A Texture Generation Approach for Detection of Novel Surface Defects

Yu-Ting Kevin Lai
Dept. of Electrical and Computer Engineering
National Chiao Tung University
Hsinchu, Taiwan
kevin83825@gmail.com

Jwu-Sheng Hu
Dept. of Electrical and Computer Engineering
National Chiao Tung University
Hsinchu, Taiwan
jshu.nctu@gmail.com

***Abstract*—Surface defect detection is challenging due to varying defect types and defect novelties. Because of this, it is hard for algorithms to implement across datasets. Moreover, current automated optical inspection (AOI) machines cannot handle this novelty effectively. In this work, we develop a new method for surface defect detection based on generative models, which can detect novelty according to learned distributions. Experimental results on real industrial datasets show that the proposed method can successfully construct the surface texture pattern generator. By transforming the image through the generator to the corresponding latent space, the defects can be separated effectively without a tedious effort of annotation in a large amount of training data.**

***Keywords*—automated optical inspection, surface defect detection, generative adversarial networks**

I. Introduction

Automated optical inspection (AOI) is critical in manufacturing processes in order to inspect defects and increase yield rate. In general, AOI machines usually deal with two-dimensional images rather than three-dimensional depth ones. Thus, surface defect detection [1], [2] becomes a major task in AOI applications. However, in real inspection processes, the defects vary among different types of surfaces, it is then hard to design a general approach to be suitable for all datasets collected. Moreover, the defects themselves are sometimes formed during manufacturing processes, in which unexpected events such as vibration or temperature change may induce new defects. This situation makes traditional image processing methods impossible to solve as defects must be modeled in advance. On the other hand, modeless methods such as the neural network [3], [4] also need to re-train the classifiers, which is time-consuming. In the absence of a standard definition of defect types, the problem becomes even more complicated.

Simply speaking, the problem described above is to automatically detect surface defects without training or modeling of the defect images in advance. The difficulty level depends on the texture of the defect-free surface. For surfaces without any pattern or with periodic patterns, to detect new defects is relatively simple since one can easily model the background. However, in manufacturing, most of the surfaces of work pieces or products contain patterns that exhibit no periodicity and vary among them. It is then difficult to build the texture model analytically. Secondly, in some cases such as solar panel, certain types of defects have close visual appearances to the normal patterns. Thus, finding appropriate models or feature descriptions to precisely distinguish these defects is a very difficult task. In this paper, we propose a novel method to solve the problem using the technique of generative adversarial networks (GANs). The method is to construct a generator that can generate the correct mapping of positive images. Next, we can distinguish the negative images which do not conform to that mapping. It is shown that the proposed method is robust to distinguish defective images from normal ones without prior knowledge of the defects. That is to say, it is not necessary to annotate the labels of defective samples like the traditional training procedure of neural networks.

The proposed method belongs to one-class classification problem. Our main contributions can be summarized as follows. (1) We proposed a novel workflow for surface novelty defect inspection under complex background texture. (2) Our method can effectively distinguish defects which are not known in advance. (3) We show that our method does not require large number of training data and defect annotation is not needed. (4) Our method is effectively adaptable to various datasets.

II. Related Works

A. Surface Defect Detection

The computer vision methods of surface defect detection depend on their objectives and are categorized into several approaches. Ngan et al. [5] and Xie [6] focused on fabric and texture analysis, while Neogi et al. [1] discussed several methods related to steel surface inspection which includes both supervised and unsupervised algorithms. On the other hand, Bouchot et al. [7] used template matching for regularly textured surfaces, which reduces the tuning of parameters. However, this method requires exact match of textural information in the background. For real-time inspection, Hu [8] implemented ring Gabor Filter on surface defect detection. Most of the methods require annotation of the defects by human first for subsequent analysis. In addition, these algorithms developed are usually

2577-1655/18/$31.00 ©2018 IEEE
DOI 10.1109/SMC.2018.00736

application dependent and need significant efforts of parameter tuning for optimal performance.

B. *Generative Adversarial Networks*

GAN was first proposed by Goodfellow et al. [9], which optimizes the generator through a minimax game. The optimization procedure and purposes had also been improved by several works [10]-[14]. Arjovsky et al. [10] and Gulrajani et al. [11], stabilized the training of GAN by improving the loss function. Furthermore, Mirza et al. [12] and Chen et al. [13] modified the structure and applied additional information to generate images conditionally.

