A Semantic Information Decomposition Network for Accurate Segmentation of Texture Defects

Hua Yang, *Member, IEEE*, Jiale Hu, Zhouping Yin, *Member, IEEE*, and Zhengjia Wang

***Abstract*—Defect detection on textured surfaces remains a challenging task due to the wide range of textures and defects. Current unsupervised learning-based texture defect detection methods based on texture background reconstruction cannot detect texture defects with high precision because it is difficult to guarantee a high-precision reconstruction of the texture background while suppressing the defect foreground. In this study, we propose a novel semantic information decomposition network (SIDN) for accurate texture defect segmentation. The SIDN is trained on artificial defective images produced by a defect generation module (DGM). First, the SIDN uses a feature extraction module (FEM) to extract latent features with both texture semantic information and defect semantic information. Then, a novel feature separation extraction module (FSEM) for decomposing the texture semantic information and defect semantic information from the feature map generated by the FEM is proposed, preventing the coupling of the texture and defect semantic information from affecting the final segmentation accuracy. Next, a novel global semantic relation module (GSRM) is proposed to determine the relevance of the global semantic information to comprehensively consider the context and improve the feature representation. Finally, a segmentation module (SM) that directly segments the textures and defects instead of reconstructing the texture background is proposed. The final detection result is obtained by calculating a weighted average of the texture and defect segmentation results. The extensive experimental tests with the most popular and most challenging texture defect dataset demonstrate that the SIDN achieves accurate segmentation of various texture defects without using real defect samples.**

***Index Terms*—Anomaly detection, defect segmentation, feature clustering, feature separation, texture defect detection.**

Manuscript received 12 June 2022; revised 24 September 2022; accepted 7 October 2022. Date of publication 28 October 2022; date of current version 20 June 2023. This work was supported in part by the project of Foshan Science and Technology Bureau under Grant 2020001006509, in part by the National Natural Science Foundation of China under Grant 51875228, and in part by National Key R&D Program under Grant 2020YFA0405700. Paper no. TII-22-2518. *(Corresponding author: Hua Yang.)*

Hua Yang, Jiale Hu, and Zhouping Yin are with the State Key Laboratory of Digital Manufacturing Equipment and Technology, School of Mechanical Science and Engineering, Huazhong University of Science and Technology, Wuhan 430074, China (e-mail: huayang@hust.edu.cn; m202070664@hust.edu.cn; yinzhp@mail.hust.edu.cn).

Zhengjia Wang is with the Hubei Key Laboratory of Modern Manufacturing Quantity Engineering, School of Mechanical Engineering, Hubei University of Technology, Wuhan 430068, China (e-mail: wangzhengjia@hbut.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TII.2022.3217751.

Digital Object Identifier 10.1109/TII.2022.3217751

I. Introduction

IN THE field of industrial automation, texture surface defects are common in various industrial products, such as fabric, wood, ceramic, and thin-film transistor liquid crystal displays, due to complex manufacturing processes. These defects typically appear as localized regions with defective texture structures and abnormal brightness variations both of which have a serious impact on the visual experience of the user. To ensure product quality, the detection of all types of texture defects has become an essential aspect of the manufacturing process.

Visual inspection technology is widely used to inspect texture defects because it is noncontact and flexible. However, the visual inspection of texture defects remains a challenging task due to the large changes in various textures and defects, such as irregular brightness variations, scale changes, low contrast, and an insufficient number of defective samples. To address these challenges, various texture defect inspection methods have been proposed in recent decades. These methods can be broadly classified into two main categories: 1) traditional methods; and 2) deep learning methods.

Traditional methods [1], [2], [3] typically utilize handcrafted features based on texture characteristics to detect texture defects [4]. Aiger and Talbot [1] proposed the phase-only transform method for inspecting defects on textured surfaces in which irregular patterns that represent defects are preserved using phase information in the frequency domain. Xie and Mirmehdi [3] proposed the texture exemplars (TEXEMSs) method, which uses a Gaussian mixture model to determine the feature distributions of local texture patches. Although these traditional methods can be used to identify defects in specific types of texture images, they cannot guarantee good performance for several textures simultaneously because the manually designed features do not represent all types of textures.

To date, deep learning-based texture defect inspection methods have been widely used due to their powerful feature extraction capabilities. Deep learning-based methods can be classified into two categories based on whether they use labeled defective samples for training: 1) supervised; and 2) unsupervised. Supervised methods [5] require a large number of labeled defective samples for training. Dong et al. [6] proposed a pyramid feature fusion and global context attention network to detect surface defects at the pixel level. However, it is difficult to collect and label texture defect samples in actual industrial manufacturing, which limits the practical application of supervised methods. In contrast to supervised methods, unsupervised methods [7], [8] require easily available defect-free samples for training; thus,

these methods show great potential for automated online defect inspection in industrial processes. Yang et al. [9] proposed an unsupervised anomaly feature editing-based adversarial network (AFEAN) for accurately identifying various texture defects. However, these unsupervised methods cannot achieve good performance for irregular texture surfaces, since they do not reconstruct the texture accurately and do not suppress the generation of defects during testing.

In this article, we propose a novel semantic information decomposition network (SIDN) to accurately segment various texture defects based on the hypothesis that the semantic information of defective images can be decomposed into the semantic information of the texture background and the semantic information of the defect foreground. The SIDN is trained with artificial defective images generated by the defect generation module (DGM). First, the proposed SIDN utilizes the feature extraction module (FEM) to extract latent features with both texture semantic information and defect semantic information. Subsequently, to prevent the combined texture and defect semantic information from influencing the final segmentation accuracy, the novel feature separation extraction module (FSEM) is proposed to decompose the texture semantic features and defect semantic features from the shared feature map generated by the FEM. Next, the novel global semantic relation module (GSRM) is proposed to determine the relevance of the global semantic information and improve the feature representation. Finally, the segmentation module (SM) is proposed to directly segment the textures and defects instead of reconstructing the entire texture background. The final result is generated by calculating the weighted average of the texture segmentation result and defect segmentation result. Experiments with the new mainstream texture defect dataset MVTec AD [10] demonstrate that the SIDN can achieve accurate segmentation in texture surface defect inspections without using real defect samples.

This study has the following three major contributions.

1) A novel SIDN for identifying texture surface defects with state-of-the-art performance is proposed.
2) A novel FSEM for decomposing the texture semantic information and defect semantic information is proposed, preventing the combined texture and defect semantic information from affecting the final segmentation accuracy.
3) A novel GSRM is proposed to determine the relevance of the global semantic information, allowing the context to be comprehensively considered and improving the feature representation.

The rest of this article is organized as follows. In Section II, related works on unsupervised texture defect inspection methods are introduced. In Section III, the proposed SIDN is discussed in detail. In Section IV, a set of experiments that demonstrate the performance of the SIDN is presented. Finally, Section V concludes this article.

