D2PGGAN: TWO DISCRIMINATORS USED IN PROGRESSIVE GROWING OF GANS

Zhaoyu Zhang, Mengyan Li and Jun Yu[*]

Department of Automation, University of Science and Technology of China

ABSTRACT

Generative adversarial network (GAN) is a powerful generative model. However, it suffers from two key problems which are convergence instability and mode collapse. Recently, progressive growing of GANs for improving quality, stability and variation (PGGAN) is proposed to better solve these two problems. Although the performance of PGGAN is good on these two problems, it is still not satisfied on mode collapse problem. In this paper, we propose a new architecture based on PGGAN called D2PGGAN to better solve the mode collapse problem. The key idea consists of one generator and two different discriminators in PGGAN. With the fact that GAN is the analogy of a minimax game, the proposed architecture is as follows. The generator (G) aims to produce realistic-looking samples to fool both of two discriminators. The first discriminator (D_1) rewards high scores for samples from the data distribution, while the second one (D_2) favors samples from the generator conversely. Specifically, a novel loss function is designed to optimize the proposed D2PGGAN. Extensive experiments on CIFAR-10 and CIFAR-100 datasets demonstrate that the proposed method is effective and obtains the highest inception scores compared with others state-of-the-art GANs.

Index Terms— GAN, PGGAN, mode collapse.

1. INTRODUCTION

Generative adversarial network (GAN) [1] is one of the most powerful generative models [2, 3] that can produce very visually appealing samples, however, it is often difficult to train and suffers from the convergence problem [4]. Most of the recent studies have been devoted to finding ways to solve convergence problem, such as WGAN [5] and WGAN-GP [6]. Some of the works such as PGGAN [7] has a great successful to solve this convergence problem and obtains the highest inception scores [8] of the GAN. In addition to the training and convergence problems, GAN also suffers from the mode collapse problem [9]. Minibatch discrimination [8] is firstly proposed to solve the mode collapse problem, PGGAN also used this method, unfortunately, the performance of the PGGAN on mode collapse problem is still not good enough.

*Corresponding author.

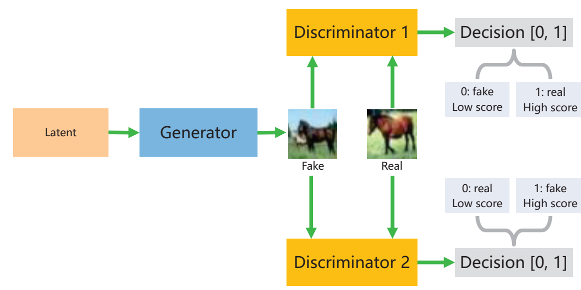


Fig. 1: The overview of our proposed D2PGGAN.

Focusing on the mode collapse problem, this paper presents a novel GAN architecture based on PGGAN called D2PGGAN which consists of one generator (G) and two different discriminators (D_1 and D_2), inspired by D2GAN [10]. As shown in Figure 1, the discriminator D_1 rewards high scores for data that are sampled from the distribution of real data (p_{data}) and gives low scores for data that are generated from the distribution of generated samples (p_G). Conversely, the discriminator D_2 is in favor of data generated from p_G and despises data sampled from p_{data}. In this circumstance, D_2 will balance D_1. The architecture can avoid the generator to generate the similar samples. This method will relieve the mode collapse problem of GAN and further improve this problem of PGGAN. In order to optimize the model, we proposed a new loss function which is similar to the original loss function of PGGAN. Experiments on CIFAR-10 [11] and CIFAR-100 [11] fully demonstrate the effectiveness of our method. The key contributions of this paper are the following:

1. Further improving the mode collapse problem of PGGAN [7]. In our architecture, two discriminators are used to balance the generator based on the PGGAN. Compared to other state-of-the-art GANs [6–8, 10, 12–17] our method can further increase the diversity of the generated samples. The experiments on CIFAR-10 and CIFAR-100 show that our method fixes the mode collapse problem of the PGGAN.
2. Obtaining the highest inception scores on CIFAR-10/100. Compared to other state-of-the-art GANs [6–8, 10, 12–17], our method obtained the highest inception scores on CIFAR-10 and CIFAR-100, which means our method is effective.

