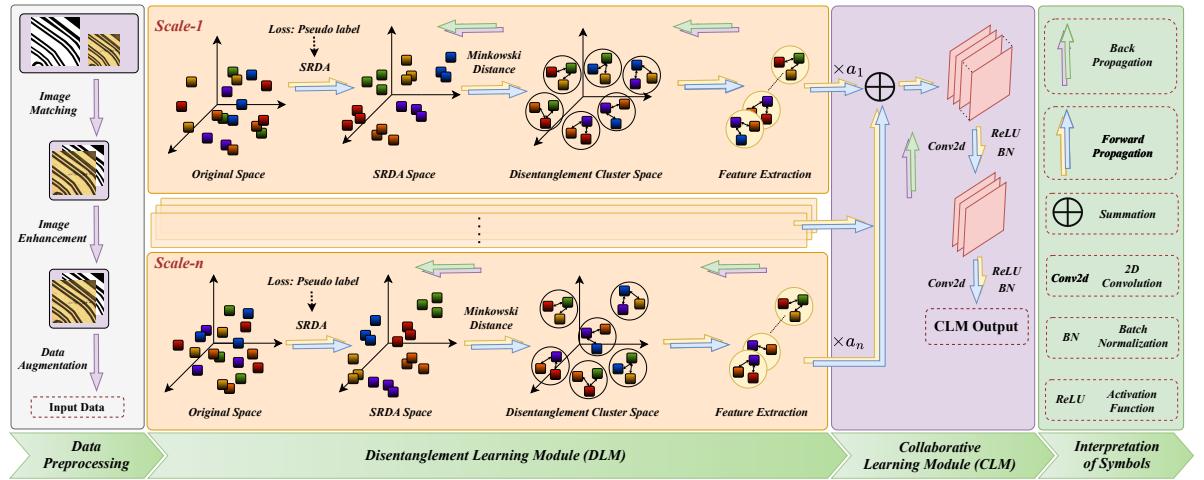


Graphical Abstract

Multi-view unwrapping collaborative network for printed circuit boards defect classification

Feng Zhan, Weiming Gan, Lingkai Hu, Zhidi Huang, Yi Cai, Hao He



Highlights

Multi-view unwrapping collaborative network for printed circuit boards defect classification

Feng Zhan, Weiming Gan, Lingkai Hu, Zhidi Huang, Yi Cai, Hao He

- A novel multi-view unwrapping collaborative network (MUCN) is proposed to enhance defect feature extraction and classification performance in printed circuit boards.
- The disentanglement learning module (DLM) and collaborative learning module (CLM) reduce overfitting in multi-view data and strengthen defect feature extraction.
- A unique clustering loss function, based on the gray-level co-occurrence matrix (GLCM), effectively handles texture similarities between true and false positives, improving classification of texture-similar defects.
- Integrating multi-scale feature extraction within the collaborative learning module boosts the model's ability to detect multi-scale defects.
- Extensive experiments demonstrate that MUCN outperforms state-of-the-art methods on HRIPCB and PCB-TP datasets, achieving superior generalization and classification results.

Multi-view unwrapping collaborative network for printed circuit boards defect classification

Feng Zhan^{a,*}, Weiming Gan^{b,*}, Lingkai Hu^b, Zhidi Huang^c, Yi Cai^{a,d,**}, Hao He^e

^aThe Hong Kong University of Science and Technology (Guangzhou), Smart Manufacturing Thrust, Nansha, Guangzhou, 511400, China

^bSchool of Mechanical and Electrical Engineering, Guangzhou University, Guangzhou, 510006, China

^cShennan Circuits Co., Ltd., Shenzhen, 518117, China

^dThe Hong Kong University of Science and Technology, Hong Kong SAR, China

^eSchool of Politics and International Relations, East China Normal University, Shanghai, 200062, China

ARTICLE INFO

Keywords:

Slim convolution
Printed circuit boards
Information redundancy
Multi-scale receptive field
Electronic product reliability

ABSTRACT

Accurate classification of printed circuit board (PCB) defects is essential for improving the reliability and yield of electronic products. However, the accuracy of PCB defect classification is influenced by both local and global data distributions of defects, and redundant correlation information can lead to overfitting in existing models. Additionally, the texture similarity between some true positive and false positive PCB defects may reduce classification accuracy. To address these challenges, this paper proposes a novel multi-view unwrapping collaborative network (MUCN), which consists of a disentanglement learning module, a collaborative learning module, and a clustering loss function. Specifically, the unwrapping and collaborative learning modules analyze PCB data by reducing redundant features from multiple views and enhancing the model's multi-scale feature selection capability, thereby mitigating overfitting caused by redundant correlations during training. The clustering loss function leverages prior knowledge from PCB design and real images to guide model updates, improving its ability to classify texture-similar defects. Experimental results demonstrate that the proposed MUCN outperforms state-of-the-art methods on two real-world surface defect datasets (HRIPCB: 99.69% and PCB-TP: 92.57%) and provides visual interpretability.

1. Introduction

Printed circuit boards (PCBs) are widely used in industrial products [1–6], making identifying and screening defective PCBs critical. Currently, two major systems are employed for PCB defect detection: Automated Optical Inspection (AOI) and Visual Repair Systems (VRS) [7]. AOI offers fast and reliable defect detection by automatically scanning printed circuit boards for flaws using high-resolution cameras and advanced algorithms. In contrast, VRS involve manual verification and correction of defects identified by AOI, allowing human operators to address errors that automated systems might overlook. However, both systems have limitations in practical applications [8, 9]. For example, traditional image recognition techniques used in AOI often generate a high number of false positives, which are mistakenly identified as defects. If defective PCBs reach the market, they can cause damage to machinery or even lead to personal injury [10, 11]. On the other hand, discarding PCBs falsely identified as defective results in unnecessary financial losses for companies [12, 13]. Additionally, VRS requires substantial manual labor to classify AOI-detected defects into true positives and false positives, significantly prolonging PCB production time and imposing a heavy economic burden. Therefore, developing intelligent methods for PCB defect classification is crucial. However, several challenges persist in PCB surface defect classification tasks. First, redundant correlation information from multiple views can lead to model overfitting [14, 15]. Second, the texture similarity between some true positive and false positive defects greatly complicates classification [16, 17].

According to the literature review, researchers have conducted extensive research to mitigate the negative impacts of these issues [18–20]. These studies [21, 22] can be divided into two categories: 1) model optimization methods and 2) prior knowledge integration methods. In terms of model optimization, researchers have improved model structures to enhance defect detection performance. For instance, a multi-scale feature fusion method for small object detection was proposed to detect surface defects [23]. Another approach developed a spatial attention-based

*Co-first author.

**Corresponding author.

E-mail address: yicai@hkust-gz.edu.cn (Yi Cai)

bilinear Convolutional Neural Network for non-destructive detection of casting surface defects [24]. Additionally, a collaborative learning classification model was introduced for PCB defect classification [25]. Most of these methods enhance feature extraction capabilities to improve the detection accuracy of prominent defect features. However, these deep learning approaches may struggle to effectively reduce the overfitting caused by redundant correlation information from multiple views during feature extraction [26]. This is primarily because these methods stack feature maps from different layers without weighting or compressing the redundant feature layers. As the model deepens, the overfitting caused by view redundancy can decrease the overall model accuracy.

Regarding prior knowledge integration methods, existing studies have enhanced defect detection accuracy by incorporating prior knowledge of the data into models [27]. For instance, a transfer learning approach based on Fourier transform feature fusion was proposed to improve bearing fault diagnosis accuracy [28]. In [29], a wafer defect pattern classification method based on topological data analysis was introduced. In [30], a research method for small circuit breaker fault diagnosis was developed, where fractal techniques were used to capture nonlinear signal features, followed by accurate fault classification using a probabilistic neural network. Additionally, in [31], a dynamic weight wavelet attention neural network was proposed to improve the model's noise resistance and detection capability for minor defects. However, most of these methods require the integration of complex prior knowledge calculations into the model, which can increase both the computational burden and the difficulty of feature learning [32].

To address the aforementioned challenges, this article proposes a novel multi-view unwrapping collaborative network (MUCN). To mitigate overfitting caused by redundant correlation in multi-view data, an unwrapping learning module and a collaborative learning module are introduced. First, the unwrapping learning module establishes an unwrapping space where redundant correlated features are clustered and fused, reducing overfitting and enhancing classification accuracy. Second, the collaborative learning module improves the model's feature extraction capabilities for multi-scale defects by integrating multi-scale feature space information. A clustering loss function is proposed to resolve the texture similarity issue between true positive and false positive defects. This function leverages the gray-level co-occurrence matrix (GLCM) entropy as prior knowledge, dynamically adjusting sample loss values based on variations in data distribution and confidence between PCB design images and real images, thereby significantly improving the classification of texture-similar defects.

In summary, this article presents an advanced model for true- and pseudo-positive PCB defect classification. The model integrates a multiscale information extractor with a dynamically weighted loss function. Evaluation is conducted on a PCB dataset from actual industrial production, which contains various challenging defects that are hard to classify based on local texture distribution alone.

In general, the main contributions of this article are as follows:

1. Disentanglement learning module (DLM) and collaborative learning module (CLM) were designed to mitigate overfitting caused by redundant correlation information in Convolutional Neural Networks (CNNs), thereby enhancing the model's defect feature extraction capabilities.
2. A GLCM-based loss function, referred to as clustering loss (CL), is proposed to create a loss metric space by comparing the texture complexity of the design map with that of the scan map, effectively reducing task complexity by calculating the loss value within this metric space.
3. Experimental results show that MUCN outperforms state-of-the-art methods on two datasets (HRIPCB and PCB-TP), demonstrating superior generalization performance.

2. Preliminaries

2.1. PCB defect classification

Based on the literature analysis, numerous scholars have conducted extensive research on PCB defect classification methods based on deep learning [28, 33, 34]. Xu et al. [35] proposed a depthwise separable spatial pyramid and global average pooling branch method, which effectively enhances the correlation between local and global information in the model. Zhang et al. [36] introduced a multi-model hybrid fusion network that not only improves classification accuracy for small defects but also enhances feature reuse. Han et al. [37] proposed a feature fusion network based on PCA and Bayesian theory that effectively combines shallow image features with deep semantic features, improving the ability of feature vectors to represent defects. Makwana et al. [38] developed a correlation residual module to establish the relationship between defects and their surrounding environment, significantly improving PCB defect classification accuracy. Zhang et al. [39] introduced a cost-sensitive residual network that alleviates the training complexity caused by imbalanced data through a cost-sensitive adjustment layer.

2.2. Multi-view feature redundancy

Feature redundancy across views leads to model overfitting, thereby reducing the model's feature extraction capability. Thus, reducing feature redundancy between multiple views can effectively improve model robustness [40]. In recent years, several scholars have proposed methods to address this issue. Li et al. [41] utilized spatial and channel redundancy to compress CNN features, developing an efficient CNN module. Qiu et al. [42] introduced a SlimConv module, which reduces channel redundancy through three steps: Reconstruct, Transform, and Fuse. Zhao et al. [43] proposed a frequency regularization method to constrain the non-zero elements of network parameters in the frequency domain, thus suppressing the expression of redundant features. Huangfu et al. [44] developed the Ghost module, which reduces the computational complexity of redundant features through linear mapping, enhancing the model's feature extraction capability. Zhang et al. [45] proposed a Segmentation Convolution Module, which extracts features from feature maps rich in information while lightweight redundant feature maps, improving the model's computational efficiency.

