Animation Figure Generation

Haodong Song 30920231154360, Huidong Feng 30920231154344, Yuntao Jin 30920231154351, Yuxin Lin 30920231154333, Wangzheng Shi 30920231154359

School of Informatics, Xiamen University
{ 30920231154360,30920231154344,30920231154351, linyx,30920231154359}@stu.xmu.edu.cn

Abstract

Due to the lack of pairing training data, current algorithms face challenges in directly generating anime faces. Our goal is to solve this problem by proposing a diffusion based animated face generation framework. But in addition to the diffusion algorithm, other existing vision models also perform very well in various tasks such as image generation. We select three additional algorithms VAE, GAN and StyleGAN2 to perform the same task, and analyze and compare the advantages and disadvantages of each method. Through the experiment, we analyzed the generated results from both quantitative and qualitative perspectives, and found that StyleGAN2 was the best among the four methods, while diffusion did not achieve the desired effect. In general, we have found a good way to generate high-quality animation faces based on the existing animation pictures.

1 Introduction

In the past, the acquisition of images entailed engaging a professional artist, where detailed specifications were provided, and subsequently, the artist would create a meticulously crafted image based on the provided instructions. Regrettably, this method suffered from inefficiency, labor intensity, error susceptibility, and was often unable to produce the intended results. Presently, image generation tools rooted in advanced image generation techniques have considerably streamlined the creation of a multitude of high-quality images. Notably, image generation techniques have perpetually constituted a focal point within the domain of computer vision, encompassing notable generative models such as Generative Adversarial Networks (GANs), Variational Auto-Encoders (VAEs), and flow models.

The Diffusion Model, initially introduced in 2015 in the seminal article titled "Deep Unsupervised Learning using Nonequilibrium Thermodynamics," addressed a significant challenge confronted by generative models like VAEs at the time. These models required the definition of both the conditional distribution and variant biases, necessitating simultaneous optimization, a process fraught with complexity. In contrast to the clarity and intuitive nature of GANs, the Diffusion Model did not receive widespread recognition upon its initial publication. However, in recent years, the landscape of generative models has witnessed rapid evolution, with cutting-edge text-to-image generation models such as Google's Imagen and OpenAI's DALL·E 2 being rooted in the architecture of the Diffusion Model.

Recent years have witnessed the demonstration of remarkable potential of diffusion models across various domains. These models have demonstrated the capacity to eliminate noise and generate high-resolution synthetic images with exceptional efficiency, surpassing the performance of GANs. A pivotal strength of diffusion models is their proficiency in generating high-quality images distinguished by attributes like clarity, contrast, and color fidelity. Furthermore, these models facilitate the generation of images characterized by specific textures, styles, and visual effects, underscoring their adaptability. This characteristic endows diffusion models with substantial utility across applications encompassing image recognition, computer vision, and natural language processing. Consequently, diffusion models represent a versatile tool for image generation, holding considerable research potential and underpinning diverse applications.

Although Gans have been widely used in the field of image generation before, this paper attempts to compare the changes of different algorithms on image generation capabilities, especially focusing on the use of anime faces as data sets, VAE, GAN, StyleGAN2 and DDPM to generate high-resolution anime faces with characteristics.

In summary, we took a dataset with 63,632 anime faces as input, trained under four methods respectively, and according to the qualitative and quantitative results of the experiment, we found the best method for generating anime face pictures among the four methods.

2 Relatedwork

2.1 Image Generation

Image generation task refers to the task of generating new images or image subsets using computer algorithms. This task is often used in image enhancement, image reconstruction, image generation, style transfer and other application scenarios.

In image generation tasks, we usually need to input an image as input and output a new image. This new image can be very different from the input image, or it can be based on the input image with some changes, such as color, texture, shape, and so on. At present, image generation task

has been widely studied and applied. With the development of deep learning technology, more and more researchers begin to apply deep learning method to image generation task, which also brings more solutions and possibilities for image generation task.

