ORIGINAL PAPER

V-DAFT: visual technique for texture image defect recognition with denoising autoencoder and fourier transform

Jongwook Si[1] · Sungyoung Kim[2]

Abstract

Texture is the surface qualities and visual attributes of an object, determined by the arrangement, size, shape, density, and proportion of its fundamental components. In the manufacturing industry, products typically have uniform textures, allowing for automated visual inspections of the product surface to recognize defects. During this process, texture defect recognition techniques can be employed. In this paper, we propose a method that combines a convolutional autoencoder architecture with Fourier transform analysis. We employ a normal reconstructed template as defined in this study. Despite its simple structure and rapid training and inference capabilities, it offers recognition performance comparable to state-of-the-art methods. Fourier transform is a powerful tool for analyzing the frequency domain of images and signals, which is essential for effective defect recognition as texture defects often exhibit characteristic changes in specific frequency ranges. The experiment evaluates the recognition performance using the AUC metric, with the proposed method showing a score of 93.7%. To compare with existing approaches, we present experimental results from previous research, an ablation study of the proposed method, and results based on the high-pass filter used in the Fourier mask.

Keywords Texture image · Defect recognition · Denoising autoencoder · Fourier transform · Image reconstruction

1 Introduction

Defect recognition can be categorized into two primary types in previous research, object defect recognition and texture defect recognition(detection). Both of these techniques are employed in industrial manufacturing and production processes to visually identify product defects. While these two research domains may appear to converge on similar objectives, each domain emphasizes a distinct aspect and utilizes a specific methodology tailored to that emphasis.

Object defect recognition is employed to pinpoint flaws in components or parts. Its primary function is to recognize anomalies related to the component's geometric configuration, dimensional accuracy, and material composition, all of which could compromise the overall functionality or safety integrity of the end product. On the other hand, Texture defect recognition is primarily utilized to identify defects on the surface texture of a product. It systematically assesses the characteristics of the surface textures on the products to identify defects, specifically focusing on anomalies in texture uniformity, variations in color, and distortions in the inherent pattern.

Texture defect recognition holds significant importance in the fields of image processing and computer vision, as textures offer valuable insights into patterns and structures within images [1]. Within the manufacturing industry, the identification of texture defects plays a pivotal role in evaluating product quality and uncovering flaws in the manufacturing process [2].

Upholding product specifications to predefined standards is of utmost importance in advanced manufacturing processes. Anomalies in surface textures can substantially degrade the end quality of products. The identification and rectification of these irregularities in texture patterns are pivotal for comprehensive quality assessment and assurance. This research proposes a novel methodology integrating

✉ Sungyoung Kim
sykim@kumoh.ac.kr

Jongwook Si
jwsi425@kumoh.ac.kr

[1] Department of Computer·AI Convergence Engineering, Kumoh National Institute of Technology, Daehak-ro 61, Gumi, Gyeongbuk 39177, Korea

[2] Department of Computer Engineering, Kumoh National Institute of Technology, Daehak-ro 61, Gumi, Gyeongbuk 39177, Korea

Fourier transform techniques with convolutional autoencoders to discern and rectify texture defects proficiently.

During fabrication, the manifestation of texture anomalies is not uncommon, underscoring the necessity for robust detection mechanisms to maintain product integrity. The incorporation of precise texture defect recognition strategies allows manufacturers to promptly diagnose discrepancies, facilitating immediate remedial measures. Additionally, such detection results in substantial economic benefits by curtailing the production of subpar products, diminishing material wastage, and enhancing operational throughput.

Leveraging deep learning and computer vision methodologies for the automated detection of texture anomalies offers a substantial advantage over traditional manual inspection methods, which are contingent upon individual subjective assessments. This automated approach not only streamlines the inspection process, decreasing time and monetary expenditures but also ensures a more rapid and objective criterion for defect recognition.

There are numerous ongoing studies focused on texture defect recognition. Most of these studies focus on enhancing recognition accuracy by integrating classification techniques into deep learning architectures. This has led to these recognition models becoming larger and more intricate, demanding the deployment of high-end equipment for both the training phase and real-time inference. In contrast, this paper proposes a novel method for detecting texture defects using a simple deep-learning model combined with image processing techniques to reduce the size and complexity of the recognition model while increasing recognition accuracy.

In this study, we integrate the Fourier transform and a denoising autoencoder architecture to recognize texture defects. The Fourier transform is an indispensable mechanism for analyzing the frequency spectrum of images or signals, facilitating the elucidation of texture frequency attributes. Since texture defects often induce discernible alterations within specific frequency bands, the Fourier transform emerges as a potent instrument for defect recognition. Concurrently, we leverage an autoencoder to ascertain the standard texture properties of the material and subsequently reproduce them. An autoencoder inherently encodes input data into a latent dimension and decodes it, thus extracting salient features. This autoencoder emulates regular texture patterns, generating a canonical reconstruction blueprint. This archetype aids defect recognition by discerning deviations between the regenerated texture and its baseline. The study provides a comprehensive explanation of the autoencoder's reconstruction process and a methodology for creating a normal reconstruction template for texture defect recognition, aiming to contribute fresh insights and advancements to the field.

In this paper, we present a method for Visual technique for texture image defect recognition with Denoising Autoencoder and Fourier Transform, which we define by the abbreviation V-DAFT. This study makes several significant contributions, which are outlined as follows:

- Performance Improvement: The proposed simple denoising autoencoder architecture has achieved a high level of performance that is comparable to state-of-the-art (SOTA) methods. Despite its simplicity, the autoencoder has proven effective in texture defect recognition. This demonstrates that efficient defect recognition can be attained without complex deep-learning models.
- Integration of Techniques: This study introduces a hybrid approach combining deep learning and image processing methodologies to address texture defect recognition. Specifically, deep learning is applied to accomplish denoising and reconstruction tasks, while image processing methods are utilized to extract relevant texture features and facilitate defect recognition. This fusion of techniques offers several advantages, including eliminating extensive data training. As a result, this study presents a streamlined and efficient methodology for texture defect recognition.
- Experimentation and Analysis: The study conducts comprehensive experiments and analyses, meticulously adjusting various parameters to optimize texture defect recognition performance. These experiments provide insights into the parameters that most significantly influence performance and offer practical guidelines for real-world applications. Through these contributions, this study not only showcases the potential for detecting texture defects but also highlights the advantages of combining deep learning and image processing techniques.

2 Related works

In anomaly detection research, methods grounded in reconstruction have garnered substantial attention. These approaches typically involve training on normal data to generate reconstructed data and then leverage the disparities between the input anomalous data and the original image for detection. Frequently, models such as autoencoders and generative adversarial networks (GANs) are employed for reconstruction and the utilization of reconstruction errors for anomaly detection.

2.1 Anomaly detection

For instance, AnoGAN [3] introduced a fundamental approach to anomaly detection, combining unsupervised learning with GAN techniques. It learns the distribution of normal data by inputting only normal data and calculates anomaly scores to identify anomalies. Building upon

this, f-AnoGAN [4], an extension of AnoGAN [3], enhances performance through the incorporation of fast mapping techniques for new data and the inclusion of an Encoder within the GAN for more refined reconstruction. The data generated by f-AnoGAN [4] exhibit impressive generation performance, often challenging even experts to distinguish them from real data. GANomaly [5] takes a comprehensive approach, learning both generation and latent spaces using only normal data, while anomaly scores are computed based on differences in latent vectors. Skip-GANomaly [6] further extends GANomaly [5] with a U-Net-based network architecture and introduces adversarial training, which incorporates a loss function for the discriminator's feature maps, ultimately leading to improved reconstruction performance. MemAE [7] addresses the limitations associated with using autoencoders for anomaly detection by incorporating a memory module. This unique addition makes reconstruction more challenging for abnormal samples, which is particularly advantageous when autoencoders might otherwise reconstruct abnormal regions, a drawback that this study focuses on addressing. OCGAN [8] presents a model for one-class anomaly detection, which learns latent representations of in-class examples and confines the latent space to a given class. By utilizing a denoising autoencoder network and discriminator, it generates in-class samples and explores anomalies beyond class boundaries, achieving high-performance results.

