CIS 680 Bicycle GAN Final Report

Harsh Goel, Nicholas Marion, Aditya Singh

**Abstract**

This document outlines the design and results of the Bicycle GAN project our group implemented for our final project. This network utilizes properties of both VAEs as well as GANs to improve results when compared with traditional Generative Adversarial Networks. In utilizing the abilities of VAEs to reconstruct a wide set of images from the latent space, and leveraging the image quality that GANs are capable of generating, Bicycle GAN is able to generate a diverse set of realistic image results, utilizing a single image and noise vectors for inputs. For our application, we chose to utilize the edges2shoes dataset (Isola et al) which provides edge maps paired with shoe images.. This network utilizes two separate discriminators called cVAE-GAN and cLR-GAN which each work to improve the quality of the image reconstruction, as well as an encoded noise vector to encourage diversity. Furthemore, our team performed both qualitative and quantitative evaluations of the network results, including utilizing FID and LPIPS scores in order to rate the network over the edge2shoe dataset. These results as well as example output images are exhibited later in the document as we explain the subtleties of our network implementation. Additionally, we test our network over the night2day dataset (Laffont et al) which provides an evening image paired with daytime images and the maps dataset(Isola et al) which provides matching google maps images with satellite data.

**Problem Description**

The issue with many GAN implementations is that given an input, there is only one possible output that the network generates. In some cases, this can lead to a lack of diversity in respect to potential outputs from the GAN. For instance, the pix2pix model (Isola et al.) is able to generate exceptional image reconstructions, however every input image pairs with a single output. In addition, the pix2pix paper specifically mentions diminished color response between the ground truth and the generated outputs. Bicycle GAN can remedy some of these issues as it creates a larger range of possible outcomes in terms of both color as well as texturing for each image. Additionally, according to the original bicycle GAN paper(Zhu et al.), because bicycle GAN utilizes cVAE-GAN and KL divergence loss, it is able to achieve a more realistic result than pix2pix. Furthermore, while cLR-GAN may create less realistic images by itself due to mode collapse(Zhu et al.), its implementation in bicycle GAN avoids this as the network is set up to improve the diversity of the outputs, thus improving cLR-GAN’s accuracy.

**Literature Review**

One very similar network to Bicycle GAN is Cycle GAN (Zhu et al), which was implemented by the creator of bicycle GAN. The loss function for cycle GAN has multiple similarities to Bicycle GAN. It utilizes adversarial loss and has two discriminators as does Bicycle GAN. Additionally, Cycle GAN uses cycle consistency loss which essentially checks for the pixel wise difference from the input image and a reconstructed version of the input image generated from the outputs of the network. This type of loss guides the generator to reconstruct an image as close to the original as possible. Therefore, making it possible to generate a near copy of the input image given only the output of the network. Because of this desired feature, Cycle GAN has two discriminators, one to generate the output image, and one to restore the input image, this differs from Bicycle GAN which cannot restore an image once converted by the generator.

Another type of GAN called Deep Convolutional GANs (DCGANs) seek to solve the issue of unsupervised learning with GANs(Radford et al). The DCGAN paper outlined that the discriminator is able to effectively learn different high level features in the image, for example in a picture of a room, objects such as beds and windows are activated as feature, however, sometimes objects were reported missing from the generated outputs and less common objects were often not generated. One other issue they mention in the paper is mode collapse when training over long periods of time. The loss functions in Bicycle GAN seek to reduce this issue, however it is difficult to fully prevent it from occurring.

**Formal Description of Proposed Methodology**

For this project we implemented Bicycle GAN as described by the paper “Multimodel Image-to-Image Translation by Enforcing Bi-Cycle ConsistencyZhu et al. This network consists of an encoder, generator and two discriminators. We constructed the generator following the UNET architecture (Ronneberger et al) which convolves an input image through six sets of convolutions, before using transposed convolutions to restore the image to a specified size. The “U” shape of the architecture is set up to concatenate each same sized layer of convolved images, with its corresponding matching transposed convolved image layer.

The discriminators for the network were set up to utilize the Patch GAN architecture(Li et al), which in our implementation consisted of five convolutions each followed by a 2d instance norm and leaky Relu activation function. The encoder network utilizes a Resnet 18 backbone as well as leaky Relu activation and two separate linear layers for calculating the mu and log variance values. For this project, we experimented with 3 different discriminator architectures. This first network utilized a single shared discriminator for both the cVAE and cLR network. Next, we tried using two separate discriminators one for cVAE and another for cLR. Lastly, we tried implementing multipatch discriminators with the two discriminator setup.

