information

MDPI

Article

WTSM-SiameseNet: A Wood-Texture-Similarity-Matching Method Based on Siamese Networks

Yizhuo Zhang *, Guanlei Wu, Shen Shi and Huiling Yu

Aliyun School of Big Data, School of Software, School of Computer Science and Artificial Intelligence, Changzhou University, Changzhou 213161, China; s22150812052@smail.cczu.edu.cn (G.W.); shishen@cczu.edu.cn (S.S.); yhl2016@cczu.edu.cn (H.Y.)

* Correspondence: yzzhang@cczu.edu.cn

Abstract: In tasks such as wood defect repair and the production of high-end wooden furniture, ensuring the consistency of the texture in repaired or jointed areas is crucial. This paper proposes the WTSM-SiameseNet model for wood-texture-similarity matching and introduces several improvements to address the issues present in traditional methods. First, to address the issue that fixed receptive fields cannot adapt to textures of different sizes, a multi-receptive field fusion feature extraction network was designed. This allows the model to autonomously select the optimal receptive field, enhancing its flexibility and accuracy when handling wood textures at different scales. Secondly, the interdependencies between layers in traditional serial attention mechanisms limit performance. To address this, a concurrent attention mechanism was designed, which reduces interlayer interference by using a dual-stream parallel structure that enhances the ability to capture features. Furthermore, to overcome the issues of existing feature fusion methods that disrupt spatial structure and lack interpretability, this study proposes a feature fusion method based on feature correlation. This approach not only preserves the spatial structure of texture features but also improves the interpretability and stability of the fused features and the model. Finally, by introducing depthwise separable convolutions, the issue of a large number of model parameters is addressed, significantly improving training efficiency while maintaining model performance. Experiments were conducted using a wood texture similarity dataset consisting of 7588 image pairs. The results show that WTSM-SiameseNet achieved an accuracy of 96.67% on the test set, representing a 12.91% improvement in accuracy and a 14.21% improvement in precision compared to the pre-improved SiameseNet. Compared to CS-SiameseNet, accuracy increased by 2.86%, and precision improved by 6.58%.

Keywords: Siamese network; wood texture similarity; concurrent attention; multi-receptive field

Citation: Zhang, Y.; Wu, G.; Shi, S.; Yu, H. WTSM-SiameseNet: A Wood-Texture-Similarity-Matching Method Based on Siamese Networks. *Information* **2024**, *15*, 808. https://doi.org/10.3390/info15120808

Academic Editors: Jie Liu, Shanmei Liu, Fang Yang and Khalid Sayood

Received: 4 November 2024
Revised: 4 December 2024
Accepted: 11 December 2024
Published: 16 December 2024

1. Introduction

Texture is an important visual feature of wood, reflecting its local structure and surface details. In high-end applications such as the production of premium wooden furniture, the restoration of ancient architecture, and the defect-free joining of wood, material selection is based not only on strength or durability but also on the uniformity and aesthetics of appearance. As consumer demands for visual consistency increase, the market's need for highly consistent wood-texture matching is growing. Traditional manual selection methods are not only time-consuming and labor-intensive but also struggle to ensure accuracy, making them unsuitable for large-scale production. Therefore, the development of automated wood-texture-similarity matching technology has become an inevitable trend in industrial development.

Wood texture similarity matching refers to the process of calculating and analyzing the similarity of surface textures in order to assess the degree of similarity in texture between different wood samples. This is a subtask of content-based texture matching. Unlike other subtasks such as face recognition [1], fingerprint matching [2], medical diagnosis [3], and