These studies predominantly concentrate on reconstruction-based methods for anomaly detection, wherein training on normal data is the key to assessing and detecting anomalies. Anomaly detection extends into various subfields [1] and [9], encompassing applications such as disease recognition, accident recognition, and fall recognition.

2.2 Defect detection

Tsai et al. [1] proposed the use of the Fourier transform to detect defects in printed circuit boards. This study demonstrated the ability to detect small irregular pattern defects by comparing the Fourier spectra of an image and a template, thus highlighting the effectiveness of this method. While drawing inspiration from the concept of using templates, this approach differs from the one proposed in this study by forgoing Fourier spectrum comparison and instead incorporating deep learning elements. Tsai et al. [1], presented research which relied on templates, this study demonstrates that improved performance can be achieved by focusing solely on component removal. Consequently, this study introduces additional methods and achieves performance enhancements for various textures.

DRAEM [2] introduced a defect recognition method that deviates from the conventional approach of training on normal data for reconstruction. It concurrently learns two networks for reconstruction and discrimination to preserve and

detect defective regions. However, the objective of this study is to achieve performance improvement using a straightforward approach by training exclusively on normal data without generating anomalous data, which might result in slightly lower performance compared to the former approach. Liang et al. [10] noted the limitations of the reconstruction capabilities of other methods and introduced a defect recognition method from a frequency perspective, aligning with the viewpoint and approach of this study. They propose two novel methods, frequency decoupling, and channel selection, to reconstruct from various frequency perspectives and combine them for more accurate defect recognition. N-Pad [11] introduced a defect recognition method that uses the relative positional information of each pixel. Relative positional information is represented in eight directions, and a loss function is employed to demonstrate the utilization of this positional information. An anomaly score is proposed using Mahalanobis and Euclidean distances, and various experiments on neighborhood sizes demonstrate the significance of the method.

Si et al. [12] directed their focus toward applying reconstruction techniques to thermal images of solar panels for defect recognition. Given that the distribution of thermal images is sensitive to color and lacks pronounced edge features, this study proposes a method that uses patches instead of reconstructing the entire image. The proposed method introduces a technique called the "difference image alignment technique" by sorting pixel values, which enables easy recognition of defects using only specific pixels. However, the application of this approach may not be straightforward due to significant differences in data characteristics between the focus of this study (manufacturing) and thermal images. Tsai et al. [13] introduced a defect recognition method that considers the similarity between patches to extract representative and important information from images. Using patches of different sizes, the method performs representation learning at different scales and applies K-means clustering and cosine similarity to improve defect recognition. While this method detects both object and texture, it exhibited lower performance, particularly for texture recognition, as this study primarily focuses on texture.

Liu et al. [14] proposed a method for enhancing defect recognition performance in grayscale images through post-processing techniques, including color space and image processing. The network was designed to reconstruct the original colors using grayscale images to prevent the incorrect classification of color information, resulting in improved performance through the incorporation of various augmentation techniques and morphologies. Shi et al. [15] stand out from most other studies that concentrate on image reconstruction. This study utilizes a pre-trained model to extract feature maps from various layers, combines them, and performs reconstruction to better restore the features. By basing

all content on diverse feature maps, the method can better preserve defective regions in the results. Hou et al. [16] introduced a divide-and-assemble approach to overcome the limitations of autoencoder models for unsupervised anomaly detection. This approach involves modulating the model's reconstruction capability and introduces a multi-scale block-wise memory module, adversarial learning, and meaningful latent representations to enhance anomaly detection performance, demonstrating significant improvements in anomaly detection.

Patch SVDD [17] is a defect recognition method that extends the traditional SVDD algorithm into a patch-based deep learning approach. It involves constructing input patches from arbitrary portions of the image, rather than the entire image, for the deep learning model, and proposes a method of integrating multiple-sized Anomaly Maps. Uniformed Student [18] approach embeds the Receptive Field for defect recognition using both Teacher and Student modules, calculating the Regression error and considering various Multiscale scenarios. This method identifies anomalies when the Student's output differs from that of the Teacher. The final results, obtained by utilizing diverse Receptive Field sizes, demonstrate high-performance defect recognition. Notably, the Multiscale approach shares similarities with Patch-SVDD [17], while the proposed method in this paper distinguishes itself by not employing Multiscale but instead focusing on detecting defects through a single scale. DAGAN [19] proposes a method to address the issue of data imbalance in the domain of defect recognition by utilizing two consecutive deep learning models trained through adversarial generative learning for decision making. This approach relies solely on the use of these two models for recognition, leading to the analysis that there is a significant emphasis on the deep learning network itself, as observed when comparing our research findings. PEDNet [20] is a novel approach that consists of three models: Patch Embedding, Density Estimation, and Local Prediction. It proposes a method to project data into an Embedding Space and detect anomalies therein. To effectively represent this embedding space, dimension reduction and patch embedding processes are employed, followed by prediction through clustering. Zhang et al. [21] introduced a distillation approach for various features and proposed a system for anomaly and defect recognition using a pretrained model on ImageNet. Despite significant differences among multiple datasets, their approach demonstrated a common capability for recognition.

3 Texture defect detection

Noise is an inevitable factor that emerges during image processing, posing a significant challenge to image analysis and defect recognition, especially in the context of defect identifi-

cation. Noise can complicate the precise delineation of defect areas, potentially leading to elevated recognition scores and false positives during defect identification.

In this study, we introduce a defect recognition methodology employing deep learning networks and Fourier transformation. The proposed approach encompasses the following sequential steps:

3.1 Generation of reconstructed images through denoising

Initially, a deep learning network is deployed to execute a denoising process on the input images. This process yields reconstructed images from which fine details are removed. The network utilized in this step is a simple denoising autoencoder featuring a straightforward structure exclusively trained on normal images. The primary focus here is to generate images that closely resemble the input data, with a paramount emphasis on noise elimination.

3.2 Preservation of defects through fourier transform

Defect recognition transpires during the inference phase. This process entails the creation of normal reconstructed templates using a dataset of normal experimental data. The term 'normal reconstructed template' denotes a collection of results generated by a model trained solely on normal images. This approach leverages the reconstruction mechanism of an autoencoder, acknowledging the constraints in precisely replicating the original imagery from normal videos. To overcome these limitations, the generated outcomes are considered as representative of the normal distribution. Consequently, by employing the normal reconstructed template to construct differential images, it becomes possible to eliminate the normal distribution and accentuate only regions containing defects. A specific template is selected, and the same Fourier transformation process is applied to it. Examination of the Fourier spectrum reveals that a region affected by defects exhibits marked dissimilarity compared to a normal region. Significantly, the region afflicted with defects displays a pronounced energy concentration, particularly within the high-frequency domain. To exclusively preserve the high-frequency components corresponding to the defective region, a high-pass filtering operation is executed. Subsequently, an inverse Fourier transform is applied to generate an image corresponding specifically to the high-frequency component.

