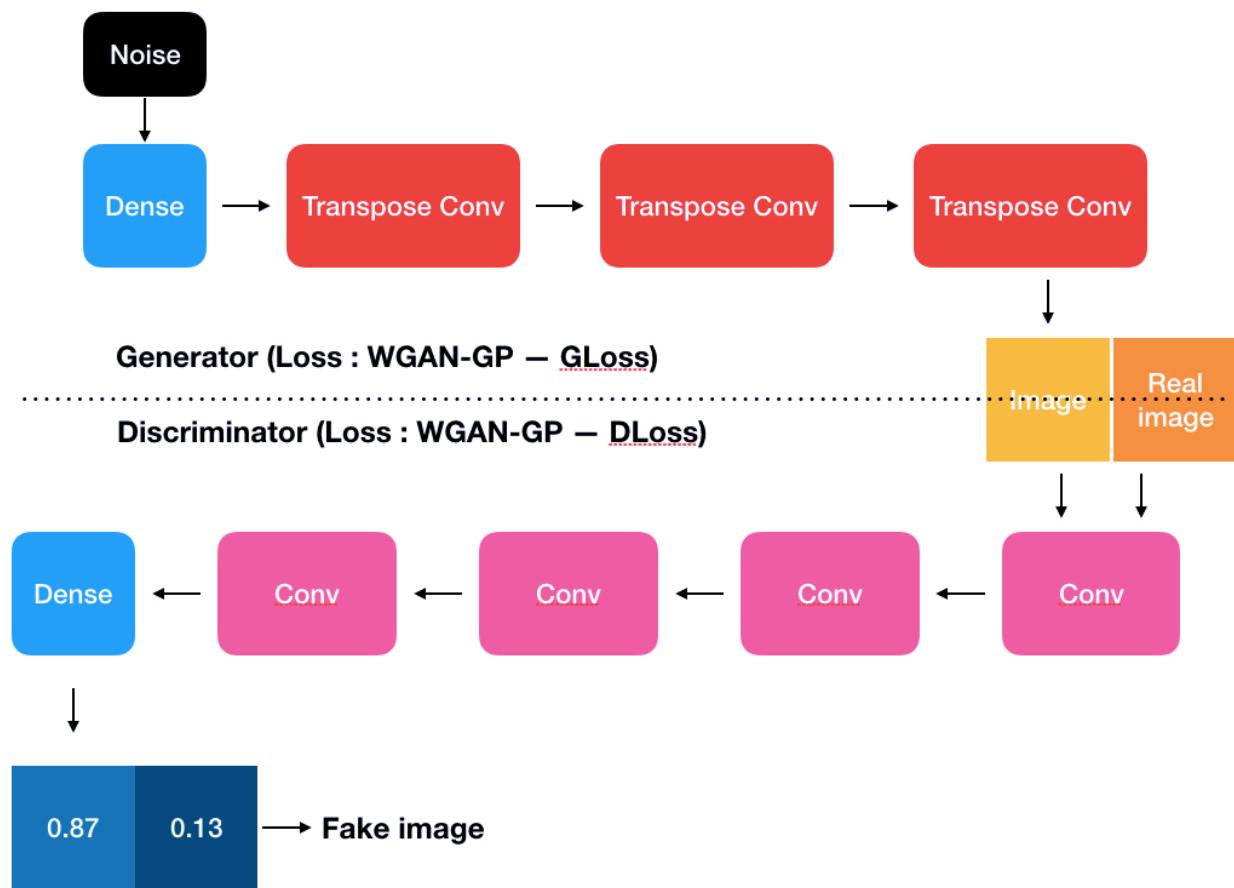


MLDS HW3 GAN

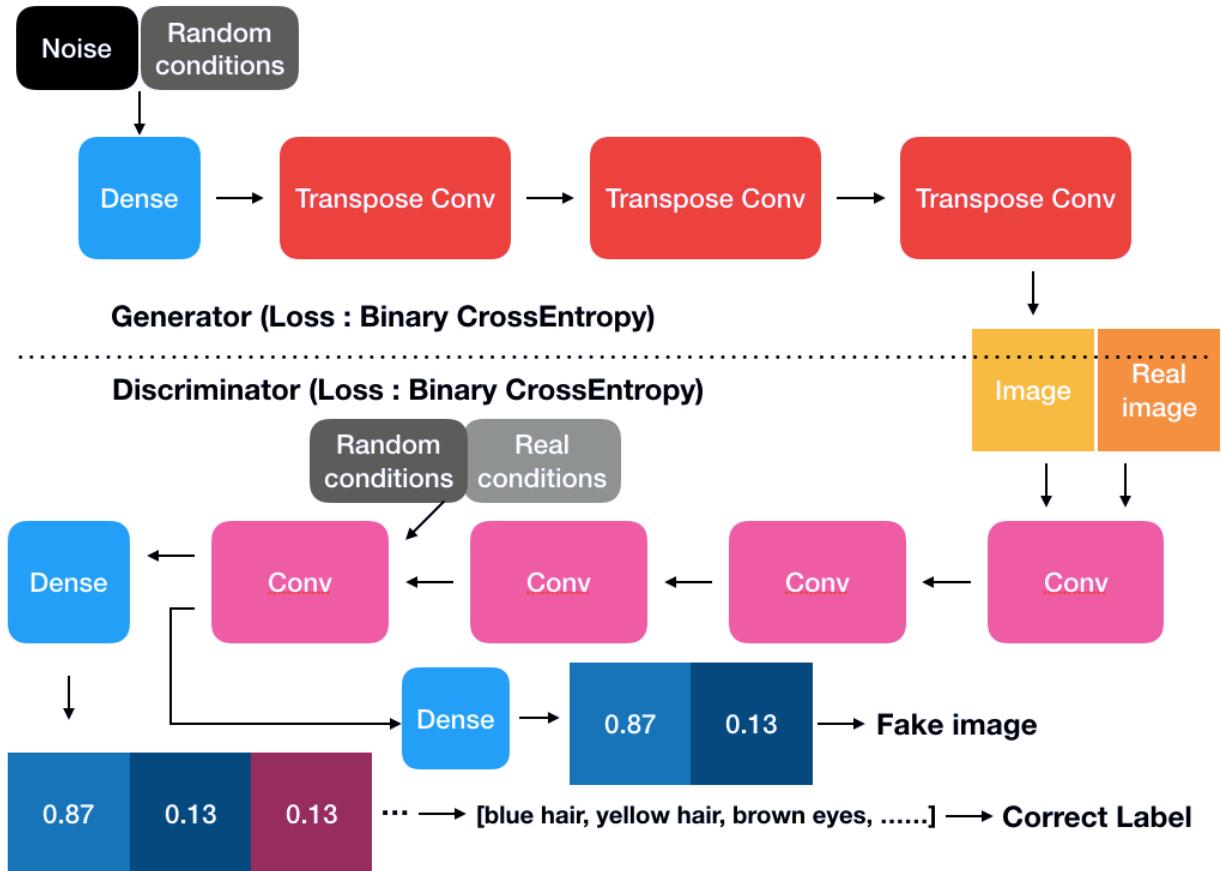
學號：b05902031, 系級：資工二, 姓名：謝議霆, 學號：b05902008, 系級：資工二, 姓名：王行健

1. Model Description

- Image generation (**WGAN-GP**)



