2023 IEEE Symposium Series on Computational Intelligence (SSCI)
Mexico City, Mexico. December 5-8, 2023

A Self-Supervised Few-Shot Detection Method for Magnetic Tile Defects Detection

Zhiyu Zhang
School of Software Engineering
Xi'an Jiaotong University
Xi'an, China
zhangzhiyu@stu.xjtu.edu.cn

Liangjie Dong
School of Software Engineering
Xi'an Jiaotong University
Xi'an, China
liannedli@ stu.xjtu.edu.cn

Hao Luo
School of Software Engineering
Xi'an Jiaotong University
Xi'an, China
luohao4122@stu.xjtu.edu.cn

Zhiqiang Tian*
School of Software Engineering
Xi'an Jiaotong University
Xi'an, China
zhiqiangtian@xjtu.edu.cn

***Abstract*—Current Defect detection methods have made significant progress on ideal datasets that typically contain a large number of defect samples. This enables the traditional defect detection methods to achieve great detection performance. However, in practical applications, the training samples obtained are often highly imbalanced, with the majority being non-defective samples and only a few being defective samples. It will generally lead to performance degradation of traditional methods if using such imbalanced samples for training. To address this issue, we propose a few-shot defect detection method based on self-supervised learning. Specifically, we propose to use a transfer learning strategy to transfer from traditional full-shot learning to few-shot learning. Next, we propose to process the training data in a self-supervised manner. As a result, the proposed method is enabled to achieve satisfactory detection performance on the industrial magnetic tile defect dataset. Experimental results verify the effectiveness of our proposed method.**

***Index Terms*—few-shot detection, magnetic tile defects, self-supervised learning, transfer learning**

I. Introduction

The objective of defect detection is to identify and locate defects and flaws in products, processes, systems, or components as quickly as possible to ensure quality, reliability, and performance, which plays an important role in industrial development [1], [2]. It is a key step to determine whether a product is allowed to be put into the market because any potential defect will lead to catastrophic accidents. For instance, in 2007, the I-35W highway bridge in the United States collapsed and the cause of this accident investigated by the Transportation Safety Board was that due to backward technology and untimely re-inspection, the gusset plate on the deck truss bridge was not detected, resulting in the rupture of the gusset plate on the deck truss bridge.

*Zhiqiang Tian is corresponding author. This work was supported by NSFC under Grant Nos. 62173269 and 61971343, the Natural Science Basic Research Plan in Shaanxi Province of China under Grant No. 2022JM-324, the Social Science Foundation of Shaanxi Province of China under Grant No. 2021K014, the Huawei Intelligent Base Project under Grant No. 22ZNJZ10, and Shaanxi Joint Key Laboratory for Artifact Intelligence, China.

Existing Defect Detection methods have made significant progress, especially on the datasets that generally contain a large number of defect samples. This enables traditional to achieve satisfactory detection performance [3], [4]. Shang *et al.* [5] cropped the original image and fine-tuned it in a convolutional neural network to extract part features for railway classification. They are combined with an improved loss function as supervision, which significantly promotes the accuracy of their detectors. Wang *et al.* [6] also established a model to solve the defect detection problem using a deep convolutional neural network that automatically extracted image features. This network was robust to noise. Fully convolutional networks have pixel-level prediction capabilities, so Wang *et al.* [7] incorporated a fully convolutional network to extract features from tire images. They combined upsampling operations to obtain outputs with the same dimensions as the original image and used multi-scale feature maps to refine the rough segmentation results. Panda *et al.* [8] used a defect plaque feature-driven recurrent neural network to complete the detection task. They used an improved cumulative zero-count local binary algorithm and directional differential energy patch features to train the recurrent neural network to classify boundary pixels. The experimental results showed high localization accuracy of the boundaries.

However, the methods discussed above universally require a large amount of samples for training, which is not always practicable in real-world applications. Moreover, they may suffer from high risk, low efficiency, and high cost, which severely limit the development of traditional manufacturing industries. Therefore, it is of great necessity to find a more efficient and low-cost way to defect detection. With the rapid improvement of modern computing power and the development of image processing algorithms, the completion of industrial product defect detection based on computer vision has attracted engineers, but utilizing computers to deal with defect detection still has challenges such as differences of

defects in various types, few defective samples, blurred borders between defects and background, and inconsistent labeling [9].