Figure 1 illustrates an example of testing the accuracy of defect detection within a high-frequency texture background. The first image is the original image, while the second and third images are the results after applying a Fourier transform, selectively removing certain frequency bands, and then
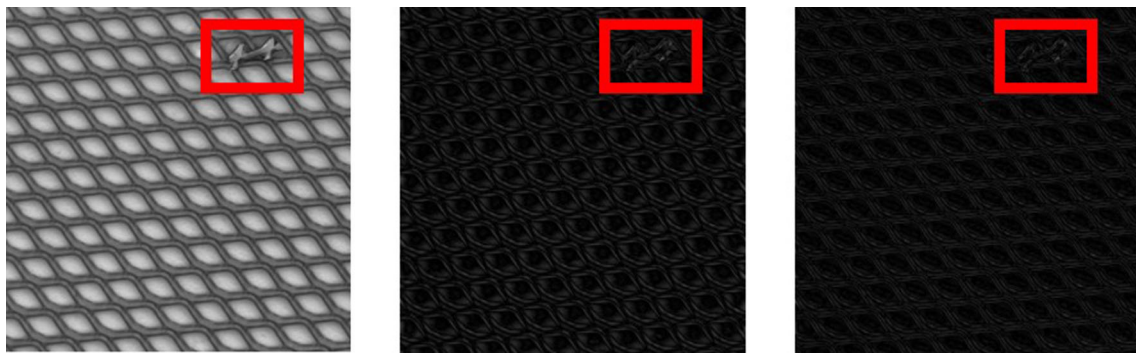
Fig. 1 Examples of fourier transform in high-frequency textures

performing an inverse transform. Notably, the third image retains more of the high-frequency components compared to the second image. These results demonstrate that, even in textures comprised of high frequencies, there exists a subtle yet distinct difference in frequency bands between defects and the background.

3.3 Generation of difference images and binary-level thresholding

The computation of difference images between each high-pass-filtered normal reconstructed template and the input image is the next step. In these difference images, pixels with nonzero values are regarded as potential defect regions, and higher values indicate a greater likelihood of defects. A binary image is derived from the difference image through binary thresholding. This approach facilitates the creation of a defect score by representing items with a potential for defects as one and those with a lower likelihood as zero in the binary image. By generating a binary image, the defect score is formulated by tallying the occurrences of the value one. A higher score signifies a greater likelihood of the presence of defects. In binary images, pixels corresponding to defects exhibit values distinct from those of normal pixels. The binarization process effectively accentuates and represents the defect regions.

The proposed methodology facilitates defect recognition using the above-outlined steps. This process allows differentiation between normal and defective images, effectively highlighting the regions where defects are present. The overall workflow of the proposed method is depicted in Fig. 2.

3.4 Normal texture image reconstruction

The model processes input images sized at 256×256 pixels, composed of RGB color channels. Consequently, the input shape is (256, 256, 3), with a depth of five layers. The autoencoder model comprises an encoder and a decoder. Input data are compressed into a low dimensional latent vector through the encoder and then reconstructed to the same size as the input image via the decoder. Figure 3 illustrates the structure of the reconstruction model proposed in this paper for noise removal.

The encoder part follows a convolutional neural network (CNN) structure. The first convolutional layer employs 64 filters, each utilizing a 3×3 kernel, the ReLU activation function, and same padding to maintain output size. Subsequent convolution layers extract spatial features from the image and gradually reduce its size. The decoder receives the output of the encoder for the restoration process. It uses up-sampling layers to match the size of each layer in the encoder, employing a skip-connection structure to combine information from each encoder layer. This process generates high-quality output results of the same size as the input image.

The model's loss function Eq. 1 combines basic L1 and L2 losses. The L1 loss calculates the absolute error between actual and predicted values, while the L2 loss calculates the squared error between them. By combining these two losses, the reconstruction loss is computed and minimized during model training. This combined loss function captures various aspects of errors in a balanced manner, allowing the model to effectively learn and optimize its parameters. In conventional autoencoders, the typical objective is to train the model to reconstruct the input to be exactly the same as the output, often resulting in complex model architectures and loss functions. However, in this paper, the autoencoder is utilized solely for the purpose of removing noise from input images, necessitating a simpler structure and loss function. In other words, a straightforward architecture and loss function do not aim to make the output identical to the input but rather focus on modeling the overall distribution of the input. The L1 and L2 loss functions employed in this context are fundamental functions leveraged inversely to indicate that fine-grained reconstruction of individual components is not the primary goal. The L1 Loss is robust when it comes to preserving detailed information, making it effective for noise removal. However, using L1 alone can potentially remove regions of interest, including normal areas. Therefore, the approach here involves incorporating an appropriate weight
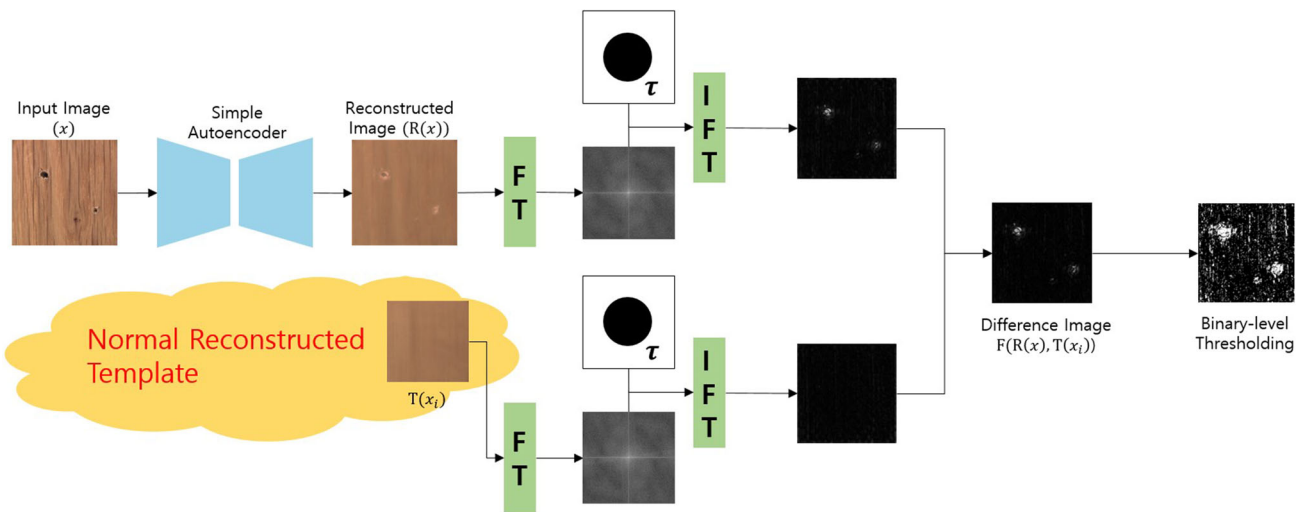
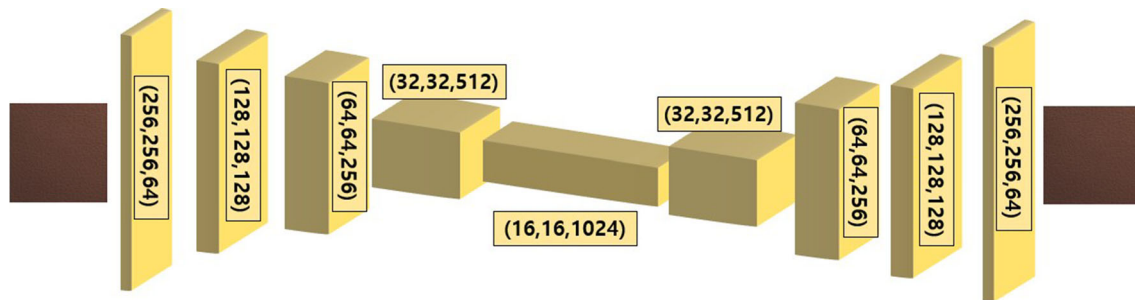**Fig. 2** Overall architecture of the proposed method