- Text to image

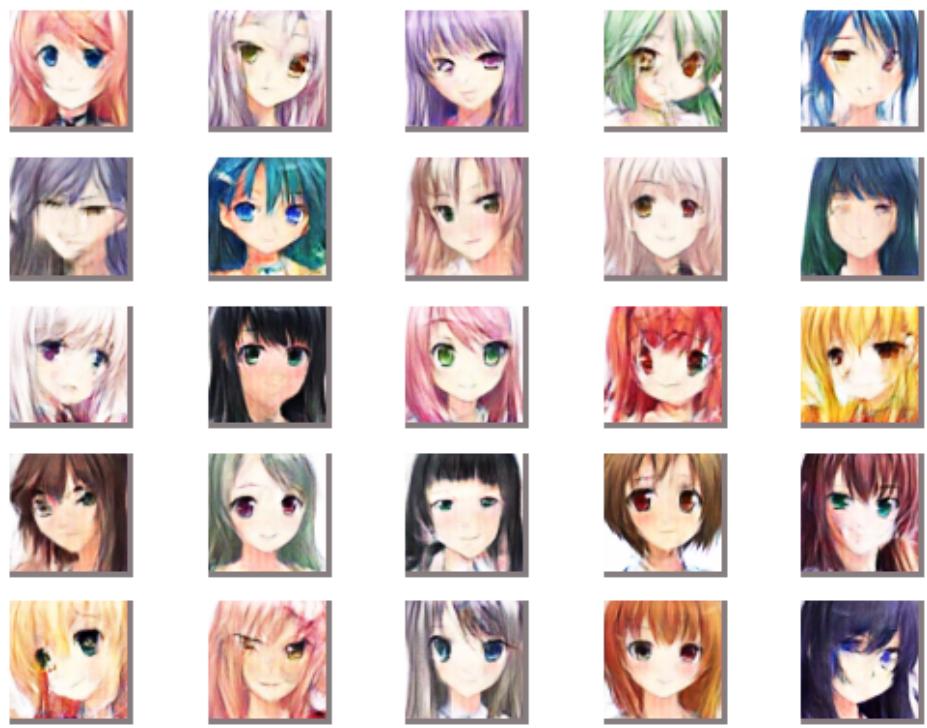


2.Experiment settings and observation

Image Generation

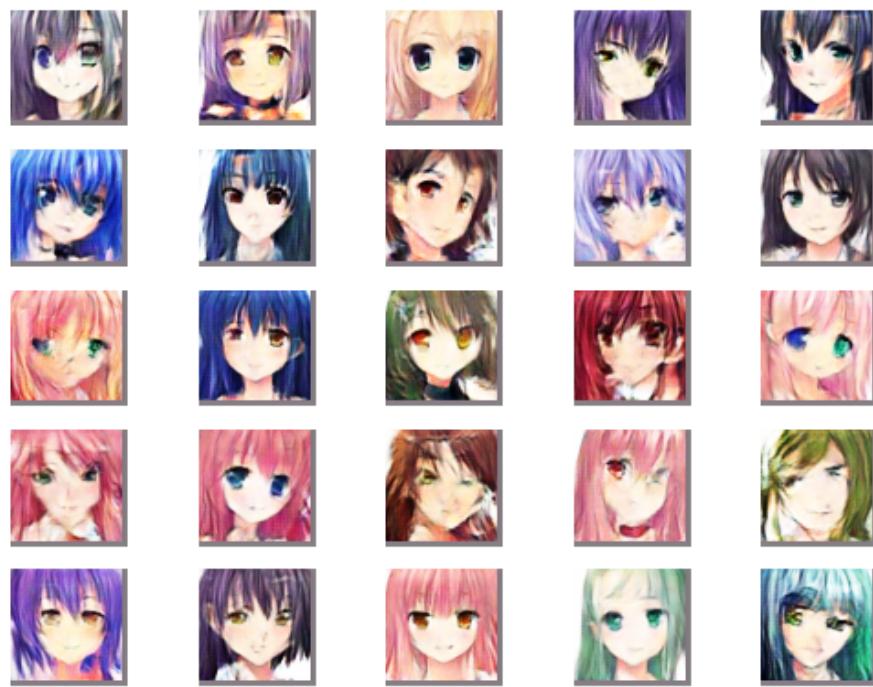
Filter 數量 : many

- Generator : TransposeConv2d(512->256) -> TransposeConv2d(256 -> 128) -> TransposeConv2d(128 -> 3)
- Discriminator : Conv2d(3 -> 128) -> Conv2d(128 -> 256) -> Conv2d(256 -> 512) -> Conv2d(512 -> 1024)



