IET Journals

The Institution of Engineering and Technology

Adversarial semi-supervised learning method for printed circuit board unknown defect detection

Wei Shi[1], Linlin Zhang[1] ✉, Yihui Li[2], Hong Liu[1]

[1]*Key Laboratory of Machine Perception, Peking University, Shenzhen Graduate School, Beijing 100871, People's Republic of China*
[2]*Shenzhen Skyworth-RGB Electronic, CO.,Ltd, Shenzhen, People's Republic of China*
✉ *E-mail: catherinezll@pku.edu.cn*

Abstract: Due to the lack of training data and fuzziness of unknown defects, unknown defect detection, which aims to identify no clearly defined defects, is still a challenging task. In practical industrial scenarios, defects on a printed circuit board account generally for a small proportion, so the data sets are highly biased towards no defect class. To this end, unknown defect detection can be treated as an anomaly detection problem. According to this, a semi-supervised learning method is proposed in this study to solve the above-mentioned problems. Inspired by the conditional generative adversarial network, the authors propose an improved end-to-end architecture for detecting unknown defects. The designed architecture is composed of three networks: a generator, a discriminator, and an encoder. Among them, the generator and the discriminator are trained by competing with each other, while collaborating to learn the distribution of underlying concepts in the target class. During training, the authors only train normal samples, and unknown defects do not appear in the process. In the testing phase, unknown defects are detected by calculating the distance between generated samples and real samples under the feature space. Experimental results over several benchmark data sets show the effectiveness of the model and superiority on state-of-the-art approaches.

1 Introduction

Defect detection of the printed circuit board (PCB) is critical in industrial manufacturing, which directly affects the quality of product production. Traditionally, PCB defect detection is handled manually, which is time-consuming, labour-intensive and unreliable. With the rapid development of computer vision technology, many methods are proposed for detecting the defects on PCBs. Recently, deep learning methods have also been applied to industrial detection [1–3]. To the best of our knowledge, these methods heavily rely on the large amounts of data and label information. In reality, it is hard to get a large number of data and annotations of the defects on PCBs. Besides, these methods can not detect the defects which are not annotated. They are limited to detecting well-defined defects classes. However, there is a large number of unknown defects in industrial detection. To this end, considering the defect detection problem as an abnormal detection task, we propose an improved semi-supervised learning method based on adversarial training.

1.1 Related works

1.1.1 PCB defect detection method: Lots of advanced approaches have been proposed for PCB defect detection. In tradition, there are mainly three kinds of methods in PCB defect detection: reference comparison methods [4, 5], non-reference comparison methods [6] and hybrid methods [7]. Reference comparison methods detect defects by comparing with defect-free PCBs. Non-reference comparison methods are based on pre-defined PCB design rules. Hybrid methods combine the above two kinds of methods. These methods are basically supervised methods which rely heavily on a large amount of labelled data. In addition, due to the need for clearly defined types of defects in these methods, unknown defects cannot be detected. However, in industry, undefined defects are inevitable. To this end, in this paper, semi-supervised learning is used for detecting unknown defects on PCBs.

1.1.2 Abnormal detection: Anomaly detection is a classic problem in computer vision, usually used in fraud detection, intrusion detection, and the medical field. They usually use an explicit representation of the normal data distribution in the feature space and determine the outliers based on the local density of the observation points in the feature space. The traditional method is based on distance metrics [8]. With the successful application of deep learning in many fields, it shows that deep learning can express the rich relationships and structures inherent in data. Based on the deep learning method, some unsupervised anomaly detection algorithms have gained lots of attention recently. For instance, Sabokrou *et al.* proposed a sparse representation model for video anomaly detection which can measure for separating normal and abnormal samples [9]. In some other works, by using the sample reconstruction to test samples of the target category, the reconstruction error can be used to judge whether the data is normal or abnormal [10–12]. That is, the high reconstruction error of the sample indicates that it is more likely to be an abnormal sample. Moreover, due to generative adversarial networks (GANs) can model high dimensional distributions of data, some works explored the use of GANs for the anomaly detection task [12–15].

However, due to the unavailability of the abnormal class during training, it is hard to train an end-to-end deep network. In recent works, few of them could train an end-to-end feature learning and classification model.

