Bilkent University
CS 485 Deep Generetive Networks Homework

Utku Kurtulmus 21903025

I. Introduction

THIS paper aims to study further the generative adversarial networks (GANs) and their various applications in image synthesis, style transfer, and improvement techniques. To do that, 6 papers published in the last two years are selected to be surveyed, and what they propose is discussed along with their limitations.

II. Literature Review (Part 1)

1) Scaling Up GANs for Text-to-Image Synthesis [1]: This paper investigates a major drawback of GANs, which is scalability. This is a huge drawback for GANs, considering the success of autoregressive and diffusion models on large-scale generative models. Authors suggest that simply increasing the capacity of the famous StyleGAN architecture quickly becomes unstable. They introduce a new architecture called GigaGAN to overcome these issues, and make GigaGAN suitable for text-to-image synthesis tasks.

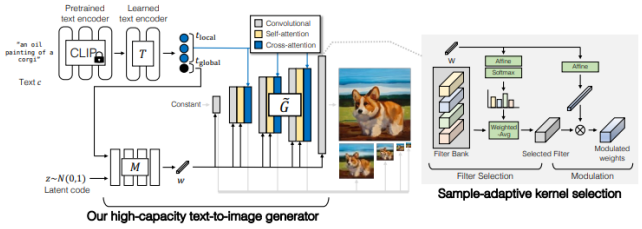


Fig. 1: GigaGAN Architecture

GigaGAN uses an improved version of the StyleGAN Architecture. In Generator, it has a mapping network starting with a random Latent code, and this latent code is mapped into the w space, which includes the style information. Nonetheless, since this is a text-to-image generation task, we need the information of the given input text while generating the w vector, therefore, along with the random noise vector, we pass the global context embedding of the given input text. Then, we feed this style information to our Convolutional Layers.

However, unlike StyleGAN, GigaGAN introduces a new method called ”sample-adaptive kernel selection” to handle the diversity of the internet images. Instead of simply increasing the width of the kernels, they propose a less demanding approach to overcome the issue. They have a bank of filters, each focusing on different features of the images. According to the style vector, which includes the information of the global context of the given input text, they adaptively select the most suitable filter for the task. Then, they use a self-attention layer after each convolutional layer to focus on the whole image instead of increasing the receptive fields of the convolutional layers to capture long-distance relationships. Finally, they introduce a cross-attention layer after each self-attention layer, which is fed by each word embedding called as t_{local}.

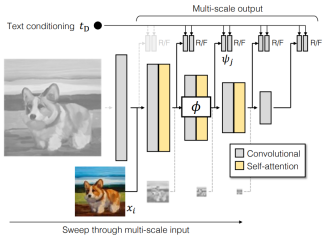


Fig. 2: Discriminator Architecture of GigaGAN

In the discriminator architecture, they use a text-conditional discriminator to investigate the success of generated images based on given input text. Similar to the generator, they use a pre-trained encoder and a learnable encoder to get the text conditioning information t_{D}. Moreover, they process each generated output from each layer of the generator separately to avoid high-level features dominating the output of the discriminator. Furthermore, they introduce a self-attention layer after each convolutional layer to capture the global context.

	Model	Type	# Param.	# Images	FID-30k ↓	Inf. time
256	DALL-E [54]	Diff	12.0B	1.54B	27.50	-
	GLIDE [46]	Diff	5.0B	5.94B	12.24	15.0s
	LDM [58]	Diff	1.5B	0.27B	12.63	9.4s
	DALL-E 2 [53]	Diff	5.5B	5.63B	10.39	-
	Imagen [59]	Diff	3.0B	15.36B	7.27	9.1s
	eDiff-I [4]	Diff	9.1B	11.47B	6.95	32.0s
	Parti-750M [73]	AR	750M	3.69B	10.71	-
	Parti-3B [73]	AR	3.0B	3.69B	8.10	6.4s
	Parti-20B [73]	AR	20.0B	3.69B	7.23	-
	LAFITE [80]	GAN	75M	-	26.94	0.02s
512	SD-v1.5* [57]	Diff	0.9B	3.16B	9.62	2.9s
	Muse-3B [9]	AR	3.0B	0.51B	7.88	1.3s
	GigaGAN	GAN	1.0B	0.98B	9.09	0.13s

Fig. 3: Test Results of GigaGAN Compared to Other Generative Models

As a result, they were able to generate a 512px image with a 0.13-second inference time, beating all the non-GAN-based architectures with achieving good FID scores. Nonetheless, the authors suggest that even though GigaGAN achieved those results, it has limitations compared to well-known models like DALL-E. It fails to produce high-quality images when compared to DALL-E in terms of photorealism and text-to-image alignment.