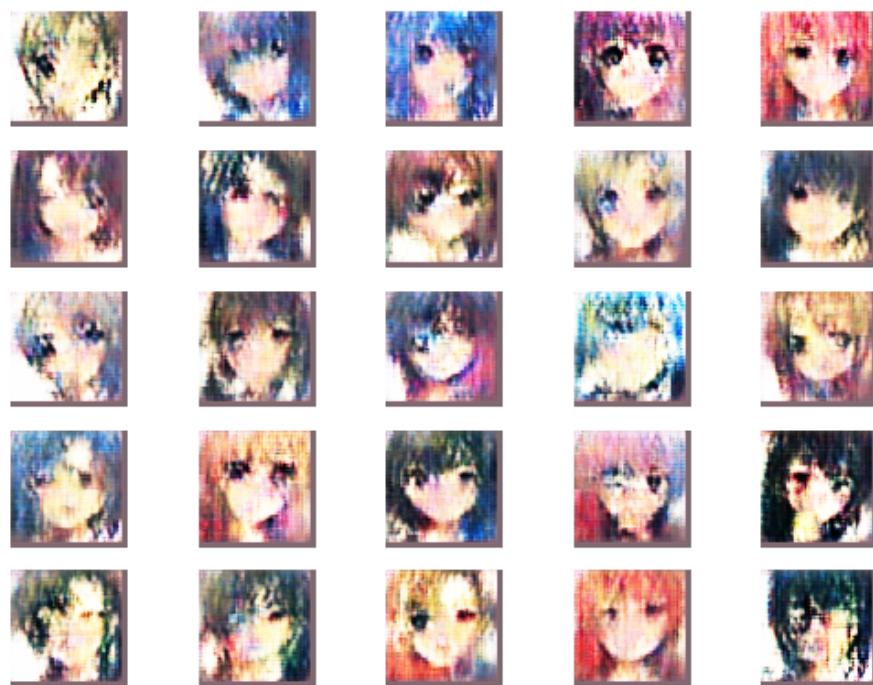
Filter 數量 : medium

- Generator : TransposeConv2d(256->128) -> TransposeConv2d(128 -> 64) -> TransposeConv2d(64 -> 3)
- Discriminator : Conv2d(3 -> 64) -> Conv2d(64 -> 128) -> Conv2d(128 -> 256) -> Conv2d(256 -> 512)



Filter 數量 : few

- Generator : TransposeConv2d(24->12) -> TransposeConv2d(12 -> 6) -> TransposeConv2d(6 -> 3)
- Discriminator : Conv2d(3 -> 6) -> Conv2d(6 -> 12) -> Conv2d(12 -> 24) -> Conv2d(24 -> 48)



Text to image

Filter 數量 : many

- Generator : TransposeConv2d(512->256) -> TransposeConv2d(256 -> 128) -> TransposeConv2d(128 -> 3)
- Discriminator : Conv2d(3 -> 128) -> Conv2d(128 -> 256) -> Conv2d(256 -> 512) -> Conv2d(512 -> 1024)



Filter 數量 : medium

- Generator : TransposeConv2d(256->128) -> TransposeConv2d(128 -> 64) -> TransposeConv2d(64 -> 3)
- Discriminator : Conv2d(3 -> 64) -> Conv2d(64 -> 128) -> Conv2d(128 -> 256) -> Conv2d(256 -> 512)



Filter 數量 : few

- Generator : TransposeConv2d(24->12) -> TransposeConv2d(12 -> 6) -> TransposeConv2d(6 -> 3)
- Discriminator : Conv2d(3 -> 6) -> Conv2d(6 -> 12) -> Conv2d(12 -> 24) -> Conv2d(24 -> 48)