2.3. GLCM texture analysis

Texture is widely present on the surfaces of various objects in nature. Traditionally, in images, texture refers to a macroscopic representation of some locally repetitive patterns in feature intensity [46, 47]. Aribowo et al. [21] proposed a GLCM-based defect classification method for THT-PCB, utilizing GLCM to analyze the information within the region of interest, thereby improving the accuracy of disturbed soldering defect detection. Redford et al. [48] introduced an image defect classification algorithm based on the GLCM with multi-characteristic averaging, achieving favorable results in defect classification.

3. Methodology

3.1. Overall structure

This section provides a detailed explanation of the proposed method. First, the unwrapping learning module creates an unwrapping clustering space by learning projection features. Subsequently, the collaborative learning module performs feature fusion within this multi-scale unwrapping clustering space. Additionally, GLCM extracts prior knowledge and incorporates it into the loss function to guide model updates. The overall architecture of the multi-view unwrapping collaborative network is shown in Fig. 1.

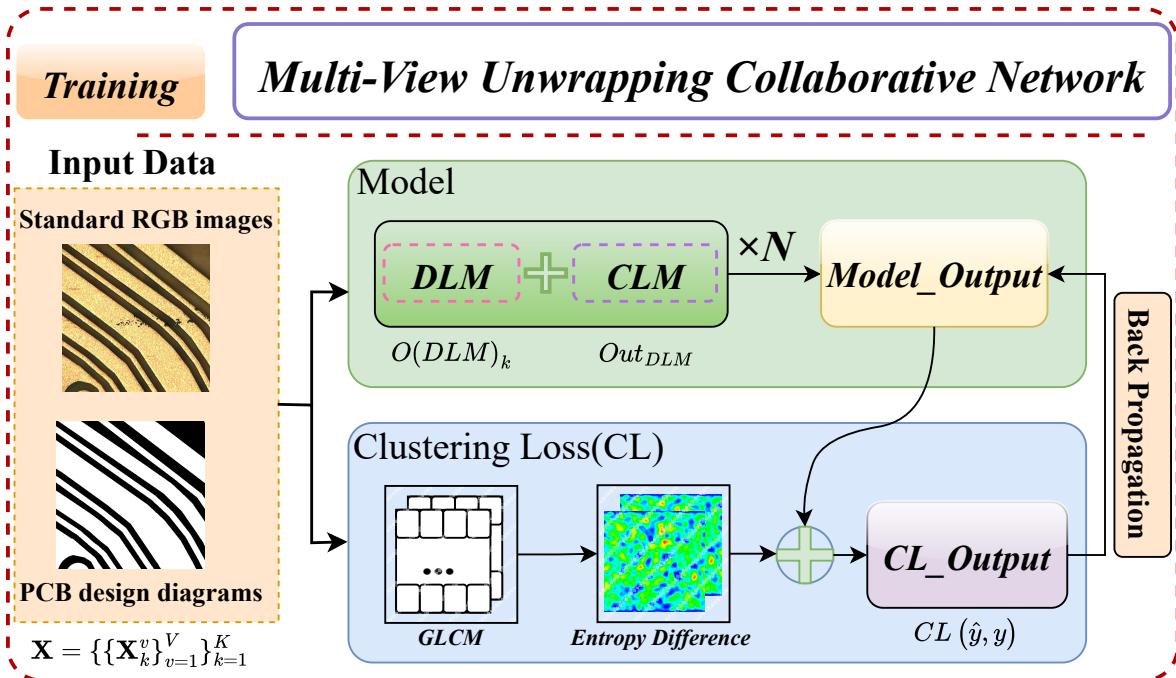


Fig. 1. General structure of the proposed MUCN.

In this paper, a multi-view data matrix \mathbf{X} , containing k layers and v feature maps, is denoted as $\mathbf{X} = \left\{ \{\mathbf{X}_k^v\}_{v=1}^V \right\}_{k=1}^K$. In the k -th layer, the data matrix \mathbf{X}_k^v includes $w^k \times h^k$ feature maps of each sample, denoted as $\mathbf{X}_k^v = [x_k^1, x_k^2, \dots, x_k^v] \in \mathbb{R}^{w^k \times h^k \times v}$. In this section, the proposed method is presented in detail. Specifically, the disentanglement learning module establishes a disentanglement clustering space by learning projection features, and the collaborative learning module performs feature fusion in the multi-scale disentanglement clustering space.

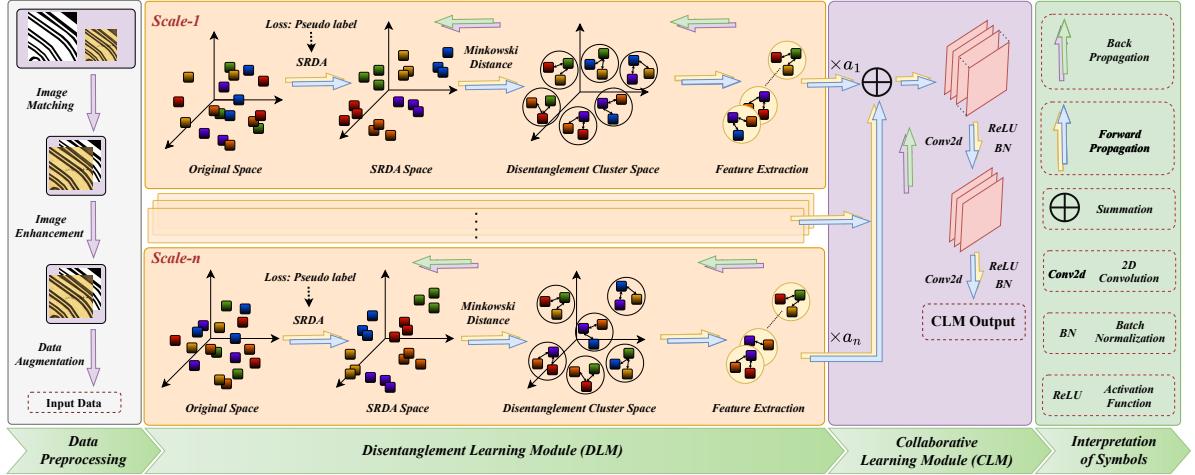


Fig. 2. Detailed schematic diagram of the DLM & CLM structure.

3.2. Disentanglement learning module

Feature extraction from multi-view data using CNNs often leads to overfitting due to redundant features across views, which reduces the model's ability to extract cross-view information. Therefore, it is essential to separate redundant features within views to minimize the risk of overfitting. In this paper, the Spectral Regression Discriminant Analysis algorithm, based on Linear Discriminant Analysis (LDA) [49], is employed to map the views into a disentanglement clustering space. This space distances redundant and similar feature views, allowing for more effective feature extraction and fusion. This not only enhances the diversity of feature fusion across views but also suppresses redundant information within the views.

LDA is a supervised dimensionality reduction method that aims to find a projection direction where points from the same class are closely clustered, and points from different classes are well separated. In this subsection, the SRAD algorithm, derived from LDA, is used to build a disentanglement clustering space for feature extraction. The objective function, guided by class labels, is defined as follows:

First, encodes the input data once and is computed as follows:

$$\xi_k^v = \arg \min_{\xi_k^v} \frac{\text{tr} \left(\xi_k^v S_t (\xi_k^v)^T \right)}{\text{tr} \left(\xi_k^v S_b (\xi_k^v)^T \right)} \quad (1)$$

$$\text{s.t. } \xi_k^v S_b (\xi_k^v)^T = 1,$$

where ξ_k^v represents the projection matrix, and $\text{tr}(\bullet)$ denotes the trace of a matrix. The total scatter matrix S_t and the between-class scatter matrix S_b are defined as follows:

$$S_t = \sum_{i=1}^n (x_i - \bar{x}) (x_i - \bar{x})^T \quad (2)$$

$$S_b = \sum_{j=1}^{\pi} n_k (\bar{x}^j - \bar{x}) (\bar{x}^j - \bar{x})^T, \quad (3)$$

where x_i symbolizes the i -th sample, \bar{x} represents the average of all samples, ϖ is the overall amount of classes, n_k indicates the amount of samples in the k -th class, and \bar{x}^j is the average of the samples in the k -th class.

The solution of Eq. (3) can be achieved by singular value decomposition (SVD). However, its computational cost in a single-view case is $O(v_k w_{v_k} h_{v_k} t + t^3)$, where $t = \min(v_k, w_{v_k} h_{v_k})$. To reduce time complexity, this paper employs Spectral Regression Discriminant Analysis (SRDA), a variant of linear discriminant analysis. The time cost of SRDA is significantly lower than that of SVD. Using the spectral regression framework, Eq. (3) is reformulated as follows:

$$\xi_k^v = \arg \min_{\xi_k^v} \frac{\text{tr} \left(\xi_k^v S_t (\xi_k^v)^T \right)}{\text{tr} \left(\xi_k^v S_b (\xi_k^v)^T \right)} = \arg \min_{\xi_k^v} \left\| \xi_k^v \left(\bar{X}_k^v \right)_n - \mathcal{T}_k \right\|_F^2, \quad (4)$$

where $\left(\bar{X}_k^v \right)_n$ is the data matrix of the class in the k -th layer, $\left(\bar{X}_k^v \right)_n = [(\bar{x}_k^v)_1, (\bar{x}_k^v)_2, \dots, (\bar{x}_k^v)_n] \in \mathbb{R}^{w^k \times h^k \times n}$. $\left(\bar{X}_k^v \right)_n = [(X_k^v)_1 - \bar{x}^v, (X_k^v)_2 - \bar{x}^v, \dots, (X_k^v)_n - \bar{x}^v]$, where \bar{x}^v denotes the total sample mean matrix in the k -th layer, and n denotes the total number of samples in the class for the k -th layer. Pseudo-label \mathcal{T}_k can be obtained through backpropagation of the model. Assuming the loss function of the model is $\mathcal{J}(\bullet)$, the gradient of a certain layer can be calculated according to backpropagation. \mathcal{T}_k will be represented in detail in the following Eq. (7).

Moreover, to enhance the selection of discriminative and informative features, sparsity is introduced in the regularization term ξ_k^v . The $\mathcal{L}_{1,2}$ -norm function promotes column sparsity by maximizing the number of zero-value elements in each column during the minimization process. Therefore, the $\mathcal{L}_{1,2}$ -norm serves as a regularizer to improve feature discrimination. The formula is given below:

$$\xi_k^v = \arg \min_{\xi_k^v} \left\| \xi_k^v \left(\bar{X}_k^v \right)_n - \mathcal{T}_k \right\|_F^2 + \left\| \xi_k^v \right\|_{1,2}. \quad (5)$$

To further enhance the nonlinearity of the model and improve its adaptability to samples, ξ_k^v is encoded through the encoder $\mathcal{H}(\bullet)$. These concepts can be precisely formulated mathematically as follows:

$$\mathcal{G}_k^v \left(\bar{X}_k^v \right) = \frac{\exp \left(\mathcal{N} \left(\mathcal{H} \left(\xi_k^v \left(\bar{X}_k^v \right)_n \right) \right) / t \right)}{\sum_c \exp \left(\mathcal{N} \left(\mathcal{H} \left(\xi_k^v \left(\bar{X}_k^v \right)_n \right) \right) / t \right)}, \quad (6)$$

where $\mathcal{H} \left(\xi_k^v \left(\bar{X}_k^v \right)_n \right) \in \mathbb{R}^{h \times n}$, h is the length of the feature vector, and $\mathcal{G}_k^v \in \mathbb{R}^{1 \times 1 \times c}$ represents the low-dimensional representation of high-dimensional multi-view features. Additionally, $\mathcal{N}(\bullet)$ represents a matrix containing normalization coefficients, which are adaptively optimized during the training process. $\mathcal{N}(\bullet)$ is capable of normalizing the projection matrix observations $\mathcal{H} \left(\xi_k^v \left(\bar{X}_k^v \right)_n \right)$, and t is the temperature parameter.