1.2 Our works

Inspired by the framework in [13], we propose an improved anomaly detection architecture for detecting the unknown defects on PCBs, especially for the tiny defects. The network proposed in this paper consists of three sub-networks, including a generator, a discriminator and an encoder. By training the normal samples, it transforms images from the image space into the feature vectors space. The distance between the generated image and the real image is compared under the vector space, and the gap is narrowed during the adversarial training so that the generator network can learn the feature expression of the normal class. However, since the generator network does not learn the characteristics of the abnormal samples during the training, the generated images and the corresponding real images have a large difference in the potential
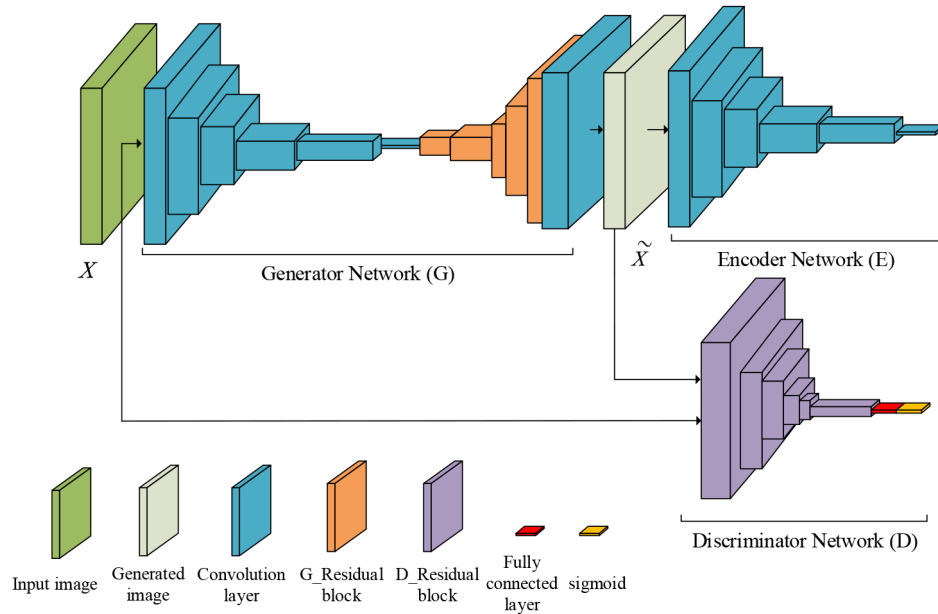
Fig. 1 *Architecture of Tinynomaly: the architecture contains three sub-networks a generator, a discriminator and an encoder. Here, the generator is not the symmetric simple convolutional network and the discriminator is not the same structure as the additional encoder*

vector space. Therefore, the normal class and the abnormal class can be classified by the distance between the generated image and the real image under the potential vector space. Considering that the main feature of the PCB is a texture feature and the semantic information of the PCB is weak, here we propose to use the residual network as the main structure of the network to enhance the proportion of the underlying features in the final decision. Moreover, regardless of the comparison at the image pixel level, the features obtained in our approach cannot achieve well image reconstruction but more representative of the image category information. It demonstrates the model in our approach achieves the reconstruction at the feature level.

The contributions of this study are threefold

(i) We propose a semi-supervised method based on the adversarial generation network for industrial detection, which effectively reduces the requirement for large amounts of labelled data. To the best of our knowledge, this is the first work to achieve semi-supervised learning using the adversarial generation network in industrial detection.
(ii) Different from previous works to obtain low-level features, we obtain image information at the semantic level through deep learning to enrich the diversity of PCB information to better represent image categories.
(iii) For detecting tiny unknown PCB defects, this paper proposes to select features by comparing the reconstructed image with the feature vector of the real image instead of the image level reconstruction effect. It reduces redundant information and makes the obtained PCB features more representative.

The remainder of the paper is organised as follows: Section 2 details the methods used in this work. Experiments on the PCB data set and two benchmarks are shown in Section 3. Finally, Section 4 concludes this work.

2 Tinynomaly approach

The aim of the work is to train a network that can detect unknown defects on PCBs, and the proposed detection framework is composed of three main modules: (i) a generator network (network G), (ii) a discriminator network (network D) and (iii) an additional encoder network (network E). The network G acts as the reconstruction, while the network D is discrimination. These three networks are learned in an end-to-end adversarial and unsupervised manner. In this section, we will cover them in details. The overall network architecture is shown in Fig. 1, and the pipeline of the proposed approach is based on [13]. The architecture contains two

encoders, a decoder, and discriminator networks, employed within three sub-networks. Here, the encoder–decoder pipeline acts as the generator, and the structure of the discriminator is an encoder. The additional encoder sub-network transforms the images from the image space to the latent vector space.

