Defect Information Synthesis via Latent Mapping Adversarial Networks

Seunghwan Song
Department of Industrial and Management Engineering
Korea University
Seoul, South Korea
ss-hwan@korea.ac.kr

Jun-Geol Baek*
Department of Industrial and Management Engineering
Korea University
Seoul, South Korea
jungeol@korea.ac.kr

***Abstract*— This research presents a new image synthesis methodology for automated visual inspection (AVI) in steel manufacturing process. We develop a novel methodology, termed Latent Mapping Adversarial Networks. As the end product of the manufacturing process is directly linked to economic factors, various methods are being utilized to improve the quality of the product. Among them, the defect detection steps carried out in advance are important as it greatly impacts productivity. However, new challenges have emerged for several reasons. First, it requires prior knowledge of the expert to define the defect image and perform detection. To alleviate this problem, various companies have started utilizing AVI to reduce this dependence on domain knowledge. Secondly, defect detection is an arduous task since fewer defect images are available compared to normal images. This underlying problem leads to a classification model that is biased toward the majority class, which degrades the final performance. In this paper, we propose a method to synthesize defect images to solve the above-mentioned problems. Inspired by StyleGAN, we build mapping networks for latent space of the generator. Through this, we can synthesize defect images of various sizes in the manufacturing process. In addition, we experiment to find the most suitable loss function to solve the common problems of Generative Adversarial Networks (GAN). We also optimized the proposed method in terms of convergence and computation speed by estimating the size of optimal latent space. The experimental results using quantitative metrics illustrate the improved performance of the proposed methodology. As a result, it is now possible to solve the quality problem and increase productivity by reducing misclassification in the model through AVI experiments using the generated images**

Keywords—Automated visual inspection, generative adversarial networks, latent mapping, mapping network, synthesize defect

I. Introduction

Manufacturing process technologies are becoming increasingly fragmented and complex. The end product of the manufacturing process is directly related to economic factors as it affects productivity. Therefore, if defects in the product are not detected in advance, the cost of processing defective products occurs, affecting the entire manufacturing process [1]. Recently, there is a continuous rise in the demand for improvement in surface and shape quality of steel products [2]. Among them, detecting defects in advance to control defects in manufacturing is essential as it directly affects productivity and business competitiveness.

Defect refers to physical and chemical failures caused due to certain problems in the manufacturing process, facility, or manufacturing environment. Steel is manufactured through various processes such as rolling and forging. In this process, defects such as crazing, inclusion, pitted surface, rolled-in scale, and scratch occur as shown in Fig. 1 [3].

* Corresponding author--Tel: +82-2-3290-3396; Fax: +82-2-3290-4550

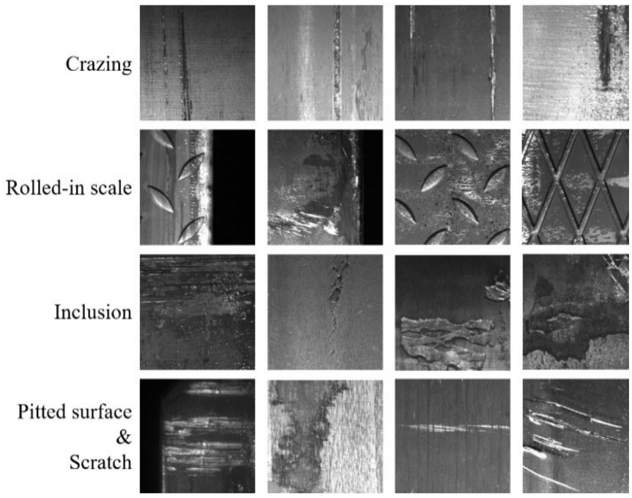

Fig. 1. Types of defects in the steel manufacturing process.