Fig. 3 Model architecture for reconstruction

for the L2 Loss, specifically to target noise removal in the background while preserving the relevant content.

$$\mathcal{L}_{recon}(x, R(x)) = \lambda_{L1} \sum_{i=1}^{N} |x_i - R(x_i)|$$
$$+ \lambda_{L2} \sum_{i=1}^{N} (x_i - R(x_i))^2 \quad (1)$$

In this study, the simple denoising autoencoder structure was utilized for defect recognition tasks. The primary objective is to detect defects using this structure, which effectively compresses and reconstructs input data, enabling differentiation between normal and defective data. Moreover, this structure introduces denoising effects. The encoder part extracts features from the input image and removes noise, reducing noise in the input data. Noise refers to elements that can be misclassified as defects in the texture, such as patterns in the backgrounds of normal images. Removing noise results in a reconstructed image with a nearly uniform color, representing a normal distribution. This aids in distinguishing defects or normal regions in the frequency band. With

reduced noise, input data represent clearer and more accurate frequency bands, making it easier to distinguish defect areas from the results of the Fourier transformation. Consequently, the simple denoising autoencoder structure with denoising effects is well-suited for Fourier transformation and tasks such as defect recognition and frequency-band division.

3.5 Application of fourier transform

The Fourier transform plays a crucial role in this study, as it allows the transformation of a 2D image from the spatial domain into the frequency domain. This transformation provides valuable information regarding the image's frequency components. The Fourier transform process entails converting the original image into the frequency domain, conducting necessary operations and restoring it to the original domain through an inverse transformation.

First, the given 2D square image (256, 256) is represented in the spatial domain using (x, y) coordinates. The Fourier transform is applied to this image using Eq. (2). In this equation, $F(u, v)$ represents a complex number in the fre-

quency domain and (u, v) represents the coordinates in the frequency domain. Equation (2) illustrates the multiplication of a complex exponential function in the frequency domain with the image values at each spatial domain position, followed by summation across all positions. This procedure yields frequency information about the image in the frequency domain.

$$F(u, v) = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) \cdot e^{-j2\pi\left(\frac{ux+vy}{N}\right)} \tag{2}$$

The inverse Fourier transform, as defined in Eq. (3), exhibits a critical property. It ensures that when the original image undergoes transformation and subsequent inverse transformation, it is restored to the original image, highlighting the relationship between the Fourier transform and its inverse.

$$f(x, y) = \frac{1}{N^2} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} F(u, v) \cdot e^{j2\pi\left(\frac{ux+vy}{N}\right)} \tag{3}$$

In this study, multiple reconstructed results denoted as $T(x_i)$ are generated for a "normal reconstructed template" representing normal data. To detect the presence of defects, both the target image under evaluation and the normal reconstructed template undergo Fourier transformation, converting them into the frequency domain. Within the frequency domain, the component at frequency 0 is centered, with higher frequencies shifting towards the edges through a frequency shift process.

Following this, a "Fourier Mask" is introduced to eliminate low-frequency components, as depicted in Fig. 4. An ideal mask with a radius of τ, centered at the origin, is created for this purpose. This mask is employed to perform pixel-wise operations on the Fourier-transformed result, effectively removing low-frequency components. Consequently, only the high-frequency region, where defects are present, is retained, while background and unnecessary components are eliminated. Subsequently, an inverse transformation is applied to convert the results from the frequency domain back to the spatial domain, preserving only the defective regions of the texture image in the spatial domain.

3.6 Difference images and thresholding

The process involves the generation of two transformed results, denoted as $F(R(x))$ and $F(T(x_i))$. These outcomes represent the removal of low-frequency components while preserving high-frequency components for the "normal reconstructed template" Therefore, the normal reconstructed template retains fine high-frequency details while eliminating the rest. When this process is applied to a normal image as input, the difference from the normal reconstructed template is minimal. However, for an image with defects as input, the defective regions remain unaltered, and some background noise may persist. Subtraction of these two generated images results in a difference image with nonzero values in the defective regions and values close to zero in unaffected areas. This difference image serves as the basis for creating the final map for defect recognition. Subsequently, a threshold value (th) is applied to the generated map to produce final maps that indicate the presence or absence of defects. Values exceeding the threshold are set to one, while those below it is set to zero. This binary map effectively distinguishes defect presence (set to one) from absence (set to zero). To fill empty areas in the binary image, a dilation operation with a (5,5) kernel is performed three times. The proposed defect score is calculated as a normalized value based on the count of pixels set to 1 in a binary image. The decision threshold for determining the presence of defects is identified at the point where the difference between the True Positive Rate (TPR) and the False Positive Rate (FPR) across each category is maximized.

The binary image obtained through this process marks one in locations with defects and zero in unaffected areas, facilitating defect recognition. The calculation of scores for the entire image, followed by normalization, leads to the deter-
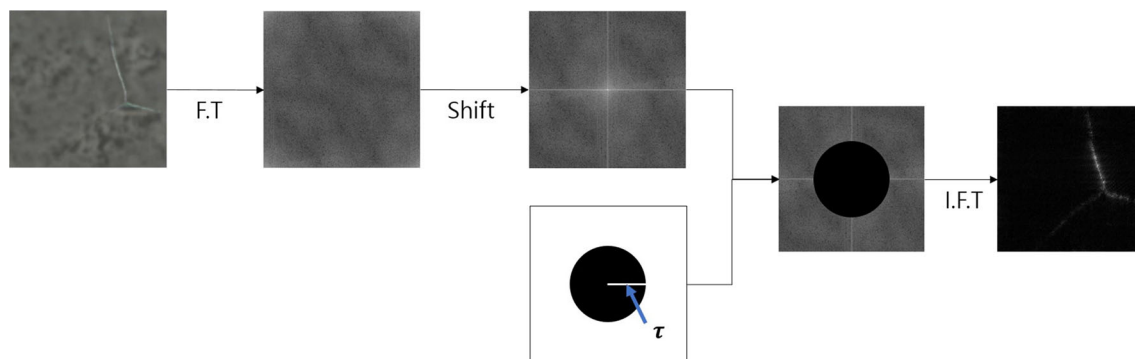

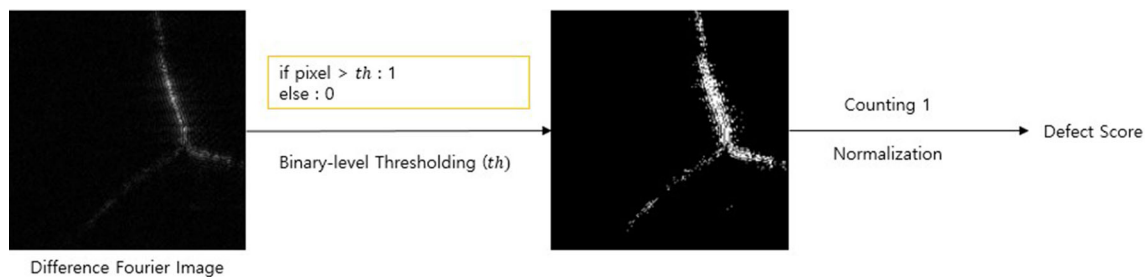**Fig. 4** Fourier mask and pixel-wise

Fig. 5 Procedure for calculating the defect score

Table 1 Detailed datasets with data augmentation

Category	Train(Normal)	Test(Normal)	Test(Defect)
Carpet	280	84	89
Grid	264	63	57
Leather	245	96	92
Tile	230	99	84
Wood	247	57	60

mination of the normalized defect score. Figure 5 illustrates the process of calculating the defect score.