978-1-5386-4658-8/18/$31.00 ©2019 IEEE

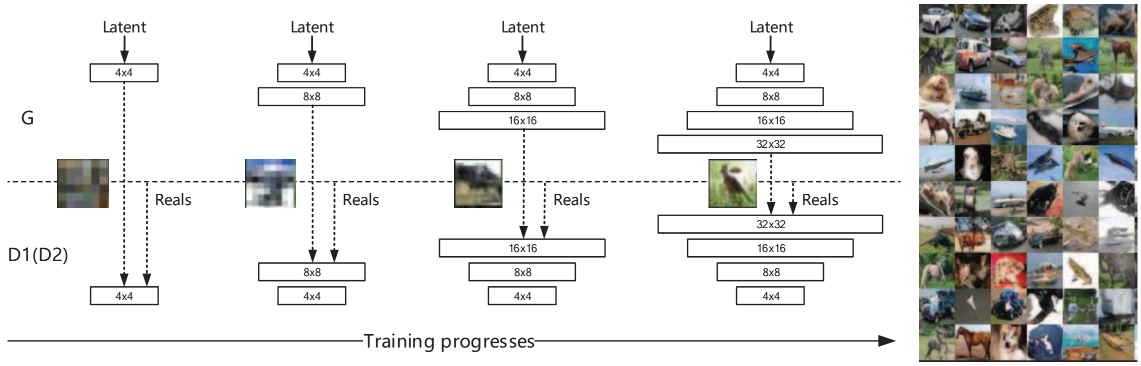


Fig. 2: The architecture and training process of our proposed D2PGGAN

2. PROPOSED METHOD

2.1. The Mode Collapse Problems of GAN

Although GAN has a significant influence, the learning goals will lead to mode collapse, i.e., the generator generates only a small number of training samples to fool the discriminators. This methodology is a type of nostalgia for the estimation of the maximum likelihood density in past Gaussian mixtures: through the collision of changes in each component, some permanent similarities can be obtained, and then, we store these similarities in the dataset. However, these similarities are useless for the density estimates that can be generated.

2.2. The architecture and training process of D2PGGAN

In this section, we will introduce the architecture of our GAN(D2PGGAN), the architecture of the G, D_1 and D_2 can be found in Fig.2. As it can be seen in Fig.2, the two discriminators have the same architecture, which is because of that two discriminators cannot both satisfy the generator if they are different, and this will cause one of the discriminators do not affect. The generator is an architecture which on different spatial resolutions. Thus in order to satisfy the generator, the architecture of the discriminators are also on the same spatial resolutions with the generator. If D_2 uses different architecture as D_1, D_2 will have less effect so that our D2PGGAN will obtain the lower inception scores. The architecture of Fig.2 is combining from the architecture of D2GAN [10] and PGGAN [7], it has the following benefits. Firstly, in the early of training, the generation of smaller images is substantially more stable because there are less class information and fewer modes. Secondly, this architecture can reduce training time. With progressively growing GANs, most of the iterations are done at lower resolutions, and comparable result quality is often obtained up to $2 \sim 6$ times faster, depending on the final output resolution. Furthermore, using two discriminators can also improve the mode collapse problem, our architecture can train a better model of the GAN.

