Online Adversarial Knowledge Distillation for Image Synthesis of Bridge Defect

Jiongyu Guo
College of Software, Zhejiang University, Hangzhou, China
jy.guo@zju.edu.cn

Can Wang*
College of Computer Science, Zhejiang University, Hangzhou, China
wcan@zju.edu.cn

Yan Feng
College of Computer Science, Zhejiang University, Hangzhou, China
fengyan@zju.edu.cn

ABSTRACT

Bridge defect detection is an essential task of its daily maintenance, which aims to protect people's life and property safety. However, for a variety of reasons, research institutions have been faced with the scarcity of anomaly samples. One solution is using generative adversarial network (GAN) to generate extra samples for data augmentation. In this paper, we draw on the idea from online knowledge distillation to improve the self-attention GAN, and propose a new framework called Online Knowledge Distillation -Self Attention Generative Adversarial Network (OKD-SAGAN). We introduce a new module called connector which has the same structure with discriminator to train multiple groups of SAGAN together. The role of the connector is to control the output distribution of the corresponding generator to be consistent with the surrounding generators in order to achieve the purpose of mutual learning. We have conducted experiments on the CODEBRIM dataset and in order to further illustrate the effectiveness of OKD structure, we also applied OKD on ACGAN for experiments. The results show that the performance of some generators has exceeded a single set of SAGAN and ACGAN. Compared with SAGAN, OKD-SAGAN G_2's FID score decreases by 15.4% and the average FID score decreases by 5.5%. As for ACGAN, OKD-ACGAN G_1's FID score decreases by 7.6% and the average FID score decreases by 3.8%, which proves the validity of OKD structure.

CCS CONCEPTS

• **Computing methodologies**; • **Artificial intelligence**; • **Computer vision**;

KEYWORDS

Generative adversarial network, Image generation, Online knowledge distillation

ACM Reference Format:
Jiongyu Guo, Can Wang*, and Yan Feng. 2021. Online Adversarial Knowledge Distillation for Image Synthesis of Bridge Defect. In *The 5th International Conference on Computer Science and Application Engineering (CSAE 2021), October 19–21, 2021, Sanya, China.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3487075.3487171

CSAE 2021, October 19–21, 2021, Sanya, China

ACM ISBN 978-1-4503-8985-3/21/10…$15.00
https://doi.org/10.1145/3487075.3487171

1 INTRODUCTION

In the field of transportation infrastructure, the trend of integrating roads, bridges, tunnels, geotechnical and other infrastructure construction maintenance with modern science, especially with artificial intelligence, knowledge graph and other technologies has become increasingly popular. Studies have shown that assisted by artificial intelligence, bridge intelligent operation and maintenance has greatly improved efficiency [1]. Besides, the building state assessment method based on knowledge graph technology [2] and bridge maintenance knowledge base system [3] can greatly reduce facility maintenance costs. Intelligent maintenance of transportation infrastructure has huge market demand. However, in traditional machine learning, in order to ensure the accuracy and high reliability of the classification model, enough available training samples are necessary. Due to the advanced design of the bridge, strict construction, and short time to come into service, the knowledge discovery of defect of the bridge also faces the problem of lacking in samples. Therefore, exploring sample augmentation methods to expand the sample size of bridge defect has extremely important value.

Image synthesis is an important technology in artificial intelligence. Since the introduction by Goodfellow et al. [4] in 2014, Generative Adversarial Network (GAN) has become the mainstream method of image synthesis. By alternately training a generator and a discriminator, GAN can finally get a well-behaved generator which can generate fake samples with similar distribution to real samples.

In recent years, in order to overcome some serious problems in the training of the original GAN, many variants [5-10] have sprung up. Among them, Self-Attention GAN [10] has achieved great success by combining a variety of GAN training tricks and adding the self-attention mechanism. SAGAN used the feature maps with self- attention to replace the traditional convolution feature maps which made SAGAN be able to obtain larger receptive fields while maintaining computational efficiency and complexity. Compared with several GAN models [5, 6], SAGAN got the best performance on ImageNet [11].