2.1 Generator network architecture

A lot of research works have shown that the reconstruction error obtained by training the target class samples by the auto-encoder can effectively distinguish the normal category and the anomaly category. Since auto-encoder only trains for the normal samples, it just learns the features of the normal samples. On the contrary, it would loose the features of the abnormal samples. Therefore, the reconstruction error of the abnormal category would be high. Using the same idea, the network G in the paper is an auto-encoder structure which consists of an encoder and a decoder.

The structure of most auto-encoder networks is symmetrical. Different from most works, the encoder and decoder structures in the paper are different. To implement the image compression, the encoder network is mainly composed of several simple convolution layers. Besides, convolution layers are utilised to realise invariant feature extraction without supervision. For reducing the loss of important information and improving network stability, the pooling layer is not used in the network structure. Moreover, after each convolution layer, the batch normalisation [16] operation is adopted, which increases the stability of the structure. Eventually, through the encoder network, the image would be transformed into an *m*-dimensional feature vector. The main requirement of an encoder is to retain (as much as possible) important information about the original data. To determine whether the encoding vector retains important information, there is a natural idea that the encoding vector should also be able to restore the original image. Therefore, the aim of training a decoder is to try to reconstruct the original image. For obtaining valid features, instead of the simple convolution layers, the decoder is a residual network structure composed of a series of residual blocks [17]. Through the residual blocks, it is better to retain the low-level features and pass the compressed features. The main purpose of the decoder is to restore the image to verify that the feature information extracted by the encoder is complete and valid.

2.2 Discriminator network architecture

The discriminator is trained to maximise the probability of distinguishing the real training samples and the generator samples. The discriminator in this paper is an encoder structure. Similar to

the encoder in network G, it is composed of a series of residual blocks. In addition, spectral normalisation [18] is added in the structure to solve the problem of unstable GAN training.

Spectral normalisation: Applying regularisation by means of spectral normalisation from the perspective of 'layer parameters', discriminator has the Lipschitz continuous condition [19]. Spectral normalisation replaces all the parameters in f with $w / \| w \|^2$. Here, the Lipschitz condition can be described as

$$\| f_w(x_1) - f_w(x_2) \| \leq C(w) \cdot \| x_1 - x_2 \|, \qquad (1)$$

where $C(w)$ is a constant which depends on the parameters rather than the input, f is the model and x is the input. Through this Function, the model is constrained by linear function which can satisfy the Lipschitz condition to increase the robustness of the model.

The residual blocks: Here, the residual blocks in the discriminator consist of the activation function ReLU [20], the average pooling layer and spectral normalised convolutional layers. Using the residual blocks makes the network to have a better converge and reduces the probability of gradient disappearing. The details of the residual blocks are shown in Fig. 2.

The dilated convolution: The size of the receptive field expands with the increased deep of the network layers. It requires the large size of convolution kernels which may lose some details information. However, the details information is important for PCB defect detection. In order to solve the loss of internal data structures and loss of spatially hierarchical information, the dilation convolution is used in the discriminator. The dilation convolution expands the receptive field without resolution loss and also reduces the additional parameters.

2.3 Additional encoder network architecture

An additional sub-network E of encoder structure comes on the heels of the G network. In order to keep the feature vectors obtained from the generated image as consistent as possible with the feature vectors of the real image compression, network E has the same network structure as the encoder in the generated network. The purpose of it is to obtain the efficient features of the generated image. Through a series of the convolution layers, the generated image is transformed into an m-dimensional feature vector which contains the important information of the image. Thus, the comparison between generated images and the real images transform from the image space into the feature vector space.

2.4 Adversarial training

Goodfellow *et al.* [21] introduced an effective method of adversarial learning between the generator and discriminator, which is called GANs. In this work, given M images without defects perform the distribution of the normal category. The input image x belongs to the defect-free category M first feeds to network G. Through the encoder part in network G, x is compressed to a vector z, where z is an m-dimensional vector. z is also referred to as the bottleneck feature of G and is assumed to have a minimum dimension containing the best representation of x. From experiments, we set $m = 100$ in the work. Then, the decoder part upsamples the vector z to reconstruct the image x as $\hat{x}$. Similar to the decoder part in network G, network E compresses the image $\hat{x}$ into its feature representation $\hat{z}$. For comparison, the dimension of the vector $\hat{z}$ is the same as the dimension of z. The discriminator output $D(x)$ can be interpreted as the probability that the given input of network D is the real image $\times$ sampled by the training data x or $G(z)$ generated by the network G. Network D tries to discriminate between actual data and the fake data generated by network G. Network D and network G are simultaneously optimised through the following two-player mini-max game with value function $V(G, D)$

$$\min_{G} \max_{D} V(G, D) = E_{x \sim p_{\text{data}}(x)}[\log D(x)]$$
$$+ E_{z \sim p_z(z)}[\log(1 - D(G(z)))], \qquad (2)$$

where p_{data} is the data distribution and $p_z(z)$ is the generator distribution to be learned through the adversarial min–max optimisation.

