EasyChair Preprint

№ 2549

Study of GANs using a Few Images for Sealer Inspection Systems

Dongwook Seo, Yejin Ha, Seungbo Ha, Kang-Hyun Jo and Hyun-Deok Kang

EasyChair preprints are intended for rapid dissemination of research results and are integrated with the rest of EasyChair.

February 4, 2020

Study of GANs using a Few Images for Sealer Inspection Systems

Dongwook Seo[1], Yejin Ha[2], Seungbo Ha[3], Kang-Hyun Jo[1], and Hyun-Deok Kang[2,3]

[1] School of Electrical Engineering, Uinversity of Ulsan, Ulsan 44610, Republic of Korea
{seodongwook99,acejo2208}@gmail.com
http://islab.ulsan.ac.kr
[2] Seohong Tech Co., Ulsan 44919, Republic of Korea
{yejinha24,khd0425}@gmail.com
http://feetness.co.kr
[3] School of Electrical and Computer Engineering, UNIST, Ulsan 44919, Republic of Korea
{mj0829,khd0425}@unist.ac.kr

Abstract. This paper describes a comparative study of the performance of Generative Adversarial Networks (GANs) through the quality of the generated images by using a few samples. In the deep learning-based systems, the amount and quality of data are important. However, in industrial sites, data acquisition is difficult or limited for some reasons such as security and industrial specificity, etc. Therefore, it is necessary to increase small-scale data to large-scale data for the training model. GANs is one of the representative image generation models using deep learning. Three GANs such as DCGAN, BEGAN, and SinGAN are used to compare the quality of the generated image samples. The comparison is carried out based on the score with different measuring methods.

Keywords: Generative Adversarial Networks · Sealer · Vision Inspection Systems

1 Introduction

The machine vision acquires images using cameras, optical systems, lights, etc. to inspect products and detect defects during manufacturing processes [3]. The automated machine vision system surpasses human abilities and realizes high optical resolution, consistency, and high accuracy. However, traditional machine vision systems show limitations to various environmental conditions at industrial sites. Especially, they are very sensitive to illumination variation and difficult to adapt to different inspected items. Also, they have possibilities of image distortion from changes of the angle and the positions. It is very inefficient to modify the systems due to each of the environmental changes for maintaining successful inspection systems. A deep learning model combines the self-learning ability of

humans and the data processing speed of computing systems and overcomes the limitations of traditional visual inspection systems [3]. For the machine vision inspection system using deep learning, data acquisition is difficult due to the specificity of the data, security issues, and environmental limitations.

The accuracy and validity of deep learning models are significantly affected by the quality and quantity of input data [8,27]. Learning with limited data cause over-fitting or fails to train. According to Goodfellow et al. [8], at least 5 thousand learning data are needed for each category to reach acceptable performance, and more than 10 million training samples are required to reach or surpass human abilities. For this reason, the importance of large datasets has grown fast and, therefore, various attempts such as data augmentation, data generation, data labeling, etc. have been tried to increase data [27]. Currently, datasets based on daily lives (ImageNet [5], Visual Object Classes (VOC) [6], Common Object in Context (COCO) [20], etc.) are available for deep learning researches, but it is hard to acquire data from the industrial process [7].

A sealing image is one example of hardly accessible data. A sealant is used to attach heterogeneous materials in manufacturing processes, and a machine vision system inspects whether the sealant is applied properly. Thousands of high-quality sealing images are required for deep learning-based machine vision inspections, but data is not acquired easily because the images are obtained from specific processes at the industrial site, and the security issues are related to the products.

Generating artificial data has been tried by learning the properties of the given training data, and a Generative Adversarial Network (GAN) is a representative image generative model based on deep learning [32]. Since 2014, the GAN is first introduced by Goodfellow, et al. [9], it has been modified and improved for various image manipulation tasks including realistic image synthesis [34,35], image editing [12,19,22,31], image-to-image translations [14,17,26,38], etc.

Although GANs also show higher performance with a larger number of input data just like any deep-learning models [4,24,36], some versions of GANs shown outstanding performances with very limited data [11,25,29,33]. These researches mainly focus on modifying networks.