For the operations of latent representation, Radford et al. [14] proposed Deep Convolutional Generative Adversarial Network (DCGAN), which only used convolutional layers in the architecture. They showed that arithmetic operations in the latent space is feasible in producing customized images. In other words, the latent space can be viewed as a transformed space of the target image data. It then follows that the latent space might be a good candidate of the detection if an image belongs to the target group. Based on this idea, we use DCGAN as our GAN architecture.

C. *Reverse Generative Models*

In representation learning, many works passed data through an encoder [15] to get latent representations. However, some works [16], [17] attempted to learn the latent space by reversing the training procedure of the generator in GAN. Due to the differentiability of the generator, Bora et al. [18] applied gradient descent algorithm with respect to the latent representations such that the corresponding generated images have small measurement errors. Bruna et al. [19] further investigated on the theoretical conditions for a generator to be reversible. The reversed training to obtain the latent space for representation is adopted in this paper.

III. GAN-based Anomaly Detection

In this section, we describe the proposed workflow for surface defect inspection. The workflow is divided into four components. First, the generative mapping from the latent space to the image space is computed followed by a qualifier to access the degree of success of the generator. In testing stage, the optimized latent representations are computed by back-propagation from the image space. We further measure the Fréchet distances of these latent representations, which serve as the classification mechanism. The workflow requires no parameter tuning. Its detail is shown in Fig.1.

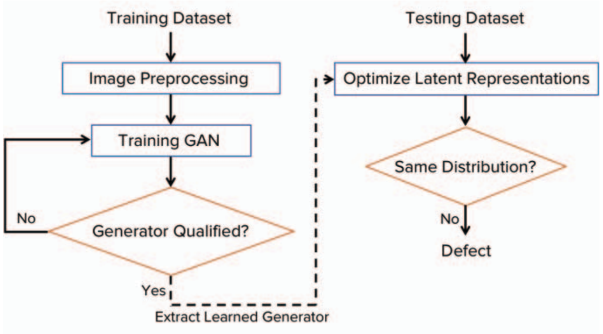


Figure 1. The workflow of our proposed method.

A. *Generative Mapping*

GANs are well-known for their ability to capture the distribution of the given training dataset [20]. Rather than obtaining plausible results of generated images, we focus on the learned mapping of GANs from the latent space to the image space.

The common structure of GAN consists of two neural network modules, one is generator *G*, and the other is discriminator *D*. The goal of the generator is to generate plausible images to fool discriminator, while the discriminator is to make judges on whether these images are from real images or results of the generator. Geometrically, the generator serves as a decoder, which projects the latent codes in the latent space to the image space as illustrated in Fig.2. The latent codes are randomly sampled from Gaussian distribution in a unit ball. Conversely, the discriminator encodes the incoming images from real images and generated images and then classify them according to the correct labels.

The formulation of GAN is as following,

$$\min_{G}\max_{D} V(G,D) \\ = E_{x\sim p_{data}(x)}[\log D(x)] + E_{z\sim p_{z}(z)}\left[\log\left(1\text{-}D\left(G(z)\right)\right)\right], \qquad (1)$$

where x and z represent images and latent codes respectively. For the discriminator, the objective function maximizes the

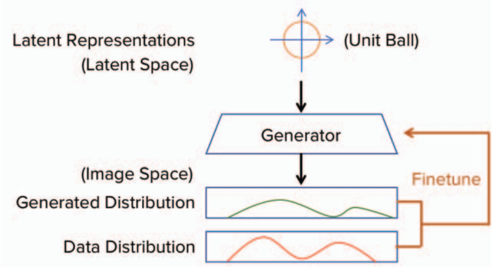


Figure 2. The illustration of using generative adversarial network in the training stage.