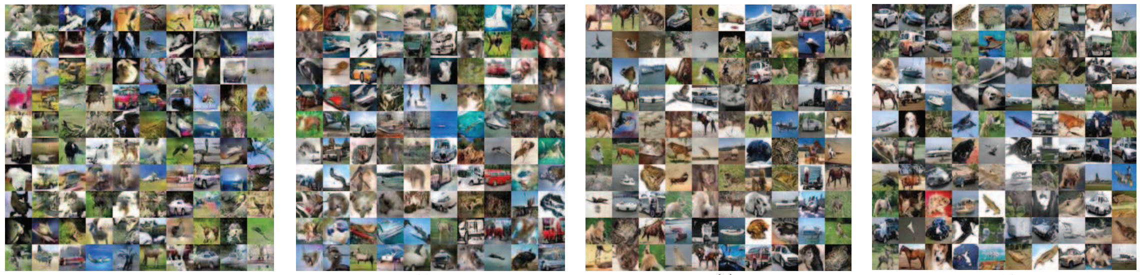
Fig.2 shows the training process of our method on the CIFAR-10 dataset with unsupervised learning (no labels). Our training starts with both the generator (G) and two discriminators (D_1 and D_2) having a low spatial resolution of 4×4 pixels. Two discriminators have the same architecture in order to both satisfy the generator. As the training advances, we incrementally add layers to G and $D_1(D_2)$, thus increasing the spatial resolution of the generated images. All existing layers remain trainable throughout the process. Here $N \times N$ refers to convolutional layers operating on $N \times N$ spatial resolution, which allows stable synthesis in high resolutions and also speeds up training considerably. On the right, we show some example images generated using progressive growing at 32×32.

2.3. Loss Function

The loss function is one of the most important parts of the GAN in that it can determine the performance of the GAN. Many works which improve GAN only change the loss function to obtain the better performance. In this section, we will introduce the loss function of our architecture as follows.

$$\min_{G} \max_{D_1,D_2} V(D_1,D_2,G) = \underset{z\sim p_z}{E}[D1(G(z))] - \underset{x\sim p_{data}}{E}[D1(x)] + gp + \underset{x\sim p_{data}}{E}[D2(x)] - \underset{z\sim p_z}{E}[D2(G(z))] + gp. \tag{1}$$

Similar with the PGGAN, the loss function of our architecture also use the loss function of the WGAN-GP [6], but the gradient penalty item gp is different, in the paper of WGAN-GP, gp is $\lambda \underset{z\sim p_z}{E}[(\left\|\nabla_{G(z)} D(G(z))\right\| - 1)^2]$. But in our architecture, the gradient penalty item gp is $\lambda \underset{z\sim p_z}{E}[(\left\|\nabla_{G(z)} D(G(z))\right\| - \gamma)^2/\gamma^2]$, and $\gamma = 750$. With the fact that the gradient penalty item gp is the $\gamma = 1$ in WGAN-GP, it is significantly better to prefer fast transitions ($\gamma = 750$) to minimize the ghosts. In our loss function, two discriminators all use Gradient penalty in order to better optimize the model, in the paper of WGAN-GP, they proof the loss function of WGAN-GP is convergent and the gp is



(a) (b) (c) (d)

Fig. 3: The results of different GANs applied to CIFAR-10 dataset (a) DCGAN-vanilla (b) D2GAN (c) PGGAN and (d) Our GAN(D2PGGAN). It can be seen that the images generated by our GAN (D2PGGAN) are better and diversity.

not influencing the convergence of the loss function, we will prove the loss function in Eq.(3) has the same convergence with the loss function of WGAN-GP. The loss function of WGAN-GP is as follow.

$$\min_G \max_D V(D,G) = \underset{z\sim p_z}{E}[D(G(z))] - \underset{x\sim p_{data}}{E}[D(x)] + \lambda \underset{z\sim p_z}{E}[(\left\|\nabla_{G(z)} D(G(z))\right\| - 1)^2]. \tag{2}$$

With the fact that gp is not influencing the convergence of the loss function, we only focus on the items without gp. According to the induced measure theorem [18], two expectations are equal, i.e.