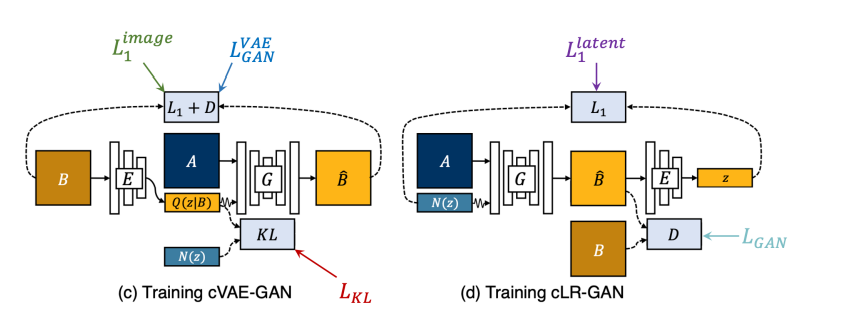
The generator loss consists of multiple distinct loss terms added together. These included l1 loss between the output of the generator and the real image. Next, KL divergence loss was calculated using the mu and log variance taken from the output of the encoder given the real image. The first adversarial loss term is calculated as the mse loss between the valid(equal to 1) input and the output of the cVAE GAN discriminator given an input of the generator. That generator used an input of the edge image and reparameterized encoder output of the real image. The second adversarial loss term is calculated as the mse loss between the valid input and the output of the cLR GAN discriminator given an input of the generator. That generator used an input of the edge image and a random z vector. Lastly another l1 loss is applied using the random z vector and the generated z vector from the encoder.Both discriminators follow similar loss functions to those found in PatchGAN. The main difference in the inputs between the cLR and cVAE loss functions is that the cVAE loss uses the reparametrized output from the encoder as the z variable in its generator input, while the cLR loss uses a random z variable for the generator input. Lastly, encoder loss uses the same loss functions as the generator loss, as well as the l1 loss across z and kl divergence. A diagram showing the cVAE and cLR is shown below.

Figure 1: Comparison of cVAE and cLR GANs and where their loss data is derived.

**Summary of Experiments Performed**

We trained our network primarily on the edge 2shoe dataset. In addition to the shoe dataset, we also tested our network on the night2day dataset and the google maps to satellite dataset. Given more training time and some network tuning, it's possible that the network would have generated more realistic results. Additionally, we considered adding dropout to the network in an effort to further improve these results. However using the current code, we couldn’t generate believable images on the test set.

**Training Visualizations**

We demonstrate the visualizations of our training for the model with 2 single patch discriminators.

**Generator Plots edges2shoes over 20 epochs on the 2 discriminator network**

Our model initially trains well, suffers from mode collapse at around epoch 18 and recovers from the mode collapse when 20 epochs are completed.

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Fig: Visualization of Training from epoch 1 top left to epoch 20 bottom right

**Loss Plots edges2shoes over 20 epochs on the 2 discriminator network**

|  |  |  |  |
| --- | --- | --- | --- |

Fig: Loss Plots (Generator loss, Encoder Loss, DcVAE Loss, DcLR Loss

**Results**

In order to test different network setups for the discriminator, we evaluated three different networks. Overall our team was satisfied with the results we obtained. We were able to obtain satisfactory reconstructions spanning a wide range of output possibilities. The results of our FID and LPIPS score calculations are shown below. This first network utilized a single shared discriminator for both the cVAE and cLR network. This network provided the highest LPIPS score, as well as the highest FID score. The high LPIPS score indicated to us that it had the highest variation given varying noise vector values but poor image quality.e tried implementing multipatch discriminators with the two discriminator setup. This resulted in a FID that was between the single and double discriminator implementations and our lowest LPIPS score. The model with 2 single patch discriminators for both CLR and CVAE Gans produced the best performance in terms of FID scores while giving a reasonable multi-modal in terms of LPIPS score distribution**. The Project Video Link can be found at** for the visualisations voer the different datasets: [**https://drive.google.com/file/d/1C2HJ7rebTL3pMM9YtfxV04gmxRBZH1Wg/view?usp=sharing**](https://drive.google.com/file/d/1C2HJ7rebTL3pMM9YtfxV04gmxRBZH1Wg/view?usp=sharing)

| Edges2shoes Dataset Results | Shared Discriminator | Two Discriminators | Two Discriminators with Multi Patch |
| --- | --- | --- | --- |
| FID Score Between two real sets of images | 58.97 | | |
| FID Score Between Image set 1 and generated images | 191.29 | 148.99 | 175.44 |
| FID Score Between Image set 2 and generated images | 196.08 | 156.97 | 174.91 |

| LPIPS Score (edges2shoes dataset) | | |
| --- | --- | --- |
| Shared Discriminator | Two Discriminators | Two Discriminators with Multi Patch |
| 0.241 | 0.219 | 0.212 |

**Discussions**

Given more time, we would have liked to test our model on additional datasets to explore additional possibilities for the network. Additionally, the Bicycle GAN paper also discusses how changes in where the random z variable is injected into the forward pass of the generator modifies the generator output. Given additional time, we would have investigated this further, trying some additional implementations to improve the diversity of our results. Moreover, we would also like to investigate why our double patch discriminator performs poorly as compared to the single patch gan discriminator. Lastly, we would have attempted modifying our loss function, perhaps attempting to merge one of the loss functions from another GAN architecture into our own to further increase the realism of our generated images. Additionally testing changes to some of the hyperparameters of the network as well as other implementation details could have potentially increased the realism shown in our results.

