Cycle-Consistent Adversarial Networks

Project Report
BY

Name of the Student	ID Number
Sahdev Nuwal	***2018B3A70900G***
Varun Banker	***2018B2A70295G***
Anurag Nagpal	***2018B1A70939G***

Deep Learning CS F425
IC: Prof. Tirtharaj Dash

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI

Spring semester 2021

Contents

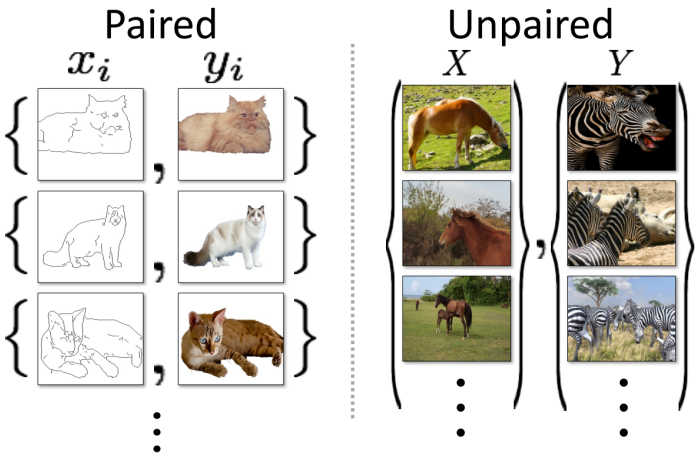
Introduction

Image-to-image translation is a class of vision and graphics problems where the goal is to learn the mapping between an input image and an output image using a training set of aligned image pairs. However, for many tasks, paired training data will not be available. Our project is based on the Paper titled "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks" by Jun-Yan Zhu et al 2017. We tried to replicate the architecture of the paper and test the model on three different datasets - horse2zebra, cityscapes, monet2photo.

Main Text

Understanding Cycle GANS and Architecture

CycleGANS results in unpaired image-to-image translation. This means that in order to train it to translate images from domain X to domain Y, we do not have to have exact correspondences between individual images in those domains. The figure below shows Paired training data on the left $\{x_i , y_i\}_{i=1}^{N}$ where the correspondence between x_i and y_i exists and on the right are unpaired images set, consisting of a source set $\{x_i\}_{i=1}^{N}$ ($x_i \in$ X) and a target set $\{y_j\}_{j=1}^{N}$ ($y_j \in$ Y), with no information provided as to which x_i matches which y_j.



We need to map the input image coordinate X to domain coordinated Y and this problem is an image to image translation. Much literature exists trying to address this problem but every single one of these uses paired image data, their models are trained on both the original image and the corresponding acquired image after translation. But creating such a data paired dataset is difficult and time-consuming. Here lies the motivation behind Cycle GANS, which tries to address this problem using unpaired image to image translation. Unpaired data is much easier to gather.

Suppose for example we have a set of photo style images X and we have another set of Monet's style Y, but we don't have access to Monet's painting for every single input sample image we try to learn a mapping G between images X to its corresponding images Y.

$$G: X \rightarrow Y$$

Out goal is to train a model to learn this mapping G.

Since we don't have paired data, we use cycle consistency loss where we learn two mappings F: Y → X and G: X → Y and using that we calculate the loss. F and G are inverse mappings of each other. So we not only need a mapping of G that generates similar distribution but we also need one distribution that is cycle consistent with respect to the inverse mapping F.

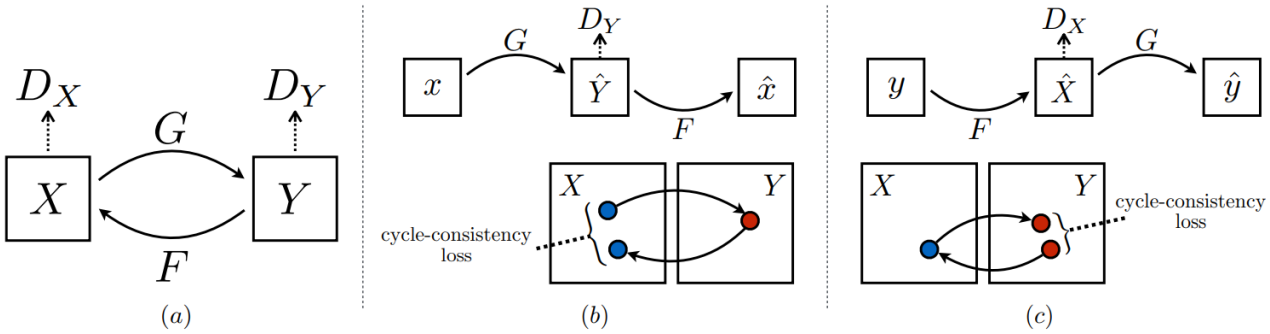
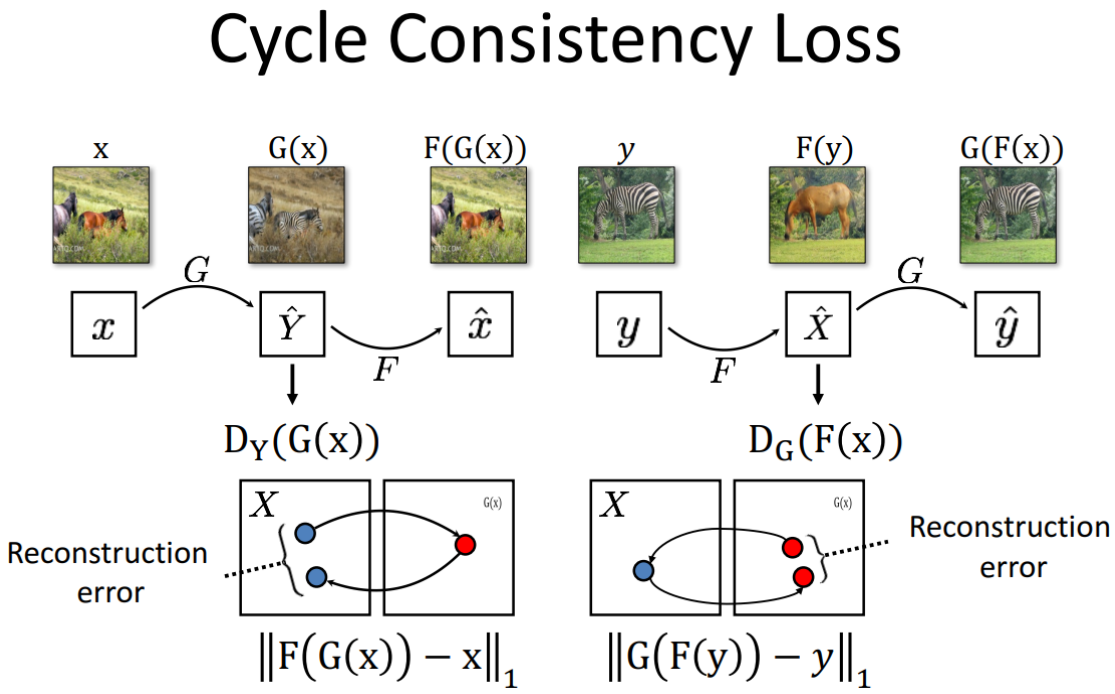