Increasing the amount of industrial data using GANs would help us developing machine vision inspection systems. In this article, artificial sealing images are generated using three GANs - DCGAN (Deep Convolutional GAN) [26], SinGAN [29], and BEGAN (Boundary Equilibrium GAN) [2] and the input data is a small number of sealing images. We will evaluate those models by FID (Fréchet Inception Distance) and SIFID (Single Image FID) [13, 29]. This research will provide a guide to use GANs to supplement datasets not only for industrial sites but also for many cases with a lack of training data.

2 Methods

It is not easy to obtain images from industrial sites with different light conditions and camera positions. Therefore, the goal is developing a deep learning model to

augment learning data with a stable performance using GANs. In this research, three GANs - DCGAN, BEGAN, and SinGAN - are used to generate sealing images from a few input images, and compared in an aspect of the quality of the generated images. Modifications of Inception Score (IS) [28] are used to evaluate the performance of the generated images using each network of three GANs. The Fréchet Inception Distance (FID) [13] is calculated for evaluating images from DCGAN and BEGAN, and SIFID [29] is used for images from SinGAN.

2.1 Generative Adversarial Networks

DCGAN The DCGAN first succeeded in constructing images using CNNs. Moreover, its process is stable. Consequently, many recent researchers compare their model based on it [1, 16]. The DCGAN achieved its goal by adopting some changes to CNN architectures: no pooling layer, no fully connected layer and applying batch normalization which is newly demonstrated in those days. These techniques helped a stable procedure. The discriminator distinguished real and fake images by a classifier on the convolutional features, while the generator upsampled images by transpose convolutional layers.

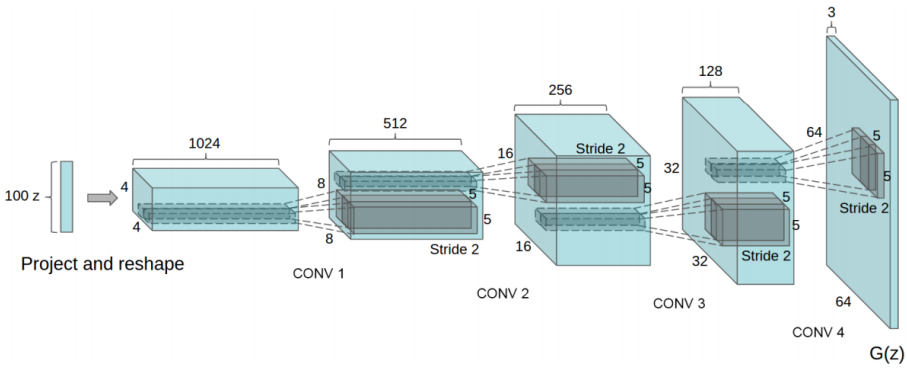


Fig. 1: DCGAN generator [26]. A 100-dimensional uniform distribution z is projected into a spatial convolutional representation with many feature maps, resulting in a higher dimension 64×64 pixel image through four fractionally-strided convolution layers.

BEGAN It is a variation of EBGAN [37] which first built a discriminator with an autoencoder. The two models ultimately try to match autoencoder loss distributions of real and fake. BEGAN is different from its predecessor in that it measures the gap between the two-loss distributions using the Wasserstein distance.

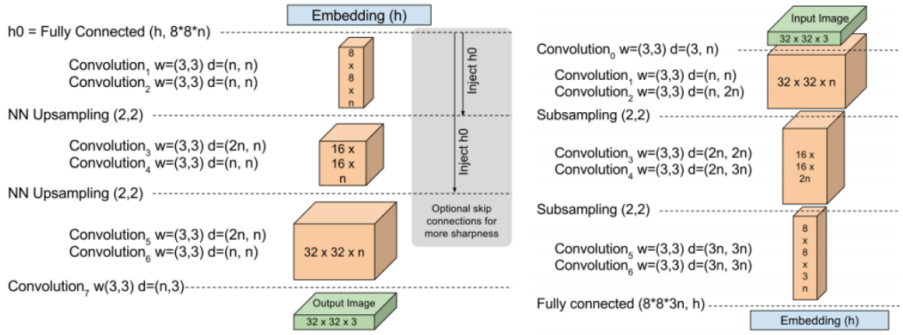


Fig. 2: BEGAN's architecture for the generator and discriminator [2]. The 3×3 convolutions with exponential linear units (ELUs) are applied to each layer. The convolution filters increase linearly with each down-sampling. At the boundary between the encoder and the decoder, the tensors of the processed data are mapped through fully connected layers.