Defect detection for steel surfaces is an important step in ensuring the quality of industrial production. Steel surface defect detection undergoes 3 preliminary steps as shown in Fig. 2. First is the inspection step: Through this step, defects on the steel surface are detected by inspection tools [4]. Second is the review step: In this step, images of detected defects are captured by a specific tool. Third is the detection step: Detecting and classifying the types of defects according to the captured images. Steel surface defect detection processes allow the engineer to perform cause analysis and defect control. However, this visual inspection requires great reliance on the experience and the ability of individual engineers. Additionally, this process is usually done manually in the industry, making it unreliable and time-consuming. Therefore, automated visual inspection (AVI) targeting surface quality emerges as a standard configuration for steel manufacturing mills to improve product quality and promote production efficiency[5]. AVI, which performs classification through Convolutional Neural Networks (CNN), is not only widely applied to steel manufacturing process, but to glass, fiber, and semiconductor production process as well [6].

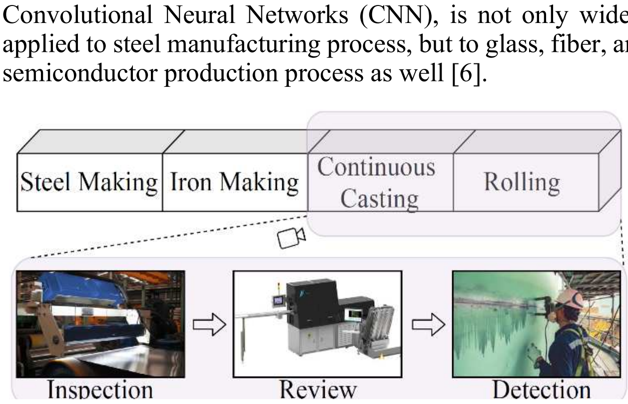

Fig. 2. Steel manufacturing process and defect detection steps.

Although the deep learning-based AVI model shows excellent classification performance for numerous defect types, it inheres two practical problems in steel manufacturing process. First, the frequency of defect data occurrence is extremely low that very little data exist to be used for deep learning model development [7]. In general, enough training data for both defect and normal class is required to improve the classification performance of a deep learning model [8]. However, in the actual industry, the number of defective data is minimal compared to that of normal data. Were if the AVI was conducted with collected data alone, the imbalance may lead to the lower learning rate of defect types and to degradation of performance. Therefore, it is necessary to balance normal and defective class. Class imbalance refers to the substantial proportional difference of each class in the total dataset. When the class distribution is unbalanced, the model is trained with a bias toward the majority class, classifying well for the class with a lot of data, but the opposite for the minority class. Furthermore, the imbalanced class distribution can also lead to serious type II errors. Therefore, preprocessing for class imbalance is essential in improving the overall classification performance in defect detection.

Second, the steel defect data has consisted of defects of various sizes. When using a generative model to solve the imbalance problem, large-sized defects can be easily generated with a simple generator. However, generation of small-sized defects is greatly influenced by which type of generative model is used [5]. Especially, a sophisticated generator is essential in situations where the defect size of the final product is about 0.2 mm, such as in the cold rolling process [9]. This study proposes a novel deep learning model for synthesizing defect data in steel manufacturing process. The proposed method generates a defective image similar to that of the real one, constructing effective training data for detecting detailed defective patterns.

In this study, we propose latent mapping adversarial networks to overcome two practical problems in steel manufacturing process. Our methodology is inspired by Style-based Generative Adversarial Networks (StyleGAN), which exhibits state-of-the-art in the data generation field [10]. The proposed method uses mapping networks in the latent space of the generator network. As the latent space goes through the mapping network, it becomes possible to learn the disentanglement of the training data distribution. This is the first step in the direction of explicit learning on real data. Mapping networks allow for a sophisticated generation of small-size defects. Our methodology also cares about the stability of learning. We use the Wasserstein distance as a distribution distance metric instead of the Jensen-Shannon (JS) divergence. The Wasserstein distance solves problems such as vanishing gradient and mode collapse witnessed in vanilla GAN [11]. The advantages of using the Wasserstein distance are discussed in section III.

To demonstrate the performance of the proposed method, images of flat steel plates are used in the production process. The data generation aspect of the proposed method was first evaluated by the quantitative evaluation metric, Fréchet Inception Distance (FID), and visual results. Also, the second evaluation is performed on the classification performance through a simple CNN structure [12]. Finally, to reduce the computational cost, we performed a task to find the optimal latent space and mapping network size for the data used in the experiment.