In the process of the training, the adversarial generator improves the ability to generate realistic images, and the discriminator improves the ability to correctly identify real images and generate images. When the adversarial training is completed, the generator has learned from the potential space representation z to the mapping $G(z)$ of the real image x.

In this paper, encoder loss and adversarial loss are proposed for the tiny unknown defects detection. Different from GANomaly [13], we remove reconstruction loss and only perform the detection in feature vector space. In our opinion, reconstruction is not a necessary condition for important feature selection, especially for the abnormal region occupying a small proportion of the whole image. The basic principle of important features should be the ability to identify the sample from the entire data set, i.e. to extract (the most) unique information about the sample. The majority of the PCB defects are tiny, such as mouse bite, spur and so on. Therefore, the anomaly category and the normal category may be similar, and it increases the difficulty of distinguishing between normal and abnormal category. Therefore, unlike most existing work, the method proposed for the PCB unknown defect detection in this paper no longer compares images in the pixel level, but only compares the feature vectors in the potential vector space. Moreover, the model adding reconstruction loss is proposed for the anomaly area taking up a large proportion in the whole image.

The encoder loss enhances the similarity between the feature vector compressed by generated image $G(x)$ and the feature vector compressed by the real image x. The aim of it is to minimise the distance between the bottleneck features of the input x and the encoded features of the generated image $\hat{x}$. It is formally defined as

$$L_{\text{enc}}(x) = \frac{1}{m} \| D(x) - D(G_E(x))) \|, \qquad (3)$$

where m denotes the number of the vector dimension which is equal to 100 in our experiments.

Since lacking label information during adversarial training, learning the features of classification is not the aim of the discriminator. Instead, it concentrates on learning good representations. Thus, during adversarial training, the training objective of the discriminator regardless of classification but instead, it uses the idea of feature matching to improve the mapping to the latent space. Therefore, the adversarial loss in the paper is also under the latent vector space. It compares the difference between the generated image and the real image through the last fully connected layer in the discriminator. Using a richer intermediate feature representation of the discriminator, it is formally defined as:

$$L_{\text{adv}}(x) = \frac{1}{m} \| G_E(x) - E(G_D(G_E(x))) \|^2. \qquad (4)$$

The reconstruction loss is a similarity comparison of the generated image and the original image at the image level. In the case where the abnormal region occupies a large area in the whole image, and the image lacks details, the extracted features are limited. Therefore, the difference between the abnormal image and the normal image mainly lies at the image level

$$L_{\text{rec}}(x) = \frac{1}{m} \| x - G_D(G_E(x)) \|. \qquad (5)$$

For mapping to the latent space, the overall loss is defined as the weighted sum of three components

$$L(x) = \alpha L_{\text{enc}}(x) + \beta L_{\text{adv}}(x) + \lambda L_{\text{rec}}(x). \qquad (6)$$
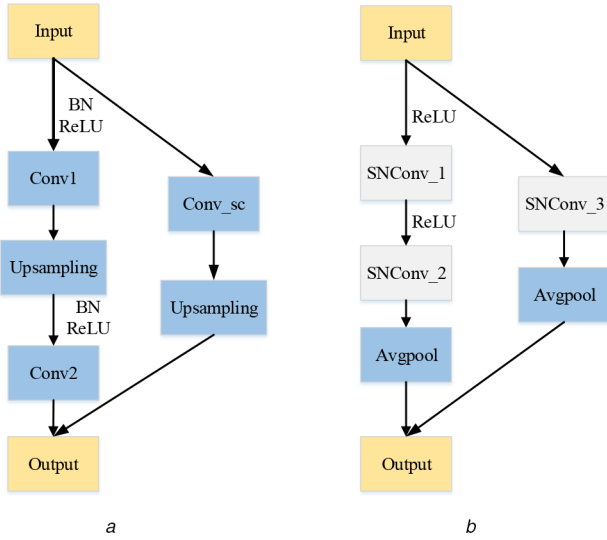
Fig. 2 *Details of residual modules*