$$\underset{z\sim p_z}{E}[f(G(z))] = \underset{x\sim p_G}{E}[f(x)],$$

where $f(x) = D_1(x)$ or $f(x) = D_2(x)$. So the objective function of WGAN-GP can be rewritten as:

$$\min_G \max_D V(D,G) = \underset{x\sim p_G}{E}[D(x)] - \underset{x\sim p_{data}}{E}[D(x)] + \lambda \underset{x\sim p_G}{E}[(\|\nabla_x D(x)\| - 1)^2]. \tag{3}$$

From the Eq.(5) it can be seen that the propose of optimizing the loss function of WGAN-GP is to obtain $p_G = p_{data}$, so that the loss function of WGAN-GP is convergent and optimizing the loss function of WGAN-GP is equal to optimize $p_G = p_{data}$. Again, according to the induced measure theorem [18], the loss function of our method in Eq.(3) can be written as:

$$\min_G \max_{D_1,D_2} V(D_1,D_2,G) = \underset{x\sim p_G}{E}[D1(x)] - \underset{x\sim p_{data}}{E}[D1(x)] + gp + \underset{x\sim p_{data}}{E}[D2(x)] - \underset{x\sim p_G}{E}[D2(G(x))] + gp. \tag{4}$$

where the gp is $\lambda \underset{x\sim p_G}{E}[(\|\nabla_x D(x)\| - \gamma)^2/\gamma^2]$ and $\gamma = 750$. From the Eq.(6) it can be seen that optimizing the loss function of our method is also equal to optimizing $p_G = p_{data}$, so that the loss function of our method is also convergent. Because D_1 and D_2 will optimize G at different aspects so that the loss function of our method will better optimize the model.

3. EXPERIMENTS

We demonstrate the superiority of our proposed network on two datasets (CIFAR-10 [11] and CIFAR-100 [11]) using our proposed architecture and compare with other state-of-the-art GANs [6–8, 10, 12–17]. A workstation with Intel i7-7700K 4.2G, 64G memory and two NVIDIA GTX1080Tis are used for the experiments.

3.1. Evaluation with Inception Scores

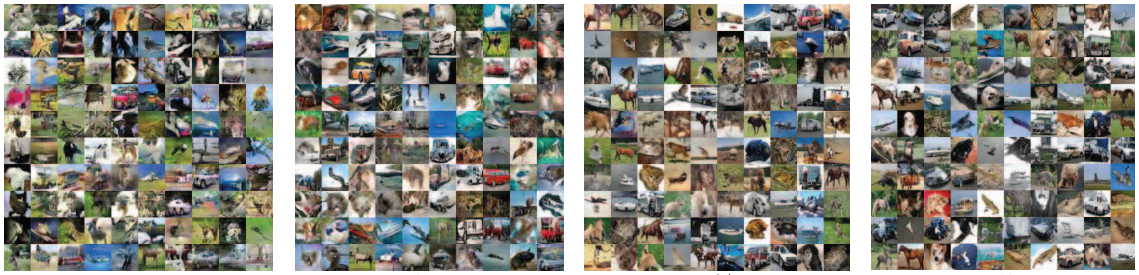
Evaluating the quality of the image produced by generative models is notoriously challenging due to the variety of probability criteria and the lack of a perceptually meaningful image similarity metric [19]. Even a model can generate plausible images, it is not useful if those images are visually similar. Therefore, to quantify the performance of covering data modes as well as producing high-quality samples, we adopt the inception score proposed in [8], which is computed by:

$$exp(E_x\left[D_{KL}(p(y|x) \parallel p(y))\right]) \tag{5}$$

where $p(y|x)$ is the conditional label distribution for image x that is estimated using a pretrained inception model [20, 21], and $p(y)$ is the marginal distribution: $p(y) \approx 1/N\sum_{n=1}^{N} p(y|x_n = G(z_n))$. This score can adequately reflect the variety and visual quality of the images [22].

