CSC321: Assignment #4

Xiangyu Kong
kongxi16

March 28, 2018

Problem 1

1. **Implement the Discriminator of the DCGAN**

   (a) Let P = number of padding, W = input width, K = kernel size, S = stride, O = output width

   According to the structure of the convolutional network,

   $$O = \frac{W}{2}$$

   Using the formula for calculating output dimension

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

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

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

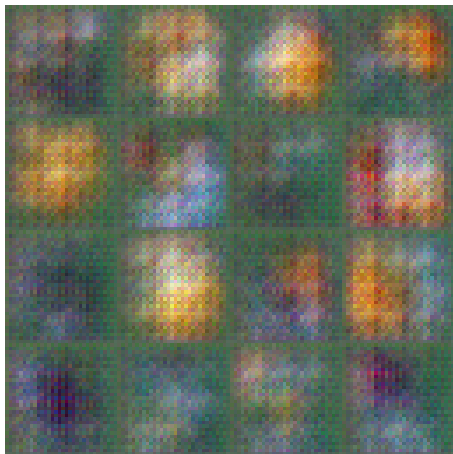 $$= \frac{W - 2 + 4 - W}{2}$$

   $$= 1$$

   (b) See implementation in models.py

2. **Generator**

   See implementation in models.py

3. **Experiment**

   Samples from iterations 200 and 5000 are shown in Fig1. The sample improves in their resolution and quality. We can clearly see what each emoji is (human figure or objects) after a very long period of iterations.

(a) Sample from iteration 200   (b) Sample from iteration 5000
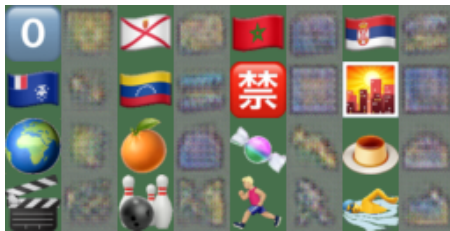
Figure 1

Problem 2

1. **Generator**
   See implementation in model.py

2. **CycleGAN Training Loop**
   See implementation in cycle_gan.py

3. **CycleGAN Experiments**

   (a) The samples without cycle-consistency loss at iteration 400 are shown in Fig2
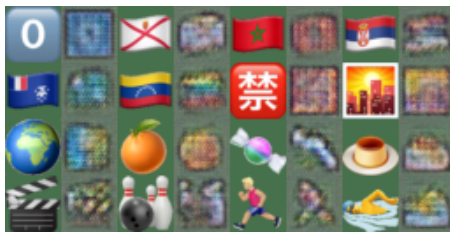

   (a) Sample from iteration 400 X-Y   (b) Sample from iteration 400 Y-X

   Figure 2

   (b) The samples with cycle-consistency loss at iteration 400 are shown in Fig3

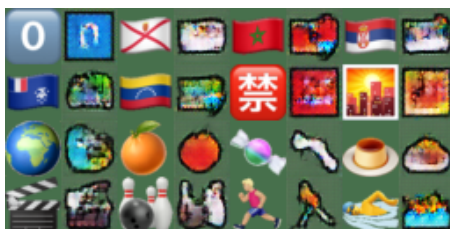

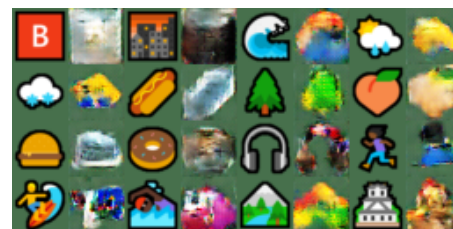   (a) Sample from iteration 400 X-Y   (b) Sample from iteration 400 Y-X

   Figure 3

   (c) The samples from the pre-trained network without cycle-consistency loss are shown in Fig4

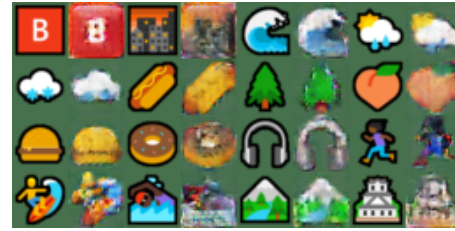   (a) Sample from iteration 100 X-Y   (b) Sample from iteration 100 Y-X

   Figure 4

(a) Sample from iteration 100 X-Y

(b) Sample from iteration 100 Y-X

Figure 5

(d) The samples from the pre-trained network with cycle-consistency loss are shown in Fig5

(e) By observing the samples of the same iteration but different loss (with and without cycle consistency loss), it is obvious that the models using cycle consistency loss generate similar and clearer emojis. This is because by using the cycle consistency loss, we make the network account for the reconstruction loss. Then the generators will try to generate emojis with style more similar to the target. Thus the samples will get a better resemblance and become clearer.