We cannot help thinking that a set of generator and discriminator can achieve such good results, so how about several sets? Inspired by online knowledge distillation [12, 13], we propose a new structure called OKD-SAGAN to regard multiple generators and their associated discriminators as student networks and let them learn from each other. OKD-SAGAN adds a new module called the connector between all the generators. The connector and the discriminator have the same structure but different tasks. The discriminator is used to control the distribution of the fake samples close to the ground truth, but the connector is used to balance the output distribution between the generators. We have designed experiments on the CODEBRIM [14] dataset to verify our ideas. The results show that some generators have achieved better performance than the single set of SAGAN.

2 RELATED WORK

Data Augmentation. Data plays an extremely critical role in deep learning tasks. The training of a deep learning model always requires a lot of data. However, in some specific tasks, it is difficult for researchers to obtain a large amount of data for various reasons, which makes the effect of the model unsatisfactory. Data augmentation technology can expand data for datasets. For the time-series data, Wen Q [15] et al. proposed three data augmentation methods: decomposition-based methods, statistical generative models and learning-based methods and further preformed experiments on practical tasks such as classification and prediction. For image data, some simple methods are often applied to some easy tasks, such as flipping, rotation, scale, cropping and translation. Cherry Khosla [16] et al. proposed a variety of data augmentation techniques based on data distortion and oversampling, and further reduce the problem of data over fitting to a certain extent. Gan and its variants are more advanced data enhancement technologies. They can not only expand the data similar to the datasets, but also realize a variety of data transformations such as style migration, seasonal transformation and so on.

Generative Adversarial Networks. The process of training GAN is not easy. It has to face the following two fatal problems: (1) mode collapse and (2) divergence. In response to the difficulty in GAN training, recently, lots of tricks [5, 7, 17] have been proposed. Compared with GAN, WGAN [7] made the following four simple changes: (1) removing the sigmoid function in the last layer of the discriminator, (2) removing 'log' from the loss function of generator and discriminator, (3) gradient clipping or gradient penalty [17] after the update of discriminator, (4) using RMSProp or SGD optimizer rather than Adam. Although WGAN was a refinement of GAN, WGAN did not completely solve the 1-Lipschitz problem. SNGAN [5] solved the remaining problems of WGAN and proposed a method of 1-Lipshcitz constraint. Besides, utilizing imbalanced learning rates (TTUR) uses different learning rates for the discriminator and generator. Generally, the discriminator needs to be updated more frequently than the generator. By using this method, we only need to adjust the learning rate of the generator and the discriminator to let them update at a ratio of 1:1. In defect images generation, Zhang G et al. [18] proposed Defect-GAN, which synthesized the normal bridge images and the bridge defect images. For image-to-image conversion methods, they choose Star GAN [19] and SPADE [20] as competitive methods. Through training, Defect-GAN can depict defects on normal bridge images and achieve very good results.

Our model is based on SAGAN. In order to learn the global features, SAGAN adds a self-attention module in the generator and discriminator when using above tricks. Differently from classical GANs, we get inspiration from knowledge distillation and co-train several sets of generators and discriminators.

Knowledge Distillation. Knowledge distillation is widely applied in model compression and transfer learning. Hinton [21] first proposed the concept of knowledge distillation, which used the soft labels extracted from the pre-trained teacher network as a part of the total loss of the student network to guide its training and finally realized the transfer of knowledge from the large(teacher) network to the small(student) network. Soft labels contain rich information of teacher network, which can improve the generalization ability and robustness of student network. There are many variants of knowledge distillation, such as offline knowledge distillation [21, 22], online knowledge distillation [12, 13, 23] and self-knowledge distillation [24, 25]. Compared with offline knowledge distillation, online knowledge distillation has higher efficiency without a pre-trained teacher network. On the contrary, the teacher network and student network are updated at the same time. In Deep Mutual Learning (DML) [12], all neural networks have the same structures. They train together, make progress together, and use soft labels to 'exchange experience'. Chen D et al. [23] added multiple auxiliary peers and one group leader in the training process to perform two-level distillation, which exceeded state-of-the-art models without increasing the complexity of training. In addition to class probabilities, AFD [13] also considers feature maps that contain rich image information. AFD proposes an online knowledge distillation method based on the adversarial training framework to exchange feature map information.