In summary, our contributions are as follows:

- Solve the data imbalance problem by generating steel defect data through the proposed method. Based on the quantitative evaluation metric, visual results, and classification results, we confirm the excellent results of the generation model.
- Set up the optimal potential space and mapping network to achieve the highest efficiency in the optimal time.

The rest of this paper is organized as follows. In Section II, we introduce previous works on AVI. Also we take a look at the background of our study and review some previous work. In Section III, we describe the proposed methodology. In Section IV, the performance of the steel surface defect dataset is evaluated using the proposed method. Finally, conclusions based on the experimental results and directions for future research in Section V.

II. RELATED WORK

In Section I, two problems needed to be solved by this study were shown. This section deals with the underlying class imbalance problem. Numerous solutions which have been proposed to solve class imbalance problems in AVI can be divided into two methods, a method of correcting the model itself and a method of directly processing the data [13]. In the former, similar to active learning or kernel-based methods, data instances of different classes are treated differently. As of the latter, the direct processing of data utilizes methods such as sampling or data generation to directly control the number of instances.

Sampling is a method to correct the bias between classes in data with an overwhelmingly small proportion of abnormal data compared to normal data. Representative methods to deal with class imbalance are oversampling and undersampling. Oversampling is a method of creating new data of the minority class in order to even the class ratio, and undersampling is a method of removing existent data of the majority class to match the ratio. As undersampling reduces the number of sample data from the majority class, it has the advantage of reducing model training time. However, it also has the chance to distort data features by removing crucial information. As for oversampling, the risk of data distortion is relatively small due that it creates new data while preserving the original data information. The oversampling methods mainly used for AVI include random oversampling, synthetic minority oversampling technique (SMOTE) [14], and adaptive synthetic sampling approach (ADASYN) [15]. Random oversampling increases the number of minority class data by randomly selecting and replicating a sample from the minority class. SMOTE synthesizes data by selecting random data belonging to a minority class and randomly selecting among the closest top k number of data. ADASYN is a method of adaptively synthesizing k number of data from marginal minority data according to the number of majority classes after calculating the ratio of data of a majority class. This oversampling method for image data has the problem of generating an image with a low resolution.

To simplify this problem, we used the technique of handling the raw image itself. This method is called data augmentation and is commonly being used as model regularization technique in recent studies [16]. Some common

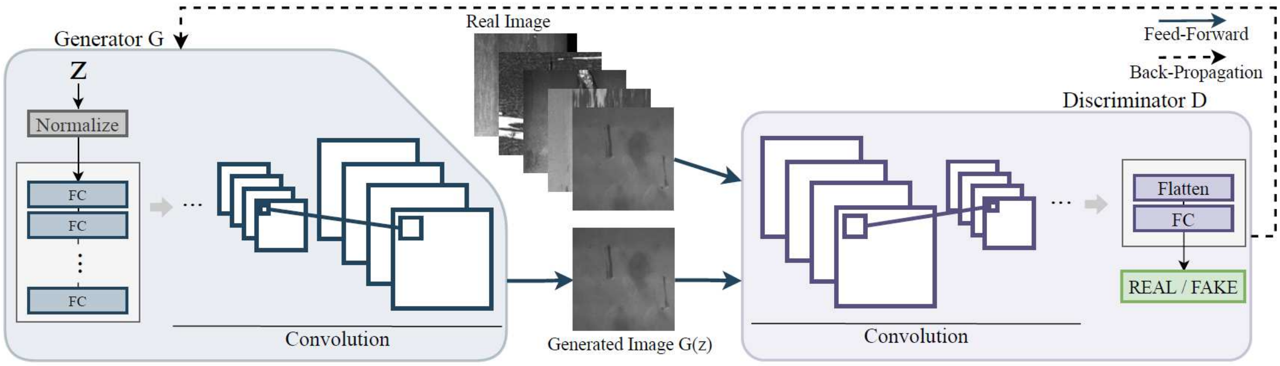

Fig. 3. Our latent mapping adversarial networks framework in manufacturing defect synthesis.

augmentation methods include flipping the image vertically or horizontally, shifting the image vertically or horizontally, and slightly rotating or zooming the image. This method helps the training model to be robust to small changes in the image. However, simple geometric transformations do not significantly change the characteristics of the image, making it impossible to identify more features in the image.

