Research on Image Generation Methods Based on Adversarial Neural Networks

Renjie Ding

Shanghai Maritime University, 1550 Harbour Avenue, Shanghai, 200135, China

wakwac@foxmail.com

Abstract. Image generation has become a heated research topic in recent years owing to its wide landing scenes and great potential in all walks of life. Especially after the emergence of adversarial neural networks, both the training process and results have been greatly improved compared with previous model methods. This paper focuses on the advantages of directly using Generative Adversarial Nets (GAN) to generate images, as well as its main problems: training instability, pattern collapse, and global correlation, and introduces the strategies and skills of subsequent improved GAN for these problems. Through experiments, we compare the improved network with the original GAN and try to combine the core strategies of these networks. In the experiment, the image quality generated by the combined network is higher.

Keywords: image generation, adversarial neural networks, WGAN, SAGAN.

1. Introduction

With the rise of deep learning, computer vision and image processing have ushered in a new chapter. Image classification, target detection, image segmentation, and other algorithms have become mature, a large number of projects have been completed, and good results have been achieved. If artificial intelligence can create images automatically, it means its “wisdom” will go further. As a basic technology, image generation has a wide range of applications, such as image restoration, image super-resolution, image denoising, etc.

Variational Autoencoders (VAEs) [1], Deep Belief Networks (DBNs) [2], and Autoregressive (AR) Models [3] are influential in the generation model but are very difficult to learn due to the high dimension of the modeled random variables, which are mainly reflected in the challenge of statistical injury and computational injury. The challenge of statistics is that these generation models cannot well generalize the generated results. The challenge of calculation is mainly from the inference of execution of difficult solutions and the normalized distribution, and the adversarial neural network can effectively avoid these problems. The method using generative adversarial nets (GAN) [4] can be divided into three categories: direct method, iterative method, and hierarchical method. Compared with hierarchical and iterative methods, direct design and implementation are relatively simple and direct and usually get good results. In this paper, the direct method is applied to generate images to explore the problems of the original GAN and the improvement scheme.

After GAN was proposed, although the results were impressive, also it had some shortcomings, such as unstable training, mode collapse, poor quality of generated images, and lack of interpretability. Then, in view of these problems, various improved versions of GAN also appeared. This paper mainly focuses on the two models Wasserstein GAN (WGAN) and Self-Attention GAN (SAGAN). WGAN introduces the bulldozer distance by modifying the loss function to solve the training stability problem, while SAGAN introduces the attention mechanism into GAN to solve the locality of the CNN receptive field and enhance the understanding of high-resolution features and the long-distance correlation of various features.

This article is divided into five sections. Section 2 reviews the classic literature and states the proposal and development of GAN. Section 3 focuses on model selection and structural principles. Section 4 is mainly about the application, and Section 5 is the summary and future work.

2. Review

2.1. Methods based on traditional models

As an unsupervised learning algorithm, the automatic encoder was first proposed in 1986 [5]. It is often used for feature extraction and dimension reduction of high-dimensional data. This concept promotes the development of neural networks. It applies to deep learning and can be used to determine the initial value of weight matrix before training. In supervised learning, the neural network requires labeled data. However, in fact, the neural network can also deal with unlabeled data. The automatic encoder is an algorithm that uses the neural network to deal with unlabeled data. The automatic encoder uses the back propagation algorithm, and the neural network uses large-scale data sets to train the encoder. Finally, the input and output of the automatic encoder are equal. The automatic encoder is divided into encoding and decoding. After the input data are encoded, the code is obtained, and then after the decoding module is processed, the output result is finally obtained.

By controlling the output dimension of coding, the encoder network can be forced to learn high-dimensional data features with low dimensions in the decoding process. VAE makes the potential vector of image coding subject to Gaussian distribution on the basis of autoencoder to achieve image generation and optimizes the lower bound of log-likelihood of data. VAE is parallel in image generation. It still has many shortcomings. Firstly, it only learns the inference network for one problem. Secondly, the samples from VAE trained on images are often vague, resulting in some challenges in optimization.

2.2. Methods based on the autoregressive model

The autoregressive (AR) model creates an explicit density model, which is easy to handle to maximize the possibility of training data (manageable density). Therefore, this can help us calculate the likelihood of data observations more easily and obtain evaluation metrics for generating models. Autoregression is a practical method, which provides explicit modeling of the likelihood function. However, to model data with multiple-dimension features, autoregressive models need some additional improvements.