II. Related Works

Unsupervised learning-based methods for identifying texture defects are based on the assumption that a model trained only on defect-free samples cannot accurately reconstruct defective samples, resulting in large reconstruction errors in defect regions. Over the last two decades, these methods have been widely studied because they use defect-free samples. Unsupervised learning-based methods can be divided into two main categories: 1) autoencoder (AE)-based methods; and 2) generative adversarial network (GAN)-based methods.

AE-based methods [11], [12], [13], [14] extract the encoding features of the latent space from the original data and reconstruct the data with these features. Gong et al. proposed a memory module to improve the AE. Chow et al. used convolutional AE to detect defects on concrete structures. Yang et al. [8] proposed the multiscale feature clustering-based fully convolutional AE (MS-FCAE), which uses feature clustering to improve the texture reconstruction accuracy. Recently, Yang et al. [9] proposed an unsupervised AFEAN that detects anomalous features by learning the distribution of the latent features with a central-constraint-based clustering method. These AE-based methods usually minimize the squared Euclidean loss between the original and reconstructed images to train the model, resulting in blurry reconstructed images because the structural information of the images is overlooked.

GAN-based methods [15], [16], [17], [18] generate high-quality images with the min–max adversarial learning mechanism. Perera et al. [15] proposed one-class novelty detection using gans (OCGAN), which learns the latent representation of in-class samples for anomaly detection. Hu et al. [17] presented a novel unsupervised method for automatically detecting defects in fabrics based on a deep convolutional GAN. Schlegl et al. [18] proposed unsupervised anomaly detection with generative adversarial networks (AnoGAN), which detects defects by reproducing a given image patch after learning the distribution of the defect-free texture image patches with a GAN. The adversarial training mechanism of GANs can be used to train an AE model capable of reconstructing high-quality images. However, convergence is difficult while training GAN models.

Overall, to achieve good performance, these texture reconstruction methods must have small residuals in nondefective regions and large residuals in defective regions in the residual images of the original and reconstructed images. However, with these methods, it remains challenging to ensure high-precision reconstruction of the texture background while suppressing the defect foreground.

In this article, to address the abovementioned issues associated with current unsupervised learning methods, the novel SIDN is proposed, which directly segments the textures and defects instead of reconstructing the texture background. The SIDN extracts the texture semantic information and defect semantic information separately with the FSEM. Then, the GSRM is proposed to determine the relevance of the global semantic information to improve feature representation. Finally, the SM utilizes the features extracted by the GSRM to segment the textures and defects. Therefore, the SIDN can accurately segment various texture defects by combining the texture and defect segmentation results.

III. Proposed SIDN Method

In this section, the proposed SIDN is introduced in detail. First, the overall architecture of the SIDN is briefly introduced. Then, the five modules of the SIDN: 1) the DGM; 2) FEM;

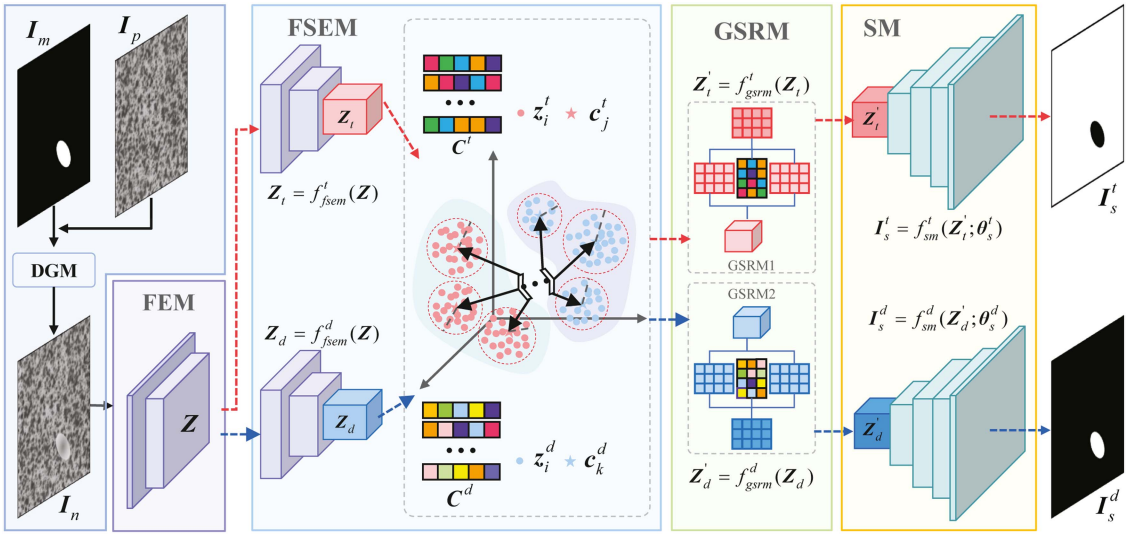

Fig. 1. Overall architecture of the proposed SIDN method. The SIDN includes a DGM, FEM, FSEM, GSRM, and SM.

3) SM; 4) FSEM; and 5) GSRM, are presented in detail. Furthermore, the training and inference procedures are introduced. Finally, the parameter setup of the SIDN is discussed.

A. SIDN Network Architecture

The key to unsupervised learning-based methods for defect detection is to produce a high-quality texture background image with the defects removed. However, current unsupervised learning methods cannot ensure high-precision texture background reconstruction and maximum defect foreground suppression at the same time, resulting in poor performance. To address this problem, based on the hypothesis that the semantic information of defective images can be decomposed into the semantic information of the texture background and the semantic information of the defect foreground, a novel SIDN for accurate segmentation of various texture defects is proposed.

Fig. 1 shows the overall architecture of the SIDN. The SIDN includes five components: 1) DGM; 2) FEM; 3) FSEM; 4) GSRM; and 5) SM.

As shown in Fig. 1, the SIDN is trained with artificial defective images $\boldsymbol{I}_n$. The artificial defective images are not true defective samples; instead, they are generated by combining defect-free images $\boldsymbol{I}_p$ and random masks $\boldsymbol{I}_m$ with the DGM. First, the FEM extracts representative features $\boldsymbol{Z}$ of the input to generate a shared feature map for the subsequent network. Then, a novel FSEM is used to decompose the texture semantic features $\boldsymbol{Z}_t$ and defect semantic features $\boldsymbol{Z}_d$ in the shared feature map $\boldsymbol{Z}$ generated by the FEM, which prevents the coupling of the texture and the defect semantic information from affecting the final segmentation accuracy. Subsequently, a novel GSRM is used to determine the relevant global semantic information, allowing the context to be comprehensively considered and thus improving the feature representation. Finally, an SM utilizes the features $\boldsymbol{Z}'_t$ and $\boldsymbol{Z}'_d$ enhanced by the GSRM to directly segment the textures and defects instead of reconstructing the texture background. The final inspection result is obtained by calculated the weighted average of the texture segmentation result $\boldsymbol{I}^t_s$ and defect segmentation result $\boldsymbol{I}^d_s$. The SIDN is trained in an end-to-end manner with a joint loss function composed of two losses: 1) the feature separation loss; and 2) the segmentation loss.