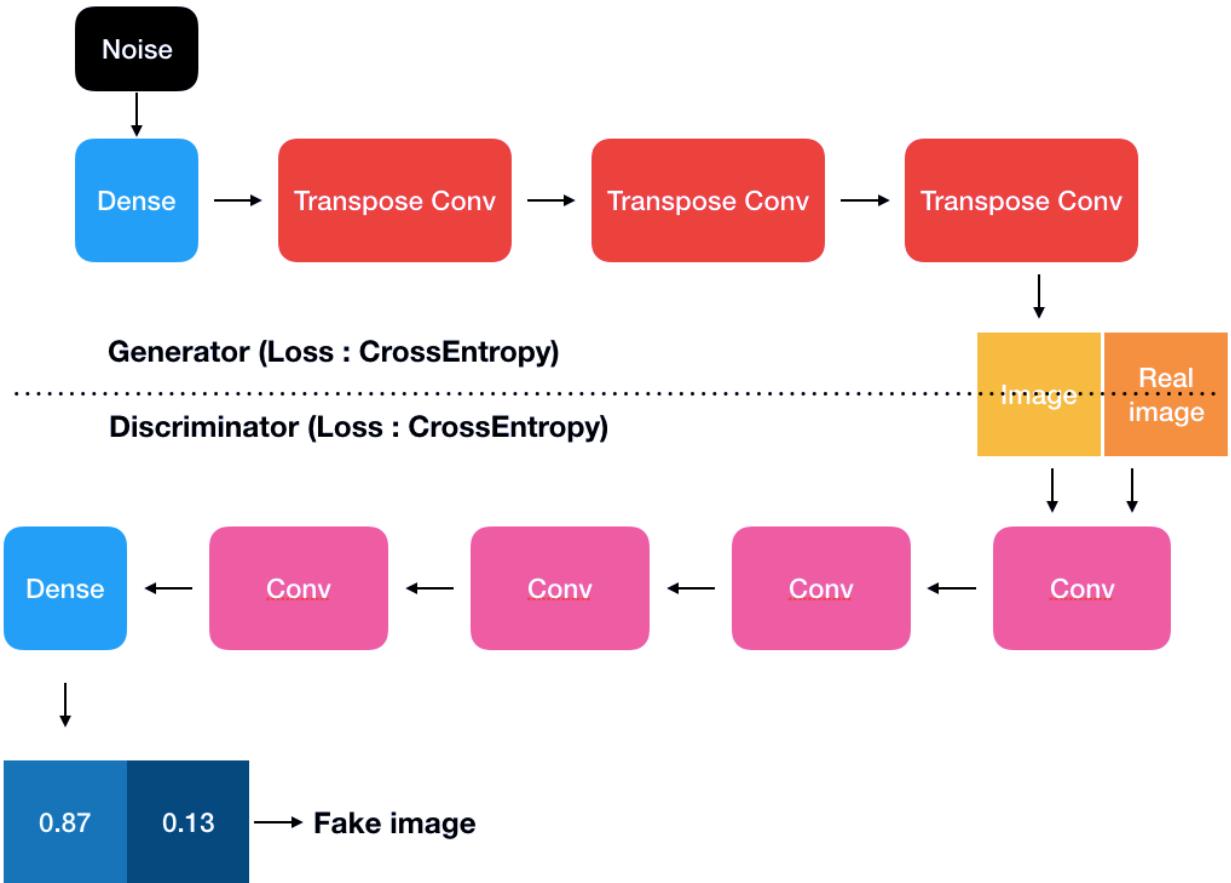
分析

- Filter數量越少，能學到的線條便越粗糙，只看得出臉的形狀和五官頭髮。不過在filter數量 medium跟many的圖片大致上是同等細緻、沒什麼差別的。

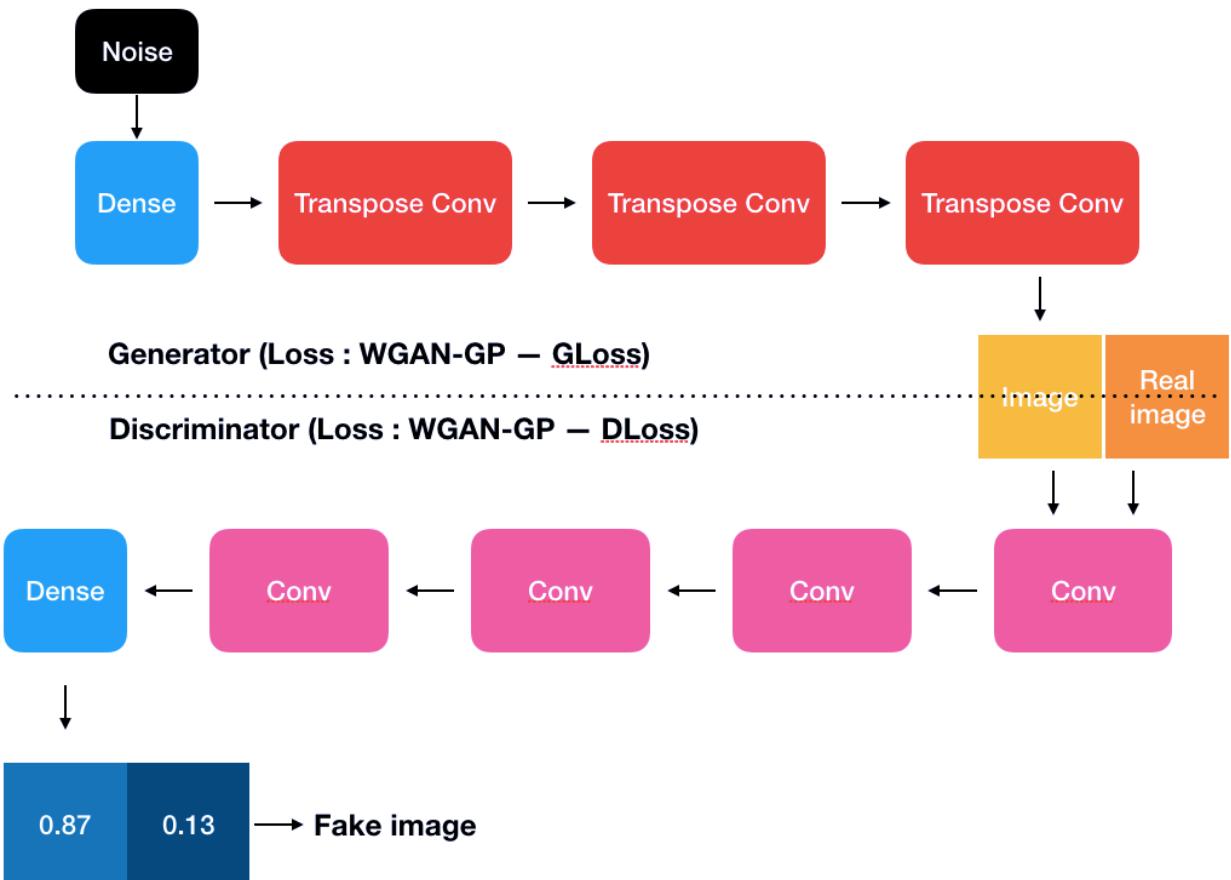
3. Compare your model with WGAN, WGAN-GP, LSGAN

