IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 19 July 2022, accepted 1 August 2022, date of publication 5 August 2022, date of current version 12 August 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3196935

APPLIED RESEARCH

A Four-Stage Product Appearance Defect Detection Method With Small Samples

XIANG XIE[1], RONGFENG ZHANG[2], LINGXI PENG[3], AND SHAOHU PENG[1]
[1]School of Electronics and Communication Engineering, Guangzhou University, Guangzhou, Guangdong 510006, China
[2]College of Information Science and Technology, Zhongkai University of Agriculture and Engineering, Guangzhou 510225, China
[3]School of Mechanical and Electrical Engineering, Guangzhou University, Guangzhou 510006, China

Corresponding author: Shaohu Peng (pengsh@gzhu.edu.cn)

This work was supported in part by the Guangdong Provincial Scientific and Technological Project under Grant 2017B020210005, and in part by the Science and Technology Commissioner Project of Guangdong Provincial Department of Science and Technology under Grant KTP20210242.

ABSTRACT With the automation of industrial production, appearance defect detection based on machine vision plays an important role in product quality control. The scarcity of defect samples and real-time requirement are the main challenges in this field. Many existing studies are based on semantic segmentation network, but they cannot provide a classification confidence score for each image and only report the segmentation tasks metrics, which ignore that the positive or negative decisions are the key of defect detection. Therefore, this paper proposes a four-stage appearance defect detection model: contrast enhancement, segmentation, correction, and decision, which can achieve high detection accuracy with a severe shortage of positive samples. Since the proposed model simplifies U-Net to segment those candidate defect regions, and constructs a lightweight decision network based on the candidate regions and segmented mask, the proposed method not only achieves fast inference speed, but also obtain good performance with fewer defect samples. Experiments are implemented on three public datasets: magnetic tile dataset, Kolektor surface defect dataset and DAGM2007 dataset. The influence of each module on the detection accuracy is analyzed. Experimental results show that the proposed model achieves excellent performance comparing with other state-of-art methods.

INDEX TERMS Appearance defect detection, convolutional neural network, semantic segmentation, small defect samples.

I. INTRODUCTION

Appearance defects (such as cracks, spots, holes and wear) adversely affect the products performance, service life and users experience, so appearance defect detection is a vital link of quality control in industrial production.

Since manual inspection has some apparent disadvantages: inefficient, susceptible to the subjective judgment of inspectors, and increases labor costs, scholars proposed two main methods for appearance defect detection: the machine vision-based methods and the deep learning-based methods.

For a classic defect detection system based on machine vision, it usually consists of two parts: image collection and image processing. In 1983, Suresh *et al.* [1] designed a steel plate surface defect detection system, this is one of the earliest machine vision system applied in industrial production. They used CCD (Charge Coupled Device) cameras to collect images under three light sources, then Robert gradient operator [2] was used for edge enhancement, and the thresholding results were sent to a statistics-based classifier. This system can achieve real-time detection automatically, but the detection results are easily interfered by illumination and lack robustness, so it is not widely used in factories. Wang *et al.* [3] applied PCA (Principal Components Analysis) in submerged arc weld X-ray image defect detection and classified the collected features, although the inference time is short, the recognition accuracy is only 84.65%. Liu and Zheng [4] used Fourier transform to detect fabric defects, they transformed the image to the frequency domain for filtering, and then reconstructed the image by inverse

The associate editor coordinating the review of this manuscript and approving it for publication was Kumaradevan Punithakumar.

Fourier transform. However, in the frequency domain, background and defect features are easily mixed, resulting in the defects being filtered. In addition, there are algorithms that use SVM (Support Vector Machine) [5], template matching and KNN (K-Nearest Neighbor) classifier [6] to detect defect. These machine vision-based methods realized the automation and intelligence of defect detection, but they lack generality and require a lot of modification or redesign when deploying them to new scenes.

With the improvement of computer performance, scholars also propose many methods based on deep learning. Defect detection methods based on convolutional neural networks can be classified into supervised learning and unsupervised learning. Supervised learning is using labeled images for training [20]–[23], when the label information is not precise or incomplete, it is also called weakly supervised learning [24]–[26]. These methods can achieve high accuracy when the number of samples is sufficient. The main methods of unsupervised learning is to use auto encoder or GAN [27]–[29] to learn features from defect-free samples, then input a defect image and convert it into a defect-free image, finally make difference between the input and output to achieve defect detection. These unsupervised learning methods perform well on regular texture images, such as fabric, but can not handle the images in complex scenes. Park and Kwon [7] designed a simple CNN (Convolutional Neural Network) to classify the dirt, wear and other defects on the components, they used 3,000 images to train the network, and achieved 98% accuracy rate. Cha *et al.* [8] achieved crack detection based on the sliding window method. They designed two sliding window paths and input the image blocks into the CNN respectively. Finally, the defect image blocks were combined and restored. However, this method can only achieve rough crack location, and it takes a long time to scan the image twice. Chen *et al.* [9] applied CNN to defect detection of rail vehicle fasteners. They first used cascaded SSD [10] and YOLO [11] network to locate fasteners, and finally used classifier to classify the defect types. This system can achieve a high detection rate with good adaptation and robustness in complex environments. However, training the three networks requires a large number of samples and manual annotations, which also increases inference time. Tabernik and Šela [12] proposed a method based on image segmentation to detect metal surface defects, this method achieved 99% average precision with only a few defect samples, but its network is not designed as a multi-scale architecture, which causes some small area defects to be lost. Tao *et al.* [31] used two cascaded auto encoders to segment defects, and then used a compact CNN to achieve defect classification. However, in the case of small samples, too deep network will result in overfitting and increase the calculation time. In short, these methods based on deep learning have better robustness and can achieve higher accuracy, but they still do not solve the problems of large requirements of defect samples and real-time requirement.

As mentioned above, there are still challenges for the defect detection system:

1) The system based on deep learning requires a large number of defect samples, which is difficult to satisfy in many product lines. Some defects (such as scratch, white point, back point) are difficult to collect since there don't occur frequently. However, to maintain the performance of the system, enough samples (usually needs more than one hundred samples) are required when training the deep learning network.
2) In order to achieve high detection accuracy, the architecture of deep learning network is usually deep and complex, which limits the requirement of real-time detection. Simplify the network will greatly degrades the performance of the system. Therefore, how to balance the performance and the processing time is still a problem.

The motivation of this paper is to develop a defect detection system based on deep learning, which can solve the above difficulties. To do this, we propose a lightweight four-stage appearance defect detection model, as shown in FIGURE 1.

1) Preprocessing approach is applied to enhance the contrast of the input images. In order to make the defect features easier to be extracted by segmentation network, histogram stretch is first employed to enhance the contrast of the defect.
2) A mini U-Net [18] is constructed to predict the mask. In order to simplify the architecture and maintain its performance, dilated convolution is employed to expand the receptive field so that the number of the U-Net layers is reduced, while keeping the performance of the network.
3) The prediction mask is corrected and the candidate defect region is extracted. A rule based approach is utilized to eliminate the interference of mis-segmentation. As a result, fewer candidate defect regions are sent to the next decision network, which not only speeds up the processing, but also reduce the false alarm rate of the system.
4) The prediction mask of the mini U-Net and candidate defect region (generated by the third stage) are fed into the decision network to further verify the prediction results of segmentation network. The prediction mask contains the prior knowledge of the segmentation network, which provides a shortcut for the decision network, thus reduce the parameters, and makes it easier to train with small samples.

As described, the four stages are cascaded and interrelated. The first stage enhances the contrast of the defect region, which improves the performance of the segmentation network. The second stage is critical for the whole system. Since it generates the candidate defect regions, it is the guarantee of the detection accuracy of the system. Due to the usage of mini U-Net, the system can segment defect regions with fast speed. The third stage provides samples for decision-making and reduces irrelevant interference, resulting in a low false

