Journal of Manufacturing Processes 99 (2023) 65–77

Contents lists available at ScienceDirect

Journal of Manufacturing Processes

journal homepage: www.elsevier.com/locate/manpro

LSA-Net: Location and shape attention network for automatic surface defect segmentation

Weifeng Li, Bin Li *, Shuanlong Niu, Zhenrong Wang, Miao Wang, Tongzhi Niu

Digital Manufacturing Equipment and Technology, School of Mechanical Science and Engineering, Huazhong University of Science and Technology, Wuhan 430074, China

ARTICLE INFO

Keywords:
Image processing
Deep learning
Defect segmentation

ABSTRACT

Neural network algorithms for segmenting defects are widely used in industrial production. However, how to fuse the location information of defects in a single model and avoid the features extracted by different submodules gradually tend to be similar during the training is still a problem. To solve these problems, an end-to-end network is proposed that focuses on defect location and shape features, which can guarantee the difference between features extracted by different submodules. In the encoding stage, the location attention module enhances the perception of defect locations. The shape detection module with feature difference loss is designed to strengthen the detection of defect shapes. In the decoding stage, the features of different scales are fused to obtain the final defect region. The experimental results confirm the effectiveness of the proposed location and shape detection modules in the intersection over union on four datasets.

1. Introduction

Surface defect segmentation is an important method for accurate defect detection. Currently, defects are detected through visual inspection by trained workers; this approach has a low efficiency and is labor intensive and subjective. Therefore, manual detection methods do not meet the needs of industrial development. Machine vision has the advantages of being more accurate, robust, and objective. As a result, defect segmentation methods based on machine vision have received increasing interest from researchers.

However, due to the variety of defects and backgrounds, the segmentation of surface defects has the following challenges: (1) Since industrial defects are mostly concentrated in a certain position in the image, defect detection is easily affected by uneven illumination and machined textures in the image background, as shown in Fig. 1(a)(b). Fig. 1(b) shows the grayscale image thresholding results of Fig. 1(a). The gray value of the defect in the image has a high similarity with the background. But around the defect region, it is possible to distinguish the defect from the background. (2) Defects have variable shapes. The inconsistency of machining parameters (e.g., speed, cutting force, and cutting angle) leads to variability in the shape of defects. Fig. 1(c) shows examples of different defect shapes.

Defect segmentation methods can be divided into two classes depending on their feature extraction methods: traditional detection methods and deep learning methods [1]. Early developments in the field, such as edge detector [2] and Histogram [3,4], focused on constructing methods by analyzing target features. However, these methods are generally highly sensitive to noise and have a low generalizability to other defects.

In recent years, the establishment of massive datasets and the rapid development of high-performance computers have promoted the application of neural networks, such as object detection [5] and segmentation [6]. In the existing methods, to avoid the influence of background in the image, a typical method is to use two-stage detection method. This method first uses a localization network [7,8] to locate the detection region and then uses a segmentation network for segmentation. The two-stage detection method has advantages in the segmentation of small defects in large images. However, when the image size is small and the difference in the defect size is obvious, the real-time performance of this method is reduced due to the two-step process. Especially, if the defect localization result is inaccurate, the defect input to the segmentation network after cropping is incomplete, and part of the information is lost, resulting in an incomplete segmentation result. At the same time, in order to adapt to different scales and shape features of segmentation targets, convolutional neural networks (CNN) based on multi-layer feature fusion have been widely used, such as FCN [9], UNet [10], Refinenet [11] and so on. These methods fuse different levels of features to enhance the detection performance but obtain wider receptive field features only through deeper networks. In addition, researchers have designed convolution kernels or pooling

* Corresponding author.
E-mail address: libin999@hust.edu.cn (B. Li).

https://doi.org/10.1016/j.jmapro.2023.05.001
Received 10 February 2023; Received in revised form 28 March 2023; Accepted 1 May 2023
Available online 17 May 2023

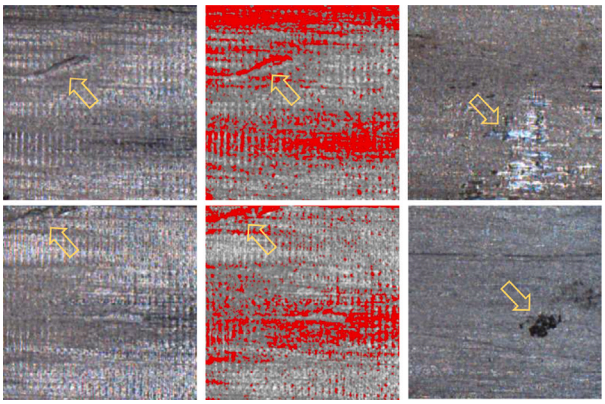

Fig. 1. Typical defect images are used to illustrate the challenges faced in the industry. (a)(b) Defect segmentation is easily affected by the background. (c) Defect with great difference of shape. The yellow arrow indicates the defect region. The red region in (b) is the threshold result. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

kernels with different receptive fields to process features to increase the diversity of features so that the network can adapt to segmentation targets with different shapes and scales, such as pyramid pooling and strip pooling [12]. However, the differences between the features extracted by the submodule composed of the strip pooling kernel and the features extracted by the submodule composed of the square pooling kernel gradually decrease and tend to be similar during training, as shown in Fig. 2 blue line. This is contrary to the original intention of designing different modules to obtain diverse information. We are not aware of any publications describing how to avoid the tendency of features extracted by submodules consisting of different pooling kernels to be similar during training.

Based on the above two limitations, we designed an end-to-end detection model that is compatible with the advantages of two-stage detection and end-to-end detection and that avoids the situation in which the features extracted by different submodules gradually tend to be similar in the training process, as shown in Fig. 3. In particular, the model can meet real-time requirements. In the model, inspired by the attention mechanism [13], we propose a location attention module (LAM) that focuses on defect locations to reduce the effects of background noise and disturbances in images. Under the constraint of defect localization loss, this module obtains defect location information by projecting and reconstructing images, which is beneficial to the detection of concentrated defects. In Fig. 3, after being processed by the defect location attention module, the noise interference in the non-defect area is significantly reduced, and the location information of the defect is obtained. Second, a shape detection module (SDM) is proposed, in which the designed feature difference loss is used to ensure the feature difference between different submodules, as shown in the orange line in Fig. 2. This improves the network's ability to adapt to defects of various shapes and sizes. In Fig. 3, different submodules refer to sub-modules composed of square pooling kernels or strip pooling kernels, and the feature difference loss acts on the feature groups extracted by the two modules. Finally, we used a fine extraction module (FEM) to extract features again by dilation convolution to obtain a greater accuracy in a larger perceptual field. Last, we used skip connections to fuse the different scale features, and the decoder obtains the target defect region by decoding.

According to the method proposed above, we conducted validation experiments on three public datasets and one self-built dataset to demonstrate the robustness and effectiveness of our network. The three major contributions of this paper are as follows:

- Based on image projection and reconstruction, a defect location attention module is proposed, which can be integrated in the segmentation network to make the model focus on the location of defects and weaken the influence of noise and texture in the background.
- A feature difference loss is designed to avoid features tending to be similar during training. This loss can maximize the dissimilarity between the extracted features of different submodules. The features extracted by different submodules refer to the features extracted by the submodule composed of the strip pooling kernels and the features extracted by the submodule composed of the square pooling kernel in SDM. Finally, we improved the adaptation of the network to defects with various shapes and sizes using SDM with feature difference loss.
- We design a surface defect detection network based on deep learning to integrate the advantages of two-stage detection into the end-to-end detection model. Excellent results have been achieved on four surface defect datasets while ensuring real-time performance. This demonstrates that the proposed method is robust and has application value.

The remainder of this article is organized as follows: Section 2 introduces the advantages and disadvantages of traditional methods and neural network-based methods for surface defect detection. Next, the specific structure of the proposed location and shape attention network (LSA-Net) is described in detail in Section 3. In Section 4, we verify the effectiveness of the proposed network by testing the proposed method on publicly available and self-built datasets. In Section 5, we describe experiments designed to verify the effectiveness of each submodule and loss function in the proposed model. Finally, Section 6 concludes this article.

2. Related work

2.1. Traditional detection methods

Traditional detection methods focus on designing algorithms to extract effective features for defect segmentation. In [14], traditional defect detection algorithms are classified into texture-based methods and shape feature-based methods. The texture-based methods are realized by analyzing the regular features of the surface to be detected. Specifically, this method constructs appropriate features or methods to detect defects by analyzing the grayscale relationship between pixels and their adjacent pixels [15]. Liu [16] et al. proposed an improved multiblock local binary pattern algorithm that described defect features by changing the block size to find a suitable scale to generate a gray histogram vector for defect identification and finally achieved high-speed and high-precision detection. Yang [17] et al. applied a nonsubsampled shearlet transform (NSST) to decompose an image into different subband images, merged the processed subband images, and finally detected the surface defects of magnetic tiles by threshold segmentation. Hu [18] proposed a method for texture surface defect segmentation using an optimized elliptical Gabor filter (EGF). Li [19] detected surface defects by analyzing the frequency domain characteristics of the image. Wang [20] et al. implemented a strip steel surface defect segmentation algorithm by analyzing the grayscale distribution of each column of an image to construct a template. Similarly, [21] also used a template-based method to segment defects. Shape feature-based methods obtain image features through shape descriptors, so the accuracy of shape description becomes the key to image defect recognition algorithms. In the method of applying shape descriptors to detect defects, Yong [22] et al. calculated the contour shape descriptor describing the outer edge of the object area to obtain features with translation invariance to detect longitudinal crack defects in complex backgrounds. Wang [23] et al. detected product surface defects by constructing a region shape descriptor to describe the entire object region at the region of interest (ROI) extraction stage. In addition, some traditional detection methods have been used for defect segmentation