object detection [4], wood-texture-similarity matching presents greater challenges due to the complexity, randomness, and diversity of wood textures. First, wood textures not only exhibit significant differences between different wood species, but the texture of the same species can also vary considerably due to factors such as growth environment, cutting method, and annual rings. This high degree of texture diversity [5] makes it difficult for traditional texture-matching methods to effectively capture and represent the similarities of wood surfaces. For example, the textures of different woods may be similar in overall shape, but local details (such as small pores or color differences) may vary significantly, increasing the difficulty of matching algorithms to handle these details. Secondly, the randomness of wood textures arises from the complex structures naturally formed during the growth process, and this randomness is reflected in aspects such as the direction, density, and morphology of the textures. Therefore, wood-texture-similarity matching requires the model not only to handle textures at different scales and angles but also to possess sufficient robustness to address texture variations caused by factors such as lighting, shooting angles, or deformations during processing. In contrast, other texture-matching tasks, such as face recognition and fingerprint matching, often rely on relatively regular features (such as facial characteristics or unique fingerprint patterns), which are either not prominent or exhibit considerable variability in wood. Additionally, although medical image diagnostics involve complex texture analysis, the targets in medical images usually have clearer boundaries and contrasts [6], thus differing from the processing of wood textures. Although object-detection tasks also involve texture analysis, their primary goal is the localization and classification of target objects rather than fine-grained texture matching [7]. Therefore, the challenges faced during texture analysis are different. In conclusion, the difficulty of wood-texture-similarity matching arises not only from the high complexity of wood textures but also from their randomness and diversity. These factors make wood-texture matching more challenging than other texture-matching tasks and require the development and optimization of algorithms specifically designed to account for the unique characteristics of wood. Currently, the mainstream methods for texture-similarity matching can be categorized into the following three types:

The first method is based on pixel differences for texture similarity comparison, such as calculating the similarity between images using the Gray-Level Co-occurrence Matrix (GLCM) [8] and the Structural Similarity Index Measure (SSIM). However, it is important to note that the GLCM is not directly used to compute texture similarity; rather, it describes the texture structure by extracting statistical features of the image, such as contrast, homogeneity, and energy, thereby providing a basis for further similarity calculation. Srivastava D [9] et al. proposed that by using the GLCM to statistically analyze the joint distribution of grayscale levels at specific directions and distances, one can determine texture patterns, such as horizontal, vertical, and diagonal stripes, and assess whether the textures are similar. Furthermore, in fields such as camouflage clothing design [10] and image forgery detection [11], the GLCM is also used to assess texture similarity. Calculating features such as contrast, homogeneity, and entropy can reveal areas of texture inconsistency, which helps to evaluate the rationality of camouflage designs and the likelihood of image manipulation. However, this method has notable limitations when dealing with wood textures. It is not sensitive enough to details and color variations in the texture, making it difficult to accurately capture the complex and intricate texture features of wood surfaces. On the other hand, the Structural Similarity Index Measure [12] (SSIM) is mainly used to compare texture similarity based on local differences in brightness, contrast, and structure. Although SSIM can compare the local features of images, in wood-texture matching, it relies too heavily on local information, making it difficult to reflect the global structure of the texture. Especially for wood with significant differences in texture details, the sensitivity of SSIM is too high, often leading to inaccurate matching results. Therefore, pixel-level comparison methods like the Gray-Level Co-occurrence Matrix and SSIM, while capable of capturing some texture features, exhibit significant shortcomings in addressing the complexity and

diversity of wood textures, making them inadequate for precise wood texture-similarity matching tasks.

The second method is the Bag of Visual Words (BoVWs) approach based on local feature extraction and clustering [13]. This method extracts representative key points from images using feature detection algorithms such as SIFT [14], SURF [15], and ORB [16] and compiles these local features into a "visual vocabulary". Subsequently, the similarity of images is computed based on the feature distribution in the visual vocabulary. However, the limitation of the Bag of Visual Words method in wood texture-similarity matching lies in its neglect of the spatial relationships among texture features. Wood textures possess complex spatial structures, and variations in relative positioning are crucial for assessing texture similarity. The Bag of Visual Words method focuses solely on the frequency distribution of local features without considering their arrangement within the image. Therefore, while this method is effective in certain scenarios, it performs poorly in wood texture-similarity matching because it cannot reflect the global layout of the texture.