Let x be real images and z be random noises. Let z_D and z_G are random samples from z. Let $\mathcal{L}$ be the loss of an auto-encoder. Then given a parameter θ_D for a discriminator and θ_G for the generator, the losses of D and G are defined as

$$\begin{cases} \mathcal{L}_D = \mathcal{L}(x;\theta_D) - \mathcal{L}(G(z_D;\theta_G);\theta_D) & \text{for } \theta_D \\ \mathcal{L}_G = -\mathcal{L}_D & \text{for } \theta_G \end{cases} \tag{1}$$

Note that the discriminator has two roles: auto-encode the real data and distinguish between real and generated images. The research also suggests a new formula of equilibrium to keep the balance between the discriminator and generator. D and G are considered to be at equilibrium when

$$\mathbb{E}[\mathcal{L}(x)] = \mathbb{E}[\mathcal{L}(G(z))] \tag{2}$$

The researchers defined a new hyper-parameter $\gamma \in [0, 1]$ to relax the equilibrium and called it the diversity ratio. The parameter allows us to adjust the two tasks of the discriminator.

$$\gamma = \frac{\mathbb{E}[\mathcal{L}(x)]}{\mathbb{E}[\mathcal{L}(G(z))]} \tag{3}$$

Consequently, the overall objective of BEGAN is:

$$\begin{cases} \mathcal{L}_D = \mathcal{L}(x) - k_t \cdot \mathcal{L}(G(z_D)) & \text{for } \theta_D \\ \mathcal{L}_G = \mathcal{L}(G(z_G)) & \text{for } \theta_G \\ k_{t+1} = k_t + \lambda_k(\gamma\mathcal{L}(x) - \mathcal{L}(G(z_G))) & \text{for each training step } t \end{cases} \tag{4}$$

where λ_k is a learning rate for k.

The BEGAN has an advantage of its simpler architecture relative to former GANs. It avoids conventional GAN tricks such as batch normalization or trans-pose convolution. Also, we do not have to train D and G alternately. Furthermore, it converges fast and stably using its convergence measure:

$$\mathcal{M}_{global} = \mathcal{L}(x) + |\gamma\mathcal{L}(x) - \mathcal{L}(G(z_G))| \tag{5}$$

SinGAN This method is to learn an unconditional generative model that captures the internal statistics of a single training image. To do this, it captures global properties such as the arrangement and shape of objects in the image, as well as fine details and texture information.

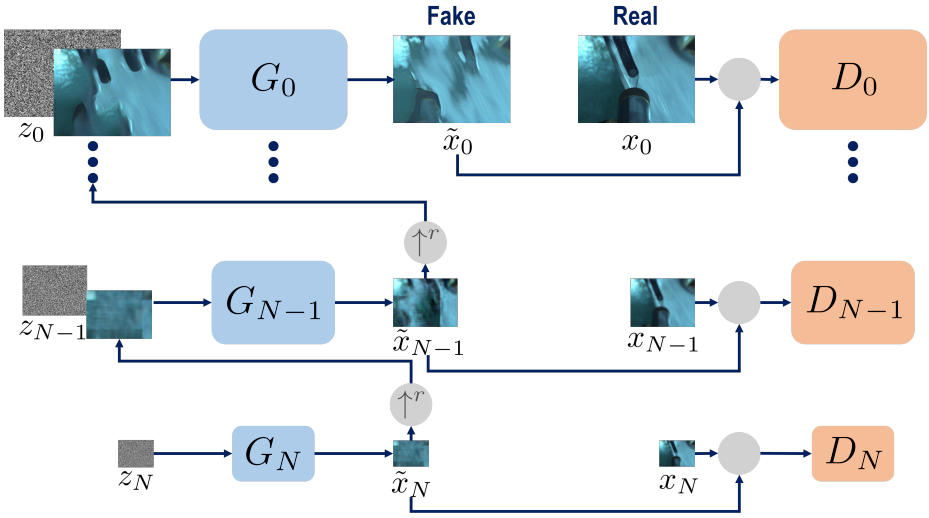


Fig. 3: SinGAN's multi-scale pipeline. Architecture consists of a pyramid of GANs, where both training and inference are done in a coarse-to-fine fashion.

