Proceedings of the 2018 IEEE/ASME International
Conference on Advanced Intelligent Mechatronics (AIM),
Auckland, New Zealand, July 9-12, 2018

# Industrial Anomaly Detection and One-class Classification using Generative Adversarial Networks

Y. T. K. Lai [1], J. S. Hu [1,2], Y. H. Tsai [2,3], and W. Y. Chiu [3]
[1]Dept. of Electrical Control Engineering, National Chiao Tung University, Taiwan.
[2]Dept. of Computer Science, National Chiao Tung University, Taiwan.
[3]Mechanical and Mechatronics Systems Research Labs (MMSL), Industrial Technology Research Institute (ITRI), Taiwan.

***Abstract*— Industrial image datasets for quality inspection are mostly sparse in defects. It is then hard for both automated optical inspection (AOI) machines and simple neural network classifiers to inspect all defects effectively. In this work, we develop a novel framework for industrial anomaly detection in one-class classification manner, which utilized pre-trained generative adversarial networks (GANs) as the rule of thumb to perform anomaly detection. Our results show that GANs are able to capture arbitrary and structural industrial images and can effectively discern defects when the query images are defective.**

## I. Introduction

With the fast development of techniques in Artificial Intelligence, many advanced algorithms have been applied and verified on open-source datasets (e.g. ImageNet, COCO, CIFAR-10). Despite their good qualities and great performance on these popular datasets, the performance of these algorithms drops when directly applying on real-world industrial datasets, which possess large variations and uncertainties. Moreover, due to the demand of extremely low false-positive rate, industrial datasets are highly imbalanced, which means that qualified samples greatly outnumber defective samples. This predicament makes the training of neural network classifiers even harder.

In real situations, defects in industrial datasets, compared to the whole image, are often very small in size. Convolutional neural networks (CNNs) often perform poorly to capture these small features. Therefore, it is hard to develop general algorithms across different industrial datasets. One-class classification (OCC) [1] has been used for anomaly detection tasks [2, 3] by learning only the positive class of data, where the negative class of data is either absent, poorly sampled, or ill-defined. In industrial applications, where the qualified samples occupy nearly the whole dataset, it is then logical to adopt the idea of one-class classification. However, there has not been work of OCC for industrial datasets due to the lack of an effective procedure to handle the large variety of images.

In this paper, we proposed a novel one-class classification method for industrial anomaly detection. Instead of directly applying neural networks as classifiers on industrial datasets, we use the framework of generative adversarial networks (GANs) [4] to perform anomaly detection. GANs have been successfully used for dataset training such as MNIST, CelebA, or LSUN etc. A major advantage of GANs is that they are able to capture the distributions of the input samples of these datasets for its ability to represent the associated contents using an effective model-based approach. This advantage is helpful in industrial datasets, where the positive data possess large variance and its number far exceeds negative data. When we have large number of qualified[1] samples, it is then likely to represent (or learn) the distribution of these data through the methods of GANs. Subsequently, the defective samples, which do not conform to the learned distributions, exhibit different characteristics that can be discriminated easily. The proposed approach is successfully verified through a number of image datasets produced by manufacturing processes in this paper.

Our main contributions are summarized as follows. (1) We show that generative adversarial networks perform well on generating industrial image datasets; (2) We proposed a new method using a generative adversarial network for anomaly detection on industrial datasets in the one-class classification manner; (3) We reconstruct the image using the optimized latent vectors of several industrial datasets and show that it is able to visually discern the defects; (4) We provide a quantitative measurement to distinguish defective samples from qualified samples and can determine how defective the samples are.

## II. Related Works

### A. Classification and Anomaly Detection in Industry

The goal of anomaly detection is to detect the data that do not fit in the same distribution of normal data. Deep learning approaches [5, 6, 7, 8, 9, 10] have been used in industrial applications recently. Ren et al. [5], Kim et al. [7], and Park et al. [8] utilized pre-trained Convolutional Neural Network (CNN) models on open source datasets and discovered that transfer learning performs well on industrial datasets. Cha et al. [6] also designed a CNN for detecting crack damages using vision-based methods. On the contrary, Ritcher et al. [10] developed a new work flow for AOI machines based on deep learning techniques. However, these works still need large amount of both positive and negative data to train the classifiers.