4 Experimental results

4.1 Datasets

For texture defect recognition in this study, we utilized the MVTec-AD dataset [22], comprising five textures and ten objects. However, our evaluation focused on the five textures. Due to data limitations for both training and testing, data augmentation was employed. Table 1 provides details on the dataset composition, including the number of samples in each texture category. The majority of the data were used for training since we exclusively used normal data for this purpose. To balance the sample count between normal and defective data, we applied augmentation techniques to normal data. This process not only increased training data diversity but also ensured an adequate number of defective samples, thereby enhancing model performance.

4.2 Training details

In the context of texture defect detection, we adopted the following training approach. We normalized the pixel values of the input image data to fall within the 0 to 1 interval. To generate diverse training data, we performed data augmentation, which encompassed techniques such as shearing (20% magnitude), zooming (20% magnification), and both horizontal and vertical flipping. Training employed the Adam optimizer

with an initial learning rate of 1e-4 and was conducted over 500 epochs on the entire dataset, with a batch size of 16. The loss function included the hyperparameter $\lambda_L 2$ set to 100 for L2 loss and $\lambda_L 1$ set to 1 for L1 loss, resulting in a combination of these simple loss functions.

4.3 Performance evaluation and ablation study

The neural network utilized in this study was exclusively trained on normal data, leading to generated images with different distribution characteristics when processing images containing defects. Figure 6 provides a visual representation of the images reconstructed from examples of defective data across various categories. Notably, all the data depicted in the figure contain defects, with varying types of defects evident. The first row showcases the original images with defects, while the second row displays the images that have undergone reconstruction using the autoencoder. In these original images, patterns are discernible, even within the background, featuring prominent characteristics. However, the defective regions exhibit even more pronounced features. Therefore, the process of removing noise from the background serves to effectively highlight the defective regions. As a result, the overall reconstructed images appear somewhat blurred, with a notable reduction in background noise, except for the grid category. While the defective regions may also appear somewhat blurred, the elimination of background patterns simplifies the task of extracting defects. It is noteworthy that the grid category maintains a consistent pattern that closely resembles the original, except in areas with defects, where observable differences emerge.

The utilization of normal reconstructed templates was necessitated by the network's inherent limitations in terms of its reconstruction capabilities. As a consequence, when creating various images from the original images, we observe significant disparities. In practical terms, even normal data cannot be perfectly reconstructed to replicate the original images. Therefore, this approach proves invaluable in selectively retaining only the defective regions by optimizing the reconstructed results for the original images. The normal reconstructed template represents the outcomes of restor-
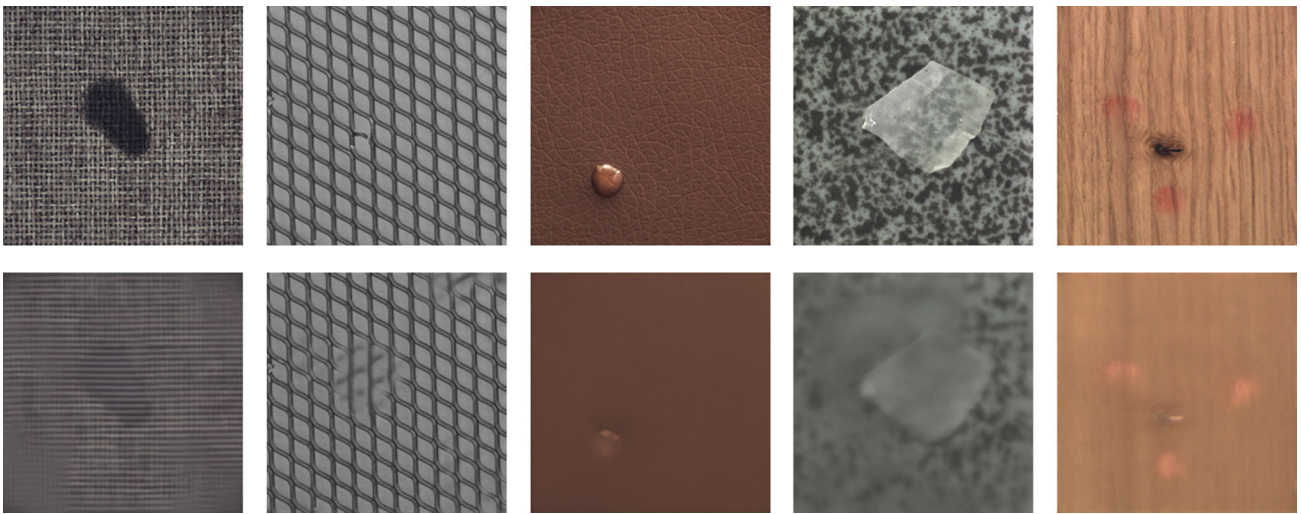
Fig. 6 Examples of reconstructed images with defects

Fig. 7 Examples of normal reconstructed templates

ing normal data and encompasses a diverse array of forms and patterns. These templates are employed to generate distinct images, subsequently harnessed for detecting the regions within the images that exhibit defects. Given that the network was exclusively trained on normal images, the processing of images containing defects results in reconstructed images that display slight deviations in their distributions. Consequently, the approach involves calculating differences between the normal reconstructed templates and the regions within the images that manifest defects, ultimately facilitating the extraction of these defects.

This method excels in precisely detecting defects by effectively distinguishing between normal and defective regions. Through the differentiation of reconstructed images, the non-defective areas are eliminated, leaving behind only the defective regions. This enhances the distinctiveness of the defects, resulting in more accurate recognition. This approach leverages the inherent constraints of the autoencoder's reconstruction capabilities. It utilizes the normal reconstructed template to enhance defect recognition performance. By eliminating the normal areas and emphasizing the defective regions, we can achieve more accurate defect recognition. The images reconstructed from normal data encompass a variety of normal restoration templates. Consequently, it is crucial to select the most suitable template

for each data category. Figure 7 provides a visual representation of the chosen templates for each category, based on the experimental results. These templates can be applied to generate different images, ensuring consistent performance improvement across all data. It is important to note that, when generating these different images, we excluded the 10-pixel edge from the evaluation. This exclusion is necessary because the edge exhibits a different distribution compared to the original data, mainly due to padding, and this difference could lead to misclassification.