Model Description

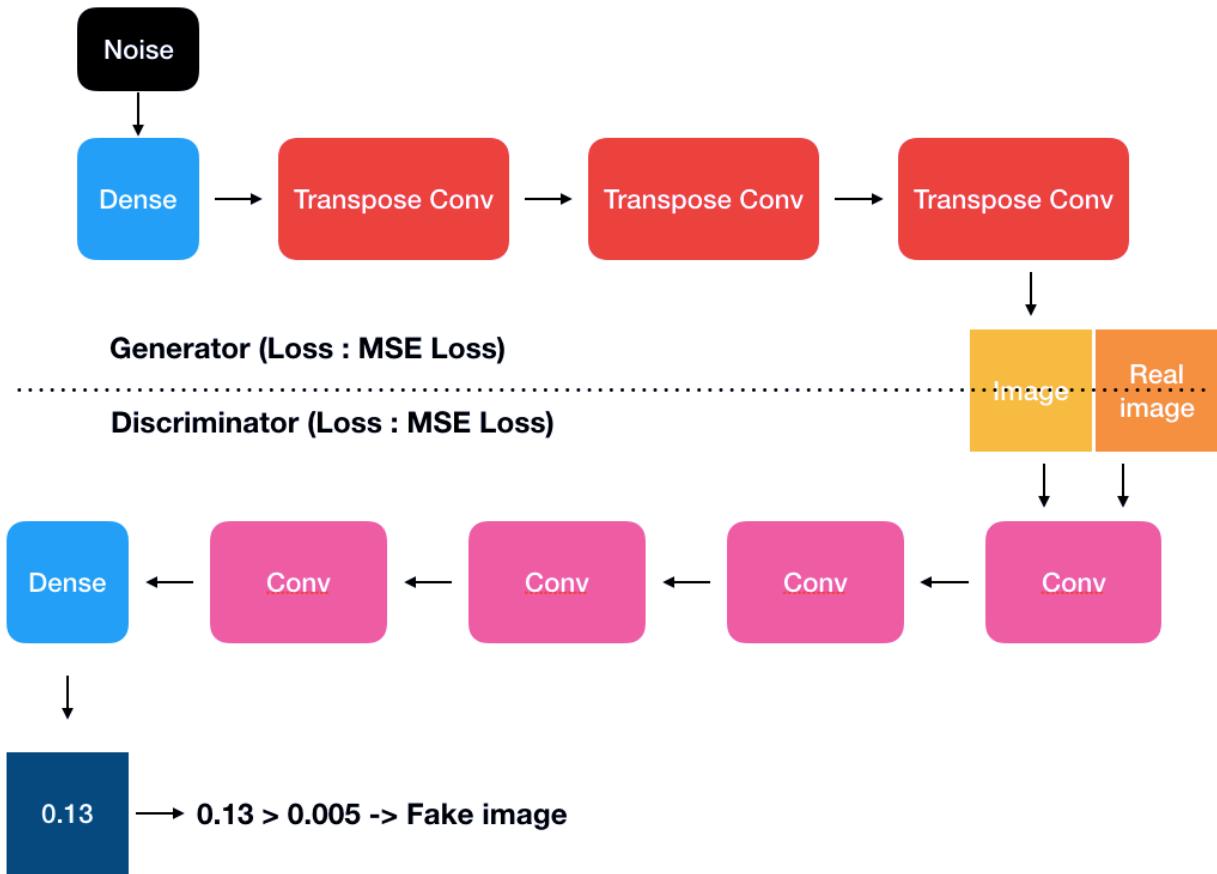
- GAN



- WGAN-GP

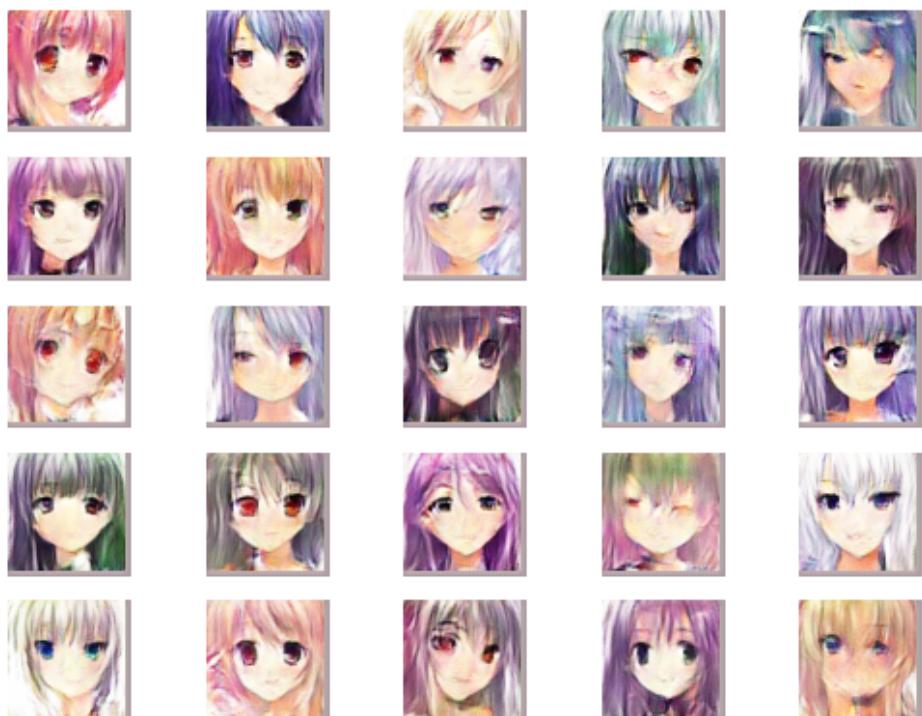


- LSGAN

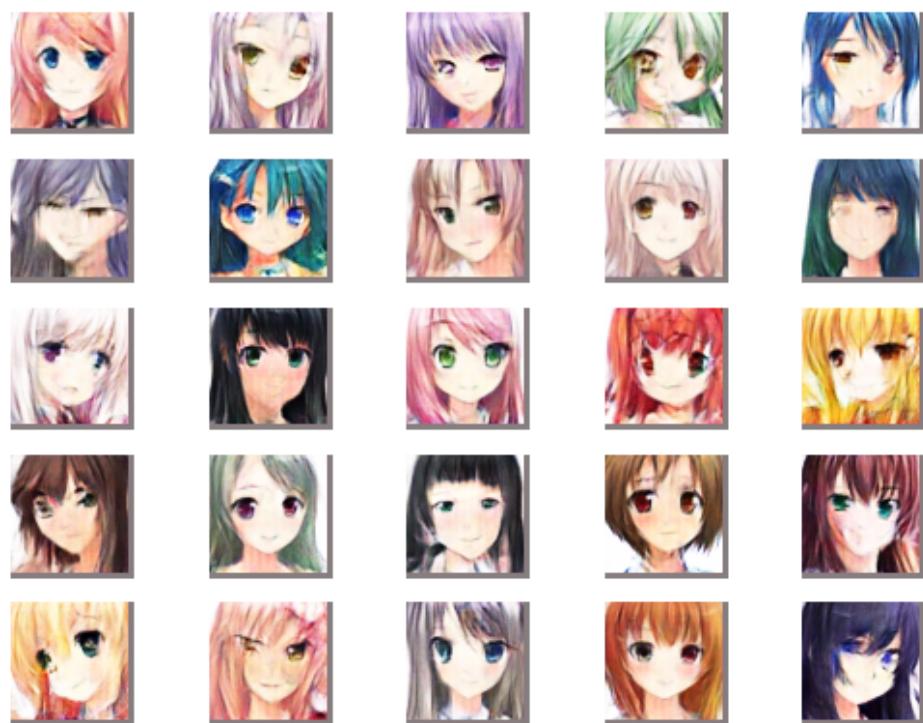


Result

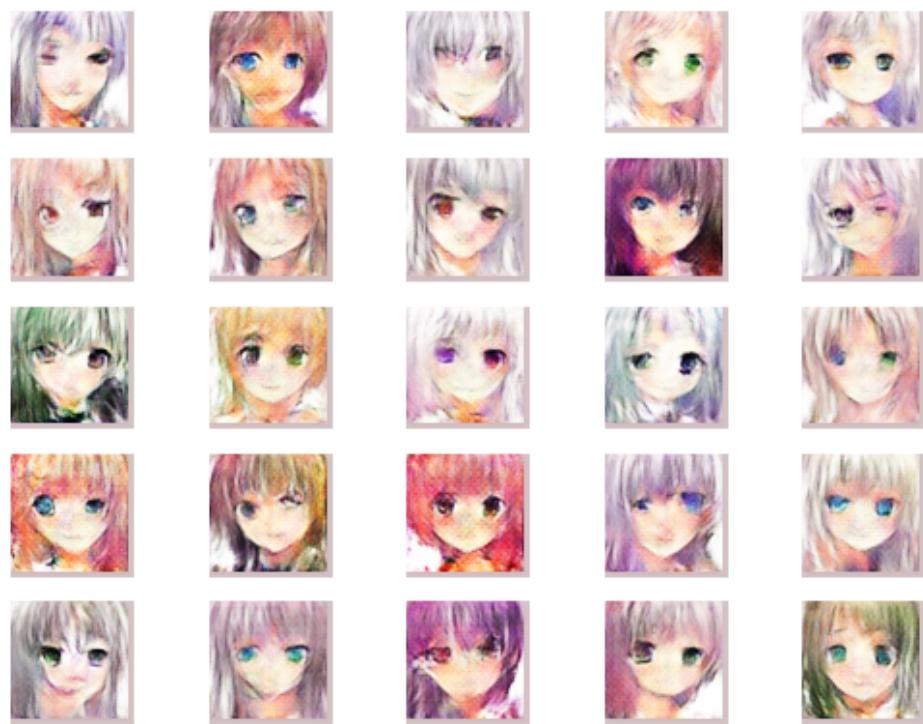
- GAN



- WGAN-GP



- LSGAN



Analysis

- 這邊的GAN跟LSGAN的輸出，相較於WGAN-GP來說，訓練的時間都比較少，因為這兩個訓練到最後都容易mode collapse，原因是GAN跟LSGAN的Loss function都是尋找一個最低點逼近他，但是WGAN-GP的loss function做的是在訓練Generator的時候，讓評分的直越靠近越好，而在訓練Discriminator的時候，讓評分的值越遠越好，較不容易發生mode collapse的情形。

4.Training tips for improvement