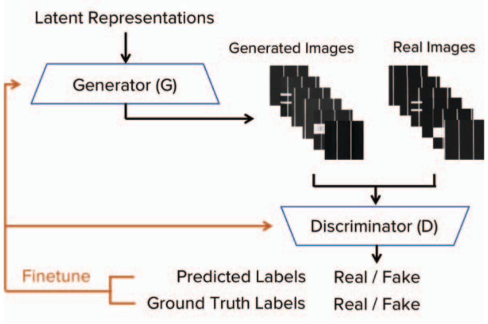


Figure 3. GAN structure we used in our proposed method.

probability that x comes from the real data distribution, and the probability that $G(z)$, i.e. generated images, comes from generated data distribution. On other hand, the objective function for the generator minimize the probability that $G(z)$ comes from the generated data distribution, i.e. it tries to make the generated images as real as possible. The architecture of GAN is shown in Fig.3. For surface defect detection, we train a GAN only using positive data, i.e., to learn the positive mapping from the latent space to the image space.

B. Evaluation and Visualization

To acquire an optimal generator, two measurements are applied in our experiments. Intuitively, we directly evaluate mean squared error (MSE) and similarity structural analysis (SSIM) [21] on our generated images because in industrial datasets, the images are often structured. MSE and Peak Signal-to-Noise Ratio (PSNR) are common metrics for the evaluation of image quality. However, PSNR is only suitable for the evaluation of restoration of images. Given two images $x(m,n)$ and $y(m,n)$ with the dimension of $M \times N$, we can compute its MSE as

$$\mathrm{e_{MSE}} = \frac{1}{\mathrm{MN}}\sum_{\mathrm{m=1}}^{\mathrm{M}}\sum_{\mathrm{n=1}}^{\mathrm{N}}\|\mathrm{x(m,n)\text{-}y(m,n)}\|^2, \tag{2}$$

SSIM further enhance the measurement of MSE, which takes luminance, contrast, and structure information into consideration. This measurement is suitable for evaluating similarities between two natural images, to which industrial images often belong. Given two images x and y, we compute SSIM according to their mean μ_x and μ_y, the variance σ_x and σ_y, and small numbers c_1 and c_2 as shown in (3).

$$\mathrm{SSIM} = \frac{(2\mu_x\mu_y+c_1)(2\sigma_{xy}+c_2)}{(\mu_x^2+\mu_y^2+c_1)(\sigma_x^2+\sigma_y^2+c_2)}. \tag{3}$$

Alternatively, t-distributed Stochastic Neighbor Embedding (t-SNE) [22] is well suited for visualization of high-dimensional datasets. We use t-SNE to visualize the similarities between generated images and real images. The objective function minimizes Kullback-Liebler (KL) divergence of high-dimensional map p_{ij} and low-dimensional map q_{ij} . The pairwise similarities of the maps p_{ij} and q_{ij} are given by (4) and (5).

$$\mathrm{p_{ij}} = \frac{\exp(-\|x_i\text{-}x_j\|^2/2\sigma^2)}{\sum_{k\neq l}\exp(-\|x_k\text{-}x_l\|^2/2\sigma^2)}, \tag{4}$$

$$\mathrm{q_{ij}} = \frac{\exp(-\|y_i\text{-}y_j\|^2)}{\sum_{k\neq l}\exp(-\|y_k\text{-}y_l\|^2)}. \tag{5}$$

where in (4), x and σ represent high-dimension data points and the variance, while in (5), y represents low-dimensional data points. Hence, if the generated images and real images are close to each other in the corresponding low-dimensional map, t-SNE can further provide a verification of how well the generator is constructed.

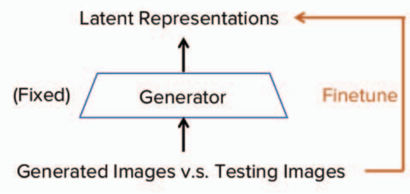


Figure 4. Illustration of back-propagating to the latent space.

C. Back-propagation to the Latent Space

Given the learned generator, the latent representations of the testing data are computed through the optimization procedure. The testing data contains both defect-free images and defective images. However, the subtle differences between these two categories often confuse current AOI machines such that it cannot identify them correctly. Back-propagating to the latent space enables us to discriminate the differences effectively. In this procedure, we fix the parameters of the generator and directly perform gradient descent to the latent representations.