Current popular image generation models include: VAE: VAE is an unsupervised learning model that generates new images through the interaction between encoders and decoders. The encoder compresses the input image into a low-dimensional representation, and the decoder reconstructs that representation into a new image. This model can be used in image compression, image reconstruction and image generation. Generative adversarial network (GAN) : A GAN is a model consisting of a generator that tries to generate data similar to the real data and a discriminator that tries to distinguish the real data from the generated data. This model can be effectively trained to generate new images. However, it is prone to training instability and mode collapse. Diffusion models: This is a Markov chain-based model that can generate high-fidelity images from pure noise, but requires long sampling times and is sensitive to hyperparameters.

2.2 GAN

Generative Adversarial Network (GAN) is a generative model that learns through the game between two neural networks. Generative adversarial networks can learn generative tasks without using annotated data.

The generative adversarial network generally consists of a generator (generative network) and a discriminator (discriminant network). The generator generates something, inputs it into the discriminator, and then the discriminator determines whether the input is real data or generated by the machine. If the discriminator is not fooled, the generator continues to evolve, outputs the second generation Output, and then inputs the discriminator. The discriminator is also evolving at the same time, and has stricter requirements on the output of the generator. The generator and discriminator confront each other and continue to learn, and the two networks are alternately trained and their capabilities are improved synchronously until the data generated by the generated network can be fake and real, and reach a certain equilibrium with the ability of the discriminator network.

GAN adopts an unsupervised learning training mode, which can be widely used in unsupervised learning and semi-supervised learning. Compared with other models, GAN can produce clearer and more authentic samples. Better modeling of data distribution (sharper, clearer images)

But Gans also have some drawbacks: they are difficult to train and unstable. The Mode Collapse problem requires good synchronization between generators and discriminators. The learning process of GANs may have a pattern loss, and the generator starts to degenerate, always generating the same sample points, and cannot continue learning.

2.3 Diffusion

Today AIGC is mainly based on diffusion models, diffusion models are the new SOTA in depth generation models. And has excellent performance in many application fields, such as computer vision, NLP, waveform signal processing, multimodal modeling, molecular diagram modeling, time series modeling, adversarial purification, etc. In addition, diffusion models are closely related to other research areas, such as robust learning, representation learning, and reinforcement learning.

Diffusion Model is a kind of generation model, the principle of which is similar to denoising an image. By learning the process of denoising an image, you can understand how a meaningful image is generated. The diffusion model surpasses the original SOTA: GAN in the image generation task. By making the picture generated by the Generator as close as possible to the real picture , the GAN model achieves the purpose of being fake and real. In essence, it still generates new pictures that are close to the real picture, so the pictures generated by GAN may not have too many highlights. DDPM, on the other hand, fits the whole process from real picture to random Gaussian noise, and then generates new pictures through the reverse process, which is essentially different from GAN.

Therefore, compared with GAN model, the images generated by diffusion model are more accurate and more in line with human visual and aesthetic logic. Meanwhile, with the accumulation of sample number and deep learning time, diffusion model shows better imitation ability of artistic expression style.

However, the original diffusion model also has disadvantages, its sampling speed is slow, often requiring thousands of evaluation steps to extract a sample; Its maximum likelihood estimation cannot be compared with the model based on likelihood; Its ability to generalize to various data types is poor.

3 Method

In order to complete the task of generating animation avatars, we adopt a variety of algorithms with excellent performance in image generation tasks to try to solve them. We will introduce VAE in section 3.1, stylegan in section 3.2 and diffusion in section 3.3. Through the analysis and comparison of the three methods, we find the best method to solve the problem. In order to complete the task of generating animation avatar, we adopt a variety of algorithms with excellent performance in image generation tasks to try to solve the problem. We will introduce VAE in section 3.1 and stylegan in section 3.2. section 3.3 introduces diffusion. Through the analysis and comparison of the three methods, the best method to solve the problem is obtained.

3.1 VAE

Variational Autoencoder (VAE) is a variant of the autoencoder,Instead of mapping the input to a fixed encoding in the hidden space, the variational self-encoder is converted into an estimate of a probability distribution over the hidden space; for ease of representation we assume that the prior distribution is a standard Gaussian distribution. Similarly, we train a probabilistic decoder modeling the mapping from the distribution in the hidden space to the real data distribution. When given an input, we estimate the parameters about

the distribution (the mean and covariance of the multivariate Gaussian model) from the posterior distribution and sample over this distribution, which can be made derivable (as a random variable) using a reparameterization trick, and finally output the distribution about it through the probabilistic decoder as shown in Figure 1. In order to make the generated image as realistic as possible, we need to solve the posterior distribution with the goal of maximizing the log-likelihood of the true image.