As outlined in Sect. 3, it is crucial to determine the most suitable template by combining the Fourier mask, denoted as τ, with binary-level thresholding represented by th. To assess performance, we employed the area under the curve (AUC) as the evaluation metric. AUC is a widely used metric for evaluating the performance of classification models, ranging from 0 to 1, where a value closer to 1 indicates superior performance. Given that each category possesses distinct characteristics, it necessitates varying parameter values, leading us to explore different combinations of these parameters. Consequently, we used the AUC to identify the optimal parameter combination for each category, selecting the combination that yielded the highest AUC value. The most suitable parameter values for τ and th are determined by
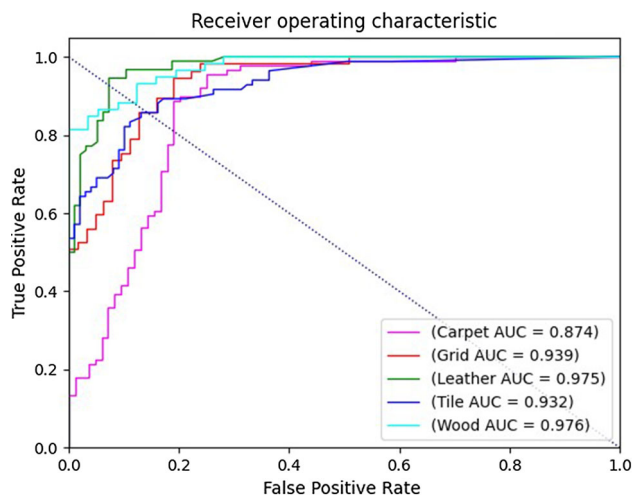
Fig. 8 ROC and AUC of the proposed method for textures in MVTec AD dataset [22]

Table 2 Performance analysis

Carpet

	7	8	9	10	11	12
37	0.736	0.727	0.720	0.715	0.712	0.713
38	0.732	0.731	0.730	0.726	0.725	0.731
39	0.763	0.758	0.758	0.762	0.766	0.771
40	0.815	0.820	0.822	0.821	0.825	0.822
41	0.869	0.871	**0.874**	0.864	0.845	0.836
42	0.845	0.827	0.812	0.783	0.759	0.743

Grid

	18	19	20	21	22	23
42	0.907	0.815	0.920	0.931	0.925	0.913
43	0.918	0.918	0.921	0.919	0.915	0.902
44	0.919	0.934	**0.939**	0.933	0.922	0.903
45	0.928	0.934	0.937	0.929	0.919	0.889
46	0.923	0.931	0.929	0.929	0.915	0.890
47	0.931	0.924	0.926	0.921	0.906	0.867

Leather

	4	5	6	7	8	9
1	0.910	0.932	0.924	0.901	0.892	0.871
2	0.955	0.964	**0.975**	0.900	0.824	0.788
3	0.968	0.951	0.888	0.847	0.825	0.793
4	0.953	0.914	0.887	0.845	0.785	0.760
5	0.946	0.909	0.859	0.800	0.752	0.722
6	0.948	0.890	0.846	0.787	0.745	0.716

Tile

	1	2	3	4	5	6
37	0.685	0.898	0.818	0.766	0.696	0.643
38	0.695	0.915	0.820	0.748	0.685	0.637
39	0.727	0.896	0.786	0.743	0.685	0.631
40	0.736	**0.932**	0.786	0.741	0.679	0.625
41	0.735	0.914	0.794	0.735	0.667	0.625
42	0.733	0.923	0.791	0.723	0.655	0.625

Wood

	7	8	9	10	11	12
1	0.875	0.897	0.929	0.953	**0.976**	0.962
2	0.886	0.910	0.939	0.932	0.938	0.935
3	0.914	0.918	0.944	0.940	0.939	0.946
4	0.906	0.930	0.917	0.918	0.918	0.917
5	0.901	0.925	0.911	0.904	0.912	0.910
6	0.900	0.911	0.898	0.906	0.915	0.912

Bold text indicates the highest performance
row: Fourier mask length τ, column: threshold value th

systematically evaluating their performance across all possible combinations, ultimately leading to the final selection.

In this paper, an evaluation is conducted using the test data mentioned in Table 1 for each category. Additionally, an experiment comparing all candidates of the Normal Reconstructed Template with combinations of (τ, th) values is performed. The combination of (τ, th) has been evaluated for all integer values, with comparisons made for every possible combination in the ranges τ: [1, 50] and th: [1, 25]. The results of this analysis are presented in Table 2, with the parameter combinations that achieved the highest AUC for each category outlined as follows (τ, th): Carpet: (41, 9), Grid: (44, 20), Leather: (2, 6), Tile: (40, 2), Wood: (1, 11).

Fig. 8 represents the ROC curves for the combinations of parameters that achieved the highest performance for each category presented in Table 2. The overall average AUC was 93.1%, signifying that our proposed simple method delivers performance on par with state-of-the-art approaches. These results underscore the effectiveness of our method in defect recognition despite its simplicity. Notably, the Carpet category exhibited a relatively lower AUC compared to the other categories. This discrepancy can be attributed to the smaller difference between the defect regions and the background in the Carpet category when compared to the other categories.

When the distinction between the defect regions and the background is minimal, removing noise from the background may inadvertently lead to the elimination of defect regions, thereby making accurate recognition more challenging. In the case of carpets, the performance of the method is observed to be comparatively lower when compared to other categories. This suggests that the underlying cause lies in the distinct characteristics of carpet data. This analysis is attributed to the presence of various high-frequency regions within the background noise removal process, which possesses frequencies

similar to actual defects and thus remain unremoved. Our study demonstrates that a straightforward method can yield high-performance results. Furthermore, the adaptability of
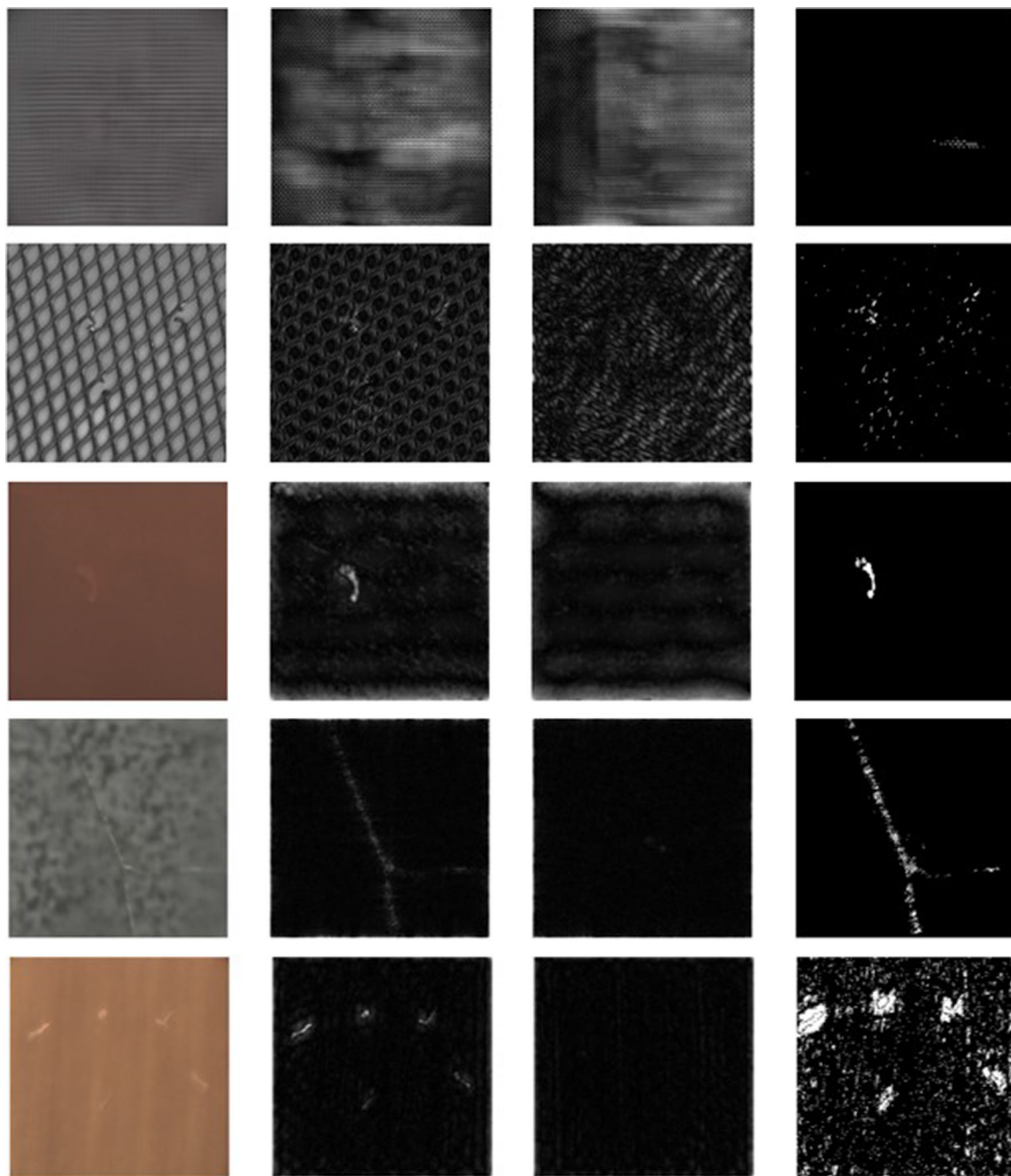
Fig. 9 Workflow for defect detection (1st column: reconstruction, 2nd–3rd columns: Fourier transform results, 4th column: binary-level thresholding)