3 METHOD

3.1 SAGAN

Although convolutional neural networks (CNN) can help GAN generate high-quality images, due to the limitation of the local receptive field of CNN, GAN cannot balance global information to make the generated images appear coordinated. The traditional method is to increase the size of the convolution kernel or deepen the network layers which will increase the amount of calculation and parameters. SAGAN uses the self-attention feature maps to replace the traditional convolution feature maps. Suppose now we get output $x \in R^{C\times N}$ from the previous layer, let

$$f(x) = W_f x\ ,\ g(x) = W_g(x)\ ,\ h(x) = W_h(x) \tag{1}$$

where $W_f \in R^{\bar{C}\times C}, W_g \in R^{\bar{C}\times C}, W_h \in R^{C\times C}$ are weight matrices composed by 1×1 convolution kernels. C is the number of channels and N is the number of features. We use $\beta_{i,j}$ to indicate the contribution degree of the position i when synthesizing the area j

$$\beta_{i,j} = \frac{e^{s_{i,j}}}{\sum_{i=1}^{N} e^{s_{i,j}}}, where\ s_{i,j} = f(x_i)^T g(x_j) \tag{2}$$

Next, we get the output of the self-attention layer $O = (o_1, o_2, \cdot s, o_N) \in R^{C\times N}$ and

$$o_i = \sum_{j=1}^{N} \beta_{i,j} h(x_j) \tag{3}$$

The final output is

$$y_i = \gamma o_i + x_i \tag{4}$$

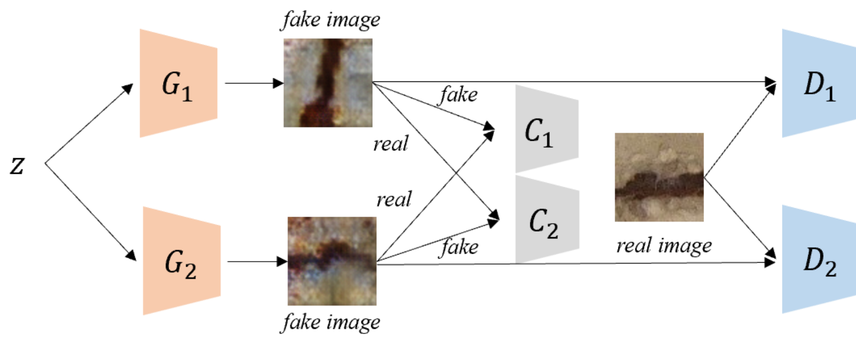

Figure 1: Overall Structure of OKD-SAGAN. For the Connector, Its Task Is to Identify Fake Images from Different Generators. For the Generator, It Not only Needs to Fool the Corresponding Discriminator, but also the Connector.

where $\gamma = 0$ at first and will update during training. Self-attention module is applied in both generator and discriminator. SAGAN is trained by minimizing the loss function of WGAN-gp version.

$$L_D = E_{z\sim P_z} D(G(z)) - E_{x\sim P_{data}} D(x) + \lambda E_{\hat{x}\sim P_{\hat{x}}} (\nabla_{\hat{x}} D(\hat{x})_2 - 1)^2 \tag{5}$$

3.2 OKD-SAGAN

Inspired by online knowledge distillation, OKD-SAGAN adds multiple sets of generators and discriminators and let them train together. The structure is shown in figure 1. In the process of training, each set of SAGAN is not isolated. On the contrary, they will exchange experience with each other. OKD-