### B. Generative Adversarial Networks

The original GAN structure was first proposed by Goodfellow et al. [4], and its training mechanism and

[1] In our work, we refer positive data to qualified samples, negative data to defective samples.

978-1-5386-1854-7/18/$31.00 ©2018 IEEE

functionalities have been improved and modified by several important works [11, 12]. Generally, a GAN consists of a generator $G$ and a discriminator $D$, where in most cases, are both deep convolutional networks. Radford et al. introduced Deep Convolutional Generative Adversarial Network (DCGAN) [13], which investigates on the stability of training GANs and shows comparatively prominent results among other models. They also demonstrated that arithmetic operations in latent space is feasible in producing customized results. In this paper, we will use the structure and configuration similar to DCGAN across our experiments.

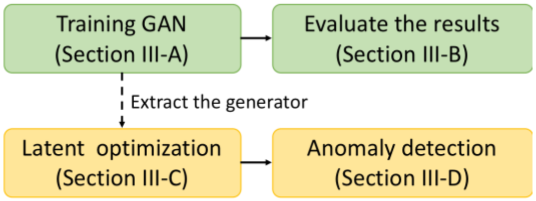
### C. Inverting Generators

In literature, many of the works in GANs are applied only to train the generator to generate images. However, several works [14, 15, 16] attempted to recover the latent representation of an image with respect to the generator, and they showed that it is possible to learn the inverse transformation of a generator. Similar to our work, Bora et al. [17] utilized the differentiability of GAN and applied the gradient descent algorithm to optimize the representation such that the corresponding image has small measurement error. Further, Bruna et al. [18] explored the theoretical conditions for a network to be invertible. For visualization of representations, Zeiler et al. [19] used the gradients of a particular feature of a convolution layer and propagated back to the image space in order to visualize what the feature stands for. Other works [20, 21] have used an optimization of a latent representation to generate realistic images. In their works, as well as ours, the loss used in training the generator and optimize the latent space are different.

## III. The Proposed Approach

Industrial anomaly detection is a challenging task because defects in industrial datasets are often not well-defined as in other areas and it is usually hard to know the variations of defects. For example, the defective areas in wood texture are blurry and small, and the distributions of the defects are different from one another. Our method aims at solving this problem with one-class unsupervised learning via GANs. The overview of the proposed method is shown in Fig. 1.

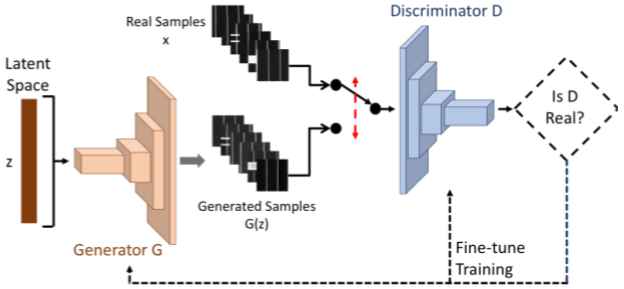
Figure 1. The overview of our proposed method.



### A. Training Generative Adversarial Networks

GANs are able to learn the distribution of real data via adversarial training. In our approach, we train a generative model and focus on the learned mapping from latent space to image space. We are given $N$ industrial qualified samples $x_n \in \mathbb{R}^{m\times m}$, where $n$ = 1, 2, ..., $N$, and $m$ is the height of input images. During training stage, we only feed $x_n$ into our GAN, so that we can learn the standard mapping of qualified images from latent space to image space.

Figure 3. The structure of the generative adversarial network we used in the proposed approach. First, latent codes are initialized in the latent space.