As shown in Fig. 3, this model has a pyramid structure, where $x0$ is a training image, down-sampling by a factor $r^n(r > 1)$ as step by step. At each scale, the generator combines noise and the resulting image from the previous step, and the discriminator at the current step is trained to distinguish the down-sampled GT from the real image. The generator sequentially constructs images from the coarsest scale to the finest scale and the noise is added at every scale. At the coarsest scale, the generation is purely generative, i.e. G_N maps spatial white Gaussian noise z_N to an image sample $\tilde{x}_N$,

$$\tilde{x}_N = G_N(z_N). \tag{6}$$

Each of the generators G_n at finer scales ()$n < N$) adds details that are not generated by the previous scales. Thus, in addition to spatial noise z_n, each generator G_n accepts an upsampled version of an image from the coarser scale, i.e.,

$$\tilde{x}_n = G_n(z_n, (\tilde{x}_{n+1}) \uparrow^r), \quad n < N. \tag{7}$$

Learning is similar to learn traditional GANs. Training loss for nth GAN consists of adversarial term and a reconstruction term,

$$\min_{G_n} \max_{D_n} \mathcal{L}_{adv}(G_n, D_n) + \alpha \mathcal{L}_{rec}(G_n). \tag{8}$$

The WGAN-GP loss [10] for adversarial loss $\mathcal{L}_{adv}$ is used. The adversarial loss penalized for the distance between the distribution of patches in x_n and the distribution of patches in generated samples $\tilde{x}_n$. Reconstruction loss $\mathcal{L}_{rec}$ aims to reduce the pixels difference between the generated image and the down-sampled (GT) image at each scale by using the squared loss.

2.2 Evaluation

Inception Score It is one of the most widely used methods to assess the quality of generated images. The desirable outcome of generation is sampled containing meaningful objects from diverse class labels. Salimans et al. [28] proposed an approach to combine this requirement. They used a pre-trained Inception Network [30] on the ImageNet [5] to the generated samples to obtain the conditional label distribution $p(y|\boldsymbol{x})$. If it has low entropy, the generated images contain meaningful objects. Next, they calculate the marginal distribution $p(y)$ from all sample images. When various images are generated, the marginal label distribution has high entropy. Finally, the score is the expectation of KL-divergence between $p(y|\boldsymbol{x})$ and $p(y)$.

$$\text{IS} = \exp(\mathbb{E}_{\boldsymbol{x} \sim p_q} D_{KL}(p(y|\boldsymbol{x})||p(y))) \tag{9}$$

Fréchet Inception Distance The disadvantage of the Inception score (IS) is that the statistics of real generated samples are not used, and compared with the statistics of synthetic samples. The Fréchet Inception Distance (FID) [13] proposed to improve on the IS. It is a metric for evaluating GAN measures the deviation between deep features of the generated images and that of the real samples. The FID score is then calculated using the following equation:

$$\text{FID}^2 = ||\mu_r - \mu_g||^2 + Tr(\Sigma_r + \Sigma_g - 2(\Sigma_r \Sigma_g)^{1/2}), \tag{10}$$

where μ_r and μ_g refer to the feature-wise mean of the real and generated images. The Σ_r and Σ_g are the covariance matrix for the real and generated feature vectors. $X_r \sim \mathcal{N}(\mu_r, \Sigma_r)$ and $X_g \sim \mathcal{N}(\mu_g, \Sigma_g)$ are the 2048-dimensional activation of the Inception Network pool3 layer for real and generated samples respectively.

Single Image Fréchet Inception Distance Shaham et al. [29] proposed the Single Image FID (SIFID) metric. Instead of using the activation vector after the last pooling layer in the Inception Network, they use the internal distribution of deep features at the output of the convolutional layer just before the second pooling layer. SIFID is the FID between the statistics of those features in the real image and the generated sample.