SAGAN has the same structure of generator and discriminator with SAGAN, but further use a module called connector to balance the output of all generators. The connector has the same structure (see figure 2) with the discriminator, but they have completely different tasks. Generally, the discriminator is used to distinguish fake images from true images to encourage the generator to generate more realistic images. For the connector, its task is to make the images generated by all generators have a closer distribution. So how to realize this function? Actually, it is very similar to the principle of the discriminator. Suppose we have two different generators(G_1 , G_2) and two different connectors (C_1, C_2). For the connector C_1, the output of G_1 is regarded as a fake and the output of G_2 is classified as a true and do vice versa for $\boldsymbol{C}_2$. $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$ are supposed to learn to distinguish the outputs of $\boldsymbol{G}_1$ and $\boldsymbol{G}_2$. However, for the generator $\boldsymbol{G}_1$ and $\boldsymbol{G}_2$, they need to find ways to fool the corresponding connector and make them misjudge. In this adversarial training, the generator $\boldsymbol{G}_1$ and $\boldsymbol{G}_2$ will generate images with the distribution as close as possible. In the knowledge distillation scene, each network has logit-based distillation loss for classification tasks. However, there is no logit-based loss in image synthesis. Actually, the connector has solved this problem. For example, the connector $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$ are trained to distinguish the output of $\boldsymbol{G}_1$ and $\boldsymbol{G}_2$, which means the connector can reflect the difference in generation distribution between different generators. This is consistent with the purpose of logit-based. For more detailed analysis, see section 3.3.

3.3 Lost Function

Suppose we have OKD-SAGAN with N sets of SAGAN, now we will discuss the loss functions of generator, discriminator and connector separately.

The lost function of discriminator. In OKD-SAGAN, the task of the discriminator is consistent with SAGAN, so its loss function has not changed.

$$L_{D_i} = E_{z\sim P_z} D_i(G_i(z)) - E_{x\sim P_{data}} D_i(x) + \lambda E_{\hat{x}\sim P_{\hat{x}}} (\nabla_{\hat{x}} D_i(\hat{x})_2 - 1)^2 \tag{6}$$

where $i = 1, 2, \cdot sN$.

The lost function of generator. In OKD-SAGAN, the generator G_i faces two tests: the detection of the discriminator and the connector. The images it generates must be similar to the real images and the images generated by the other generators to fool the discriminator and the connector simultaneously. Therefore, on the basis of the original, the loss function of the generator needs to add the item about the connector.

$$L_{G_i} = -E_{z\sim P_z} D_i(G_i(z)) - \rho E_{z\sim P_z} C_i(G_i(z)) \tag{7}$$

where $i = 1, 2, \cdot sN$, ρ is a hyperparameter used to control the degree of influence among generators.

The lost function of connector. The connector is the innovation of this paper. It connects several sets of SAGAN for collaborative training by identifying images from different generators. The connector's working principle is similar to the discriminator, which results its loss function also derived from the discriminator.

$$\begin{cases} L_{C_i} = E_{z\check{}P_z} C_i(G_i(z)) - E_{z\check{}P_z} C_i(G_{i+1}(z)) + \lambda E_{z\check{}P_z} \\ (\|\nabla_z (C_i(\varepsilon G_i(z) + (1-\varepsilon) C_{i+1}(z)))\|_2 - 1)^2 \text{if } i = 1, \cdots, N-1 \\ L_{C_i} = E_{z\check{}P_z} C_i(G_i(z)) - E_{z\check{}P_z} C_i(G_1(z)) + \\ \lambda E_{z\check{}P_z} (\|\nabla_z C_i(\varepsilon G_i(z) + (1-\varepsilon) G_1(z))\|_2 - 1)^2 \text{ if } i = N \end{cases} \tag{8}$$