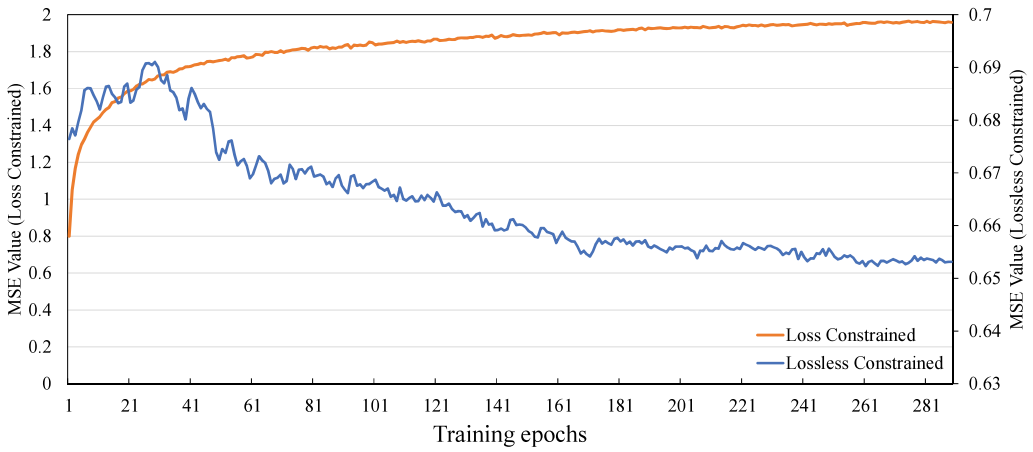

Fig. 2. The difference (MSE Value) between the features extracted by the submodule composed of the square pooling kernel and the features extracted by the submodule composed of the strip pooling kernel. The blue line is the change of the difference between the features extracted by the two submodules with the training process when there is no loss constraint. The orange line is the change of the difference between the features extracted by the two submodules with the training process under the loss constraint we designed. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

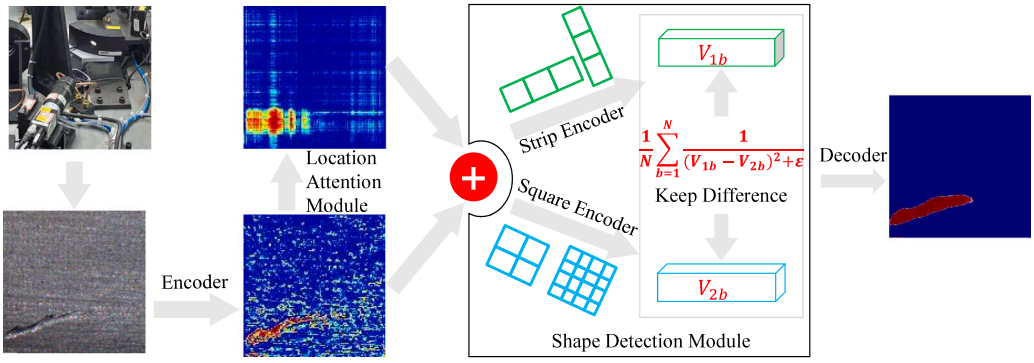

Fig. 3. The key points of the proposed network.

in industrial applications and have achieved good results [24]. However, traditional defect segmentation methods often require adjusting a large number of parameters for different defects and even redesigning the detection plan, which is not universal [25].

2.2. Deep-learning-based segmentation methods

Compared with traditional defect segmentation methods, CNNs have been widely used in various defect segmentation fields due to their excellent adaptability, higher detection accuracy, and strong plasticity. Inspired by the excellent results obtained for various public datasets, the application of CNNs in the industrial field is also gradually increasing. In these applications, according to the different methods of using labels, it can be divided into supervised learning, weakly supervised learning and unsupervised learning. Compared with weakly supervised and unsupervised learning, the segmentation accuracy of supervised learning is higher. This paper uses supervised learning, so it mainly discusses supervised learning methods.

Two-stage detection methods can be used in supervised learning methods to reduce the influence of the background in images. Fu [26] et al. proposed a two-stage attention model to identify the defects of bearing shell oil by first using the localization network to locate the bearing position and then using the segmentation network to segment the oil defect. Fang [27] et al. also used the Faster R-CNN [28] to divide the location of large concrete defects into several parts, input each part into a regression network to predict the defect direction, and finally used the Bayesian integration algorithm to determine the defect region. Li [29] et al. also proposed a two-stage defect detection model that uses an instance segmentation network to segment the detection target and a defect detection network to detect defects in the image. Ni [30] et al. designed an attentional neural network for track surface defect detection by consistency of intersection-over-union (IoU)-guided center-point estimation in a two-stage detection model. Methods based on two-stage detection have been widely used in defect segmentation for tooth segmentation [31] and other applications [32]. The most common approach is to use a location network to realize localization and different models to deal with the identified region. However, the method based on two-stage detection is more suitable for segmenting large images. When applied to small or medium images, the real-time advantage of the model is reduced. In contrast, using different models for localization and segmentation may result in information loss and poor detection results.

In addition to two-stage detection methods, additional methods for single-model segmentation that avoid the loss of information caused by image cropping have been applied. A single-model segmentation method usually contains multiple submodules to increase the ability of the model to segment defects and suppress background interference. For example, Dong [25] et al. used a pretrained neural network to extract pyramid features, fused the pyramid features to obtain features of different resolutions, and finally fused the global information and boundary information of features with different resolutions to detect surface defects. Song [1] et al. also used different modules to fuse the pyramid features extracted by the neural network. They introduced the convolutional block attention module (CBAM) in each encoding block, with a greater focus on the channel information and residual information in the decoding stage compared to Dong's approach.

Liu [33] et al. introduced a global attention mechanism, spatial attention mechanism and channel attention mechanism in the network to enhance the network's ability to detect defects in images. Li [34] et al. designed CASI Net, which uses a lightweight feature extractor to extract image features and then uses a biological vision-based collaborative attention mechanism and self-attention mechanism to process features to detect surface defects. Similar to [25] and [1], Xie [35] et al. also used different modules to fuse the features extracted by the neural network to segment the surface defects of magnetic tiles. In addition to the publications mentioned above, [36–39] also used a single model to accurately segment the surface defects of chips and wafers. Compared with the two-stage detection method, the method based on a single model achieves end-to-end segmentation and designs different submodules to adapt to different defects. However, there is still room for improvement, specifically in terms of preventing the tendency of features extracted by different modules in the network to be similar and fusing the defect location information while ensuring real-time performance.

3. Methods

3.1. Overall architecture

Unet has been widely used in the field of industrial image and medical image segmentation due to its symmetric encoding and decoding structure and the classical skip connection. Therefore, we use the Unet architecture as the basic skeleton and fuse the proposed LAM, SDM and FEM modules to construct the neural network of LSA-Net, as shown in Fig. 4. Table 1 presents the detailed structure of the encoding and decoding stage. In the proposed network, LAM obtains the location information of defects by projecting and reorganizing defects and using the minimum bounding rectangle of the constructed defect as the training target. These location information can enhance the ability of the network to perceive the defect location information in the shallow layer by adding with the backbone features. SDM contains submodules composed of convolution kernels and pooling kernels with different shapes. The feature difference loss designed in SDM amplifies the information difference extracted between each sub-module, makes the deep features more diverse, and ultimately enhances the ability of the network to detect different defects. Finally, FEM is used to further refine the extraction, and the decoder fuses the information of each layer to predict the segmentation result.

Table 1
The detailed structure of the encoding and decoding stage.

Block	Output_size	Type
1	256 × 256 × 32	[Conv3 × 3+ BN+ ReLU]×2(VGG_B)
2	128 × 128 × 64	Maxpool(2 × 2) [Conv3 × 3+ BN+ ReLU]×2(VGG_B)
3	64 × 64 × 128	Maxpool(2 × 2) [Conv3 × 3+ BN+ ReLU]×2(VGG_B)
4	32 × 32 × 256	Maxpool(2 × 2) [Conv3 × 3+ BN+ ReLU]×2(VGG_B)
5	16 × 16 × 512	Maxpool(2 × 2) [Conv3 × 3+ BN+ ReLU]×2(VGG_B)
–	16 × 16 × 512	FEM
6	32 × 32 × 512	Upsample
7	64 × 64 × 256	[Conv3 × 3+ BN+ ReLU]×2(VGG_B) Upsample
8	128 × 128 × 128	[Conv3 × 3+ BN+ ReLU]×2(VGG_B) Upsample
9	256 × 256 × 64	[Conv3 × 3+ BN+ ReLU]×2(VGG_B) Upsample
10	256 × 256 × 32	[Conv3 × 3+ BN+ ReLU]×2(VGG_B)
11	256 × 256 × 1	Conv1 × 1

3.2. Location attention module

Fu [26] showed that performing localization first, followed by segmentation, is beneficial for reducing background interference in images. Inspired by this, when segmenting small-sized images, we hope to design a network structure that reduces the interference of background noise in the image without increasing too much model running time. The defect region shown in the orange region in Fig. 5 illustrates the principle of obtaining defect location information. We use the principle of projection to project defects along the horizontal and vertical directions through strip pooling to obtain two vectors that reflect the position information of defects in the horizontal direction and vertical direction, as shown in Fig. 5(b). The two vectors are multiplied to construct the feature image that reflects the location information of the defect, as shown in Fig. 5(c).