D. Statistics of the Latent Representations

The effectiveness of using the latent representations to distinguish between defect-free and defective images depends on the distance of the corresponding latent sub-spaces. The sub-space is usually characterized by statistics for the data-driven approach. In our proposed approach, we calculate the statistics according to Fréchet distance [23], which was also used in [24] to calculate the statistical differences between features. Fréchet distance not only computes the distances but also finds the minimal course of the curves, which in our case, is the set of latent representations. Given two sets of mean and variance (μ_1, c_1) and (μ_2, c_2), we can calculate their Fréchet distances in (6).

$$\mathrm{FD} = \|\mu_1\text{-}\mu_2\|^2 + \mathrm{trace}(c_1 + c_2\text{-}2 \times (c_1 \times c_2)^{\frac{1}{2}}). \tag{6}$$

IV. Experimental Results

The proposed method is applied to defect inspection of two products in manufacturing: solar panel and wood panel, which is illustrated in Table I. For solar panel dataset, there are 200k defect-free images and each type of defects has 30k images. For wood texture dataset, each label only has 2k images. We randomly sample 90% of the defect-free images for training. The GAN computation is performed on a single Nvidia GeForce GTX 1060 6GB GPU.

A. Evaluation of Generative Models

For the evaluation of GAN, we randomly select from generated images and real images to form pairs and calculate their MSE error and SSIM value respectively. Fig.5(a) shows a successful case that the generator can generate images that are close to real ones while in Fig.5(b), the generator is not optimal

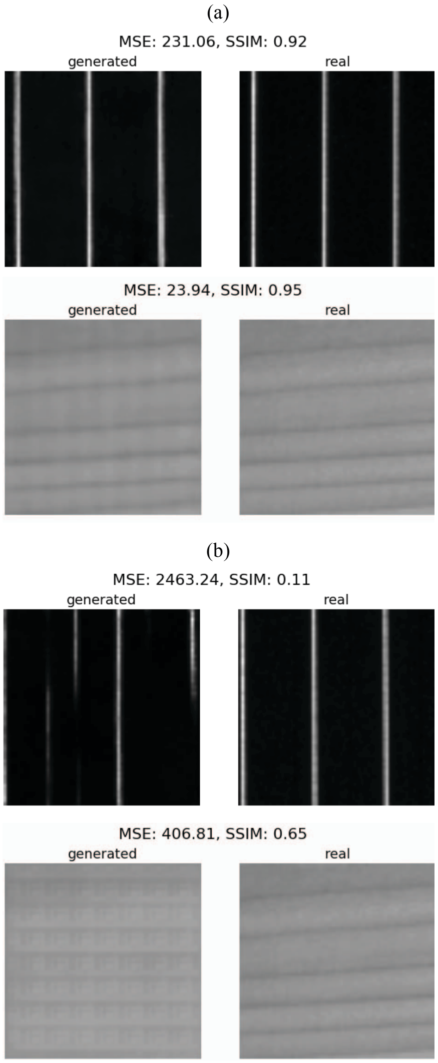


Figure 5. MSE error and SSIM value between pairs of generated images and real images of solar panel dataset and wood texture dataset. The desired results are shown in 5(a), while in 5(b), the generator is not optimal.

and the generated image is visually different to the real one. The MSE and SSIM values shown for both cases indicates that these two indices are consistent to the performance of GAN results.

The t-SNE visualization is computed directly on the image space of solar panel and wood texture datasets as shown in Fig.6(a) and Fig.6(b) respectively. For each result, red cross dots represent the generated images, whereas blue circle dots represent real images. The results show that the generated images are within the same manifold of real images, which again verify that our generative model has learned the correct mapping from the latent space to the image space.

