Generative Adversarial Network Application in Pathological Image Generation

Tingting Wang
Supervised By: Prof. David Doermann
Department of Computer Science and Engineering
University at Buffalo
twang49@buffalo.edu

Abstract

Generative Adversarial Network has gained lots of attention in different research areas since 2014 when Ian Goodfellow[1] proposed the paper about GAN; A lot of researchers have applied GAN in different domains, such as image generation, data segmentation, object detection and so on. It's time-consuming and expert-needed to get numerous paired pathological images with high quality, which may contribute in development of medical image research. So this independent study mainly pay attention to paired pathological image generation.

1. Introduction

Generative Adversarial Network, a unusual kind of deep learning network model, always trained two network at the same time, with one called Generator, another one called Discriminator. Generator mainly focused on generating high quality image, and Discriminator, which is different from Generator, majorly pay attention to discrimination of generated images. This adversarial training scheme will lead to that Generator and Discriminator can learn important information from each other. And this scheme has reached state-of-art performance in a wide range of tasks, such as generation, classification, segmentation and detection. Due to the success of GAN in those areas, we applied GAN on pathological image generation, which is a hard and useful problem.

In summary, our main contributions in this independent study are:

- We made use of limited size of dataset to train GAN.
- We compared performance of different strategy on Discriminator's architecture.
- We employed trained network to generate medical image, and compared the quality of generated images and real images.

2. Related Work

The research of application of GAN in medical area are very limited, when compared to image generation in other domains, such as scene, human face and other normal real-life images. But Prof. David Doermann provied us two good and useful paper to start with. I got beneficial and import information from those two paper . They are:.

- An integrated iterative annotation technique for easing neural network training in medical image analysis [2].
- Adversarial U-net with spectral normalization for histopathology image segmentation using synthetic data [3].

2.1. Human-AI-Loop annotation technique

Deep learning network has achieved massive state-of-art performance on medical domains, But almost all neural network study on medical fields mainly depended on volume and quality of paired data : pathological image and its annotation(expert-generated), which is super time-consuming and only specialists can do this kind of annotations. So H-AI-L, human-in-the-loop try to address this hot issue by adopting a pre-trained model: DeepLab v2, which already reached fabulous results on other domains. And in the end, they also show the adaptability of H-AI-L to different type of medical images, like human prostate glands and mouse renal micro compartments.

2.2. Adversarial U-Net with SN for image segmentation

Most current neural network style approaches for medical image segmentation are paired-data-based. As we said before, this kind of data are vary limited. So in this paper, they propose to use cycleGAN[4] to do data augmentation with small amount of paired data and massive unpaired synthetic data before image segmentation. After having numerous paired medical images, they used

an adversarial U-Net network with spectral normalization to train the segmentation network in order to increase the training stability. Finally, they achieved that the total average accuracy for medical image(multi-organ nuclei segmentation) are increased from about 79% to about 94% because of the influence of synthetic data.

3. Approach

In order to address the issue of lack of paired medical image in medical research, this independent study mainly focused on paired pathological image generation. After exploring different network architecture, I decided to use U-Net [5] architecture on Generator like most medical image generation in other paper. For the Discriminator, I used patchGAN[6].

3.1. U-Net Generator

Figure 1 U-Net Generator

As we can it from Figure 1, in the Generator, I adopted a typical U-Net architecture, whose first half part are normal down sample part and second half part are up sample part.

Layer	Output Size	(Kernel, Stride)
Inputs	256 × 256 × 5	(-, -)
Downsample_1	128× 128 × 64	(4, 2)
Downsample_2	64 × 64 × 128	(4, 2)
Downsample_3	32 × 32 × 256	(4, 2)
Downsample_4	16 × 16 × 512	(4, 2)
Downsample_5	8 × 8 × 512	(4, 2)
Downsample_6	4 × 4 × 512	(4, 2)
Downsample_7	2 × 2 × 512	(4, 2)
Downsample_8	1 × 1 × 512	(4, 2)
Upsample_1	2 × 2 × 1024	(4, 2)
Upsample_2	4 × 4 × 1024	(4, 2)
Upsample_3	8 × 8 × 1024	(4, 2)
Upsample_4	16 × 16 × 1024	(4, 2)
Upsample_5	32 × 32 × 1024	(4, 2)
Upsample_6	64 × 64 × 1024	(4, 2)
Upsample_7	128 × 128 × 1024	(4, 2)
Conv_transpose	256 × 256 × 3	(4, 2)