Among data generation methods, GAN is an algorithm that is of great interest [17]. GAN generates data based on distribution and mainly exhibits excellent performance in image generation. Some prior researched GAN are as follows: There was a ball-bearing failure detection method through Deep Convolutional GAN (DCGAN) [18]. In addition, a wafer defect image was adaptively generated using Conditional GAN (CGAN) [19]. Progressive Growing of GAN (PGGAN) increased the model training speed by gradually increasing the generator and discriminator and produced a good quality image [20]. In addition, to address the shortcomings such as vanishing gradient or mode collapse of GAN, Wasserstein GAN (WGAN) has been proposed [21]. This study compares and applies existing GAN-based generation models whose data generation performance has already been verified and finds an optimal generation model suitable for field data application. The focus of the proposed method generates effective learning data for detecting detailed defects.

III. Latent Mapping Adversarial Networks

This section describes the framework of the latent mapping adversarial networks, an approach to solving the imbalance problem for defect images. Fig. 3 is a schematic diagram of the overall structure of the proposed method. GAN is a neural network in which the generator and the discriminator learn adversarial to each other. The generator is trained to generate an image that is similar to the real image, while the discriminator is trained to discriminate between the real image and the generated image. The components of the proposed method are as follows: (1) Generator: Improved the quality of data generation by adopting mapping network structure for latent space. (2) Discriminator and Loss Function: By using Wasserstein distance with gradient penalty applied, addressed the imbalanced loss function problem occurring when the discriminator is backpropagated. Mapping network for latent space is discussed in Section III. A. Similarly, the imbalanced loss function is discussed in Section III. B.

A. Mapping Network for Latent Space

Defects occurring on the steel surface have a significant influence on the quality of the final steel product. Thus, it is crucial to correctly detect defects to ensure the quality of the final product and prevent the delivery of defective products to customers. However, as a result of imbalance in the steel surface defect data, the increase in misclassification of such data leads to deterioration of the classification performance. Therefore, there exists a need for an oversampling method that generates defect data.

In this study, the mapping network structure for the latent space was used to improve the quality of the generated data. The latent space of a well-trained GAN model has linear subspaces which permit direct variation adjustment[10]. However, direct control of latent space z is impossible since z of a vanilla GAN tends to form the training data into a single Gaussian distribution. The mapping network overcomes this problem by impeding the latent space z from entering the generator as an input value. Instead, we input w passed through the mapping network as input value to the generator. While latent space z cannot accurately match the feature distribution of the training data, w can because it undergoes a nonlinear transformation through the mapping network. Therefore, the disentanglement characteristic of w, suited for the train data, leads to improved data generation. In summary, in the structure of the vanilla GAN model generator, the latent space is passed through the mapping network composed of fully connected layers.

This approach may seem very simplistic. To learn the distribution of data, we used GAN. When we generate a noise vector and put it as an input to the GAN, we can generate random images similar to our training data but not present in the training data. However, it is not easy to create a random image with the desired characteristics. The reason we get this result is that z is all related to any other feature. One of the reasons why an axis is entangled usually happens when the degree of z is not sufficient. The mapping network makes the axes disentangle by making the z degree sufficient. Therefore, it achieves a performance improvement in terms of generative for the training data.

B. Imbalanced Loss Function

Existing oversampling methods do not use data distribution. Additionally, the GAN problems, vanishing gradient and mode collapse, have a detrimental effect on the quality of the generated data [21]. Vanishing gradient refers to a problem that occurs when the discriminator learns to perfection, as in (1). If the discriminator D is perfect, the loss function of GAN will approach zero and the gradient will not be obtained in the learning process.

$$D(x) = \mathbf{1}^{\forall} x \in p_r,\ D(x) = \mathbf{0}^{\forall} x \in p_g \tag{1}$$

In Equation (1), p_r is denotes the distribution of the real data, and p_g denotes the distribution of the generated data.