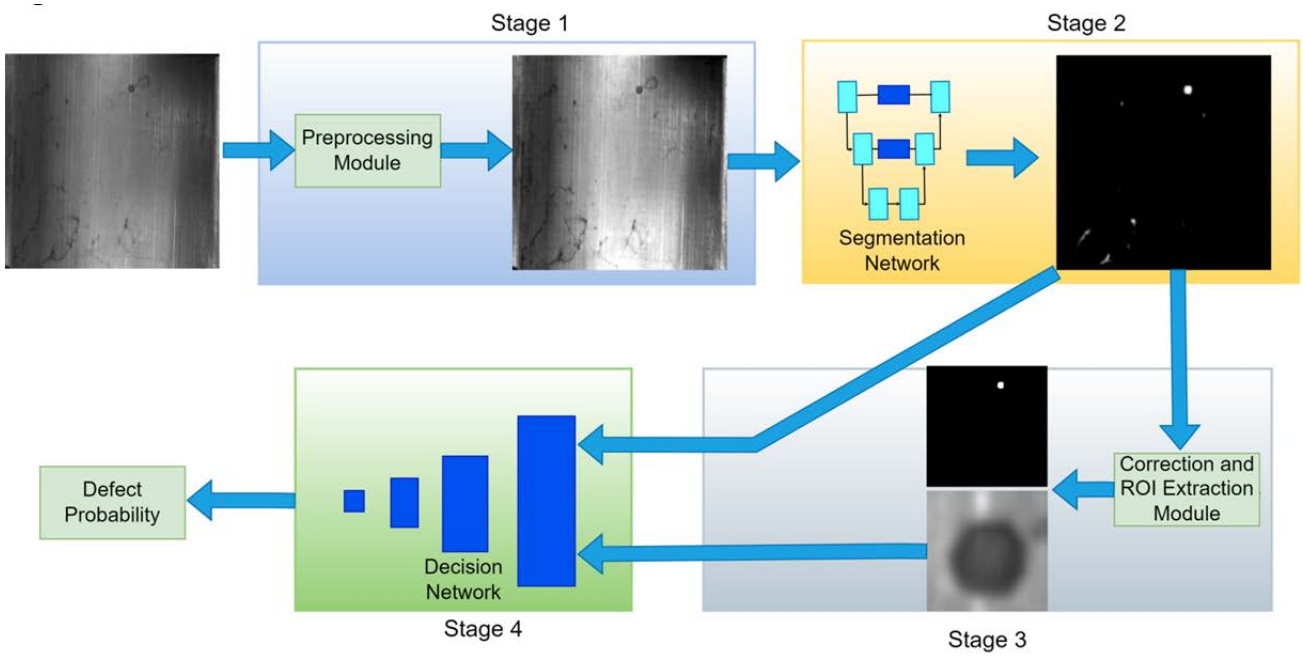

FIGURE 1. Overview of the proposed model. In the first stage, the contrast of the input image is enhanced and the data distribution is stretched to the same range. In the second stage, the proposed Mini U-Net is used to predict the mask. In the third stage, some irrelevant interference in the prediction mask is removed and candidate defect regions are output. In the fourth stage, the decision network combines the candidate region and prediction mask to output defect probability.

alarm and fast decision. Finally, the fourth stage gives the final judgment to make the detection results accurate and credible.

The remainder of the paper is structured as follows. In Section 2, we introduce the related works. Our proposed model is introduced in detail in Section 3. In section 4, the experimental results of this model are presented and compared with other methods. Finally, we make a summary and prospect in Section 5.

II. RELATED WORK

LeNet [14] is one of the earliest convolutional neural networks for image recognition. It extracted features through convolution, parameter sharing, pooling and other operations, and used full-connection layer for classification, which laid a foundation for other visual tasks.

In the field of image segmentation, inspired by FCN (Fully Convolutional Networks) [15], Shelhamer *et al.* designed an end-to-end model U-Net. It is composed of a group of symmetrical encoder and decoder, as shown in FIGURE 2 The encoder implements extracting the semantic features of the input image, and decoder performs reconstructs the high-level semantic features back to the original resolution. This pyramid structure with skip connection enables it to capture multi-scale semantic information, and it has achieved good performance in medical image segmentation. Yu and Koltun [16] proposed to use a series of cascaded dilated convolution for image segmentation. Dilated convolution can expand the receptive field without increasing parameters (see in FIGURE 3), and it can obtain more dense feature

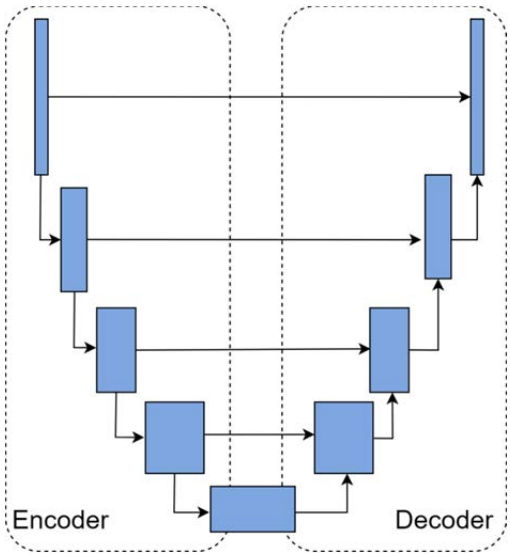

FIGURE 2. U-Net architecture. The encoder extracts the deep abstract features of the image through convolution and pooling, and then the decoder restores these abstract features to the original image size through linear interpolation or transposed convolution and outputs the prediction mask. Feature fusion is performed at each feature map scale, it is achieved by concatenating the encoder feature map to the decoder feature map on the channel dimension.

information compared with downsampling. In the case of limited computing resources and real-time requirements, the induction of dilated convolution is a good choice.

In order to improve the performance of CNN with increasing only a few parameters, Hu *et al.* [17] proposed

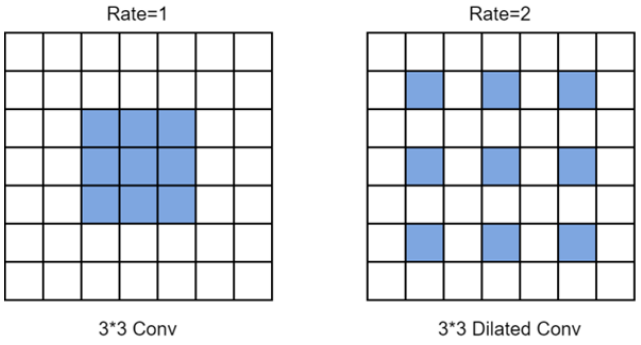

FIGURE 3. Dilated convolution. Because of the information redundancy of adjacent pixels, the dilated convolution can capture the long-term contextual semantic information without adding additional parameters. Dilation rate represents the sampling interval, when it is 1, which is regular convolution. Rate = 2 means that 1 (R-1) pixel between adjacent pixels do not involve the convolution operation.

a channel attention mechanism: SE Net, which squeezes redundant channels and excites important channels of the feature maps, SE Net-based model won the ImageNet Challenge [18] in 2017.

In order to solve the problem of excessive parameters and computation of convolutional neural network, Sifre [19] proposed the depthwise separable convolution, that is, each channel of the feature maps is convoluted separately, then a 1×1 convolution kernel is used to adjust the number of output channels.

III. PROPOSED SYSTEM

A. PREPROCESSING MODULE

Considering that the samples are collected under different illumination and the gray level distribution is in a small range, in order to enhance the contrast of the defects, histogram stretch is adopted to extend the gray level to the whole grayscale arrange. The steps of preprocessing module are show in Algorithm 1.

As shown in FIGURE 4, the grayscale of the input image is extended. As a result, the contrast of the defects are enhanced, as show in FIGURE 5.

B. SEGMENTATION NETWORK

Because of the complexity of medical images and high requirements for segmentation accuracy, the original U-Net was designed as a five layers encoder-decoder architecture. However, Zhou *et al.* [30] noted that increasing the number of layers of U-Net does not always improve the segmentation performance, but resulted in a significant increase in parameters. In fact, the semantic information of industrial images is relatively simple, under the deep encoder-decoder architecture, each layer may not produce sufficient feature differentiation, and multiple feature reuse would cause the redundancy of parameters. In addition, a few pixel-level errors in the segmentation network are acceptable because our system has next correction module and decision network.

Algorithm 1 Preprocessing Module

Input: Grayscale image $I(x, y)$; Probability thresholds: T_{Max} and T_{Min}.
Output: Preprocessed image $I'(x, y)$

1. Calculate the gray histogram of $I(x, y)$ and normalize it to [0, 1], denote as $H[i]$, where i $\in$ [0, 255];
2. Initialize the cumulative sums of probabilities: $S_1 = 0, S_2 = 0$;
3. **while** $((S_1 < T_{Min})$ or $(S_2 < T_{Max}))$ **do**
   for $j = 0 to 255$ **do**
   if $(S_1 < T_{Min})$ **then**
   $S_1 \leftarrow S_1 + H[j]$
   $min\_index \leftarrow j$
   if $(S_2 < T_{Max})$ **then**
   $S_2 \leftarrow S_2 + H[255 - j]$
   $max\_index \leftarrow 255 - j$
   end for
   end while
4. $I'(x, y) \leftarrow 255 * \frac{I(x,y) - min\_index}{max\_index}$
5. $I'(x, y) \leftarrow \begin{cases} 0, \& I'(x, y) < 0 \\ 255, I'(x, y) > 255 \end{cases}$
6. **return** $I'(x, y)$

Therefore, we propose a mini U-Net for segmentation, see in FIGURE 6.

The mini U-Net follows the design idea of U-Net, it consists of a group symmetrical encoder and decoder. Since we reduced the network layers, this leaded to the limited size of the receptive field, so we replaced the two convolution layers with dilated convolution. In addition, fewer downsampling layers can make small area defects less likely to be lost. For example, holes and dust defects account for a small proportion in the image, while cracks or fray defects account for a large proportion, proposed network can capture large area defects as completely as possible without losing small area defects.

We employ attention mechanism in skip connection. SE Net [17] is a channel attention mechanism, which can suppress the characteristic response of unrelated region and excite important channels with only a few parameters to achieve better segmentation performance. Its architecture is shown in FIGURE 7.

It is worth noting that Sigmoid activation function is added after the last convolution layer, which can make the segmentation results distribute in (0, 1).

