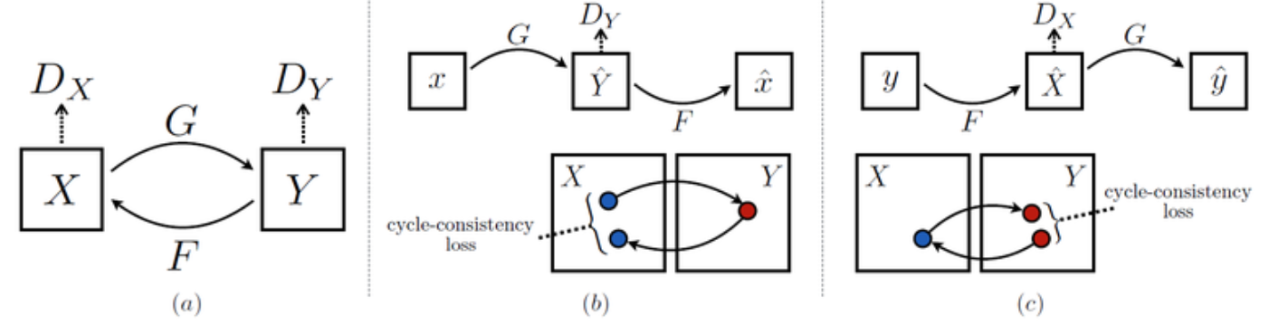


The principle of common GAN is to generate images of random noise z through generator , and the generated images are denoted as . Discriminator is responsible for determining whether an image is real or not, and for binary classification of image x and . In the process of training, generator aims to generate as many images as possible to deceive discriminator . Discriminator D aims to distinguish real images and generated image. The whole process of GAN could be considered as a game theory between generator and discriminator, and at the end we would find an equilibrium point as our final network.

Suppose the distribution of real picture data is and the distribution of noise z is . Therefore, according to the cross-entropy loss, we can construct the following loss function

CycleGAN is an improvement of ordinary GAN. CycleGAN can generate one kind of picture into another kind of picture. The structure of CycleGAN is shown in the figure below:



CycleGAN aims to learn the mapping from dominant to dominant Let the mapping from to be , the first generator in CycleGAN. The Generator G map x in picture to the picture , and then use the first discriminator to identify it. Hence, we could get the first loss function:

Similarly, we can obtain a mapping from to X as the second generator. The Generator maps picture in Y to the picture we also will use the second discriminator to identify it. Therefore, we could get the second loss function:

At the same time, there are also two Generator loss functions in CycleGAN. After the combination, the following loss functions can be obtained:

Finally, three loss functions are merged to form the final loss function

Since CycleGan does not need a one-to-one corresponding sample, we can use loss function to train a network that could transfer one category image to another category image.

Following results shows changes of pictures from CUHKsz style to German style:









