The 4th International Conference on Computing and Data Science (CONF-CDS 2022)
DOI: 10.54254/2755-2721/2/20220563

Style-based Image Manipulation Using the StyleGAN2-Ada Architecture

Yuhong Lu

The University of Edinburgh, 57 George Square, Edinburgh, EH8 9JU, UK

m18251930597@163.com

Abstract. Style-based image manipulation is to fuse the types of two arbitrary images, which is a popular task in computer vision. StyleGAN is a sophisticated architecture for generating images of high qualities. The framework allows the generator to operate on a latent space that is disentangled and allows us to do scale-specific manipulation on the semantic information of the generated images. In this paper, the author managed to fuse the styles of two given images on a controllable degree. The resultant images have natural appearances approximating real human portraits. Our method provides qualitative results for style-fusion of two given images, which achieves satisfy. Since StyleGAN offers an unraveled latent space representing disentangled semantics, the author hopes to use it on tasks like GAN inversion and manipulate images in a fine-grained control, which is the future work.

Keywords: Style-based image manipulation, StyleGAN, GAN inversion.

1. Introduction

Image manipulation includes the change or modification of a photograph utilizing a variety of methods to reach required results [1]. The handling aspects involve but not limited to lighting, surrounding, and subject styles such as poses, face shapes, accessories, etc. for a portrait. Isola et al. [2] saw this as an Image-to-image translation problem and solved it by the pix2pix architecture under paired training data. For cases where paired relationships are not easily available, CycleGAN [3] managed to translate images from a source domain X to a target domain Y in the absence of the one-to-one correspondence in the training data. This paper aims to investigate the image style manipulation. Specifically, if there are a portrait demonstrating masculinity and another showing femininity, then this paper works to create an image whose style combines the two by a certain degree that is controllable. The idea is motivated by the sad ending of the television series Game of Thrones, where Jon Snow had to, sadly, kill Daenerys Targaryen for a peaceful world in the final season. Many audiences, including the author, expressed their great regret about that and try to imagine a parallel world where the two protagonists live happily together and continue the bloodline of House Targaryen. So, if someday HBO responds to the public and decides to remake the final season, they would probably need a portrait of their descendant. Then the approach described here can offer help. Besides the style-fusion job, due to the ability to project images to a latent space which is highly disentangled, the StyleGAN2-ada architecture [4] holds considerable potential in applications related to a concept of GAN inversion [5]. And the author makes the related studies as the future work described in the discussion section of this paper.

DOI: 10.54254/2755-2721/2/20220563

2. Related Works

2.1. Image style manipulation

Image style manipulation deals with styles, which include the change of color or texture of an image. Some scholars argue that the contents can also be one type of style in human perception, and this paper tends to take this broad view when talking about the style-fusion in this project. Achievements abound in this field. One outstanding example is the style transfer [6] which keeps the contents while converting the image to a target style, as shown in Figure 1 below. It can be realized with the proposed AdaIN [6] operation, which normalizes the feature map of the given image before anti-normalizing it by using the statistics (i.e., the mean and the variance) extracted from the feature map of the target image. By adopting the Generative Adversarial Network (GAN) [7] architecture with the Attention module [8] on the generator and discriminator, J. K. et al. [9] saw this style transfer as an Image-to-Image Transformation task, with AdaLIN [9] to increase robustness. Although style transfer is not a part of the project, it is straightforward as an example to illustrate the one of the effects of the image style manipulation.

Figure 1. A style transfer example where the content of Lena is retained, while its style is altered by the target image of different degrees [6].

2.2. Development path of StyleGAN architecture

The project heavily relies on idea of StyleGAN [10] proposed in 2019. A highlight of this architecture is that it incorporates a mapping network f and therefore provides the opportunity for the generator to generate images from the disentangled latent codes that contains semantic information. In the next year, NVIDIA published StyleGAN2 [11], an enhanced architecture on the previous one. New frameworks, including structures for weight demodulation, skip connections and residual network, were embedded. The Path Length Regularization was used as a part of the loss function, and increased capacity was given to the last feature map for a more stable learning outcome of the highest resolution. During the same year, StyleGAN2-ada was proposed as a solution for the generative model to work under limited data. Last year the newest StyleGAN3 [12] came out and they also published their work on GitHub. In this paper, the author decided to not go that advanced and rely on StyleGAN2-ada for the implementation.

3. GAN inversion

Modern GANs are able to encode rich semantic information in the latent space for the generated images to enjoy diverse attributes, such as aging and expression. Researchers want to utilize such a rich expressiveness, but they do not want to operate only on a limited scope of images the generator can produce. They want an extended scope and manipulate images in the real world on such diversity. GAN inversion [13] was born in response to this appeal. A real image is projected back to the latent space of a pretrained generator to find the optimal latent code, which should be able to reconstruct the image back faithfully and photo-realistically by the generator. If this criterion is met, we can navigate through the latent space to find interpretable and disentangled directions, obtain latent codes with desired semantics, and achieve fine-grained controls over the given image on the semantic level. Figure 2 illustrates this idea. And as suggested by the result of this paper, benefitting from a disentangled latent space offered by StyleGAN, there is a high hope to achieve the fine-scale manipulations as Figure 2 tells.

DOI: 10.54254/2755-2721/2/20220563

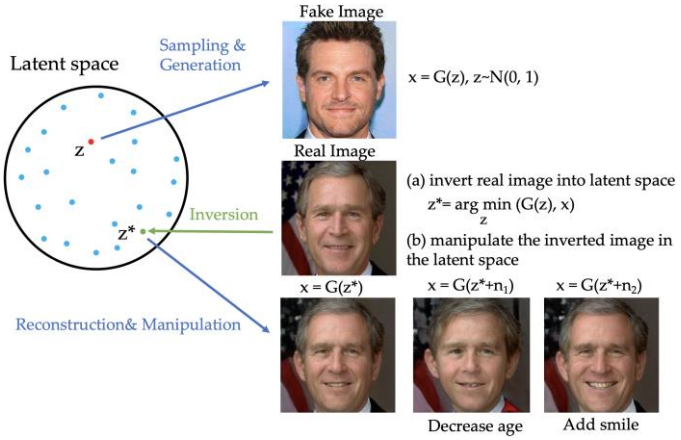

Figure 2. Image manipulations using GAN inversion. The attributes of George W. Bush are edited by varying on the latent code z^* [13].

4. Our Approach

As described in Section 1, this paper aims to combine styles of two given images in different ways and produces results accordingly. The results directly benefit from a disentangled latent space $\mathcal{W}$, as well as the network's ability to assign one style to feature maps of one resolution (i.e., scale-specific modifications). Since both two merits are from the legacy of styleGAN, here the author feels it necessary to briefly introduce them before going into the implementation details of the style-fusion work. In the following subsections, methods are detailed.

4.1. Disentangled latent space $\mathcal{W}$ containing semantic information

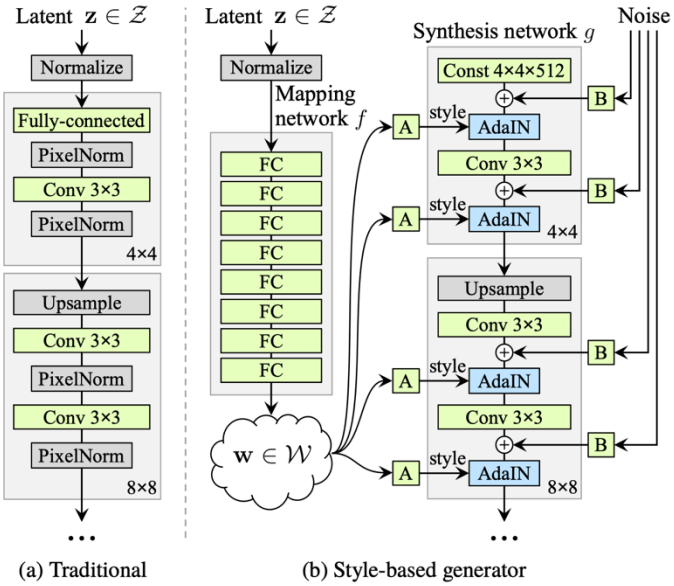

Figure 3. An innovative idea of StyleGAN is involving the mapping network f to disentangle the latent space $\boldsymbol{z}$ [10].

DOI: 10.54254/2755-2721/2/20220563

Table 1. PPL scores for different configurations [10]. The style-based generator with noise inputs (E) has a much lower score than that of a traditional generator (B).

Method	Path length	
	FULL**	END***
B Traditional generator $\mathcal{Z}^*$	412.0	415.2
D Style-based generator $\mathcal{W}$	446.2	376.6
E + noise inputs $\mathcal{W}$	**200.5**	**160.6**
+ Mixing 50% $\mathcal{W}$	231.5	182.1
F + Mixing 90% $\mathcal{W}$	234.0	195.9

*Traditional generator $\mathcal{Z}$ means that the PPL score is measured w.r.t. $\mathcal{Z}$, i.e., by Eq. (1) accordingly.

**FULL means the full-path length in favor of the entangled latent space $\mathcal{Z}$ rather than $\mathcal{W}$. For details, please refer to [10].

***END means that measurements are restricted to the path endpoints. For details, please refer to [10].

The StyleGAN allows the generator to produce images from a latent space $\mathcal{W}$ that is disentangled, rather than from the entangled latent space $\mathcal{Z}$. Otherwise, the generator has to operate on a raw latent code $\mathbf{z} \in \mathcal{Z}$, leading to a high score (shown in Table 1, configuration B) of Perceptual Path Length (PPL) defined as

$$l_{\mathcal{Z}} = \mathrm{E}\left[\frac{1}{\varepsilon^2}\mathrm{d}\left(\mathrm{G}\left(\mathrm{slerp}(\mathbf{z}_1, \mathbf{z}_2; t)\right), \mathrm{G}\left(\mathrm{slerp}(\mathbf{z}_1, \mathbf{z}_2; t+\varepsilon)\right)\right)\right],$$

$$\text{where} \begin{cases} \mathbf{z}_1, \mathbf{z}_2 \in \mathcal{Z} \\ t \sim U(0,1) \\ \mathrm{G} \text{ is the traditional generator} \\ \text{step size } \varepsilon \text{ is a small positive, } e.g., 10^{-4} \\ \mathrm{d}(\cdot,\cdot) \text{ evaluates the perceptual distance of two generated images} \\ \mathrm{slerp}(\cdot) \text{ denotes the spherical interpolation.} \end{cases} \tag{1}$$

We see from Eq. (1) that the PPL score indicates the averaged distance between the two images generated from adjacent latent codes on a particular path connecting $\mathbf{z}1$ and $\mathbf{z}2$. The specific measurement for such a "distance" can be the Fréchet Inception Distance (FID) [14] comparing the distributions of the two feature maps on the same layer close to the output node on the Inception v3 architecture. And because of the location of the feature map, FID agrees well with the human perceptual similarity judgments. So, a high PPL score for the entangled latent space $\mathcal{Z}$ means that even a small perturbation in $\mathcal{Z}$ can lead to a huge difference in appearance of the two generated images. In this case the results of the generator are highly unpredictable. On the contrary, if the generator operates on a disentangled space $\mathcal{W}$, then as shown in Table 1 (configuration E), with noise inputs the PPL score of the generator,

$$l_{\mathcal{W}} = \mathrm{E}\left[\frac{1}{\varepsilon^2}\mathrm{d}\left(\mathrm{g}\left(\mathrm{lerp}(f(\mathbf{z}_1), f(\mathbf{z}_2); t)\right), \mathrm{g}\left(\mathrm{lerp}(f(\mathbf{z}_1), f(\mathbf{z}_2); t+\varepsilon)\right)\right)\right],$$

$$\text{where} \begin{cases} \mathbf{z}_1, \mathbf{z}_2 \in \mathcal{Z} \\ t \sim U(0,1) \\ f \text{ is the } 8-\text{layer mapping network} \\ \mathrm{g} \text{ is the style}-\text{based generator} \\ \text{step size } \varepsilon \text{ is a small positive, } e.g., 10^{-4} \\ \mathrm{d}(\cdot,\cdot) \text{ evaluates the perceptual distance of two generated images} \\ \mathrm{lerp}(\cdot) \text{ denotes the linear interpolation.} \end{cases} \tag{2}$$

It is much lower. Therefore, the behavior of the style-based generator is much more predictable in that it works on a disentangled latent space $\mathcal{W}$, in which a perturbation is more in line with the effect it should induce. A latent space enjoying such a property can be said to contain semantic information,

DOI: 10.54254/2755-2721/2/20220563

because such a predictability offers us the opportunity to fine-tune $\mathbf{w} \in \mathcal{W}$ and yield resultant images accordingly.