3 Results and Discussion

We use 40 real sealing images obtained from an automotive manufacturing plant to generate artificial sealing images using three GANs DCGAN, BEGAN, and SinGAN.

Fig. 4: Representative images of the real sealing images. A sealing gun (a green arrow) is moved from left to right direction, and the sealant (a blue arrow) is loaded on the surface. Each image size is 658×490.

Fig. 4 shows 6 representative images of the real sealing images. A sealing gun (green arrow) is moved from left to right, and the sealant (blue arrow) is loaded on the surface with a shape of continuously linked circles. Since a camera is attached to above of the sealing gun, various types of sealing images are obtained by the angles between the sealant loaded material and the sealing gun. While the gun is moving, the shapes of sealant in images are maintained, but the illumination of images is affected by the angles of sealing gun, light changes, the angle changes of input light, etc.

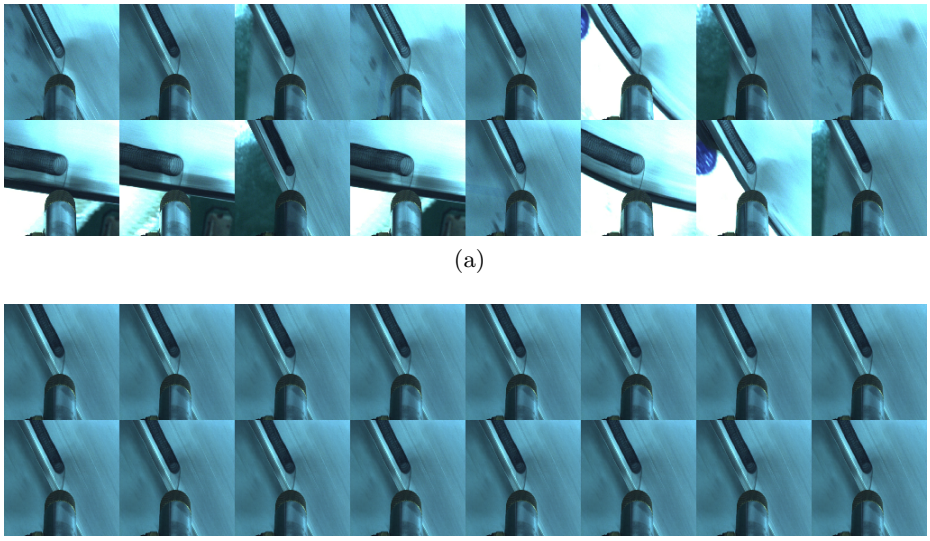
Fig. 5 shows the result of generated image using DCGAN. For the DCGAN, we resize the generated images into the 229×229. In Fig. 5, some of the sealant shows vague boundaries, and most of the backgrounds are not clear. In this case,

the performance of classification with generated images can be debased, and it could lead to the low performance of the inspection system using deep-learning. On the other hand, the generated images using DCGAN reflect the patterns of the input images and maintain the diversity of the real images. Among 16 images in Fig. 5, 9 images show the sealant is loaded from the left-up to the right-down, and the other 7 images present the sealant is laid in a horizontal direction from left to right.

Fig. 5: Samples of the generated images using DCGAN.

In Fig. 6, the result of BEGAN is likely to be mode collapse. It is one of the problems with GANs that has not been solved yet. For example, when input image distributions show 0 to 9, generated images also required to have the same distribution as the input images. However, the mode collapse generates only one of the easiest numbers, such as 1, from the training data due to the model sinks into only one mode. Although various input images are used, the results seem to produce almost on an image. When mode collapse occurs, the variety of generated images is not enough, but the quality of a single image itself can be quite good.

For SinGAN, we set the minimal dimension at the coarsest scale to 25px and chose the number of scales N s.t. the scaling factor r is as close as possible to 4/3. We resize the training image to maximal dimension 250px. The generated



(a)

(b)

Fig. 6: The samples for BEGAN. (a) Randomly selected input images. (b) Generated images using BEGAN.

images are not resized from the real images. We resize the training image to maximal dimension 250px. The generated images are not resized from the real images. Fig. 7 shows a sample of the generated image using SinGAN. SinGAN generates one artificial image from only one input image, and the three right columns are random samples and the left-most column shows input images in Fig. 7. SinGAN can be trained to capture the internal distribution of patches within an image and then generate a variety of high-quality samples that deliver the same visual content as the image [29]. Thus, when images are generated using SinGAN from three different input images, high-quality images are obtained by image distribution of input images.