在<https://github.com/soumith/ganhacks>的tip中，選用1, 5, 6

5: Avoid Sparse Gradients: ReLU, MaxPool

- 實作：中間不用Max pool，在Generator使用Transpose Conv2d加上stride，Discriminator使用Conv2d加上stride，中間的Activation function都使用Leaky Relu(slope = 0.2)

不用leaky relu	使用leaky relu
	

- 分析：大致上來說兩種模型的表現是差不多的，但是放大仔細來看，會發現左邊的圖，臉部的部分有很多會有灰灰髒髒像是污漬的區塊在上面，在使用leaky relu之後，這樣的問題便減少了許多。

1. Normalize the inputs

- 實作：將input的圖片(0~1)，乘以二再剪掉一，標準化從-1到1，最後在generator的output加上一層tanh。

不標準化到(-1,1)，generator的output沒用tanh	使用tip

- 分析：兩張圖可以很明顯的看出來左邊的只能看出輪廓，卻沒有顏色的表現在裡面，相較於右邊在使用這個Tip之後顏色就好多了許多。

6: Use Soft and Noisy Labels

- 實作：再拿取每一筆資料(batch)，訓練Generator和Discriminator時，將Generator生成的圖片的label隨機給成0.0~0.3，將真實資料的label隨機給成0.7~1.0。

不使用Noisy labels	使用Noisy label

- 分析：在test分數上面這兩張都是25/25，仔細放大觀察也沒看出沒使用這個tip有什麼缺失，所以這兩個模型的表現是差不多的。

5.Style Transfer

Result

- 我們兩種圖片的domain分別是普通的馬跟斑馬

普通的馬(original)	斑馬(fake)
	
	

分析

- 我們是用<https://github.com/junyanz/pytorch-CycleGAN-and-pix2pix>的Cycle Gan的pre-trained model來測試馬的圖片。
- 從下面的圖片可以觀察出這個模型比較難判斷出從正面拍攝的馬

普通的馬(original)



斑馬(fake)