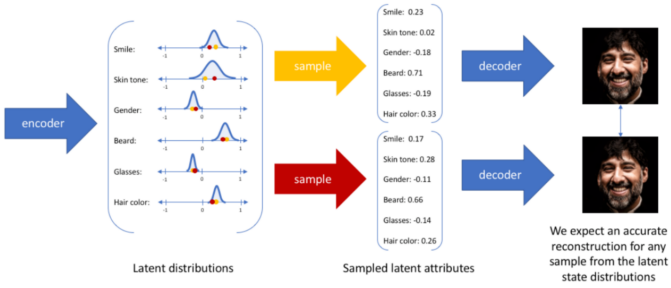

Figure 1: Sample generation process of the variational self-encoder

The true posterior distribution is not directly solvable according to the Bayesian model containing an integral over a continuous space of pairs. To solve the above problem, the variational autocoder uses variational inference by introducing a learnable probabilistic encoder to approximate the true posterior distribution, and using the KL scatter measure to measure the difference between the two distributions, transforming this problem from solving the true posterior distribution to how to reduce the distance between the two distributions.

In synthesis, the variational process described above is the core idea of the VAE and its various variants, whereby the problem is transformed into a lower bound of evidence that maximizes the generation of real data through variational reasoning.

3.2 StyleGAN

Since the introduction of GANs, rapid advancements have been witnessed in the field of technology, with applications spanning various domains. GAN technology has been successfully implemented in areas such as image and video generation, data simulation and augmentation, diverse image stylization tasks, facial and body image editing, as well as image quality enhancement. StyleGAN stands out as a cutting-edge, high-quality image generator. Recognized as a powerful framework for controlling the attributes of generated images, StyleGAN introduces the concept of a style space, allowing users to manipulate styles at different levels during image generation, thereby achieving more flexible and diverse image outcomes. The progressive resolution enhancement strategy of StyleGAN enables the generation of facial images at resolutions up to 1024×1024, with precise control and editing capabilities for attributes.

The comparison between StyleGAN and the structure of traditional image generation models is illustrated in Figure 2. The core of StyleGAN2 lies in its Style Modulation Layers, from which its name is derived. These layers enable the generation of high-quality image data while achieving controllability over high-level features.

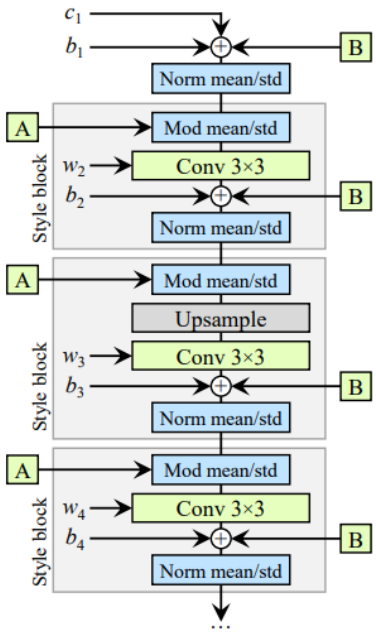

Figure 2: StyleGAN2 architecture

As StyleGAN gained widespread usage, inherent drawbacks of the model became apparent, such as the issue of artifacts, including pseudo-shadows. Additionally, researchers observed a phenomenon known as texture sticking, where certain attributes in generated images, such as teeth or eyes, exhibited pronounced spatial biases that were challenging to address even through latent space interpolation.In subsequent research identified the sources of pseudo-shadows and redesigned the algorithm to enhance the network. StyleGAN2 addresses the pseudo-shadow problem of StyleGAN, leading to the generation of higher-quality image.

