A4 Writeup Question

Kaixuan Hu

March 2018

1 Deep Convolutional GAN

Discriminator 1. **Padding:** Given the input dimension $W \times H \times D$, the output dimension $W' \times H' \times D'$. The relationship between kernel size K, stride S and the padding P is given by:

$$W' = \frac{W - K - 2P}{S} + 1$$

$$H' = \frac{H - K - 2P}{S} + 1$$

Given $K = 4$, $S = 2$, and the relationship $W = 2 \times W'$, $H = 2 \times H'$, we have:

$$\frac{W}{2} = \frac{W - 4 - 2P}{2} + 1$$

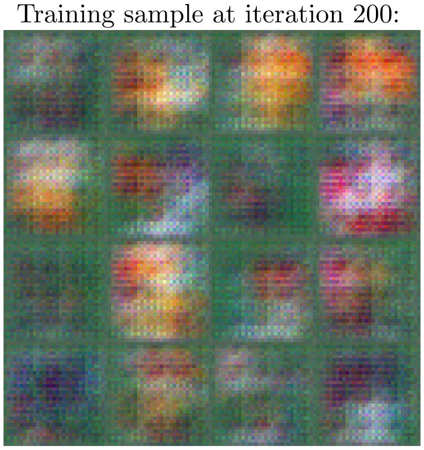
$$\frac{H}{2} = \frac{H - 4 - 2P}{2} + 1$$

We can get:

$$P = 1$$

Experiment 1.

Training sample at iteration 200:



Training sample at iteration 5000:

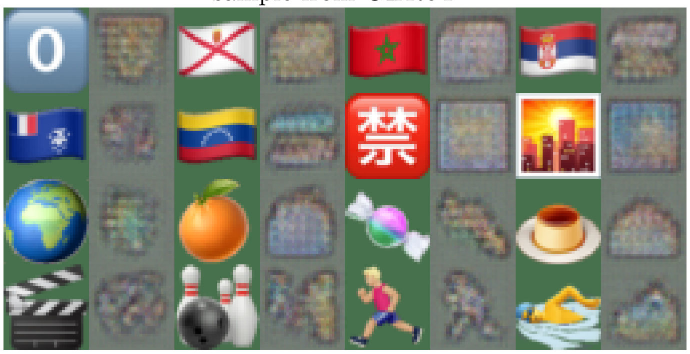
Comment: the quality of the sample has a great improvement as the training goes from iteration 200 to iteration 5000, in the way that the figure of the emoji becomes clearer and more distinguishable, while the noise of the image becomes less.

2 CycleGAN

CycleGAN Experiments

1. The samples from both generators at iteration 600: (train the CycleGAN without the cycle-consistency loss from scratch)

sample from G_XtoY

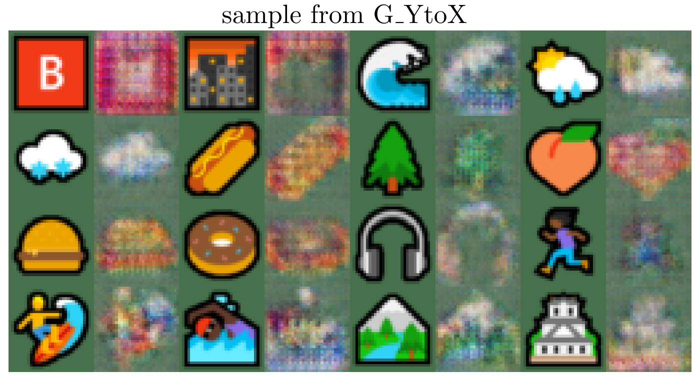


sample from G_YtoX

2. The samples from both generators at iteration 600: (train the CycleGAN with the cycle-consistency loss from scratch)

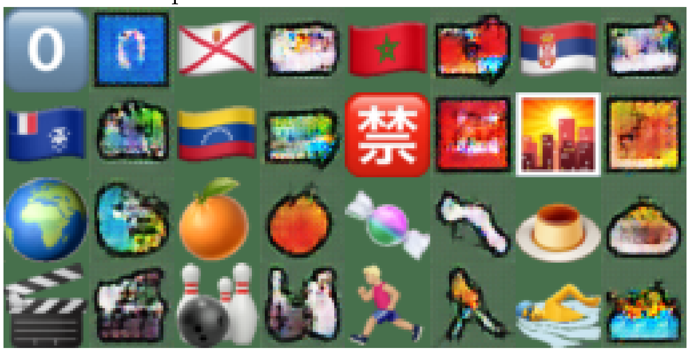
sample from G_XtoY

sample from G_YtoX

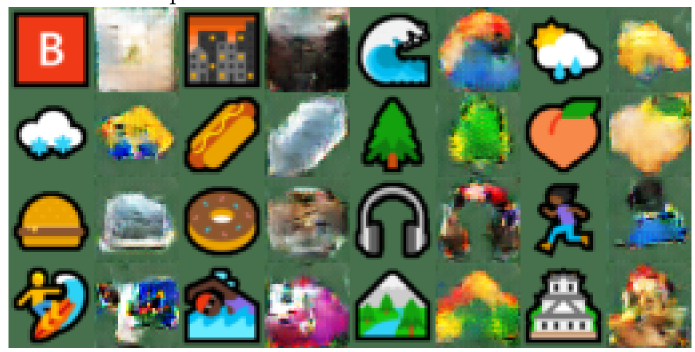


3. The samples from both generators at iteration 20 and 100: (train the pre-trained CycleGAN without the cycle-consistency loss)

sample from G_XtoY at iteration 20



sample from G_YtoX at iteration 20



sample from G_XtoY at iteration 100

sample from G_YtoX at iteration 100

4. The final samples output from both generators at iteration 100: (train the pre-trained CycleGAN with the cycle-consistency loss)

sample from G_XtoY at iteration 100

sample from G_YtoX at iteration 100

5. I do notice that there are differences between the results with and without the cycle consistency loss. The observations are that in the results with the cycle consistency loss, the generated emoji looks better than the results without the cycle consistency loss in terms of:

1) the colours have a better match to the source image, while in the one without cycle consistency loss, some colour of the emoji does not match the original image.

2) the overall shape of the generated emoji with the cycle consistency loss match the source image while the one without cycle consistency loss have some examples of 'broken shape' generated results, even after a large number of training iteration. (notice that in experiment 3, there are some results that don't match or even similar to the original one at all, after 40,000 iterations of training).

3) given the same number of training iteration, the model with the cycle consistency loss generates emoji that contains more details then the model without the cycle consistency loss, especially for the first 2 experiments where we do the training from scratch. (Converge faster)

Reasons on why there is a difference between the two are that: by using the cycle consistency loss in our model, we are given a measure to determine whether a particular translation is good or not good, if a particular translation is bad, then translate the generated image back to the original one will yield a relatively high cost (which is the most direct and intuitive way to let the model know how it is doing), and by minimizing the cost during the training, we are able to get a better result faster and more actuate than the model without using cycle consistency loss.

Also, by using the cycle consistency loss, we give our model an additional way to determine if it is on the right track to translate the image, which lower the chance of getting some 'broken shape' translation for which the model think it is reasonable, but actually not.