Since ξ_k^v needs to be optimized through the pseudo-label \mathcal{T}_k , the calculation method of \mathcal{T}_k is introduced here. The pseudo-label \mathcal{T}_k is correlated with the final output of the model, so every time the model updates its parameters, the SRDA parameter matrix ξ_k^v also requires an update. To ensure that the SRDA parameter matrix ξ_k^v remains correlated with the model, after performing gradient descent on $\mathcal{G}_k^v \left(\bar{X}_k^v \right)$, a forward pass is carried out with the updated weights and input parameters. The newly obtained parameters are treated as pseudo-labels, and these are used to update the parameter matrix ξ_k^v . The \mathcal{T}_k can be expressed as follows:

$$\mathcal{T}_k = \left[\xi_k^1 \bar{X}_k^1 - \eta \frac{\partial \mathcal{J} \left(\mathcal{G}_k^v; x, y \right)_1}{\partial \left(\mathcal{H} \left(\xi_k^1 \bar{X}_k^1 \right) \right)_1}, \xi_k^2 \bar{X}_k^2 - \eta \frac{\partial \mathcal{J} \left(\mathcal{G}_k^v; x, y \right)_2}{\partial \left(\mathcal{H} \left(\xi_k^2 \bar{X}_k^2 \right) \right)_2}, \dots, \xi_k^c \bar{X}_k^c - \eta \frac{\partial \mathcal{J} \left(\mathcal{G}_k^v; x, y \right)_c}{\partial \left(\mathcal{H} \left(\xi_k^c \bar{X}_k^c \right) \right)_c} \right]. \quad (7)$$

Here, x and y denote the data and label of the model, respectively. $\mathcal{J}(\bullet)$ represents the loss function of the proposed model, and η denotes the learning rate.

However, the SRAD clustering space solely relies on feature similarity for clustering. To enhance the diversity of feature fusion across views, in this paper, the disentanglement clustering space is constructed by clustering the feature

vectors $\mathcal{G}_k^v \in \mathbb{R}^{1 \times 1 \times c}$ corresponding to the view \overline{X}_k^v . This allows for the disentanglement of the clustered space, which improves the diversity of feature fusion across multiple views.

- Step 1: Sort the elements of the feature vector G_k^v and the corresponding views in descending order based on their magnitudes, resulting in a new feature vector $G_k^{v'}$ and new views $\overline{X}_k^{v'}$.
- Step 2: Cluster the views based on the sorted indices. To enhance cross-view information extraction, sample the view indices using a specified Minkowski distance.

Generally, the number of hidden layers in the model is set to 2^λ . To improve the applicability of the proposed method, the Minkowski distance for multi-view indices is also defined as 2^λ . The specific mathematical expression can be expressed as follows:

$$dist_{mk} \left(\left(\overline{X}_k^{v'} \right)_\alpha, \left(\overline{X}_k^{v'} \right)_\theta \right) = \left(\left| \left(\overline{X}_k^{v'} \right)_\alpha - \left(\overline{X}_k^{v'} \right)_\theta \right|^p \right)^{\frac{1}{p}}, \quad (8)$$

$$s.t. \quad dist_{mk} \left(\left(\overline{X}_k^{v'} \right)_\alpha, \left(\overline{X}_k^{v'} \right)_\theta \right) = 2^\lambda, \quad \left(\overline{X}_k^{v'} \right)_\alpha \notin \left(\overline{X}_k^{v'} \right)_\theta, \quad \theta \in \{\alpha + 1, \dots, k\}. \quad (9)$$

Here, $\left(\overline{X}_k^{v'} \right)_\alpha$ represents the view obtained from the α -th index, and $\left(\overline{X}_k^{v'} \right)_\theta$ denotes the views excluding the α -th index. In the disentanglement clustering space, each class view has an index distance of 2^λ in the corresponding information distribution sequence, resulting in greater feature diversity for each class view. Compared to fully connected convolution, grouped convolution, and deformable convolution in CNNs, feature extraction in this space enhances the diversity of feature fusion across views, reduces redundant features, and mitigates the risk of model overfitting.

Then, feature learning is performed in the disentangled clustering space, and this can be achieved using the following equation:

$$\mathcal{Z}_g^k(u, v) = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} \left(\mathcal{G}_k^v \left(\overline{X}_{i+u, j+v}^{v(k-1)} \right) \right)_g \mathcal{K}_{\text{rot}_{i,j}}^{(k)} + b^k \quad (10)$$

$\mathcal{Z}_g^k(u, v) \in \mathbb{R}^{w^v \times h^v \times 2g}$ is the final output of DLM.

It is worth noting that compared to previous feature extraction methods, feature extraction in the disentangled clustering space can significantly reduce computational complexity. Assuming the total number of views at the k -th layer is v , and there are c categories of views, the overall computational complexity of the disentangling learning module is:

$$O(DLM)_k = O \left(\mathcal{G}_k^v \left(\overline{X}_k^v \right) \right) + O \left(\mathcal{Z}_c^k(u, v) \right) \quad (11)$$

$$O \left(\mathcal{G}_k^v \left(\overline{X}_k^v \right) \right) = O \left(\frac{1}{2} v_k w_{v_k} h_{v_k} \min(v_k, w_{v_k} h_{v_k}) + \frac{1}{6} \min(v_k, w_{v_k} h_{v_k}) + h^2 \right) \quad (12)$$

$$O \left(\mathcal{Z}_c^k(u, v) \right) = O \left(\frac{w_{v_k} h_{v_k} k v}{c} \right) \quad (13)$$

$$O(DLM)_k = O \left(\min(v_k, w_{v_k} h_{v_k}) \left(\frac{1}{2} v_k w_{v_k} v_k + \frac{1}{6} \right) + h^2 + \frac{w_{v_k} h_{v_k} k v}{c} \right) \quad (14)$$

Given that O_{DLM} focuses on feature disentanglement and clustering, it avoids the extensive convolution operations typically required in CNNs, leading to $O_{\text{DLM}} \ll O_{\text{CNN}}$. This reduction in computational cost demonstrates the advantage of performing feature extraction in the disentangled clustering space compared to traditional CNN methods.

3.3. Collaborative learning module

Section 3.2 provides a detailed explanation of the specific computational methods used in the disentanglement learning module at a given scale. This subsection focuses on feature fusion across multiple scales to enhance the model's ability to recognize defect characteristics of varying sizes. The specific fusion approach is illustrated in Fig. 2, and it can be achieved using the following equation:

$$Out_{DLM} = \mathcal{Z}_g^k(u, v)_1 a_1 + \mathcal{Z}_g^k(u, v)_2 a_2 + \dots + \mathcal{Z}_g^k(u, v)_n a_n, \quad (15)$$

where $\{a_1, a_2, \dots, a_n\}$ represents the parameters obtained through model training. However, the feature information provided at different scales varies for different input samples. To ensure the adaptiveness of these parameters, they are derived from global feature calculations. Assuming the input feature map $F^a \in I^{C \times H \times W}$, global average pooling followed by a convolutional layer is initially employed to generate a global feature. Subsequently, the global feature information is compressed using two fully connected layers. Furthermore, to enhance the convergence of the model, the parameters a_i are normalized to the range of 0 to 1. The specific calculation formula is as follows:

$$\begin{cases} a'_i = \vartheta(Conv(GAP(F^{a_i}))) \\ a_i = \frac{\exp(a'_i)}{\sum_n \exp(a'_i)} \end{cases}, \quad (16)$$

where $GAP(\bullet)$ denotes the global average pooling operation, $Conv(\bullet)$ indicates the $Conv1 \times 1 + BN + ReLU$ operations, and $\vartheta(\bullet)$ represents the fully connected layer. After obtaining $\{a_1, a_2, \dots, a_n\}$, the remaining computation method is illustrated in Fig. 2.

3.4. Construction strategy of clustering loss

The gray-level co-occurrence matrix is a powerful tool for analyzing image texture features because of its robustness and stability. However, it cannot be directly incorporated into the loss function of neural networks for two reasons. First, the GLCM is calculated for a single image, and thus cannot be used to compare the texture differences between the two images directly. Second, the eigenvalue entropy of the GLCM typically falls between 1 and 10. Adding this value directly to the loss function often leads to challenges in reducing the model's gradient. To address these limitations, the properties of industrial production datasets are leveraged, and CL is proposed to enhance the GLCM method.

The grayscale co-occurrence matrix starts from the image element (x, y) with grayscale value i , and counts the frequency $\rho = (i, j|a, b, \theta)$, which co-occurs with the image element $(x + \alpha, y + \beta)$ at a distance d and with grayscale value j . The specific mathematical expressions are presented as follows:

$$\rho(i, j|\alpha, \beta) = \{(x, y)|f(x, y) = i, f(x + \alpha, y + \beta) = j\}, \quad (17)$$

$$(i, j) \in [0, L - 1].$$

Most previous studies have focused solely on scan maps with visible defects and even publicly available defective datasets consisting only of scan maps. However, the value of design maps without surface defects is often underestimated. Design maps data can provide a wealth of a priori knowledge for identifying defects, and acquiring such data is often straightforward in many manufacturing scenarios. Leveraging this property, a formula is constructed for calculating the entropy difference of the texture distribution of the samples as follows:

$$C_k = \frac{1}{L^2} \sum_{k=0}^{L^2} \left| \log(\rho_f(i, j)_k) - \log(\rho_s(i, j)_k) \right|, \quad (18)$$

where $\rho_f(i, j)$ denotes the grayscale co-occurrence matrix of the scanned image, $\rho_s(i, j)$ represents the grayscale co-occurrence matrix of the design image, and C_k denotes to the expected value of the information entropy metric distance between the grayscale co-occurrence matrix $\rho(i, j)$ of the real image $\rho_f(i, j)$ and the design image $\rho_s(i, j)$. The expectation value C_k and the true labels of the corresponding samples can be used to cluster each sample in the space, thus establishing the metric space of the samples. In the metric space, different data clusters represent the different metric coefficients that should be present in the loss function.

