Embedding System Proposal Report
Team 6

Team Member：

91299482X　Lena Mattea Stöxen

61275047H　黃聖崴

61275055H　謝允喆

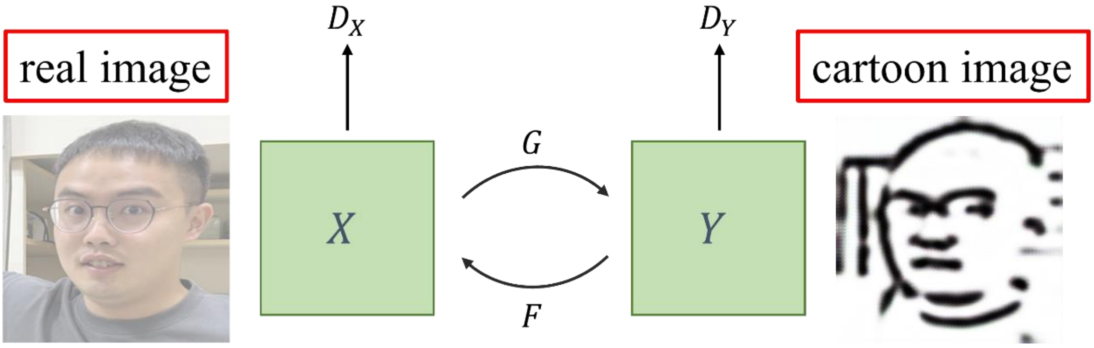
61275068H　李柏昇

1. Proposed Model – Cycle GAN

In the section where we select the model for our software, we chose Cycle GAN as our implemented model. The reason we selected Cycle GAN for our implementation is that it can learn how to transform data from one distribution (B) to resemble the distribution of another (A) after structural learning.

The architecture of Cycle GAN consists of two generators and two discriminators. Discriminator for judging whether the image is generated or original and Generator for generating the fake image to disturb discriminator's judgement.

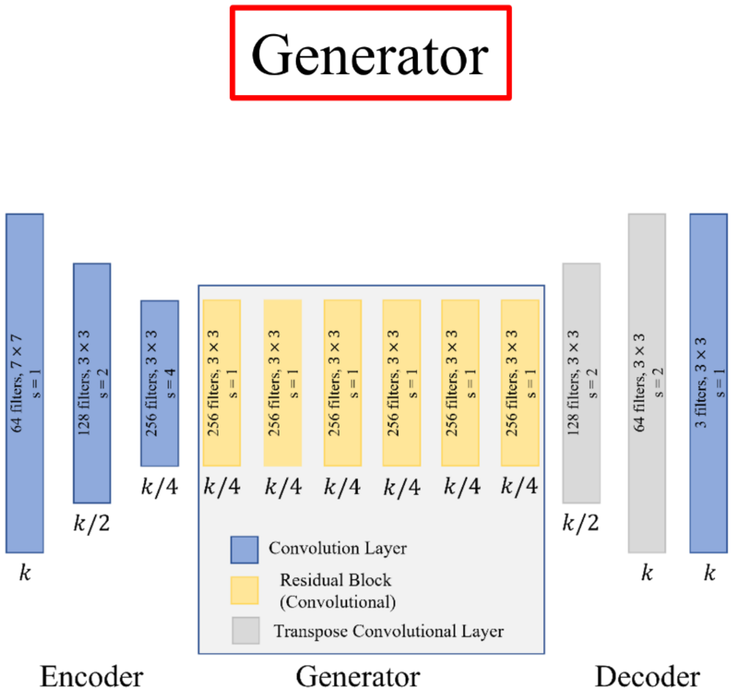
The reason for having two sets of generators and discriminators is to avoid situations where data from distribution A generates types that cannot be recognized as resembling distribution B. Specifically, Generator A (G) generates images resembling the distribution of Data Distribution B, which are then evaluated by Discriminator B (DY). Similarly, Generator B (F) generates images resembling the distribution of Data Distribution A, which are evaluated by Discriminator A (DX). The purpose of this setup is to "reconstruct" the original appearance of the images through the other set of generators and discriminators.



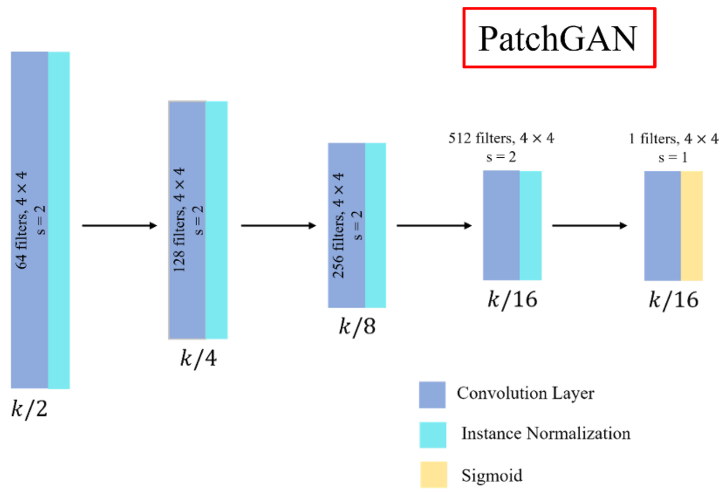
2. Cycle GAN Model

In Generator model, it consists of three parts： Encoder part, Generator part and Decoder part.
The Encoder part do the “down-sampling” operation to extract the feature from the image, the Generator part is composed of six residual Block to keep the image feature and non-linear transformation and the Decoder part do the “up-sampling” operation to restore the low-resolution image to high-resolution image.