We use cross entropy loss function to train the segmentation network:

$$\mathcal{L}_{seg} = -\frac{1}{n}\sum_{i=1}^{n}(w_1 * y_i ln x_i + w_2 * (1 - y_i) ln(1 - x_i)) \quad (1)$$

where, n is the total number of pixels in a mini-batch, y_i is the pixel of the label mask, and x_i is the pixel of the prediction mask. w_1 and w_2 are the weights of positive samples (defects) and negative samples (background) respectively, because

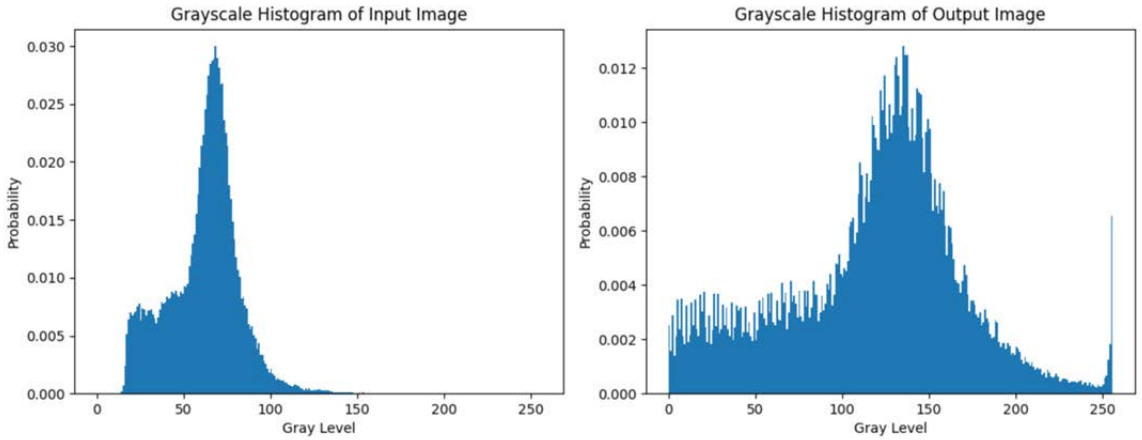

FIGURE 4. Grayscale histogram.

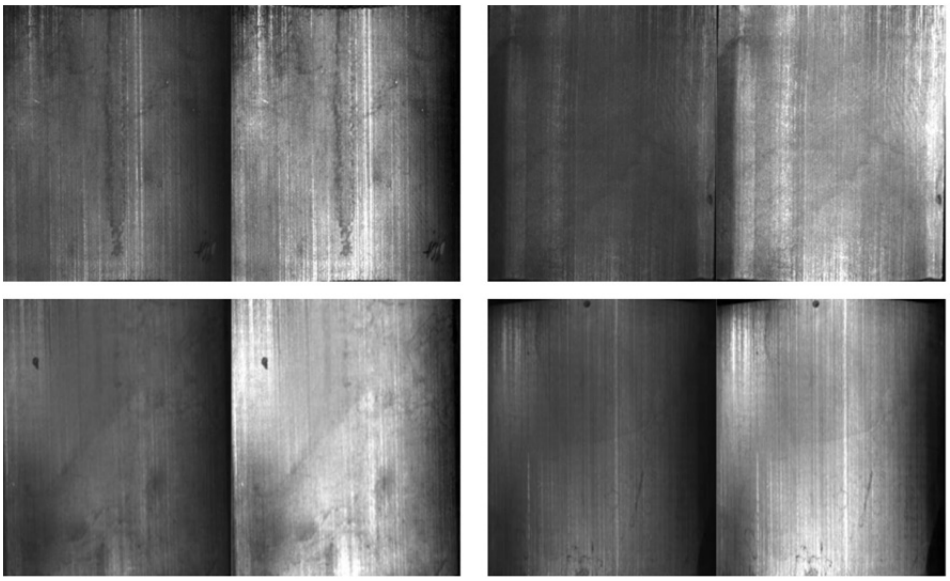

FIGURE 5. Original(left) and preprocessed images(right).

most areas in the image are background while defect areas account for a small proportion, we set w_1 and w_2 to 0.8 and 0.2 respectively.

C. CORRECTION AND ROI EXTRACTING MODULE

Affected by noises, dirt and illumination, there are some misclassified pixels in the prediction mask. The purpose of this module is to correct the prediction mask to ensure that candidate defect regions can be correctly extracted. This module includes four steps: (1) Binarization; (2) Morphological operation; (3) Filtering out small area regions. (4) Extracting ROI (Regions of Interest).

IV. BINARIZATION

The purpose of binarization is to filter out some pixels with low defect probability, which are the darker pixels in the mask (see in FIGURE 8). Given a threshold T, the calculation formula of binarization is:

$$Mask_{binary}(x, y) = \begin{cases} 0, & Mask(x, y) < T \\ 255, & Mask(x, y) \geq T \end{cases} \quad (2)$$

V. MORPHOLOGICAL OPERATION

We implement close operation with a 7×7 kernel for the result of previous step. Sometimes the cracks in the image are not continuous, which will lead to the increase of candidate defect regions, so we use close operation to connect near pixels, as shown in FIGURE 9.

VI. FILTERING OUT SMALL AREA REGIONS

In some scenarios, small defects such as hair, dirt and dust layers on the product surface are acceptable. Label masks do not mark the above conditions, but they have similar

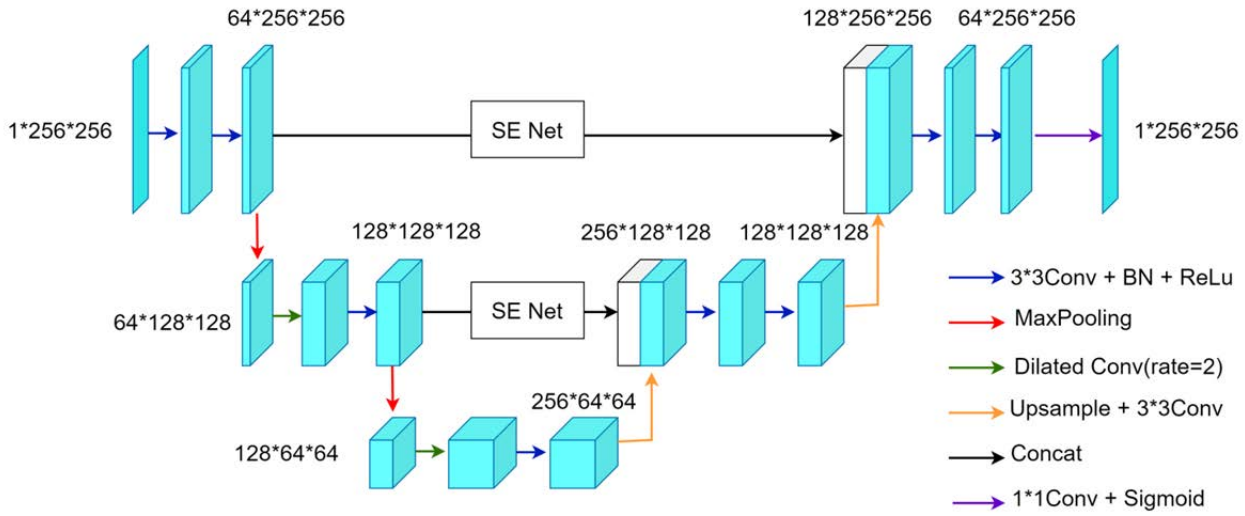

FIGURE 6. Mini U-Net architecture. On the basis of U-Net, the network depth is reduced, and the dilation convolution and SE Net are employed to improve segmentation performance.

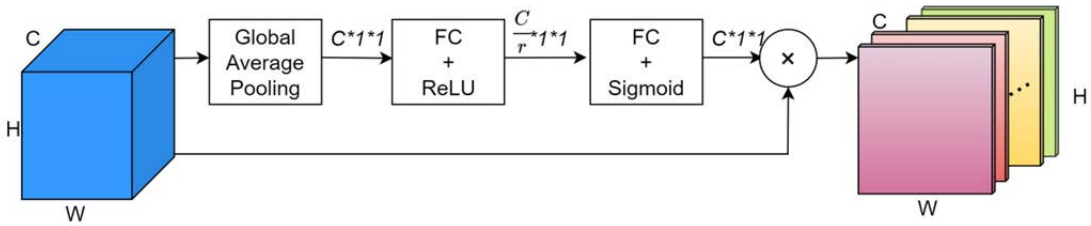

FIGURE 7. SE Net. A channel attention mechanism in which important channel features are excited and irrelevant channel features are suppressed.