Mode collapse, another characteristic problem of GAN, is when the generator always outputs the same result during the learning process due to GAN using JS divergence as a distance metric. In this study, 1-Wasserstein is used as the distance metric instead of the JS divergence to deviate from the problems of gradient loss and mode collapse. However, the 1-Wasserstein distance has a problem where the weight is clipped. Also, gradient penalty (GP) technique is used to solve the weight clipping problem of 1-Wasserstein [11]. Therefore, the imbalance loss function, WGAN-GP, is expressed as (2), and it is learned in the direction of minimizing this constraint.

$$L = \mathbb{E}_{x\sim \boldsymbol{p_r}}[D(x)] - \mathbb{E}_{z\sim \boldsymbol{p_g}}[D(z)] + \lambda \mathbb{E}_{\hat{x}\sim \mathbb{P}_{\hat{x}}}[(\|\nabla_{\hat{x}} D(\hat{x})\|_2 - 1)^2] \tag{2}$$
$$where \quad \hat{x} = tx + (1-t)z \quad with \ 0 \le t \le 1$$

In Equation (2), x denotes actual data and z denotes data generated in the latent space. The remainder of (2) denotes the part for the gradient ∇ of the discriminator D with $\hat{x}$ uniformly sampled between x and z at the ratio of t. When the L2 Regularization (L2 Norm) of this gradient has a value other than 1, it is optimized by giving a penalty as much as λ. Consequently, by manipulating the loss function to have a meaningful value when the two distributions do not overlap in a low-dimensional manifold can solve the loss of slope and mode collapse problems.

IV. Experimental Results and Discussion

All experiments are performed using the Pytorch software package and Scikits Learn(Sklearn), Panda library, together with Python3 language, running on a Desktop with Intel(R) Core(TM) i7-9700K CPU 3.60GHz, 32GB RAM with NVIDIA GeForce RTX 3080 10GB. For comparative purposes, we also implement some of the other leading GANs using Pytorch.

A. Datasets

The data used for performance verification in this study is acquired from Severstal: steel manufacturing process. This data is collected by a high-frequency camera capturing images of flat sheet steel during the production process. This dataset is usually subjected to defect location and type prediction found in steel manufacturing. The dataset contains a single class of defect type data, multiple class of defect type data, and non-defect type data. Fig. 4 shows an example of the data used in the experiment. Steel defect data is a schematic diagram of each class of defect data consisting of tiny defects to large defects. In this study, image data of 256×1600 was cropped into a square image of size 256×256, tailored to be utilized as an input value.

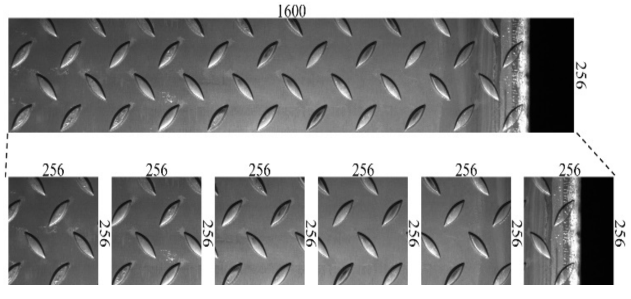

Fig. 4. Example of the cropping steel defect images (256×1600 to 256×256).

B. Experimental Design

The preprocessed dataset was partitioned at a ratio of $Data_{train}: Data_{test} = 7:3$. The experiment consists of two main steps. The role of the first stage experiment is to verify the generator model of the proposed method. The superior performance of the proposed method is demonstrated in comparison with other GAN-based generator models. Each GAN layer was uniformly composed of 5 layers, and 100 dimensions were used for the latent space. For an optimization function, RMSProp, which is frequently used in the GAN model, was used.