4.2. Scale-specific modifications to the styles

As shown by the architecture in Figure 3 (b), styles are sent into blocks of different resolutions with the purpose of the scale-specific modifications. The effect of each of the styles is localized within the block, thanks to the AdaIN operation,

$$\mathrm{AdaIN}(\mathbf{x}_i, \mathbf{y}) = \mathbf{y}_{s,i} \frac{\mathbf{x}_i - \mu(\mathbf{x}_i)}{\sigma(\mathbf{x}_i)} + \mathbf{y}_{b,i},$$
$$\text{where} \begin{cases} styles\ \mathbf{y} = (\mathbf{y}_s, \mathbf{y}_b) \text{ as the results of the learned affine transformation from } \mathbf{w} \\ \mathbf{x}_i \text{ denotes the } i^{\text{th}} \text{ feature map in the progressive growing generator,} \end{cases} \tag{3}$$

where each of the feature maps has to get normalized first before being edited by the style. So, each style manages only one convolution before being overridden by the next AdaIN operation.

4.3. Loss functions of the StyleGAN model

The loss of discriminator is defined as

$$\mathcal{L}_{\mathrm{D}} = \mathcal{L}_{\mathrm{D\ for\ generated\ images}} + \mathcal{L}_{\mathrm{D\ for\ real\ images}}$$
$$= \mathrm{E}_{\mathbf{w}\sim \mathrm{P}_{\mathcal{W}}} \left\{ -\log\left(1 - \mathrm{sigmoid}\left(\mathrm{D}\left(\mathrm{G}(\mathbf{w})\right)\right)\right)\right\} + \mathrm{E}_{\mathbf{x}\sim \mathrm{P}_{\mathrm{real}}} \left\{ -\log\left(\mathrm{sigmoid}\left(\mathrm{D}(\mathbf{x})\right)\right)\right\}. \tag{4}$$

And the loss of the generator is defined as

$$\mathcal{L}_{\mathrm{G}} = \mathrm{E}_{\mathbf{w}\sim \mathrm{P}_{\mathcal{W}}} \left\{ -\log\left(\mathrm{sigmoid}\left(\mathrm{D}\left(\mathrm{G}(\mathbf{w})\right)\right)\right)\right\}. \tag{5}$$

From Eq. (4) and Eq. (5) we learn that, on the discriminator's side if it wants to reduce its loss *L*D, it has to produce a large value for real images while a low value for the generated ones. On the generator's side, however, it is committed to produce images that the discriminator thinks as real. So, by investigating the loss functions we can feel the adversarial positions of D and G.

4.4. Style-fusion of two given images

A disentangled latent space $\mathcal{W}$ and the network's ability to do scale-specific modifications w.r.t. styles provide us the chance to do style-fusion of two given images on purpose. Note that the referred style-fusion is different from the public job where images are generated from random seeds in $\mathcal{Z}$ whose semantic information is unknown to us. This paper combines styles of two given images whose semantic information is explicitly open to us. The method is described by the following algorithm. As shown below, this paper used a *style layer range* of latent code **w** from *image_2* as an ingredient to replace that part of latent code of *image_1*.

Algorithm Style-fusion for two given images

Input: *image_1* and *image_2* from the FFHQ [10] dataset, the *pre-trained network* [4] on this dataset, *style layer range*.
Process: the function *style_merge* (*image_1*, *image_2*, *pre-trained network*, *style layer range*)
1: Find the optimal latent code *w1_opt* of *image_1* by 1000 iterations using the *pre-trained network* and a part of the script projector.py [4] published
2: Find the optimal latent code *w2_opt* of *image_2* by 1000 iterations using the *pre-trained network* and a part of the script projector.py [4] published
3: *w_temp* = *w1_opt*
4: *w_temp*[*style layer range*] = *w2_opt*[*style layer range*]
5: Synthesize a new image from the *w_temp* using the generator in the *pre-trained network*
Output: the image combining the styles of *image_1* and *image_2* with a degree controlled by *style layer range*.