Due to the feature separation extraction function of the FSEM and the feature enhancement function of the GSRM, the SIDN can accurately segment the textures and defects instead of reconstructing the texture background, which is essential for texture defect inspection.

B. DGM, FEM, and SM

1) Defect Generation Module: It requires defective samples to train the SIDN. Because it is difficult to collect and label texture defect samples in actual industrial manufacturing, the defective samples $\boldsymbol{I}_n$ are generated by combining defect-free images $\boldsymbol{I}_p$ with random anomaly masks $\boldsymbol{I}_m$ through the DGM.

$$\boldsymbol{I}_n(x,y) = (1 - \boldsymbol{I}_m(x,y)) \cdot \boldsymbol{I}_p(x,y) + \boldsymbol{I}_m(x,y) \cdot \boldsymbol{I}_g(x,y) \tag{1}$$

where $\boldsymbol{I}_n, \boldsymbol{I}_p, \boldsymbol{I}_m, \boldsymbol{I}_g \in R^{W\times H\times 1}$ and W and H denote the width and height of the images, respectively, which were set to 256 in this study. In addition, $x = 1, \ldots, W, y = 1, \ldots, H$, and $\boldsymbol{I}_g$ denotes a noisy image generated by random Gaussian sampling.

2) Feature Extraction Module: In this study, the FEM utilizes two convolutional blocks to extract the latent features $\boldsymbol{Z}$ of the input $\boldsymbol{I}_n$.

$$\boldsymbol{Z} = f_{\text{fem}}(\boldsymbol{I}_n; \boldsymbol{\theta}_e) \tag{2}$$

where $\boldsymbol{Z} \in R^{\frac{W}{2}\times\frac{H}{2}\times 2C}$ and $f_{\text{fem}}(\cdot)$ and $\boldsymbol{\theta}_e$ denote the function and parameters of the FEM, respectively. All convolution kernels had a size of 3×3.

a) The first convolutional block consisted of a convolutional layer and a residual block, with a stride and channel number of 1 and C ($C = 16$ in this study), respectively.

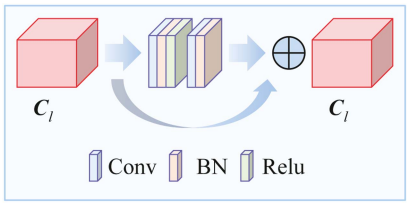

Fig. 2. Structure details of the residual block of the FEM.

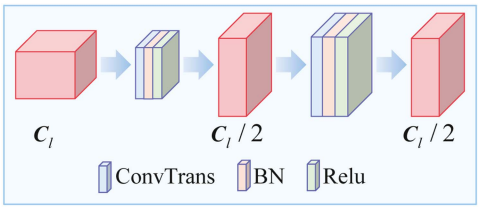

Fig. 3. Structure details of the deconvolution block of the SM.

b) The second convolutional block consisted of two residual blocks, where the stride and channel number of the first residual block were 2 and $2C$ and the stride and channel number of the second residual block were 1 and $2C$, respectively.

The structure of the residual block is shown in Fig. 2. The feature map $\boldsymbol{Z}$ contains texture semantic features and defect semantic features, which were used as the shared feature map in the subsequent network.

3) Segmentation Module: The SM uses the texture semantic features $\boldsymbol{Z}'_t$ and defect semantic features $\boldsymbol{Z}'_d$ extracted from the previous network to segment the textures and defects, respectively; thus, generating the following texture segmentation result $\boldsymbol{I}_s^t$ and defect segmentation result $\boldsymbol{I}_s^d$:

$$\boldsymbol{I}_s^t = f_{\text{sm}}^t(\boldsymbol{Z}'_t; \boldsymbol{\theta}_s^t) \tag{3}$$

$$\boldsymbol{I}_s^d = f_{\text{sm}}^d(\boldsymbol{Z}'_d; \boldsymbol{\theta}_s^d) \tag{4}$$

where $\boldsymbol{Z}'_t, \boldsymbol{Z}'_d \in R^{\frac{W}{16}\times\frac{H}{16}\times 16C}$, $f_{\text{sm}}^t(\cdot)$ and $f_{\text{sm}}^d(\cdot)$ denote the functions of the SM, and $\boldsymbol{\theta}_s^t$ and $\boldsymbol{\theta}_s^d$ denote the parameters of the SM. The SM contains four deconvolution blocks, a deconvolution layer, and a Sigmoid activation layer. The structure of the deconvolution block is shown in Fig. 3. Since the defect area is significantly smaller than the nondefect area, to prevent any adverse effects due to category imbalance, the pixel-level weighted cross-entropy loss is used to train the SM.

$$\begin{aligned} L_s = &\mathbb{E}_{\boldsymbol{I}_n}\left[\left\|\lambda_1 \cdot (1-\boldsymbol{I}_m)\odot \log(\boldsymbol{I}_s^t) + \boldsymbol{I}_m \odot \log(1-\boldsymbol{I}_s^t)\right\|_1\right] \\ &+\mathbb{E}_{\boldsymbol{I}_n}\left[\left\|\lambda_1 \cdot (1-\boldsymbol{I}_m)\odot \log(1-\boldsymbol{I}_s^d) + \boldsymbol{I}_m \odot \log(\boldsymbol{I}_s^d)\right\|_1\right] \end{aligned} \tag{5}$$

where λ_1 is the weight of loss L_s and $\mathbb{E}$, $\odot$, and $\|\cdot\|_1$ denote the expectation calculation, Hadamard product, and L1 norm, respectively.

C. Feature Separation Extraction Module

Current unsupervised learning methods cannot guarantee high-precision texture background reconstruction while suppressing the defect foreground, resulting in poor defect detection performance. To address this issue, the proposed SIDN directly segments the textures and defects instead of reconstructing the texture background. Since the coupling of the texture and defect semantic features can affect the final segmentation accuracy, an FSEM is proposed to learn the intrinsic feature representations in latent space. The goal of the FSEM is to increase the difference between the feature distributions of the textures and defects to allow the FSEM to effectively decompose the texture semantic features $\boldsymbol{Z}_t$ and defect semantic features $\boldsymbol{Z}_d$ from $\boldsymbol{Z}$.

$$\boldsymbol{Z}_t = f_{\text{fsem}}^t\left(\boldsymbol{Z}; \boldsymbol{\theta}_f^t\right) \tag{6}$$

$$\boldsymbol{Z}_d = f_{\text{fsem}}^d\left(\boldsymbol{Z}; \boldsymbol{\theta}_f^d\right). \tag{7}$$

