TOWARDS FASTER AND STABILIZED GAN TRAINING FOR HIGH-FIDELITY FEW-SHOT IMAGE / FREIMPLEMENTATION REPORT

Zeyu Dong & Wenyu He &Ruoxuan Hu & Sihao Dai
Electronic and computer science
University of Southampton
{zd1m20,wh2n21,rh5n21,sd1a21}@soton.ac.uk

ABSTRACT

Since training Generative Adversarial Networks (GANs) on high-fidelity images usually requires large GPU clusters and a large number of training images, the authors of the selected replication paper investigate a GAN architecture to perform the new-shot image synthesis task with minimal computational cost. The authors of the selected replication paper investigated a GAN architecture to perform the new-shot image synthesis task with minimal computational cost. Trained as a feature encoder. This report investigates the reproducibility of the optimised structure proposed for the team and confirms the superiority of this lightweight GAN structure. Code and report: https://github.com/Always066/COMP6248-Reproducibility-Challenge

1 INTRODUCTION

The state-of-the-art (SOTA) Generative Adversarial Networks (GANs) have great potential for real-life applications,Goodfellow et al. (2014); however, the high computational cost and the size requirement of training data limit its application. Aiming at the problem of training data scale, Mo et al. (2020) and Wang et al. (2020b) tried to use pre-training transfer learning to solve the problem of small samples, but Zhao et al. (2020) believed that if the data sets were not compatible, will get worse performance. Karras et al. (2020a) proposed to stabilize GAN training with small-scale datasets. However, SOTA models such as StyleGAN2 Karras et al. (2020b) and BigGAN Brock et al. (2018) still suffer from high computational costs. Our replicated paper aims to learn an unconditional GAN on high-resolution images, achieving a structure with low computational cost and few training samples. As this training condition is prone to overfitting, to train the GAN smoothly, the team upgrades it in two ways. On the one hand, designing the Skip-Layer channel-wise Excitation (SLE) module to correct the channel response of the large scale feature maps by low scale activation. On the other hand, a self-supervised discriminator D with an additional decoder is proposed to train G by forcing D to learn a more descriptive feature map that covers more regions of the input image, generating more signals.

2 ANALYSIS OF THE ORIGINAL PAPER

This paper has two contributions, which are the use of skip-layer channel-wise excitation module (SLE) generator and Self-supervied discriminator and the improved DCGAN model combining the former two.

In order to synthesize higher-resolution images, it means that the generator needs a deeper network to learn super-sparse features, but this brings new problems, that is, it needs to learn more parameters, use more samples and longer training time. The model of this paper is improved on the basis of DCGAN. On DCGAN,Radford et al. (2015)introduced the convolutional neural network into GAN, and Wang et al. (2020a) added Residual Block to GAN in order to solve the gradient transfer problem of deeper networks. However, this further increases the number of parameters and further increases the computational cost. Liu et al. (2021) proposed SLE to transfer gradients not by addition but by multiplication, which saves computational cost without keeping the resolution of feature maps the same . At the same time, SLE has another advantage, it can separate style and content, and only need to change a little feature map to transfer style.

Another contribution ofLiu et al. (2021) is to propose a Self-supervied discriminator to regularize the discriminator. In the GAN network, if the model is not constrained, it is difficult to ensure the centrality and descriptiveness of the feature map, especially in the case of using a small sample high-resolution training set. This problem will be more obvious. To this end, the author added a decoder to the discriminator, allowing the discriminator to decode the generated features into images and compare them with the real images to ensure that the features are mapped to the real image set. At the same time, in order to ensure that both texture and structure can be constrained, in addition to judging the down-sampled image of the real image and the image generated by the decoder, the original image is also partially cropped to achieve texture constraints. The reconstruction loss function mentioned in the paper is:

$$\mathcal{L}_{\text{recons}} = \mathbb{E}_{\mathbf{f}\sim D_{\text{encode}}(x), x\sim I_{\text{real}}}\left[||\mathcal{G}(\mathbf{f}) - \mathcal{T}(x)||\right]$$

3 Experiment

The code comes from the author. In order to get different test results, the code is slightly modified. In order to obtain the baseline model for comparative experiments, a small part of forward is modified on the model of the official code.

3.1 Reproduce the experimental goal

For the DCGANs series of networks, the generator and the discriminator are the two main components. Our report listed the authors' contributions to these two parts and what drove these innovations. Since this paper is a paper that has received more attention at the ICLR conference, the official website source code has great reference significance. These complex structures are more intuitive to understand through code than structural model diagrams.