The third method is based on deep learning, particularly the Siamese network. This approach effectively captures multi-level features of wood texture, from local details to global structures, using deep convolutional neural networks (CNNs), allowing for the automatic learning and extraction of subtle changes and complex patterns in wood surfaces. Compared to traditional Bag of Visual Words methods, deep learning can not only handle local features but also preserve the spatial positional information of textures, thereby better reflecting the global characteristics of wood textures. This gives the Siamese network a strong advantage in wood-texture-similarity matching. However, the Siamese network also faces several challenges. First, the fixed receptive field of the network cannot adapt to texture features of different scales, limiting the model's ability to capture the diversity of complex wood textures. Secondly, the continuous use of multiple attention mechanisms may lead to interference among these mechanisms, restricting each mechanism's ability to capture features. Thirdly, overly simplistic feature fusion methods may disrupt the spatial structure of the original data, weakening the global understanding of textures. Finally, directly using fully connected networks for similarity calculation, while capable of capturing global information, results in a dramatic increase in model parameters, thereby increasing computational complexity and reducing efficiency.

To address the issues present in the aforementioned Siamese network and enhance the performance of wood-texture-similarity matching, this paper designs the WTSM-SiameseNet (Wood-Texture-Similarity Matching-SiameseNet) network model. The main contributions of this paper are as follows:

1. A feature extraction network with multi-receptive field fusion was designed. By integrating feature extraction networks with different receptive fields, the model can adaptively select the optimal receptive field, addressing the issue of fixed receptive fields not being able to accommodate textures of different sizes. This design allows the model to exhibit better flexibility and accuracy when dealing with wood textures of varying scales;
2. A concurrent attention mechanism was designed. By employing a dual-stream parallel attention mechanism, the interdependence among layers in traditional serial attention mechanisms is reduced, enhancing the overall performance of the attention mechanism. This not only enhances the capability of feature capture but also avoids interference among multiple attention mechanisms;
3. A feature fusion method based on feature correlation was designed. The newly designed feature fusion method retains the spatial structure of the original texture features while enhancing the interpretability of the fused features, optimizing the overall expressive capability of the model, reducing parameter complexity, and improving the stability and accuracy of the model;
4. By optimizing the Siamese network structure with depthwise separable convolutions, the model's parameter count is significantly reduced, thereby improving training

efficiency. This optimization significantly reduces the consumption of computational resources while ensuring the model's performance.

2. Related Work

Research on texture-similarity matching not only has practical value but is also an important research direction in the fields of computer vision and deep learning. Siamese networks can not only capture the global and local features of two input texture images through two shared-weight neural networks but also precisely calculate their similarity by comparing the feature vectors of the two images. Furthermore, Siamese networks are sensitive to spatial structures. In wood texture-similarity matching, the relative positions and arrangement of textures are crucial. By using deep feature representation, Siamese networks can fully consider these spatial relationships, making them more suitable for wood-texture-similarity matching than traditional methods. At the same time, Siamese networks have demonstrated strong capabilities in many other computer vision tasks, such as video object tracking [17], change detection [18], handwriting recognition [19], and self-supervised learning [20]. These applications further demonstrate the broad applicability and flexibility of Siamese networks in various scenarios, providing valuable insights and inspiration for addressing the wood-texture-similarity matching problem in this study.

The accuracy of similarity computation in the Siamese network primarily relies on the feature extraction performance of its texture feature extraction module and the learning ability of the similarity metric function in the texture feature aggregation matching module. To enhance the performance of the texture feature extraction module, researchers have proposed various methods. Figueroa-Mata G [21] used multiple convolution kernels of different sizes (e.g., 11×11, 8×8, and 5×5) in the feature extraction network, which can improve the model's ability to extract texture information at different scales and enhance its generalization capability. However, for wood texture, in addition to the most prominent and regular primary textures on the wood surface, there are also finer-grained secondary textures, which reflect the details of wood fibers, tiny pores, and color variations. Although using larger convolution kernels (such as 11×11, 8×8, and 5×5) can effectively capture primary textures, it also captures excessive irrelevant information, leading to overfitting on secondary textures and unrelated details, thereby reducing the model's ability to generalize to new samples. Moreover, larger convolution kernels require training more parameters compared to smaller ones, resulting in reduced training speed. Hudec L [22] proposed using AlexNet as the feature extraction network for the Siamese network, which mitigates overfitting while enhancing the stability and robustness of texture feature extraction by reducing kernel size and incorporating pooling operations. However, multiple pooling operations may lead to the loss of significant detail texture features, affecting the model's ability to perceive texture details. VGGNet [23,24] increased the model's receptive field by using multiple 3×3 convolution kernels while requiring fewer parameters for training. Building on this, Cao W [25] and others introduced the CBAM attention mechanism in VGGNet to help the network effectively select high-value information, further enhancing the model's feature extraction capability. However, this method still faces challenges regarding the smoothness of the network. Yan R [26] designed CS-SiameseNet by combining an improved VGG16 with a Siamese network, using the Mish function instead of the ReLU function to enhance the network's smoothness, non-linearity, and tolerance. Nevertheless, the fixed receptive field size of the feature maps of CSNet limits its performance in capturing complex texture features, leaving room for improvement. Additionally, Peng Z [27] and others proposed that cascaded self-attention modules can deteriorate local feature details, and using concurrent structures can maximize the preservation of local features and global representations. However, these studies have not been widely applied in texture-similarity matching based on Siamese networks.