The latent features $\boldsymbol{Z}_t$ and $\boldsymbol{Z}_d$ can be viewed as a set of local features; that is, $\boldsymbol{Z}_t = \{\boldsymbol{z}_1^t, \boldsymbol{z}_2^t, \ldots, \boldsymbol{z}_N^t\}$ ($\boldsymbol{z}_i^t \in R^{16C\times 1}, i = 1, \ldots, \frac{W}{16}\times\frac{H}{16}$) and $\boldsymbol{Z}_d = \{\boldsymbol{z}_1^d, \boldsymbol{z}_2^d, \ldots, \boldsymbol{z}_N^d\}$ ($\boldsymbol{z}_i^d \in R^{16C\times 1}, i = 1, \ldots, \frac{W}{16}\times\frac{H}{16}$). For preliminary feature separation, it is necessary to increase the difference between $\boldsymbol{Z}_t$ and $\boldsymbol{Z}_d$ at the corresponding positions as much as possible.

$$L_z = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{z}_i^t \cdot \boldsymbol{z}_i^d. \tag{8}$$

For additional feature separation, a novel central constraint model based on multiple hyperspheres is proposed. The different types of local features, $\boldsymbol{Z}_t$ and $\boldsymbol{Z}_d$, should be clustered into K classes to learn the inherent textons of the textures and defects through feature clustering [19]. Let $\boldsymbol{C}^t = \{\boldsymbol{c}_1^t, \boldsymbol{c}_2^t, \ldots, \boldsymbol{c}_K^t\}$ and $\boldsymbol{C}^d = \{\boldsymbol{c}_1^d, \boldsymbol{c}_2^d, \ldots, \boldsymbol{c}_K^d\}$ denote the cluster centers ($\boldsymbol{c}_k^t, \boldsymbol{c}_k^d \in R^{16C\times 1}$), which also form the centers of the hyperspheres. The residuals between the latent features and the cluster centers can be calculated as

$$\boldsymbol{e}_{ij}^t = \boldsymbol{z}_i^t - \boldsymbol{c}_j^t \tag{9}$$

$$\boldsymbol{e}_{ik}^d = \boldsymbol{z}_i^d - \boldsymbol{c}_k^d \tag{10}$$

where $i = 1, \ldots, N$, $j, k = 1, \ldots, K$, and N is the number of latent features. Then, a score that measures the similarity between the embedded features $\boldsymbol{z}_i^t$, $\boldsymbol{z}_i^d$ and the centroid $\boldsymbol{c}_j^t$, $\boldsymbol{c}_k^d$ can be calculated as

$$\boldsymbol{Q}_{ij}^t = \frac{\left(1+\left\|\boldsymbol{e}_{ij}^t\right\|^2/\alpha\right)^{-\frac{\alpha+1}{2}}}{\sum_{j'=1}^{K}\left(1+\left\|\boldsymbol{e}_{ij'}^t\right\|^2/\alpha\right)^{-\frac{\alpha+1}{2}}} \tag{11}$$

$$\boldsymbol{Q}_{ik}^d = \frac{\left(1+\left\|\boldsymbol{e}_{ik}^d\right\|^2/\alpha\right)^{-\frac{\alpha+1}{2}}}{\sum_{k'=1}^{K}\left(1+\left\|\boldsymbol{e}_{ik'}^d\right\|^2/\alpha\right)^{-\frac{\alpha+1}{2}}} \tag{12}$$

where α is set to 1. Each latent feature $\boldsymbol{z}_i$ has a distance of d_{ki} to $\boldsymbol{c}_k$, with the minimum distance indicating which center the feature belongs to

$$\boldsymbol{d}_{ki} = \min_k \|\boldsymbol{e}_{ik}\|_2. \tag{13}$$

The radius $\boldsymbol{r}_k$ of each hypersphere is calculated as follows:

$$\boldsymbol{r}_k = \frac{1}{N_k}\sum_i \boldsymbol{d}_{ki} + 3\sigma_{\boldsymbol{d}_k} \tag{14}$$

where $\sigma_{\boldsymbol{d}_k}$ is the standard deviation of $\boldsymbol{d}_k$. To ensure that the original distribution approaches the target distribution, an auxiliary score is proposed.

$$\boldsymbol{P}_{ij}^t = \frac{\boldsymbol{Q}_{ij}^{t\ 2}/\boldsymbol{f}_j^t}{\sum_{j'=1}^K \boldsymbol{Q}_{ij'}^{t\ 2}/\boldsymbol{f}_{j'}^t} \tag{15}$$

$$\boldsymbol{P}_{ik}^d = \frac{\boldsymbol{Q}_{ik}^{d\ 2}/\boldsymbol{f}_k^d}{\sum_{k'=1}^K \boldsymbol{Q}_{ij'}^{d\ 2}/\boldsymbol{f}_{k'}^d} \tag{16}$$

where $\boldsymbol{f}_j^t = \sum_{i=1}^N \boldsymbol{Q}_{ij}^t$ and $\boldsymbol{f}_k^d = \sum_{i=1}^N \boldsymbol{Q}_{ik}^d$ are the soft cluster frequencies. The Kullback–Leibler (KL) divergence is used to minimize the difference between the distance scores and the auxiliary target scores as

$$\begin{aligned} L_{\text{kl}} &= \text{KL}\left(\boldsymbol{P}_{ij}^t \parallel \boldsymbol{Q}_{ij}^t\right) + \text{KL}\left(\boldsymbol{P}_{ik}^d \parallel \boldsymbol{Q}_{ik}^d\right) \\ &= \sum_{i=1}^N \sum_{j=1}^K \boldsymbol{P}_{ij}^t \log \frac{\boldsymbol{P}_{ij}^t}{\boldsymbol{Q}_{ij}^t} + \sum_{i=1}^N \sum_{k=1}^K \boldsymbol{P}_{ik}^d \log \frac{\boldsymbol{P}_{ik}^d}{\boldsymbol{Q}_{ik}^d}. \end{aligned} \tag{17}$$

To reduce the interclass similarity between the textures and defects, hence achieving the goal of feature decomposition, a novel central constraint based on multiple hyperspheres is proposed.

$$L_{td} = \frac{1}{K \times K} \sum_{j=1}^K \sum_{k=1}^K \frac{\boldsymbol{c}_j^t \cdot \boldsymbol{c}_k^d}{\left\|\boldsymbol{c}_j^t\right\|_2 \cdot \left\|\boldsymbol{c}_k^d\right\|_2}. \tag{18}$$

The parameters of the FSEM are updated based on the loss as

$$L_{fd} = \lambda_2 L_z + \lambda_3 L_{\text{kl}} + \lambda_4 L_{td}. \tag{19}$$

With the auxiliary score and the central constraint, the FSEM can decompose the texture semantic information and defect semantic information from the shared feature map $\boldsymbol{Z}$, improving the accuracy of the subsequent segmentation.