Fig. 6 depicts our proposed LAM. Let $X \in \mathbb{R}^{C\times H\times W}$ be an input tensor, where H and W are the spatial height and width, respectively, and C denotes the number of channels. We first input X into a VGG_B to fuse different levels of information followed by a set of 1×1 convolutions to change the channel dimension and divide it into n groups of features $X_i \in \mathbb{R}^{C'\times H\times W}, i = 1, 2, \ldots, n, C' = C/n$. For each X_i, we input X_i into two parallel pathways, namely, row strip pooling (PoolR) and column strip pooling (PoolC). The output is denoted as $X_i^r \in \mathbb{R}^{C'\times H}$ and $X_i^c \in \mathbb{R}^{C'\times W}$, respectively. Then, X_i^r and X_i^c are multiplied to obtain the feature $Y_i \in \mathbb{R}^{C'\times H\times W}$ that reflects the location of the defect. Finally, a concatenation layer follows a VGG_B for aggregate individual data, yielding $Y^{C\times H\times W}$. An expression for the above process is as follows:

$$Y = VGG\_B(concat[f(X_i^r, X_i^c)])i = 1, 2, \ldots, n. \quad (1)$$

where $f(.,.)$ refers to matrix multiplication, $concat[]$ represents the concatenation operation, and $VGG\_B()$ is a VGG block.

It is worth noting that different from the target recognition networks, such as Mask R-CNN [40] and Mask Scoring R-CNN [41], LAM does not return any specific data of location information, nor does it outline the specific framework of reacting defect location, and only obtains the location attention information of defects. This information will not be displayed, but only exists in the feature map, which is used to enhance the perception ability of the network for defect location information in the shallow layer. Fig. 7 shows the effect of LAM on shallow features. Comparing the feature map Fig. 7(d) before being processed by the LAM module and the feature map Fig. 7(e) after being processed by the LAM module, it can be found that the LAM module effectively reduces the influence of background noise in the image and gives information about the location of the defect. Defect locations in the output feature map Fig. 7(f) used to compute the loss function are more accurate.

Based on the above analysis, the acquisition of defect location information is not obtained through a separate defect location network, such as Faster R-CNN [28], so that the increase in network running time is still within an acceptable range. At the same time, LAM is constrained by the segmentation loss. During training, the constraint target is obtained by finding the minimum bounding rectangle of the defect, which further enhances the network's ability to utilize label information.

3.3. Shape detection module

After LAM extracts the location information of the defect, we design the feature extraction module to extract the defect shape information in the deep layer of the network. As described in [28], pooling kernels with different shapes yield richer receptive field information, which is helpful for the detection of defects with various shapes. It is worth noting that in the defect detection task, pyramid pooling is the most widely used method. However, detect defects by fusing features

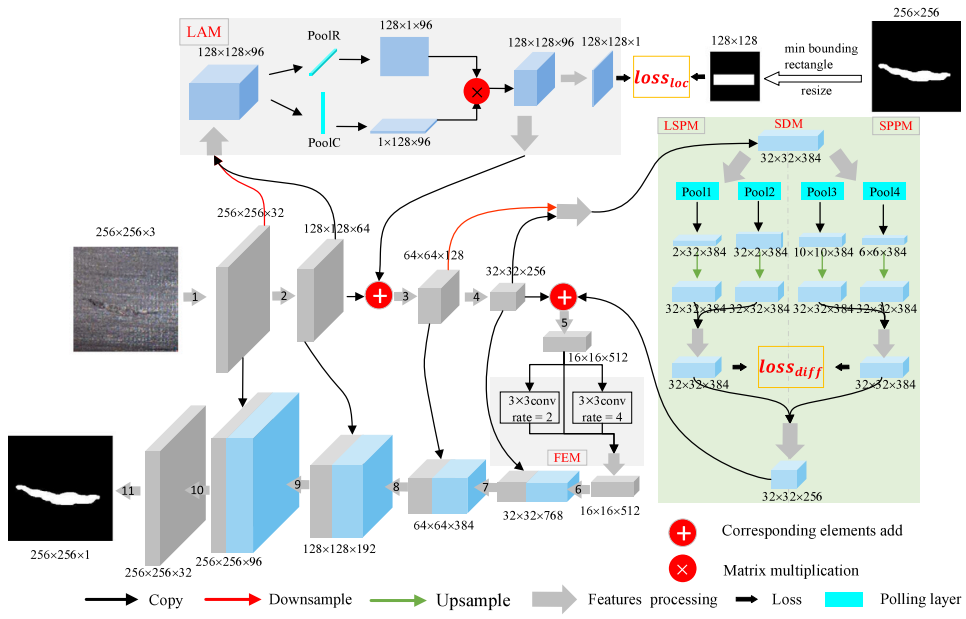

Fig. 4. The overall architecture of LSA-Net.

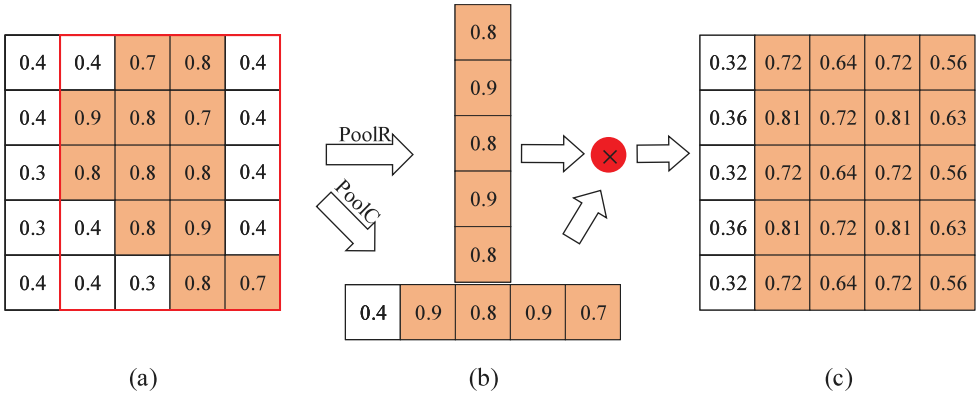

Fig. 5. The principle of strip pooling to realize defect location. (a) The orange shading denotes the defect region, and the red rectangle is the minimum bounding rectangle of the region. (b) Strip pooling output result. (c) The result of matrix multiplication of two pooling out, in which the orange region denotes the defect location. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

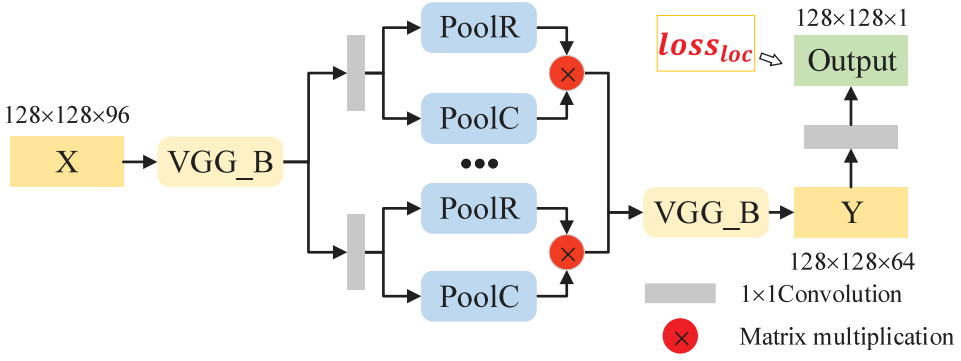

Fig. 6. Details of the LAM. The LAM first inputs features into a VGG_B to fuse features from different levels followed by a set of 1×1 convolutions to change the channel dimension. Then, matrix multiplication of the row and column strip pooling output is used to obtain the defect location. Finally, VGG_B is used again to fuse different outputs. The LAM is characterized by deep-supervision based on the location of the defect.

obtained by strip pooling kernels in deep features is rare. To apply the strip pooling kernel and pyramid pooling to the field of defect detection, we designed the SDM, and the structure is shown in Fig. 8. The SPPM and LSPM are the two core submodules. The outputs of the SPPM and LSPM are denoted as $V_1 \in \mathbb{R}^{C_1 \times H_1 \times W_1}$ and $V_2 \in \mathbb{R}^{C_1 \times H_1 \times W_1}$, respectively. The main components of the two submodules are shown in Tables 2 and 3. SPPM and LSPM are used to extract square receptive field features and strip receptive field features, respectively. Fig. 9

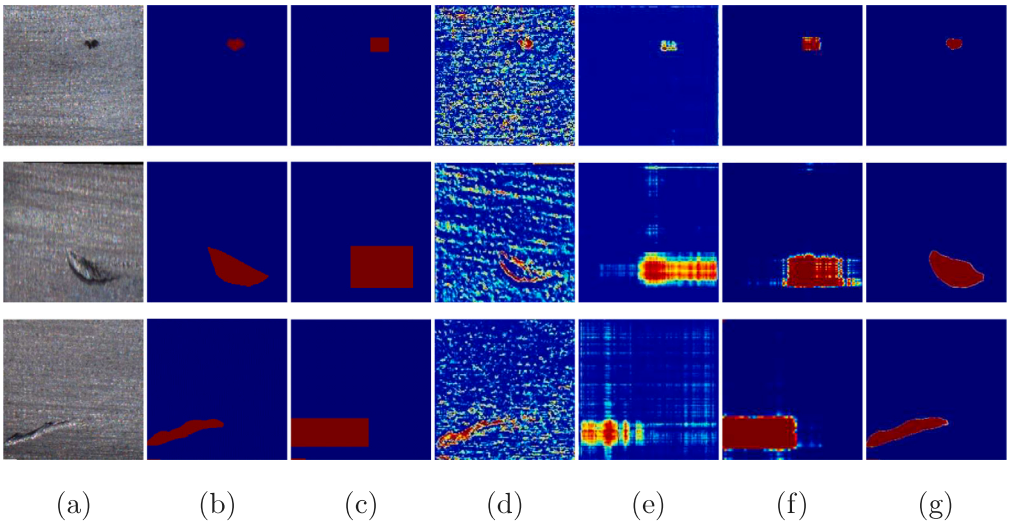

Fig. 7. Visualization of feature maps at different stages for the proposed LSA-Net. (a) Original image. (b) Defect label. (c) Location label. (d) Feature visualization before LAM processing. (e) Feature visualization after LAM processing. (f) Feature map used to compute the localization loss of the LAM output. (g) Defect segmentation result of the final output of the network. Comparing (d), (e) and (f), shows that after LAM processing, the network pays more attention to the location of defects and reduces the noise in the background.