After the code review, our team found no baseline model in the official source code, which provided the final model with added technical improvements. Due to the limitation of computing resources, StylyGan2 belongs to a very large-scale network, and personal computer training is tough. Therefore, this group is devoted to the structural analysis of the author's final model structure. Rather than reproduce the various complex ablation experiments in the paper's appendix. Code analysis is achievable for our computing resources and abilities for the learning goal.

3.2 Source code analysis

All model structures are defined in the models.py file in the official implementation project. After the analysis and comparison of the code and the paper, in the Generator part, the specific implementation structure of the model is precisely the same as the expression submitted in the original paper. However, there are some changes in implementing the model in Discriminator, especially in the loss function.

Figure1 shows the four parts of the loss function, which is different from the expression of the paper. Brach 3, mentioned in the original paper, is the Perceptual loss generated by part of the image in the original image and part of the image in the Discriminator. Branch 5 is The perceptual loss of all images extracted from the downsampled original image and the feature map 2 of the Discriminator.

In addition to the two branches mentioned in the original text, Branch 1 performs a simple convolution structure independent of the discriminator structure on the real image in implementing the official source code. Moreover, this simple convolutional structure plays a specific downsampling function named Downsample Conv Component. Downsample Conv Component looks like a small discriminator. Its output Brach 4 adds the sum of logits score and the Discriminator's logits for the Logits Loss. Branch 1 and 4 complete a simplified Discriminator, which extracts low-dimensional information. The Downsmaple Conv Component computes the Discriminator and the logits for computing the generator loss.

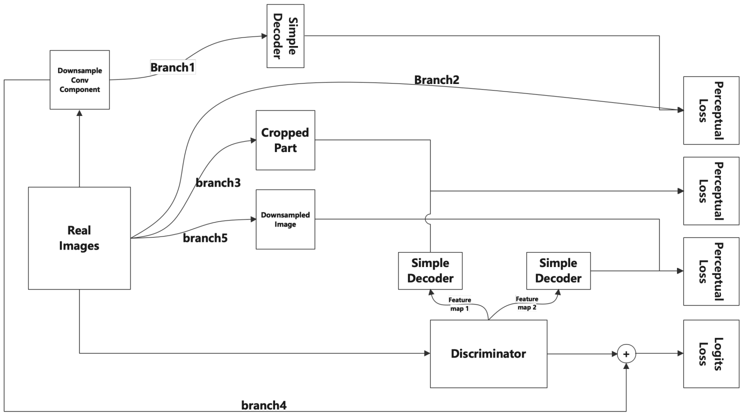

Figure 1: Structure of Discriminator for real images.

3.3 Analysis of training results

On the student computer in the University of Southampton ECS laboratory, the 8Gb memory of the RTX2070, batch size 4. After 8 hours of 50,000 iterations on the anime-face few-shot learning datasets.

We end up with the following loss image:

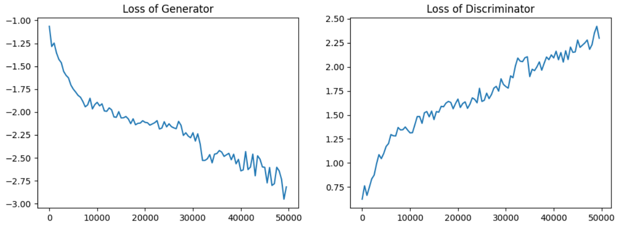

Figure 2: Loss on Generator and Discriminator.

Liu et al. (2021)did not mention the loss function in the paper, they just used the Fréchet Inception Distance (FID) to illustrate the generated results. However, in the reproducible, the loss of the discriminator has been on the rise. Intuitively, guessing that the discriminator also needs to judge the difference between the decoded picture and the real picture extracted by the discriminator, which forms a constraint on the discriminator, which means that the discriminator cannot only judge the generator. Generate images as the only training target.

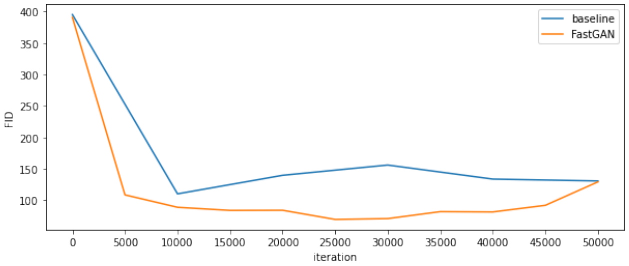

Figure 3: FID of baseline and FastGAN

In order to verify the effect of the two structures proposed by A, after removing the generator's BLE in the official code, the training speed on the same device is reduced by about 7%. Since the author mentioned the details of the compared baseline in detail, on the basis of the previous step, the reconstruction loss was continued to be removed, and its FID is shown in Figure 3, with the removal of SLE and the self-monitoring discriminator, the convergence rate of the model improved slightly, and the model crash caused by overtraining was alleviated. Even after overtraining, it is more stable.