For quantifying the performances of each GANs with the limited images, the qualities of the generated images are evaluated. The FID and SIFID are used for comparing the performances of three GANs. To compute the FID and SIFID scores, we use the Inception model of pre-trained on ImageNet.

Table 1 shows the FID scores of DCGAN and BEGAN. The lower value of the FID score means the better image quality, and the FID score of DCGAN is lower than that of BEGAN. The FID score contains image quality and diversity. Generated images by DCGAN have lower quality than real images, but reflect the diversity of real images, and show relatively low FID score. On the other hand, obtained images by BEGAN show a similar image quality to real images with a sufficient number of iterations but lose the diversity due to the mode collapse. As a result, it is assumed that they show higher FID score even with the high quality images.

Table 1: Fréchet Inception Distance (FID). Smaller is better.

Methods	Epoch/Iteration	Noise	Best FID	Image Size
DCGAN	300 (epoch)	Gaussian	52.27	658×490
BEGAN	100,000 (iter)	Uniform	169.97	128×128

The SinGAN is a network that generates images using the internal distribution of the input images. With its pyramid structure, the inputs in each level are affected by the previous level. Inference at N scale means generation from noise, and inference at $N-1$ scale means down-sampling of the input image and putting it as the input of the $N-1$ generator. An image with a shape and array similar to the input image is created at scale N. Consequently, the larger scale leads to the better quality of the images. As shown in Table 2, the average SIFID is lower for generation from scale $N-1$ than for generation from scale N. It means the image quality of $N-1$ is better than that of N, although the difference is small.

Table 2: Single Image FID (SIFID) for SinGAN.

Scale	SIFID
N	0.1750302
N-1	0.1750298

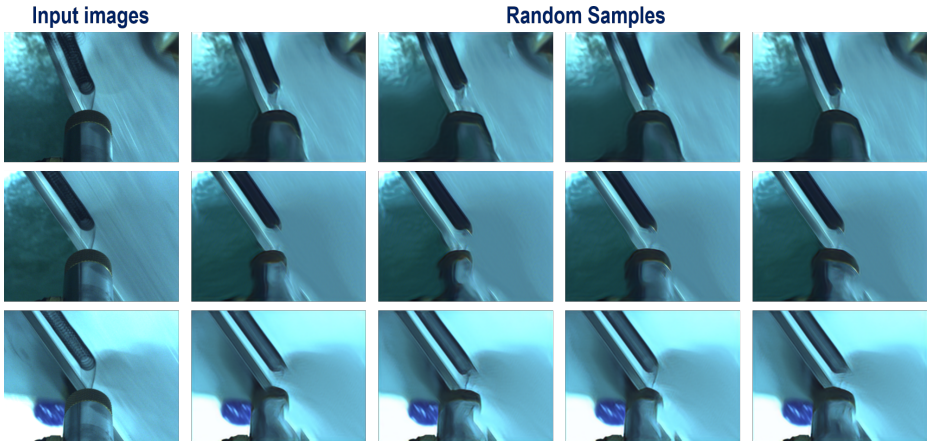


Fig. 7: Samples of the generated images using SinGAN.

In fact, this method has several drawbacks to compare GANs for generating sealing images with very limited input images. Firstly, one uniformed system is not used to evaluate three GANs, and the direct comparison of them is not possible. DCGAN and BEGAN are compared with FID scores, and SinGAN is evaluated with SIFID. Also, the evaluation methods possess limitations. An evaluation method for GANs has been controversial. A verified system for evaluating generated images dose not exist yet. In this situation, modified inception scores are the most commonly used evaluating methods, but they need to be used with more than 5 thousand images from ImageNet. However, generating images with very limited industrial data is meaningful in academically and industrially. Moreover, the direct comparison of FID scores between DCGAN and BEGAN provides the significant information of the image quality and diversity for generating images using GAN to develop the machine vision inspection system.