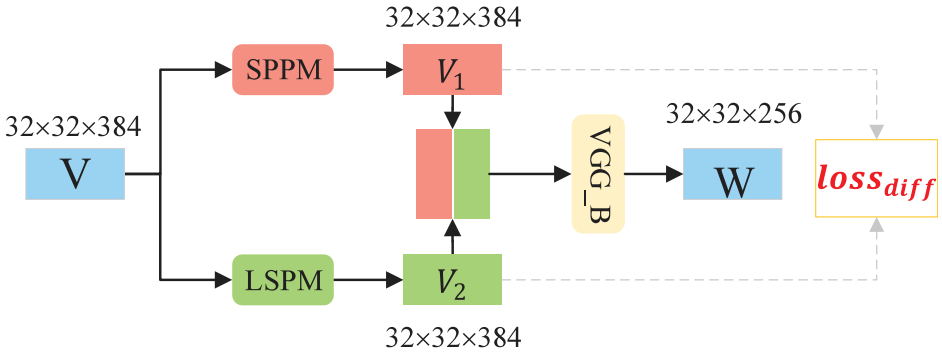

Fig. 8. Architecture of the SDM. The SDM inputs feature into two submodules (SPPM and LSPM) to obtain different receptive field features, followed by a VGG_B to fuse and reduce the dimensionality of the concatenated input features and different receptive field features.

shows the feature maps before and after SPM processing and the feature maps output by SPPM and LSPM. Fig. 9 shows that, unlike the shallow features in which the outline and texture of the defect can be clearly seen in Fig. 7, the deep features show only the activation state of the region where the defect is located. The background value is also no longer zero and appears green instead of blue in the pseudocolor map. However, the feature map still reveals that the features processed by SPPM significantly differ from those processed by LSPM. Specifically, the rectangular pooling kernel has a better effect on processing block features, and the strip pooling kernel performs better on strip defects, as shown in Fig. 9. This characteristic is in line with the original intention of designing the shape detection module, which is to obtain more diverse features to enhance the adaptability of the network to different shape defects. More importantly, the two different methods reduce the noise interference in the image in different ways and finally enhance the model's ability to detect defects.

These feature extraction submodules composed of convolution kernels or pooling kernels of different shapes have been applied in many papers, such as PSPNet [42] and SPNet [12]. However, we found that during network training, the information extracted between different modules will gradually tend to be consistent under the influence of the final segmentation target, which is contrary to the original intention of designing different modules to obtain diverse information. To solve this

Table 2
Detailed structure of the SPPM.

Output_size	SPPM	
32 × 32 × 384	[Conv3 × 3+ BN+ ReLU]×2(VGG_B)	
–	AVG_Pool(5 × 5)	AVG_Pool(3 × 3)
32 × 32 × 384	Upsample	Upsample
32 × 32 × 768	Concat	
32 × 32 × 384	Conv1 × 1	
32 × 32 × 384	[Conv3 × 3+ BN+ ReLU]×2(VGG_B)	

Table 3
Detailed structure of the LSPM.

Output_size	LSPM	
32 × 32 × 384	[Conv1 × 3+ BN+ ReLU]×2	[Conv3 × 1+ BN+ ReLU]×2
–	Max_Pool(1 × 16)	Max_Pool(16 × 1)
32 × 32 × 384	Upsample	Upsample
32 × 32 × 384	Corresponding elements add	
32 × 32 × 384	[Conv3 × 3+ BN+ ReLU]×2(VGG_B)	

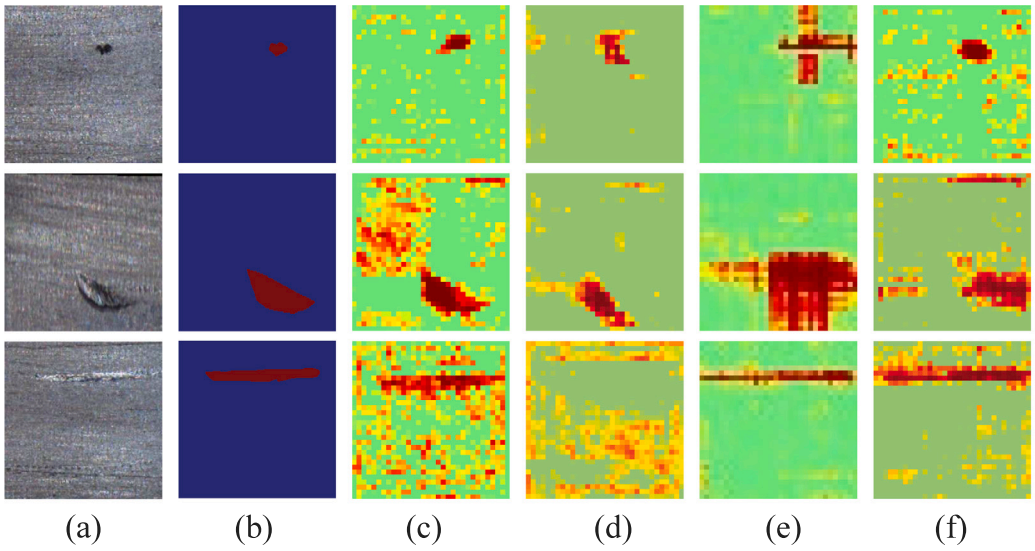

Fig. 9. Visualization of feature maps at different stages for the proposed SDM. (a) Original Image, (b) Label, (c) Import image features of SDM, (d) Output image features of SPPM, (e) Output image features of LSPM, (f) Output image features of SDM. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

problem, we introduce a feature difference loss, which aims to enhance the difference in the features acquired by the different modules. That is, we force the features extracted by the two submodules to be different by using the feature difference loss.

For the feature difference loss, a simple and effective approach is to calculate the difference between two sets of features. Therefore, the loss function can use the MSE [43] loss:

$$MSE = \frac{1}{N}\sum_{b=1}^{N}(V_{1b} - V_{2b})^2 \tag{2}$$

However, the MSE [43] loss is not suitable for optimization by gradient descent. During training, the optimization target is to gradually increase the MSE loss. For example, in training a segmentation task, the MSE loss will gradually increase, but the segmentation loss (cross-entropy loss [44]) will gradually decrease. To optimize the MSE loss by gradient descent, similar to the cross-entropy loss, we modify the above formula as follows

$$loss_{diff} = \frac{1}{N}\sum_{b=1}^{N}\frac{1}{(V_{1b} - V_{2b})^2 + \varepsilon} \tag{3}$$

In the formula, ε is to avoid parameter explosion when two sets of different features are completely the same. N denotes the batch size, and V_{1b} and V_{2b} are the flattened output features of the different modules of the bth image, respectively.

For $loss_{diff}$, with the decrease of the loss, the difference will be gradually increased between the features extracted by different submodules. As a result, the $loss_{diff}$ with other losses can be optimized by gradient descent together. Fig. 10 shows the change in the feature difference loss ($loss_{diff}$) during training. In the figure, *Included in total loss* refers to computing $loss_{diff}$ and participates in backpropagation as part of loss (blue), while *Not included in total loss* refers to computing $loss_{diff}$ but not participating in backpropagation as part of loss (orange). By comparing the two curves, we found that under the constraint of $loss_{diff}$, the difference between V_1 and V_2 gradually increased ($loss_{diff}$ gradually decreases); in contrast, without the constraint of $loss_{diff}$, the difference between V_1 and V_2 gradually decreased ($loss_{diff}$ gradually increases). This finding proves that the effect of $loss_{diff}$ was consistent with the original design target, that is, to increase the dissimilarity of the two sets of features and to reduce the similarity.

The SDM module implementation details are described as: Given a set of features $V \in \mathbb{R}^{C_1 \times H_1 \times W_1}$, we adopt the SPPM for square receptive field feature collection. We first input the feature V into a VGG_B to fuse each group of features and then input them into two square pooling layers ($3 \times 3, 5 \times 5$). The processed features are upsampled and concatenated into a set of features and then processed by a convolutional layer consisting of a 1×1 convolution and VGG_B to reduce the dimension of the features and to fuse them. The LSPM inputs feature V into the row convolution kernel (3×1) and column convolution kernel(1×3) for further feature extraction and then uses local row strip pooling ($\frac{H_1}{2} \times 1$) and local column strip pooling ($1 \times \frac{W_1}{2}$) to obtain the strip receptive field information. After that, the two features are upsampled and added together to fuse information from different directions in the same region. The VGG_B is introduced to merge information for the adjacent region. Finally, V_1 and V_2 are concatenated and input into a VGG_B to change the channel dimension and modulate the current location and its neighboring features. Thereafter, the feature $W \in \mathbb{R}^{C_1 \times H_1 \times W_1}$ is obtained. $loss_{diff}$ constrains the two sets of features to always have a certain difference between them during training.