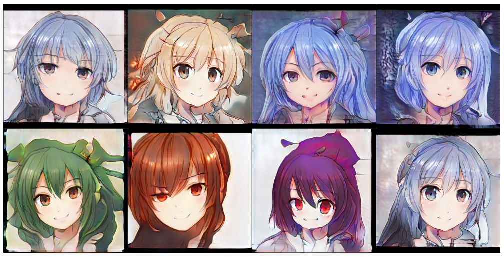

Figure 4: Generation from iteration 25000

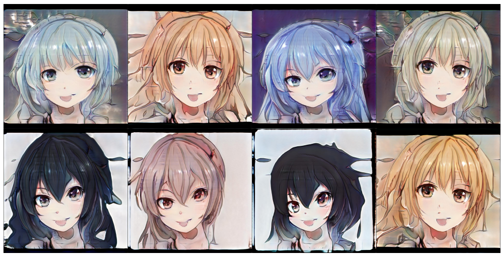

Figure 5: Generation from iteration 35000

The generation result of the generator is shown in Figure 4. When the iteration is 25000, it is the time when the FID is the lowest and the generation effect is the best. When the iteration reaches 35000, the FID starts to rise due to the transition training. The generated image is shown in Figure 5, and the image quality decreases.

4 Summary

4.1 Code differences from the original paper

In the code, Liu et al. (2021) changed the reconstruction loss to Perceptual loss, and the loss proposed by Johnson et al. (2016) includes feature loss and style loss,

$$\ell_{\text{style}}^{\phi,j}(\hat{y},y)=\left\|G_j^{\phi}(\hat{y})-G_j^{\phi}(y)\right\|_F^2$$

where G is the Gram matrix, which can compare the structural information of the image and lose high frequency less information. Especially in the style transfer, since the style loss compares some common semantic information of the whole image, and this commonality happens to form the artistic style of the image. Intuitively, whether this is valuable for the constraint method of the decoder, because, in the Generator, the style of the image is learned, but if the reconstruction loss does not consider the style, then when the decoder restricts the discriminator through the loss, the potential style information will be ignored to a certain extent because only the features are constrained.

In addition, what is not mentioned in the paper is that the SLE block is also used in the discriminator, but unlike the SLE block in the generator, the SLE block in the discriminator is not used for layer jumping but in the process of extracting features Multiplying the previously extracted features not only does not reduce the computational cost but also increases it. This may be used to preserve the style of the image.

4.2 summary of result

In the reproducible, the role of BLE and the self-supervised discriminator is also obvious. Although BLE is not very obvious in computing cost savings, it has the ability of style transfer to make it valuable. The effect of the self-supervised discriminator is more significant. Although the decoder requires additional computational cost, the convergence speed of the model is significantly improved, and the training model is more stable and less prone to mode collapse.

References

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

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling (eds.), *Computer Vision – ECCV 2016*, pp. 694–711, Cham, 2016. Springer International Publishing. ISBN 978-3-319-46475-6.

Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. *Advances in Neural Information Processing Systems*, 33:12104–12114, 2020a.

Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8110–8119, 2020b.

Bingchen Liu, Yizhe Zhu, Kunpeng Song, and Ahmed Elgammal. Towards faster and stabilized GAN training for high-fidelity few-shot image synthesis. *CoRR*, abs/2101.04775, 2021. URL `https://arxiv.org/abs/2101.04775`.

Sangwoo Mo, Minsu Cho, and Jinwoo Shin. Freeze the discriminator: a simple baseline for fine-tuning gans. *arXiv preprint arXiv:2002.10964*, 2020.

Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks, 2015. URL `https://arxiv.org/abs/1511.06434`.

Yaxing Wang, Abel Gonzalez-Garcia, David Berga, Luis Herranz, Fahad Shahbaz Khan, and Joost van de Weijer. Minegan: Effective knowledge transfer from gans to target domains with few images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020a.

Yaxing Wang, Abel Gonzalez-Garcia, David Berga, Luis Herranz, Fahad Shahbaz Khan, and Joost van de Weijer. Minegan: effective knowledge transfer from gans to target domains with few images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9332–9341, 2020b.

Shengyu Zhao, Zhijian Liu, Ji Lin, Jun-Yan Zhu, and Song Han. Differentiable augmentation for data-efficient gan training. *Advances in Neural Information Processing Systems*, 33:7559–7570, 2020.