As shown in Fig. 3, a generative adversarial network is composed of two modules, a generator $G$ and a discriminator $D$. The generator learns the distribution $p_g$ over data $x$ by mapping $z$ sampling from the latent space $\mathbb{Z}$ to image space $\mathbb{R}$ as $G(z;\theta_g)$, where $\theta_g$ is the parameters of the $G$. Meanwhile, the discriminator learns the probability that $x$ comes from the distribution of real data $p_r$ or the distribution of generator $p_g$, which is denoted as $D(x;\theta_d)$, where $\theta_d$ is the parameters of $D$. In other words, during optimization, $G$ and $D$ play the following two-player minimax game with value function $V(G,D)$ as formulated in equation (1).

$$\min_G \max_D V(G,D) \\ = \mathbb{E}_{x\sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}\Big[\log\Big(1-D\big(G(z)\big)\Big)\Big], \qquad (1)$$

where $G(z;\theta_g)$ and $D(x;\theta_d)$ are shorthanded as $G(z)$ and $D(x)$ respectively. The objective of the discriminator is to maximize the probability of assigning real data from $p_r$ to true labels and generated samples from $p_g$ to fake labels correctly. Practically, rather than minimizing objective function $\log(1-D(G(z)))$ the generator is trained to maximizing $\log(D(G(z)))$ for the consideration of gradients when training. The generator continues improving its ability to generate realistic images, and the discriminator also increase its capacity to discern real data from generated data. Finally, through this adversarial training, the generated images are able to fit the distribution within the manifold of the real data. Note that the GAN we used only trains on single class of data, which in our method, we only trains on qualified samples. Defective samples no longer needed in the training of neural networks.

### B. Evaluation of Generated Results

Before applying the learned GAN for industrial anomaly detection, quantitative evaluations should be applied on the generated results to ensure the generated results are valid. T-distributed Stochastic Neighbor Embedding (t-SNE) [22] provides a visualization of the cluster, and we use it as a visualization of relationships between our generated results and the input real samples. To achieve this visualization, T-SNE performs dimension reduction by measuring scaled squared Euclidean distance of the incoming data in high-dimensional space while retaining that relationships in low-

978-1-5386-1854-7/18/$31.00 ©2018 IEEE

dimensional space. Equations of measurement in high-dimensional space and low-dimensional space is given in equations (2) and (3) respectively.

$$p_{ij} = \frac{\exp(-\|x_i - x_j\|^2/2\sigma^2)}{\sum_{k\neq l}\exp(-\|x_k - x_l\|^2/2\sigma^2)}. \qquad (2)$$

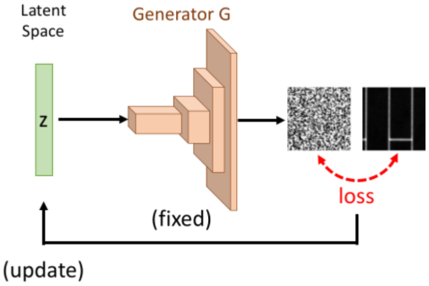
$$q_{ij} = \frac{\exp(-\|y_i - y_j\|^2)}{\sum_{k\neq l}\exp(-\|y_k - y_l\|^2)}. \qquad (3)$$

In equation (2), $p_{ij}$ calculates the pair-wise dissimilarity of high-dimensional data points $x_i$ and $x_j$, while $q_{ij}$ in equation (3) calculates that of low-dimensional data points $y_i$ and $y_j$.

### C. Learning Representation by Optimizing the Latent Space

The generators of GANs learn the mapping from latent space to image space. However, we are more interested in the mapping learned by the generator in the training stage because the latent space is full of information although it is not often visualized. Therefore, by optimizing the latent representations, we presume that the discrepancies are clear between the qualified samples and defective samples in the latent space.

Figure 4. Optimization process of the latent space. We use Laplacian pyramid L1 loss as our reconstruction loss and propagate back to the latent space.



Once the training of GAN is finished, we have learned the latent representations from the pre-trained generator (trained by ourselves). In this stage, we view the trained generator as the standard mapping and fix the parameters of the generator during optimization. Fig. 4 is an illustration of the optimization process. The gradient with respect to *z* is obtained by back-propagating the gradients through the generator from the image space ℝ to the latent space ℤ. In our settings, we use Stochastic Gradient Descent (SGD) to perform the optimization.