3.4. Fine extraction module

In deep CNNs, atrous convolution with different atrous rates can effectively capture multiscale information and increase the weight of the target region to accurately detect objects at different scales [45]. Following [45], we used dilation convolution to obtain information features of different scales and enhance the expression of effective features, as shown in Fig. 11. Based on the output of the SDM, we propose inputting $U \in \mathbb{R}^{C_2 \times H_2 \times W_2}$ into the FEM to acquire multiscale features. The outputs from both dilation $3 \times 3conv$ with $rate = 2$ and dilation $3 \times 3conv$ with $rate = 4$ are concatenated together and then fed into a 1×1 convolutional layer for channel reduction. The output of the FEM is denoted as $Z \in \mathbb{R}^{C_2 \times H_2 \times W_2}$.

3.5. Loss function

The proposed network defect detection method is realized by supervised learning, so it requires data label constraint training. The loss constraint addresses three main problems in the network defect detection process: defect segmentation, defect location identification, and defect shape recognition.

Defect segmentation: The ultimate goal of the model is to accurately segment the defect region. In selecting the loss function, we

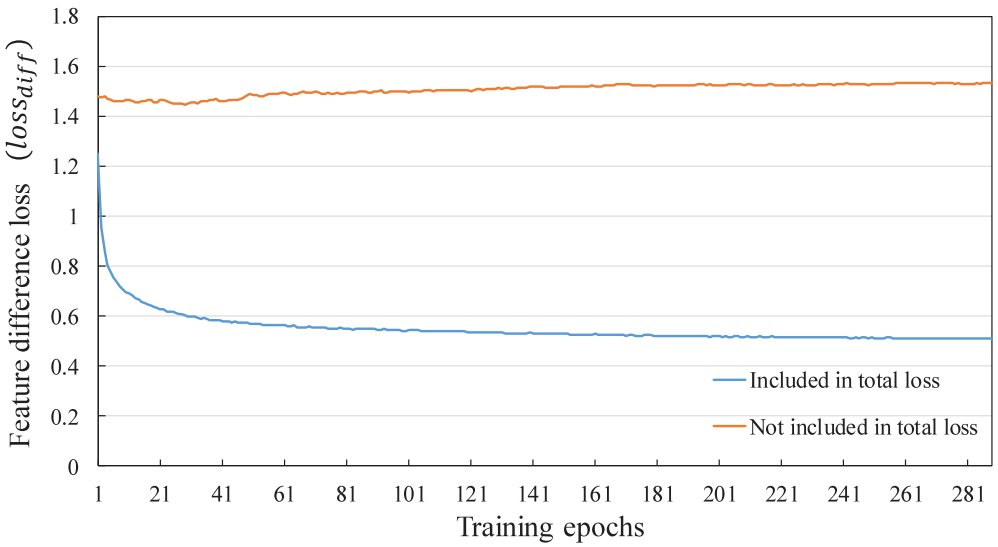

Fig. 10. The change of feature difference loss ($loss_{diff}$) with different setting conditions on the MCSD-C dataset. *Included in total loss* refers to computing $loss_{diff}$ and participates in backpropagation as part of loss (blue). *Not included in total loss* refers to computing $loss_{diff}$ but not participating in backpropagation as part of loss (orange). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

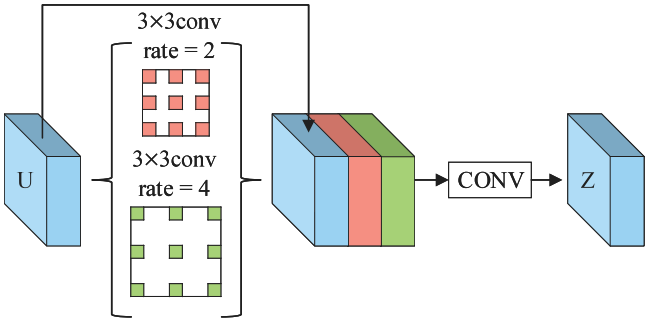

Fig. 11. Details of the FEM.

first select the BCE loss, which is widely used in segmentation tasks. BCE loss focuses on measuring the overall distribution of the image. In industrial data, because the defect region usually accounts for a small proportion, BCE loss is easily affected by the background. To give more attention to defect regions, we choose the Dice loss, which focuses on measuring the overlapping regions between defects and labels. The weighted loss function based on BCE and Dice can combine the advantages of the two loss functions [46–48]. When this weighted loss is applied to industrial data, it can not only focus on the overall distribution of the image but also focus on the defect region, which is denoted by $loss_{seg}$, as in [6].

$$loss_{seg} = -\frac{1}{N}\sum_{b=1}^{N}\left(\frac{1}{2}Y_b log\hat{Y}_b + \frac{2Y_b\hat{Y}_b}{Y_b+\hat{Y}_b}\right) \tag{4}$$

where N denotes the batch size, and $\hat{Y}_b$ and Y_b are the flattened predicted localization probabilities and the flattened localization ground truths of the bth image, respectively. $\hat{Y}_b$ is the prediction result output by feature $Y^{C\times H\times W}$ using 1×1 convolution dimension reduction in the location attention module, and Y_b is the defect location label generated by the label.

Defect location identification: In the proposed neural network, the LAM is an auxiliary module and is mainly used to strengthen the ability of the model to perceive defect positions in the shallow layer and weaken the influence of the nondefect region. Because this process is realized by strip pooling, we use the label image to generate the bounding rectangle of the defect region to calculate the loss function, as shown in Fig. 4. The calculation method of the loss function is the same as that of $loss_{seg}$, expressed as $loss_{loc}$.

Defect shape recognition: As mentioned above, in the SDM module, different receptive field information is obtained through the SPPM and LSPM to obtain more detailed shape information. To avoid the reduction of the difference of the extracted features during the training process of the two modules, we introduce the feature difference loss, which aims to enhance the difference of the features acquired by different modules.

The total loss function is written as follows:

$$loss = \lambda loss_{seg} + \mu loss_{loc} + \nu loss_{diff} \tag{5}$$

where $loss_{seg}$, $loss_{loc}$, and $loss_{diff}$ represent the segmentation loss, location loss, and feature difference loss, respectively, and λ, μ, and ν are the weights of the three loss functions.

4. Experiments and results

4.1. Datasets

To verify the effectiveness of LSA-Net, we tested our method on various datasets. These datasets contain defects such as point defects, block defects, and strip defects and a variety of complex background environments, which ensure that the method has sufficient coverage on the test dataset. We assessed the proposed LSA-Net on four industrial datasets, including three benchmark datasets, KSDD [49], NEU-Seg [50], and MT Defect Dataset [51], and one dataset collected and established from the commutator production line (Fig. 12), referred to as MCSD-C. Fig. 13 shows partial sample images and corresponding labels for the four datasets.

The motor commutator cylinder dataset (MCSD-C) is a motor commutator surface dataset that was acquired by using a fixed arc light source and a high-speed industrial camera, as shown in Fig. 12. It shows the appearance of the commutator workpiece and the image taken from the inspection region. MCSD-C consists of different batches of images; due to changes in the processing parameters, large differences in the image background and defect shape are present, which makes the background complex and the defect features changeable. This dataset consists of 542 nondefective samples and 566 defective samples from different batches of workpieces. The image size was 256 × 256. There are five types of defects including slag inclusion, scratch, crush, dirty

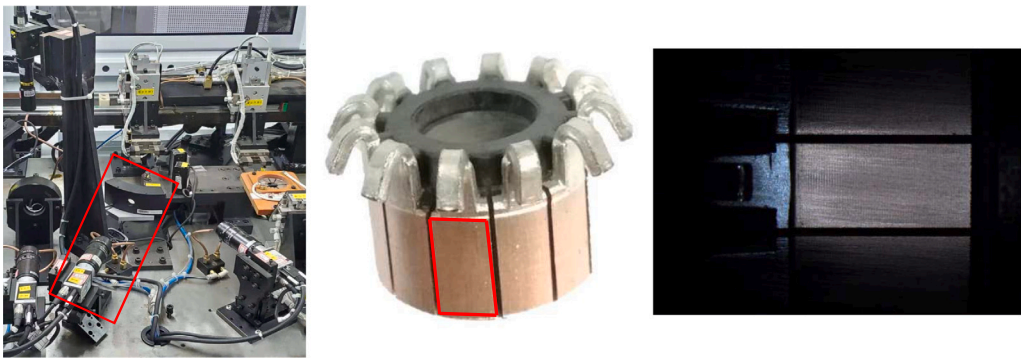

Fig. 12. (a) Photo location in the production line,(b) Commutator workpiece, (c) an image of a contact piece image. The red rectangle is the detection region of this paper. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 4
Image size, training set and test set split for different datasets.

Datasets	Image size	Training set	Tests set
MCSD-C	256 × 256	886	222
KSDD	512 × 512	147	49
NEU-Seg	256 × 256	480	120
MT defect dataset	256 × 256	341	51

surface and tin remaining. These five types of defects include point defects (mainly slag inclusion defects), block defects (mainly surface dirt, tin remaining, and partial scratch defects), and strip defects (mainly crush and partial scratch defects).

KSDD was proposed in [49] and includes 50 motor commutator workpiece images. The KSDD dataset has more concentrated defects and weaker features than MCSD-C. This dataset includes a total of 399 images and corresponding labels but only 54 defective samples. All the defects are crack defects, showing stripe characteristics, which can reflect the detection ability of the network for stripe defects. To expand the defect samples, the defect images were divided in steps of 500 × 500 to obtain a dataset that contains 49 test images and 147 training images. The image size was scaled to 512 × 512 during training.

The MT defect dataset is a magnetic tile (MT) surface defect dataset. Compared with MCSD-C and KSDD, the MT dataset has more diverse defect shapes, more obvious size differences, and worse illumination intensity uniformity, which all pose challenges for defect detection. The MT dataset contains 5 kinds of defects, including blowhole, break, crack, fray, and uneven, a total of 392 sets of defect images and corresponding labels of different resolutions, and the size of the training image is adjusted to 256 × 256. Similar to the MCSD-C dataset, this dataset also contains point defects (blowhole), block defects (most break defects and fray defects and uneven defects) and strip defects (crack and a small amount of break).