The second stage experiment finds the optimal latent space size and mapping network. By structuring part of the proposed method with mapping network, the proposed method was able to acquire disentanglement features. Through the proposed model, the optimal size of the initial latent space and the mapping network were experimentally discovered. All experiments were evaluated by the quantitative evaluation metric FID. During data division, the seed was changed and the average value of the ten performed results was used as the final metric.

C. Performance Measurement Metric

Manufacturing data is primarily comprised of normal data. However, in many cases, abnormal data is more critical in defect control than normal data. This imbalance becomes a problem as it leads to an increase in the misclassification error rate of abnormal data, consequently degrading overall classification performance. In this study, the oversampling method, a method of randomly generating abnormal data using the GAN model, is used to solve the imbalance of abnormal data.

Early GAN was accompanied by problems such as instability of learning and mode collapse, exhibiting difficulties in performance evaluation [17]. To address such problems, the development of various GAN models on top of inception scores (IS) and FID using the inception model, have made evaluating the performance of GAN possible [12]. The inception model, widely used for transfer learning and fine-tuning, is a CNN model that pre-trained ImageNet data. ImageNet consists of 1,000 class and 1.2 million images. When an image is inputted into the model, the inception model outputs probability vectors belonging to each 1,000 class. Using the generated image as an input value to the inception model, it can calculate the IS as shown in (3).

$$Inception\ Score = \exp\left(\mathbb{E}_{z\sim p_g} KL(p(y|z)||p(y))\right) \tag{3}$$

In Equation (3), $p(y|z)$ is the conditional class distribution and $p(y)$is the marginal class distribution. The inception score can have a value between 1 or more and 1,000 or less but is usually around 2. However, the inception score encompasses the disadvantage of not using the real data distribution. In this study, the shortcomings of the IS are overcome by FID which is a measure of the difference between the two normal distributions, as shown below.

$$FID = \|m - m_w\|_2^2 + Tr(C + C_w - 2(CC_w)^{1/2}) \tag{4}$$

Smaller FID means better quality, and (m, C) and (m_w, C_w) denotes the mean and covariance of the distribution between the generated image and the real image. As it is widely accepted that FID captures the quality of generated data better than IS, this study adopts FID as a measure to assess the quality of the generated image.

D. Experimental Results

1) Performance Compared to Generative Model: The proposed method differentiates by using mapping network structure and imbalanced loss function (WGAN-GP) to improve the quality of the data. The latent space used in previous GAN models displayed difficulties in avoiding entanglement due to its tendency to follow the probability density of the training data. However, we use a mapping network to solve this problem and exhibit the disentanglement of latent space. Thus, making direct adjustments to changes possible.

Table I shows the results of comparing the proposed method with vanilla GAN, DCGAN, and DCGAN+WGAN-GP. Baseline is vanilla GAN, and DCGAN which deep convolutional structure is added to the baseline. DCGAN+WGAN-GP is the loss function of DCGAN changed to WGAN-GP. The proposed method adds mapping network composed of 8 fully connected layers to the previous methods. The reason for constructing the control group as follows is to check the effect of each method briefly. This also shows the gradual evolution of the GAN-base model.

TABLE I. COMPARISON AVERAGE FID OF GENERATIVE MODELS

Method	FID
Baseline (GAN)	105.17
DCGAN	31.47
DCGAN+WGAN-GP	26.64
Proposed Method	**15.81**

By Table I, it was confirmed that the proposed method showed excellent performance in terms of average FID. As each method was sequentially added, it was possible to confirm that the FID improved sequentially as well. Fig. 5 is a visual result of the comparison of the real image with the generation results of each method.

As a result of comparing the creation results of each method as a 5x5 matrix, it is difficult to recognize a large difference when visually confirmed. As shown in Table 2, we applied the generated data to the classification task. To do this task, we used a simple fully convolutional network (FCN) [22]. We train FCN algorithm on the generated samples and test the accuracy on the real image. By the classification accuracy, the proposed method generates similarly to the real image in terms of creation.

TABLE II. CLASSIFICATION ACCURACY USING FCN FOR THE RESULTS OF GENERATIVE MODELS.