***Reconstruction Loss*** In the process of optimization, we define the loss of reconstruction in image space as the reconstruction loss. To ensure the optimization is able to learn the correct latent representation, we choose L1 loss as our reconstruction loss during optimization. On the other hand, squared-loss function leads to blurry effects, which is not desirable in reconstruction of industrial images. In addition, the loss used in training GAN is a convolutional network (the discriminator D), which focuses more on the edges. Hence, L1 loss is a better choice for reconstructing the correct latent representations.

### D. Latent Space Measurements

Quantitative measurements are directly applied after the optimization to determine how defective the incoming samples are. Our measurement modifies from Fréchet Inception Distance [23], which is a similarity measurement for generated images. We apply measurement similar to FID. In order to provide a concrete decision of whether this sample is defective or not, we use Fréchet distance [25] as our measurement between two multivariate normal distributions in the latent space. This calculation does not consume much resources for computation, thus it is able to do fast prediction.

Figure 5. Generated images based on different industrial datasets. Each part contains 16 images. (a) and (b): wood texture datasets. (c): solar panel dataset. (d): DAGM open-source industrial dataset.

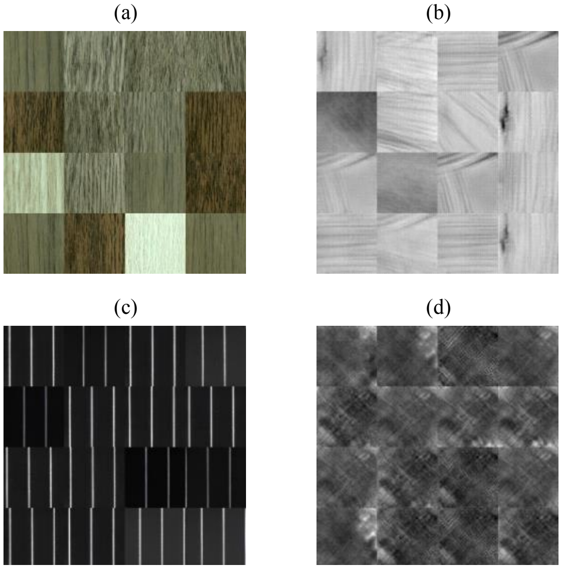
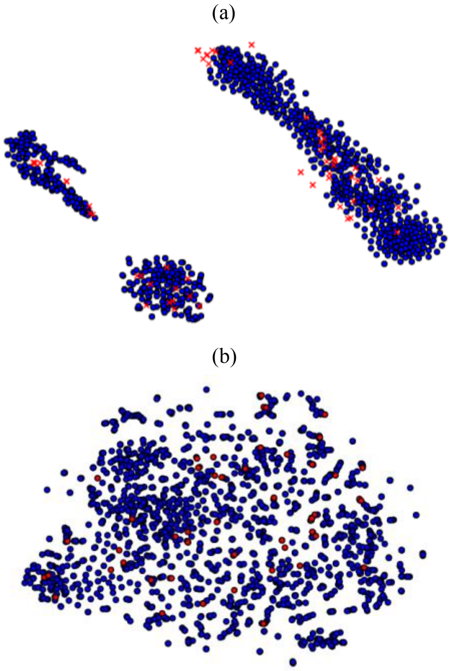


Figure 6. T-SNE visualization of the relationships between generated results and the real samples. This is directly applied in image space in order to observe the pixel-wise relationships without loss of information. (a): wood texture. (b): solar panel.



## IV. Experimental Results

We trained our models on a single 1080ti GPU and mainly choose two types of industrial datasets: the solar panel dataset and the wood texture dataset. They are highly structured texture and arbitrary texture respectively. We also tested on open-source DAGM industrial dataset in some experiments.

### *A. Generated Samples and Visualization*

The GAN we used is able to capture the probabilities of industrial datasets. In Fig. 5(a) and 5(b), we input random wood texture images into the GAN, whereas in Fig. 5(c), we input solar panels that are highly structured. After training on these different industrial datasets, we observe that GANs are robust in generating both non-structured images and highly-structured images. In Fig. 5(d), we further generate from the open-source industrial dataset.