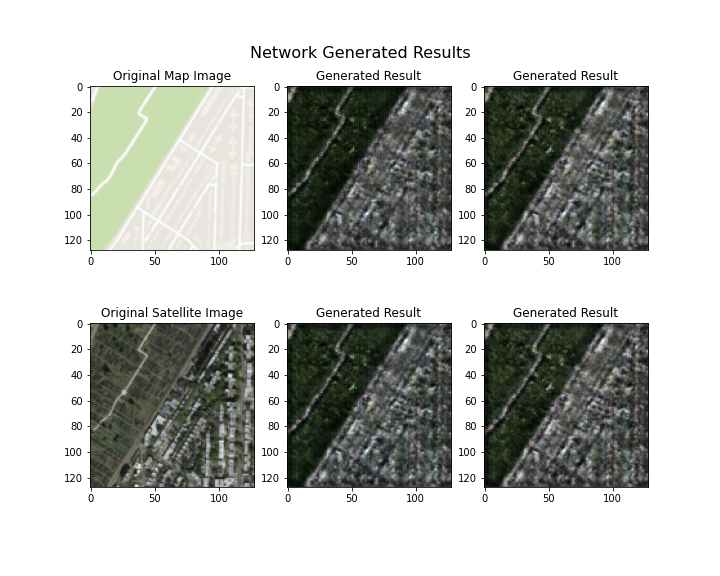
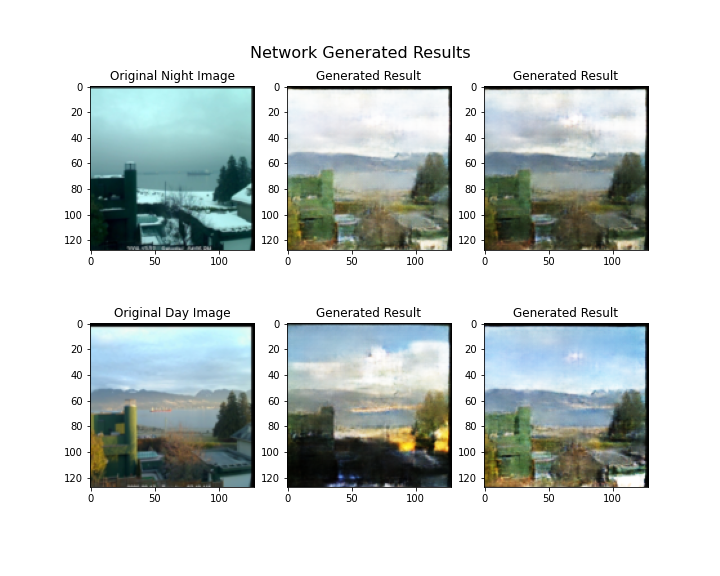
**Image Results : Edge2Shoe**

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |

Fig: Visualization of Different Input Edges and generated outputs

**Maps and Night2Day**

We also trained our network over the Maps and Night2Day dataset and produced the following visualization.



**References:**

Bansal, Aayush, et al. "Recycle-gan: Unsupervised video retargeting." Proceedings of the European conference on computer vision (ECCV). 2018.

Isola, Phillip, et al. "Image-to-image translation with conditional adversarial networks." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017.

Kim, Taeksoo, et al. "Learning to discover cross-domain relations with generative adversarial networks." International Conference on Machine Learning. PMLR, 2017.

Laffont, Pierre-Yves, et al. "Transient attributes for high-level understanding and editing of outdoor scenes." *ACM Transactions on graphics (TOG)* 33.4 (2014): 1-11.

Li, Chuan, and Michael Wand. "Precomputed real-time texture synthesis with markovian generative adversarial networks." *European conference on computer vision*. Springer, Cham, 2016.

Mao, Qi, et al. "Mode seeking generative adversarial networks for diverse image synthesis." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019.

Radford, Alec, Luke Metz, and Soumith Chintala. "Unsupervised representation learning with deep convolutional generative adversarial networks." *arXiv preprint arXiv:1511.06434* (2015).

Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation." *International Conference on Medical image computing and computer-assisted intervention*. Springer, Cham, 2015.

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.

Zhu, Jun-Yan, et al. "Multimodal Image-to-Image Translation by Enforcing Bi-Cycle Consistency." *Advances in neural information processing systems*. 2017.