Considering the above issues, we proposed a few-shot detection method based on self-supervised learning for industrial magnetic tiles, which benefits improving the efficiency of detecting defection in traditional industries, liberating manual detection, reducing detection cost, and promoting detection accuracy. The significant contributions of this paper are summarized as follows.

1) The proposed method effectively solves the imbalance of defective and non-defective samples in real-world scenarios.
2) The proposed method is universal, and it is applicable to various defect detection benchmarks.

The rest parts are organized as follows. The details of the proposed method and the overall framework are presented in the section Proposed Method. The procedure and details of the experiment are introduced in the section Experiments and Analysis. Conclusions, remaining problems, and future work are described in the section Conclusion.

II. Proposed Method

The cost of collecting, labeling, and organizing defective samples of the magnetic tiles is very large. The normal dataset has a majority of samples without defects and a minority of samples with defects. If all samples are used for training, the model will learn more features from those without defects, which is unfavorable for detecting defective samples. If a balanced dataset is used, the trained model may lack generalization ability and even overfit. A few-shot detection method for magnetic tiles based on self-supervised learning is proposed to address the above problems. A pre-trained model is obtained on a large-scale dataset with a similar distribution before training on the tile dataset. The performance of the model is shown to be comparable to that of the model based on supervised learning. Few-shot defect detection for tiles can be viewed as a mathematical model, where the input is the actual defect sample in the industry and the output is the position and label of defects. Sample enhancement and classifier training are based on self-supervised learning, and subsequent training is performed on the few-shot detection model.

A. Detection Model

Most generic detection models are based on large amounts of data and supervised learning. However, due to various limitations of some specific scenarios, obtaining a large number of samples is impossible. In addition, supervised learning requires a large amount of manual labeling, but the time and labor costs of labeling are enormous, and the labeling relies on professional workers. Otherwise, incorrect labels are detrimental to the model learning the correct features. Therefore, the method of few-shot tasks and self-supervised learning is necessary for some industrial scenarios. Self-supervised learning can deal with a number of data labeling problems. Its goal is to make the model learn the corresponding feature information without labels. The proposed method applies self-supervised learning to the few-shot problem and exhibits excellent performance. The magnetic tile defect dataset is used as input, which typically contains 6 types of labels and a total of 350 samples per label.

B. Self-supervised Learning Process

Fig. 1 illustrates the self-supervised learning process in the proposed method, which is a sort of contrast-based self-supervised learning. The contrast refers to calculating the similarity for several sets of input data. The contrast network learns a set of features about the unlabeled data. Then it fine-tunes the features with the help of a few random labels to maximize the similarity between different enhanced views of the same image via a contrast function while minimizing the similarity of enhanced views between different images. The features learned by the contrasting network are optimized by continuously repeating the above operations and updating the parameters. This ensures that enhanced samples belonging to the same image or category attract each other, while repelling samples from different images or categories. As shown in Fig. 1, a batch of images are randomly selected as inputs, and random transformations (including translation, rotation, mirror flip, and random cropping, etc.) are applied to each of them. One or more of these transforms are randomly selected and applied to the same input one. Repeat this operation two times for each image to obtain two enhanced images.

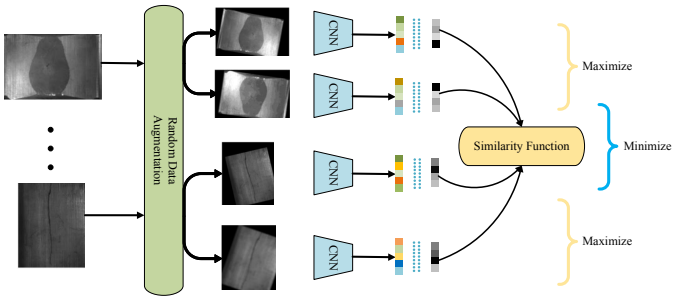

Fig. 1. Contrast-based Self-supervised Learning Process

The ResNet101 network is fed with these augmented views to generate corresponding features, which are further projected via an MLP module. The similarity function is used to calculate the comparison loss between projected features. The similarity function quantifies the degree of similarity between different views. For two views originating from the same input image, the similarity function is optimized for the maximum similarity. For enhanced views from different input images, the similarity value is minimized. To achieve this, the similarity function is constructed based on the cosine distance, as represented in Eq. (1).