Figure 3: (a) Our model contains two mapping functions $G : X \rightarrow Y$ and $F : Y \rightarrow X$, and associated adversarial discriminators D_Y and D_X. D_Y encourages G to translate X into outputs indistinguishable from domain Y, and vice versa for D_X and F. To further regularize the mappings, we introduce two *cycle consistency losses* that capture the intuition that if we translate from one domain to the other and back again we should arrive at where we started: (b) forward cycle-consistency loss: $x \rightarrow G(x) \rightarrow F(G(x)) \approx x$, and (c) backward cycle-consistency loss: $y \rightarrow F(y) \rightarrow G(F(y)) \approx y$



The Reconstruction error is the cycle consistent loss that needs to be minimized. Two kinds of loss functions are used to learn unpaired image to image transformation.

1) Adversarial loss: results in style transfer between two domains X and Y

Adversarial Loss: change the style

$$\mathcal{L}_{\text{GAN}}(G, D_Y, X, Y) = \mathbb{E}_{y \sim p_{\text{data}}(y)}[\log D_Y(y)] + \mathbb{E}_{x \sim p_{\text{data}}(x)}[\log(1 - D_Y(G(x))]$$

Similarly, for F, we also have $\mathcal{L}_{\text{GAN}}(F, D_X, X, Y)$.

2) Cycle Consistency Loss (Reconstruction Error) Which preserves the content

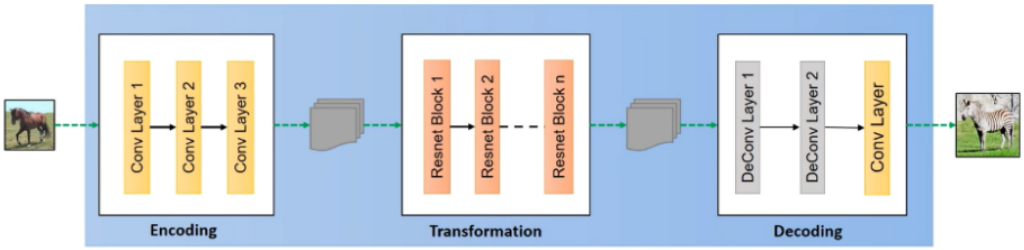
Cycle Consistency Loss: preserve the content

$$\mathcal{L}_{\text{cyc}}(G, F) = \mathbb{E}_{x \sim p_{\text{data}}(x)}[\|F(G(x)) - x\|_1] + \mathbb{E}_{y \sim p_{\text{data}}(y)}[\|G(F(y)) - y\|_1].$$

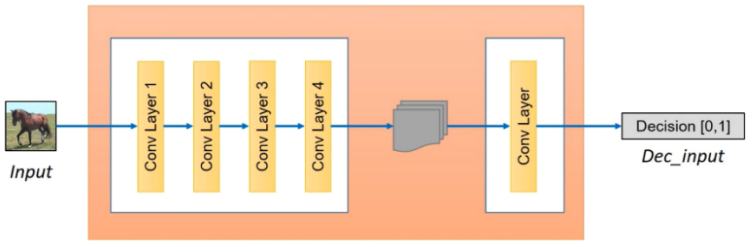
Two empirical assumptions are that style is easy to change and content is easy to keep.

Therefore, the full objective is:

$$\mathcal{L}(G, F, D_X, D_Y) = \mathcal{L}_{\text{GAN}}(G, D_Y, X, Y) + \mathcal{L}_{\text{GAN}}(F, D_X, Y, X) + \lambda \mathcal{L}_{\text{cyc}}(G, F)$$



Generator Architecture



Discriminator Architecture

Algorithm 2 CycleGAN Training Loop Pseudocode

procedure TRAINCYCLEGAN

Draw a minibatch of samples $\{x^{(1)}, \ldots, x^{(m)}\}$ from domain X

Draw a minibatch of samples $\{y^{(1)}, \ldots, y^{(m)}\}$ from domain Y

Compute the discriminator loss on real images:

$$\mathcal{J}^{(D)}_{real} = \frac{1}{m}\sum_{i=1}^{m}(D_X(x^{(i)}) - 1)^2 + \frac{1}{n}\sum_{j=1}^{n}(D_Y(y^{(j)} - 1)^2$$

Compute the discriminator loss on fake images:

$$\mathcal{J}^{(D)}_{fake} = \frac{1}{m}\sum_{i=1}^{m}(D_Y(G_{X\to Y}(x^{(i)})))^2 + \frac{1}{n}\sum_{j=1}^{n}(D_X(G_{Y\to X}(y^{(j)})))^2$$