D. Global Semantic Relation Module

Conventional convolutional networks are inherently constrained to local receptive fields that provide only short-range contextual information. This limitation on the contextual information has a considerable negative effect on the performance. With a self-attention mechanism [20], a single feature at any position can perceive features at all other positions, allowing the contextual information of the full image to be obtained.

$$\text{Attention}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \text{soft}\max\left(\frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{d_k}}\right)\boldsymbol{V}. \tag{20}$$

However, this mechanism requires large attention maps to be generated to assess the relationships between each pair of pixel, leading to a very high complexity of $O(N^2)$ in both time and space, where N is the number of input features. Since the matrices $\boldsymbol{Q}$ and $\boldsymbol{K}$ have very large dimensions, self-attention mechanism-based methods often have high computational complexities and consume a large amount of graphics processing unit (GPU) memory. To address this issue, the novel GSRM is proposed.

To model the dependencies of the local feature representations in an entire image with a low computational complexity and low memory use, we introduce a novel GSRM as

$$\boldsymbol{Z}_t' = f_{\text{gsrm}}^t\left(\boldsymbol{Z}_t; \boldsymbol{\theta}_g^t\right) \tag{21}$$

$$\boldsymbol{Z}_d' = f_{\text{gsrm}}^d\left(\boldsymbol{Z}_d; \boldsymbol{\theta}_g^d\right) \tag{22}$$

where $\boldsymbol{\theta}_g^t$ and $\boldsymbol{\theta}_g^d$ denote the parameters of the GSRM. Our motivation is to replace the large matrix $\boldsymbol{Q}$ in the self-attention mechanism with a smaller matrix $\boldsymbol{C}$ generated by the FSEM. As shown in Fig. 4, given a local feature map $\boldsymbol{Z}_t$ (or $\boldsymbol{Z}_d$) $\in R^{\frac{W}{16} \times \frac{H}{16} \times 16C}$, the module first applies two convolutional layers with 1×1 filters on $\boldsymbol{Z}_t$ (or $\boldsymbol{Z}_d$) to generate two feature maps: 1) $\boldsymbol{K}$; and 2) $\boldsymbol{V}$, where $\boldsymbol{K}, \boldsymbol{V} \in R^{\frac{W}{16} \times \frac{H}{16} \times 16C}$. After $\boldsymbol{K}$ and $\boldsymbol{V}$ are obtained, we generate an attention map $\boldsymbol{M} \in R^{(\frac{W}{16} \times \frac{H}{16}) \times K}$. Then, the contextual information is collected by an aggregation operation on $\boldsymbol{C}$ and $\boldsymbol{M}$. Finally, the contextual information is added to the local feature map $\boldsymbol{Z}_t$ (or $\boldsymbol{Z}_d$) to improve the pixel-wise representation, yielding $\boldsymbol{Z}_t'$(or $\boldsymbol{Z}_d'$). Therefore, the complexity can be reduced from $O(N^2)$ to $O(N \times K)$, which is approximately equal to $O(N)$ since K is much smaller than N. The process is as follows:

$$\boldsymbol{M} = \text{soft}\max\left(\boldsymbol{K}'\boldsymbol{C}^T\right) \tag{23}$$

$$\boldsymbol{Z}_t' = \boldsymbol{V} + \text{reshape}\left(\boldsymbol{M}\boldsymbol{C}\right) \tag{24}$$

where $\boldsymbol{K}$ is reshaped to $\boldsymbol{K}' \in R^{(\frac{W}{16} \times \frac{H}{16}) \times 16C}$ and $\boldsymbol{C}$ denotes the $\boldsymbol{C}^t$ or $\boldsymbol{C}^d$ in the FSEM. Table I gives the comparison of the computational effort of the traditional self-attentive mechanism and GSRM.

TABLE I
COMPARISON OF THE COMPUTATIONAL EFFORT

	Number of multiplication and addition	
Calculation	Traditional self-attentive mechanism	GSRM
$\boldsymbol{K}'\boldsymbol{Q}^T$ or $\boldsymbol{K}'\boldsymbol{C}^T$	33 488 896	1 569 792
$\boldsymbol{M}\boldsymbol{Q}$ or $\boldsymbol{M}\boldsymbol{C}$	33 488 896	1 507 328

The GSRM determines the relevance of global semantic information and captures the global contextual information with low computational complexity and low memory. Moreover, the global representation property of clustering center features in $\boldsymbol{C}$ of GSRM is better than the local features in $\boldsymbol{Q}$ of traditional self-attention mechanism, thus augmenting the feature representation. Therefore, $\boldsymbol{Z}_t'$ and $\boldsymbol{Z}_d'$ have a wide contextual view and can thus selectively aggregate different contexts according to the spatial attention map, allowing for more robust segmentation.

E. Training and Inference Procedures

To accurately segment the texture surface defects, a multitask loss function with two types of losses is designed to optimize the SIDN. The two types of losses are: 1) the feature separation loss; and 2) the segmentation loss.

$$L = L_s + L_{fd}. \tag{25}$$

The training procedure of the SIDN is is as follows. First, after the DGM generates a number of artificial defective samples, the

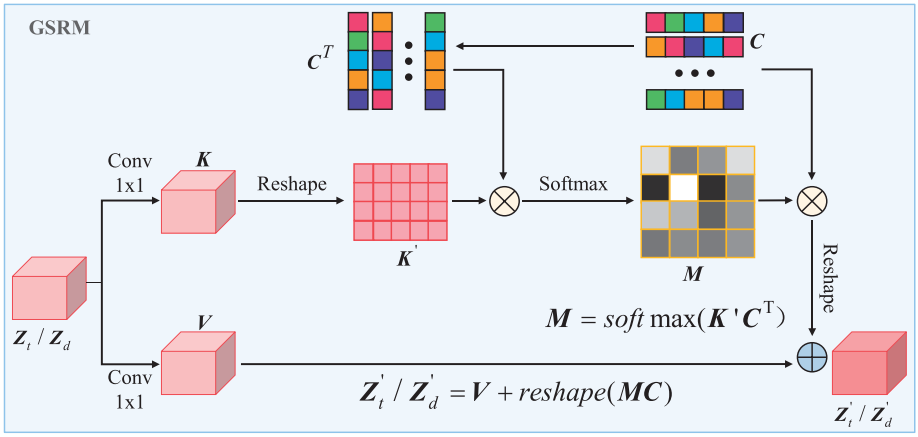

Fig. 4. Structure details of the GSRM.

FEM extracts the features $\boldsymbol{Z}$ of the input. Then, to prevent the coupling of the texture and defect semantic information from affecting the final segmentation accuracy, the FSEM decomposes the texture semantic features $\boldsymbol{Z}_t$ and defect semantic features $\boldsymbol{Z}_d$ from $\boldsymbol{Z}$ with (6) and (7). Next, the GSRM is trained to establish the global semantic correlation to enhance $\boldsymbol{Z}_t$ and $\boldsymbol{Z}_d$, thereby obtaining $\boldsymbol{Z}_t'$ and $\boldsymbol{Z}_d'$ by using (21) and (22). Finally, the SM utilizes $\boldsymbol{Z}_t'$ and $\boldsymbol{Z}_d'$ to directly segment the textures and defects by using (3) and (4). In this study, the SIDN is trained using the Adam optimizer, with a learning rate of 0.0001 and a batch size of 16.