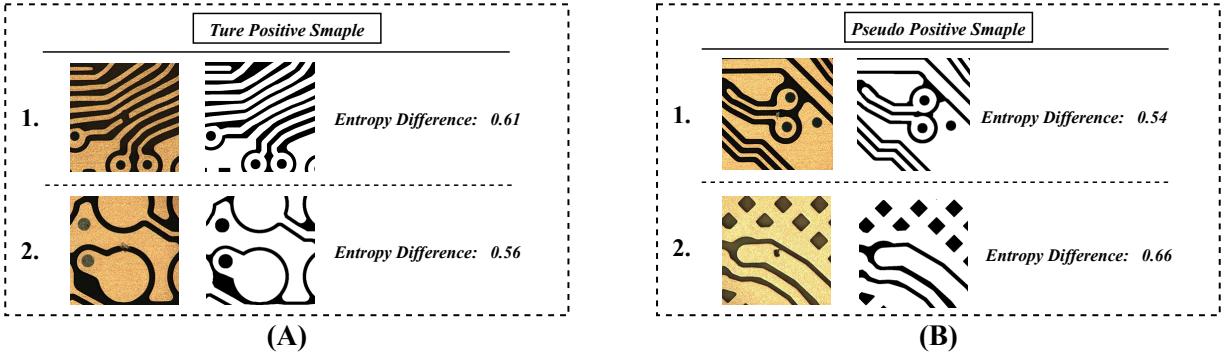


Fig. 3. (A) The entropy difference of true-positive defects samples. (B) The entropy difference of pseudo-positive defects samples.

For true-positive samples, the smaller the difference between the entropy of the scanned image and the design map (i.e., the texture distribution is similar), the higher the network's ability to fit it. As Fig. 3A shows, sample 2 has a data distribution similar to the standard graph and some pseudo-positive defect samples (e.g., sample 1 in Fig. 3B). However, most true-positive defect samples (e.g., sample 1 in Fig. 3A) have more prominent local data distributions. The model often struggles to fit sample 2 because it is labeled as a true-positive defect but resembles pseudo-positive samples. To address this issue, the entropy difference based on GLCM is used as prior knowledge, and the complexity of model training is reduced by increasing the optimization weight of the loss function for these samples. For pseudo-positive samples, the larger the difference between the entropy of the scanned image and the design map (i.e., the texture distribution is not similar), the higher the network's ability to fit this sample. Various metric values result in different metric coefficients for samples with different labels.

$$S_k = \frac{C_k - \text{Min}(C_k)}{\text{Max}(C_k) - \text{Min}(C_k)} + a, (a > 0), \quad (19)$$

$\text{Max}(\bullet)$ is used to find the maximum value in the sequence. $\text{Min}(\bullet)$ is used to find the minimum value in the sequence. a is a hyperparameter used to prevent S_k from going to zero and causing the gradient to vanish. It is necessary to normalize C_k such that $C_k \in (0, 1)$ and map it into a probability form to match the parameter distribution of the neural network.

Let $f^w(x)$ be the output of the neural network. The following probabilistic model is defined. For the binary classification task, the probabilistic output of the model is defined as follows:

$$\hat{y} = \sigma(f^w(x)) \quad (20)$$

$$\hat{y}_{\text{class}} = \begin{cases} 1, \hat{y} \geq 0.5 \\ 0, \hat{y} < 0.5 \end{cases}, \quad (21)$$

$\sigma(\bullet)$ is the sigmoid function. From Eq. (21), if the model output $\hat{y} \in [0, 0.5]$, it is judged to be a pseudo defect. If $\hat{y} \in [0.5, 1]$, it is judged to be a true defect. The log-likelihood equation for the model probability output \hat{y} is given by:

$$\delta = \left| \frac{\partial L_{CE}}{\partial x} \right| = |\hat{y} - y|, \quad (22)$$

the final loss function $CL(\hat{y}, y)$ is constructed as follows:

$$CL(\hat{y}, y) = \begin{cases} L_{CE}(\hat{y}, y)(1 - S_k)\delta\lambda_1, & \text{if } y = 1 \text{ and } \lambda_1 \neq 0 \\ L_{CE}(\hat{y}, y)S_k\delta\lambda_2, & \text{if } y = 0 \text{ and } \lambda_2 \neq 0 \end{cases} \quad (23)$$

The loss function applied to neural network training is given by the final constructive equation $CL(\hat{y}, y)$, where λ_1 and λ_2 are hyperparameters. The loss function utilizes the texture complexity entropy difference between the sample design map and the scan map to establish the loss metric space. It reduces the training task complexity caused by the imbalance in this type of dataset by enhancing MUCN's ability to fit samples with high texture differences.

4. Experimental setting and results

Several experiments are conducted, and the results are analyzed in detail. First, the datasets used for the experiments are discussed in Section 4.1. The experiment details and evaluation protocols are introduced in Section 4.2. In Section 4.3, the proposed model is compared with several well-known defect classification models on the PCB dataset. Section 4.4 summarizes the evaluation indicators, while Section 4.5 presents ablation experiments on the modules and CL. Finally, heatmap visualization is performed using gradient-weighted class activation mapping (Grad-CAM) [50].

4.1. Description of the PCB datasets

Considering the various forms of PCB data in practical applications (e.g., standard RGB images and PCB design diagrams with prior knowledge), this study uses the HRIPCB [51] and PCB-TP [52] datasets as case studies. The performance of the proposed MUCN method is evaluated and compared with other recently published methods using these two datasets, as shown in Fig. 4.

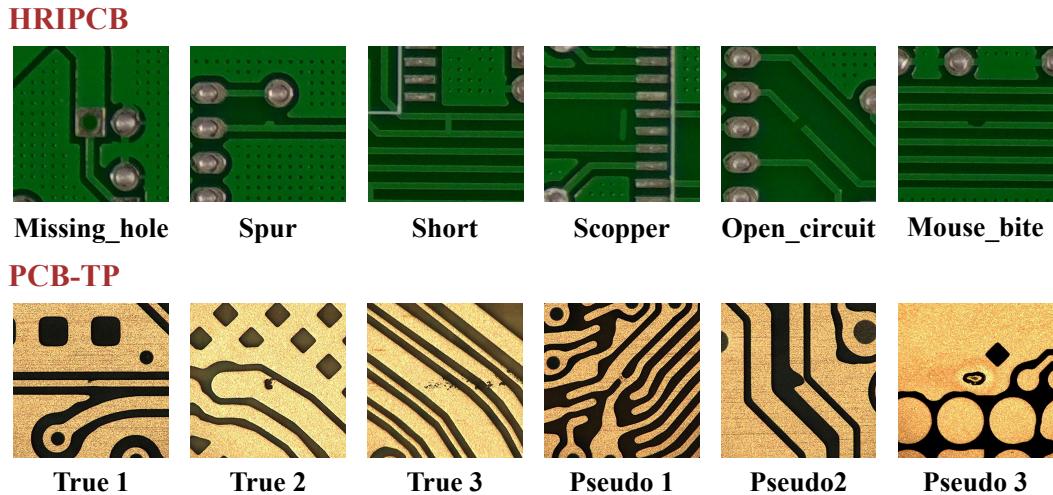


Fig. 4. Representative images of HRIPCB and PCB-TP datasets.

1) *Description of the HRIPCB Dataset:* The publicly available HRIPCB dataset is widely used in PCB defect classification studies. Acquired using the AOI imaging method, it contains 10,688 augmented images, each with a resolution of 4068×3456 pixels, onto which six defect types were manually added using Adobe Graphics editor. These defect types include “missing hole (hole),” “open,” “mouse bite (bite),” “spur,” “short,” and “spurious copper (Scopper).” All images in the dataset have been resized and standardized to 256×256 pixels.

2) *Description of the PCB-TP Dataset:* The PCB-TP dataset was collected from the practical manufacturing process. This dataset contains 20,000 sets of image data. Within this dataset, channels 1 to 3 correspond to images collected directly from the production line, whereas channel 4 represents the design standard map. PCB-TP includes “True,” and “Pseudo.” Each image is unified as 512×512 pixels. (In the download link of the dataset, the “datasets introduction” document will provide a more detailed introduction to the dataset) [52].

Table 1

Description of the PCB-TP & HRIPCB datasets.

HRIPCB Number	Hole 1832	Open 1740	Bit 1852	Spur 1752	Short 1732	Scopper 1760	Size $256 \times 256 \times 3$	Total 10,668
PCB-TP Number	True 10000			Pseudo 10000		Size $256 \times 256 \times 4$	Total 20,000	

The detailed information for the two datasets is summarized in Table 1, with representative images shown in Fig. 4. Each dataset is divided into a training set and a test set through random sampling, with 80% of the dataset allocated to the training set and the remaining 20% assigned to the test set.

4.2. Experiment details and baselines

1) *Implementation Details:* The input PCB image size is configured at 256 pixels \times 256 pixels. The entire network is trained using the Adam optimizer for 50 epochs, with a batch size of 32, an initial learning rate of 10^{-3} , and a decay factor of 0.99 for the learning rate. The performance of various defective classifiers is evaluated using several metrics based on the testing set. All experiments are implemented with Pytorch 1.10 framework and performed on a PC with GTX 3090 and i9-10900X CPU. Structure of the proposed MUCN for HRIPCB and PCB-TP datasets in Table 2.

Table 2

The parameters of the PCB-TP & HRIPCB dataset experiment.

		Block Name	Channels	Params	
Step 1	HRIPCB PCB-TP	Conv	3 \times 3, Stride=1	1.9k	
		MaxPool ₁	3 \times 3, Stride=2		
		DLM ₁ & CLM ₁	64		
		DLM ₂ & CLM ₂	128	0.28M	
		MaxPool ₂	3 \times 3, Stride=2		
		DLM ₃ & CLM ₃	128		
		DLM ₄ & CLM ₄	256	1.2M	
		MaxPool ₃	3 \times 3, Stride=2		
		DLM ₅ & CLM ₅	256		
		DLM ₆ & CLM ₆	512	4.7M	
Step 2		MaxPool ₄	3 \times 3, Stride=2		
		DLM ₇ & CLM ₇	512		
		DLM ₈ & CLM ₈	1024	18.6M	
		MaxPool ₅	3 \times 3, Stride=2		
		HRIPCB	PCB-TP		
		Block Name	Channels	Params	
Step 2		GAP	512	/	
		FC	512 to 6	1.5KB	
		Loss Function	CrossEntropy Loss (CE)	Clustering Loss (CL)	
		Total Params	6.3MB	24.8MB	

2) *Comparison with Well-Known Methods:* To further verify the effectiveness of the proposed MUCN, the following 12 convolution-based defect classification methods, along with state-of-the-art (SOTA) approaches, are considered as baselines: ResNet50 [53], FPN [54], SENet [55], ShuffleNetV2 [56], CS-ResNet50 [39], LS-ResNet50 [57], NLNet [58], DANet [59], GCNet [60], EPSANet [61], NCENet [62], and MCPC [63].

More details of the baseline models are provided to better demonstrate the advantages of the proposed MUCN: 1) ResNet50, which proposed a deep residual learning network that solves the degradation problem in deep neural networks by using residual blocks. 2) FPN, which is used for extracting feature maps of different scales. 3) SENet, which improves the feature extraction capabilities between feature channels. 4) ShuffleNetV2, which proposed an efficient and lightweight convolutional neural network. 5) CS-ResNet50, which proposed ResNet50 with the cost layer. 6) LS-ResNet50, which proposed ResNet50 with the label smoothing loss function. 7) NLNet, which proposed non-local operations as a generic family of building blocks for capturing long-range dependencies. 8) DANet, which proposed dual attention to integrate local features with their global dependencies adaptively. 9) GCNet, which proposed a simplified network based on a query-independent formulation. 10) EPSANet, which proposed a lightweight and effective pyramid squeeze attention module. 11) NCENet, which uses defects and surrounding information to distinguish defects. 12) MCPC, which proposed a multiscale convolution-based probabilistic classification methodology to classify defects under uncertainty.