DOI: 10.54254/2755-2721/2/20220563

5. Results and Analysis

Before using the pre-trained network to achieve style-fusion on the FFHQ dataset, the author trained the StyleGAN2-ada model on two datasets: the resized FFHQ (from 1024×1024 to 256×256, 70k images) dataset and the LSUN CAT dataset (256×256, 200k images) and saw the results. The author used 2×Tesla V100 for the training work, and for the same 5000k images (how many thousands of images have been shown to the discriminator), the LSUN CAT takes 17h 21m 49s while the resize FFHQ takes 17h 48m 28s.

5.1. Training results on the LSUN CAT (256×256, 200k images)

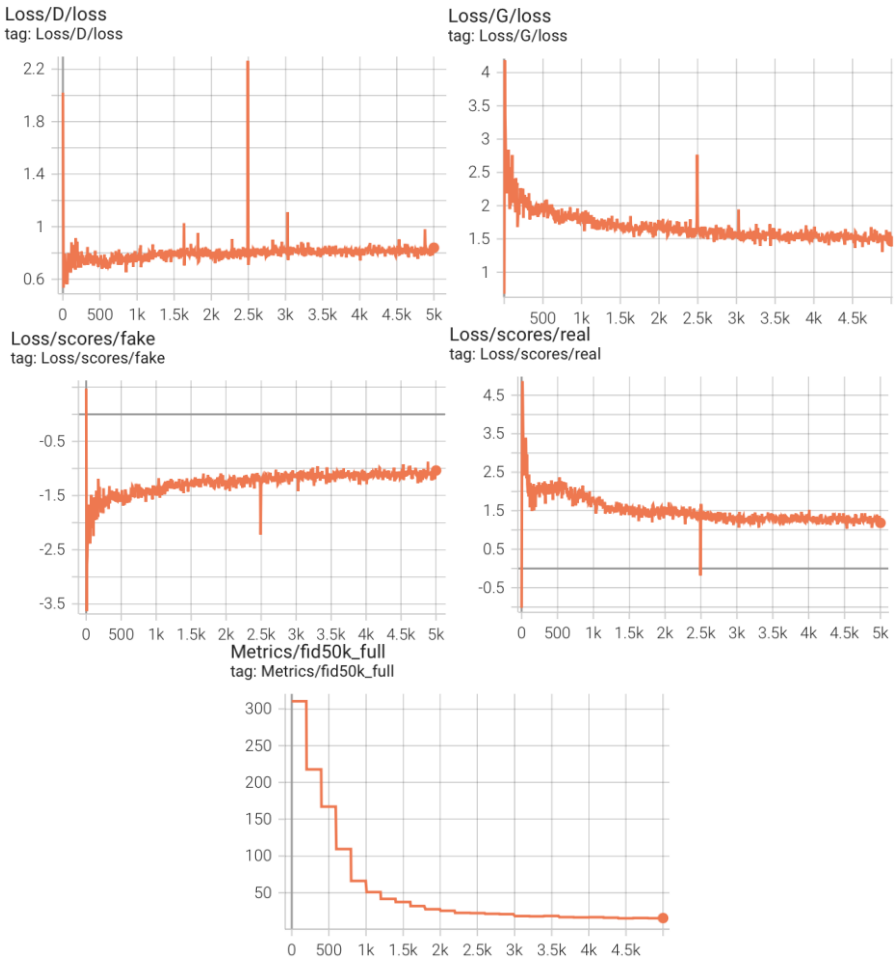

Figure 4. (From left to right) Loss of discriminator || Loss of generator || Discriminator's outputs for generated images || Discriminator's outputs for real images || Fréchet Inception Distance (FID) scores recorded for every 200k images.

From the curves we see that during training, L_G goes down steadily while L_D fluctuates on a level that is not high. The remarkable spike at 2.5k images is probably resultant from a broken image failing to fit the model trained at that time, because after that point the training goes back to normal instantly. It is more interesting to investigate the outputs of D w.r.t. the real and generated images during training. At start D just produces random numbers for the two kinds, but it learns something quickly afterwards and produces a negative number for the fake image while a positive number for the real. The exciting thing is that as training goes, the difference between the outputs is shrinking, which means that the G managed to cause trouble to the D and makes it less certain for a decision than cases before. And we can also see

DOI: 10.54254/2755-2721/2/20220563

from the FID that the generated images are more and more resemble the real ones. All of these are positive signs for training the GAN.

The following Figure 5 shows a series of images generated by the trained model.