PixelCNN was proposed by Deepmind [6] in 2016, which opened the most promising autoregression generation model family. Since then, it has been used to generate voice, video, and high-resolution images. Owning to convolution operation, the distribution of every pixel in the image can be learned in parallel. However, as for calculating the probability of some specific pixels, the receptive field of the normal convolution layer violates the order prediction of the autoregressive model. When processing information about a centre pixel, all the pixels around it will be considered by the convolution filter in order to get the output feature map, not just the previous pixels. So, masks are needed to stop information flow from unpredicted pixels. Masks can be completed by zeroing all unimportant pixels that should not be considered. In our implementation, a mask with the same size as the convolution filter and values of 1 and 0 is created. Before the convolution kernals, this mask is multiplied by the weight tensor. In PixelCNN, there are two types of masks. The one applies only to the first convolution layer. It limits access to pixels by zeroing the center pixels in the mask. After that, we can ensure that the model does not access the pixels it will predict. The other type of mask is used in all subsequent convolution layers, and the limitation is relaxed by allowing connections from pixels to itself. explaining prediction of pixels of the first layer.

2.3. Methods based on adversarial neural networks

The methods of image generation using GAN are mainly divided into three categories: direct method, hierarchical method, and iterative method. The core of these three methods is the confrontation network, but the structures are different. The direct method follows the principle of using a generator and a discriminator in its model, and the structures of generators and discriminators are direct and have no branches. Many earliest GAN models belong to this category, such as GAN, DCGAN [7], ImprovedGAN [8], InfoGAN [9], f-GAN [10], and GANINT-CLS [11]. Among them, DCGAN is one of the most classic, and its structure is used by many later models. The overall structure of DCGAN is similar to that of GAN, but the convolutional neural network (CNN) is introduced into the network for the first time. Both the discriminator and generator of DCGAN use CNN to replace the multi-layer perceptron in GAN, and batch normalization operations are added to both discriminator and generator to solve the problem of poor initialization. DCGAN has better performance mainly due to its strong feature extraction ability of the convolution layer.

Recently, SS-GAN [12] uses two GANs for processing random noise, and LAPGAN [13] uses the Laplacian pyramid to generate images from coarse to fine by iterative methods. StackGAN [14] is an iterative method with only a two-layer generator. SGAN [15] is another iterative method, whose stack generator takes lower-level features as input and outputs higher-level features, while the bottom generator takes noise vector as input and the top generator outputs images. This paper focuses on the performance of GAN networks in the scene of image generation. Because the structure of the direct method is simple, it is selected for image generation. The initial core confrontation thought has encountered many problems after it was proposed, mainly training problems, pattern collapse, and image quality. Therefore, GAN is developed by a combination of attention mechanism and GAN (self-attention GAN) [16]. The introduction of the attention mechanism makes GAN's modeling more long-term correlation, accelerates the training of the regularization discriminator while accelerating the training, and has better performance on image classification.

In summary, compared with the deep learning GAN network, the traditional methods based on texture and structure lack feature acquisition, and the overall semantic information is also difficult to grasp. Under the condition of new computational support, the GAN network has been widely used in many fields. Therefore, this paper mainly studies the image restoration method based on the GAN network.

3. Methodology

3.1. problems

Throughout the development of GAN, the main problems can be divided into three categories: training stability, pattern collapse, and global connectivity. This paper mainly discusses two models: WGAN-GP [17] and self-attention GAN. Both of them enhance the stability of training to a certain extent. The difference is that in addition to the stability of training, the former also has a significant improvement in solving the problem of mode collapse, while the latter focuses on the global connection.

The solution of WGAN-GP for training instability is that the weight truncation satisfies Lipschitz continuity, abandons the previous momentum-based optimization algorithm (momentum and Adam), replaces it with other SGDs, and slightly modifies the activation function (remove sigmoid) and loss (remove log). In the mode collapse problem, by adding the Wasserstein distance to the loss function and adding the Lipschitz constraint as a regular term to the Wasserstein loss, the GAN loss is optimized and the Lipschitz constraint is satisfied as much as possible.

Self-attention GAN uses weight spectrum normalization and Two Time-Scale Update Rule to improve training instability. At the same time, the attention mechanism is added to the GAN network, and the long dependence is established, which effectively compensates for the weakness of CNN in overall control. Each part of the final output is not only related to the corresponding receptive field, but also related to the overall global.

This paper will do experimental research on the two models, analyze the principal structure and advantages and disadvantages of the two models, compare the experimental results, and try to find a method to integrate the advantages of the two models.

3.2. Model Selection

The reason for choosing WGAN-GP and SAGAN is that they have a good performance in solving the problem of training instability, and they put forward targeted skills in dealing with the collapse of the mode and long dependence.