In Discriminator model, usually use PatchGAN model, which will use the convolution layer extract feature by down-sampling operation.The Fully Connected Layer or Global Average Pooling to give a final decision for classification.

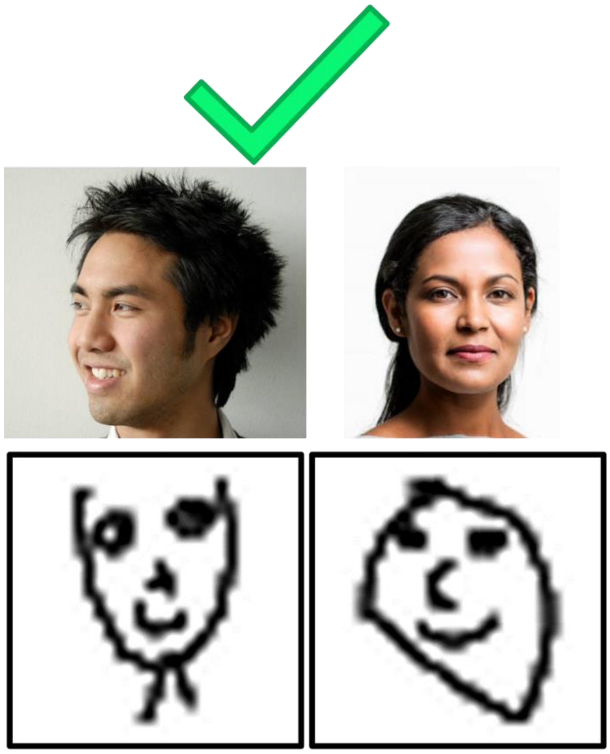


Discriminator



3. Image Pre-processing

Before feeding the data into the model, we filter out real images with complex backgrounds and defective cartoon images to ensure that the generated cartoon characters have complete facial features.



4. Training & Testing Part

The Discriminator loss is caculated by mean square error and the Generator loss is caculated by L1 loss.

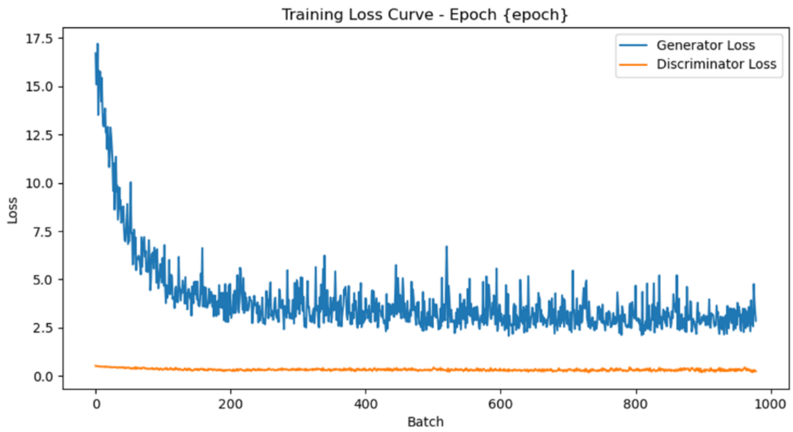
The Discriminator loss is both oscillating from the begin to the end, it converges fast.
In the loss curve of epoch 0, we can observe that the Generator loss in the batch before 200 is decrease drastically and oscillate in the latter batches.
In the loss curve of epoch 1, it's the same situation in the latter batches of epoch 0.

The reason why we choose training two epochs for early stopping condition is because of the generated image after two epochs training has the best visualize performance.

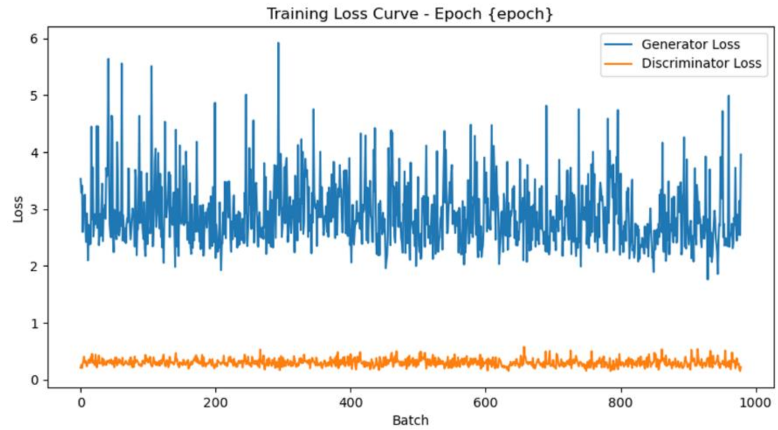
In Testing Part, we load the path file from the training part and use the Generator model to generate the cartoon-like images.

The data for training, validation, and testing are as shown in the following table: The training set consists of 749 real images and 979 cartoon images. The validation set includes 103 real images and 198 cartoon images. Finally, the testing set comprises 200 real images and 150 cartoon images.

Epoch 0



Epoch 1



	real image	cartoon image
Training Data	704	979
Validation Data	103	198
Testing Data	200	150