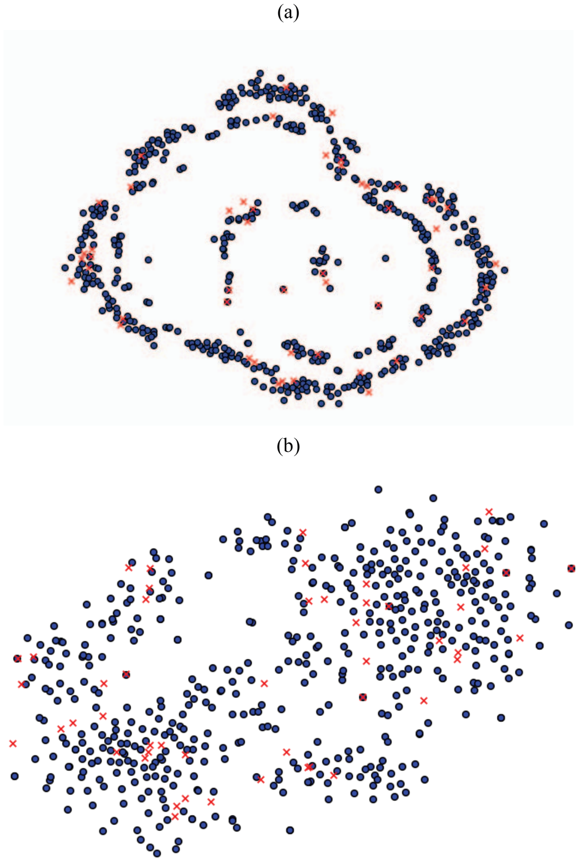


Figure 6. t-SNE visualization of generated images and real images of solar panels in (a), and wood textures in (b).

B. Defect Separation in the Latent Space

Table I shows the defective images and normal images in two different datasets according to ground-truth annotations. Graphically, solar panel dataset is highly structured in the background, while wood texture dataset is rather random in the background. Noted that in solar panel dataset, the defects are

TABLE I. DEFECTIVE DATA AND NORMAL DATA OF EACH DATASET

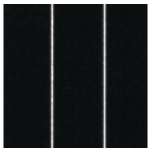
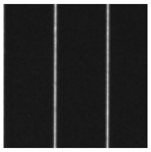
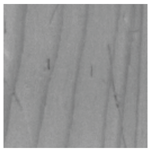
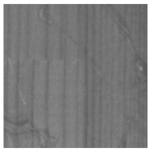
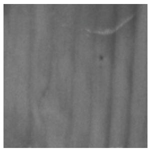
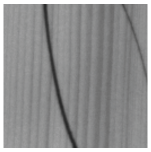
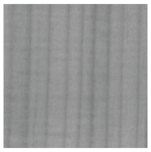
Solar Panel			**Wood Texture**				
Defective		***Normal***	***Defective***				***Normal***
							
Type I defect	Type II defect	None	Crack	Hot glue	Lubricant	Marker	None

TABLE II. FRÉCHET DISTANCES OF DEFECTS IN TESTING DATASETS

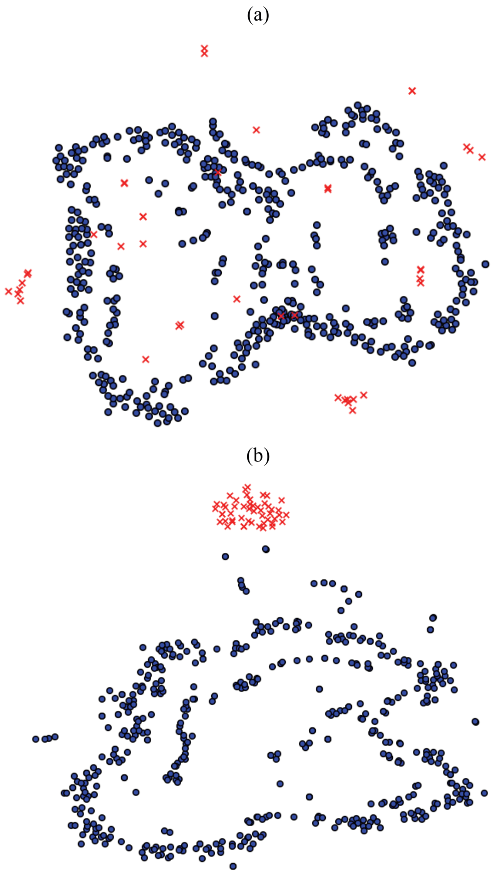
Dataset	Types	Fréchet Distance of Latent Representations
Solar Panel	**Normal**	**0.011**
	Type I defect	0.315
	Type II defect	0.402
Wood Texture	**Normal**	**0.079**
	Crack defect	0.714
	Marker defect	0.567
	Lubricant stain defect	0.626
	Hot glue defect	0.685

TABLE III. FRÉCHET DISTANCES OF DIFFERENT TESTING DATASETS

Dataset	Defect Type	Defect Type	Fréchet Distance of Latent Representations
Solar Panel	Type I	Type II	0.214
Wood Texture	Crack	Marker	0.408
	Crack	Lubricant	0.374
	Crack	Hot glue	0.479
	Marker	Lubricant	0.201
	Marker	Hot glue	0.133
	Lubricant	Hot glue	0.238

nearly undiscernible for human vision, and it also cannot be detected by current AOI machines.