3.2.1. WGAN & WGAN-GP. WGAN-GP is upgraded from WGAN [18], so its main structure is WGAN. Like all confrontation networks, WGAN is composed of a discriminator and a generator. The main parts of the two inherit the convolution layer and batch normalization operation in DCGAN and better adapt to image data input. The loss function is greatly changed, which is the core idea of WGAN. The loss function (Eq. 1) derived from cross-entropy in the original GAN is changed to Eq. 2, which can be used to measure the Wasserstein distance between Pr and Pg.

$$\mathrm{min_G max_D V(D, G)} = \mathrm{E_{x\sim P_{data}}[logD(x)]} + \mathrm{E_{z\sim p_z(z)}}\left[\log\left(1 - \mathrm{D}\left(\mathrm{G(z)}\right)\right)\right] \tag{1}$$

$$\mathrm{W(P_r, P_\theta)} = \mathrm{E_{x\sim P_r}[D(x)]} - \mathrm{E_{x\sim P_g}[D(x)]} \tag{2}$$

D(x), G(x) are the discriminator and generator respectively. Through gradient descent, the loss function of the judge is maximized and that of the generator is minimized. However, if the difference in the Eq. continues to grow to infinity, the loss function is difficult to converge and cannot be properly trained. Therefore, the author of WGAN proposed that the generator model needs to meet the 1-lipschitz condition, and the loss function is limited to a smooth curve to avoid excessive or small output values. In the specific implementation part, WGAN simply adds a constant c to limit the parameter value to be too large, which is called weight clipping. The improved function is shown in Eq.3.

$$\mathrm{W}\left(\mathrm{P_r, P_g}\right) = \mathrm{sup}_{||f||_L\leq 1}\mathrm{E_{x\sim P_r}[D(x)]} - \mathrm{E_{x\sim P_g}[D(x)]} \tag{3}$$

$||f||_L \leq 1$ is the condition of clipping and the function also meets the condition of Lipschitz. Since the idea of approximate fitting Wasserstein distance is used, the discriminant also changes from the true-false binary classification problem of the original GAN to the regression problem, so accordingly, the sigmoid function of the last layer of the discriminant is also removed. Furthermore, on the basis of WGAN, the upgraded version of WGAN-GP appeared, which mainly changed the clipping operation. The gradient penalty term was added to the loss function, and the points that did not meet the Lipschitz condition were sampled and punished as shown in Eq. 4.

$$\mathrm{L} = \mathrm{E_{x\sim P_g}[D(x)]} - \mathrm{E_{x\sim P_r}[D(x)]} + \lambda\mathrm{E_{x\sim P_x}}\left[\left(\left|\left|\nabla_x \mathrm{D(x)}\right|\right|_2 - 1\right)^2\right] \tag{4}$$

The second part is a penalty term, and λ is a penalty coefficient. The 1-lipschitz condition was no longer simulated as simple as WGAN, but the restriction condition was truly and accurately restored. To a certain extent, the problem of gradient disappearance in the multi-layer neural network caused by improper adjustment of weight clipping parameters in WGAN was solved.

The loss function in the original GAN is equivalent to the js divergence. After the original author conducted "-logD trick", the discriminator remained unchanged, and the generator loss function was equivalent to KL divergence. No matter which method is used, the problems of js divergence and KL divergence cannot be avoided: when the two data distributions are not overlapped, the js divergence is a constant, which will cause the disappearance of the gradient to be unable to train normally, and this is very easy to be encountered in the initial stage. Moreover, the asymmetry of KL divergence and the contradiction between the variation trend of KL divergence and js divergence will cause the collapse of the model. The Wasserstein distance can reflect the distance between the two non-intersection distributions. By decreasing and increasing the gradient of the loss function, the fitting and far-off of the two distributions can be achieved, and the 1-lipschitz condition can provide meaningful gradients to solve the mutation problem of KL and JS divergence.

3.2.2. SAGAN. SAGAN is mainly divided into one core structure and two skills. The core structure is to add the self-attention mechanism to the neural network and generate three feature maps by three 1 * 1 convolutions of the input feature map, two of which are transposed and multiplied. The purpose is to obtain the correlation between each pixel of the two feature maps and obtain a correlation matrix. Finally, the third feature map that is not used before is multiplied by this correlation matrix to obtain a new feature map with attention. In addition, two training techniques, spectral normalization, and imbalanced learning rate also appear in the original paper. Spectral normalization limits the spectral norm of each layer to constrain the Lipschitz constant of the discriminator, while unbalanced learning rates allow the generator and the decision maker to train at different learning rates. It is worth mentioning that the use of spectral normalization here is somewhat different from the original SN: 1. The original spectral normalization is based on the W-GAN theory, only used in the decision maker to constrain the decision maker function to be 1-Lipschitz continuous. In Self-Attention GAN, Spectral Normalization appears in both decision and generator to make the gradient more stable. In addition to the last layer of the generator and discriminator, each convolution or deconvolution unit has SpectralNorm. 2. When spectral normalization is used on the generator, BatchNorm is retained. There is no BatchNorm on the judge, only SpectralNorm.