Next, we will explain why the connector has solved the problem of logit-based loss. Let's see the first two parts $E_{z\sim P_z} C_i(G_i(z))$ and $E_{z\sim P_z} C_i(G_{i+1}(z))$. If G_i and G_{i+1} use random vector $\boldsymbol{z}$ to generate two images with the same distribution, then

$$E_{z\sim P_z} C_i(G_i(z)) - E_{z\sim P_z} C_i(G_{i+1}(z)) = 0 \tag{9}$$

and

$$L_{C_i} = \lambda E_{z\sim P_z} (\nabla_z (C_i(\varepsilon G_i(z) + (1-\varepsilon) C_{i+1}(z)))_2 - 1)^2 \tag{10}$$

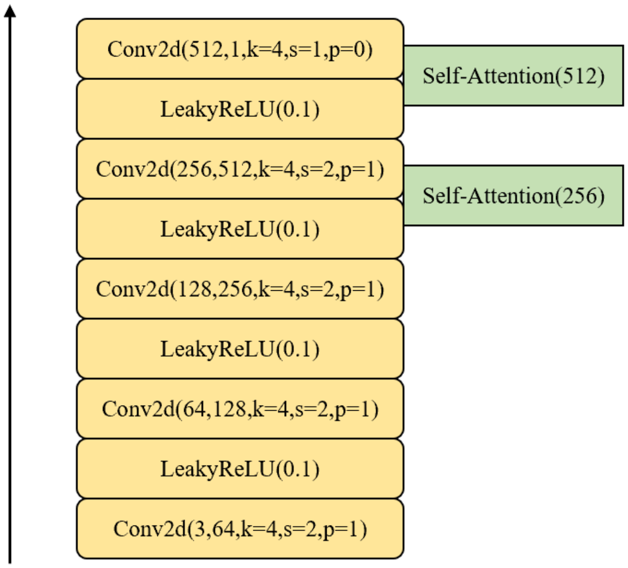

Figure 2: The Structure of Connector.

Actually, this part is a gradient penalty term introduced by WGAN-gp to satisfy the 1- Lipschitz constraint. For a well-trained connector, it's a very small number and we can consider it equal to 0 to a certain extent. In the classification task, if the prediction p_1, p_2 of the two networks are the same, then

$$L_{kl}(p_1, p_2) = 0$$
$$L_{kl}(p_2, p_1) = 0 \quad (11)$$

On the whole, they all measure the difference in the distribution of the outputs from different networks, and as the degree of dissimilarity increases or decreases, the loss functions between the two networks gradually become larger or smaller. Thus, the connector and KL divergence are different in form but equally satisfactory in results.

4 EXPERIMENT

We implement all networks and training procedures in Pytorch 1.5.0 with python3.7. The experimental environment configuration is as follows: Intel(R) Xeon(R) Gold 6254 CPU @ 3.10GHz, 8 NVIDIA GeForce GTX 2080ti (11GB) GPU.

Dataset. Our paper uses the CODEBRIM dataset. The CODEBRIM was first put forward in [14] and contains 1,600 images of apparent bridge defects. The authors photographed 30 defective bridges, and divided all photos into the following five categories: crack, exposed reinforcement bar, spallation, corrosion (stains) and efflorescence (calcium leaching). We randomly crop 2642 64×64 samples from the CODEBRIM as the dataset for this paper.

Implementation details. Our framework is compared with the single set of SAGAN [10] and ACGAN [6] on CODEBRIM dataset. All models generate 64×64 images from 1×128 random vectors. Each set of SAGAN's parameters in OKD-SAGAN are set according to the original SAGAN. We set $\beta_1 = 0$ and $\beta_2 = 0.9$. The learning rate for the discriminator is 0.0004 and the learning rate for the generator is 0.0001. Further, in OKD-SAGAN, the learning rate for the connector is 0.00005, and the number of SAGAN is 3. The batch size is 25 and total epoch is 50000 for SAGAN and OKD-SAGAN. In order to further illustrate the effectiveness of OKD structure, we also applied OKD on ACGAN for experiments. For ACGAN and OKD-ACGAN, the batch size is 5 and total epoch is 3000.