4.3. Evaluation indicators

The performance of the proposed MUCN method is evaluated using accuracy, precision, and sensitivity rate. Precision indicates the accuracy of defect detection, while recall represents the detection rate of defects. High precision and sensitivity rates indicate strong detection performance, with precision highlighting the model's accuracy in identifying true positives and sensitivity demonstrating its ability to detect actual cases. A lower FLOP (Floating Point Operations Per Second) count generally suggests faster computations and reduced energy consumption, while fewer parameters contribute to a simpler model architecture, minimizing memory usage and improving deployment feasibility, especially in resource-constrained environments.

1. FLOPs: The total number of floating-point operations performed by the model.
2. Params: The total number of learnable parameters in the model.
3. True Positive (TP): The number of defective PCBs correctly classified as defective.
4. True Negative (TN): The number of non-defective PCBs correctly classified as non-defective.
5. False Positive (FP): The number of non-defective PCBs incorrectly classified as defective.
6. False Negative (FN): The number of defective PCBs incorrectly classified as non-defective.

The definitions of the evaluation metrics Precision and Sensitivity used for the experiments are presented as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100\%, \quad (24)$$

$$Precision = \frac{TP}{TP + FP} \times 100\%, \quad (25)$$

$$Sensitivity = \frac{TP}{TP + FN} \times 100\%. \quad (26)$$

The evaluation metric EH for the learning ability of distribution imbalance data is specifically defined as follows:

$$EH = \frac{\sum C_k}{len(C_k)}, k \in (y = 1, \hat{y} = 0), \quad (27)$$

where $len(\bullet)$ represents the number of elements when calculating C_k . The specific expression of C_k is given by Eq. (18), which measures the entropy difference between the real image and the standard image. Therefore, the greater the chaos in the data distribution of the two images, the smaller the entropy difference, indicating a lower similarity between the two images. In the case of true-positive defects, a larger entropy difference between the real image and the standard image makes it more challenging for the model to fit the data distribution.

4.4. Comparison with existing defect detectors

4.4.1. Results discussion on HRIPCB dataset

In this study, ten repeated experiments were conducted to evaluate the effectiveness of 13 methods across five measures—accuracy, precision, sensitivity, FLOPs, and the number of trainable parameters with experimental results shown in Figs. 5–8 and Table 3. For clarity, Table 3 highlights optimal accuracy results and minimal model parameters in bold, while suboptimal accuracy results, along with model FLOPs and parameters, are underlined.

Table 3

Performance of thirteen approaches in terms of five metrics on HRIPCB dataset.

Methods	Accuracy	Precision					Sensitivity					FLOPs ↓	Params ↓		
		Hole	Open	Bite	Spur	Short	Scopper	Hole	Open	Bite	Spur	Short	Scopper		
ResNet50(2016)	93.25%	95.67%	93.66%	91.26%	93.56%	92.57%	93.38%	92.96%	93.51%	93.57%	94.36%	91.57%	94.32%	2.07G	23.52M
FPN(2019)	94.93%	94.52%	95.23%	93.22%	94.17%	93.63%	95.94%	94.56%	94.24%	93.57%	94.07%	92.54%	91.62%	5.57G	29.7M
SENet(2019)	95.81%	100.00%	91.74%	98.03%	96.39%	100.00%	100.00%	100.00%	99.11%	96.23%	100.00%	95.00%	94.78%	0.65G	26.1M
ShuffleNetV2(2019)	96.12%	100.00%	92.37%	100.00%	100.00%	96.92%	98.07%	100.00%	100.00%	91.89%	98.18%	96.95%	100.00%	0.23G	1.3M
CS-ResNet50(2021)	92.65%	99.13%	98.88%	100.00%	92.86%	72.59%	100.00%	100.00%	77.19%	81.63%	99.05%	98.99%	99.09%	2.10G	24.21M
LS-ResNet50(2021)	94.01%	100.00%	98.94%	100.00%	80.14%	95.79%	96.52%	93.69%	92.08%	86.87%	100.00%	91.00%	99.11%	2.11G	23.65M
NLNet(2022)	97.72%	100.00%	93.50%	92.31%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	92.50%	87.52%	100.00%	5.22G	43.20M
DANet(2022)	97.41%	100.00%	97.85%	88.30%	98.89%	96.04%	100.00%	100.00%	89.22%	99.10%	95.70%	94.27%	100.00%	8.01G	82.18M
GCNet(2022)	98.69%	100.00%	91.35%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	94.64%	99.11%	98.32%	100.00%	3.86G	25.56M
EPSANet(2022)	98.59%	100.00%	96.59%	98.02%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	93.81%	99.05%	99.07%	4.72G	27.9M
NCEnet(2023)	98.44%	100.00%	97.32%	98.10%	95.10%	100.00%	100.00%	100.00%	100.00%	95.37%	100.00%	95.61%	100.00%	5.42G	60.01M
MCPCE(2023)	95.52%	95.77%	96.14%	95.26%	95.04%	96.15%	96.03%	96.24%	95.57%	96.23%	95.65%	94.88%	94.52%	0.52G	<u>1.77M</u>
MUCN	99.69%	100.00%	98.11%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	99.03%	100.00%	99.01%	100.00%	0.26G	6.3M

As shown in Table 3, the proposed MUCN method outperforms all other approaches with the highest overall accuracy of 99.69% and near-perfect precision and sensitivity values across all defect types. Notably, MUCN demonstrates 100% precision in detecting Hole, Bite, Spur, Short, and Scopper defects, as well as 100% sensitivity in all but one category. Furthermore, MUCN achieves these results with a remarkably low computational cost, boasting only 0.26G FLOPs, which is the second lowest among all methods, while maintaining a moderate parameter size of 6.3M. In contrast, models like ShuffleNetV2 and MCPC are highly efficient in terms of FLOPs (0.23G and 0.52G, respectively) and parameter count (1.3M and 1.77M, respectively), but they exhibit lower accuracy, with ShuffleNetV2 achieving 96.12% and MCPC reaching 95.52%.

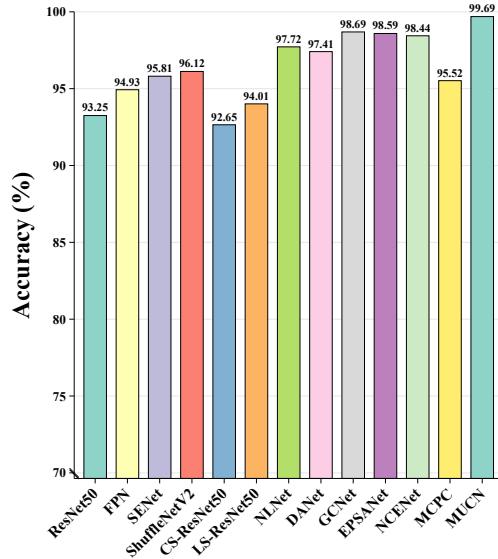


Fig. 5. Accuracy of various comparison methods on the HRIPCB dataset.

On the other hand, several other methods such as GCNet, EPSANet, and NLNet achieve accuracies exceeding 98%, but at the expense of higher computational demands, with FLOPs ranging from 3.86G to 5.42G and parameter counts exceeding 25M. DANet, despite achieving a high accuracy of 97.41%, requires the most computational resources with 8.01G FLOPs and 82.18M parameters, making it less suitable for resource-constrained environments. Earlier methods like ResNet50 and FPN, with accuracies below 95%, exhibit relatively higher computational costs compared to more recent models. Therefore, when considering deployment in resource-limited environments, methods like MUCN and ShuffleNetV2 offer the best trade-off between accuracy and efficiency, while models such as GCNet and DANet may be more suitable for scenarios where computational resources are abundant and higher accuracy is essential.

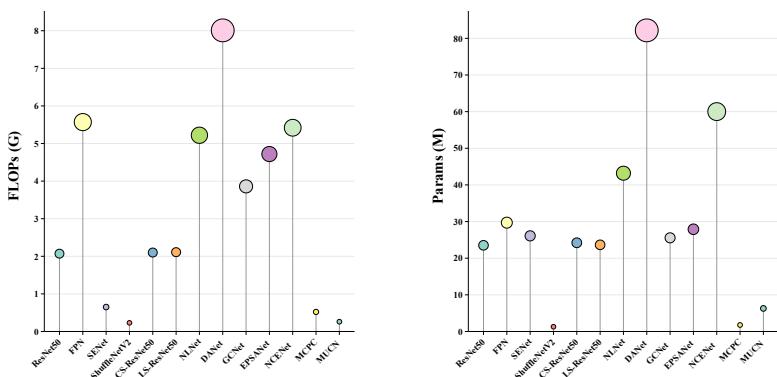


Fig. 6. FLOPs and parameters of various comparison methods on the HRIPCB dataset.

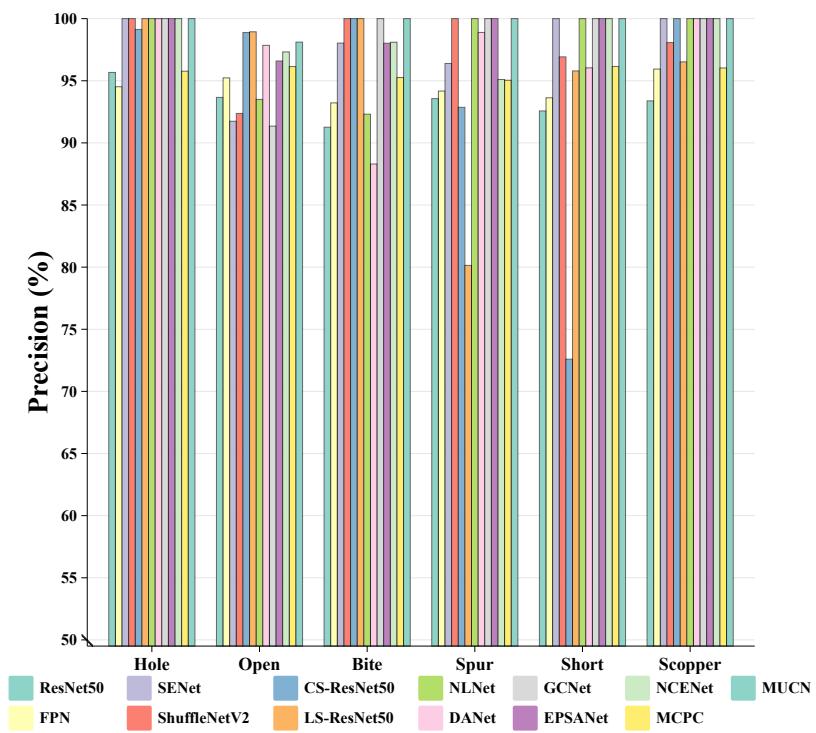


Fig. 7. Precision of various comparison methods on the HRIPCB dataset.