3.2.3. Advantages. The previous GAN model and the DCGAN proposed later largely depend on convolution to simulate the dependence between various image regions. Convolution operation is based on a local receptive field, and the long-distance correlation in the image can only be processed after several convolution layers. This learning long-term correlation problem may not be represented in convolutional networks because it is not easy to capture multiple layer correlation parameters in the model optimization phase and these parameters may be statistically significant, which brings some problems. Increasing the kernel size can improve the representation ability of the network, but this will also lose the computational and statistical efficiency obtained by using the old convolution structure. Compared with the original GAN and the DCGAN proposed later, the SAGAN with the attention mechanism can make the extracted features have long dependence and get the global geometric features of the image step by step. The fine details of each position can be carefully coordinated with the fine details of the image in the distance.

In terms of training techniques, the starting point of SAGAN's spectral normalization method is also used to satisfy the Lipschitz constraint and discard the original KL and JS divergence, which is similar to the idea of WGAN. However, compared with other regularization techniques, spectral norm regularization does not require additional hyperparameters (setting the spectral norm of the ownership layer to 1 performs well in practice). In addition, the calculation cost is relatively low. It is found that spectral norm regularization can also be used to generate networks, which can prevent the large parameter values of the generated networks and the abnormal gradient, resulting in unstable training. Moreover, spectral norm regularization is used in both discriminator and generator networks. In the training process, the discriminator needs less update times (eliminating numerical oscillation and faster convergence), which improves the training efficiency. Another SAGAN technique, TTUR, is used to solve the problem of slow learning in regularized discriminators. In the original GAN, some techniques are used for training, such as training several discriminators and then training the generator again. TTUR makes it possible to use fewer generator steps for each discriminator step by distinguishing the learning rate of the generator decisioner from the individual training. Using this method, we can produce better results in the same unit of time.

3.3. model comparison

On the basis of GAN [4], WGAN [18] and SAGAN modify the loss function and model structure. The comparison of the three models is shown in Figure 1.

Accordingly, the three models address different problems, which are summarized in Table 1. WGAN and WGAN-GP use Wasserstein distance to deal with model collapse while SAGAN uses the attention mechanism to enhance long-distance correlation. As for the problem of training, Wasserstein distance and gradient penalty used in WGAN and WGAN-GP are effective while SAGAN adds spectrum normalization and TTUR into the model so that the training stability is improved.

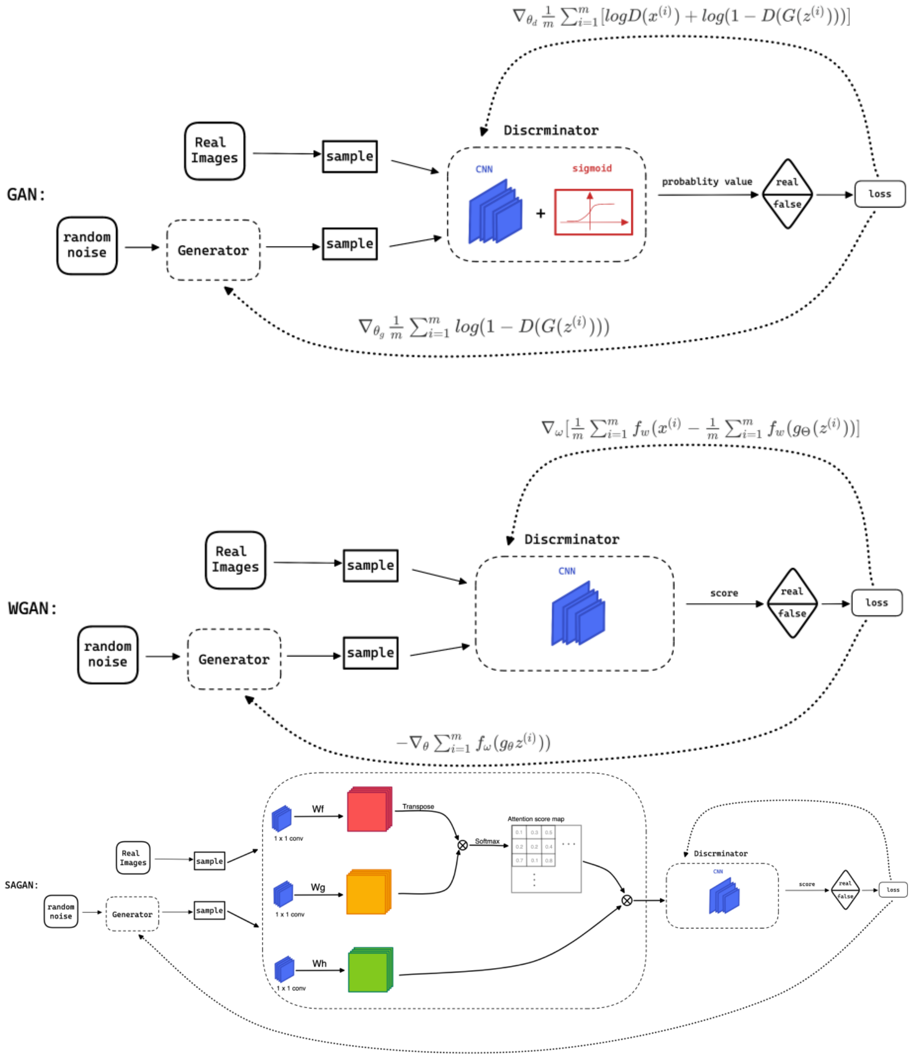


Figure 1. The comparison of three model structures. Compared with the original GAN, the loss function of WGAN is modified to measure Wasserstein distance. The other obvious change is that the sigmoid in GAN is deleted in WGAN in order to solve the problem of convergence. On the other hand, the attention mechanism is added to the network in SAGAN, and the input feature map is divided into three parts, each of which calculates the Hadamard product with others and puts the combined result into the discriminator as the normal GAN structure.