After the training procedure, the trained SIDN is tested. With the trained SIDN, a defective image can be segmented into a texture segmentation result $\boldsymbol{I}_s^t$ and a defect segmentation result $\boldsymbol{I}_s^d$. Then, $\boldsymbol{I}_s^t$ can be combined with $\boldsymbol{I}_s^d$ to generate the resulting image $\boldsymbol{I}_s$ as

$$\boldsymbol{I}_s = (1 - \lambda_f)(1 - \boldsymbol{I}_s^t) + \lambda_f \boldsymbol{I}_s^d \tag{26}$$

where $\boldsymbol{I}_s \in R^{W \times H}$ and $0 \leq \lambda_f \leq 1$ are the weights of the two terms. Since $\boldsymbol{I}_s^t$ and $\boldsymbol{I}_s^d$ are obtained in parallel, the SIDN is efficient during testing. A median filter operation, a threshold segmentation operation, and a morphological operation are applied to the resulting image to generate the inspection result.

F. Parameter Setup

The key hyperparameters of the SIDN include K, λ_1, λ_2, λ_3, λ_4, and λ_f.

The number of clusters, K, influences the segmentation accuracy. If K is too small, the texture and defects cannot be effectively represented, and if K is too large, the model performance will not be effectively improved and the computational effort will be increased. The recommended range of K is 8–20. In this study, K was set to 12. λ_1 is the balance coefficient of the loss in (5), which was set to 0.1 in this study and can be used to address the issue of category imbalance. λ_2, λ_3, and λ_4 are the weights of the three loss terms in (19). To balance the influence of each loss, the weights were set as $\lambda_2 = 0.01$, $\lambda_3 = 0.01$, and $\lambda_4 = 0.01$ in this study. $0 \leq \lambda_f \leq 1$ is the weight associated with combining the results in (26). In this study, λ_f was set to 0.5.

IV. Experimental

In this section, a series of experiments are used to verify the performance of the proposed SIDN. Specifically, an ablation analysis of the SIDN was conducted. The overall defect inspection performance of the SIDN was qualitatively and quantitatively compared with some state-of-the-art methods. The SIDN was trained on a computer with NVIDIA Titan Xp GPU and was implemented using PyTorch in Python [21].

A. Dataset and Evaluation Criteria

MVTec AD [10] is a novel dataset for anomaly detection that mimics real-world industrial inspection scenarios. MVTec AD includes five types of textures. Each texture category contains 230–280 defect-free images in the training set and 50–100 defective images with pixel-level annotations in the test set. The complexity and variety of textures and defects in MVTec AD make MVTec AD very challenging. Experiments were conducted with the following texture images from the MVTec AD dataset: carpet, wood, leather, tile, and grid. In each batch, SIDN was trained with about 1500 training images

Area under the receiver operating characteristic curve (AuROC) is seen as a standard way to evaluate distribution prediction models where the output is a continuous probability value and a threshold is needed to binarize the prediction result. It avoids the influence of subjective assumptions in threshold selection by summarizing the overall performance of the model under all possible thresholds. To quantitatively evaluate the performance of the proposed SIDN, AuROC was adopted as an evaluation metric. AuROC is an indicator that comprehensively reflects the false positive rate (FPR) and true positive rate (TPR), and it is insensitive to thresholds, enabling an objective assessment of the model's detection performance. In our experiments, the TPR represents the percentage of anomalous pixels that were correctly classified as anomalous, and the FPR represents the percentage of normal pixels that were incorrectly classified as anomalous.

B. Ablation Analysis of the SIDN

In this section, to verify the effectiveness of the FSEM and GSRM in the proposed SIDN, some ablative settings were

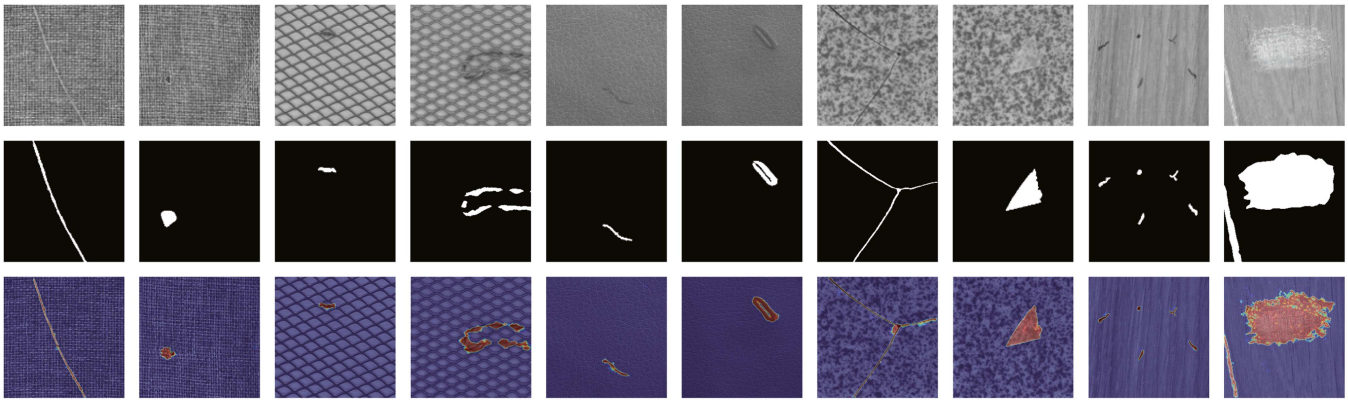

Fig. 5. Segmentation results on the texture images. The three rows display the original defective images, ground truth, and segmentation results.

TABLE II
SIDN ABLATION ANALYSIS FOR CARPET AND WOOD TEXTURE IMAGES FROM THE MVTEC AD DATASET

Module	A	B	C	D
FSEM without clustering	✓	✗	✗	✗
FSEM without L_{td}	✗	✓	✗	✗
FSEM	✗	✗	✓	✓
GSRM	✗	✗	✗	✓
AuROC on carpet	0.981	0.988	0.989	0.990
AuROC on wood	0.938	0.939	0.953	0.956

designed, as given in Table II. The ablation experiments were performed on images of regularly textured carpet and irregularly textured wood.

1) Effectiveness of the FSEM: The goal of the FSEM is to decompose the texture semantic information and defect semantic information from the shared feature map in order to minimize performance degradation due to the coupling of these two kinds of semantic information. As given in Table II, regardless of whether the texture was regular or irregular, method B improved the AuROC by 0.007 more on the carpet and 0.001 more on the wood than method A, while method C improved the AuROC by 0.001 more on the carpet and 0.014 more on the wood than method B, which demonstrates that the FSEM can improve the performance of the SIDN by decomposing the semantic information.