our method to accommodate variations in optimal parameter combinations for different categories underscores its flexibility and simplicity, rendering it applicable to a wide range of defect recognition challenges. Figure 9 illustrates a partial process for generating the final decision map for each category. The first column displays the reconstructed image, while the second and third columns depict the frequency domain images of the input image and the normal reconstructed template, respectively. The fourth column showcases

the result of binary-level thresholding applied to different Fourier images. Defective regions are represented by white areas, demonstrating the preservation of actual defective areas.

Table 3 presents the results of evaluating various anomaly recognition algorithms with SOTA across different categories, with performance assessed using the AUC value. In the comparative analysis, the algorithms chosen for comparison with the proposed method are evaluated by citing metrics

Table 3 Comparisons of performance with related works

Methods	Patch svdd [17]	Uniformed student [18]	KDAD [21]	DAGAN [19]	PEDENet [20]	Ours
Carpet	0.48	0.695	0.879	0.903	0.922	**0.866**
Grid	0.83	0.819	0.952	0.867	0.959	**0.957**
Leather	0.69	0.819	0.945	0.944	**0.976**	0.975
Tile	0.70	0.912	0.946	**0.961**	0.926	0.932
Wood	0.94	0.725	0.911	**0.979**	0.900	0.976
Average	0.728	0.794	0.927	0.931	0.936	**0.939**

Bold text indicates the highest performance

Table 4 Ablation study results

Normal reconstructed template	X	O	O
Fourier transform	O	X	O
Carpet	0.791	0.659	0.874
Grid	0.885	0.508	0.939
Leather	0.344	0.634	0.975
Tile	0.819	0.505	0.932
Wood	0.923	0.867	0.976
Average	0.752	0.634	0.939

Table 5 Comparative performance of ideal filter and Butterworth filter

Filters	Ideal	Butterworth (ours)
Carpet	0.828	0.874
Grid	0.920	0.939
Leather	0.969	0.975
Tile	0.881	0.932
Wood	0.963	0.976
Average	0.912	0.939

according to the settings specified in each study, due to the use of identical datasets in the experiments. A value of 1 indicates the presence of defects, while 0 indicates their absence.

The studies utilized for performance comparison [17], [18], [19], [20] and [21] all focus on detecting defects using the MVTec AD [22]. These methods evaluate performance by combining complex deep learning networks with conventional approaches. When examining each category individually, our method demonstrates lower performance compared to the state-of-the-art methods DAGAN [19] and PEDENet [22]. However, while we may not achieve the highest performance in each category, our overall performance is 8% and 3% higher, respectively, when compared to these methods.

Analyzing the results for each category, as per the analysis in Table 4, the Carpet category exhibits relatively lower performance. In the Grid, our performance is 7.2% higher than DAGAN [19], but it is more than 2% lower than PEDENet [20] and KDAD [21]. However, in the "Leather", our performance is slightly lower than PEDENet [20] but still quite similar, with less than a 1% difference. Additionally, in the "Tile" and "Wood", our performance is higher than PEDENet [20] but slightly lower than DAGAN [19].

However, the core objective of this paper is to introduce an efficient inference method using a simple network architecture. Furthermore, our proposed method preserves defects in the frequency domain, rather than using deep learning for feature extraction, making it superior in this regard. Consequently, an AUC of 93.9% demonstrates excellent performance. In real-time inference scenarios, our method is more practical compared to other studies, as it does not require a lengthy inference process involving complex deep learning networks. Additionally, if the Normal Reconstructed Template is predefined, the inference time can be reduced to less than $20\mu s$ per image, enabling rapid inference speeds suitable for real-time applications. The method presented in this study comprises two primary components: the utilization of a denoising autoencoder for generating a normal reconstructed template and its integration with the Fourier transform to segregate and eliminate defective features. The performance evaluation was conducted with a focus on the combined use of the normal reconstructed template and Fourier transform. It is worth noting that solely employing the Fourier transform makes it exceedingly challenging to distinguish defective features in the frequency domain, resulting in a slightly higher overall AUC of 75.2% compared to the obtained results. Notably, the Leather category recorded a significantly lower value of 0.344 when compared to other categories, possibly due to the data's unique characteristics, making defect recognition through frequency bands less effective in this category. When using only the normal reconstructed template in conjunction with the Fourier transform, the average AUC was low at 63.4% across all categories except Leather. This suggests that relying solely on the normal reconstructed template to preserve defective regions is challenging for other categories, while the reconstructed results alone can effectively preserve defective regions. In contrast, the proposed method, which combines both the normal reconstructed template and Fourier transform, achieved the highest performance with an AUC of 93.9%. Therefore, the methodology introduced in this study

Table 6 Additional performance evaluations and decision threshold for each category in the proposed method

Category	Precision	Recall	F1-Score	Macro-F1	Decision thresholds
Carpet	0.940	0.750	0.834	0.853	0.094
	0.802	0.955	0.872		
Grid	0.944	0.810	0.872	0.875	0.974
	0.818	0.947	0.878		
Leather	0.947	0.927	0.937	0.936	0.110
	0.926	0.946	0.935		
Tile	0.878	0.869	0.873	0.863	0.029
	0.847	0.857	0.852		
Wood	0.838	1.000	0.912	0.906	0.117
	1.000	0.817	0.899		

demonstrates superior performance compared to evaluations using existing methods in isolation. In particular, when compared to the common approach of using only the Fourier transform, this method exhibited a remarkable 18.7% performance improvement, achieving the highest performance across all categories.