3.3 Diffusion

3.3.1 Diffusion Model

The Diffusion Model is a probabilistic model used for image generation. Introduced in 2015 but initially overlooked, the model gained prominence with the introduction of the Denoising Diffusion Probabilistic Model (DDPM) in 2020, leading to a surge in the popularity of generative models. The core of the diffusion model is a random walk process, where each pixel's value gradually diffuses to its surrounding pixels. This process involves calculating the diffusion speed of each pixel based on differences with adjacent pixels and updating them accordingly. By controlling the diffusion speed and the number of iterations, the effect of the generated image can be adjusted.

Inspired by non-equilibrium thermodynamics, the diffusion model now produces the most advanced image quality,

with examples as follows:

Figure 3: Diffusion Model Generation Examples

In terms of training efficiency, diffusion models boast added advantages in scalability and parallelization. The fundamental principle of the Diffusion model can be explained through probability theory and stochastic process theory. Specifically, the diffusion and reverse diffusion processes of the Diffusion model are akin to a random walk process, where the variance of random noise decreases over time, facilitating gradual image generation. The Diffusion model can also be trained using optimization algorithms like gradient descent. This allows the generator to adapt to various data distributions and generate high-quality images. After training, randomly sampled noise is input into the model and then denoised to obtain the corresponding data.The entire process is shown in Figure 4.

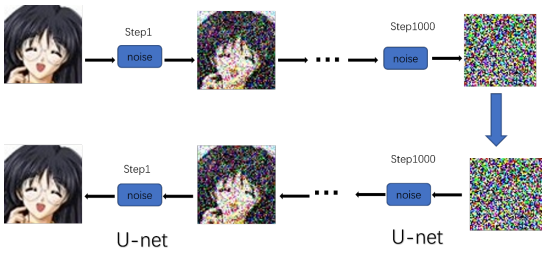

Figure 4: Diffusion Model Working Principle

More precisely, the diffusion model utilizes a Markov Chain (MC) corresponding to an implicit variable model in latent space. Through the Markov Chain, noise is gradually added to the data x at each time step t to obtain the posterior probability $q(x_(1 : T)x_0)$, where x_1,...,x_T represent the input data and also constitute the latent space.

Diffusion Models are divided into a forward diffusion process in Figure 5 and a reverse reverse diffusion process in Figure 6. The following diagram exemplifies the diffusion process, where the transition from x_0 to the final x_T exhibits Markov Chain properties, transitioning from one state to another independent of the previous state. The subscript denotes the corresponding diffusion process in the diffusion model.

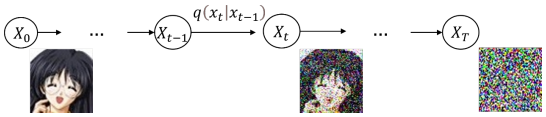

Figure 5: Diffusion Process

Finally, in the Diffusion Models, a real image input as x_0 is gradually transformed into an image of pure Gaussian noise x_T. The primary training objective of the diffusion model focuses on the reverse process, learning to generate new images from pure Gaussian noise and learning the posterior probability of the forward process: specifically, training the probability distribution $p_{(x_(t-1)x_t))}$. By traversing backwards along the Markov Chain, new data x_0 can be regenerated.

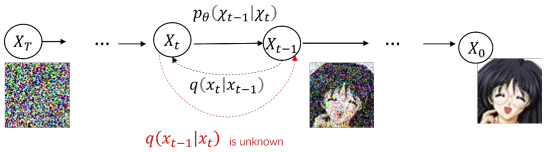

Figure 6: reverse diffusion process

3.3.2 Loss function of Diffusion Model

Through experience, found that training diffusion models with a simplified objective that ignores the weighting terms yields better results

Algorithm 1: Training

1: **repeat**
2: $\mathbf{x_0} \sim q(\mathbf{x_0})$
3: $t \sim \text{Uniform}(\{1, \ldots, T\})$
4: $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
5: Take gradient descent step on
$\nabla_\theta \left\| \epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, t) \right\|^2$
6: **until** converged

Algorithm 2: Sampling

1: $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
2: **for** $t = T$ to 1 **do**
3: $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\mathbf{z} = \mathbf{0}$
4: $\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}} \epsilon_\theta(\mathbf{x}_t, t) \right) + \sigma_t \mathbf{z}$
5: **end for**
6: **return** x_0

3.3.3 Denoising network architecture

For the reverse process in diffusion models, the parameterization/model structure of Gaussian distribution is chosen. The diffusion model offers considerable flexibility, with the only requirement for our architecture being that its input

and output have the same dimensions. Considering this constraint, image diffusion models typically employ architectures similar to U-Net. U-Net enables multi-level analysis and abstraction of the input image, capturing detailed information more effectively. This architecture allows the network to learn both local and global features of the image and combine them for more accurate segmentation results.