2) Pastiche Master: Exemplar-Based High-Resolution Portrait Style Transfer [2]: This paper suggests an alternative

version of the StyleGAN to provide a more natural style transfer. They offer separate style mappings for the content and the style of the images, which are called intrinsic and extrinsic style paths.

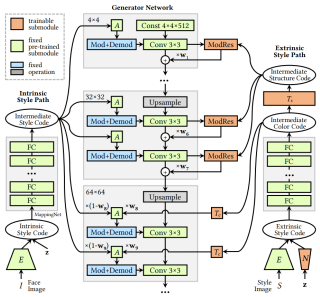


Fig. 4: DualStyleGAN Architecture

The intrinsic style path is similar to the StyleGAN and is used to control the style of the original domain. On the other hand, an extrinsic style path is introduced to the model, and it controls the target style, unlike the StyleGAN, which uses the StyleMixing strategy. Similar to how we feed our domain style, we progressively feed the generator with the target style coming from the extrinsic style path.

As a result, they have achieved beating other generative models like GNR, StarGANv2, and UI2I-style according to user-preference scores. Nevertheless, the authors claim that their work still has limitations. Firstly, DualStyleGAN fails to capture non-facial regions. Secondly, it generates unnatural nose styles for anime due to the abstract nose styles in anime. Lastly, it can have a bias on the training data.

3) SemanticStyleGAN: Learning Compositional Generative Priors for Controllable Image Synthesis and Editing [3]: This paper suggests a revolutionary way to control image synthesis and editing tasks. It emphasizes the main problem of StyleGANs, which are global latent codes. While global latent codes are a good way to construct realistic images, they have limitations in capturing nuanced features and having precise control over them. Even though these global latent codes can be learned and used in image editing tasks, they suffer from biases in the latent space. For instance, while trying to change a learned global style in StyleGAN, unexpected local changes can occur due to their correlations with the global latent codes.

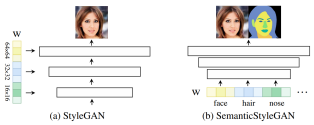


Fig. 5: Comparison between StyleGAN and SemanticStyleGAN

Latent Codes in SemanticStyleGAN are learned from different parts of the images, and they can represent both the structural and textural design of the local parts, unlike the StyleGAN, where coarse styles and fine styles are separated. Therefore, to change the texture of a specific part in the image, SemanticStyleGAN doesn't need to change the global texture design.

As a result, they achieve similar ID, and MSE scores compared to StyleGAN2 while allowing more precise control of the local styles. However, SemanticStyleGAN still has limitations. Firstly, SemanticStyleGAN cannot scale to domains with classes that are too diverse. Secondly, fully decoupling pose, shape, and texture information may not be possible due to their ambiguous boundaries.

4) InsetGAN for Full-Body Image Generation [4]: This paper introduces an approach to full-body image synthesis tasks. GANs like StyleGAN are very successful in generating images in the face domain, but generating full-body is more challenging due to the diversity of the images. InsetGAN uses multiple Generators to generate different parts of the images, and after they combine these generated images. The main challenge here is to learn latent codes in such a way that generated images can be combined.

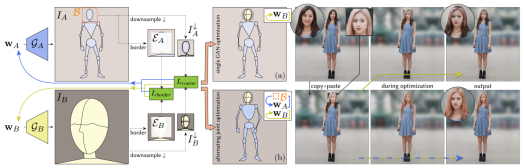


Fig. 6: InsetGAN Pipeline

They first train a canvas generator to generate a canvas. Then, specialized parts of the images generated by separate generators are pasted into this canvas. Then, they seek the latent codes to combine those pasted insets in a consistent way, such as smooth color changes in the boundaries and the pasted parts looking like they belong to the image.

As a result, they achieve good FID scores both for the faces and the body when the results are untruncated. Untruncated FID scores are 13.96, and with t=0.7 and t=0.4, FID scores go up to approximately 25 and 70, respectively. However, as the author emphasizes, increasing truncation reduces the variation and increases FID scores, but it is crucial for getting natural images with fewer artifacts.

Nonetheless, as the authors suggest there are several limitations of the InsetGAN. Firstly, during the optimization phase, where we seek latent codes to combine images, some styles, such as hairstyle, neckline, or clothing details, might change. Secondly, there are symmetry problems in generated images. Finally, the model fails to generate variable options in body type and pose styles.