(a) The residual module in G, *(b)* The residual module in D. The most striking divergence of the residual module between G and D is that the spectral normalisation layer [18] is used in D to increase the stability of training
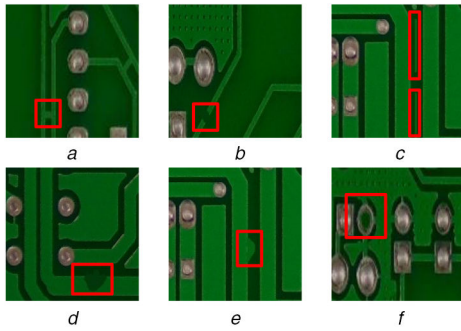

Fig. 3 *Unknown defects*

(a) Short circuit, *(b)* Open circuit, *(c)* Spurious copper, *(d)* Mouse bite, *(e)* Spur, *(f)* Missing hole. The red bounding box marks the defective area. All of them have no appearance in the training phase and belong to unknown defects

For the PCB unknown defect detection where the abnormal region is tiny, we set $\alpha = 1$, $\beta = 1$ and $\lambda = 0$.

3 Experiments and discussions

In this section, the proposed method is evaluated on PCB defect data set. The experimental results are analysed in details and are compared with state-of-the-art techniques. In addition, the validity of the model is verified on MNIST and CIFAR-10.

3.1 Data set and metric protocols

The data set used in this paper is split into the training set and test set which are two separate parts of the data set. Since there is not a public data set of PCB defects images, we perform image acquisition and get a small PCB defect data set. As shown in Fig. 3, the categories of defects include open circuit, short circuit, spurious copper, mouse bite, spur and missing hole. In the detection task, one category of defects is chosen to be an abnormal category. The PCB without defects is regarded as the normal category. In the original data set, the number of the defect-free images is 10 and each defects category has 20 images. There are commonly three or four defects in each defect image. The data set in this paper is extended by clipping the original image into the fixed-size image blocks. In our data set, the total number of normal images in the training set is 4400 in the test set. The number of images in each abnormal category in the test set is 50. Here, the number of images in the normal category is larger than the abnormal category.
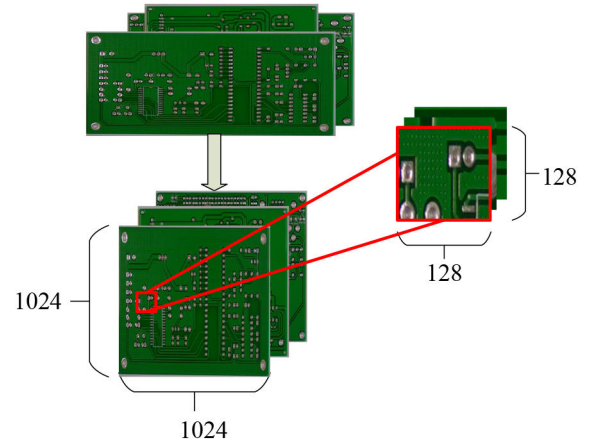

Fig. 4 *Preprocessing: Through the bilinear interpolation, the original image is resized into the same size* 1024×1024*, and the image blocks are obtained as the network input through the sliding window processing. The size of the image block is* 128×128

3.2 Implementation details

In this section, we briefly introduce the image pre-processing and the implementation details of our approach.

Pre-processing: Since the defects in the PCB are tiny and the circuits among it are dense, it is not easy to detect. To solve the difficulty in detection, sliding window processing is taken in the paper. As shown in Fig. 4, first, the original PCBs are scaled to the same size (1280×1280). The slider processing is performed in a fixed step size to obtain a small image block of the same size. Here, to expand the data set, the step size is chosen to be 32, 64 and 128 pixels. In addition, the image blocks are different scales of the original image but they are resized to the size of 128×128 in the final. We obtain the number of defect-free image blocks we need from ten defect-free PCBs as the normal samples.

Training details: The approach is implemented in PyTorch (v0.4.0 with Python 3.6.5). Adam [22] is chosen to optimise the networks and the initial learning rate of it is set as 0.0001. Meanwhile, the batch sizes of the training set and test set are both 16. Here, each abnormal class is trained for 200 epochs.

Testing process: In the test phase, the whole image is divided into patches of the same size, and then the image patches are fed into the model for classification.

Metrics: The evaluation of the experimental results are based on the area under curve (AUC) [23]. AUC is a model evaluation index which is often used as the evaluation of the two-category model. It is defined as the area under the receiver operating characteristic curve. AUC evaluates the two-category model by the probability that the positive example is in front of the negative example. It can well describe the overall performance of the model. AUC is defined in

$$AUC = \frac{\sum_{i \in \text{positiveClass}} \text{rank}_i - (M(1 + M)/2)}{M \times N}, \tag{7}$$

where M is the number of positive samples and N is the number of negative samples.