Researchers have also explored various methods to enhance the performance of the texture-feature-aggregation matching module. First, the Pearson correlation coefficient [28] and the Spearman correlation coefficient [29] are often used as metrics in the Siamese

network due to their simplicity in computation and ease of interpretation, providing direct and effective standardized results for similarity measurement. However, their assumptions of linear and monotonic relationships limit their ability to reflect the complex relationships between wood textures, and both methods may perform poorly in wood-texture-similarity matching. Additionally, Hayale W [30] attempted to use Euclidean distance as the similarity metric function for the Siamese network to enhance the smoothness of the metric function and assist in model training. However, Euclidean distance assumes equal weights for all features, making it ineffective in distinguishing the importance of different features. Since fully connected networks can capture complex nonlinear relationships from input data. Yu J [31] and others replaced the conventional similarity metric function with a fully connected network, learning the mapping relationship between the high-order features of the data and the similarity score through multiple fully connected layers and directly outputting the similarity score. This approach addresses the issue of traditional Siamese networks being overly influenced by thresholds in texture-similarity classification. However, directly using fully connected networks to compute similarity significantly increases the model's parameter count, leading to slow training; thus, further improvements are necessary.

3. WTSM-SiameseNet

The Siamese network is commonly used to analyze the similarity between data. Its core components include the texture feature extraction module and the texture feature aggregation and matching module, as shown in Figure 1. The texture feature extraction module (Figure 1a) uses two identical feature extraction networks to extract features from the target image, x_1, and the comparison image, x_2. The texture feature aggregation and matching module (Figure 1b) first uses a feature fusion module to merge all texture features into a new feature; then, it uses a similarity calculation module to obtain the similarity between the two images. However, in real scenarios, wood textures have significant differences in size, shape, and detail. The fixed receptive field of the existing texture feature extraction module cannot automatically adjust according to the characteristics of the texture, making it unsuitable for extracting all types of wood texture features. This limitation can lead to feature loss during extraction, resulting in a misjudgment of wood texture similarity. Additionally, the structure of the feature fusion module and the similarity calculation module in the Texture Feature Aggregation and Matching Module is too simplistic, making it difficult to accurately map the relationship between texture features and similarity scores, leading to inaccurate similarity scores.

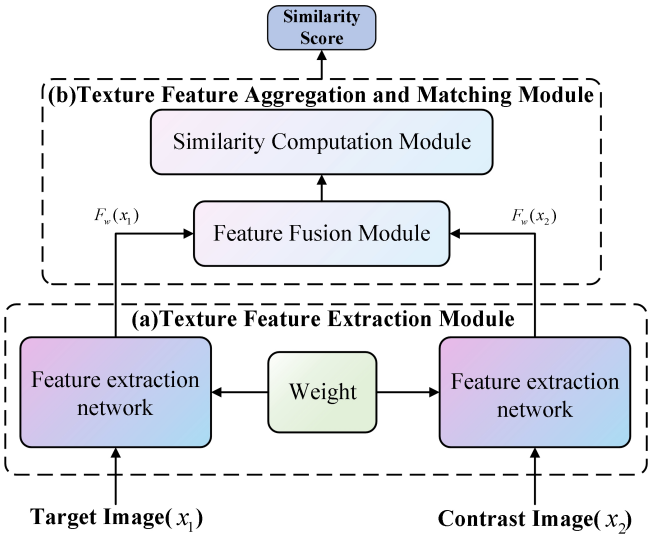

Figure 1. Diagram of the SiameseNet architecture.