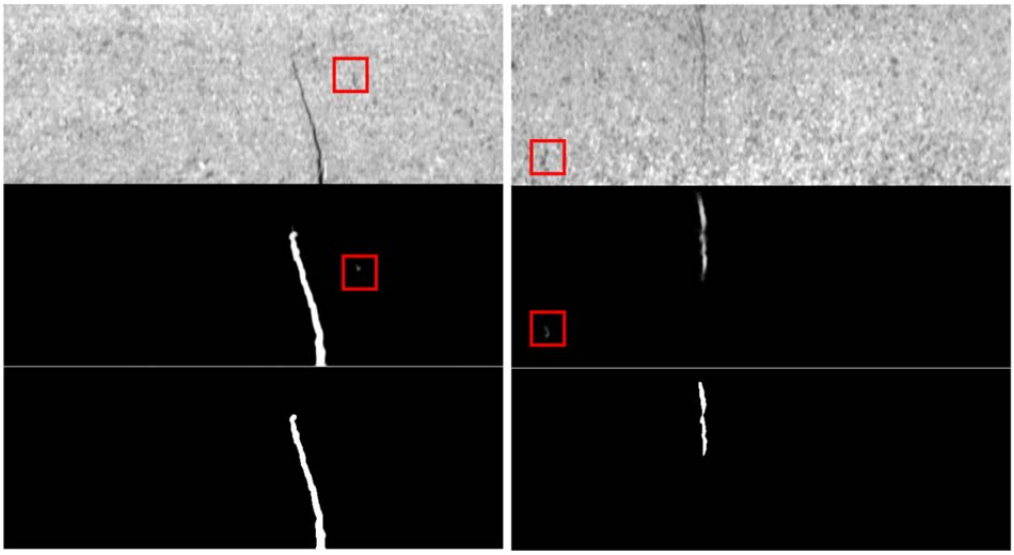

FIGURE 8. Binaryzation. The pixels in the prediction mask are set to 0 or 255 to achieve filter interference and highlight the foreground.

characteristics to defects, so the segmentation network produces uncertain output, which is reflected in the segmentation response is not strong enough and is intermittent, as shown in FIGURE 10, the soil causes mis-segmentation. By the filtering of binarization module, their area is much smaller than the real defects, filtering these small area regions can

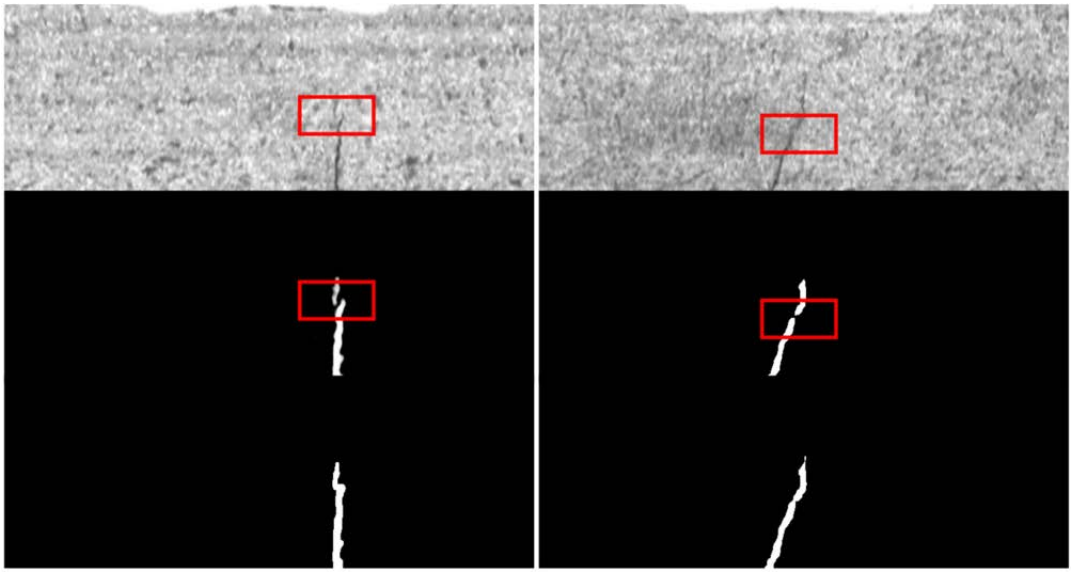

FIGURE 9. Morphological operation. Due to the discontinuity of the crack, the close operation is used to connect near pixels to reduce the generation of candidate defect regions.

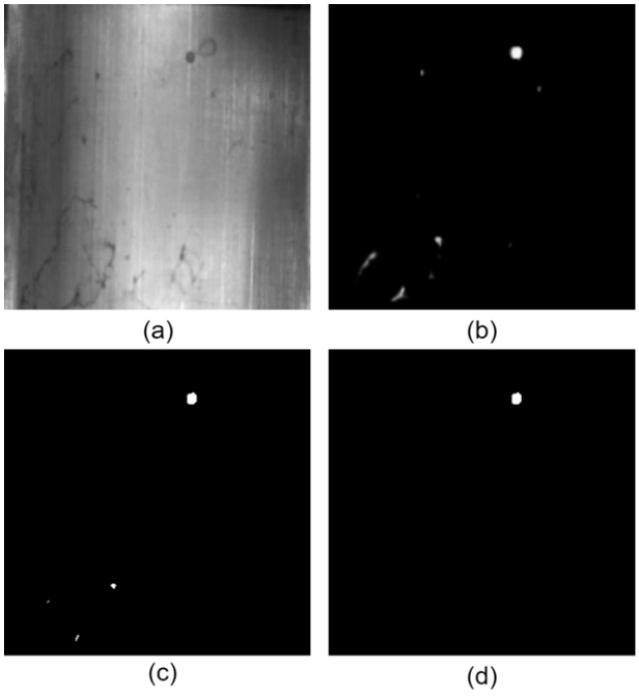

FIGURE 10. Filtering out small area regions. (a)Input images; (b)prediction mask; (c) Binary image; (d) Filtering out small area regions.

accelerate the decision-making speed and reduces false alarm. Users can define the area threshold according to the practical application scenarios (considering that in some special cases the defect area below a certain threshold is also acceptable). In the experiments, we counted the defect area of all masks in the training set and the threshold is set to 80% of the minimum.

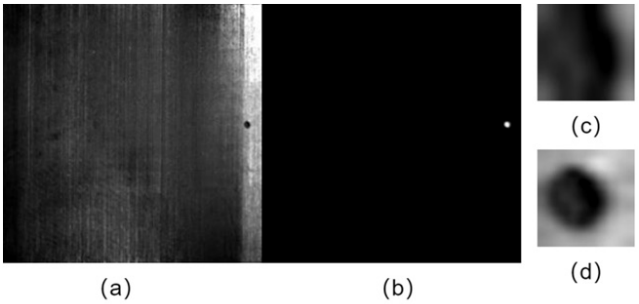

FIGURE 11. Rectangle box extension. (a)Input image; (b) prediction mask; (c) ROI (without box extension); (d) ROI (with box extension).

VII. EXTRACTING REGIONS OF INTEREST

We extract the candidate defect regions by calculating the minimum enclosing rectangle and then enlarge its size. This step ensures that rich semantic information of the candidate regions is captured. In some cases, due to the pixel-level errors of the segmentation network, the extracted ROI may be incomplete or dominated by black pixels (see in FIGURE 11), which is not conducive to the decision network learning features.

A. DECISION NETWORK

As described in the previous section, segmentation network may respond to non-product quality phenomena, and the candidate defect regions output by the correction module are difficult to distinguish true positive or false positive for the segmentation network. For some defect-free images, there are some interferences (for example, for detecting crack defects, the image possibly includes dirt, fingerprints, etc.), they are similar to defects. As show FIGURE 12, due to uneven

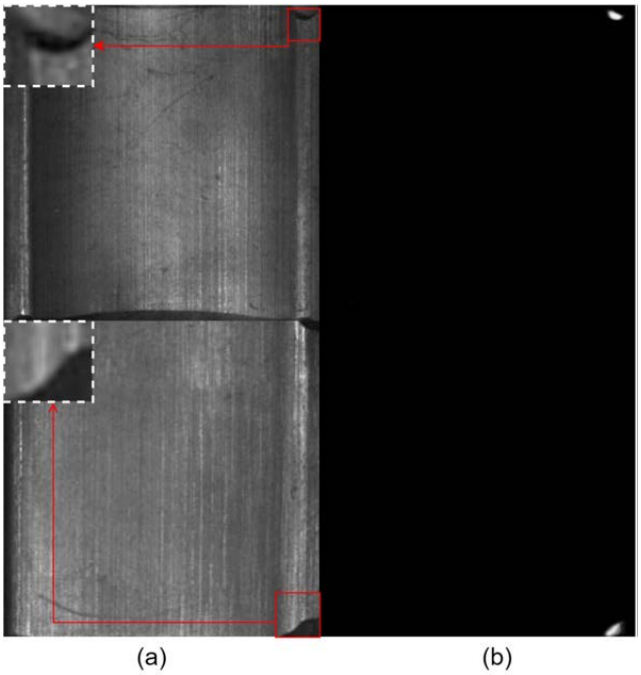

FIGURE 12. Mis-segmentation. (a) defect-free images; (b) prediction masks.

lighting, the corner region of the image produces similar defects representation.

Therefore, we design a decision network to determine the true defects and remove those false alarms. The network has two inputs: the prediction mask and ROI (candidate defect regions). In the experiments, the input regions are all resized to 100 × 100 pixels. Due to the fact that the mask contains pixel-wise segmentation information of the defect region, and the ROI includes the defect region and its background (giving global information of the defect), the two inputs of the network provide rich information for the next decision.

The network is a compact encoder-decoder architecture, as shown in FIGURE 13. The encoder extracts features by increasing dimension, then the decoder reduces the dimension and projects it to a probability map. In order to accelerate the inference speed, most convolution operations are depth-wise separable convolution (DSC). When the feature map is downsampled to 12 × 12, every probability value's receptive field is a 38 × 38 patch in the input images. Sigmoid activation function is used to output the defect probability of each patch. Finally, global average pooling is used to output the final defect probability. We do not use the full connection layer to classify the candidate regions, because it forms a scalar output from the images, which may cause the classification results dominated by mask (the network only focuses on whether there is segmentation information in the mask). Since not every mask patch can provide segmentation information, the proposed decision network combines mask information and ROI features to output a probability map, ensuring output a high confidence score.