Update the discriminators

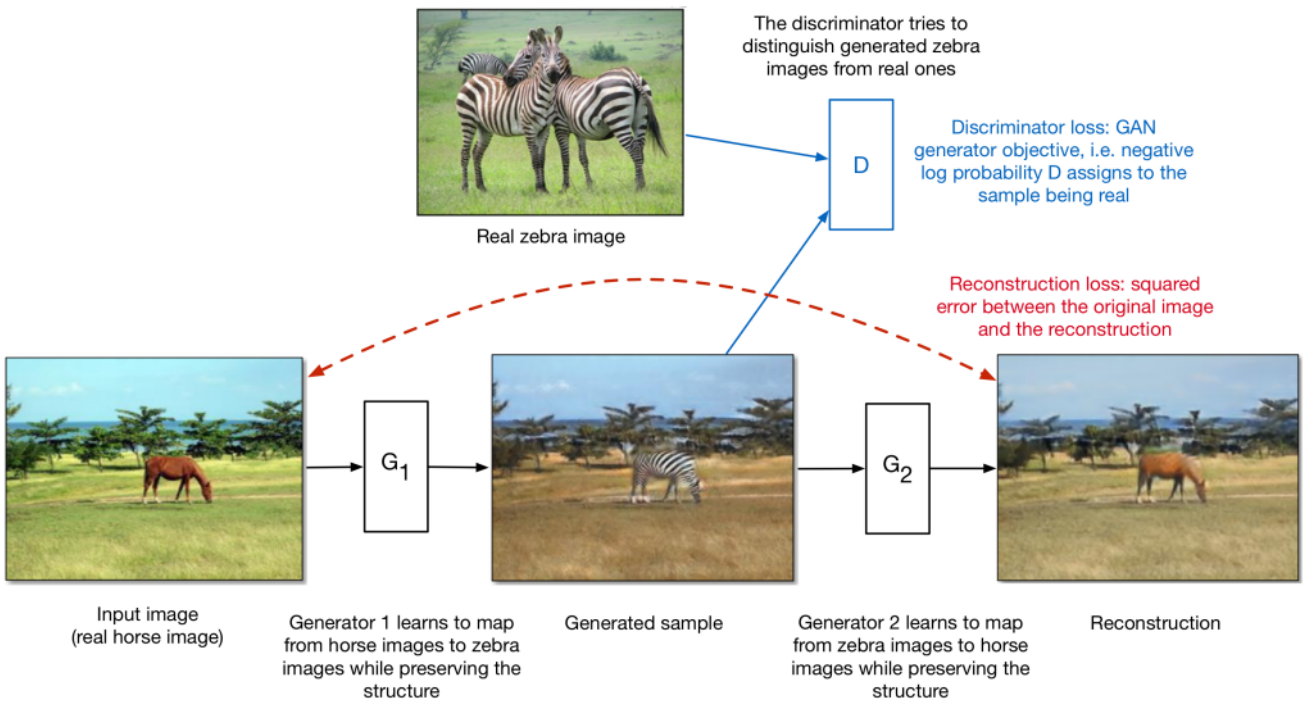
Compute the $Y \to X$ generator loss:

$$\mathcal{J}^{(G_{Y\to X})} = \frac{1}{n}\sum_{j=1}^{n}(D_X(G_{Y\to X}(y^{(j)})) - 1)^2 + \mathcal{J}^{(Y\to X\to Y)}_{cycle}$$

Compute the $X \to Y$ generator loss:

$$\mathcal{J}^{(G_{X\to Y})} = \frac{1}{m}\sum_{i=1}^{m}(D_Y(G_{X\to Y}(x^{(i)})) - 1)^2 + \mathcal{J}^{(X\to Y\to X)}_{cycle}$$

Update the generators



Datasets

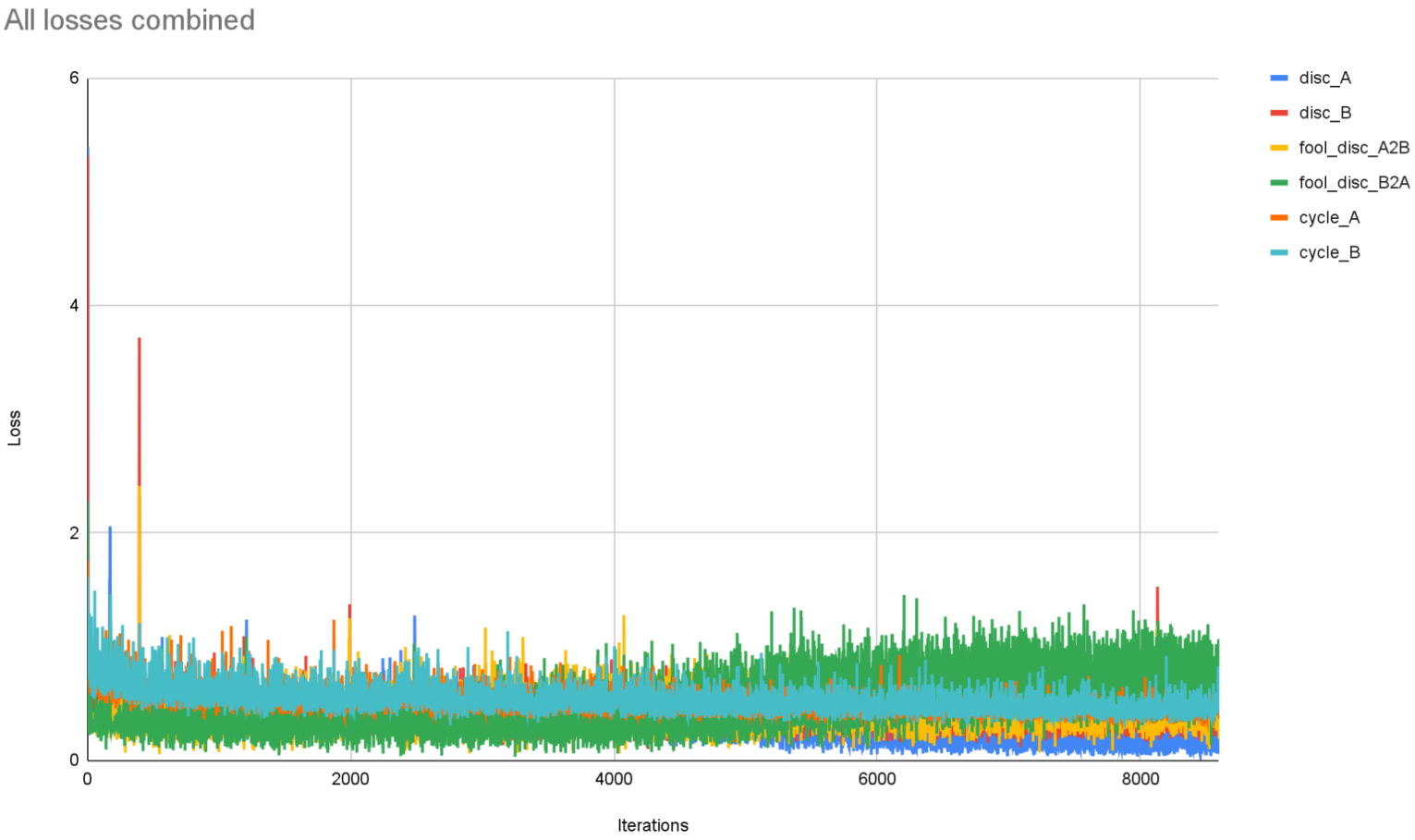
We used three different datasets from those used in the paper and one new but similar dataset. The three data sets are [horse2zebra, cityscapes, and monet2photos](#).