The NEU-Seg dataset is a hot-rolled steel strip surface defect dataset. The difference between this dataset and other datasets is that each image contains multiple defect regions, which is challenging for defect detection. This dataset includes three types of defects, each with 300 defects, and the annotation quality of the labels is high. The size of the training image is adjusted to 256 × 256. We selected two kinds of defects, inclusions and scratches, both showing a strip defect.

To ensure the coverage of various defects in the test set and training set, when dividing the above dataset, we use a random partition method to divide the dataset into two parts: the test set and the training set. The split results are shown in Table 4. Additionally, to prevent a certain defect type from being completely divided into the training set or test set, we fine-tune the divided dataset to ensure that each type of defect will appear in the training set and test set.

4.2. Performance metrics

The result of the network prediction is a black and white image, where black represents the normal region and white represents the defect region. To measure the segmentation result at the pixel level, we use the intersection over union (IoU), pixel accuracy (PA), and Dice's Coefficient (Dice) as an indicator to denote the similarity between the segmentation results and the labels.

$$IoU(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cup B|} \tag{6}$$

$$Dice(A,B) = \frac{2|A \cap B|}{|A| + |B|} \tag{7}$$

$$PA = \frac{\sum_{i=0}^{k} p_{ii}}{\sum_{i=0}^{k} \sum_{j=0}^{k} p_{ij}} \tag{8}$$

where A represents the prediction result, and B is the image label. p_{ii} is the total number of pixels with true pixel class i that are predicted to be class i. p_{ij} represents the total number of pixels with true pixel class i that are predicted to be class j.

4.3. Experimental setup

Parameter Setting: The batch size for all datasets during training was 16. The weight of each submodule in the loss function was a fixed value, and λ, μ, and ν were set to 1.4, 0.4, and 0.2, respectively. We adopted the 'Cosine Annealing LR learning rate policy' in training. The base learning rate, power, momentum, and weight decay rates were set to 0.001, 0.9, 0.9, and 0.0001, respectively. The training epochs were set to 500 to ensure that the network fully converged. For data augmentation, we performed random vertical flipping, hue, saturation and value, random change brightness, and contrast of the input image.

Computation platform: The network was implemented on PyCharm using the open source toolbox PyTorch, running on an NVIDIA Tesla P100 with a 32 GB GPU unit on Centos 8 Linux.

4.4. Experiment and analysis

MCSD-C Dataset: The main challenge for MCSD-C dataset was the complex and changing background. Table 5 shows that the proposed method outperformed the previous state-of-the-art methods, and the performance was approximately 3.74% (0.7224 → 0.7494) higher on IoU than the best performance among the other methods. The results in Rows 1–3 in Fig. 13 are listed as the partial detection results of our network on the MCSD-C dataset. Row 1 shows that LSA-Net better preserves details, and Row 2 shows that LSA-Net retains segmentation results well under strong background noise. Row 3 shows the defect-free segmentation results of LSA-Net.

KSDD Dataset: The KSDD dataset mainly tested the ability of the proposed LSA-Net to detect weak contrast strip defects with a small

Table 5
Performance on four datasets for different detection methods.

Methods	MCSD-C			KSDD			MT Defect			NEU-Seg		
Metrics	PA	Dice	IoU	PA	Dice	IoU	PA	Dice	IoU	PA	Dice	IoU
U-Net [10]	0.9923	0.4473	0.7171	0.9946	0.8220	0.7151	0.9867	0.7810	0.8813	0.9733	0.8093	0.7281
RefineNet [11]	0.9926	0.4500	0.7165	0.9941	0.8049	0.6873	0.9848	0.7883	0.8503	0.9731	0.8157	0.7232
DeepLabV3+ [45]	0.9927	0.4486	0.7224	0.9930	0.7859	0.6469	0.9868	0.7656	0.8815	0.9728	0.8013	0.7124
FPA-Net [37]	0.9914	0.4406	0.6732	0.9923	0.7583	0.6177	0.9849	0.7431	0.8539	0.9690	0.7791	0.6744
DANet [13]	0.9925	0.4433	0.7075	0.9931	0.7710	0.6487	0.9873	0.8052	0.8780	0.9754	0.8246	0.7471
SPNet [12]	0.9910	0.4134	0.6499	0.9904	0.5852	0.4755	0.9636	0.3081	0.6395	0.9598	0.7134	0.5789
PGA-Net [25]	0.9928	0.4389	0.7130	0.9926	0.7385	0.6111	0.9862	0.8018	0.8688	0.9749	0.8228	0.7420
NDD-Net [38]	0.9765	0.4556	0.4081	0.9947	0.8244	0.7141	0.9871	0.8038	0.8834	0.9751	0.8275	0.7447
LSA-Net	**0.9936**	**0.4557**	**0.7494**	**0.9950**	**0.8451**	**0.7343**	**0.9883**	**0.8178**	**0.9175**	**0.9758**	**0.8283**	**0.7594**

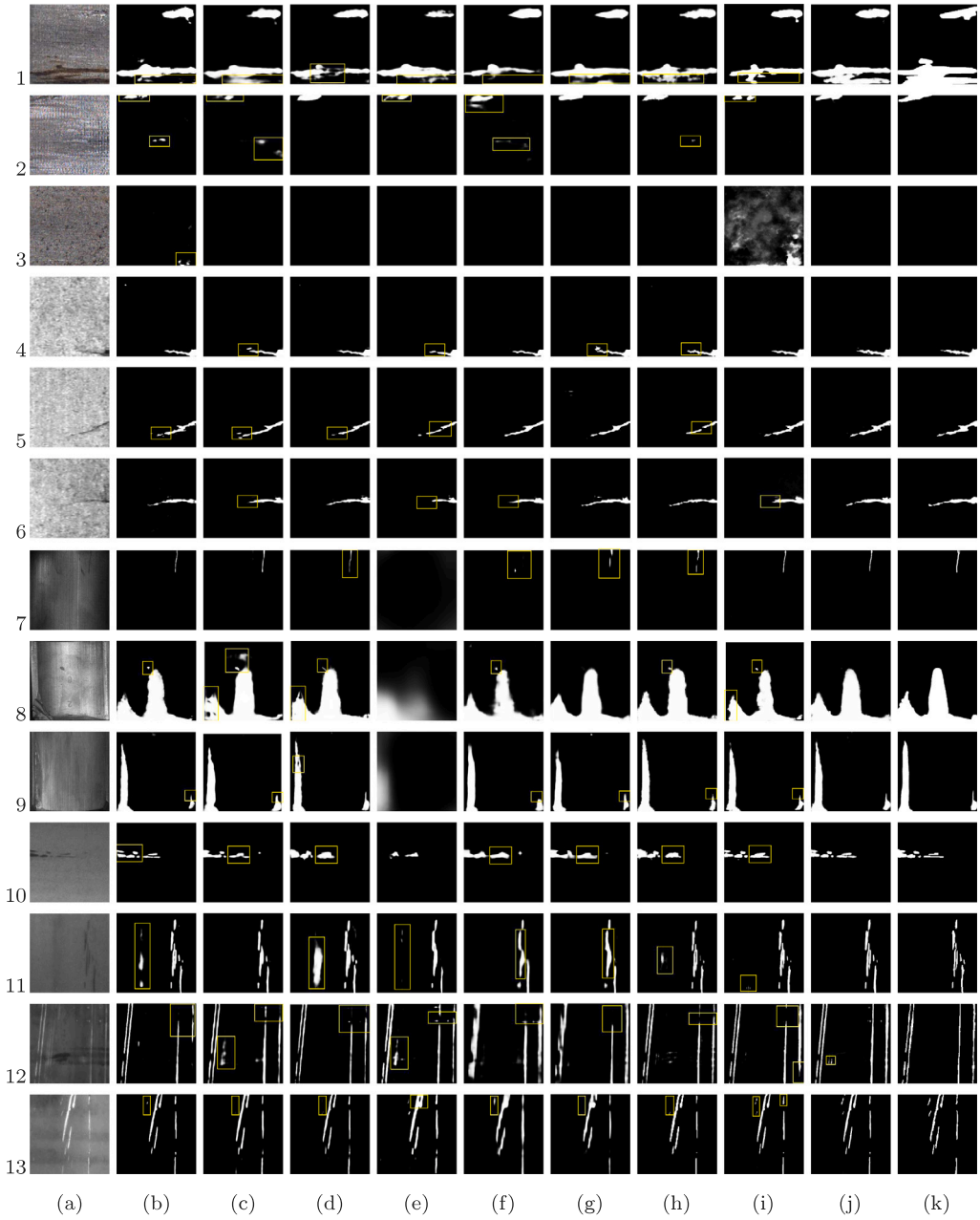

Fig. 13. Comparison of the detection results of the different detection methods for the different datasets. (a) Original image. (b) U-Net. (c) RefineNet. (d) DeepLabV3+. (e) FPA-Net. (f) DANet. (g) SPNet. (h) PGA-Net. (i) NDD-Net. (j) LSA-Net. (k) Label. Rows 1–3 are the MCSD-C dataset, Rows 4–6 are the KSDD dataset, Rows 7–9 are the MT defect dataset, Rows 10–13 are the NEU-Seg dataset.

Table 6
Comparison of the two-stage detection network on the MCSD-C dataset.