In Fig. 6, t-SNE visualization is directly applied in image space to observe the relationships between generated results and the real samples. We do not perform Principle Component Analysis (PCA) or other dimension reduction operations in advance so there is no loss of information for comparison. From the visualization results, the generated samples (red crosses) are within the same manifold of the input real samples (blue circles) which shows the effectiveness of the generators. This visualization is important because if the generator cannot fully represent the mapping from the latent space to the image space, the optimization of the latent space would be biased.

### *B. Latent Space Convexity*

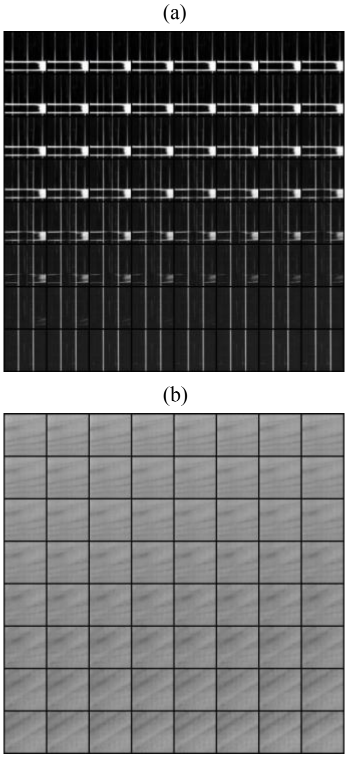
It is important that the latent space formed by the GANs is convex. Fig. 7 shows the interpolation between the learned latent vectors over qualified samples. We randomly sampled a pair of latent vectors from latent space and linearly interpolated 64 images (from top-left to bottom-right) between that pair. The reconstructed results show that our optimized representation space is smooth and exhibits convexity. This continuous property in the latent space provides us another way to visually evaluate whether the incoming sample is defective or not. It is helpful in anomaly detection because we can easily distinguish the qualified samples from defective ones if the interpolation of representations in the latent space is not in the original convex space.

Fig. 8 illustrates this phenomenon. We linearly interpolated 64 images between the pairs in the latent space from defective samples and found that the optimized representation space after reconstruction is not smooth and obviously generates images which are not meaningful. There are huge differences between qualified samples and defective samples in the latent space, whereas these differences are almost undiscernible in the image space.

### *C. Identification of the Defects*

The defects used in this paper are not very clear visually and the defective regions are vague (see Fig. 9). Nevertheless, we demonstrate that identifying the defects by using our proposed approach can be easily achieved. Because we only train our GAN only to generate qualified samples, it is hard for the generator to generate defective samples even if the reconstructed images of the latent vectors are trying to minimize the reconstruction loss. The reconstructed images shown in Fig. 10 best describe this observation. We can easily distinguish defective samples from qualified samples after reconstruction.

Figure 7. Interpolation of the learned latent vectors over qualified samples. (a): solar panel. (b): wood texture.



### *D. Fréchet Distance of Defective Latent Vectors*

In calculating Fréchet distances, we compare our generated samples with the optimized latent vectors of the qualified samples so that these comparisons have a common basis. For solar panel dataset, we compare three sets of latent vectors in Table I: (1) qualified generated data; (2) type I defect of contamination; (3) type II defect of contamination. In the first row of Table I (generated from qualified solar panels), we compared the generated images with original qualified samples. Besides the t-SNE visualization described previously, Fréchet distance again verify that our generator can generate qualified samples effectively. Alternatively, it indicates that the proposed method learns the correct mapping from the latent space to the image space. Table II shows the comparisons of wood texture dataset. We also tested on different types of defects and show that our method can predict the defective samples effectively.

TABLE I. The Fréchet Distances of the Latent Space of Solar Panel Dataset

| **Name of Optimized Latent Vectors** | **Fréchet Distance** |
|---|---|
| Generated samples of solar panels | 0.03 |
| Type I defect of Contamination | 0.31 |
| Type II defect of Contamination | 0.33 |

Figure 8. Interpolation of the learned latent vectors over defective samples. (a): solar panel. (b): wood texture.