4 Experiments

We evaluate the performance of above three models using AnimeFace dataset. The dataset has 63,632 anime face images. We randomly split 80% as the training set and 20% as the test set. All models are trained on the train set and evaluated on the test set.

4.1 Implementation Details

The anime face generation experiments were conducted on the AnimeFace dataset. Since the resolution of the images in the dataset is not uniform, we first resize all the images to 64x64 pixels. The implementation of the three models is based on PyTorch. We use Adam optimizer with the learning rate of 0.005. It takes about 30 minutes to train the Vanilla-VAE with the batch size of 64, about 14 hours to train the Vanilla-GAN with the batch size of 64, about 12 hours to train the StyleGAN2 with the batch size of 32 and about 3 day to train the DDPM model with the batch size of 16 on a single RTX3090 GPU.

4.2 Method Comparison

4.2.1 Quantitative Comparison

We numerically compared the three models in Table 1. We use the Fréchet inception distance (FID) to quantify the similarity of generated images and real images. We found that the DDPM model achieved the best result on FID.It is worth noting that the principle of GAN and StyleGAN2 is similar. By changing the network structure, such high performance can be improved, and the structure of DDPM can be reasonably predicted and appropriately modified, and the capability of the original network can also be optimized.

Method	FID
Vanilla-VAE	198.35
Vanilla-GAN	222.64
StyleGAN2	**4.17**
DDPM	19.07

Table 1: It shows the results of the various methods

4.2.2 Qualitative Comparison

We use the same parameters to generate the same number of avatars for the trained model, and compare the generated effect according to the naked eye. We found that as the FID went from high to low, the resolution of the image also increased and the details became more perfect. Among them, stylegan2 results are the best, maintaining a high degree of agreement with the original data set.

Figure 7: result of VAE**(left)** and GAN**(right)**

In Figure 7,The original VAE and GAN are about 200 in FID index. From the perspective of qualitative visualization, the results generated by VAE generally meet the characteristics of animation, but the overall picture is fuzzy and many details are lost, such as eye color and hair separation. On the contrary to VAE, GAN produces better results in detail, and the characteristics of different animation avatars can be reflected. On the macro level, there is more random noise, which proves that the learning of the model is not perfect.

Figure 8: result of StyleGAN2**(left)** and DDPM**(right)**

In Figure 8,StyleGAN2, which has the lowest FID index, performs the best on the animation face generation task, and gets the best results in detail and general; In addition to the face part, it can even understand and generate the corresponding dress up, and each avatar has obvious characteristics and can be applied in practice; diffusion is indeed similar to StyleGAN2 in all aspects, but the pixels in the mouth and eyes are occasionally distorted, which greatly reduces the production effect.

5 Conclusion

In this paper, how to generate animation avatar problem, we start from the image generation, using the current excellent image generation algorithm to solve the problem; We used VAE, stylegan and diffusion methods. At the same time, using the same data set as input, we tested the performance of these methods by both quantitative and qualitative criteria, and finally found that stylegan2 was the best in both criteria. This is contrary to the best performance of conventional diffusion, and we assume that we need to modify the basic diffusion framework similar to stylegan2 to get better results, which is what we plan to do later.

Reference

Sohl-Dickstein J, Weiss E, Maheswaranathan N, et al. Deep unsupervised learning using nonequilibrium thermodynamics[C]/International Conference on Machine Learning. PMLR, 2015: 2256-2265.

Ho J, Jain A, Abbeel P. Denoising diffusion probabilistic models[J]. Advances in Neural Information Processing Systems, 2020, 33: 6840-6851.

Yang L, Zhang Z, Song Y, et al. Diffusion models: A comprehensive survey of methods and applications[J]. arXiv preprint arXiv:2209.00796, 2022.