3.3 Results on PCB defect data set

The results in Table 1 show that our approach improves the detection performance among different defects. We consider each class in the data set as an abnormal class and the defect-free class as a normal class. Here, * denotes reconstruction loss is added in our model. Compared with GANomaly, our network structure is more capable of detecting tiny defects. The main reason is that the comparison of GANomaly is under the image space, while our approach only under the feature vector space without the reconstruction loss. Compare Tinynomaly with Tinynomaly*, the results demonstrate our approach without reconstruction loss

Table 1 AUC results on PCB defect data set

Method	Average AUC	Short circuit	Open circuit	Spurious copper	Mouse bite	Spur	Missing hole
GANomaly [13]	0.559	0.553	0.587	0.556	0.552	0.468	0.637
Tinynomaly	**0.869**	**0.876**	**0.908**	**0.899**	**0.810**	**0.860**	**0.861**
Tinynomaly*	0.728	0.709	0.734	0.661	0.756	0.773	0.734

*denotes that reconstruction loss is added in our model.
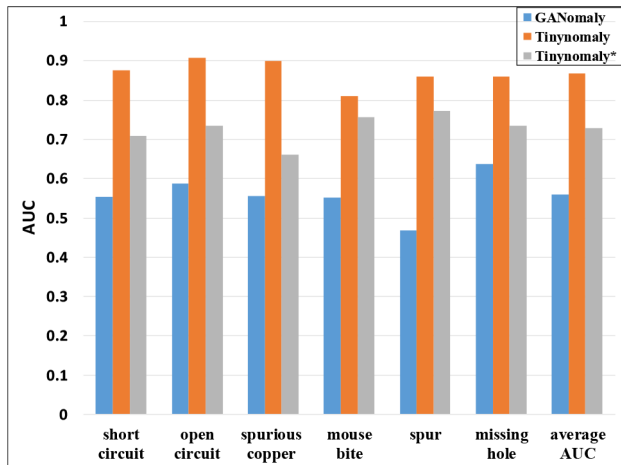
Fig. 5 *Diagram of AUC results on PCB data set*

function achieves better performance on PCB defect data set. GANomaly depends on the performance of image reconstruction and it is hard to distinguish the abnormal category and the normal category when they are similar to each other. Thus, this method is not suitable for tiny defect detection. It leads to weak detection performance on PCB defect data set. Instead, the approach without reconstruction loss in the paper concentrates on the representation of features and pays no attention to image reconstruction. The reason is that the PCB is complicated and the defects only occupy a small part of the image. It leads to more redundant feature information in image level comparison. Therefore, although the approach without reconstruction loss in the paper does not perform well at the reconstructed pixel level, its reconstruction at the feature level is successful. Therefore, the approach can capture the important features of the normal category which can help distinguish the abnormal category and the normal category.

In a word, instead of the accurate recovery of the pixel level, the model without reconstruction loss in this paper learns the advanced abstract features. The experiments denote the best feature for PCB defect detection task is the worst one to reconstruct the input at the pixel level. The model in our approach can extract effective features, especially for the tiny defects. Fig. 5 presents the experimental results more intuitively. The experimental results on the PCB data set show that important features can be obtained by comparing them from the feature vector level.

3.4 Results on MNIST and CIFAR-10

To evaluate our detection model, the other two benchmarks: MNIST [24] and CIFAR-10 [25] is used. In addition, we compare and do the analysis with other anomaly detection methods on these two benchmark data sets.

3.4.1 Results on MNIST: The MNIST data set is a handwritten digital data set where each image is a single number from 0 to 9. Treating one category as an anomaly and the rest categories as a normal category, there are 10 sets of experiments in total. The model is trained for 15 epochs on MNIST. Table 2 presents the results obtained on MNIST. Each digit is regarded as an abnormal class. All but Tinynomaly and Tinynomaly* results are obtained from [13]. Compared with some typical anomaly detection methods, our method achieves the best AUC performance among most anomaly categories on MNIST data set. Our method without image reconstruction loss performs better than VAE [14], AnoGAN

[15] and EGBAD [12] but worse than GANomaly. The main reason is that the anomaly area is the whole image in MINIST and the difference between the anomaly category and the normal category is the pixel-level to a large extent. Besides, a single channel is used as the input, texture, colour and some details are ignored. The feature distribution difference between the normal class and abnormal class is weaker than the appearance distribution difference. Therefore, anomaly detection in the image space is easier, which results in a different representation of the PCB data set and the MNIST data set. In the condition, comparing the image in the pixel-level is required. Adding the reconstruction loss which compares the image under image space, our approach is superior to the GANomaly. It represents our method can extract effective image features through better fusing the image space and potential feature space.