Training and Epochs

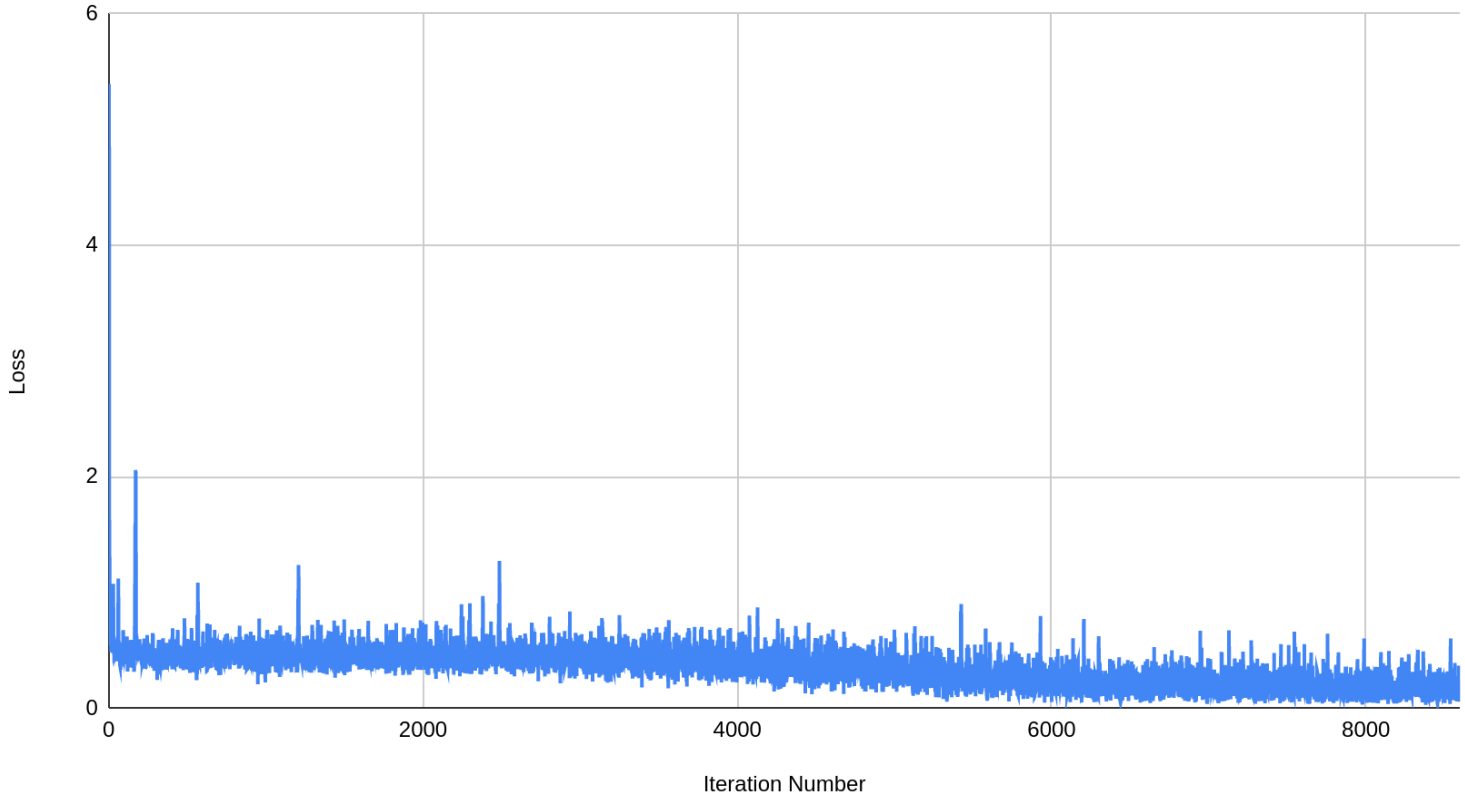
Total number of epochs = 29
Number of Iterations per epoch = 214
Adam Optimizer was used
Add screenshots of last training epochs

```
Debug Output:
Iteration: 720
Disc A Loss: 0.06755250692367554
Disc B Loss: 0.3581489622592926
Fool Disc Loss A2B: 0.28648442029953003
Fool Disc Loss B2A: 0.8311122059822083
Cycle Loss A: 0.4563145041465759
Cycle Loss B: 0.43075913190841675


Debug Output:
Iteration: 721
Disc A Loss: 0.08424031734466553
Disc B Loss: 0.3619263768196106
Fool Disc Loss A2B: 0.5194675326347351
Fool Disc Loss B2A: 1.0486863851547241
Cycle Loss A: 0.4273029565811157
Cycle Loss B: 0.36301401257514954


Debug Output:
Iteration: 722
Disc A Loss: 0.16360032558441162
Disc B Loss: 0.3867865800857544
Fool Disc Loss A2B: 0.5702834725379944
Fool Disc Loss B2A: 0.7987803816795349
Cycle Loss A: 0.37391197681427
Cycle Loss B: 0.3457791209220886


Debug Output:
Iteration: 723
Disc A Loss: 0.12710599601268768
Disc B Loss: 0.2555253505706787
Fool Disc Loss A2B: 0.49111172556877136
Fool Disc Loss B2A: 0.4880658686161041
Cycle Loss A: 0.3765408396720886
Cycle Loss B: 0.3894083499908447


Debug Output:
Iteration: 724
Disc A Loss: 0.21715083718299866
Disc B Loss: 0.27839767932891846
Fool Disc Loss A2B: 0.495213121175766
Fool Disc Loss B2A: 0.8631212711334229
Cycle Loss A: 0.3724614083766937
Cycle Loss B: 0.4944200813770294
```