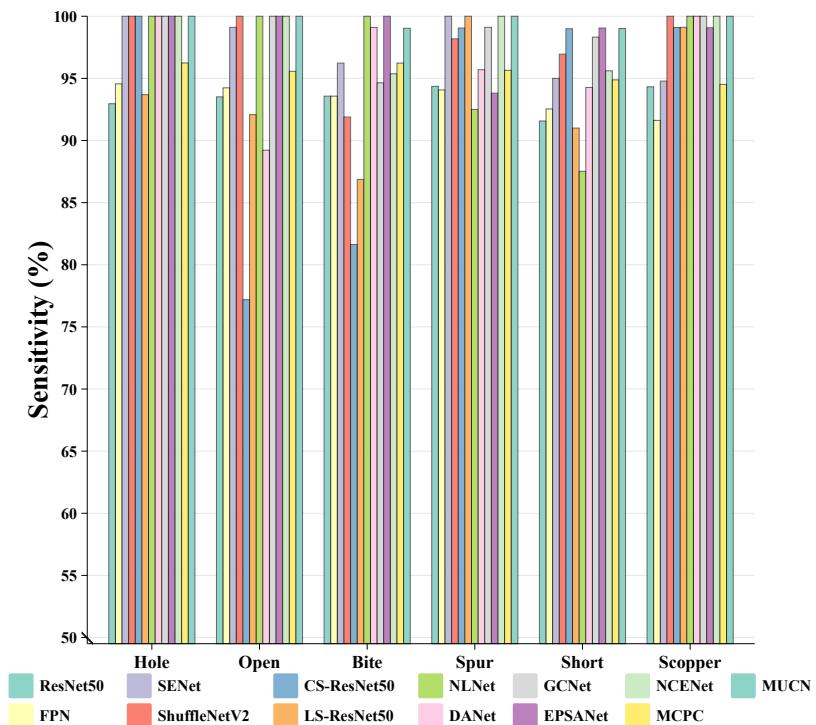


Fig. 8. Sensitivity of various comparison methods on the HRIPCB dataset.

4.4.2. Results discussion on PCB-TP dataset

The models (ShuffleNetV2, NLNet, DANet, GCNet, EPSANet, and NCENet) that perform well on the HRIPCB dataset are selected for comparison of evaluation metrics on PCB-TP. Table 4 presents a comprehensive comparison of six approaches evaluated on the PCB-TP dataset, using five key metrics: Accuracy, Precision (Pseudo and True), Sensitivity (Pseudo and True), FLOPs, Parameters (Params), and Energy Headroom (EH). Compared to the HRIPCB dataset, PCB-TP is more complex, with more samples exhibiting defect and non-defect texture similarities. As a result, the results demonstrate that the MUCN model outperforms all other methods, achieving the highest overall accuracy of 92.57%, followed closely by NCENet (91.21%) and EPSANet (90.92%). This indicates that MUCN provides the most reliable predictions across the dataset, as demonstrated by the visualization results in Fig. 9.

Table 4

Performance of seven approaches in terms of five metrics on PCB-TP dataset.

Methods	Accuracy	Precision		Sensitivity		FLOPs ↓	Params ↓	EH ↓
		Pseudo	True	Pseudo	True			
ShuffleNetV2	86.18%	82.53%	90.17%	91.66%	80.76%	0.23G	1.7M	2.71
NLNet	89.82%	87.95%	91.91%	92.45%	87.14%	5.22G	43.20M	2.66
DANet	89.96%	91.19%	88.79%	88.70%	91.25%	8.01G	82.18M	2.67
GCNet	89.69%	92.25%	87.50%	86.35%	92.94%	3.86G	25.56M	2.69
EPSANet	90.92%	91.22%	90.62%	90.54%	91.30%	4.72G	27.9M	2.65
NCENet	91.21%	90.82%	91.63%	91.86%	90.57%	5.42G	60.01M	2.62
MUCN	92.57%	92.44%	92.70%	92.76%	92.37%	0.29G	24.8M	2.59

Additionally, the evaluation protocol EH in Table 4 shows that the proposed method has the lowest value, indicating stronger fitting ability for samples with high similarity between functional and non-functional defects. Table 4 also provides model complexity, including FLOPs and the number of parameters (Params). While ShuffleNetV2 has the lowest FLOPs and Params, its accuracy is only 86.18%. GCNet demonstrates the highest sensitivity to pseudo defects but is 2.88% less accurate than the proposed model.

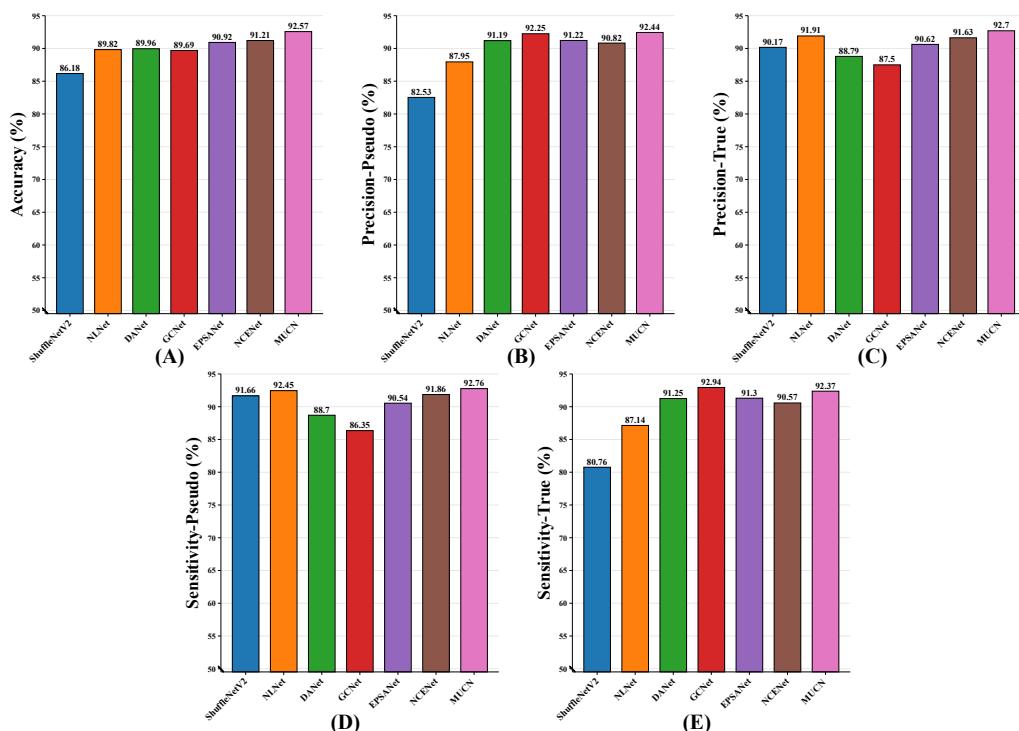


Fig. 9. Accuracy, precision, and sensitivity of various comparison methods on the PCB-TP dataset.

In terms of Precision, MUCN also leads, achieving 92.44% for the Pseudo-class and 92.70% for the True-class, which confirms its robustness in detecting both pseudo and true defective instances. In contrast, GCNet demonstrates strong precision for the Pseudo-class (92.25%), but its performance is notably lower for the True-class (87.50%). Sensitivity results further underscore the strengths of MUCN, which attains the highest sensitivity for the Pseudo-class (92.76%), while GCNet achieves the highest sensitivity for the True-class (92.94%).

An essential consideration in practical applications is computational efficiency. ShuffleNetV2 exhibits the lowest FLOPs (0.23G) and parameter count (1.7M), making it the most computationally efficient model. However, this efficiency comes at the cost of lower accuracy and sensitivity. MUCN, on the other hand, balances computational demands with performance, offering 0.29G FLOPs and 24.8M parameters while maintaining top-tier accuracy and sensitivity. Furthermore, MUCN achieves the lowest EH score (2.59), indicating an optimal balance between performance and energy consumption, as shown in the visualization results in Fig. 10.

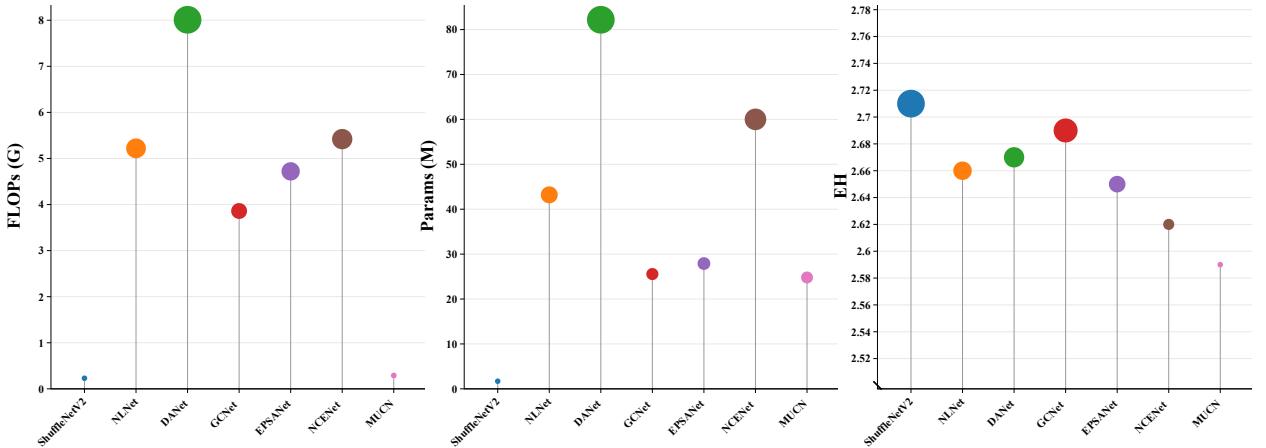


Fig. 10. FLOPs, parameters, and EH scores of various comparison methods on the PCB-TP dataset.

In summary, the proposed MUCN emerges as the optimal approach, delivering superior accuracy, precision, sensitivity, and computational efficiency. It is particularly suited for defect detection tasks where both performance and resource constraints are critical. While ShuffleNetV2 may be preferable in highly resource-constrained environments, the trade-off in detection performance must be carefully considered.

4.5. Ablation study

4.5.1. Ablation study of modules

This section presents several ablation experiments conducted on the HRIPCB dataset to assess the effectiveness of the proposed modules, including DLM, Multi-DLM, and CLM.

Table 5

Ablation study of modules.

Dataset	DLM	Multi-DLM	CLM	Accuracy
HRIPCB	NO.1	\	\	93.12%
	NO.2	\	\	98.73%
	NO.3	\	\	99.48%
	NO.4	\	\	99.69%

1) Impact of the DLM: To explore the positive impact of the DLM on recognition performance, a series of experiments are conducted by adding and removing the module. As shown in Table 5, when the MUCN model is trained without the DLM module, the baseline network (No.1) achieves an accuracy of 93.12%. Upon introducing the DLM, accuracy improves by 5.61%, increasing from 93.12% to 98.73%.

2) Impact of the Multi-DLM: The contribution of the Multi-DLM is examined by incorporating it into the baseline model with DLM (No.3), as presented in Table 5. Compared to the baseline, the No.3 configuration with Multi-DLM boosts accuracy by 0.75%, from 98.73% to 99.48%. This result demonstrates the vital role of Multi-DLM in MUCN and indicates its effectiveness in enhancing the feature fusion of defect local features and contextual information.

3) Impact of the CLM: The effect of the CLM module is also evaluated by adding it to the network. As shown in No.4 of Table 5, when CLM is incorporated into the network with DLM and Multi-DLM (No.3), the accuracy of MUCN increases from 99.48% to 99.69%. This confirms the effectiveness of the CLM in further enhancing the model's performance.