Method	Classification accuracy
Baseline (GAN)	74
DCGAN	83
DCGAN+WGAN-GP	89
Proposed Method	92

2) Optimal Latent Space and Mapping Network Size: The proposed method improved the quality of image generation by adopting the mapping network structure. By doing so, optimization of latent space, where random vectors that

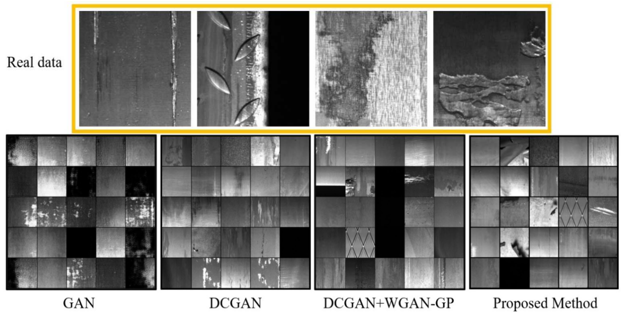

Fig. 5. Visual comparison of the results of each method.

generate images similar to that of real images, was possible. Generally, a sufficiently large latent space can adequately express the characteristics of the real data, leading to the use of 100-dimensional size for general latent space. However, adopting image generation for the steel manufacturing process requires accurate and expeditious processing. Thus, there is a need for image generation that performs well even with a simple structure. As the size of a latent space directly affects the number of parameters, the convergence speed, and computation time, finding the optimal size of the latent space is a necessary task.

In this experiment, we strive to find the optimal latent space size as well as the mapping network. Table 2 shows the results of the experiment where adjustments of mapping network to 0, 2, 4, and 8, with corresponding adjustments to the dimension size of the latent space to 1, 2, 3, 10, 50, and 100 were made.

In Table 3, 'traditional' exhibits the result of using latent space of the general GAN without mapping network, and 'style-based' indicates the number of mapping networks used. The evaluation is done through the FID, where it is known that the lower the FID, the more similar generated data is to the real data. Up to the 10th dimension, the performance shows a tendency to improve, while thereafter, performance varies with only the slightest difference. Therefore, it can be confirmed that there is not a significant performance difference between conventionally used 100 dimensions and dimensions after the 10 dimensions. Also, the mapping network shows the best performance when it is composed of 8 fully connected layers. Latent space and mapping network size are closely related to computation time. Thus, to accommodate for the need for accurate and expeditious processing characteristics of the steel manufacturing process, the proposed method has consisted of 50 latent spaces and 8 mapping networks. As a result, it was confirmed that the proposed method generates high-quality images.

V. CONCLUSION

This study proposed a method to tackle the imbalance that exists in defect detection in the steel manufacturing process. It improved the quality of the generated image by adopting mapping network. At the same time experimented to find the optimal latent space size and mapping network to achieve accurate and expeditious processing in the process. The quality of the generated images was evaluated using quantitative metric FID and visual results, and classification performance.

The method proposed in this study is applicable to AVI problems in the various manufacturing process, especially

TABLE III. COMPARISON AVERAGE FID OF OPTIMAL LATENT SPACE AND MAPPING NETWORK SIZE

Mapping network	Latent space	FID	Mapping network	Latent space	FID
Traditional	1	198.85	Style-based 4	1	139.14
	2	165.11		2	49.52
	3	69.18		3	29.48
	10	43.57		10	7.97
	50	29.21		50	7.22
	100	17.42		100	7.21
Style-based 2	1	161.59	Style-based 8	1	136.32
	2	62.87		2	49.19
	3	32.24		3	25.35
	10	11.87		10	7.46
	50	9.65		50	**6.94**
	100	7.86		100	**7.16**

processes with innate imbalance problems and the practicality of the proposed method also makes it highly applicable to various fields other than AVI. For future research, by de-riving image quality evaluation indicators suitable for manufacturing data we will seek to improve the overall quality of the proposed method.

ACKNOWLEDGMENT

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (NRF-2019R1A2C2005949, NRF-2021R1A6A3A13045200). Also, this work was supported by Samsung Electronics Co., Ltd (IO201210-07929-01).