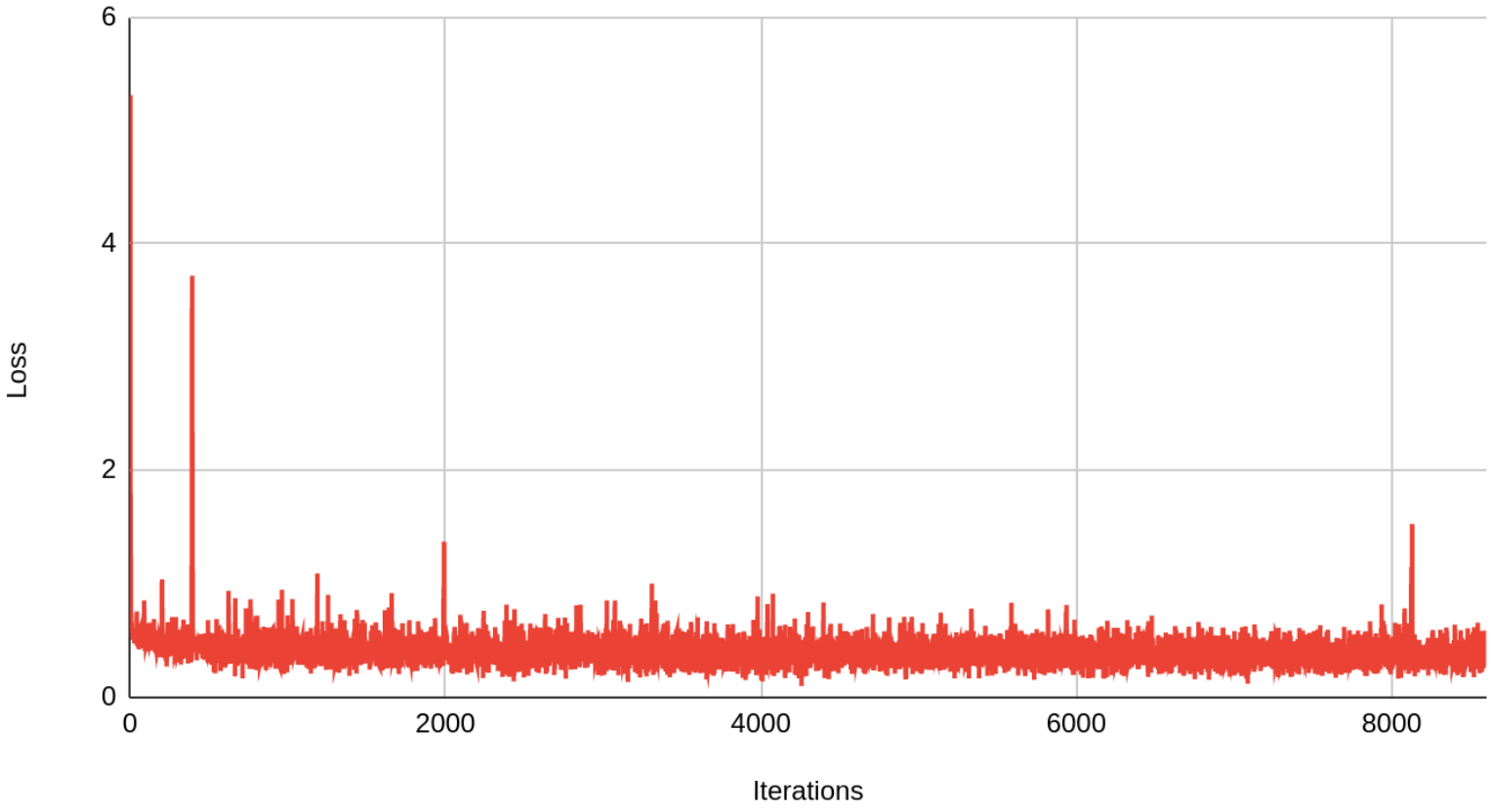
Plots


All losses combined
6
4
2
0
Loss
0
2000
4000
6000
8000
Iterations
disc_A
disc_B
fool_disc_A2B
fool_disc_B2A
cycle_A
cycle_B