Table 1 U-Net Generator

From Table 1, we can learn more detail about setting, the input size is 256 × 256 × 5, 256 × 256 means the size of training image, and the third part is 5, which is a concatenation of mask and random noise. In conclusion, the input is the combination of mask and noise, and the size of mask is 256 × 256 × 4, and the size of noise is 256 × 256 × 1.

3.2. PatchGAN Discriminator

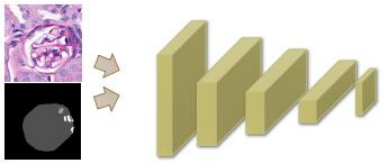

Figure 2 patchGAN Discriminator

PatchGAN Discriminator is suggested by tensorflow official website in tutorials, each 30 × 30 × 1 patch of the output of discriminator classify a portion of original image. As we can see it from Figure 2 and Table 2, the inputs size is 256 × 256 × 7, 256 × 256 means the size of image which is resized, and 7 means 3 + 4, which is the channel of pathological image and paired mask.

Layer	Output Size	(Kernel, Stride)
Inputs	256 × 256 × 7	(-, -)
Downsample_1	128 × 128 × 64	(4, 2)
Downsample_2	64 × 64 × 128	(4, 2)
Downsample_3	32 × 32 × 256	(4, 2)
Zero_padding	34 × 34 × 256	(-, -)
Conv_1	31 × 31 × 512	(4, 1)
Zero_padding	33 × 33 × 512	(-, -)
Conv_2	30 × 30 × 1	(4, 1)

Table 2 patchGAN Discriminator

3.3. Original/vanilla GAN Discriminator

Except use patchGAN as discriminator, I also try the original discriminator architecture to see if it will improve the generated images' quality. It's a surprise that it does make the training process more stable, and also solve the mode collapse problem, which we will discuss it in Experiments part.

As we can learned it from Table 2, Table 3 and Table 4, the biggest difference of patchGAN and original Discriminator is the final output size and the number of total parameters. In one hand, because of flatten and dense layer in original discriminator, its final output size is 1, but the patchGAN is 30 × 30 × 1. In the other hand, for

patchGAN, it have total 2,771,457 parameters, for original Discriminator, it have 3,816,961 parameters in total.

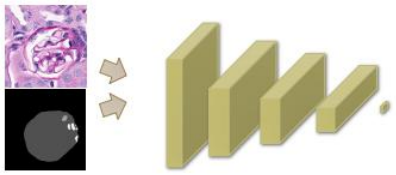

Figure 3 vanilla Discriminator

Layer	Output Size	(Kernel, Stride)
Inputs	256 × 256 × 7	(-, -)
Downsample_1	128 × 128 × 64	(4, 2)
Downsample_2	64 × 64 × 128	(4, 2)
Downsample_3	32 × 32 × 256	(4, 2)
Downsample_4	16 × 16 × 256	(4, 2)
Downsample_5	8 × 8 × 256	(4, 2)
Flatten	8 * 8 * 256	(-, -)
Dense	1	(-, -)

Table 3 vanilla Discriminator

Network	#params
U-Net Generator	54,427,907
PatchGAN Discriminator	2,771,457
Original Discriminator	3,816,961

Table 4 original Discriminator

4. Experiments

4.1. Datasets and Implementation Details

The data, including 1000 pathlogical images and corresponding 1000 masks, I used to train my model is part of dataset from (https://goo.gl/cFVxjn). There are 4 different type of medical image: nuclei podocyte data(which we used in this study), prostate radiology data, human biopsy data and mouse glomeruli data. In this independent study, I only used the first one(nuclei podocyte data) because it already provided 500 × 500 × 3 medical image and 500 × 500 × 1 masks. Images form other 3 datasets are super high-resolution medical image, which should be crop before we use it.