We use mean-squared error loss function to train the decision network:

$$\mathcal{L}_{dec} = \frac{1}{n}\sum_{i=1}^{n}(y_i - x_i)^2 \tag{3}$$

where, n is the number of samples in a mini-batch, y_i is One-Hot encoding label (0 is defect-free sample, 1 is defect sample), x_i is the defect probability predicted by the decision network.

VIII. EXPERIMENTS

Our experiments are based on three public datasets: Magnetic Tile Defect dataset [32], Kolektor Surface Defect Detection dataset [12] and DAGM2007 dataset [33]. All of our experiments were based on a computer powered by Intel Xeon Gold 5220 and NVIDIA Tesla T4, the operating system is Ubuntu 18.04. We used python3.7 coding language and PyTorch1.7 deep learning framework to implement our experiments.

For each dataset, we only used a few dozen defect images for training, and the simplest data augmentation method was used: random horizontal and vertical flip. If there are more than one defect in the image, the correction module will output multiple candidate regions, and every candidate region is fed to the decision network for classification respectively.

A. TRAINING SEGMENTATION NETWORK

In the experiments of these three datasets, the segmentation network used Adam [34] optimizer ($\beta_1 = 0.5$, $\beta_2 = 0.999$) to train 100 epochs, batch size is 2, and initial learning rate is 0.001. The following learning rate decay strategy is used:

$$lr = base\_lr * (1 - \frac{current\_epoch}{max\_epoch + 1})^{0.9} \tag{4}$$

Due to the imbalance between the number of defect images and defect-free images in these datasets, we followed the alternate training strategy of [12], that is, in each epoch, all defect images were traversed once, while the same number of defect-free images were randomly selected for training.

B. TRAINING DECISION NETWORK

Training decision network needs to freeze the weights of segmentation network. However, for most defect-free images, the defect region cannot be extracted. In order to make the decision network learn the background features, the ROI is replaced by a randomly cropped 100 × 100 region in the pre-processed image. The decision network uses Adam optimizer ($\beta_1 = 0.5$, $\beta_2 = 0.999$) to train 50 epochs, with a learning rate of 0.001, and also adopts the strategy of alternate training between defect images and defect-free images.

C. TESTING

When testing the model, the decision threshold is set to 0.5. In addition, in order to speed up the inference time, if no defect region can be extracted, the decision network will be skipped and directly be classified as defect-free samples.

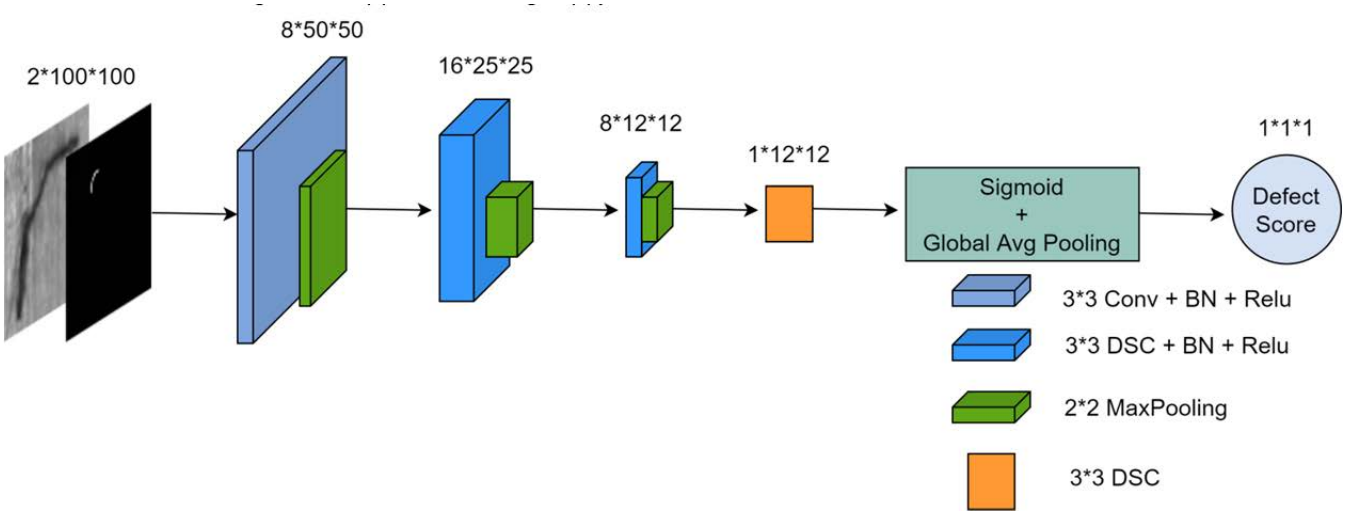

FIGURE 13. Decision network. The prediction mask and ROI are jointly input into the network, and a patch-based decision-making method is used to output the final defect probability.

We also performed ablation experiments on Kolektor Surface Defect Detection dataset and Magnetic Tile Defect dataset to explore the contribution of preprocessing module, correction modeule and decision network respectively.

When the decision network is not enabled, we used a logistic regression model to replace it:

$$y = Sigmoid(WX^T + b) \tag{5}$$

where, X is a two-dimensional tensor, which consists of the global max pooling value and average pooling value of the prediction mask.

Since the function of correction module is to output candidate defect regions, and the logistic regression model only requires the prediction mask as input. Therefore, when the correction module is enabled separately, only the mask postprocessing steps of the correction module is enabled and the postprocessed masks are input into the logistic regression model.

TPR (True Positive Rate) and TNR (True Negative Rate) were used to evaluate the classification performance on positive and negative samples respectively:

$$TPR = \frac{TP}{TP + FN} \tag{6}$$

$$TNR = \frac{TN}{TN + FP} \tag{7}$$

where, TP and TN represent the number of correctly classified as positive and negative samples respectively. In contrast, FP and FN are the number of misclassified as positive and negative samples.

D. INTRODUCTION TO THE DATASETS

Magnetic tile defect dataset is a real world dataset collected under different illumination by the Institute of Automation, Chinese Academy of Sciences. We used blowhole (115 samples), crack (57 samples) and free (1,042 samples) images (see in FIGURE 14) for experiments. We evaluated the model with three-fold cross validation. In each experimental group, we ensured that there were 30 blowhole and 30 crack images for training fold, and the rest defect images were used for testing fold. For the defect-free samples, they were evenly divided into three folds and keep one fold for testing in each experimental group. Since most defects are represented as darker regions and only accounts for a small portion in this dataset. So, the threshold parameter T_{Min} in the preprocessing module is set to a small value 0.002, and T_{Max} is set to a higher value 0.01 to expand the gray level of brighter regions, so that foreground and background can make a difference. Because the segmentation network outputs strong results for true defect regions, the binarization threshold is set to 150, which generates best experimental results. All images and their masks were resized to 256×256.

Kolektor surface defect dataset (KolektorSDD) contains 50 electrical commutators surface images (see in FIGURE 15). Each commutator was collected 7 to 8 non-overlapping surface images, and at least one was defective. This resulted in 399 images in this dataset, of which 52 were defective. We still used three-fold cross validation. In each experimental group, we ensured the training fold had 30 commutators' images, the remaining 20 were for the testing fold. Parameters in the experiment were: $T_{Min} = 0.0001$, $T_{Max} = 0.01$ and $T = 150$. All images and their masks were resized to 704×256.

DAGM2007 is a surface defect image dataset with 10 texture class (see in FIGURE 16). The defect samples and non-defect samples are unbalanced (In class1-6, there are 150 defective images and 1,000 non-defective images; in class 7-10, there are 300 defective images and 2,000 non-defective images), and only weak labels are given, that is, the label is marked in ellipse form, it indicates the defect area roughly. Since each class is already divided into a training set and a testing set, we did not use K-fold