3.2. Results on CIFAR-10 Dataset

In this section, the values of the inception scores on our model and the others models are shown in Table 1, all the others GANs are best run results. Some inception scores of the others GANs are different from the original papers in that we use chainer not tensorflow to obtain the inception scores. This is because more and more work of GAN using chainer [23], using chainer is convenient to evaluate the model. Because GAN will obtain different results in different times, we show the average inception scores computed from 5 times run and the value is 8.59, this value is still higher than other GANs. To further showing our GAN is better, the images generated on



(a) (b) (c) (d)

Fig. 4: The results of different GANs applied to CIFAR-100 dataset (a) DCGAN-vanilla (b) D2GAN (c) PGGAN (d) Our GAN (D2PGGAN). It can be seen that the images generated by our GAN (D2PGGAN) are better and diversity.

Table 1: Inception scores on the CIFAR-10 dataset.

Method	Inception scores (chainer [23])
Real data	12.00
WGAN-GP [6]	6.80
DFM [12]	7.30
Cramer GAN [13]	6.40
SN-DCGAN [14]	7.50
DRAGAN [15]	7.10
DCGAN-vanilla [16]	6.70
Minibatch discrimination [8]	7.00
BEGAN [17]	5.40
D2GAN [10]	6.76
PGGAN [7]	8.50
Our D2PGGAN (best run)	**8.83**
Our D2PGGAN (computed from 5 runs)	**8.59**

Table 2: Inception scores on the CIFAR-100 dataset.

Method	Inception scores (chainer [23])
Real data	15.06
WGAN-GP [6]	6.74
DFM [12]	6.88
Cramer GAN [13]	6.32
SN-DCGAN [14]	7.34
DRAGAN [15]	6.76
DCGAN-vanilla [16]	7.04
Minibatch discrimination [8]	7.20
BEGAN [17]	6.24
D2GAN [10]	7.07
PGGAN [7]	7.73
Our D2PGGAN (best run)	**8.22**
Our D2PGGAN (computed from 5 runs)	**8.11**

CIFAR-10 can be found in Fig.3, which includes DCGAN, Minibatch discrimination (MD), D2GAN and Our GAN, it can be seen that the images generated by our GAN is better and more diversity.

3.3. Results on CIFAR-100 Dataset

In this section, the results on the CIFAR-100 dataset are shown in Table 2. There are two reasons for choosing this dataset. The first reason is that CIFAR-100 has the same inception model as CIFAR-10, which makes it convenient to evaluate the model with inception scores, while the other datasets cannot do that. The second reason is that CIFAR-100 has more classes, which means that each class has fewer samples to use. Using fewer samples to generate images is also a meaningful work. The quantitative results of the inception scores are shown in Table 2 and it can be observed that our model yields the highest inception score compared with other state-of-the-art GANs. As the results of the CIFAR-10 dataset, all the others GANs are best-run results and obtained by ourself. The real data on the inception scores of CIFAR-100 is 15.06 which is much higher than that of CIFAR-10 because CIFAR-100 has more classes than CIFAR-10. Because GAN will obtain different results in different times during training, we show the average inception scores computed from 5 times run and the value is 8.11, this value is higher than all of the other state-of-the-art GANs. In general, the inception scores of CIFAR-100 are lower than CIFAR-10 because fewer samples of each classes is proposed. Our method still yields highest inception scores and this implies its superiority.

Additionally, several samples generated by our proposed model and the other GANs on the CIFAR-100 dataset are shown in Fig.4 including DCGAN-vanilla [16], D2GAN [10], PGGAN and our GAN (D2PGGAN). The objects are becoming harder to recognize, but it can still be observed that our method generates better images with higher diversity.

4. CONCLUSIONS

In this paper, we propose a new architecture based on PGGAN called D2PGGAN to better solve the mode collapse problem. The key idea consists of one generator and two different discriminators in PGGAN. Extensive experiments on CIFAR-10 and CIFAR-100 datasets demonstrate that the proposed method is effective. In the future, we will focus on the large-scale datasets such as ImageNet [24] to obtain a higher value of the inception score and to train a better GAN model.