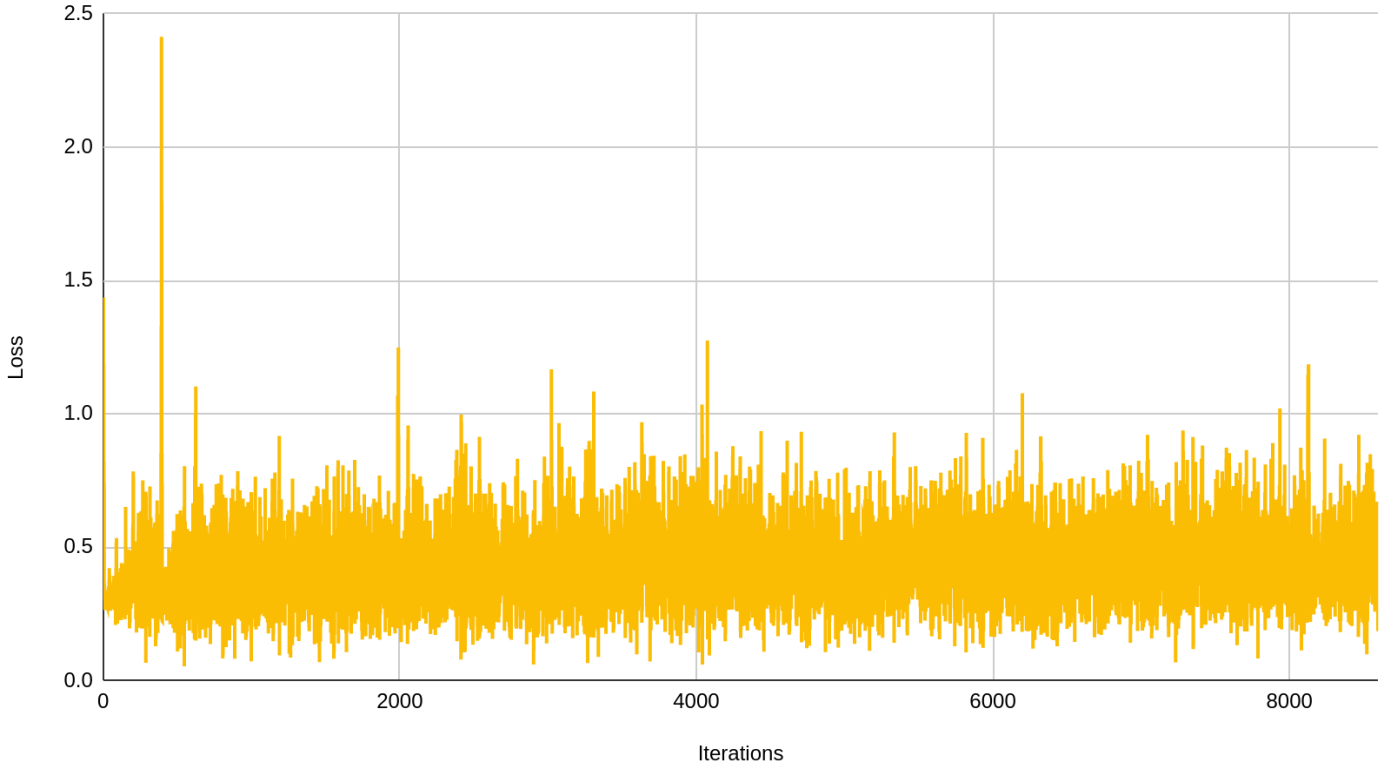
Disc_A


6
4
2
0
Loss
0
2000
4000
6000
8000
Iteration Number

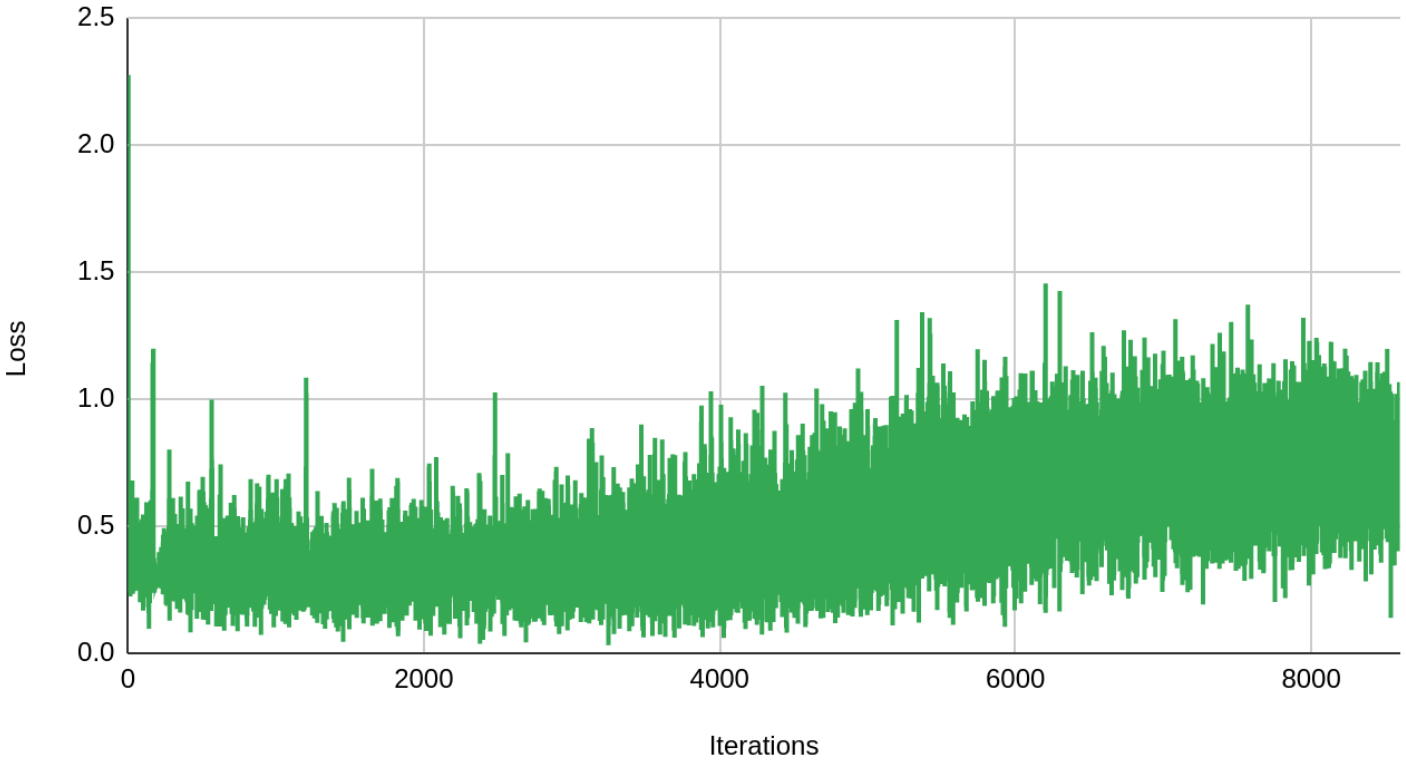
Disc_B


6
4
2
0
Loss
0
2000
4000
6000
8000
Iterations

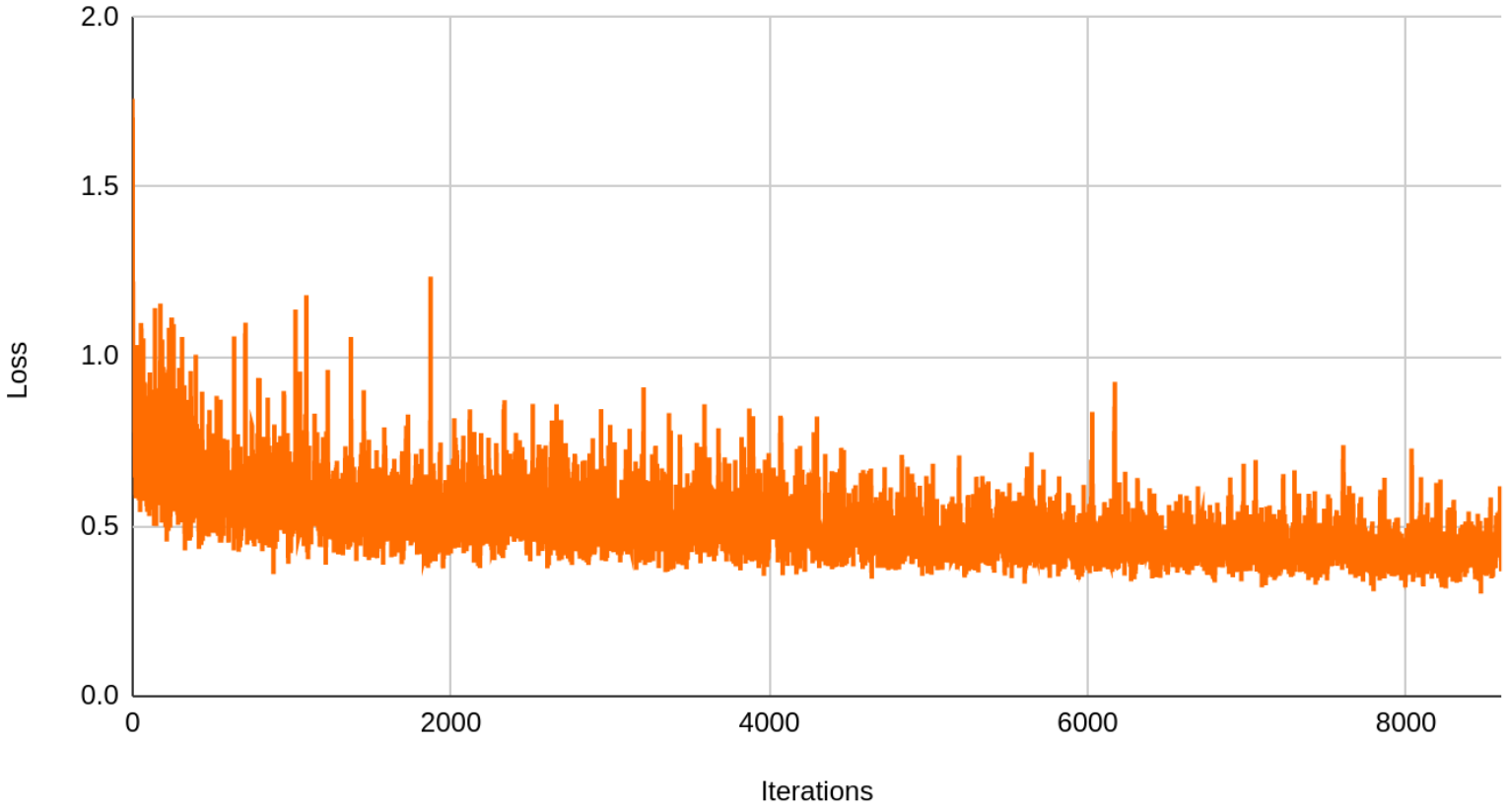
Fool_disc_A2B



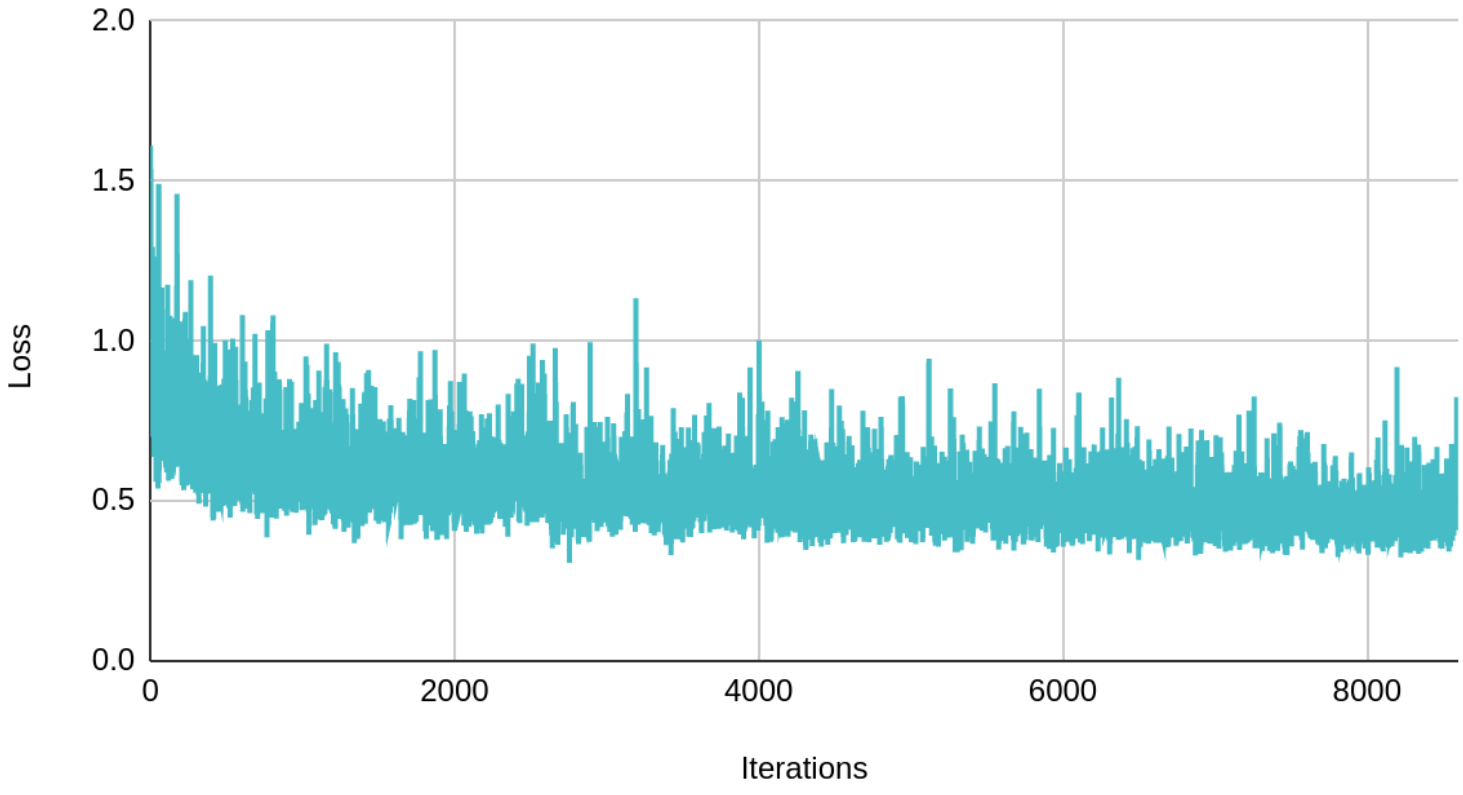
Fool_disc_B2A



Cycle_A



Cycle_B



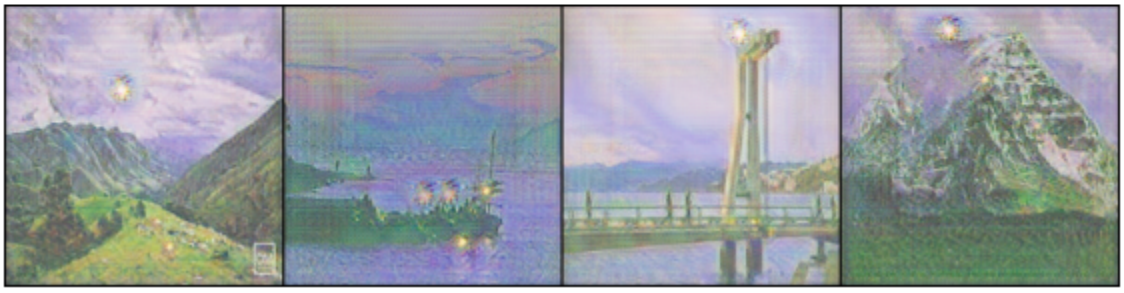
Results

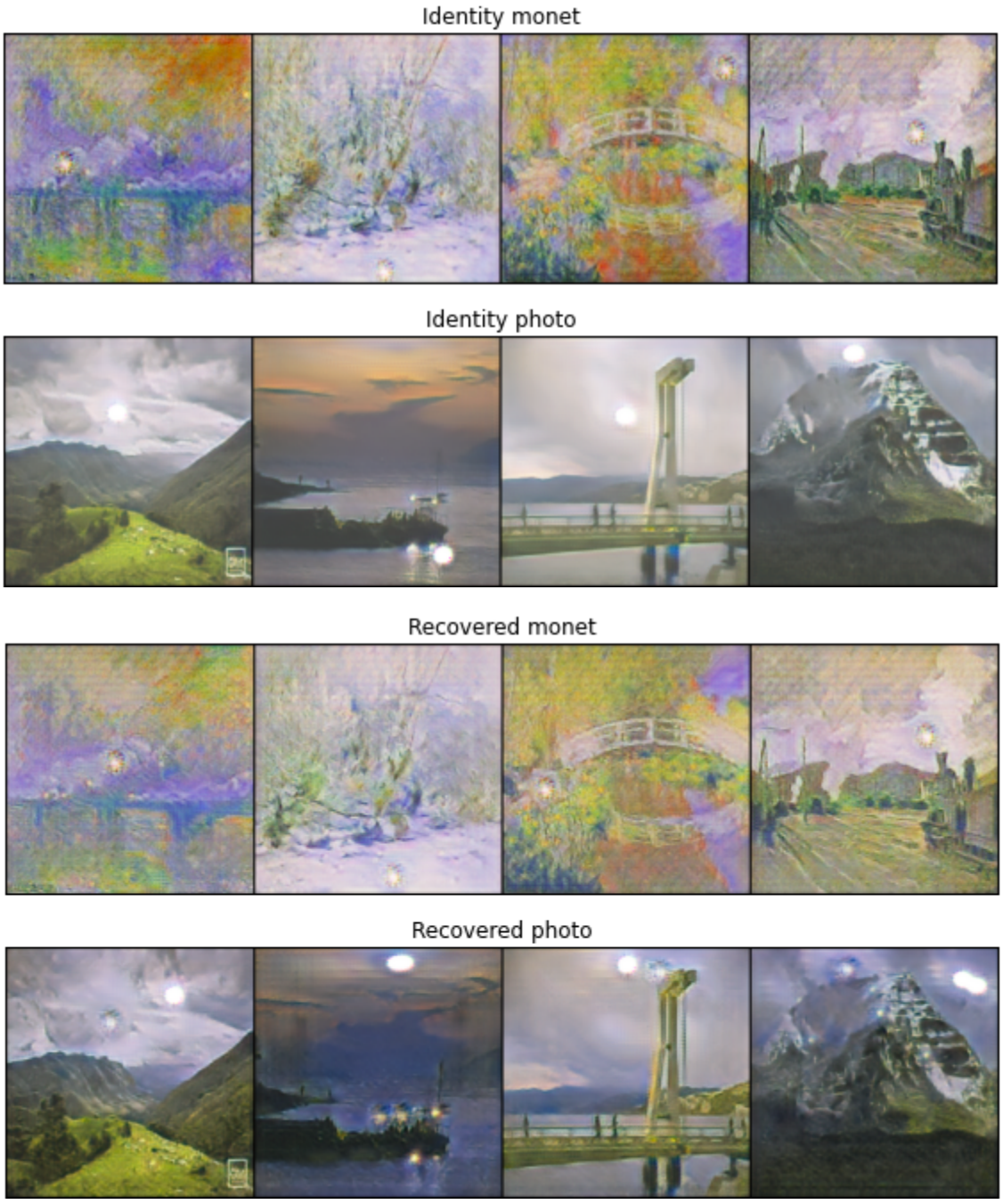
Real monet

Fake photo

Real photo

Fake monet





Conclusion and contributions

In conclusion we learned that Cycle Consistency Adversarial Networks tries to solve image-image translation problems without paired data. The objective function it uses is the Adversarial losses and Cycle consistency losses. Image-Image translation is used in Object Transfiguration, Photo Enhancement, Style Transformation, Season Transformation and Image Segmentation. We made the model and tested it on a new city dataset and performed image segmentation.

References

1) Zhu, Jun-Yan, et al. "Unpaired image-to-image translation using cycle-consistent adversarial networks." *Proceedings of the IEEE international conference on computer vision*. 2017.
2) Jun-Yan Zhu and Taesung Park Joint work with Phillip Isola and Alexei A. Efros
3) CSC321 Lecture 19: Generative Adversarial Networks
4) CycleGAN Introduction Important Note