Comparison with the baseline. As we can see in figure 3, the three generators of OKD-SAGAN G_1, G_2 and G_3 generate images about bridge defect with different content. In addition, compared to SAGAN, the images generated by OKD-SAGAN are clearer and more detailed. Obviously, it's the same with OKD-ACGAN. We can see the tiny cracks on the bridge clearly from the OKD-SAGAN G_2 and OKD-ACGAN G_1. We use FID score to measure the quality of the images. In the past work, Inception score was often used as an evaluation metric. But Inception score has been pointed out to have many unstable problems [26]. FID is a better alternative to Inception score. FID is a more comprehensive and accurate evaluation metric, which has a higher reference value in evaluating the diversity and authenticity of images. By transforming real images and fake images into feature space with an Inception-v3 network, FID calculates the Wasserstein-2 distance between the two as the final score. The formula is shown as the following:

$$FID(x,g) = u_x - u_{g}{}_{2}^{2} + Tr\left(\Sigma_x + \Sigma_g - 2\left(\Sigma_x \Sigma_g\right)^{\frac{1}{2}}\right) \quad (12)$$

Where Tr is trace in matrix theory, Σ is covariance and u is mean value. The lower the FID, the closer the generated images are to the real images. For each model, we have generated 10,000 fake images to calculate their FID. The final results are shown in Table 1.

5 CONCLUSION

This paper draws on the idea from online knowledge distillation and conducts mutual distillation of multiple sets of SAGAN. We co-train all generators by introducing a module called the connector, which has the same structure with the discriminator. OKD structure has achieved amazing results on the CODEBRIM dataset. Through comparative experiments with the baseline model SAGAN, we discover that the FID scores of some generators in our approach are lower than SAGAN (**118.23 vs 139.76**), and further, it's the same with the average FID score of the three generators (**132.13 vs 139.76**), which fully demonstrates that all three generators learn from each other through the connector and make progress together. In order to further illustrate the effectiveness of OKD structure, we also carried out experiments on ACGAN. The results are equally satisfactory. Compared with ACGAN, the FID score of OKD-ACGAN G_1 decreased by 7.6%(**165.39vs178.92**) and the average FID score decreased by 3.8%(**172.13vs178.92**). Of course, the method in this paper can be extended to GANs with any structure. Future work will adjust the proportional relation between each part loss function of the generator and the connector's hyperparameters to generate clearer and more diversified images.

ACKNOWLEDGMENTS

This work is supported by National Key R&D Program of China (Grant No: 2019YFB1600700), National Natural Science Foundation of China (Grant No: U1866602).

Table 1: Comparison of the Proposed OKD-SAGAN with SAGAN and OKD-ACGAN with ACGAN on CODEBRIM Dataset

Model		FID	Average
ACGAN		178.92	
OKD-ACGAN		165.39	172.13
		171.68	
		179.31	
SAGAN		139.76	
OKD-SAGAN	G_1	144.05	132.13
	G_2	**118.23**	
	G_3	134.12	

Figure 3: 64×64 Images Randomly Generated by ACGAN, OKD-ACGAN, SAGAN and OKD-SAGAN.

During the period of writing, many people have given me help and encouragement, and I would like to say thanks to them.

Firstly, Professor Can Wang gave me hardware support, enabling me to realize this idea.

Secondly, Defang Chen from Zhejiang University gave me guidance on the paper writing and proposed many ingenious opinions on the model.

Finally, my aunt and uncle helped me solve the problem of accommodation so that I can carry out my research without any worries.

REFERENCES

[1] P. Kruachottikul *et al.* (2019). Bridge Sub Structure Defect Inspection Assistance by using Deep Learning. 2019 IEEE 10th International Conference on Awareness Science and Technology (iCAST), Morioka, Japan, pp. 1-6.

[2] Hamdan A H N, Scherer R J (2019). A knowledge-based Approach for the Assessment of Damages to Constructions[C]//Proceedings of the 36th CIB W79 2019 Conference. Newcastle, UK. 572-582.