5. ACKNOWLEDGEMENT

This work is supported by the National Natural Science Foundation of China (U1736123, 61572450, 61303150), Anhui Provincial Natural Science Foundation (1708085QF138), the Fundamental Research Funds for the Central Universities (WK2350000002).

6. REFERENCES

[1] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[2] Aaron van den Oord, Nal Kalchbrenner, Lasse Espeholt, Oriol Vinyals, Alex Graves, et al., "Conditional image generation with pixelcnn decoders," in *Advances in Neural Information Processing Systems*, 2016, pp. 4790–4798.

[3] Zhaoyu Zhang, Mengyan Li, and Jun Yu, "On the convergence and mode collapse of gan," in *SIGGRAPH Asia 2018 Technical Briefs*. ACM, 2018, p. 21.

[4] Martin Arjovsky and Léon Bottou, "Towards principled methods for training generative adversarial networks," *In International Conference on Learning Representations*, 2017.

[5] Martin Arjovsky, Soumith Chintala, and Léon Bottou, "Wasserstein gan," *arXiv preprint arXiv:1701.07875*, 2017.

[6] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville, "Improved training of wasserstein gans," in *Advances in Neural Information Processing Systems*, 2017, pp. 5769–5779.

[7] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," *arXiv preprint arXiv:1710.10196*, 2017.

[8] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen, "Improved techniques for training gans," in *Advances in Neural Information Processing Systems*, 2016, pp. 2234–2242.

[9] Yunus Saatci and Andrew G Wilson, "Bayesian gan," in *Advances in neural information processing systems*, 2017, pp. 3622–3631.

[10] Tu Nguyen, Trung Le, Hung Vu, and Dinh Phung, "Dual discriminator generative adversarial nets," in *Advances in Neural Information Processing Systems*, 2017, pp. 2667–2677.

[11] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton, "The cifar-10 dataset," *online: http://www. cs. toronto. edu/kriz/cifar. html*, 2014.

[12] David Warde-Farley and Yoshua Bengio, "Improving generative adversarial networks with denoising feature matching," 2016.

[13] Marc G Bellemare, Ivo Danihelka, Will Dabney, Shakir Mohamed, Balaji Lakshminarayanan, Stephan Hoyer, and Rémi Munos, "The cramer distance as a solution to biased wasserstein gradients," *arXiv preprint arXiv:1705.10743*, 2017.

[14] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida, "Spectral normalization for generative adversarial networks," *arXiv preprint arXiv:1802.05957*, 2018.

[15] Naveen Kodali, James Hays, Jacob Abernethy, and Zsolt Kira, "On convergence and stability of gans," 2018.

[16] Alec Radford, Luke Metz, and Soumith Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *In International Conference on Learning Representations*, 2016.

[17] David Berthelot, Tom Schumm, and Luke Metz, "Began: Boundary equilibrium generative adversarial networks," *arXiv preprint arXiv:1703.10717*, 2017.

[18] Somesh Das Gupta, "Mathematical statistics," 2000.

[19] Lucas Theis, Aäron van den Oord, and Matthias Bethge, "A note on the evaluation of generative models," *arXiv preprint arXiv:1511.01844*, 2015.

[20] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2818–2826.

[21] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen, "Improved techniques for training gan," *online:http://download.tensorflow.org/models/ image/imagenet/inception-2015-12-05.tgz*, 2016.

[22] Tong Che, Yanran Li, Athul Paul Jacob, Yoshua Bengio, and Wenjie Li, "Mode regularized generative adversarial networks," *arXiv preprint arXiv:1612.02136*, 2016.

[23] Seiya Tokui, Kenta Oono, Shohei Hido, and Justin Clayton, "Chainer: a next-generation open source framework for deep learning," in *Proceedings of workshop on machine learning systems (LearningSys) in the twenty-ninth annual conference on neural information processing systems (NIPS)*, 2015, vol. 5.

[24] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 248–255.