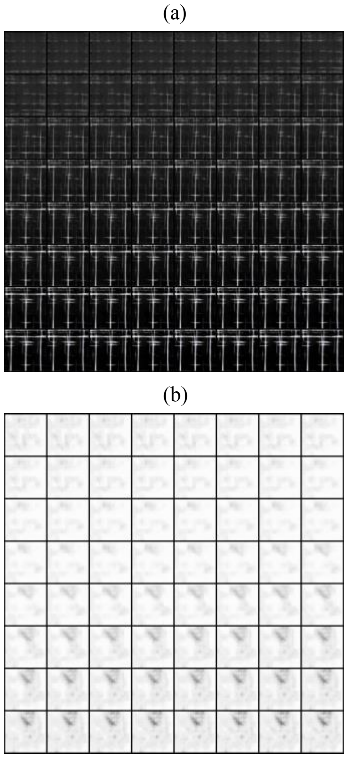


Figure 9. The defective samples along with the qualified samples of solar panel and wood texture datasets.

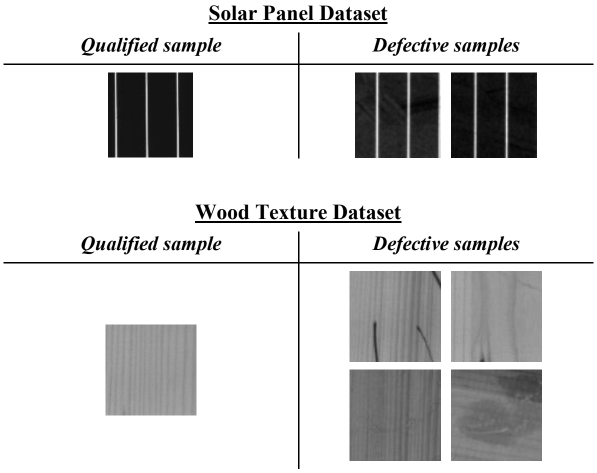


Figure 10. The comparisons of reconstructed images and input samples.

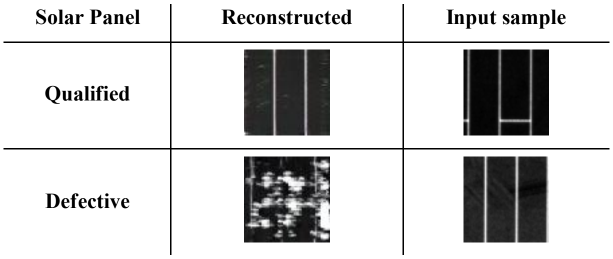


TABLE II. THE FRÉCHET DISTANCES OF LATENT VECTORS OF WOOD TEXTURE DATASET

| Name of Optimized Latent Vectors | Fréchet Distance |
|---|---|
| Generated samples of wood texture | 0.026 |
| Marker stains | 33824249.86 |
| Crack damages | 34560055.17 |
| Hot glue stains | 34635743.63 |
| Lubricant stains | 34410635.66 |

The smaller value of Fréchet distance means that it is more similar to the qualified samples. However, in Table I, the distances between two types of defects in solar panel are small. This is because defects of contamination of solar panels are nearly undiscernible under human eyes, even under the advanced optical devices in AOI machines. The optimized latent vectors increase the pixel-wise differences in the image space and magnifies this differences in the latent space. For wood texture dataset, the defects are easier to distinguish, so the Fréchet distances are in the same magnitude.

### E. Comparisons with Other Methods

In Table III, we compare our results with other unsupervised classification methods implemented on the same datasets. Gaussian Mixture Model (GMM) achieves the best accuracies among the other two methods: One-class Support Vector Machine (SVM) and Local Outlier Factor (LOF). However, the results show that our method is more robust at detecting the defects in these two datasets.