3.4.2 Results on CIFAR-10: The CIFAR-10 data set consists of colour images in ten classes, with 6000 images per class. The same with MNIST, one class is chosen as the normal and the rest are considered as the abnormal. The results are shown in Table 3. Our method can achieve the best performance both with or without the reconstruction loss. It demonstrates whether the main judgment basis is the pixel difference at the image level or the distribution difference at the feature level, our model can extract effective features containing the category representation information. Unlike the MNIST data set, the images in the CIFAR data set are based on colour images and contain more image feature information, which facilitates identification in the feature space.

Each class in the data set is regarded as an abnormal class. All but Tinynomaly and Tinynomaly* results are obtained from [13]. The results show some categories as abnormal classes perform slightly worse in the experiment. The main reason is that some categories in the data set have greater commonality, which leads to a higher similarity of the obtained features and misjudgment.

4 Conclusions

This paper presents a semi-supervised method based on GANs, which can detect unknown defects on PCBs, especially for the tiny defects. By training an end-to-end network which consists of a generator, a discriminator and an encoder network, the identification of images only needs to train the normal category. It weakens the requirement of data equilibrium and data quantity. In other words, the model enables the novel category can be detected without training.

Replacing detection from image space to feature vector space, the method addresses the issue of tiny unknown defect detection. Unknown defect detection is realised in the feature vector space by comparing image feature vectors. In contrast to the prior works, the approach underlines the deep semantics of features rather than judges the importance of features only by image reconstruction. The results show that our method achieves good performance on the tiny unknown PCB defect detection and has better generalisation ability.

5 Acknowledgments

Table 2 AUC results on MNIST data set

Anomalous digit	Average AUC	0	1	2	3	4	5	6	7	8	9
VAE [14]	0.332	0.515	0.075	0.640	0.245	0.335	0.325	0.430	0.145	0.500	0.110
AnoGAN [15]	0.430	0.610	0.300	0.535	0.440	0.430	0.420	0.475	0.355	0.400	0.335
EGBAD [12]	0.500	0.775	0.290	0.670	0.520	0.450	0.430	0.570	0.400	0.545	0.345
GANomaly [13]	0.757	0.850	0.330	0.920	0.771	0.770	0.800	0.880	0.780	**0.890**	0.580
Tinynomaly	0.706	0.906	0.465	0.827	0.645	0.693	0.671	0.753	**0.827**	0.673	0.600
Tinynomaly*	**0.850**	**0.915**	**0.704**	**0.942**	**0.900**	**0.905**	**0.880**	**0.865**	0.820	0.878	**0.693**

Table 3 AUC results on the CIFAR-10 data set

Anomalous class	Average AUC	Plane	Car	Bird	Cat	Deer	Dog	Frog	Horse	Ship	Truck
AnoGAN [15]	0.434	0.516	0.492	0.411	0.399	0.335	0.393	0.321	0.399	0.567	0.511
EGBAD [12]	0.462	0.577	0.514	0.383	0.448	0.374	0.481	0.353	0.526	0.413	0.555
GANomaly [13]	0.605	0.625	0.629	0.505	0.577	0.593	0.633	0.653	0.601	0.622	0.614
Tinynomaly	**0.774**	0.789	**0.860**	0.618	0.631	**0.791**	0.727	**0.758**	0.781	**0.970**	0.812
Tinynomaly*	0.768	**0.793**	0.826	**0.621**	**0.732**	0.657	**0.859**	0.683	**0.786**	0.834	**0.893**

6 References

[1] Soukup, D., Huber-Mork, R.: 'Convolutional neural networks for steel surface defect detection from photometric stereo images'. Int. Symp. on Visual Computing, Las Vegas, Nevada, USA, 2014, pp. 668–677

[2] Masci, J., Meier, U., Ciresan, D., *et al.*: 'Steel defect classification with max-pooling convolutional neural networks'. The 2012 Int. Joint Conf. on Neural Networks, Brisbane, Australia, 2012, pp. 1–6

[3] Cha, Y.J., Choi, W., Suh, G., *et al.*: 'Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types', *Comput.-Aided Civ. Infrastruct. Eng.*, 2018, **33**, (9), pp. 731–747

[4] Xiaojun, C.X.L.: 'A novel algorithm for inspecting and classifying the defects of printed circuit boards', *J. Huazhong Univ. Sci. Technol., Nat. Sci. Ed.*, 2010, **7**, p. 19