We show that our proposed method can effectively detect the differences after optimizing the latent representations of these datasets. Table II shows the corresponding Fréchet distances given the types of testing datasets. Besides calculating on optimized defective latent representations, we also calculate on optimized normal ones to validate our proposed method. In this table, we can easily see that the Fréchet distances of the normal latent representations are an order of magnitude smaller than the defective ones. In some cases, especially in solar panels, the differences are almost undetectable in the image space. Thus, our method is powerful in separating these defects. The defect separation phenomenon is illustrated in Fig.7. We again compare the differences with t-SNE visualization and found that we successfully separate the originally inseparable images in the optimized latent space. This also suggests that our method can be applied to different datasets, which is not dataset-specific.


(a)

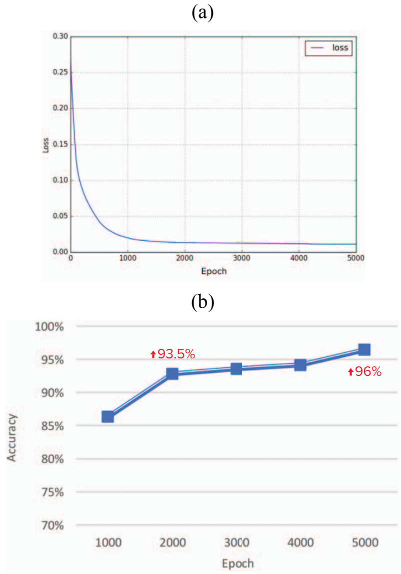
(b)

Figure 7. Phenomenon of defect enlargement. 7(a): t-SNE on the image space. 7(b): t-SNE on the optimized latent space.

Furthermore, it is also possible to separate different defects from each other according to Fréchet distances between them as shown in Table III. Each pair of defects exhibits different distributions. Consequently, if a new defect appears, we can detect it by the novel distribution of that new defect, which can be seen as the classification labels.

C. Accuracies & Elapsed Time

In real applications, the computation time should be kept as small as possible. Hence, to find the best accuracy, we measure the losses and determine how many iterations are needed in back-propagating the latent vectors. Fig.8(a) shows that the value of the loss function converges after around 2000 iterations, where in Fig.8(b), the accuracy reaches above 93.5%.


(a)

(b)

Figure 8. The loss plot 8(a) and the accuracies 8(b) in optimizing the latent representations.

Therefore, we can reduce the number of iteration and speed up inspection procedure in real applications while maintaining good accuracy. Table IV further shows the average runtime per pixel of the proposed method for each dataset. The number of iterations is set to 5000. From the results, the average runtime per pixel for different datasets do not exceed 0.25 milliseconds.

Finally, the proposed method is compared with a CNN-based classifier. We used VGG-19 [25] network as our CNN structure, which obtained the best accuracy among other networks on our datasets and we ran for 50 epochs. However, Table IV shows that our method achieves a better accuracy and requires less time on training. Moreover, CNN-based classifier needs extra annotations on different defects in advance, while our method does not need this prior knowledge to inspect defects.

TABLE IV. Average Runtime of GAN-based Method

Dataset	Average Runtime (ms)
Solar Panel	0.178
Wood Texture	0.211

TABLE V. Comparisons with CNN-based Model on solar Penal

Method	Overall Accuracy (%)	Training Time (s)
VGG-19 network	89.48%	18000
Our method	**93.75%**	**3262**

V. Conclusion

We proposed a novel method for industrial surface novelty defect detection. The core of the method is that we use a generative network and a representation learning mechanism according to statistics to detect novelties. From experiments in two representative industrial datasets, our method is able to distinguish defective images from normal ones in the optimized latent space. It is difficult to detect these defects directly using traditional methods. The proposed method also achieves a better accuracy compared to CNN-based method, and the time required on training is almost 6 times less than a CNN-based classifier.

Acknowledgment

This work is sponsored in part by the Ministry of Science and Technology of Taiwan under the grant number 105-2221-E-009-038-MY3 and Industrial Technology Research Institute (ITRI) of Taiwan.