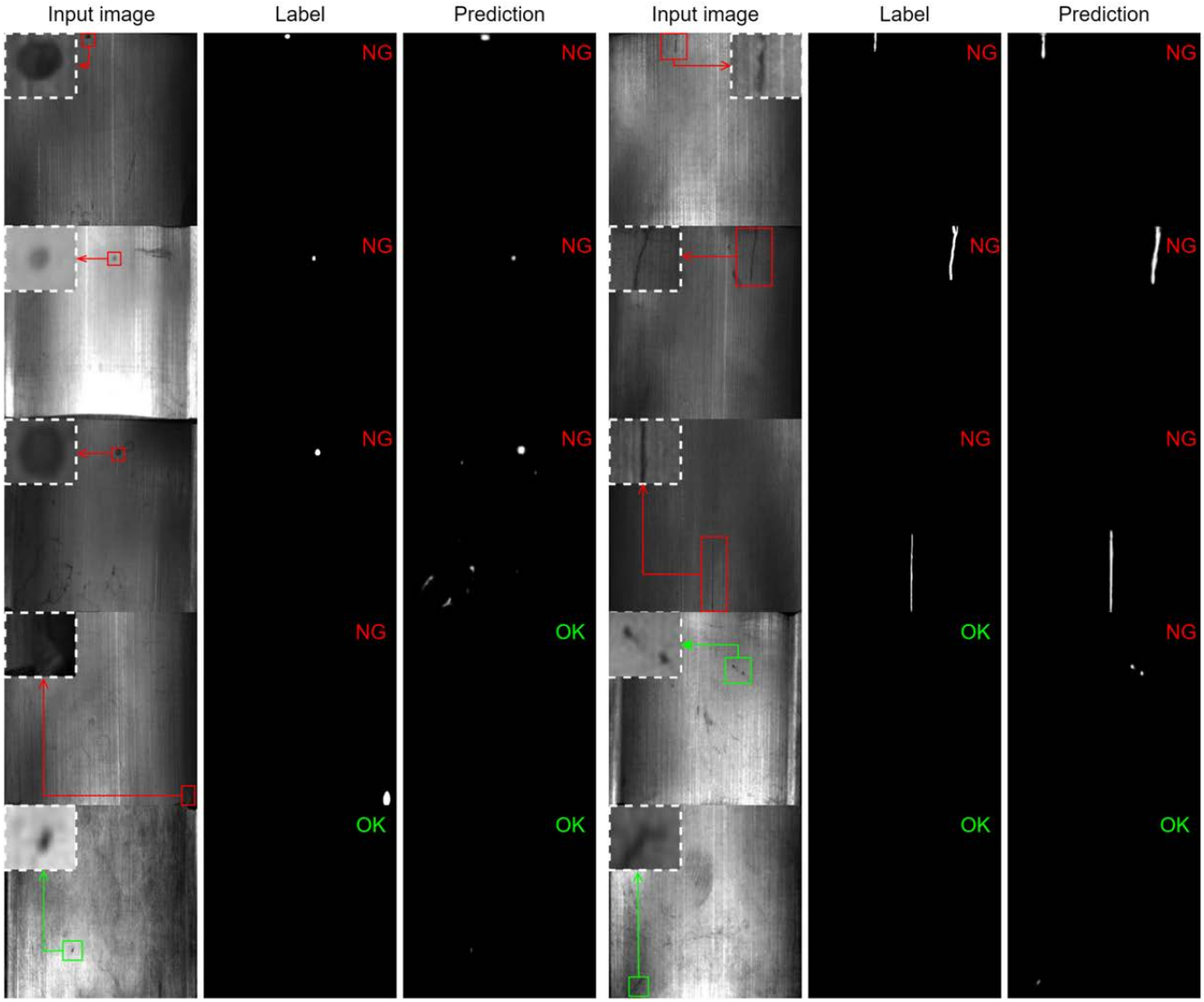

FIGURE 14. Examples on magnetic tile dataset.

cross validation and enabled all modules. The images size of each sub-dataset is 512 × 512, the binarization threshold T is 150, and other parameters are shown in the Table 3.

E. EXPERIMENTAL RESULTS AND ANALYSIS

Proposed model performance on the magnetic tile dataset is reported in Table 1. The baseline model does not enable all three modules, we can see that the preprocessing module can significantly improve the TPR, but also enhance some noises and dirt presentation, which leads to a slight drop in TNR. In industry field, it is acceptable to increase some false alarm rate in exchange for a lower missed alarm rate. The separate enabling of correction module reduces the accuracy a little, because this module is designed to output candidate defect regions, postprocessed masks can not improve the logistic regression model performance. However, when the correction and decision network are combined and enabled, model achieves the highest TNR, which indicates that correction module and decision network are correlated and can remove many false positive samples. The contribution of decision network is reflected in the improvement of TPR, because logistic regression model only uses prediction mask to classify, which is very susceptible to mis-segmentation. Overall, the number of misclassified samples (FP+FN) is the least when all three modules are enabled.

Table 2 shows model performance on KolecktorSDD. We achieved almost 100% TPR without enabling any modules, this is because the segmentation network outputs good results for positive samples. In most experimental groups, enabling any of the three modules results in a slight improvement in TNR compares to the baseline model. On the whole, when all three modules are enabled, there are only a few false positive samples and all defective samples are found by our

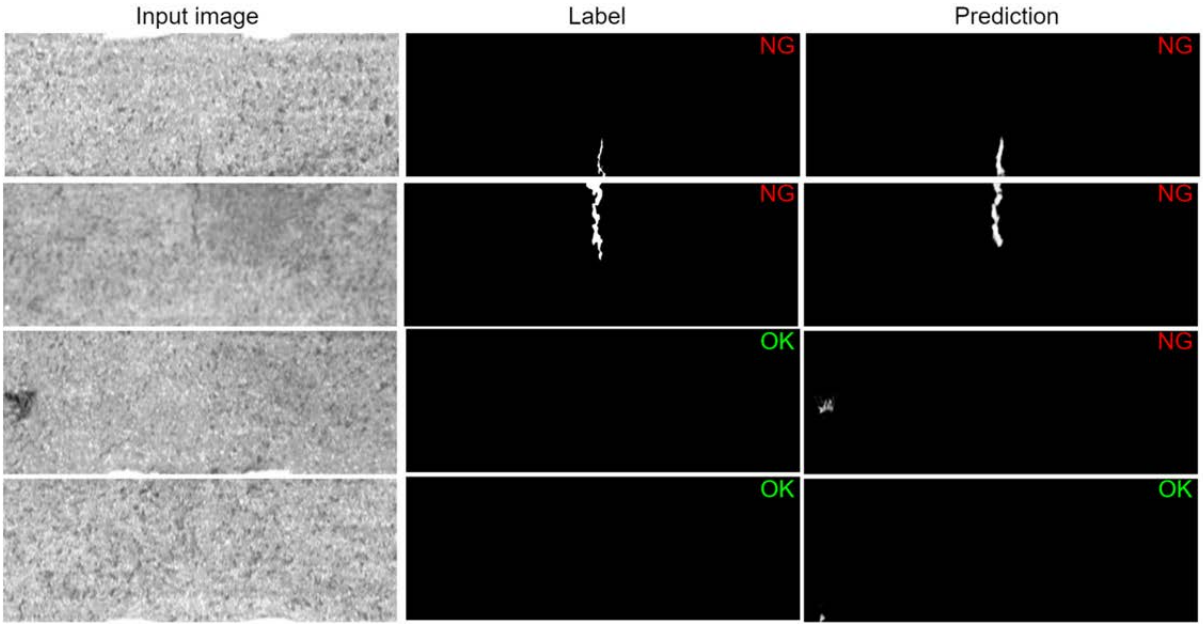

FIGURE 15. Examples on kolektor surface defect dataset.

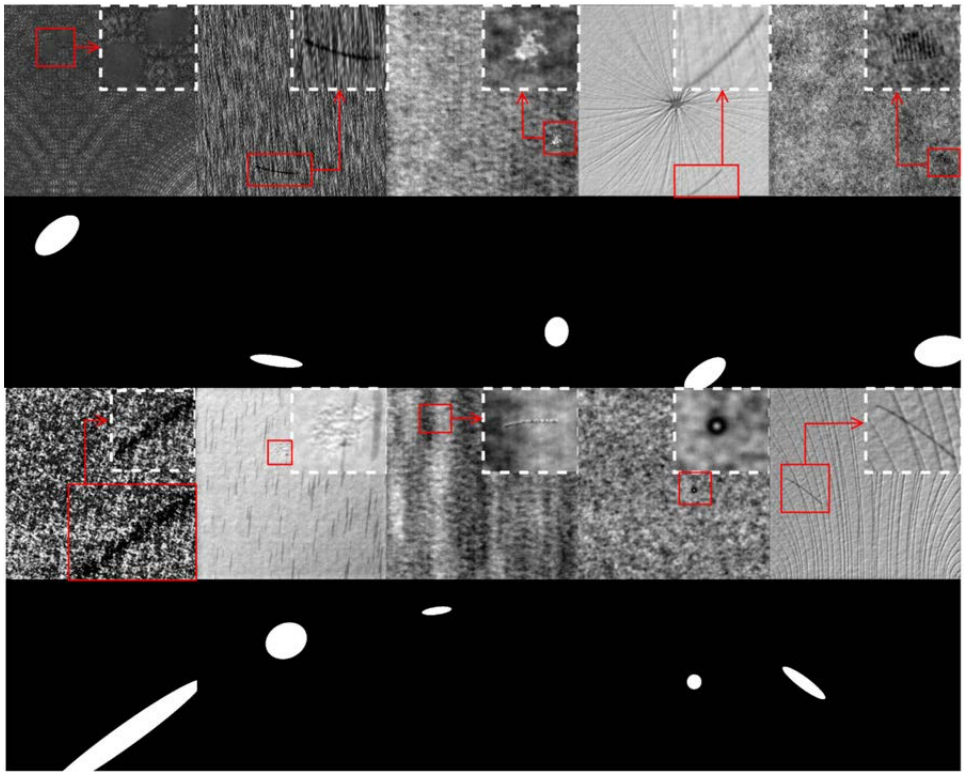

FIGURE 16. DAGM2007 dataset.

model, which meets the accuracy requirements of industrial application.