In order to avoid mode collapse and increase image diversity, when building up data pipeline, I use random crop, random mirror on dataset, just like Figure 4.

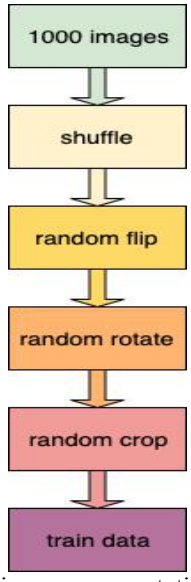

Figure 4 image augmentation process

4.2. Metrics

Fréchet Inception Distance(FID)[7], is a metric which can measure the distance of distribution for generated samples and real samples by using Inception-v3 model, which is very popular in measuring the quality of generated samples. Low FID score means that the generated image are very similar to real images. After 3k iteration, both train and test reach lowest FID score: around 150.

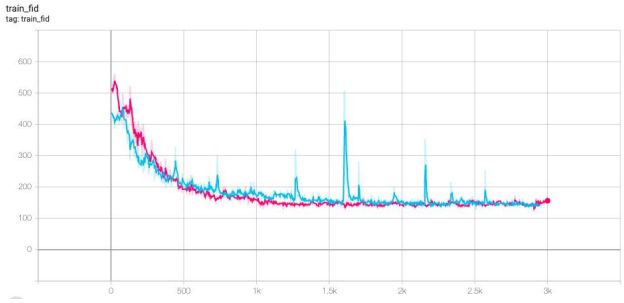

Figure 5 FID of train data

As we can see it from Figure 5 and Figure 6, the blue line means patchGAN Discriminator, and the pink one means original/vanilla Discriminator. It's obvious that the training process of original Discriminator are more stable than pathGAN. The main reason for this difference could be that original discriminator have more parameter and more complicated architecture than patchGAN, which lead to more capability, judt like Table 4.

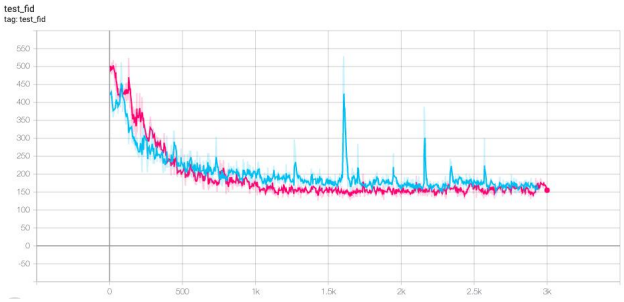

Figure 6 FID of test data

4.3. Results/Comparison to Previous Work

As we discuss before, there exists a mode collapse problem in patchGAN. Mode collapse means that given different inputs, the generator can only generate "one kind of" image, like below Figure 7 and 8, the first column means the training pathological image, the second column means the corresponding mask, the final column means generated image. Although I used same mask as part of input, it should still generated different medical image, because of random noise. Noise is 256 × 256 × 1 random normal vector.

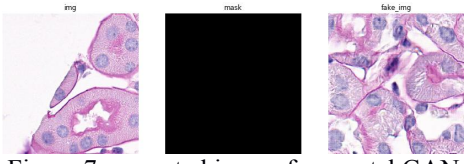

Figure 7 generated image from patchGAN

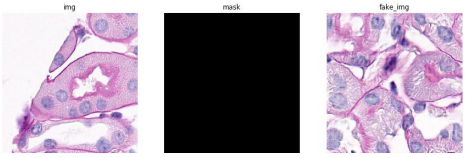

Figure 8 generated image from patchGAN

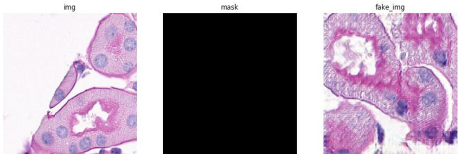

Figure 9 generated image from vanilla GAN

Figure 10 generated image from vanilla GAN

However, in original discriminator, there is no such issue. As we can see it from Figure 9 and 10. Same mask image can produce different pathological image.