$$Sim\left(m,n\right)=\frac{z_m^T z_n}{\lambda\left\|z_m\right\|\left\|z_n\right\|}, \tag{1}$$

where Sim represents the similarity calculation between views m and n; z_m and z_n denotes the projected features of m and n

respectively; λ is a hyperparameter that controls the distance of the inputs and takes the value range $[-1, 1]$. The similarity probability is calculated by softmax. The similarity probability of z_m and z_n is calculated by Eq. (2).

$$P\left(z_m, z_n\right) = \frac{e^{Sim(m,n)}}{\sum_{j \neq m}^{2N} e^{Sim(m,j)}} \ , \tag{2}$$

where Sim denotes the similarity function as in Eq. (1); N denotes the number of samples in a batch, $2N$ means two enhanced views are generated for each image; both m and j denote the order of the elements in all views of the batch. The contrastive loss between view m and n is defined as equation Eq. (3).

$$loss\left(m, n\right) = -\log\left(P\left(m, n\right)\right), \tag{3}$$

where m is the calculation subject. In summary, the total contrast loss of a batch is defined as Eq. (4).

$$L = \frac{\sum_{k=1}^{N}\left(\left(loss\left(2k-1, 2k\right)\right) + loss\left(2k, 2k+1\right)\right)}{2N} \tag{4}$$

where L denotes the total contrast loss of the batch; N denotes the number of all samples in the batch; and k denotes the sample order in the batch. After training the contrast learning model, it is used for the subsequent regression and classification sub-tasks. Experiments show that this contrast-based self-supervised learning model further improves the accuracy of magnetic tile defect detection.

C. Few-shot Defect Detection

To address the issue of insufficient labeled samples, previous work has mainly relied on meta-learning to create numerous meta-tasks and acquire corresponding meta-knowledge. This learned meta-knowledge is then transferred to a new task through transfer learning. Recently, Xin *et al.* [10] discovered that fine-tuning is simpler, more intuitive than meta-learning, and more effective for few-shot detection tasks. Fig. 2 shows the few-shot defect detection method proposed in this paper. As is depicted in Fig. 2, the training procedure is divided into two phases: in the pre-training phase, we utilize a large number of labeled samples that are not directly related to the target dataset, which is referred to as the base dataset. Pre-training is a supervised learning process that aims to learn a generalizable representation.

The pre-trained model is based on the Faster RCNN, whose loss is defined by Eq. (5).

$$L_{pre-train} = L_{RPN} + \eta L_{BBox}, \tag{5}$$

where $L_{pre-train}$ denotes the total loss of the pre-trained model; L_{RPN} denotes the loss of the RPN; L_{BBox} denotes the supervised loss of the bounding box (BBox); η denotes the relative weight, set to 0.9 in the experiments. The loss of the bounding box is given in Eq. (6).

$$L_{BBox} = L_{BBox_{cls}}\left(\boldsymbol{p}, \boldsymbol{p_{gt}}\right) + L_{BBox_{reg}}\left(\boldsymbol{b}, \boldsymbol{b_{gt}}\right), \tag{6}$$

where $\boldsymbol{p}$ denotes the predicted defect class; $\boldsymbol{p_{gt}}$ denotes the label corresponding to ground truth; $\boldsymbol{b}$ denotes the predicted bounding box and $\boldsymbol{b_{gt}}$ denotes the corresponding labeled bounding box. The classification loss of BBox is defined in Eq. (7).

$$L_{BBox_{cls}} = \sum\nolimits_i -\log\left(p_i p_{gt_i} + \left(1 - p_i\right)\left(1 - p_{gt_i}\right)\right) \ , \tag{7}$$

where p_i and p_{gt_i} is the element in $\boldsymbol{p}$ and $\boldsymbol{p_{gt}}$ respectively. The regression loss of BBox is defined in equation Eq. (8).