In the DAGM2007 dataset, our model achieves 100% classification accuracy (see in Table 3). This indicates that detecting defects from regular texture images is a simple task for proposed model, it performs well even in the case of weak labels, which can significantly reduce the time of data annotation in practical applications. In addition, the saturated

TABLE 1. Ablation Experiment on Magnetic Tile Defect Dataset.

Experiment group index	TPR	TNR	FP+FN	Preprocessing	Correction	Decision Network
1	93.75%	93.96%	21+7			
	98.21%	93.10%	24+2	√		
	91.02%	95.68%	15+10		√	
	96.42%	90.22%	34+4			√
	86.60%	**97.70%**	15+8		√	√
	98.21%	95.40%	**16+2**	√	√	√
2	94.64%	92.79%	25+6			
	98.21%	89.04%	38+2	√		
	94.64%	89.91%	35+6		√	
	95.53%	72.33%	96+5			√
	91.96%	**95.96%**	**14+9**		√	√
	98.21%	93.94%	**21+2**	√	√	√
3	79.46%	96.54%	12+23			
	92.85%	92.85%	20+8	√		
	79.46%	95.96%	14+23		√	
	89.28%	87.89%	42+12			√
	76.78%	**98.27%**	6+26		√	√
	92.85%	95.67%	**15+8**	√	√	√

TABLE 2. Ablation Experiment on KolektorSDD.

Experiment group index	TPR	TNR	FP+FN	Preprocessing	Correction	Decision Network
1	100%	97.32%	3+0			
	100%	**100%**	**0+0**	√		
	100%	98.55%	2+0		√	
	95.45%	98.55%	2+3			√
	100%	96.37%	5+0		√	√
	100%	**100%**	**0+0**	√	√	√
2	100%	83.57%	23+0			
	100%	95.71%	6+0	√		
	100%	90.71%	13+0		√	
	100%	87.85%	17+0			√
	100%	96.42%	5+0		√	√
	100%	**97.14%**	**4+0**	√	√	√
3	100%	95.71%	6+0			
	100%	97.14%	4+0	√		
	100%	97.14%	4+0		√	
	100%	97.85%	3+0			√
	100%	97.85%	3+0		√	√
	100%	**99.28%**	**1+0**	√	√	√

sample quantity is also an important reason why proposed model can achieve such good performance.

To explore whether proposed model solves the challenge of sample scarcity, we conducted defect samples sensitivity experiments on magnetic tile dataset and KolecktorSDD. We used 10, 20 and 30 defect samples to train the model respectively, and kept the same dataset division as the previous experiments to evaluate the model.

As can be seen from Table 4, even if only 10 defect samples are used for training, proposed model can achieve more than 0.7 accuracy. When the number of defect samples was increased to 30, the accuracy rate was above 0.9 in both datasets. The experimental results show that proposed model overcomes the challenge of sample scarcity to some extent. We deem the following factors are the reasons why proposed model still works with small samples:

1) The preprocessing module can improve the generalization of the model in small samples case. Because preprocessing module keeps the information (gray variation) which contain diagnostic value, and eliminates the problem of inconsistent illumination. This makes it less susceptible to extreme sample interference during training and testing. In addition, the same data distribution can make the model converge faster under batch training.

TABLE 3. Performance on DAGM2007.

Texture class	TPR	TNR	FP+FN	T_{Min}	T_{Max}
1	100%	100%	0+0	0.002	0.01
2	100%	100%	0+0	0.002	0.1
3	100%	100%	0+0	0.01	0.001
4	100%	100%	0+0	0.002	0.01
5	100%	100%	0+0	0.002	0.01
6	100%	100%	0+0	0.02	0.1
7	100%	100%	0+0	0.01	0.0001
8	100%	100%	0+0	0.02	0.02
9	100%	100%	0+0	0.002	0.1
10	100%	100%	0+0	0.002	0.1

TABLE 4. Sensitivity of the model to the defect sample Quantity.

Dataset_index	Tile_1		Tile_2		Tile_3		KSDD_1		KSDD_2		KSDD_3	
Defect samples	TPR	TNR	TPR	TNR	TPR	TNR	TPR	TNR	TPR	TNR	TPR	TNR
10	76.97%	90.81%	87.09%	85.52%	84.21%	86.16%	93.93%	84.05%	96.67%	95.58%	**100%**	83.58%
20	96.21%	**96.55%**	97.14%	91.93%	90.76%	**96.82%**	**100%**	89.13%	**100%**	96.94%	**100%**	88.57%
30	**98.21%**	95.40%	**98.21%**	**93.94%**	**92.85%**	95.67%	**100%**	**100%**	**100%**	**97.14%**	**100%**	**99.28%**

2) Proposed mini U-Net is suitable for small sample situation. Due to the pruning of the original U-Net, mini U-Net avoids the over-fitting problem of deep neural network in small samples. In addition, the usage of SE Net and dilated convolution also ensures the segmentation performance.
3) Correction module can provide more samples for decision network. In the segmentation task, an image is a sample, while the correction module may output multiple candidate regions from an image, which enriches the samples of the decision network. Because correction module filters out some irrelevant interference, the remaining candidate regions are difficult to distinguish true positive and false positive for the segmentation network. Furthermore, the decision network focuses on the classification of these difficult samples, which also improve the mode performance with small samples.

F. COMPARISON WITH THE STATE-OF-ART MODELS

We compared the proposed model with [12] and [31], both of them are segmentation-based defect detection methods. In addition, the model performance after replacing the mini U-Net with the original U-Net is also reported. We conducted experiments under the same dataset division and reported per image average inference time consumption and F1-score. Accuracy is too sensitive to the number of samples, while F1-score is a metric that combines precision and recall, which can fully demonstrate the model performance when positive and negative samples are unbalanced.

$$F1 - score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{8}$$

where,

$$Precision = \frac{TP}{TP + FP} \tag{9}$$

$$Recall = \frac{TP}{TP + FN} \tag{10}$$

Experimental results (see in Table 5) show that although our inference time is not the shortest, the F1-score is much higher than other methods. Due to Domen's segmentation network contains only convolution and downsampling layers, the predication mask is only 1/8 the size of input image, which shortens the inference time, but results in poor segmentation performance for small area defects. As shown in Table 5, Domen's F1-score is significantly lower than ours on magnetic tile dataset. Tao's method uses two cascaded autoencoders to enhance segmentation results, and they have achieved good performance with 3000 training samples. But it does not work in small sample situation, too many network parameters degrade the performance of the model. Our segmentation network keeps the same size as the input, and performs feature fusion at multiscale, which greatly ensures the segmentation performance. After replacing mini U-Net with original U-Net, the inference time is increased and the F1-score decreases a little except for the DAGM2007 dataset. The experimental results once again confirm the conclusion of [30]: deeper U-Net is not always performs well, the number of layers should depend on the difficulty of your dataset.

On the real-time issue, the proposed model won the second place. We deem that real-time should be based on excellent classification performance, otherwise faster processing speed is meaningless. Overall, the proposed model achieves the highest detection rate with acceptable processing time. It is a balance between classification performance and real-time requirement.

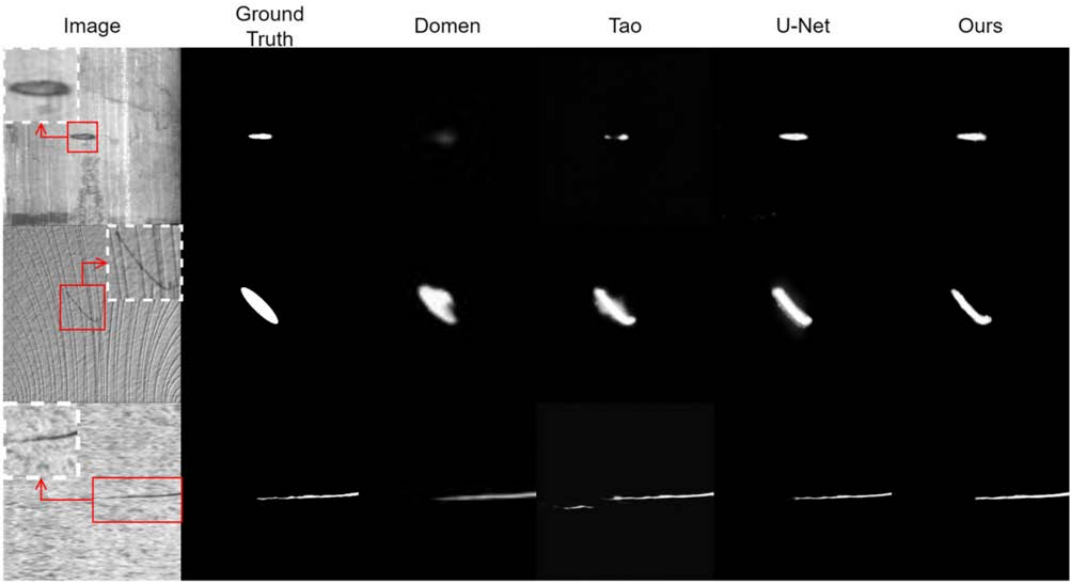

FIGURE 17. Comparison of segmentation results.

TABLE 5. Quantitative Evaluation.