While maintaining high-quality images, providing diversity to artificial sealing image using illumination changes can be tried as further work. In case of sealing images, variations among obtained images are very small. However, illumination condition can be changes, and especially, illumination changes make significant impact to images. Using the Lambertian properties [18] which give same amount of light to an observer at any angle, a surface brightness can be controlled by altering a surface slope. This can be applied to an input of a generation network for the sealing image generation by illumination changes. Then, image quality will be maintained and a robust model will be obtained against the mode collapse [23].

4 Conclusion

This paper present a comparative study of the performance of GANs for small-scale data. In qualitative results, SinGAN and BEGAN results are better than DCGAN. In the case of BEGAN, however, only one type of sample is generated due to mode collapse. In order to solve this problem, we expect to add more optical conditions such as Lambertian's law in the future to generate more diverse samples using the input of the generator. In quantitative evaluation, DCGAN, the basic form of GAN, is the best. In the case of SinGAN, a new image can be generated from a single image, but it can be found that it is not easy to apply in the industrial sites. It is not easy to evaluate because there is no standardized method of evaluating the results of GANs. A quantitative evaluation method for GANs is needed for further research. We use only three kinds of GANs for comparing the result by the effects of limited data. It is not enough to compare the effects of GANs. Our future work will be compared to the results by adding more GANs methods such as Transferring-Gans [33], DeLiGAN [11], StyleGAN [15], LSGAN [21], WGAN-GP [10], BigGAN [4], etc.

Acknowledgement

This work was supported by the Technology development Program(S2760246) funded by the Ministry of SMEs and Startups(MSS, Korea).

References

1. Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein GAN (2017), arXiv:1701.07875
2. Berthelot, D., Schumm, T., Metz, L.: BEGAN: Boundary Equilibrium Generative Adversarial Networks (2017), arXiv:1703.10717
3. Beyerer, J., León, F., Frese, C.: Machine Vision: Automated Visual Inspection: Theory, Practice and Applications. Springer, Berlin, Heidelberg (2015)
4. Brock, A., Donahue, J., Simonyan, K.: Large Scale GAN Training for High Fidelity Natural Image Synthesis (2018), arXiv:1809.11096
5. Deng, J., Dong, W., Socher, R., Li, L., Kai Li, Li Fei-Fei: ImageNet: A large-scale hierarchical image database. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 248–255 (June 2009)
6. Everingham, M., Eslami, S.M.A., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The Pascal Visual Object Classes Challenge: A Retrospective. International Journal of Computer Vision (IJCV) **111**(1), 98–136 (jan 2015)
7. Gauen, K., Dailey, R., Laiman, J., Zi, Y., Asokan, N., Lu, Y., Thiruvathukal, G.K., Shyu, M., Chen, S.: Comparison of Visual Datasets for Machine Learning. In: IEEE International Conference on Information Reuse and Integration (IRI). pp. 346–355 (Aug 2017)
8. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press (2016)
9. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative Adversarial Nets. In: Advances in Neural Information Processing Systems (NIPS). pp. 2672–2680 (2014)
10. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.C.: Improved Training of Wasserstein GANs. In: Advances in Neural Information Processing Systems (NIPS). pp. 5767–5777 (2017)
11. Gurumurthy, S., Sarvadevabhatla, R.K., Babu, R.V.: DeLiGAN: Generative Adversarial Networks for Diverse and Limited Data. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4941–4949 (July 2017)
12. He, K., Sun, J.: Statistics of Patch Offsets for Image Completion. In: European Conference on Computer Vision (ECCV). pp. 16–29 (2012)
13. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. In: Advances in Neural Information Processing Systems (NIPS). pp. 6626–6637 (2017)
14. Isola, P., Zhu, J., Zhou, T., Efros, A.A.: Image-to-Image Translation with Conditional Adversarial Networks. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5967–5976 (July 2017)
15. Karras, T., Laine, S., Aila, T.: A Style-Based Generator Architecture for Generative Adversarial Networks. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4396–4405 (June 2019)
16. Karras, T., Aila, T., Laine, S., Lehtinen, J.: Progressive Growing of GANs for Improved Quality, Stability, and Variation (2017), arXiv:1710.10196