$$L_{BBox_{reg}}\left(\boldsymbol{b}, \boldsymbol{b_{gt}}\right) = \sum\nolimits_i SmoothL_1\left(b_i - b_{gt_i}\right), \tag{8}$$

where b_i and b_{gt_i} is the element in $\boldsymbol{b}$ and $\boldsymbol{b_{gt}}$ respectively. The $SmoothL_1$ is defined as Eq. (9).

$$SmoothL_1\left(x\right) = \begin{cases} 0.5x^2, & if \ \ |x| < 1 \\ |x| - 0.5, & otherwise \end{cases} \tag{9}$$

We design an Adaptive Anchor Module(AAM) and incorporate it into the RPN, which learns the features of various defects and adaptively generates anchors that match the defect shapes according to the learned parameters. The anchor box is determined by four variables: center point coordinates, length and width. The center coordinates determine the position of the anchor box. The length and width together determine the size of the anchor box.

Fig. 3 shows the composition of the AAM. The feature map output from RPN is fed into a CNN along with the GT of defects. The CNN extracts the central points of the GTs and maps them back to the feature maps. In addition, the CNN also outputs a central point probability map (CPPM), where each point is generated through two convolution layers and a sigmoid. For each centroid, the optimal width and height of the anchor box are estimated based on the feature map, the defect truth and the centroid probability map. Given an IOU threshold, enumerate some widths and heights, and adjust the widths and heights according to the IOU until reaching the given threshold. For centroids involving multiple GTs, repeat this operation on each GT separately. The module contains a convolutional layer with the shape of $1 \times 1 \times 1$ and a conversion layer as shown in Eq. (10). The pw and ph are generated as the feature of width and height respectively.

$$\begin{cases} w = \lambda \cdot e^{pw} \\ h = \lambda \cdot e^{ph} \end{cases} \ , \tag{10}$$

where λ is the hyperparameter, set to 8 in experiments. The pw and ph is the feature of width and height on the shape probability map (SPM) respectively. Because the flexible shape of anchor boxes results in that they cannot be calculated in the same receptive field, the offsets between anchor boxes are calculated via the deformable convolution and added to the input feature map. L_{RPN} denotes the loss of RPN calculated as Eq. (11).

$$L_{RPN} = L_{RPN_{cls}} + L_{RPN_{reg}} + \eta_1 L_{ach_c} + \eta_2\left(L_{ach_w} + L_{ach_h}\right), \tag{11}$$