REFERENCES

[1] B. Chen, J. Wan, L. Shu, P. Li, M. Mukherjee, and B. Yin, "Smart fac tory of industry 4.0: Key technologies, application case, and challenge s," *IEEE Access, 6*, 6505-6519, 2017.

[2] F. Akhyar, C. Y. Lin, K. Muchtar, T. Y. Wu, and H. F. Ng, "High effi cient single-stage steel surface defect detection," *IEEE International C onference on Advanced Video and Signal Based Surveillance (AVSS),* pp. 1-4, 2019.

[3] N. Neogi, D. K. Mohanta, and P. K. Dutta, "Review of vision-based st eel surface inspection systems," *EURASIP Journal on Image and Vide o Processing*, pp. 1-19, 2014.

[4] J. Wang, Z. Yang, J. Zhang, Q. Zhang, and W. T. K. Chien, "AdaBalG AN: An improved generative adversarial network with imbalanced lea rning for wafer defective pattern recognition," *IEEE Transactions on Semiconductor Manufacturing*, *32*(3), 310-319, 2019.

[5] Q. Luo, X. Fang, L. Liu, C. Yang, and Y. Sun, "Automated visual def ect detection for flat steel surface: A survey," *IEEE Transactions on I nstrumentation and Measurement*, *69*(3), 626-644, 2020.

[6] S. Cheon, H. Lee, C. O. Kim, and S. H. Lee, "Convolutional neural ne twork for wafer surface defect classification and the detection of unkn own defect class," *IEEE Transactions on Semiconductor Manufacturi ng, 32*(2), 163-170, 2019.

[7] Z. J. Xu, Z. Zheng, and X. Q. Gao, "Operation optimization of the stee l manufacturing process: A brief review," *International Journal of Mi nerals, Metallurgy and Materials*, 2021.

[8] E. Zhang, B. Li, P. Li, and Y. Chen, "A deep learning based printing d efect classification method with imbalanced samples," *Symmetry, 11*(1 2), 1440, 2019.

[9] D. Kang, Y. J. Jang, and S. Won, "Development of an inspection syste m for planar steel surface using multispectral photometric stereo," *Opt ical Engineering, 52*(3), 039701, 2013.

[10] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," *In Proceedings of the IEEE/CV F Conference on Computer Vision and Pattern Recognition*, pp. 4401-4410, 2019.

[11] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville, "Improved training of wasserstein gans," *arXiv preprint arXiv:1704.0 0028*, 2017.

[12] A. Brock, J. Donahue, and K. Simonyan, "Large scale GAN training f or high fidelity natural image synthesis," *arXiv preprint arXiv:1809.1 1096*, 2018.

[13] N. Kondo, M. Harada, and Y. Takagi, "Efficient training for automatic defect classification by image augmentation," *IEEE Winter Conferen ce on Applications of Computer Vision (WACV)*, pp. 226-233, 2018.

[14] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SM OTE: synthetic minority over-sampling technique," *Journal of artifici al intelligence research, 16*, 321-357, 2002.

[15] H. He, Y. Bai, E. A. Garcia, and S. Li, "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," *IEEE international join t conference on neural networks*, pp. 1322-1328, 2008.

[16] E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, and Q. V. Le, "Autoa ugment: Learning augmentation strategies from data," *In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recogn ition*, pp. 113-123, 2019.

[17] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in neural information processing systems*, 27, pp. 2672-268 0, 2014.

[18] J. Viola, Y. Chen, and J. Wang, "FaultFace: Deep convolutional gener ative adversarial network (DCGAN) based ball-bearing failure detecti on method," *Information Sciences, 542*, pp. 195-211, 2021.

[19] M. Mirza, and S. Osindero, "Conditional generative adversarial nets, " *arXiv preprint arXiv:1411.1784*, 2014.

[20] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," *arXiv preprint arX iv:1710.10196*, 2017.

[21] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adv ersarial networks," In *International conference on machine learning*, p p. 214-223, 2017.

[22] K. E. Smith, and A. O. Smith, "Conditional GAN for timeseries gener ation," *arXiv preprint arXiv:2006.16477*, 2020.