Figure 5. (From left to right) Images generated on random seeds 44, 78, 457, 873, 999, 1024, 2048 from the model trained by the LSUN CAT.

5.2. Training results on the resized FFHQ (from 1024×1024 to 256×256, 70k images)

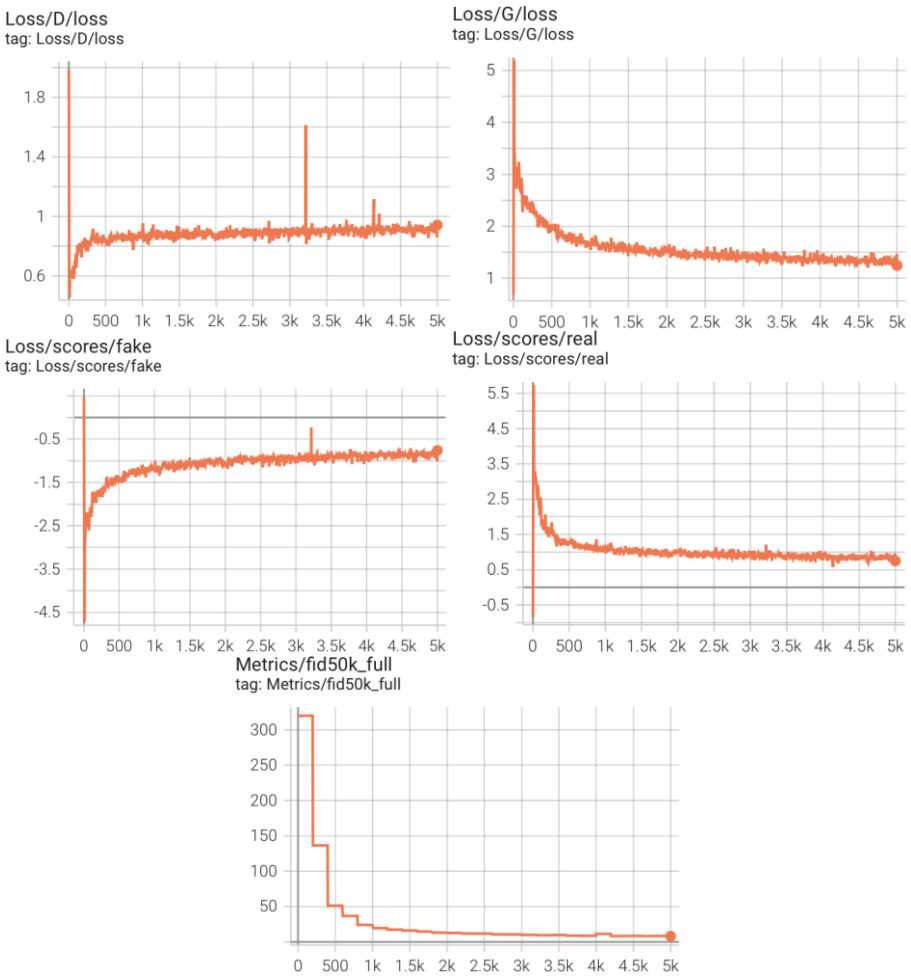

Figure 6. (From left to right) Loss of discriminator || Loss of generator || Discriminator's outputs for generated images || Discriminator's outputs for real images || Fréchet Inception Distance (FID) scores recorded for every 200 k-img.

Images generated by the model trained by LSUN CAT are not all satisfactory, and the author doubts a part of reasons are about the dataset itself. Images there are probably crawled from website and there are stuffs other than cats. The FFHQ dataset, on the other hand, are all about high-quality human faces

DOI: 10.54254/2755-2721/2/20220563

where locations for eyes, nose, and mouth are consistent with each other. It is hopeful that by using this dataset, images generated will be finer. The following Figure 7 fulfilled this wish.

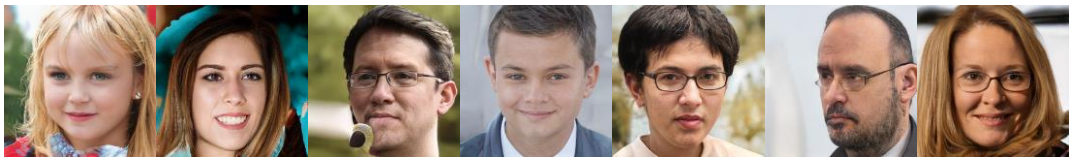

Figure 7. (From left to right) Images generated on random seeds 44, 78, 457, 873, 999, 1024, 2048 from the model trained by the resized FFHQ.