[3] Ren, G., Ding, R., & Li, H. (2019). Building an ontological knowledgebase for bridge maintenance. Advances in Engineering Software, 130 (July 2018), 24-40.

[4] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. WardeFarley, S. Ozair, A. Courville, and Y. Bengio (2014). Generative Adversarial Networks. In NIPS.

[5] Miyato T, Kataoka T, Koyama M, *et al.* (2018). Spectral Normalization for Generative Adversarial Networks[C]//International Conference on Learning Representations.

[6] Odena A, Olah C, Shlens J (2017). Conditional image synthesis with auxiliary classifier gans[C]//International conference on machine learning. PMLR: 2642-2651.

[7] Arjovsky M, Chintala S, Bottou L (2017). Wasserstein generative adversarial networks[C]//International conference on machine learning. PMLR: 214-223.

[8] A. Radford, L. Metz, and S. Chintala (2015). Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv: 1511.06434.

[9] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel (2016). Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In Advances in Neural Information Processing Systems (NIPS), pages 2172–2180.

[10] Zhang, H., Goodfellow, I., Metaxas, D. & Odena, A. (2019). Self-Attention Generative Adversarial Networks. Proceedings of the 36th International Conference on Machine Learning, in Proceedings of Machine Learning Research 97:7354-7363

[11] J. Deng, W. Dong, R. Socher, L. Li, Kai Li and Li Fei-Fei (2009). ImageNet: A large-scale hierarchical image database. 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 248-255.

[12] Zhang, Y., Xiang, T., Hospedales, T.M., Lu, H. (2018) Deep mutual learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4320–4328.

[13] Chung I, Park S U, Kim J, *et al.* (2020). Feature-map-level online adversarial knowledge distillation[C]//International Conference on Machine Learning. PMLR: 2006-2015.

[14] Mundt M, Majumder S, Murali S, *et al.* (2019). Meta-learning convolutional neural architectures for multi-target concrete defect classification with the concrete defect bridge image dataset[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 11196-11205.

[15] Wen Q, Sun L, Yang F, *et al.* (2020). Time series data augmentation for deep learning: A survey[J]. arXiv preprint arXiv:2002.12478.

[16] Khosla C, Saini B S (2020). Enhancing Performance of Deep Learning Models with different Data Augmentation Techniques: A Survey[C]//2020 International Conference on Intelligent Engineering and Management (ICIEM). IEEE, 79-85.

[17] Gulrajani I, Ahmed F, Arjovsky M, *et al.* (2017). Improved Training of Wasserstein GANs[C]//NIPS.

[18] Zhang G, Cui K, Hung T Y, *et al.* (2021). Defect-GAN: High-fidelity defect synthesis for automated defect inspection[C]//Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. 2524-2534.

[19] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo (2018). StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation. In CVPR, pages 8789–8797.

[20] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu (2019). Semantic image synthesis with spatially-adaptive normalization. In CVPR, pages 2337–2346.

[21] Hinton G, Vinyals O, Dean J (2015). Distilling the Knowledge in a Neural Network[J]. Computer Science, 14(7):38-39.

[22] Chen D, Mei J P, Zhang Y, *et al.* (2021). Cross-layer distillation with semantic calibration[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 35(8): 7028-7036.

[23] Chen D, Mei J P, Wang C, *et al.* (2020). Online Knowledge Distillation with Diverse Peers[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 34(4): 3430-3437.

[24] Zhang L, Song J, Gao A, *et al.* (2019). Be your own teacher: Improve the performance of convolutional neural networks via self distillation[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 3713-3722.

[25] Kyungyul Kim, ByeongMoon Ji, Doyoung Yoon, and Sangheum Hwang (2020). Self-knowledge distillation: A simple way for better generalization. arXiv preprint arXiv:2006.12000 (2020).

[26] Barratt S, Sharma R (2018). A note on the inception score[J]. arXiv preprint arXiv:1801.01973.