5) Improving GANs with A Dynamic Discriminator [5]: This paper introduces an alternative discriminator architecture to improve the performance of the discriminator according to the training data. When the training dataset is large, and the generator gets better and better over time, it becomes very

difficult for the discriminator to differentiate results and supervise the generator. Therefore, they increase the capacity of the discriminator to get a more powerful discriminator. When the case is the opposite, and we have limited training data Discriminator can memorize the data and overfit. Therefore, in such cases capacity of the Discriminator should be diminished. The adjustment is made on the fly without hindering the training process.

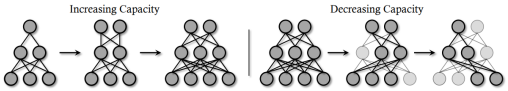


Fig. 7: On-the-fly capacity adjustment in DynamicD

If the discriminator is weak they introduce new filters every several iterations, and if the discriminator is too strong, they randomly dropout some of the filters. As a result, they achieved to get better FID scores on the tests done on StyleGAN2. When they went from baseline-full to baseline half capacity for discriminator, they decreased FID score from 179.21 to 50.37 when they had limited data (0.1k). When they have a large number of data (140k), differences in FID scores are relatively small, but still, they achieved to decrease to FID score from 4.73 to 3.53 when they go baseline-half to baseline full.

Nonetheless, as the authors suggest there are still limitations of DynamicD. Currently, DynamicD only adjusts the network capacity by shrinking or extending the layer width, but it doesn't consider other factors, such as network depth. Moreover, DynamicD lacks theoretical demonstration and experiments done only on CNN-based discriminators, it might give different results on other discriminator architectures like transformer-based discriminators.

6) MCL-GAN: Generative Adversarial Networks with Multiple Specialized Discriminators [6]: This paper suggests an alternative way to deal with the chronic mod collapse problem in GANs. Mod Collapse problem occurs when the generator starts to generate a limited number of outputs that constantly fool the discriminator. Therefore, the authors suggest a different approach where there are multiple discriminators trained on specific subsets of the input data collaborating with each other. In this case, the generator has to fool a subset of discriminators having expertise on specific parts of the input.

Firstly, expert discriminators are selected as the ones that give the lowest loss score for a given sample. This naturally creates clustering in the given dataset, and eventually, each expert model becomes an expert of a specific subset in the original dataset. On the other hand, as the authors claims non-expert discriminators shouldn't be overconfident not to lead the model to wrong predictions in score aggregation process. Therefore, instead of 0 and 1 as labels, they have given soft labels such as [0.5, 0.5] for real and fake images. Moreover, they introduce another loss function to guarantee balanced assignments of discriminators where they penalize the deviations from a balanced distribution of expert selections.

They evaluate the success of their model on the CIFAR-10 dataset with DCGAN architecture. They managed to get a lower FID score with 10 discriminators compared to the base DCGAN architecture with one generator and one discriminator. Nonetheless, as the author suggests, the MCL-GAN still has limitations. Firstly, how the MCL-GAN would perform with a limited variety of small training datasets is unambiguous. Morevover, the robustness of the additional hyperparameters introduced in MCL-GAN is not tested either.

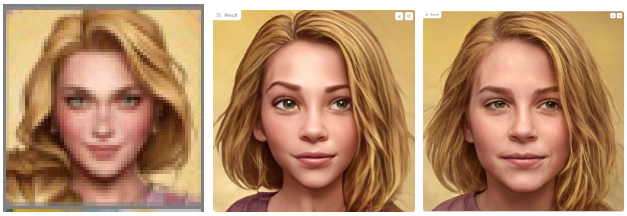
III. Selected Paper and Results On Example Images (Part 2)

A. Pastiche Master: DualStyleGan

I select the paper Pastiche Master for its high user preference scores and revolutionary style transfer mechanism. I run their model on Hugging Face with example images to see the results.

Fig. 8: Input image and reconstructed image using Z+ encoder

Initially, I selected an example input image and selected an encoder for my image. It gives the option to select either Z+ encoder or W+ encoder. W+ encoder is more successful in reconstructing the image, while Z+ encoder allows better stylizing. You can see the example image and reconstructed image using Z+ encoder in Figure 8.



(a) Selected Cartoon Style Image with image index 26

(b) Generated image with structure weight 0.6 and color weight 0.1

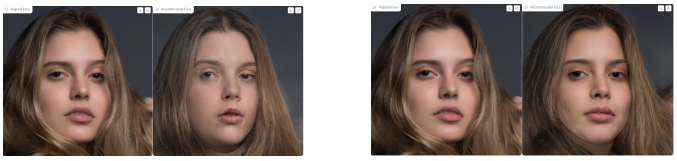
(c) Generated image with structure weight 0.1 and color weight 0.1

Fig. 9: Comparison of generated images

After that, I selected a style image from the set of example cartoon images to apply the style to the reconstructed image. The model allows us to set the structure weight and color weight of the generated image to have better control on it. As you can see, if we increase the structure weight, the original input image will start to get closer and closer to the style image. On the other hand, if we set a small structure weight, the generated image better resembles the original image.