Table 1: different model aims at different problems. WGAN and WGAN-GP try to solve the model collapse while SAGAN tries to do with picture quality by strengthening long-distance correlation between different parts of a picture. And both models also want to improve training stability which is a common and significant issue in all deep learning topics.

	Training stability	Model collapse	long-distance correlation
WGAN& WGAN-GP	Wasserstein distance, gradient penalty	Wasserstein distance	/
SAGAN	spectrum normalization, TTUR	/	Attention

4. Application

4.1. The dataset of CelebA

Celeba is a popular image dataset which is widely used. It contains over twenty thousand face images of over ten thousand celebrities. Each image has a feature tag, including a annotation part, five face feature points coordinates, and 40 attribute tags. CelebA is open to the Chinese University of Hong Kong and is used in face-related computer vision training tasks. It can be applied in various tasks of face images processing like identification, detection, and landmark marking. In this experiment, I selected 8,000 218 * 178 face images for the training set for generating networks.

4.2. Experiment and Optimization

I built WGAN, WGAN-GP, and SAGAN through PyTorch. Since the dataset size was 218 * 178 and the size of the input image in the original model was 28 * 28, the input and output size of the model was adjusted to 200 * 200.

4.3. Analysis and evaluation of experimental results

The following four images shown in Figure 2 are the results of the best epochs from each model respectively. It can be seen that WGAN-GP solves the problem of pattern collapse better than WGAN, and SAGAN gets the best image quality.

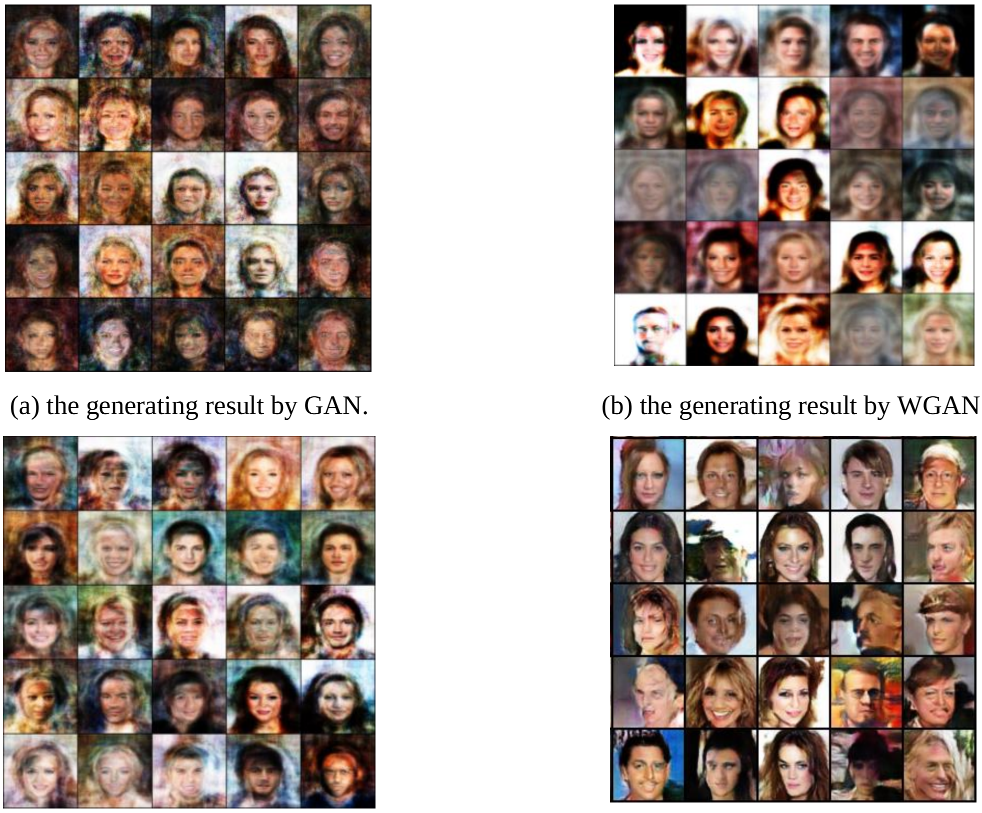
In addition to the subjective evaluation, we need some objective indicators to evaluate the quality of generated images. There are two methods: Inception Score and Fréchet Inception Distance. Besides, convergences of loss functions are also shown in this section as a reference.