References

[1] N. Neogi, D. K. Mohanta, and P. K. Dutta, "Review of vision-based steel surface inspection systems," in *EURASIP Journal on Image and Video Processing*, vol. 2014, pp. 50, 2014.

[2] M. H. Karimi and D. Asemani, "Surface defect detection in tiling Industries using digital image processing methods: Analysis and evaluation," in *ISA Transactions*, vol. 53, no. 3, pp.834-844, 2014.

[3] R. Ren, T. Hung and K. C. Tan, "A Generic Deep-Learning-Based Approach for Automated Surface Inspection," in *IEEE Trans. on Cybernetics*, vol. 48, no. 3, pp. 929-940, 2018.

[4] D. Kwon, H. Kim, J. Kim, S. C. Suh, I. Kim, and K. J. Kim, "A survey of deep learning-based network anomaly detection," in *Cluster Computing*, 2017.

[5] H. Y. T. Ngan, G. K. H. Pang, and N. H. C. Yung, "Automated fabric defect detection-A review," in *Image and Vision Computing*, vol. 29, no. 7, pp. 442-458, 2011.

[6] X. Xie, "A Review of Recent Advances in Surface Defect Detection using Texture analysis Techniques," in *Electronic Letters on Computer Vision and Image Analysis*, vol. 7, no. 3, pp. 11-22, 2008.

[7] J. L. Bouchot, G. Stübl, B. Moser, "A template matching approach based on the discrepancy norm for defect detection on regularly textured surfaces," in *International Conference on Quality Control by Artificial Vision*, 2011.

[8] G. H. Hu, "Optimal ring Gabor filter design for texture defect detection using a simulated annealing algorithm," in *International Conference on Information Science, Electronics and Electrical Engineering (ISEEE)*, 2014.

[9] I. Goodfellow, J. Pouget-Abadie, Mehdi Mirza, B. Xu, D. Warde- Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative Adversarial Nets," in *Advances in Neural Information Processing Systems (NIPS)*, pp. 2672-2680, 2014.

[10] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein gan," *arXiv preprint arXiv:1701.07875*, 2017.

[11] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courille, "Improved Training of Wasserstein GANs," in *Advances in Neural Information Processing Systems (NIPS)*, pp. 5767-5777, 2017.

[12] J. Zhao, M. Mathieu, and Y. LeCun, "Energy-based generative adversarial network," *CoRR, abs/1609.03126*, 2016.

[13] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, "Infogan: Interpretable representation learning by information maximizing generative adversarial nets," in *Advances in Neural Information Processing Systems*, pp. 2172–2180, 2016.

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

[15] Y. Bengio, A. Courville, and P. Vincent, "Representation Learning: A Review and New Perspectives," in *IEEE Trans. On Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798-1828, 2013.

[16] Z. C. Lipton and S. Tripathi, "Precise Recovery of Latent Vectors from Generative Adversarial Networks," *arXiv preprints arXiv:1702.04782*, 2017.

[17] J. Y. Zhu, P. Krähenbühl, E. Shechtman and A. A. Efros. "Generative Visual Manipulation on the Natural Image Manifold," in *European Conference on Computer Vision (ECCV)*, 2016.

[18] A. Bora, A. Jalal, E. Price, A. G. Dimakis, "Compressed Sensing using Generative Models," *arXiv preprint arXiv:1703.03208*, 2017.

[19] J. Bruna, A. Szlam, Y. Lecun, "Signal recovery from pooling representations," *arXiv preprint arXiv:1311.4025*, 2013.

[20] K. F. Wang, C. Gou, Y. J. Duan, Y. L. Lin, X. H. Zheng, and F. -Y. Wang, "Generative adversarial networks: introduction and outlook," IEEE/CAA J. of Autom. Sinica, vol. 4, no. 4, pp. 588-598, Oct. 2017.

[21] Z. Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," in *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600-612, 2004.

[22] L. Maaten and G. Hinton, "Visualizing Data using t-SNE," in *Journal of Machine Learning Research*, 2008.

[23] D. C. Dowson and B. V. Landau, "The Fréchet distance between multivariate normal distributions," in *Journal of Multivariate Analysis*, vol. 12, no. 3, pp. 450-455, 1982.

[24] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium," *arXiv preprint arXiv:1706.08500*, 2017.

[25] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *CoRR, abs/1409.1556*, 2014.