The proposed approach incorporates a high-pass filter in this study, specifically introducing a method based on the Butterworth filter. The ideal filter selectively assigns a value of one to the high-frequency region through a defined cutoff, while the rest is set to zero. However, owing to the potential of filters to frequently introduce a ringing artifact, the Butterworth filter is favored over the ideal filter. A notable distinction between the Butter-worth and ideal filters is the smoother boundary transitions provided by the former. In this study, post-filtering is employed, resulting in the generation of a binary image through a post-processing step that utilizes a threshold. The primary objective of this post-processing step is to isolate and preserve only the defects. Consequently, both the ideal and Butterworth filters prove resilient to the ringing artifact. Table 5 presents the performance evaluation results for various filters. The preference for the Butterworth filter over the Ideal filter, particularly in the context of generating binary images, underscores its superior performance in terms of defect preservation.

Table 6 presents the computed Decision Thresholds for each category according to the proposed method, and based on these thresholds, it details the Precision, Recall, and F1-Score for normal and defect categories, as well as the Macro-F1 values. It is observed that, with the exception of the Grid category, most categories are close to 0. The Grid category, due to its high-frequency patterned background, requires a threshold near 1 to detect defects, resulting in an F1-Score of 87.5%. For other categories, typically characterized by low-frequency background noise, the thresholds are calculated around 0.1, with the Tile category notably at a very low 0.029. Unlike the AUC, these figures represent actual binary classification results, with Leather achieving the highest performance at 93.6%. Although the values are generally lower than those of AUC, the metrics suggest that the proposed method holds substantial value, especially when considering the characteristics of the metrics and the perspective of inference time.

5 Conclusion

This study introduces an image analysis methodology that combines a simple denoising autoencoder with Fourier transformation to explore texture defect recognition. The proposed approach utilizes an autoencoder for noise removal and a Fourier transform process to isolate defective regions, with the presentation of the most suitable mask radius and threshold values for each dataset.

To enhance the performance of the method in future research, it is crucial to employ deeper learning networks for improved noise removal. The development and application of advanced noise removal techniques through deep learning have the potential to enhance defect detection capabilities, and they are essential for improving applicability in the real world. However, while this method demonstrated high performance in most textures, it achieved relatively lower performance in certain textures. This limitation is attributed to the fact that the method detects defects in textures based solely on frequency components rather than features. However, it offers advantages such as high efficiency and real-time usability. To enhance the performance of the method in future research, it is crucial to employ deeper learning networks for improved noise removal. The development and application of advanced noise removal techniques through deep learning have the potential to enhance defect recognition capabilities, and they are essential for improving applicability in the real world. Currently, the comparison has been conducted solely on manufacturing data. However, there is significant potential for broader application of this method in various domains. To contribute more extensively to diverse fields, research is planned on methodologies for integrative analysis of data across multiple sectors.

Declarations

Conflict of interest The authors declare no conflict of interest.

References

1. Tsai, D.M., Huang, C.K.: Defect detection in electronic surfaces using template-based fourier image reconstruction. IEEE Trans. Compon. Packag. Manuf. Technol. **9**(1), 163–172 (2018). https://doi.org/10.1109/TCPMT.2018.2873744

2. Zavrtanik, V., Kristan, M., Skočaj, D.: Draem: A discriminatively trained reconstruction embedding for surface anomaly detection. In: IEEE/CVF International Conference on Computer Vision, pp. 8330–8339. (2021)

3. Schlegl, T., Seeböck, P., Waldstein, S.M., Schmidt-Erfurth, U., Langs, G.: Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In: International Conference on Information Processing in Medical Imaging, pp. 146–157. (2017)

4. Schlegl, T., Seeböck, P., Waldstein, S.M., Langs, G., Schmidt-Erfurth, U.: f-anogan: fast unsupervised anomaly detection with generative adversarial networks. Med. Image Anal. **54**, 30–44 (2017). https://doi.org/10.1016/j.media.2019.01.010

5. Akçay, S., Atapour-Abarghouei, A., Breckon, T.P.: Ganomaly: Semi-supervised anomaly detection via adversarial training. In: Asian Conference on Computer Vision, pp. 622–637. (2018)

6. Akçay, S., Atapour-Abarghouei, A., Breckon, T.P.: Skip-ganomaly: Skip connected and adversarially trained encoder-decoder anomaly detection. In: IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 39–42. (2018)

7. Gong, D., Liu, L., Le, V., Saha, B., Mansour, M.R., Venkatesh, S., Hengel, A.V.D.: Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection. In: IEEE/CVF International Conference on Computer Vision, pp. 1705–1714. (2019)

8. Perera, P., Nallapati, R., Xiang, B.: Ocgan: One-class novelty detection using gans with constrained latent representations. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2898–2906. (2019)

9. Zhao, Y., Chen, Z., Gao, X., Song, W., Xiong, Q., Hu, J., Zhang, Z.: Plant disease detection using generated leaves based on doublegan. IEEE/ACM Trans. Comput. Biol. Bioinf. **19**(3), 1817–1826 (2021). https://doi.org/10.1109/TCBB.2021.3056683

10. Liang, Y., Zhang, J., Zhao, S., Wu, R., Liu, Y., Pan, S.: Omni-frequency channel-selection representations for unsupervised anomaly detection. arXiv preprint arXiv:2203.00259 (2023)

11. Jang, J., Hwang, E., Park, S.H.: N-pad: Neighboring pixel-based industrial anomaly detection. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4364–4373. (2023)

12. Si, J., Kim, S.: Difference image alignment technique of reconstruction method for detecting defects in thermal image of solar cells. J. Korean Inst. Inf. Technol. **21**(5), 11–19 (2023). https://doi.org/10.14801/jkiit.2023.21.5.11

13. Tsai, C.C., Wu, T.H., Lai, S.H.: Multi-scale patch-based representation learning for image anomaly detection and segmentation. In: IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 3992–4000. (2022)

14. Liu, T., Li, B., Zhao, Z., Du, X., Jiang, B., Geng, L.: Reconstruction from edge image combined with color and gradient difference for industrial surface anomaly detection. arXiv preprint arXiv:2210.14485 (2022)

15. Shi, Y., Yang, J., Qi, Z.: Unsupervised anomaly segmentation via deep feature reconstruction. Neurocomputing **424**, 9–22 (2021). https://doi.org/10.1016/j.neucom.2020.11.018

16. Hou, J., Zhang, Y., Zhong, Q., Xie, D., Pu, S., Zhou, H.: Divide-and-assemble: Learning block-wise memory for unsupervised anomaly detection. In: IEEE/CVF International Conference on Computer Vision, pp. 8791–8800. (2021)

17. Yi, J., Yoon, S.: Patch svdd: Patch-level svdd for anomaly detection and segmentation. In: Asian Conference on Computer Vision, pp. 1–16. (2020)

18. Bergmann, P., Fauser, M., Sattlegger, D., Steger, C.: Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4183–4192. (2020)

19. Tang, T.W., Kuo, W.H., Lan, J.H., Ding, C.F., Hsu, H., Young, H.T.: Anomaly detection neural network with dual auto-encoders gan and its industrial inspection applications. Sensors **20**(12), 3336–3346 (2020). https://doi.org/10.3390/s20123336

20. Zhang, K., Wang, B., Kuo, C.C.J.: Pedenet: Image anomaly localization via patch embedding and density estimation. Pattern Recognit. Lett. **153**, 144–150 (2022). https://doi.org/10.1016/j.patrec.2021.11.030

21. Salehi, M., Sadjadi, N., Baselizadeh, S., Rohban, M.H., Rabiee, H.R.: Multiresolution knowledge distillation for anomaly detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 14902–14912. (2021)

22. Bergmann, P., Fauser, M., Sattlegger, D., Steger, C.: Mvtec ad–a comprehensive real-world dataset for unsupervised anomaly detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9592–9600. (2019)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.