5.3. Results of the style-fusion algorithm

Figure 8. The left man and lady come from the FFHQ dataset [10]. The right four images combine the styles of the two with different degrees, where we see that (from left to right) the femininity is fading while the masculinity is increasing. And the corresponding style layer range are 0-6, 4-10, 8-14, 12-17, respectively.

Here the author verified the style-fusion algorithm proposed in section 3.4. The parameter of style layer range in the algorithm controls the part of latent codes of the man to be replaced with the latent codes of the lady correspondingly. The results vividly demonstrate that, in Figure 3, styles of low resolutions control high-level aspects like the pose and the face shape, while styles of high resolutions govern low-level aspects like the facial features, the eyes, and the color schemes.

5.4. Discussions

The success of the proposed style-fusion algorithm relied on the StyleGAN architecture that offers a disentangled latent space and the allowance for scale-specific modifications. In GAN inversion-based fine grained image manipulations, a disentangled latent space is crucial as it offers opportunities to find noninterference directions during the latent space navigation. Applications abound in related areas, including Image Restoration, Image Interpolation, Style Transfer, Compressive Sensing, 3D reconstruction, Image Understanding, and Multimodal Learning.

6. Conclusions

This paper aims to solve style-based image manipulation using StyleGAN. The StyleGAN is demonstrated to have a strong controllability over the image generated. Benefited from a disentangled latent space and the scale-specific manipulation of styles, we can project given images back onto their latent codes and do manipulations like the style-fusion described in this paper. Finer-grained image manipulations can then be hopefully performed on this disentangled latent space, which is one necessity for the latent space navigation in GAN inversion related works. The experiment on real-world images validates the effectiveness of our method. The author takes the StyleGAN-based GAN inversion as the future research direction.

References

[1] Silaparasetty, V. (2020). Image Manipulation. In: Deep Learning Projects Using TensorFlow 2. Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4842-5802-6_8

DOI: 10.54254/2755-2721/2/20220563

[2] Isola, P., Zhu, J. Y., Zhou, T., & Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1125-1134).

[3] Zhu, J. Y., Park, T., Isola, P., & Efros, A. A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. In Proceedings of the IEEE international conference on computer vision (pp. 2223-2232).

[4] Karras, T., Aittala, M., Hellsten, J., Laine, S., Lehtinen, J., & Aila, T. (2020). Training generative adversarial networks with limited data. Advances in Neural Information Processing Systems, 33, 12104-12114.

[5] Xia, W., Zhang, Y., Yang, Y., Xue, J. H., Zhou, B., & Yang, M. H. (2021). GAN inversion: A survey. arXiv preprint arXiv:2101.05278.

[6] Huang, X., & Belongie, S. (2017). Arbitrary style transfer in real-time with adaptive instance normalization. In Proceedings of the IEEE international conference on computer vision (pp. 1501-1510).

[7] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative adversarial nets. Advances in neural information processing systems, 27.

[8] Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., & Torralba, A. (2016). Learning deep features for discriminative localization. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2921-2929).

[9] Kim, J., Kim, M., Kang, H., & Lee, K. (2019). U-gat-it: Unsupervised generative attentional networks with adaptive layer-instance normalization for image-to-image translation. arXiv preprint arXiv:1907.10830.

[10] Karras, T., Laine, S., & Aila, T. (2019). A style-based generator architecture for generative adversarial networks. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 4401-4410).

[11] Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., & Aila, T. (2020). Analyzing and improving the image quality of stylegan. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 8110-8119).

[12] Karras, T., Aittala, M., Laine, S., Härkönen, E., Hellsten, J., Lehtinen, J., & Aila, T. (2021). Alias-free generative adversarial networks. Advances in Neural Information Processing Systems, 34.

[13] Xia, W., Zhang, Y., Yang, Y., Xue, J. H., Zhou, B., & Yang, M. H. (2021). GAN inversion: A survey. arXiv preprint arXiv:2101.05278.

[14] Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., & Hochreiter, S. (2017). Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, *30*.