TABLE III. THE COMPARISONS WITH OTHER UNSUPERVISED METHODS

| Unsupervised Method | Accuracy (%) | |
|---|---|---|
| | *Solar Panel* | *Wood Texture* |
| Our Method | **93.75%** | **92.04%** |
| GMM | 86.33% | 68.53% |
| One-class SVM | 56.96% | 44.54% |
| LOF | 46.56% | 65.74% |

## V. CONCLUSION

We present a new framework for anomaly detection in industrial datasets by training a GAN using only one class of data in a dataset. Because GAN capture the distribution of the same class of data, it is possible to learn the mapping from the latent space to the image space by input only the qualified samples. This observation allows us to learn the latent representation by inverting the GAN.

Several discoveries are found during our experiments. First, the GANs are known for generating images with random distributions. Nevertheless, our results show that GANs are also good at generating structured data in industrial images. Second, we learn the latent representation by back-propagating to the latent space using gradient descent. We reconstruct the images and found that the latent space of qualified samples is convex and visually meaningful. Our proposed method can easily distinguish the defects from the reconstructed images after optimization. This shows a significant improvement since the defects are very vague and cannot be detected by current AOI machines directly. We further calculate Fréchet distances for each set of optimized latent vectors, which gives us a concrete quantitative measurement of how defective this sample is.

Our proposed method can effectively solve the current problems encountering in examining defects of industrial datasets. The one-class classification reduces the need to collect defective samples for training using other methods. We can not only detect the existing defects but also detect unknown defects in future processing events.

## ACKNOWLEDGMENT

We thank ITRI for their support.

## REFERENCES

[1] S. S. Khan and M. G. Madden, "One-Class Classification: Taxonomy of Study and Review of Techniques," *The Knowledge Engineering Review*, vol. 29, no. 3, pp. 345-374, 2013.

[2] C. Jing and J. Hou, "SVM and PCA based fault classification approaches for complicated industrial process," in *Neurocomputing*, vol. 167, pp. 636-642, 2015.

[3] M. Wan, W. Shang and P. Zeng, "Double Behavior Characteristics for One-Class Classification Anomaly Detection in Networked Control Systems," in *IEEE Trans. on Information Forensics and Security*, vol. 12, no. 12, pp. 3011-3023, 2017.

[4] 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.

[5] 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.

[6] Y. J. Cha, W. Choi, and O. Büyüköztürk, "Deep Learning-Based Crack Damage Detection Using Convolutional Neural Networks," in *Computer-Aided Civil and Infrastructure Engineering*, vol. 32, no. 5, pp. 361-378, 2017.

[7] S. Kim, W. Kim, Y. K. Noh, and F. C. Park, "Transfer Learning for Automated Optical Inspection," *2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, 2017*, pp. 2517-2524.

[8] J. K. Park, B. K. Kwon, J. H. Park, and D. J. Kang, "Machine learning-based imaging system for surface defect inspection," in *International Journal of Precision Engineering and Manufacturing-Green Technology*, vol. 3, no.3, pp. 303-310, 2016.

[9] 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.

[10] J. Ritcher, D. Streitferdt and E. Rozova, "On the development of intelligent optical inspections," *2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, 2017*, pp. 1-6.

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

[12] 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.

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

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

[15] 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.

[16] A. Creswell and A. A. Bharath, "Inverting The Generator Of A Generative Adversarial Network," *arXiv preprints arXiv: 1611.05644*, 2016.

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

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

[19] M. D. Zeiler and R. Fergus, "Visualizing and Understanding Convolutional Networks," in *European Conference on Computer Vision (ECCV)*, 2014.

[20] J. Portilla and E. P. Simoncelli, "A Parametric Texture Model Based on Joint Statistics of Complex Wavelet Coefficients," in *International Journal of Computer Vision*, vol. 40, no. 1, pp. 49-70, 2000.

[21] A. Nguyen, J. Yosinski, Y. Bengio, A. Dosovitskiy, and J. Clune, "Plug & play generative networks: Conditional iterative generation of images in latent space," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

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

[23] 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.

[24] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, "Going Deeper With Convolutions," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[25] 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.