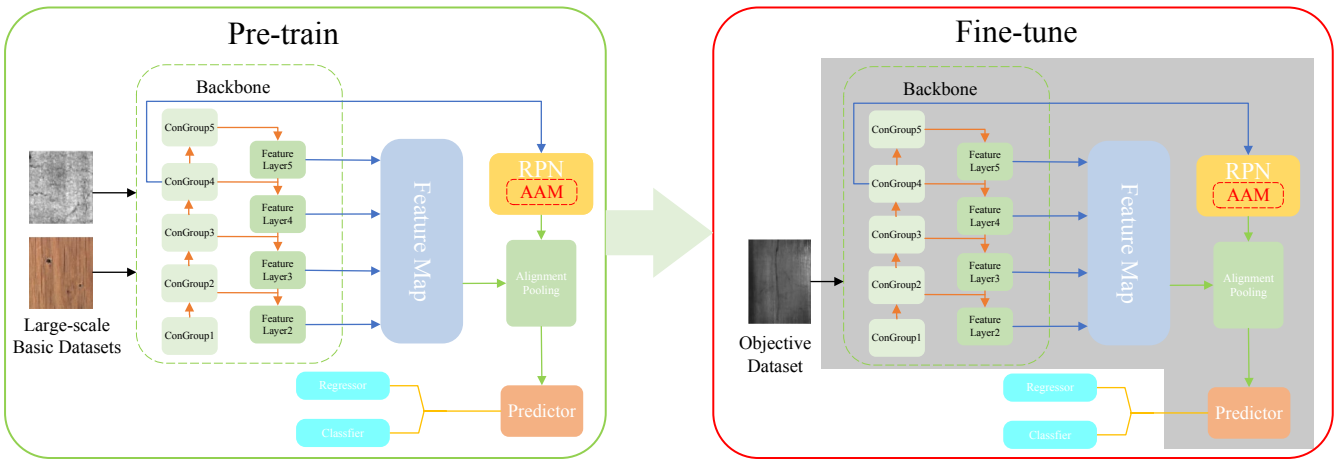

Fig. 2. Few-shot Detection Method for Magnetic Tile Defects

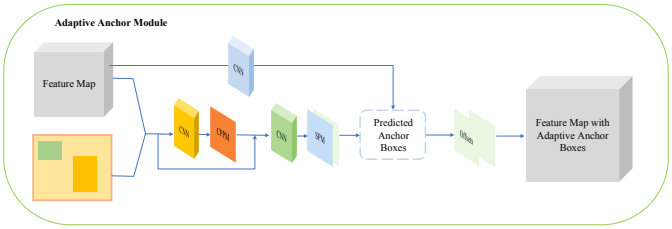

Fig. 3. Adaptive Anchor Module

where η_1, η_2 are the hyperparameters denoting relative weights and $\eta_1 = 1, \eta_2 = 1.2$ in experiments. L_{RPNcls} denotes the classification loss of the RPN. L_{RPNreg} denotes the regression loss of the RPN. They are calculated in the same way as $L_{BBoxcls}$ and L_{RPNreg}, L_{ach_c},L_{ach_w} and L_{ach_h} respectively denotes the center coordinate loss, width loss and height loss of the adaptive anchor. To balance the positive and negative samples, the loss of the adaptive anchor center coordinates is defined as Eq. (12).

$$L_{ach_c} = -\sum_{i}^{cls} \left(1 - p_i p_{gt_i}\right)^{\gamma} \log\left(p_i p_{gt_i}\right) L_{RPNcls}\left(p_i, p_{gt_i}\right), \tag{12}$$

where cls denotes the number of categories of all samples. L_{RPNcls} denotes the category loss of RPN. The p_i and p_{gt_i} is the element in $\boldsymbol{p}$ and $\boldsymbol{p_{gt}}$ respectively. The b_i and b_{gt_i} is the element in $\boldsymbol{b}$ and $\boldsymbol{b_{gt}}$ respectively. The width and height of the adaptive anchor are trained via IoU supervision and their losses are defined in Eq. (13).

$$\begin{cases} L_{ach_w} = SmoothL_1\left(1 - \min\left(\frac{ach_w}{gtb_w}, \frac{gtb_w}{ach_w}\right)\right) \\ L_{ach_h} = SmoothL_1\left(1 - \min\left(\frac{ach_h}{gtb_h}, \frac{gtb_h}{ach_h}\right)\right) \end{cases} \tag{13}$$

where $SmoothL_1$ was defined as equation (9). The ach_w and ach_h respectively denotes the width and height of the adaptive anchor. The gtb_w and gtb_h respectively denotes the width and height of the corresponding ground truth box.

III. Experiments and Analysis

A. Objective Dataset

The objective dataset in this paper is an industrial magnetic tile defect detection dataset, which is collected and compiled by the Chinese Academy of Sciences [11] on a real production line. The dataset consists of images in 6 types, their category labels are blowhole, crack, break, fray and uneven, and defect-free. The object magnetic tile defect dataset is a typical few-sample dataset. There are only 350 defective samples in total, but the number of non-defect samples reaches 952. The total number of samples is small and defective samples account for only 26.88%. Various types of defective samples and defect-free samples are divided into train set and test set at a ratio of 0.65 : 0.35.

B. Experimental Setup and Implement

Our experiments are executed on one single GeForce RTX 2080Ti single GPU with 11G RAM. Python =3.7.9, and the coding framework is Pytorch=1.7.1; CUDA=10.2 for accelerated parallel computing and CUDN=7.6.5. The average computational speed of the test process was 3.0 task/s. ResNet101 is chosen as the backbone. The baseline is the FsDet network proposed by Wang *et al*. The model is built based on the platform MMDetection [12]. We select Faster RCNN as the detection network and incorporate the feature pyramid network for fusing features on different levels. Optimize the total loss via stochastic gradient descent. The learning rate is set to 0.01 in the pre-training stage and 0.001 in the fine-tuning stage. The Momentum is set to 0.9 and the weight decay is set to 0.0001 in two stages. We replace ROIPooling with ROIAlign and introduce Soft-NMS.