4.3.1. Inception Score (IS) [8]. Inception Score uses the image category classifier to evaluate the quality of generated images. The image category classifier used is Inception Net-V3. This is also the origin of the name Inception Score. IS indicators are mainly used to evaluate the clarity and diversity of generated images.

Definition: IS generates a 1000-dimensional vector y for the generated image x entered into Inception Net-V3. Each dimension represents the probability of a certain type of data. For clear images, one dimension of y should be close to 1, and its dimension should be close to 0. That is, for the category y, the entropy of p(y|x) is small (probability comparison is certain). The calculation method is shown in Eq. 5.

$$\mathrm{IS(G)} = \exp\,(\mathrm{E}_{\mathrm{x}\sim \mathrm{p_g}} \mathrm{D_{KL}} \left(\mathrm{p(y|x)} \middle\| \mathrm{p(y)}\right)\Big) \tag{5}$$

$\mathrm{D_{KL}}$ is the KL divergence of two distributions. The larger the IS value is, the higher the quality of the generated image will be. For all generated images, they should be evenly distributed across all categories. For example, a total of 10,000 images are generated. For 1,000 categories, 10 images should be generated for each category. The entropy of $\mathrm{p(y)}=\sum \mathrm{p(y|x^i)}$ is very large, and the overall distribution is close to a uniform distribution.



(a) the generating result by GAN.

(b) the generating result by WGAN.

(c) the generating result by WGAN-GP.

(d) the generating result by SAGAN.

Figure 2. The images are generated by GAN, WGAN, WGAN-GP, and SAGAN through training 8,000 pictures from CelebA datasets shown in (a)(b)(c)(d). The first 5*5 generated pictures in each epoch of each model will be saved. And the epoch with subjectively high-quality pictures is shown in this figure.

4.3.2. FID (Fréchet Inception Distance) [20]. Considering the distance between the generated data and the real data at the feature level directly, the classifier is no longer used. So, to measure the distance between the generated image and the real image, it is well known that pre-trained neural networks can extract abstract features of images at high levels. FID uses the 2048-dimensional vector before the full connection of Inception Net-V3 as the feature of the image. FID is the distance between the reaction-generated picture and the real picture, which is the distance to measure the two multivariate normal distributions. The smaller the data is, the better it will be. The calculation method is shown in Eq. 6.

$$\mathrm{FID} = ||\mu_x - \mu_g||^2 + \mathrm{T_r}\left(\sum x + \sum g - 2\left(\sum x \sum g\right)^{1/2}\right) \tag{6}$$

Among them, μ_x and $\sum x$ are the mean and covariance matrix of the 2048-dimensional feature vector set of the real image set output in Inception Net-V3. μ_g and $\sum g$ are the mean and covariance matrix of the 2048-dimensional feature vector set output in Inception Net-V3. Tr represents the trace of the matrix.

To judge the performance of a model, the convergence of the loss function should also be considered. Curves of loss function of models with epochs are shown in Figure 3. Obviously, the convergence of loss function of GAN is worse than the other three models', and that of SAGAN is comparatively stable and low near epoch 2k. So, it shows that tricks and changes in three improved models are effective.