To address the above issues, a WTSM-SiameseNet (Wood-Texture-Similarity Matching-Siamese Network) suitable for wood-texture-similarity matching was designed, as shown in Figure 2. In the texture feature extraction module, a concurrent attention mechanism was introduced, and a feature extraction network with a multi-scale receptive field, MRF-Resnet (Multi-scale Receptive Field-Resnet), was designed. This improved the receptive field dimension and edge attention during feature extraction, enhancing the model's ability to evaluate texture similarity. In the Texture Feature Aggregation and Matching Module, the Feature Fusion Module was improved based on feature correlation, and the Similarity Computation Module was enhanced using deep convolution and point convolution, thereby improving the accuracy of similarity computation.

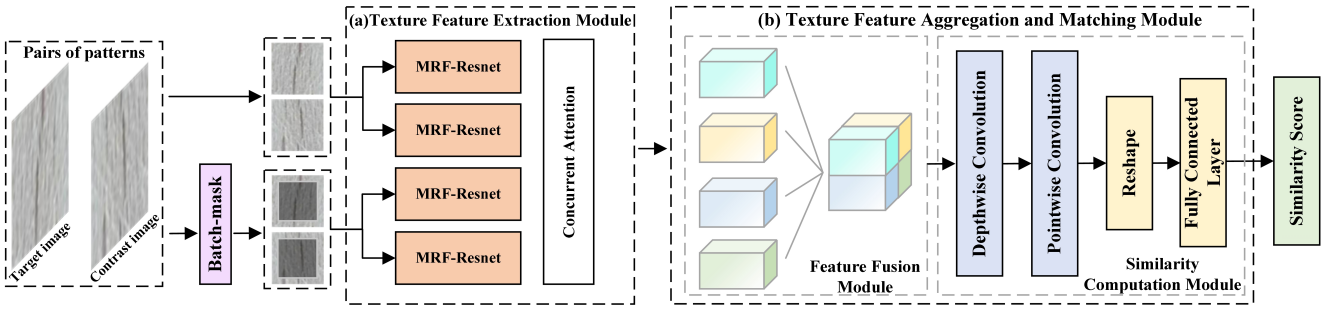

Figure 2. Diagram of the WTSM-SiameseNet architecture.

3.1. Texture Feature Extraction Module

Wood texture includes large-scale main textures and small-scale detailed textures. To address this characteristic, the MRF-ResNet model designed in this study effectively extracts texture features at different scales by using multiple receptive fields, enabling the model to consider both coarse and detailed textures in wood, thus enhancing the understanding and extraction of complex textures. Furthermore, by considering the high demands of wood-texture-similarity matching for the spatial continuity and position dependence of the textures, as well as the regularity and repetitiveness of texture distribution, this paper proposes a concurrent attention mechanism that combines edge attention and CBAM attention. By using the edge attention mechanism, the model can better assess the continuity of board edge textures, and simultaneously, the CBAM attention mechanism enhances the model's sensitivity to the importance of different textures. These designs not only overcome the shortcomings of traditional methods in handling complex textures but also significantly improve the accuracy and robustness of wood-texture-similarity matching.

3.1.1. MRF-Resnet

We designed MRF-Resnet with Resnet as the backbone. Its core components include the Base Residual Block, Bottleneck Residual Block, and multi-receptive field fusion, as shown in Figure 3.

The Base Residual Block (Figure 3a) consists of three convolutional layers. The first convolutional layer reduces the number of channels in the input feature map, thereby reducing computation and parameter count. The second convolutional layer is responsible for feature extraction and transformation, capturing local texture features in the input feature map and converting them into higher-level feature representations. The third convolutional layer restores the feature dimensions, ensuring that the output feature map has the same dimensions as the input feature map.

The Bottleneck Residual Block (Figure 3b) is similar in structure to the Base Residual Block but sets the stride of the second convolutional layer to 2. This enables the Bottleneck Residual Block to perform downsampling while simultaneously increasing the receptive field.