The random image enhancement in the training process of the fine-tuning stage is implemented based on the imaging (a third-party library for Python), and the transformations include horizontal mirror flip, upper and lower mirror flip, Gaussian perturbation, mean perturbation, random rotation, random interception, sharpening, affine transformation and

contrast transformation, etc. In addition, the experimental results of Hinton *et al* [13]. indicate that combining random cropping and color transformations benefits training, so introducing a probability parameter to each transformation guides a limited biased selection of some transformations in the random enhancement process.

C. Comparison Experiment

The proposed method is compared with the existing advanced detection methods, and the experimental results are shown in Tables I and II. All experiments are based on the magnetic tile defect dataset, and the subjects in the experiments include the typical one-stage detection models YOLO [14] and SSD [15], the two-stage detection model Cascade RCNN and the key point detection model CornerNet [16]. All metrics including various AP and AR in the tables are derived from MS COCO [17].

TABLE I
COMPARISON WITH ADVANCED METHOD ON AP (%)

Method	Backbone	AP	$AP^{0.5}$	$AP^{0.75}$	AP^s	AP^m	AP^l
YOLO	Darknet	50.4	83.3	53.7	37.1	58.3	74.0
SSD300	VGG16	47.4	76.8	51.4	34.7	**59.8**	73.0
CascadeRCNN	ResNet101	50.9	83.5	51.5	42.8	48.0	60.6
CornerNet	Hourglass104	20.3	83.3	21.1	25.7	39.5	6.5
Ours	ResNet101	**60.1**	**92.7**	**68.0**	**51.3**	49.8	**76.1**

TABLE II
COMPARISON WITH ADVANCED METHOD ON AR (%)

Method	Backbone	AR^1	AR^{10}	AR^{100}	AR^s	AR^m	AR^l
CascadeRCNN	ResNet101	59.0	59.0	59.0	53.4	49.7	65.0
CornerNet	Hourglass104	45.6	45.6	45.6	39.6	55.5	47.0
SSD300	VGG16	58.7	58.7	58.7	45.8	63.5	78.6
YOLO	Darknet	59.0	59.0	59.0	42.3	**63.7**	**79.6**
Ours	ResNet101	**66.1**	**66.1**	**66.1**	**58.1**	51.8	78.3

Both Table I and Table II indicate that the proposed method outperforms its competitors on the few-sample defect data, where the detection accuracy surpasses the other competitors in most evaluation metrics. Cascade RCNN and YOLO are two commonly used detection models, that achieve high detection accuracy, but there is a gap between them and the method proposed in this paper. As for the dataset, due to the SSD300's poor performance on small targets, it shows a lower performance on the objective dataset. CornerNet is a keypoint-based detection method, where the comparison experiment also proves that for targets without clear boundaries like defects, the keypoint-based detection method is not applicable. The proposed method in this paper introduces some inductive biases for few-sample datasets and defects. The comparison experiment indicates that it outperforms the other mentioned models.

D. Ablation study

Table III shows the detection results of introducing the adaptive anchor module based on the three optimal freeze groups. FG_n denotes freezing the first n stages of the backbone network. The results indicate that introducing AAM based on frozen stage 1 severely degrades the detection accuracy. The purpose of fine-tuning is to learn new combinatorial features, and the introduction of AAM increases the number of parameters that need to be updated. This makes it difficult to learn new combinatorial features during fine-tuning, resulting in a decrease in detection accuracy after the introduction of the AAM. In contrast, introducing AAM based on either freezing the first 2 stages or the first 3 stages improves the detection accuracy. This is because more frozen parameters make it easier to update the parameters during the fine-tuning process, which promotes the model's ability to learn new combinatorial features. It is also demonstrated that AAM can obviously improve the detection accuracy with the optimal freeze group.