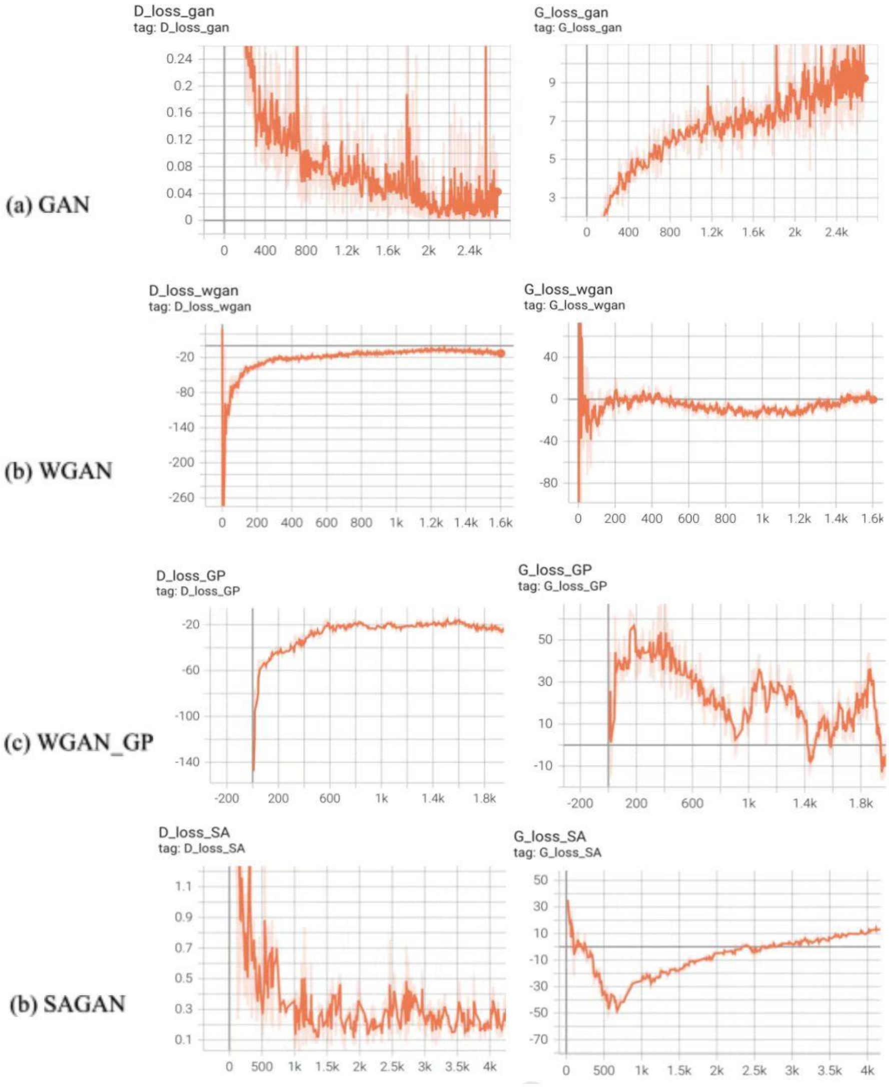


Figure 3. Loss values of the discriminator and generator of each model (GAN, WGAN, WGAN-GP, WGAN-GP + SAGAN) with epochs. In (a), the generator loss is increasing and it is difficult to converge, and this problem has been greatly improved in the curves of the improved models. In (d), the loss function in the original SAGAN is replaced by the Wasserstein distance with gradient penalty in WGAN-GP. We can find that loss functions of both the generator and discriminator converge to a smaller value than the other three models, which illustrates that the training effect gets improved.

As for IS and FID values, the variation curves are shown in Figure 4. The curves shown in the figure are all stable parts. Among the three curves of models, the IS score of GAN is the lowest while its FID score is higher than the others'. That means the quality of images generated by improved models is greater than those generated by original GAN. It is worth noting that the IS score of images generated by SAGAN using loss function of WGAN-GP is above 3 while the FID score is less than 150.