17. Karras, T., Laine, S., Aila, T.: A Style-Based Generator Architecture for Generative Adversarial Networks (2018), arXiv:1812.04948
18. Koppal, S.J.: Lambertian Reflectance, pp. 441–443. Springer US, Boston, MA (2014)
19. Lempitsky, V., Vedaldi, A., Ulyanov, D.: Deep Image Prior. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 9446–9454 (June 2018)
20. Lin, T.Y., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick, C.L., Dollr, P.: Microsoft coco: Common objects in context (2014), arXiv:1405.0312
21. Mao, X., Li, Q., Xie, H., Lau, R.Y.K., Wang, Z., Smolley, S.P.: Least Squares Generative Adversarial Networks. In: IEEE International Conference on Computer Vision (ICCV). pp. 2813–2821 (Oct 2017)
22. Mechrez, R., Shechtman, E., Zelnik-Manor, L.: Saliency Driven Image Manipulation. In: IEEE Winter Conference on Applications of Computer Vision (WACV). pp. 1368–1376 (March 2018)
23. Metz, L., Poole, B., Pfau, D., Sohl-Dickstein, J.: Unrolled Generative Adversarial Networks (2016), arXiv:1611.02163
24. Miyato, T., Kataoka, T., Koyama, M., Yoshida, Y.: Spectral Normalization for Generative Adversarial Networks (2018), arXiv:1802.05957
25. Osokin, A., Chessel, A., Salas, R.E.C., Vaggi, F.: GANs for Biological Image Synthesis (2017), arXiv:1708.04692
26. Radford, A., Metz, L., Chintala, S.: Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks (2015), arXiv:1511.06434
27. Roh, Y., Heo, G., Whang, S.E.: A Survey on Data Collection for Machine Learning: a Big Data – AI Integration Perspective (2018), arXiv:1811.03402
28. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X., Chen, X.: Improved Techniques for Training GANs. In: Lee, D.D., Sugiyama, M., Luxburg, U.V., Guyon, I., Garnett, R. (eds.) Advances in Neural Information Processing Systems 29. pp. 2234–2242 (2016)
29. Shaham, T.R., Dekel, T., Michaeli, T.: SinGAN: Learning a Generative Model From a Single Natural Image. In: IEEE International Conference on Computer Vision (ICCV). pp. 4570–4580 (2019)
30. Szegedy, C., Wei Liu, Yangqing Jia, Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1–9 (June 2015)
31. Taeg Sang Cho, Butman, M., Avidan, S., Freeman, W.T.: The Patch Transform and Its Applications to Image Editing. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1–8 (June 2008)
32. Turhan, C.G., Bilge, H.S.: Recent Trends in Deep Generative Models: a Review. In: International Conference on Computer Science and Engineering (UBMK). pp. 574–579 (Sep 2018)
33. Wang, Y., Wu, C., Herranz, L., van de Weijer, J., Gonzalez-Garcia, A., Raducanu, B.: Transferring GANs: Generating Images from Limited Data. In: European Conference on Computer Vision (ECCV). pp. 220–236 (2018)
34. Zhang, H., Xu, T., Li, H., Zhang, S., Wang, X., Huang, X., Metaxas, D.: StackGAN: Text to Photo-Realistic Image Synthesis with Stacked Generative Adversarial Networks. In: IEEE International Conference on Computer Vision (ICCV). pp. 5908–5916 (Oct 2017)

35. Zhang, H., Xu, T., Li, H., Zhang, S., Wang, X., Huang, X., Metaxas, D.N.: Stack-GAN++: Realistic Image Synthesis with Stacked Generative Adversarial Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence **41**(8), 1947–1962 (Aug 2019)
36. Zhang, H., Goodfellow, I., Metaxas, D., Odena, A.: Self-Attention Generative Adversarial Networks (2018), arXiv:1805.08318
37. Zhao, J., Mathieu, M., LeCun, Y.: Energy-based Generative Adversarial Network (2016), arXiv:1609.03126
38. Zhu, J., Park, T., Isola, P., Efros, A.A.: Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks. In: IEEE International Conference on Computer Vision (ICCV). pp. 2242–2251 (Oct 2017)