Table IV demonstrates the impact of introducing self-supervised learning (SSL) on the detection accuracy based on the freeze experiment. Combined with Tables III and IV, the performance under each metric is substantially promoted after introducing self-supervised learning. Tables IV also indicates that introducing self-supervised learning into the model based on FG2+AAM or FG3+AAM also promotes detection accuracy to some extent. This set of ablation experiment validates that the introduction of self-supervised learning can further promote the performance of the model. These combined factors make the model more adaptive for few-sample data and achieve state-of-the-art performance in the few-shot magnetic tile defect detection.

The lower level of the backbone extracts more general features, so freezing the parameters of the lower level helps to preserve general features. Freezing the parameters of the lower level not only guides the model to focus more on features with significant differences during fine-tuning, but it also reduces the hardware resources cost during training and accelerates the training process. Self-supervised learning enhances the robustness of the model and makes it more suitable for those datasets with a few samples.

E. Visualization of Defect Detection

Fig. 4 shows the visualization of defect detection on the objective dataset, including five categories of defective samples and one category of defect-free samples. The green bounding box in Fig. 4 denotes the ground truth, and the corresponding labels are also marked. The other five colors denote various categories of defects. The proposed method can efficiently and correctly detect all categories of defects. Also, the proposed method does not detect wrong bounding boxes for defect-free samples. In summary, the visualization results prove that the proposed method is competent in the defect detection tasks with few samples.

IV. CONCLUSION

In this paper, a few-shot magnetic tile defect detection method based on self-supervised learning is proposed. The proposed model that outperforms the other competitors is trained on a publicly available dataset of only 350 magnetic tile defect images, which can be applied to the real-world imbalanced dataset. A series of experiments were conducted

TABLE III
OPTIMAL FREEZE GROUPS AND ADAPTIVE ANCHOR MODULE (%)

	AP	$AP^{0.5}$	$AP^{0.75}$	AP^s	AP^m	AP^l	AR^1	AR^{10}	AR^{100}	AR^s	AR^m	AR^l
FG1+AAM	47.0	77.9	51.8	43.6	45.5	48.6	55.1	55.1	55.1	46.1	48.4	57.4
FG2+AAM	**57.3**	**87.6**	**57.2**	**48.2**	49.9	**65.2**	**63.0**	**63.0**	**63.0**	**56.1**	51.2	67.8
FG3+AAM	51.6	86.0	51.2	38.6	**50.8**	62.9	57.1	57.1	57.1	44.6	**52.6**	**69.0**

TABLE IV
OPTIMAL FG, AAM AND SSL (%)

	AP	$AP^{0.5}$	$AP^{0.75}$	AP^s	AP^m	AP^l	AR^1	AR^{10}	AR^{100}	AR^s	AR^m	AR^l
FG1+AAM+SSL	**60.1**	**92.7**	**68.0**	**51.3**	49.8	76.1	**66.1**	**66.1**	**66.1**	**58.1**	58.1	78.3
FG2+AAM+SSL	58.0	91.1	55.4	47.0	65.9	70.2	64.3	64.3	64.3	55.0	**69.5**	72.2
FG3+AAM+SSL	59.3	89.3	67.9	45.9	**66.5**	**78.1**	65.7	65.7	65.7	52.5	68.3	**82.7**

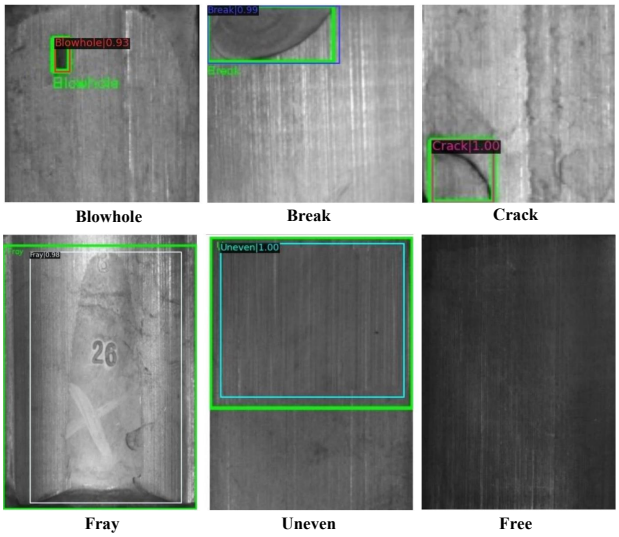

Fig. 4. Visualization of Defect Detection

from the quantitative and visualization perspectives to verify the positive contributions of the proposed partly frozen parameters, adaptive anchor module, and self-supervised learning. The research results have reference value for future research on few-shot defect detection.