2) Effectiveness of the GSRM: The goals of the GSRM are to determine the global semantic information relevance and to capture the global contextual information with low computational complexity and low memory use, thus improving the feature representation. As given in Table II, regardless of whether the texture was regular or irregular, method D, the SIDN, improved the AuROC by 0.001 more on the carpet and 0.003 more on the wood than method C, yielding the best inspection results and demonstrating that the GSRM can improve the performance of SIDN by augmenting the feature representation.

The ablation analyses showed that regardless of whether the texture was regular or irregular, the FSEM and GSRM improved the texture defect detection performance of the SIDN to some extent.

TABLE III
AUROC ON FIVE TEXTURES FROM THE MVTEC AD DATASET [10]

Method	Carpet	Grid	Leather	Tile	Wood	Mean
TEXEMS	0.88	0.72	0.97	0.41	0.78	0.752
AE	0.59	0.90	0.75	0.51	0.73	0.696
AE-SSIM	0.87	0.94	0.78	0.59	0.73	0.782
MSFCAE	0.782	0.881	0.917	0.532	0.812	0.785
AnoGAN	0.54	0.58	0.64	0.50	0.62	0.576
OCGAN	0.792	0.876	0.865	0.748	0.810	0.818
CutPaste	0.983	0.975	0.995	0.905	0.955	0.963
Patch SVDD	0.926	0.962	0.974	0.914	0.908	0.937
SBR	0.95	0.98	0.98	0.95	0.93	0.96
NSA	0.955	**0.992**	0.995	**0.993**	0.907	0.968
SPADE	0.975	0.937	0.976	0.874	0.885	0.929
PaDiM	**0.991**	0.973	0.992	0.941	0.949	0.969
FCDD	0.96	0.91	0.98	0.91	0.88	0.93
SIDN	0.990	0.983	**0.996**	0.964	**0.956**	**0.978**

Note: Boldface font and underline indicate the best and second-best results, respectively.

C. Overall Comparative Experiments With the SIDN

Fig. 5 shows the segmentation results of the proposed SIDN on ten defective images from the MVTec AD dataset. Although no real defective images were utilized for training, the proposed SIDN method accurately segmented various kinds of defects in the texture images.

To evaluate the effectiveness and performance of the SIDN, comparative studies between our method and 13 other excellent methods were carried out with the texture images from the MVTec AD dataset. The 13 methods include: 1) TEXEMS [3]; 2) AE [7]; 3) AE-structural similarity index measure (SSIM) [7]; 4) MS-FCAE [8]; 5) AnoGAN [18]; 6) OCGAN [15]; 7) CutPaste [22]; 8) Patch support vector data description (SVDD) [23]; 9) stable background reconstruction (SBR) [24]; 10) natural synthetic anomalies (NSA) [25]; 11) semantic pyramid anomaly detection (SPADE) [26]; 12) patch distribution modeling (PaDiM) [27]; and 13) fully convolutional data description (FCDD) [28].

The AuROC values of these methods and the proposed method are given in Table III. The best results are emphasized in boldface and the second-best results are underlined. As given

in Table III, AE and AnoGAN do not perform well. Although TEXEMS, AE-SSIM, MS-FCAE, and OCGAN perform slightly better, they are unable to obtain good results for all categories, especially for complex textures, such as tiles. Because these methods are based on texture background reconstruction, they cannot guarantee a high-quality reconstruction of the texture background while suppressing the defect foreground; thus, the defective structures are also reconstructed, and many defective regions remain in the reconstructed images, resulting in poor performance. On the other hand, the performance of some recent methods, such as CutPaste, Patch SVDD, SBR, NSA, SPADE, PaDiM, and FCDD, was greatly improved, and the AuROC was increased to greater than 90%. Among these seven methods, NSA and PaDiM had better average performance, with AuROC values of 0.968 and 0.969, respectively. However, because these methods have relatively large network scales, they have high computational complexity and occupy a large amount of GPU memory.

In contrast, although NSA and PaDiM performed well, the proposed SIDN achieved a more satisfactory performance with a smaller network. Overall, on these five types of texture images, the SIDN achieved the best detection results on leather and wood images and achieved the second-best detection results on carpet, grid, and tile images. Moreover, the SIDN had the highest average AuROC of the 13 methods. The SIDN achieved a satisfactory performance due to the following two factors.

1) First, the SIDN uses the FSEM to decompose the texture semantic features and defect semantic features from the shared feature map generated by the FEM, which prevents the coupling of the texture and defect semantic information from affecting the final segmentation accuracy.
2) Second, the SIDN uses the GSRM to determine the global semantic information relevance and capture the global contextual information with low computational and low memory use, thus improving the feature representation and increasing the robustness of the segmentation.

The average time for SIDN to process a 256×256 image is 85 ms. Since a single-channel grayscale image is used, SIDN only spends one-third of the GPU memory to store image data compared to the compared methods using a three-channel color image, which reduces the equipment requirements and facilitates the application in practice.

These experimental results demonstrate that the SIDN performed better than the other methods. By decomposing the texture semantic features and defect semantic features from the shared feature map and determining the relevant global semantic information, the SIDN accurately segment both textures and defects. Thus, the SIDN can accurately inspect various types of texture defects, achieving the best texture defect detection performance without requiring a large-scale network or additional datasets.

V. Conclusion

In this article, we proposed a novel SIDN for accurately segmenting various texture defects. In the SIDN, the DGM was proposed to generate a large number of defective images. As a result, this method does not require any real defective samples. In addition, a novel FSEM was proposed to decompose the texture semantic information and defect semantic information from the shared feature map generated by the FEM, which prevents the coupling of the texture and defect semantic information from affecting the performance. Then, the GSRM was proposed to determine the global semantic information relevance to comprehensively consider the context and augment the feature representation. Finally, the SM accurately segments the textures and defects instead of reconstructing the texture background. The experimental results on a mainstream texture defect dataset demonstrated that the SIDN can achieve state-of-the-art inspection accuracy without requiring a large-scale network or additional datasets.