[5] Chauhan, A.P.S., Bhardwaj, S.C.: 'Detection of bare PCB defects by image subtraction method using machine vision'. Proc. World Congress on Engineering, London, UK, 2011, vol. 2, pp. 6–8

[6] Benedek, C.: 'Detection of soldering defects in printed circuit boards with hierarchical marked point processes', *Pattern Recognit. Lett.*, 2011, **32**, (13), pp. 1535–1543

[7] Lu, Z., He, Q., Xiang, X., *et al.*: 'Defect detection of PCB based on Bayes feature fusion', *J. Eng.*, 2018, **16**, pp. 1741–1745

[8] Idé, T., Papadimitriou, S., Vlachos, M.: 'Computing correlation anomaly scores using stochastic nearest neighbors'. Seventh IEEE Int. Conf. on Data Mining, 2007. ICDM 2007, Omaha, Nebraska, USA, 2007, pp. 523–528

[9] Sabokrou, M., Fathy, M., Hoseini, M.: 'Video anomaly detection and localisation based on the sparsity and reconstruction error of auto-encoder', *Electron. Lett.*, 2016, **52**, (13), pp. 1122–1124

[10] Cong, Y., Yuan, J., Liu, J.: 'Sparse reconstruction cost for abnormal event detection'. 2011 IEEE Conf. on Computer Vision and Pattern Recognition, Colorado Springs, CO, USA, 2011, pp. 3449–3456

[11] Xia, Y., Cao, X., Wen, F., *et al.*: 'Learning discriminative reconstructions for unsupervised outlier removal'. Proc. IEEE Int. Conf. on Computer Vision, Santiago, Chile, 2015, pp. 1511–1519

[12] Zenati, H., Foo, C.S., Lecouat, B., *et al.*: 'Efficient GAN-based anomaly detection', arXiv preprint arXiv, 1802.06222, 2018

[13] Akcay, S., Atapour-Abarghouei, A., Breckon, T.P.: 'GANomaly: semi-supervised anomaly detection via adversarial training', arXiv preprint arXiv, 1805.06725, 2018

[14] An, J., Cho, S.: 'Variational autoencoder based anomaly detection using reconstruction probability', *Spec. Lect. IE*, 2015, **2**, pp. 1–18

[15] Schlegl, T., Seebock, P., Waldstein, S.M., *et al.*: 'Unsupervised anomaly detection with generative adversarial networks to guide marker discovery'. Int. Conf. on Information Processing in Medical Imaging, Boone, USA, 2017, pp. 146–157

[16] Ioffe, S., Szegedy, C.: 'Batch normalization: accelerating deep network training by reducing internal covariate shift'. Int. Conf. on Machine Learning, Lille, France, 2015, pp. 448–456

[17] He, K., Zhang, X., Ren, S., *et al.*: 'Deep residual learning for image recognition'. Proc. IEEE Conf. on Computer Vision and Pattern Recognition, Las Vegas, Nevada, USA, 2016, pp. 770–778

[18] Miyato, T., Kataoka, T., Koyama, M., *et al.*: 'Spectral normalization for generative adversarial networks', arXiv preprint arXiv, 1802.05957, 2018

[19] Armijo, L.: 'Minimization of functions having Lipschitz continuous first partial derivatives', *Pac. J. Math.*, 1966, **16**, (1), pp. 1–3

[20] Nair, V., Hinton, G.E.: 'Rectified linear units improve restricted Boltzmann machines'. Proc. 27th Int. Conf. on Machine Learning, Haifa, Israel, 2010, pp. 807–814

[21] Goodfellow, I., Pouget-Abadie, J., Mirza, M., *et al.*: 'Generative adversarial nets'. Advances in Neural Information Processing Systems, Montreal, Quebec, Canada, 2014, pp. 2672–2680

[22] Kingma, D.P., Ba, J.: 'Adam: a method for stochastic optimization', arXiv preprint arXiv, 1412.6980, 2014

[23] Hanley, J.A., McNeil, B.J.: 'The meaning and use of the area under a receiver operating characteristic (ROC) curve', *Radiology*, 1982, **143**, (1), pp. 29–36

[24] LeCun, Y.: 'The MNIST database of handwritten digits', 1998. Available at http://yann.lecun.com/exdb/mnist/

[25] Krizhevsky, A., Nair, V., Hinton, G.: 'The CIFAR-10 dataset', 2014. Available at http://www.cs.toronto.edu/kriz/cifar.html