Methods	APP-Unet [26]		LSA-Net
	Localization	Segmentation	
Times (ms)	36.67	19.77	16.73
FLOPs (G)	94.41	102.01	61.60
Params (M)	136.67	24.20	21.60
IoU	0.6455		0.7494

number of sample images. Table 5 shows that in the KSDD dataset, LSA-Net also improved the IoU value by approximately 2.68%(0.7151 → 0.7343) compared to the other networks. Combining the results in Table 5 and Fig. 13, LSA-Net retained the sensitivity of [12] to stripe defects while effectively reducing the interference of other background noises in the image.

MT Defect Dataset: The MT dataset mainly verified the detection effect of the network for different defects under the conditions of uneven illumination, complex background and large shape differences. As shown in Table 5, LSA-Net achieved better detection results compared to the other methods in the case of extreme differences in the shape of the defects (in Fig. 13, row 7 small strip defects and rows 8–9 large-region bulk defects), and the IoU value was approximately 3.86%(0.8834 → 0.9175) higher than that of the other methods.

NEU-Seg Dataset: The NEU-Seg dataset was mainly used to examine the adaptability of LSA-Net when the defects are densely distributed. Based on the data in Table 5, when the defects were densely distributed, the network detection result was approximately 1.65%(0.7471 → 0.7594 higher than that of [45] on IoU. However, Fig. 13 shows that LSA-Net was better at extracting details, especially for rows 12 and 13. Therefore, our proposed LSA-Net is also suitable for the detection of dense defects.

Compared with the two-stage detection network:

The data in Table 6 are the experimental results of LSA-Net and the method in [26] on running Times, Params, FLOPs, and IoU. The results show that the overall running time of our algorithm was approximately 3.37 times faster than that of the method in [26]. The proposed method also exhibited advantages compared to the single-stage segmentation time in [26]. The running time of LSA-Net met the needs of our production cycle. For the model size, compared to the method in APP-UNet [26], the FLOPs and Params of our designed LSA-Net are reduced by 68.72% and 86.58%, respectively. In particular, compared with the segmentation model in APP-UNet, our proposed model has lower FLOPs and Params (reduced by 39.61% and 10.7%, respectively). The main reasons for this result include the following. First, in the two-stage detection method, there is a separate encoding process for both the localization network and the segmentation network. However, our method needs to be encoded only once, and the information reflecting the defect location can be obtained after further processing the encoded information. Second, the localization stage in APP-UNet requires proposals processing and nonmaximum suppression to obtain the location information of the defect, which is complex and time-consuming. Since our method does not have proposals and nonmaximum suppression processing, the amount of calculation is further reduced. Notably, when tested on the MCSD-C dataset, LSA-Net not only has a faster running speed but also has a higher detection IoU value. The reason for this result is that the defect input into the segmentation network was incomplete due to inaccurate defect localization or localization failure in the localization network, as shown in Fig. 14.

Feature difference loss analysis: Notably, we focused on the impact of the feature difference loss on the segmentation results. To verify the effectiveness of the feature difference loss, we conducted the following experiments. First, we examined the influence of $loss_{diff}$ with different weights on the segmentation results. The overall loss of the network was calculated as shown in Eq. (5). The detection

Table 7
Detailed performance of our method under deep supervision with different weights or different settings on NEU-SEG dataset.

Loss	λ	μ	ν	IoU
$loss = \lambda loss_{seg} + \mu loss_{loc} + \nu loss_{diff}$	1.4	0.6	0	0.7405
	1.4	0.4	0.2	0.7595
	1.4	0.4	0.4	**0.7617**
	1.4	0.4	0.6	0.7598
$loss = \lambda loss_{seg} + \mu loss_{loc} + \nu loss_{shp}$	1.4	0.4	0.2	0.7478

Table 8
Detailed performance of our method in the MCSD-C Dataset with different settings.

Methods	Backbone	IoU
Base	U-Net [10]	0.7171
Base + LAM	U-Net [10]	0.7324
Base + LAM+SDM	U-Net [10]	0.7450
Base + LAM+SDM+FEM	U-Net [10]	0.7494

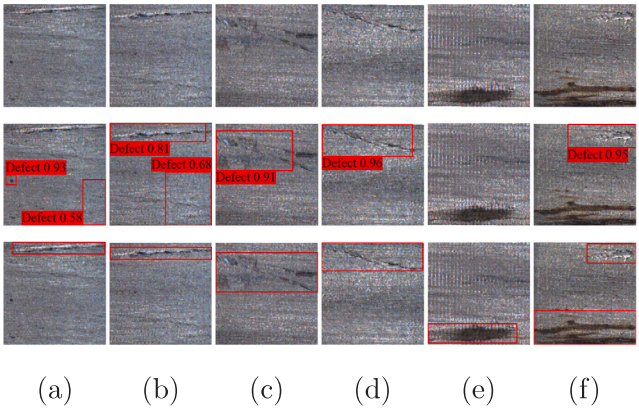

Fig. 14. Defect location results in [26]. The first row is the original image, the second row is the defect location result, and the third row is the label.

results for different weights (ν) are given in Table 7. Table 7 shows that compared with the results without $loss_{diff}$, the IoU values of $loss_{diff}$ with different weights were improved, indicating that $loss_{diff}$ was effective. Then, using shape labels to constrain the SPPM and LSPM, as shown in Fig. 15, we compared the segmentation result with the $loss_{diff}$ constraint to verify the effect of $loss_{diff}$. The loss with shape label constraint is denoted as $loss_{shp}$. Features V_{11} and V_{21} were obtained by dimensionality reduction of features V_1 and V_2 using a 1×1 convolution. According to the results in Table 7, the IoU value was higher under the $loss_{diff}$ constraint compared to using the $loss_{shp}$, which also proves the effectiveness of $loss_{diff}$.

5. Ablative study

To evaluate each submodule and loss function in the proposed LSA-Net, a series of ablation experiments were designed to verify the effects of the LAM, SDM, FEM and deep supervision.

Ablation studies for the LAM: In the proposed LSA-Net, the LAM was used to acquire the location features of defects and reduce the influence of the background. As shown in Table 8, the LAM yielded a result from 0.7171 to 0.7324 in terms of the IoU, which demonstrates the effectiveness of the LAM in our method.

Effects of the SDM: The SDM was used to further detect the specific shape of the defect on the basis of the LAM. According to the data in Table 8, after adding the SDM, the IoU also increased from 0.7324 to 0.7450. To verify the influence of different pooling kernel sizes in the LSPM module on the defect detection results, we obtained the data shown in Table 9 by adjusting the size of the strip pooling kernels in

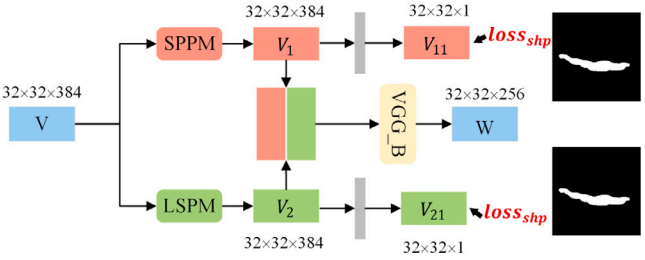

Fig. 15. SDM architecture with shape label constraints.

Table 9
Detailed performance of our method with different local strip pooling kernel sizes in LSPM.

Dataset	Kernel_size(LSPM)	IoU
MCSD-C	(1, w), (h, 1)	0.7470
MCSD-C	(1, w/2), (h/2, 1)	0.7494
NEU-Seg	(1, w), (h, 1)	0.7581
NEU-Seg	(1, w/2), (h/2, 1)	0.7594

Table 10
Detailed performance of our method with different deep-supervision.

Dataset	Methods	IoU
MCSD-C	LSA-Net ($loss_{seg}$)	0.7243
MCSD-C	LSA-Net ($loss_{seg} + loss_{loc}$)	0.7397
MCSD-C	LSA-Net ($loss_{seg} + loss_{loc} + loss_{diff}$)	0.7494
NEU-Seg	LSA-Net ($loss_{seg}$)	0.7359
NEU-Seg	LSA-Net ($loss_{seg} + loss_{loc}$)	0.7405
NEU-Seg	LSA-Net ($loss_{seg} + loss_{loc} + loss_{diff}$)	0.7594

the LSPM module. The results show that the detection of the model can be enhanced by appropriately reducing the pooling kernel size.

The effects of the FEM: To further refine the extracted features, the FEM was proposed. With the FEM, the model segmentation IoU was further improved. The results in Table 8 show that the adopted model was useful.

The effects of deep supervision: We further explored the effects of deep supervision in each module, and the quantitative results are shown in Table 10. Clearly, deep supervision in each submodule played an important role in improving the accuracy of model segmentation.

6. Conclusion

For small-sized industrial defect images, due to the relatively concentrated distribution of defects, it is time-consuming and laborious to first use the positioning network to locate the defect location and then use the segmentation network to segment the defects. To integrate the location information of defects into the defect segmentation network to achieve end-to-end detection and meet the requirements of production line cycle time, we propose a defect segmentation network that focuses on defect location and shape information. The network projects shallow features through strip pooling and then reorganizes them to obtain defect location information in a single network, which is a highlight of this paper. At the same time, another point of innovation in this paper is to design a loss function for maintaining the difference in the extracted features of different submodules, namely $loss_{diff}$. This loss function effectively prevents the features extracted by different submodules from tending to be similar during the training process, effectively enhances the diversity of features, and ensures the adaptability of the shape detection module in the network to different shape defects. In the end, this paper successfully combined the advantages of two-stage detection with the end-to-end detection model and successfully fused the location information and shape information of defects under the condition of meeting the production line cycle time requirements. As a result, the proposed method improves the segmentation IoU value by 4.5% compared to the baseline method UNet in the self-owned dataset (MCSD-C). In the three public datasets KSDD, MT and NEU-Seg, the improvements were 2.66%, 4.11%, and 4.3%, respectively. Compared with the two-stage detection method (APP-UNet), the overall detection time is shortened by 70.36% due to saving the calculation amount required for repeated encoding in the two-stage detection and the proposals and nonmaximum suppression processing required to obtain the target box. In the future, the application of $loss_{diff}$ in other fields, such as image classification and target recognition, will be explored to determine if $loss_{diff}$ enhances the network detection ability in other fields.