From Figure 11 and 12 are other generated samples, as we can see, they are all very realistic and high quality..

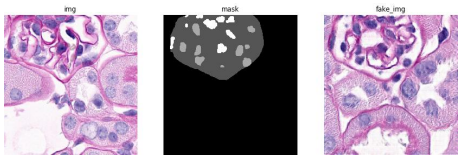

Figure 11 generated image from vanilla GAN

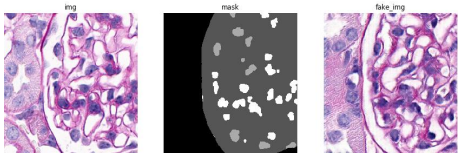

Figure 12 generated image from vanilla GAN

Figure 13 and 14 are the training process of same mask with different noise every 400 iterations, as we can it, the quality of generated image is getting better and better. After 1k iteration, the quality of generated images are quite similar from FID and visual effect, so there is no need to show all 3k iterations images.

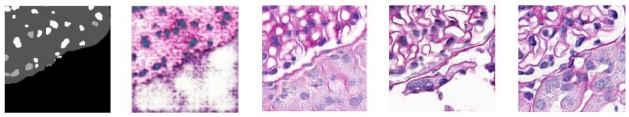
Figure 13 sample from iteration : 50, 450, 850, 1250

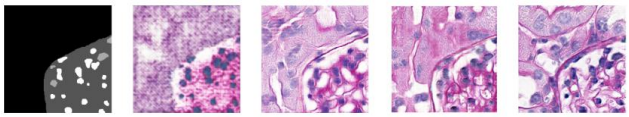
Figure 14 sample from iteration : 50, 450, 850, 1250

5. Discussion, Conclusions and Future Work

As we can see it from the generated pathological images, For one thing, it's hard to tell which image is generated and which image is real life due to the highly similarity of them. For another, it's also hard for Inception-v3 to tell that. Before training, I do calculated one special FID by passing real train images and real test images to see what's the lowest FID model may reach, it's about 140. And after training process, our model can generated images which reach around 150, which is amazing when compared to 140.

Also there are some drawbacks in this independent study. First, in my implementation, I calculated FID score by passing 64(batch size) real images and 64 generated images

to inception-v3 model. 64 may be too small, I should try to passing more images in order to get accurate result; Second, I didn't explore more paper about medical image generation, so I can't compare my result with others; Third, the main reason I train this model is to use generated images to do segmentation job to see if it can reach better performance(accuracy, F1 score and so on), but I didn't use the generated images to test that, which is a pity.

References

[1] I. J. Goodfellow et al., "Generative Adversarial Networks," arXiv:1406.2661 [cs, stat], Jun. 2014.

[2] B. Lutnick et al., "An integrated iterative annotation technique for easing neural network training in medical image analysis," Nat Mach Intell, vol. 1, no. 2, pp. 112–119, Feb. 2019.

[3] F. Mahmood et al., "Adversarial U-net with spectral normalization for histopathology image segmentation using synthetic data," in Medical Imaging 2019: Digital Pathology, 2019, vol. 10956, p. 109560N.

[4] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired Image-to-Image Translation using Cycle- Consistent Adversarial Networks," arXiv:1703.10593 [cs], Mar. 2017.

[5] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-Image Translation with Conditional Adversarial Networks," arXiv:1611.07004 [cs], Nov. 2016.

[6] C. Li and M. Wand, "Precomputed Real-Time Texture Synthesis with Markovian Generative Adversarial Networks," arXiv:1604.04382 [cs], Apr. 2016.

[7] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium," arXiv:1706.08500 [cs, stat], Jan. 2018.