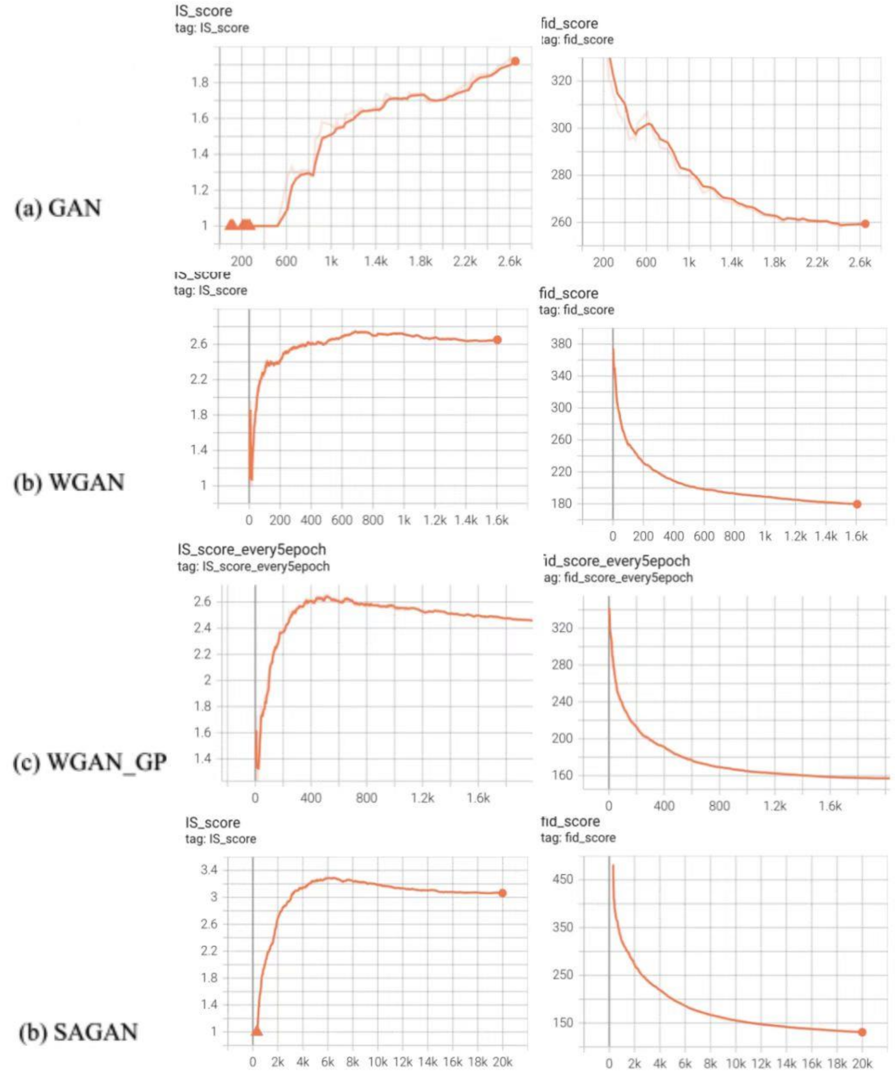


Figure 4. IS and FID score of each model (GAN, WGAN, WGAN-GP, WGAN-GP + SAGAN) with epochs. Since the first three models tend to be stable approximately after the 2,000th epoch (as shown in (a), (b), (c)), the latter part of the curve has not been displayed, while there are still significant changes in both two scores of SAGAN with loss of WGAN-GP after 10,000th epoch (d).

5. Summary and Prospect

In the field of image generation, to make the model perform better and generate high-quality images, we can increase the stability of training, solve the problem of pattern collapse, and strengthen the global feature association. The three models introduced in this paper have their own coping strategies and skills in this regard. WGAN introduces the bulldozer distance and uses the 1-Lipschitz condition to solve the training balance problem of the generator and the discriminator, which avoids mode collapse. In the subsequent WGAN-GP, the gradient penalty term replaces the previous weight clipping, which strengthens the modeling ability of the model and further solves the gradient disappearance and explosion problems left by WGAN. Therefore, the loss function and gradient penalty term in WGAN and WGAN-GP are worth references in the subsequent model improvement. On the other hand, SAGAN uses spectral normalization and TTUR to make training faster and more stable. More importantly, the attention mechanism is introduced to make training more global-dependent. SAGAN can help generate higher-quality images, especially those with fixed geometric structure features. The basis of these three models is GAN. The highlight of WGAN-GP is the improvement of the loss function, while the focus of SAGAN is to add the attention layer in the feature extraction process. In the experiment of combining the two, Wasserstein distance and gradient penalty in WGAN-GP are introduced into SAGAN, so that the convergence of loss function and the evaluation index of generated images are improved.

References

[1] Kingma D P , Welling M . Auto-Encoding Variational Bayes[J]. arXiv.org, 2014.
[2] Ekanadham C . Sparse deep belief net models for visual area V2[J]. Advances in Neural Information Processing Systems, 2008.
[3] Gregor K , Danihelka I , Mnih A , et al. Deep AutoRegressive Networks[J]. 2013.
[4] Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generative adversarial nets[J]. Advances in neural information processing systems, 2014, 27.
[5] Tim, Bollerslev. Generalized autoregressive conditional heteroskedasticity[J]. Journal of Econometrics, 1986.
[6] Van Oord A, Kalchbrenner N, Kavukcuoglu K. Pixel recurrent neural networks[C]//International conference on machine learning. PMLR, 2016: 1747-1756.
[7] Radford A , Metz L , Chintala S . Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks[J]. Computer ence, 2015.
[8] Salimans T, Goodfellow I, Zaremba W, et al. Improved techniques for training GANs[J]. Advances in neural information processing systems, 2016, 29.
[9] Chen X, Duan Y, Houthooft R, et al. InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets[J]. Advances in neural information processing systems, 2016, 29.
[10] Nowozin S, Cseke B, Tomioka R. f-GAN: Training generative neural samplers using variational divergence minimization[J]. Advances in neural information processing systems, 2016, 29.
[11] Reed S , Akata Z , Yan X , et al. Generative Adversarial Text to Image Synthesis[J]. JMLR.org, 2016.
[12] Wang X , Gupta A . Generative Image Modeling Using Style and Structure Adversarial Networks[J]. Springer International Publishing, 2016.
[13] Denton E L, Chintala S, Fergus R. Deep generative image models using a laplacian pyramid of adversarial networks[J]. Advances in neural information processing systems, 2015, 28.
[14] Zhang H , Xu T , Li H , et al. StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks[J]. IEEE, 2017.
[15] Huang X, Li Y, Poursaeed O, et al. Stacked generative adversarial networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 5077-5086.
[16] Zhang H, Goodfellow I , Metaxas D , et al. Self-Attention Generative Adversarial Networks[J]. 2018.
[17] Gulrajani I, Ahmed F, Arjovsky M, et al. Improved training of wasserstein GANs[J]. Advances in neural information processing systems, 2017, 30.
[18] Arjovsky M , Chintala S , Bottou L . Wasserstein GAN[J]. 2017.
[19] Heusel M , Ramsauer H , Unterthiner T , et al. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium[J]. 2017.