Cao H, Tan C, Gao Z, et al. A survey on generative diffusion model[J]. arXiv preprint arXiv:2209.02646, 2022.

Wang Z, Zheng H, He P, et al. Diffusion-gan: Training gans with diffusion[J]. arXiv preprint arXiv:2206.02262, 2022.

He Z, Sun T, Wang K, et al. DiffusionBERT: Improving Generative Masked Language Models with Diffusion Models[J]. arXiv preprint arXiv:2211.15029, 2022.

Mirza, Mehdi, and Simon Osindero. "Conditional generative adversarial nets." arXiv preprint arXiv:1411.1784 (2014).

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

Arjovsky, Martín et al. "Wasserstein GAN." ArXiv abs/1701.07875 (2017): n. pag.

Karras, Tero, Samuli Laine, and Timo Aila. "A style-based generator architecture for generative adversarial networks." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019.

Karras, Tero, et al. "Analyzing and improving the image quality of stylegan." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020.

Karras, Tero, et al. "Training generative adversarial networks with limited data." Advances in neural information processing systems 33 (2020): 12104-12114.

Kingma D P, Welling M. Auto-encoding variational bayes[J]. arXiv preprint arXiv:1312.6114, 2013.

Doersch C. Tutorial on variational autoencoders[J]. arXiv preprint arXiv:1606.05908, 2016.

Kingma D P, Dhariwal P. Glow: Generative flow with invertible 1x1 convolutions[J]. Advances in neural information processing systems, 2018, 31.

Yang L, Zhang Z, Song Y, et al. Diffusion models: A comprehensive survey of methods and applications[J]. arXiv preprint arXiv:2209.00796, 2022.

Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation[C]/Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18. Springer International Publishing, 2015: 234-241.

Zhao Z, Singh S, Lee H, et al. Improved consistency regularization for gans[C]/Proceedings of the AAAI conference on artificial intelligence. 2021, 35(12): 11033-11041.

Lu, Cheng, et al. "Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps." arXiv preprint arXiv:2206.00927 (2022)

Shruti Agarwal, Hany Farid, Yuming Gu, Mingming He, Koki Nagano, and Hao Li. Protecting world leaders against deep fakes. In Proceedings ofthe IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, June 2019.

Shane Barratt and Rishi Sharma. A note on the inception score. arXiv:1801.01973, 2018.

Andrew Brock, Theodore Lim, J. M. Ritchie, and Nick Weston. Neural photo editing with introspective adversarial networks. arXiv:1609.07093, 2016.

Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. arXiv:1809.11096, 2018.

Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In International Conference on Machine Learning, pages 1691–1703. PMLR, 2020.

Nanxin Chen, Yu Zhang, Heiga Zen, Ron J. Weiss, Mohammad Norouzi, and William Chan. Wavegrad: Estimating gradients for waveform generation. arXiv:2009.00713, 2020.

Rewon Child. Very deep vaes generalize autoregressive models and can outperform them on images. arXiv:2011.10650, 2021.

Peter Dayan, Geoffrey E Hinton, Radford M Neal, and Richard S Zemel. The helmholtz machine. Neural computation, 7(5):889–904, 1995.

Terrance DeVries, Michal Drozdzal, and Graham W. Taylor. Instance selection for gans. arXiv:2007.15255, 2020.

Prafulla Dhariwal, Heewoo Jun, Christine Payne, Jong Wook Kim, Alec Radford, and Ilya Sutskever. Jukebox: A generative model for music. arXiv:2005.00341, 2020.

Jeff Donahue and Karen Simonyan. Large scale adversarial representation learning. arXiv:1907.02544, 2019.

Yilun Du and Igor Mordatch. Implicit generation and generalization in energy-based models. arXiv:1903.08689, 2019.

Vincent Dumoulin, Jonathon Shlens, and Manjunath Kudlur. A learned representation for artistic style. arXiv:1610.07629, 2017.

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

Karras T, Laine S, Aittala M, et al. Analyzing and improving the image quality of stylegan[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020: 8110-8119.

Karras T, Aittala M, Laine S, et al. Alias-free generative adversarial networks[J]. Advances in Neural Information Processing Systems, 2021, 34: 852-863.