References

[1] D. Aiger and H. Talbot, "The phase only transform for unsupervised surface defect detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 295–302.
[2] W.-C. Li and D.-M. Tsai, "Defect inspection in low-contrast LCD images using hough transform-based nonstationary line detection," *IEEE Trans. Ind. Informat.*, vol. 7, no. 1, pp. 136–147, Feb. 2011.
[3] X. Xie and M. Mirmehdi, "TEXEMS: Texture exemplars for defect detection on random textured surfaces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 8, pp. 1454–1464, Aug. 2007.
[4] H. Y. T. Ngan, G. K. H. Pang, and N. H. C. Yung, "Automated fabric defect detection– A review," *Image Vis. Comput.*, vol. 29, no. 7, pp. 442–458, 2011.
[5] R. Ren, T. Hung, and K. C. Tan, "A generic deep-learning-based approach for automated surface inspection," *IEEE Trans. Cybern.*, vol. 48, no. 3, pp. 929–940, Mar. 2018.
[6] H. Dong, K. Song, Y. He, J. Xu, Y. Yan, and Q. Meng, "PGA-Net: Pyramid feature fusion and global context attention network for automated surface defect detection," *IEEE Trans. Ind. Informat.*, vol. 16, no. 12, pp. 7448–7458, Dec. 2020.
[7] P. Bergmann, S. Löwe, M. Fauser, D. Sattlegger, and C. Steger, "Improving unsupervised defect segmentation by applying structural similarity to autoencoders," in *Proc. 14th Int. Joint Conf. Comput. Vis. Imag. Comput. Graph. Theory Appl.*, 2019, pp. 372–380.
[8] H. Yang, Y. Chen, K. Song, and Z. Yin, "Multiscale feature-clustering based fully convolutional autoencoder for fast accurate visual inspection of texture surface defects," *IEEE Trans. Autom. Sic. Eng.*, vol. 16, no. 3, pp. 1450–1467, Jul. 2019.
[9] H. Yang, Q. Zhou, K. Song, and Z. Yin, "An anomaly feature-editing-based adversarial network for texture defect visual inspection," *IEEE Trans. Ind. Informat.*, vol. 17, no. 3, pp. 2220–2230, Mar. 2021.
[10] P. Bergmann, M. Fauser, D. Sattlegger, and C. Steger, " MVTec AD– A comprehensive real-world dataset for unsupervised anomaly detection," in *Proc. IEEECVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 9584–9592.
[11] J. K. Chow, Z. Su, J. Wu, P. S. Tan, X. Mao, and Y. H. Wang, "Anomaly detection of defects on concrete structures with the convolutional autoencoder," *Adv. Eng. Inform.*, vol. 45, 2020, Art. no. 101105.
[12] Z. Chen, C. K. Yeo, B. S. Lee, and C. T. Lau, "Autoencoder-based network anomaly detection," in *Proc. IEEE Wirel. Telecommun. Symp.*, 2018, pp. 1–5.
[13] V. L. Cao, M. Nicolau, and J. McDermott, "Learning neural representations for network anomaly detection," *IEEE Trans. Cybern.*, vol. 49, no. 8, pp. 3074–3087, Aug. 2019.
[14] D. Gong et al., "Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 1705–1714.
[15] P. Perera, R. Nallapati, and B. Xiang, "OCGAN: One-class novelty detection using GANs with constrained latent representations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 2893–2901.
[16] T. Jiang, Y. Li, W. Xie, and Q. Du, "Discriminative reconstruction constrained generative adversarial network for hyperspectral anomaly detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 7, pp. 4666–4679, Jul. 2020.

[17] G. Hu, J. Huang, Q. Wang, J. Li, Z. Xu, and X. Huang, "Unsupervised fabric defect detection based on a deep convolutional generative adversarial network," *Textile Res. J.*, vol. 90, no. 3-4, pp. 247–270, 2020.
[18] T. Schlegl, P. Seebock, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs, "Unsupervised anomaly detection with generative adversarial networks to guide marker discovery," in *Proc. Int. Conf. Inf. Process. Med. Imag.*, 2017, pp. 146–157.
[19] E. Min, X. Guo, Q. Liu, G. Zhang, J. Gui, and J. Long, "A survey of clustering with deep learning: From the perspective of network architecture," *IEEE Access*, vol. 6, pp. 39501–39514, 2018.
[20] A. Vaswani et al., "Attention is all you need," *Neural Inf. Process. Syst.*, vol. 30, pp. 5998–6008, 2017.
[21] A. Paszke et al., "PyTorch: An imperative style, high-performance deep learning library," *Adv. Neural Inf. Process. Syst.*, vol. 32, pp. 8026–8037, 2019.
[22] C.-L. Li, K. Sohn, J. Yoon, and T. Pfister, "CutPaste: Self-supervised learning for anomaly detection and localization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 9659–9669.
[23] J. Yi and S. Yoon., "Patch SVDD: Patch-level SVDD for anomaly detection and segmentation," in *Proc. Asian Conf. Comput. Vis.*, 2020, pp. 375–390.
[24] C. Lv, F. Shen, Z. Zhang, D. Xu, and Y. He, "A novel pixel-wise defect inspection method based on stable background reconstruction," *IEEE Trans. Instrum. Meas.*, vol. 70, 2020, Art no. 5005213, doi: 10.1109/TIM.2020.3038413.
[25] H. M. Schlüter et al., "Natural synthetic anomalies for self-supervised anomaly detection and localization," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 474–489.
[26] N. Cohen and Y. Hoshen, "Sub-image anomaly detection with deep pyramid correspondences," 2020, *arXiv:2005.02357v3*.
[27] T. Defard, A. Setkov, A. Loesch, and R. Audigier, "PaDim: A patch distribution modeling framework for anomaly detection and localization," in *Proc. Int. Conf. Pattern Recognit.*, 2021, pp. 475–489.
[28] P. Liznerski, L. Ruff, R. A. Vandermeulen, B. J. Franks, M. Kloft, and K.-R. Müller, "Explainable deep one-class classification," 2020, *arXiv:2007.01760v3*.

Hua Yang (Member, IEEE) received the B.S. and M.S degrees in engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2006 and 2008, respectively, and the Ph.D. degree in engineering from Hiroshima University, Higashihiroshima, Japan, in 2011.

From 2011 to 2012, he was a Research Associate with Hiroshima University, where he has been an Assistant Professor since 2012. From 2013 to 2019, he was an Associate Professor with the School of Mechanical Science and Engineering, Huazhong University of Science and Technology, where he is currently a Professor. His research focuses on high-speed vision and its applications (such as object recognition and detection, particle image velocity, and dynamic-based vision inspection).

Jiale Hu received the B.S. degree in mechanical design, manufacturing, and automation from Wuhan University, Wuhan, China, in 2020, and the M.S. degree in engineering from the State Key Laboratory of Digital Manufacturing Equipment and Technology, Huazhong University of Science and Technology, Wuhan, China, in 2022.

His research interests include image segmentation, object detection, and deep learning.

Zhouping Yin (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in mechanical engineering from the Huazhong University of Science and Technology, Wuhan, China, in 1994, 1996, and 2000, respectively.

Since 2005, he has been the Vice Head of the State Key Laboratory of Digital Manufacturing Equipment and Technology, Huazhong University of Science and Technology, where he is currently a Professor with the School of Mechanical Science and Engineering. He is also Leading a research group and conducting research in electronic manufacturing equipment and technology, including flexible electronics and machine vision.

Zhengjia Wang received the Ph.D. degree in mechatronics engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2010.

He is currently a Lecturer with the School of Mechanical Engineering, Hubei University of Technology, Wuhan, China. His research interests include smart manufacturing and machine vision.