4.5.2. Ablation study of CL

Similarly, this section presents ablation experiments on the PCB-TP dataset to assess the effectiveness of the proposed CL. Since CL needs to be applied to datasets with both scanned and designed images, the ablation experiment was conducted using the PCB-TP dataset. Table 6 presents the results of the ablation study conducted on the PCB-TP dataset to evaluate the impact of adding the CL loss function to the MUCN model. The study compares two configurations: MUCN with DLM and CLM (with CE) and MUCN with DLM, CLM, and the additional CL loss function. The results indicate that incorporating the CL module increases model accuracy by 1.41% and 1.20%, respectively.

Table 6

Ablation study of CL.

Dataset	Methods	Accuracy	Precision		Sensitivity	
			True	Pseudo	True	Pseudo
PCB-TP	(MUCN) DLM + CLM + CE	91.37%	90.35%	92.50%	92.58%	90.22%
	(MUCN) DLM + CLM + CL	92.57%	92.44%	92.70%	92.76%	92.37%

The results demonstrate that incorporating the proposed CL loss function leads to a notable improvement across all evaluated metrics. Specifically, the overall Accuracy increases from 91.37% to 92.57%, indicating a more reliable performance in defect detection. Precision values also show an improvement, with the True-class precision rising from 90.35% to 92.44% and the Pseudo-class precision increasing from 92.50% to 92.70%. Similarly, Sensitivity metrics benefit from the inclusion of CL, with True-class sensitivity improving from 92.58% to 92.76% and Pseudo-class sensitivity rising from 90.22% to 92.37%. As shown in Fig. 11, the accuracy during the training process also demonstrates that CL accelerates the model's convergence and achieves higher defect detection accuracy.

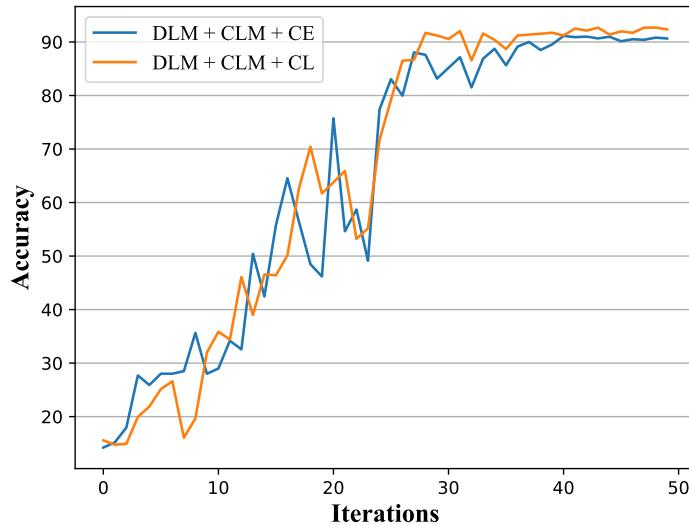


Fig. 11. Impact of CL in the PCB-TP dataset.

These improvements suggest that the CL loss function enhances both the model's precision in identifying defects and its sensitivity in detecting defective components. The enhanced precision and sensitivity across both true and Pseudo-classes demonstrate that the CL loss function enables the model to achieve more consistent and accurate predictions. This highlights the efficacy of the CL loss function in improving defect detection performance by refining the model's decision boundaries and reducing false predictions.

4.6. Network visualization with Grad-CAM

For qualitative analysis, Grad-CAM is applied to MUCN using images from the test dataset. Grad-CAM is a method that explains decisions made by deep learning networks by generating a heatmap for a specific class at each spatial location of the input image.

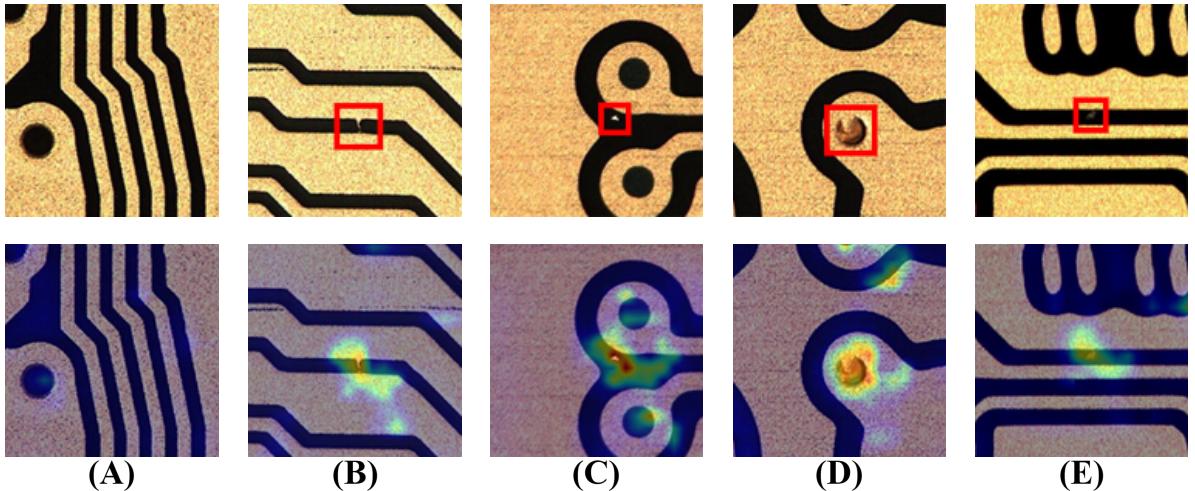


Fig. 12. Visualization of MUCN's performance on the bare PCB testing dataset using Grad-CAM.

By using Grad-CAM, the spatial regions that contribute to the final classification decision can be visualized. As shown in Fig. 12, the Grad-CAM masks generated by MUCN effectively highlight the coarse defect regions. For some tiny defects, MUCN can still extract relevant feature information for classification (as shown in Fig. 12E). A sample without defects was shown in Fig. 12A, which can be observed the model did not extract features incorrectly, resulting in classification errors. Moreover, the feature maps reveal that MUCN can effectively extract and emphasize the defect features by filtering out more invalid information and attenuating the irrelevant background and noise information, thus improving its ability to recognize defects.

5. Conclusions and future works

In this article, an adaptive Multi-view Unwrapping Collaborative Network (MUCN) is proposed. In particular, the DLM and CLM modules are designed to reduce computational complexity and mitigate overfitting caused by redundant correlation information in multi-view data, thereby enhancing the model's defect feature extraction capabilities. Furthermore, to reduce the complexity of model training, CL is introduced to cluster samples based on feature similarity. Experimental results demonstrate that the proposed MUCN outperforms SOTA methods on both the HRIPCB and PCB-TP datasets, exhibiting superior classification performance. Additionally, visualization using Grad-CAM confirms that MUCN can effectively learn to detect tiny defects.

Future work will focus on developing a highly efficient, precise, lightweight, and universal network model. AI-based visual defect detection technology plays a crucial role in advancing intelligent manufacturing in the electronic information industry, particularly in PCB quality inspection, where high precision, reliability, and customization are essential [8]. However, the rapid integration of next-generation technologies, such as edge computing and digital twins, remains a challenge that must be addressed to fully realize the potential of intelligent manufacturing [64].

CRediT authorship contribution statement

Feng Zhan: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing – original draft. **Weiming Gan:** Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing – original draft. **Lingkai Hu:** Methodology, Validation, Formal analysis, Investigation, Data curation, Writing – original draft. **Zhidi Huang:** Formal analysis, Resources, Supervision, Project administration. **Yi Cai:** Conceptualization, Methodology, Supervision, Writing – review & editing. **Hao He:** Writing – review & editing, Funding acquisition.

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.

Data availability

The data are available based on a request from the first author.

Acknowledgments

This work was supported by the Guangzhou-HKUST(GZ) Joint Funding Program (No. 2024A03J0680).

References

- [1] Tudi Huang, Tangfan Xiahou, Jinhua Mi, Hong Chen, Hong-Zhong Huang, and Yu Liu. Merging multi-level evidential observations for dynamic reliability assessment of hierarchical multi-state systems: A dynamic bayesian network approach. *Reliability Engineering & System Safety*, 249:110225, 2024.
- [2] Shun-Peng Zhu, Hong-Zhong Huang, Weiwen Peng, Hai-Kun Wang, and Sankaran Mahadevan. Probabilistic physics of failure-based framework for fatigue life prediction of aircraft gas turbine discs under uncertainty. *Reliability Engineering & System Safety*, 146:1–12, 2016.
- [3] Yayun Cheng, Li Zhang, Dalu Guo, Nannan Wang, Jiaran Qi, and Jinghui Qiu. Subregional polarization fusion via stokes parameters in passive millimeter-wave imaging. *IEEE Transactions on Industrial Informatics*, 2024.
- [4] Peisong Ma, Deshang Sha, and Debin Zhang. High efficiency and full range zvs single-stage semi-dab ac-dc converter with improved modulation and variable frequency control. *IEEE Transactions on Power Electronics*, 2024.
- [5] GAO Yuer, XU Tongqing, Zhongyuan Liao, and CAI Yi. Autonomous navigation algorithm for underwater robots based on global-local fusion under multi-objective optimization. *Authorea Preprints*, 2024.
- [6] Zhaoyi Xu and Joseph Homer Saleh. Machine learning for reliability engineering and safety applications: Review of current status and future opportunities. *Reliability Engineering & System Safety*, 211:107530, 2021.
- [7] Xing Chen, Yonglei Wu, Xingyou He, and Wuyi Ming. A comprehensive review of deep learning-based pcb defect detection. *IEEE Access*, 2023.
- [8] Yongbing Zhou, Minghao Yuan, Jian Zhang, Guofu Ding, and Shengfeng Qin. Review of vision-based defect detection research and its perspectives for printed circuit board. *Journal of Manufacturing Systems*, 70:557–578, 2023.
- [9] Dan Shi, Na Sun, Yanchi Liu, Cheng Lian, Xingyu Chen, Xiaoya Zhou, Xiaoyong Liu, Qiwei Liu, and Juejia Zhou. Fast calculation method for crosstalk of high-density printed circuit board by using machine learning. *IEEE Transactions on Electromagnetic Compatibility*, 2024.
- [10] Longxin Zhang, Jingsheng Chen, Jianguo Chen, Zhicheng Wen, and Xusheng Zhou. Ldd-net: Lightweight printed circuit board defect detection network fusing multi-scale features. *Engineering Applications of Artificial Intelligence*, 129:107628, 2024.
- [11] Yihao Xue, Rui Yang, Xiaohan Chen, Zhongbei Tian, and Zidong Wang. A novel local binary temporal convolutional neural network for bearing fault diagnosis. *IEEE Transactions on Instrumentation and Measurement*, 2023.
- [12] Jun Tang, Zidong Wang, Hongyi Zhang, Han Li, Peishu Wu, and Nianyin Zeng. A lightweight surface defect detection framework combined with dual-domain attention mechanism. *Expert Systems with Applications*, 238:121726, 2024.
- [13] Naifu Yao, Yongqiang Zhao, Seong G Kong, and Yang Guo. Pcb defect detection with self-supervised learning of local image patches. *Measurement*, 222:113611, 2023.
- [14] Furkan Ulger, Seniha Esen Yuksel, Atila Yilmaz, and Dincer Gokcen. Solder joint inspection on printed circuit boards: A survey and a dataset. *IEEE Transactions on Instrumentation and Measurement*, 72:1–21, 2023.
- [15] Ka Chun Fung, Kai-Wen Xue, Cheung-Ming Lai, Kwan-Ho Lin, and Kin-Man Lam. Improving pcb defect detection using selective feature attention and pixel shuffle pyramid. *Results in Engineering*, 21:101992, 2024.
- [16] Zhangwei Wang, Haiwen Yuan, Jianxun Lv, Chengxin Liu, Hai Xu, and Jinmeng Li. Anomaly detection and fault classification of printed circuit boards based on multimodal features of the infrared thermal imaging. *IEEE Transactions on Instrumentation and Measurement*, 2024.
- [17] Yihao Xue, Rui Yang, Xiaohan Chen, Baoye Song, and Zidong Wang. Separable convolutional network-based fault diagnosis for high-speed train: A gossip strategy-based optimization approach. *IEEE Transactions on Industrial Informatics*, 2024.