REFERENCES

[1] M. Roy, S. K. Bose, B. Kar, P. K. Gopalakrishnan, and A. Basu, "A stacked autoencoder neural network based automated feature extraction method for anomaly detection in on-line condition monitoring," in *2018 IEEE Symposium Series on Computational Intelligence (SSCI)*. IEEE, 2018, pp. 1501–1507.

[2] Y. Li, D. Wei, X. Wei, K. Wu, Y. Liang, and X. Shen, "Defects detection of catenary suspension device based on image processing and cnn," in *2019 IEEE Symposium Series on Computational Intelligence (SSCI)*. IEEE, 2019, pp. 1756–1761.

[3] Y. Fu, A. R. Downey, L. Yuan, T. Zhang, A. Pratt, and Y. Balogun, "Machine learning algorithms for defect detection in metal laser-based additive manufacturing: A review," *Journal of Manufacturing Processes*, vol. 75, pp. 693–710, 2022.

[4] Z. Ren, F. Fang, N. Yan, and Y. Wu, "State of the art in defect detection based on machine vision," *International Journal of Precision Engineering and Manufacturing-Green Technology*, vol. 9, no. 2, pp. 661–691, 2022.

[5] L. Shang, Q. Yang, J. Wang, S. Li, and W. Lei, "Detection of rail surface defects based on cnn image recognition and classification," in *2018 20th International Conference on Advanced Communication Technology (ICACT)*. IEEE, 2018, pp. 45–51.

[6] T. Wang, Y. Chen, M. Qiao, and H. Snoussi, "A fast and robust convolutional neural network-based defect detection model in product quality control," *The International Journal of Advanced Manufacturing Technology*, vol. 94, pp. 3465–3471, 2018.

[7] R. Wang, Q. Guo, S. Lu, and C. Zhang, "Tire defect detection using fully convolutional network," *IEEE Access*, vol. 7, pp. 43 502–43 510, 2019.

[8] R. Panda, N. B. Puhan, A. Rao, D. Padhy, and G. Panda, "Recurrent neural network based retinal nerve fiber layer defect detection in early glaucoma," in *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*. IEEE, 2017, pp. 692–695.

[9] N. Zeng, P. Wu, Z. Wang, H. Li, W. Liu, and X. Liu, "A small-sized object detection oriented multi-scale feature fusion approach with application to defect detection," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–14, 2022.

[10] X. Wang, T. E. Huang, T. Darrell, J. E. Gonzalez, and F. Yu, "Frustratingly simple few-shot object detection," *arXiv preprint arXiv:2003.06957*, 2020.

[11] Y. Huang, C. Qiu, and K. Yuan, "Surface defect saliency of magnetic tile," *The Visual Computer*, vol. 36, pp. 85–96, 2020.

[12] K. Chen, J. Wang, J. Pang, Y. Cao, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Xu *et al.*, "Mmdetection: Open mmlab detection toolbox and benchmark," *arXiv preprint arXiv:1906.07155*, 2019.

[13] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*. PMLR, 2020, pp. 1597–1607.

[14] T. Wang, M. Qiao, M. Zhang, Y. Yang, and H. Snoussi, "Data-driven prognostic method based on self-supervised learning approaches for fault detection," *Journal of Intelligent Manufacturing*, vol. 31, pp. 1611–1619, 2020.

[15] L. Jiang, Z. Song, Z. Ge, and J. Chen, "Robust self-supervised model and its application for fault detection," *Industrial & Engineering Chemistry Research*, vol. 56, no. 26, pp. 7503–7515, 2017.

[16] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1492–1500.

[17] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, 2014, pp. 740–755.