Ethical approval

We confirm that the content of the manuscript has not been published or submitted for publication elsewhere. All authors have seen the manuscript and approved it for submission to the journal.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] Song G, Song K, Yan Y. EDRNet: Encoder–decoder residual network for salient object detection of strip steel surface defects. IEEE Trans Instrum Meas 2020;69(12):9709–19. http://dx.doi.org/10.1109/TIM.2020.3002277.
[2] Póka G, Balázs BZ. A robust digital image processing method for measuring the planar burr length at milling. J Manuf Process 2022;80:706–17.
[3] Li X, Gao B, Woo WL, Tian GY, Qiu X, Gu L. Quantitative surface crack evaluation based on eddy current pulsed thermography. IEEE Sens J 2017;17(2):412–21. http://dx.doi.org/10.1109/JSEN.2016.2625815.
[4] Dhal KG, Das A, Ray S, Gálvez J. Randomly attracted rough firefly algorithm for histogram based fuzzy image clustering. Knowl-Based Syst 2021;216:106814.
[5] Zhou Z, Rahman Siddiquee MM, Tajbakhsh N, Liang J. Unet++: A nested u-net architecture for medical image segmentation. In: Deep learning in medical image analysis and multimodal learning for clinical decision support. Springer; 2018, p. 3–11.
[6] Yu R, Kershaw J, Wang P, Zhang Y. Real-time recognition of arc weld pool using image segmentation network. J Manuf Process 2021;72:159–67.
[7] Choi J, Chun D, Kim H, Lee H-J. Gaussian yolov3: An accurate and fast object detector using localization uncertainty for autonomous driving. In: Proceedings of the IEEE/CVF international conference on computer vision. 2019, p. 502–11.
[8] Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu C-Y, et al. Ssd: Single shot multibox detector. In: European conference on computer vision. Springer; 2016, p. 21–37.
[9] Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation. IEEE Trans Pattern Anal Mach Intell 2015;39(4):640–51.
[10] Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation. In: International conference on medical image computing and computer-assisted intervention. Springer; 2015, p. 234–41.
[11] Lin G, Milan A, Shen C, Reid I. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2017, p. 1925–34.
[12] Hou Q, Zhang L, Cheng M-M, Feng J. Strip pooling: Rethinking spatial pooling for scene parsing. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020, p. 4003–12.
[13] Fu J, Liu J, Tian H, Li Y, Bao Y, Fang Z, et al. Dual attention network for scene segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019, p. 3146–54.
[14] Wen X, Shan J, He Y, Song K. Steel surface defect recognition: A survey. Coatings 2022;13(1):17.
[15] Chen Y, Ding Y, Zhao F, Zhang E, Wu Z, Shao L. Surface defect detection methods for industrial products: A review. Appl Sci 2021;11(16):7657.
[16] Liu Y, Xu K, Xu J. An improved MB-LBP defect recognition approach for the surface of steel plates. Appl Sci 2019;9(20):4222.
[17] Yang C, Liu P, Yin G, Wang L. Crack detection in magnetic tile images using nonsubsampled shearlet transform and envelope gray level gradient. Opt Laser Technol 2017;90:7–17.
[18] Hu G-H. Automated defect detection in textured surfaces using optimal elliptical Gabor filters. Optik 2015;126(14):1331–40.

[19] Li L, Zhang X, Xiao H, Xu M. Segmentation of non-stochastic surfaces based on non-subsampled contourlet transform and mathematical morphologies. Measurement 2016;79:137–46.
[20] Wang H, Zhang J, Tian Y, Chen H, Sun H, Liu K. A simple guidance template-based defect detection method for strip steel surfaces. IEEE Trans Ind Inf 2018;15(5):2798–809.
[21] Chen B, Fang Z, Xia Y, Zhang L, Huang Y, Wang L. Accurate defect detection via sparsity reconstruction for weld radiographs. NDT & E Int. 2018;94:62–9.
[22] Ai Y-h, Ke X. Surface detection of continuous casting slabs based on curvelet transform and kernel locality preserving projections. J Iron Steel Res, Int 2013;20(5):80–6.
[23] Wang J, Fu P, Gao RX. Machine vision intelligence for product defect inspection based on deep learning and Hough transform. J Manuf Syst 2019;51:52–60.
[24] Liu K, Wang H, Chen H, Qu E, Tian Y, Sun H. Steel surface defect detection using a new Haar–Weibull-variance model in unsupervised manner. IEEE Trans Instrum Meas 2017;66(10):2585–96.
[25] Dong H, Song K, He Y, Xu J, Yan Y, Meng Q. PGA-Net: Pyramid feature fusion and global context attention network for automated surface defect detection. IEEE Trans Ind Inf 2019;16(12):7448–58.
[26] Fu X, Li K, Liu J, Li K, Zeng Z, Chen C. A two-stage attention aware method for train bearing shed oil inspection based on convolutional neural networks. Neurocomputing 2020;380:212–24.
[27] Fang F, Li L, Gu Y, Zhu H, Lim J-H. A novel hybrid approach for crack detection. Pattern Recognit 2020;107:107474.
[28] Ren S, He K, Girshick R, Sun J. Faster r-cnn: Towards real-time object detection with region proposal networks. Adv Neural Inf Process Syst 2015;28.
[29] Li B, Wang T, Hu Z, Yuan C, Zhai Y. Two-level model for detecting substation defects from infrared images. Sensors 2022;22(18):6861.
[30] Ni X, Ma Z, Liu J, Shi B, Liu H. Attention network for rail surface defect detection via consistency of intersection-over-union (IoU)-guided center-point estimation. IEEE Trans Ind Inf 2021;18(3):1694–705.
[31] Zhao Y, Li P, Gao C, Liu Y, Chen Q, Yang F, et al. TSASNet: Tooth segmentation on dental panoramic X-ray images by Two-Stage Attention Segmentation Network. Knowl-Based Syst 2020;206:106338.
[32] Ibrokhimov B, Kang J-Y. Two-stage deep learning method for breast cancer detection using high-resolution mammogram images. Appl Sci 2022;12(9):4616.
[33] Liu T, He Z. TAS2-Net: Triple-attention semantic segmentation network for small surface defect detection. IEEE Trans Instrum Meas 2022;71:1–12.
[34] Li Z, Wu C, Han Q, Hou M, Chen G, Weng T. CASI-Net: A novel and effect steel surface defect classification method based on coordinate attention and self-interaction mechanism. Mathematics 2022;10(6):963.
[35] Xie L, Xiang X, Xu H, Wang L, Lin L, Yin G. FFCNN: A deep neural network for surface defect detection of magnetic tile. IEEE Trans Ind Electron 2020;68(4):3506–16.
[36] Liu J, Guo F, Gao H, Li M, Zhang Y, Zhou H. Defect detection of injection molding products on small datasets using transfer learning. J Manuf Process 2021;70:400–13.
[37] Li H, Xiong P, An J, Wang L. Pyramid attention network for semantic segmentation. 2018, arXiv preprint arXiv:1805.10180.
[38] Yang L, Fan J, Huo B, Li E, Liu Y. A nondestructive automatic defect detection method with pixelwise segmentation. Knowl-Based Syst 2022;242:108338.
[39] Wang Q, Yang R, Wu C, Liu Y. An effective defect detection method based on improved Generative Adversarial Networks (iGAN) for machined surfaces. J Manuf Process 2021;65:373–81.
[40] He K, Gkioxari G, Dollár P, Girshick R. Mask r-cnn. In: Proceedings of the IEEE international conference on computer vision. 2017, p. 2961–9.
[41] Huang Z, Huang L, Gong Y, Huang C, Wang X. Mask scoring r-cnn. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019, p. 6409–18.
[42] Zhao H, Shi J, Qi X, Wang X, Jia J. Pyramid scene parsing network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2017, p. 2881–90.
[43] Zhao H, Gallo O, Frosio I, Kautz J. Loss functions for image restoration with neural networks. IEEE Trans Comput Imag 2016;3(1):47–57.
[44] Yu R, Guo B, Yang K. Selective prototype network for few-shot metal surface defect segmentation. IEEE Trans Instrum Meas 2022;71:1–10.
[45] Chen L-C, Papandreou G, Schroff F, Adam H. Rethinking atrous convolution for semantic image segmentation. 2017, arXiv preprint arXiv:1706.05587.
[46] Fan M, Lai S, Huang J, Wei X, Chai Z, Luo J, et al. Rethinking bisenet for real-time semantic segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021, p. 9716–25.
[47] Yang L, Fan J, Huo B, Li E, Liu Y. A nondestructive automatic defect detection method with pixelwise segmentation. Knowl-Based Syst 2022;242:108338.
[48] Deng R, Shen C, Liu S, Wang H, Liu X. Learning to predict crisp boundaries. In: Proceedings of the european conference on computer vision. ECCV, 2018, p. 562–78.
[49] Božič J, Tabernik D, Skočaj D. Mixed supervision for surface-defect detection: From weakly to fully supervised learning. Comput Ind 2021;129:103459.
[50] Song K, Yan Y. A noise robust method based on completed local binary patterns for hot-rolled steel strip surface defects. Appl Surf Sci 2013;285:858–64.
[51] Huang Y, Qiu C, Yuan K. Surface defect saliency of magnetic tile. Vis Comput 2020;36(1):85–96.