Dataset	Kolektor SDD (704×256)		Magnetic Tile (256×256)		DAGM2007 (512×512)	
Method	F1-score	Inference time	F1-score	Inference time	F1-score	Inference time
Domen	0.94	**37ms**	0.72	**13ms**	0.99	**52ms**
Tao	0.67	91ms	0.56	32ms	0.69	134ms
U-Net	0.91	87ms	0.88	30ms	**1.00**	94ms
Ours	**0.96**	65ms	**0.91**	24ms	**1.00**	82ms

Since our model is applied in the industrial field, the pixel-level error of the segmentation network is acceptable, because the mask is visualized for inspectors in practical applications, and the classification results of the decision network are the most important. So we do not quantitatively evaluate the segmentation network. However, proposed mini U-Net still achieves good segmentation results, as shown in the FIGURE 17.

IX. CONCLUSION

In this paper, we proposed a four-stage product appearance defect detection model: the first stage implements contrast enhancement, following by the second stage, which is a segmentation task, the third stage performs correction and ROI extraction, and the final stage implements decision making. We have conducted experiments on Magnetic Tile dataset, KolektorSDD and DAGM2007 dataset. The proposed model fully solved the challenge of DAGM2007 and achieved more than 0.9 F1-score on Magnetic Tile dataset and KolektorSDD. We also tested the sensitivity of the model to defect sample quantity, the experimental results show that the proposed model can still work in small samples. Comparing with similar methods, our model can achieve best classification performance with a relatively short time consuming. To sum up, the proposed model solves the challenges mentioned in the introduction section, and we believe this model can provide powerful help in the quality control of industrial production.

In the future work, we will improve the system architecture to reduce the processing time, and make the system automatically search for the optimal parameters for better robustness.

ACKNOWLEDGMENT

(Xiang Xie and Rongfeng Zhang are co-first authors.)

REFERENCES

[1] B. R. Suresh, R. A. Fundakowski, T. S. Levitt, and J. E. Overland, "A real-time automated visual inspection system for hot steel slabs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-5, no. 6, pp. 563–572, Nov. 1983.

[2] R. C. Gonzalez and R. E. Woods, "Sharpening spatial filters," in *Digital Image Processing*, 3rd ed. Beijing, China: PHEI, 2010, pp. 179–187.

[3] X. Wang and W. X. Gao, "Application of PCA in defect detection based on parallel computing," *Comput. Eng. Des.*, vol. 37, no. 10, pp. 2810–2815, Oct. 2016.

[4] G. Liu and X. Zheng, "Fabric defect detection based on information entropy and frequency domain saliency," *Vis. Comput.*, vol. 37, no. 3, pp. 515–528, Mar. 2021.

[5] X. W. Zhang, F. Gong, and L. Z. Xu, "Inspection of surface defects in copper strip using multivariate statistical approach and SVM," *Int. J. Comput. Appl. Technol.*, vol. 43, no. 1, pp. 44–50, Mar. 2012.

[6] S. Radovan and G. D. Papadopoulo, "Vision system for finished fabric inspection," *Proc. SPIE*, vol. 4664, pp. 97–103, 2002.

[7] J.-K. Park, B.-K. Kwon, J.-H. Park, and D.-J. Kang, "Machine learning-based imaging system for surface defect inspection," *Int. J. Precision Eng. Manuf.-Green Technol.*, vol. 3, no. 3, pp. 303–310, Jun. 2016.

[8] Y.-J. Cha, W. Choi, and O. Büyüköztürk, "Deep learning-based crack damage detection using convolutional neural networks," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 32, no. 5, pp. 361–378, Mar. 2017.

[9] J. Chen, Z. Liu, H. Wang, A. Nunez, and Z. Han, "Automatic defect detection of fasteners on the catenary support device using deep convolutional neural network," *IEEE Trans. Instrum. Meas.*, vol. 67, no. 2, pp. 257–269, Feb. 2018.

[10] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 21–37.
[11] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 779–788.
[12] D. Tabernik, S. Šela, J. Skvarč, and D. Skocaj, "Segmentation-based deep-learning approach for surface-defect detection," *J. Intell. Manuf.*, vol. 31, no. 3, pp. 759–776, Jun. 2019.
[13] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent. (MICCAI)*, 2015, pp. 234–241.
[14] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
[15] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, Apr. 2017.
[16] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *Proc. Int. Conf. Learn. Represent.*, Nov. 2015, pp. 1–13.
[17] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-excitation networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 8, pp. 2011–2023, Aug. 2020.
[18] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Apr. 2015.
[19] L. Sifre, "Rigid-motion scattering for image classification," Ph.D. thesis, Ecole Polytechn., Paris, France, 2014.
[20] Z. Yu, X. Wu, and X. Gu, "Fully convolutional networks for surface defect inspection in industrial environment," in *Proc. Int. Conf. Comput. Vis. Syst.*, vol. 10528, Jul. 2017, pp. 417–426.
[21] Y. Ren, J. Huang, Z. Hong, W. Lu, J. Yin, L. Zou, and X. Shen, "Image-based concrete crack detection in tunnels using deep fully convolutional networks," *Construct. Building Mater.*, vol. 234, Feb. 2020, Art. no. 117367.
[22] M. Garg and G. Dhiman, "Deep convolution neural network approach for defect inspection of textured surfaces," *J. Inst. Electron. Comput.*, vol. 2, no. 1, pp. 28–38, 2020.
[23] Y. He, K. Song, Q. Meng, and Y. Yan, "An end-to-end steel surface defect detection approach via fusing multiple hierarchical features," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 4, pp. 1493–1504, Apr. 2020.
[24] N. Enshaei, S. Ahmad, and F. Naderkhani, "Automated detection of textured-surface defects using UNet-based semantic segmentation network," in *Proc. IEEE Int. Conf. Prognostics Health Manage. (ICPHM)*, Jun. 2020, pp. 1–5.
[25] D. Racki, D. Tomazevic, and D. Skocaj, "A compact convolutional neural network for textured surface anomaly detection," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2018, pp. 1331–1339.
[26] S. Marino, P. Beauseroy, and A. Smolarz, "Weakly-supervised learning approach for potato defects segmentation," *Eng. Appl. Artif. Intell.*, vol. 85, pp. 337–346, Oct. 2019.
[27] K. Liu, A. Li, X. Wen, H. Chen, and P. Yang, "Steel surface defect detection using GAN and one-class classifier," in *Proc. 25th Int. Conf. Autom. Comput. (ICAC)*, Sep. 2019, pp. 1–6.
[28] M. Haselmann, D. P. Gruber, and P. Tabatabai, "Anomaly detection using deep learning based image completion," in *Proc. 17th IEEE Int. Conf. Mach. Learn. Appl. (ICMLA)*, Dec. 2018, pp. 1237–1242.
[29] J. Wang, G. Yi, S. Zhang, and Y. Wang, "An unsupervised generative adversarial network-based method for defect inspection of texture surfaces," *Appl. Sci.*, vol. 11, no. 1, pp. 283–298, Dec. 2020.
[30] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "UNet++: Redesigning skip connections to exploit multiscale features in image segmentation," *IEEE Trans. Med. Imag.*, vol. 39, no. 6, pp. 1856–1867, Jun. 2020.
[31] X. Tao, D. Zhang, W. Ma, X. Liu, and D. Xu, "Automatic metallic surface defect detection and recognition with convolutional neural networks," *Appl. Sci.*, vol. 8, no. 9, pp. 1575–1590, Sep. 2018.
[32] Y. Huang, C. Qiu, Y. Guo, X. Wang, and K. Yuan, "Surface defect saliency of magnetic tile," in *Proc. IEEE 14th Int. Conf. Autom. Sci. Eng. (CASE)*, Aug. 2018, pp. 612–617.
[33] M. Wieler and T. Hahn, "Weakly supervised learning for industrial optical inspection," presented at the 29th Annu. Symp. German Assoc. Pattern Recognit., Heidelberg, Germany, Sep. 2007. [Online]. Available: https://hci.iwr.uni-heidelberg.de/content/weakly-supervised-learning-industrial-optical-inspection
[34] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, San Diego, CA, USA, 2015, pp. 1–15.

XIANG XIE received the bachelor's degree in communication engineering from the Institute of Technology, East China Jiaotong University, in 2020. He is currently pursuing the master's degree with the School of Electronics and Communication Engineering, Guangzhou University. His current research interests include computer vision and defect detection.

RONGFENG ZHANG received the bachelor's and master's degrees in optoelectronic technology from Jinan University, Guangzhou, China, in 2004 and 2007, respectively, and the Ph.D. degree in information and communication engineering from the South China University of Technology (SCUT), in 2017. He is currently an Associate Professor with the College of Information Science and Technology, Zhongkai University of Agriculture and Engineering, Guangzhou. His current research interests include machine learning in image processing and the application of machine learning in agricultural informatization.

LINGXI PENG received the Ph.D. degree in computer application technology from Sichuan University, China, in 2008. He is currently a Professor with the School of Mechanical and Electrical Engineering, Guangzhou University. His current research interests include artificial intelligence and medical image processing.

SHAOHU PENG received the bachelor's and master's degrees in signal and information processing from the Guangdong University of Technology, Guangzhou, China, in 2005 and 2007, respectively, and the Ph.D. degree in control and signal processing from Dankook University, South Korea, in 2013. He is currently an Associate Professor with the School of Electronics and Communication Engineering, Guangzhou University. His current research interests include image processing, machine vision, and machine learning.