IV. Limitations (Part 3)

To further challenge the capabilities of DualStyleGAN, I mainly investigated the edge cases where the algorithm might fail. I encountered several problems.

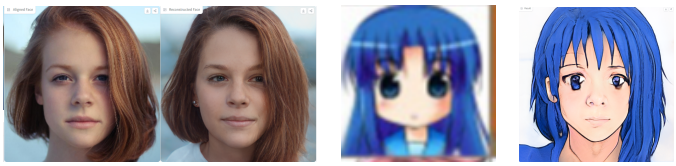


(a) Reconstruction with Z+ encoder

(b) Reconstruction with W+ encoder

Fig. 10: Comparison of Encoders

Firstly, Z+ encoder can fail to reconstruct the images. This can be problematic since Z+ encoder allows better stylizing. Furthermore, even in W+ space, the model fails to capture non-facial details around the neck and the teeth. The main reason for such failures probably is related to how we train our model. The training data mainly consists of images where the faces cover the big parts of the image. Therefore, during the encoding process; capturing the facial features is more important for the model to generate realistic images. Moreover, the variety of the non-facial regions makes it harder to learn meaningful global styles.



(a) Input Image (b) Target Style (c) Result

Fig. 11: Anime Style Failure

Secondly, I encountered several issues while transferring anime styles to example images. Initially, the size of the eyes in the anime style tends to be bigger than in real human images. Therefore, while applying the styles, the boundaries of the eyes get unnaturally spread around the face. Moreover, anime styles have very abstract noses and mixing real human images with anime styles creates unnatural-looking noses for anime styles.

(a) Input Image (b) Rebuilt Image

Fig. 12: Failure of reconstructing tattooed faces

Thirdly, their model fails to encode tattooed or pierced faces. The reason for that is probably due to insufficient training data, including the tattooed or pierced faces. Nonetheless, from the reconstructed image, we can still see that the model achieved to construct the shape of the head and the nose. However, it also tried to bring eyeglasses to the reconstructed image.

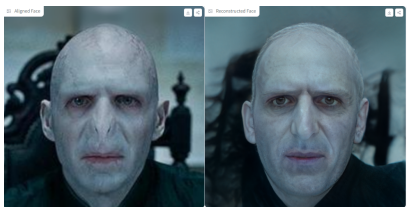


Fig. 13: Reconstruction of Voldemort

Finally, I tried the model with the famous noseless villain character from the Harry Potter series. I chose Voldemort to see how the model will perform when one of the input image's aspects is ambiguous. The model actually captures the other details of the input image without being affected by the nose of the input image. Nevertheless, it failed to capture the nose details as expected since it wasn't trained on such kinds of images.

References

[1] M. Kang, J.-Y. Zhu, R. Zhang, J. Park, E. Shechtman, S. Paris, and T. Park, *Scaling Up GANs for Text-to-Image Synthesis*, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2023, pp. 10124-10134.

[2] S. Yang, L. Jiang, Z. Liu, and C. C. Loy, *Pastiche Master: Exemplar-Based High-Resolution Portrait Style Transfer*, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022, pp. 7693-7702.

[3] Y. Shi, X. Yang, Y. Wan, and X. Shen, *SemanticStyleGAN: Learning Compositional Generative Priors for Controllable Image Synthesis and Editing*, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022, pp. 11254-11264.

[4] A. Frühstück, K. K. Singh, E. Shechtman, N. J. Mitra, P. Wonka, and J. Lu, *InsetGAN for Full-Body Image Generation*, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022, pp. 7723-7732.

[5] C. Yang, Y. Shen, Y. Xu, D. Zhao, B. Dai, and B. Zhou, *Improving GANs with A Dynamic Discriminator*, in *Advances in Neural Information Processing Systems*, vol. 35, S. Koyejo et al. (Eds.), Curran Associates, Inc., pp. 15093–15104, 2022. Available: https://proceedings.neurips.cc/paper_files/paper/2022/file/6174c67b136621f3f2e4a6b1d3286f6b-Paper-Conference.pdf.

[6] J. Choi and B. Han, *MCL-GAN: Generative Adversarial Networks with Multiple Specialized Discriminators*, in *Advances in Neural Information Processing Systems*, vol. 35, S. Koyejo et al. (Eds.), Curran Associates, Inc., pp. 29597–29609, 2022. Available: https://proceedings.neurips.cc/paper_files/paper/2022/file/beac6bfb7eac3d651307c16ac747df01-Paper-Conference.pdf.