- [18] Arnaud Bougaham, Mohammed El Adoui, Isabelle Linden, and Benoît Frénay. Composite score for anomaly detection in imbalanced real-world industrial dataset. *Machine Learning*, 113(7):4381–4406, 2024.
- [19] Danqing Kang, Jianhuang Lai, Junyong Zhu, and Yu Han. An adaptive feature reconstruction network for the precise segmentation of surface defects on printed circuit boards. *Journal of Intelligent Manufacturing*, 34(7):3197–3214, 2023.
- [20] Minghao Yuan, Yongbing Zhou, Xiaoyu Ren, Hui Zhi, Jian Zhang, and Haojie Chen. Yolo-hmc: An improved method for pcb surface defect detection. *IEEE Transactions on Instrumentation and Measurement*, 2024.
- [21] EP Aribowo, AJ Santoso, A Nugroho, and PW Anggoro. Optimisation of automatic optical inspection on tht-pcb based on image segmentation and glcm. *Advances in Materials and Processing Technologies*, pages 1–22, 2024.
- [22] Dongxu Bai, Gongfa Li, Du Jiang, Junlong Yun, Bo Tao, Guozhang Jiang, Ying Sun, and Zhaojie Ju. Surface defect detection methods for industrial products with imbalanced samples: A review of progress in the 2020s. *Engineering Applications of Artificial Intelligence*, 130:107697, 2024.
- [23] Nianyin Zeng, Peishu Wu, Zidong Wang, Han Li, Weibo Liu, and Xiaohui Liu. A small-sized object detection oriented multi-scale feature fusion approach with application to defect detection. *IEEE Transactions on Instrumentation and Measurement*, 71:1–14, 2022.
- [24] Han-Bing Zhang, Chun-Yan Zhang, De-Jun Cheng, Kai-Li Zhou, and Zhi-Ying Sun. Detection transformer with multi-scale fusion attention mechanism for aero-engine turbine blade cast defect detection considering comprehensive features. *Sensors*, 24(5):1663, 2024.
- [25] Xinyi Yu, Li Han-Xiong, and Haidong Yang. Collaborative learning classification model for pcbs defect detection against image and label uncertainty. *IEEE Transactions on Instrumentation and Measurement*, 72:1–8, 2023.
- [26] Haixia Xu, Feng Han, Wei Zhou, Yanbang Liu, Fanxun Ding, and Jiang Zhu. Esmnet: An enhanced yolov7-based approach to detect surface defects in precision metal workpieces. *Measurement*, 235:114970, 2024.
- [27] Lei Lei, Han-Xiong Li, and Hai-Dong Yang. Reliable and lightweight adaptive convolution network for pcb surface defect detection. *IEEE Transactions on Instrumentation and Measurement*, 2024.
- [28] Yihao Xue, Rui Yang, Xiaohan Chen, Weibo Liu, Zidong Wang, and Xiaohui Liu. A review on transferability estimation in deep transfer learning. *IEEE Transactions on Artificial Intelligence*, 2024.
- [29] Seungchan Ko and Dowan Koo. A novel approach for wafer defect pattern classification based on topological data analysis. *Expert Systems with Applications*, 231:120765, 2023.
- [30] Jianzhong Zhang, Yongbin Wu, Zheng Xu, Zakiud Din, and Hao Chen. Fault diagnosis of high voltage circuit breaker based on multi-sensor information fusion with training weights. *Measurement*, 192:110894, 2022.
- [31] Jinhai Liu, He Zhao, Zhaolin Chen, Qiannan Wang, Xiangkai Shen, and Huaguang Zhang. A dynamic weights-based wavelet attention neural network for defect detection. *IEEE Transactions on Neural Networks and Learning Systems*, 2023.
- [32] Niegong Cao, Daehan Won, and Sang Won Yoon. Pads: Predictive anomaly detection for smt solder joints using novel features from spi and pre-aoi data. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 2024.
- [33] Gulhan Ustabas Kaya. Development of hybrid optical sensor based on deep learning to detect and classify the micro-size defects in printed circuit board. *Measurement*, 206:112247, 2023.
- [34] Nikolay Petkov and Malinka Ivanova. Detection and classification of defects on printed circuit board assembly through deep learning. In *2024 9th International Conference on Smart and Sustainable Technologies (SpliTech)*, pages 1–5. IEEE, 2024.
- [35] Yuhang Xu and Hua Huo. Dsaspp: Depthwise separable atrous spatial pyramid pooling for pcb surface defect detection. *Electronics*, 13(8):1490, 2024.
- [36] Dehua Zhang, Xinyuan Hao, Dechen Wang, Chunbin Qin, Bo Zhao, Linlin Liang, and Wei Liu. An efficient lightweight convolutional neural network for industrial surface defect detection. *Artificial Intelligence Review*, 56(9):10651–10677, 2023.
- [37] Xixi Han, Renpeng Li, Boqin Wang, and Zhibo Lin. Defect identification of bare printed circuit boards based on bayesian fusion of multi-scale features. *PeerJ Computer Science*, 10:e1900, 2024.
- [38] Dhruv Makwana, Sparsh Mittal, et al. Pcbsegclassnet—a light-weight network for segmentation and classification of pcb component. *Expert Systems with Applications*, 225:120029, 2023.
- [39] Huan Zhang, Liangxiao Jiang, and Chaoqun Li. Cs-resnet: Cost-sensitive residual convolutional neural network for pcb cosmetic defect detection. *Expert Systems with Applications*, 185:115673, 2021.
- [40] Xudong Wang and X Yu Stella. Tied block convolution: Leaner and better cnns with shared thinner filters. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 10227–10235, 2021.
- [41] Jiafeng Li, Ying Wen, and Lianghua He. Scconv: Spatial and channel reconstruction convolution for feature redundancy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6153–6162, 2023.
- [42] Jiaxiong Qiu, Cai Chen, Shuaicheng Liu, Heng-Yu Zhang, and Bing Zeng. Slimconv: Reducing channel redundancy in convolutional neural networks by features recombining. *IEEE Transactions on Image Processing*, 30:6434–6445, 2021.
- [43] Chenqiu Zhao, Guanfang Dong, Shupe Zhang, Zijie Tan, and Anup Basu. Frequency regularization: Reducing information redundancy in convolutional neural networks. *IEEE Access*, 2023.
- [44] Zhongmin Huangfu, Shuqing Li, and Luoheng Yan. Ghost-yolo v8: An attention-guided enhanced small target detection algorithm for floating litter on water surfaces. *Computers, Materials and Continua*, 80(3):3713–3731, 2024.
- [45] Qiulin Zhang, Zhuqing Jiang, Qishuo Lu, Jia'nan Han, Zhengxin Zeng, Shang-Hua Gao, and Aidong Men. Split to be slim: An overlooked redundancy in vanilla convolution. *arXiv preprint arXiv:2006.12085*, 2020.
- [46] Zhile Chen, Yuhui Quan, Ruotao Xu, Lianwen Jin, and Yong Xu. Enhancing texture representation with deep tracing pattern encoding. *Pattern Recognition*, 146:109959, 2024.
- [47] Yehao Li, Ting Yao, Yingwei Pan, and Tao Mei. Contextual transformer networks for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(2):1489–1500, 2022.
- [48] Jesse Redford and Brigid Mullany. Construction of a multi-class discrimination matrix and systematic selection of areal texture parameters for quantitative surface and defect classification. *Journal of Manufacturing Systems*, 71:131–143, 2023.

- [49] Heng-Da Cheng, Juan Shan, Wen Ju, Yanhui Guo, and Ling Zhang. Automated breast cancer detection and classification using ultrasound images: A survey. *Pattern recognition*, 43(1):299–317, 2010.
- [50] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.
- [51] Weibo Huang, Peng Wei, Manhua Zhang, and Hong Liu. Hripcb: a challenging dataset for pcb defects detection and classification. *The Journal of Engineering*, 2020(13):303–309, 2020.
- [52] PCB-TP Open Source Address, 2024. Available at: https://pan.baidu.com/s/1qUyi0_wH22jSmVwlFwQ0mA (Access code: dikc).
- [53] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [54] Runwei Ding, Linhui Dai, Guangpeng Li, and Hong Liu. Tdd-net: a tiny defect detection network for printed circuit boards. *CAAI Transactions on Intelligence Technology*, 4(2):110–116, 2019.
- [55] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.
- [56] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *Proceedings of the European conference on computer vision (ECCV)*, pages 116–131, 2018.
- [57] Wei Hou, Xian Tao, and De Xu. A self-supervised cnn for particle inspection on optical element. *IEEE Transactions on Instrumentation and Measurement*, 70:1–12, 2021.
- [58] Daxiang Li, Yang Li, and Ying Liu. Shoeprint image retrieval based on dual knowledge distillation for public security internet of things. *IEEE Internet of Things Journal*, 9(19):18829–18838, 2022.
- [59] Zhikui Chen, Han Wang, Suhua Zhang, and Fangming Zhong. Dual-attention network for few-shot segmentation. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2210–2214. IEEE, 2022.
- [60] Jingqi Huang, Yue Ning, Dong Nie, Linan Guan, and Xiping Jia. Weakly-supervised metric learning with cross-module communications for the classification of anterior chamber angle images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 752–762, 2022.
- [61] Hu Zhang, Keke Zu, Jian Lu, Yuru Zou, and Deyu Meng. Epsanet: An efficient pyramid squeeze attention block on convolutional neural network. In *Proceedings of the asian conference on computer vision*, pages 1161–1177, 2022.
- [62] Wenkai Huang, Jiafu Wen, Weiming Gan, Lingkai Hu, and Bingjun Luo. Neighborhood correlation enhancement network for pcb defect classification. *IEEE Transactions on Instrumentation and Measurement*, 72:1–11, 2023.
- [63] Lei Lei, Han-Xiong Li, and Hai-Dong Yang. Multiscale convolution-based probabilistic classification for detecting bare pcb defects. *IEEE Transactions on Instrumentation and Measurement*, 72:1–8, 2022.
- [64] Huangyi Qu and Yi Cai. An image segmentation algorithm for extracting melt pool parameters in wire arc additive manufacturing. In *International Conference on Computational & Experimental Engineering and Sciences*, pages 180–194. Springer, 2024.