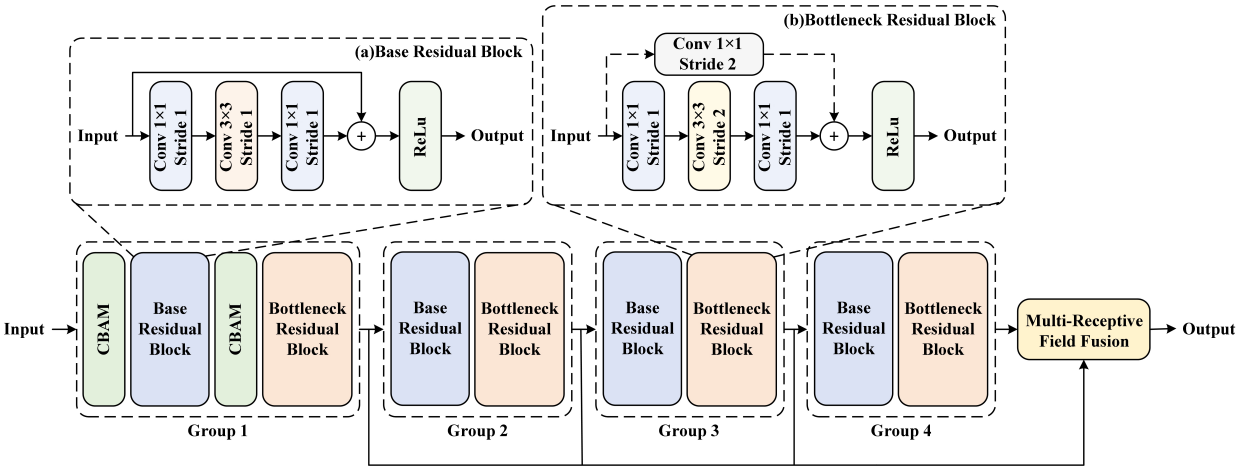

Figure 3. Diagram of the MRF-Resnet architecture.

The receptive field refers to the model's visual field on the input image during feature extraction. Small textures in wood require a smaller receptive field for coverage, whereas larger textures necessitate a larger receptive field. Therefore, a multi-scale receptive field fusion method has been designed to enhance the receptive field dimension in features by integrating features from different receptive fields extracted by the network. The principle of multi-scale receptive field fusion is illustrated in Figure 4 as follows:

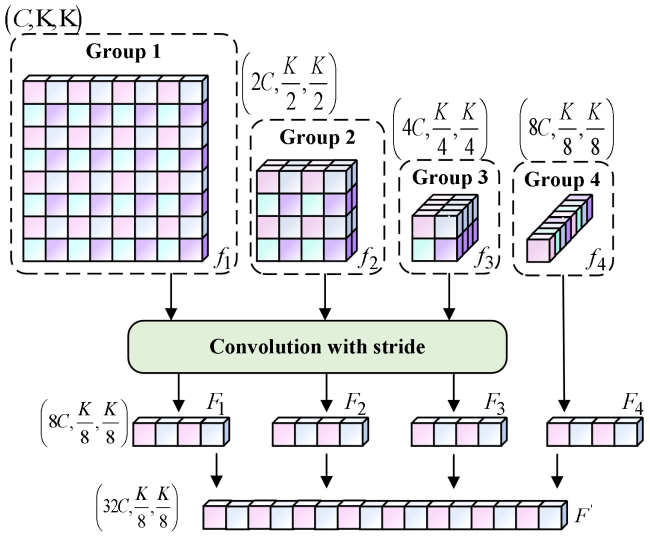

Figure 4. Multi-scale receptive field fusion.

Because the Bottleneck Residual Block enlarges the receptive field of the feature map while changing its dimensions, integrating features from different receptive fields requires the dimension transformation of the features beforehand. The initial size of the feature map is (C, K, K), where C denotes the number of channels in the feature map, and K represents the dimensions (length and width) of the feature map. After each pass through the Bottleneck Residual Block, the number of channels in the feature map doubles from its original amount, and the size of the feature map becomes $\frac{1}{2}$ times the original. After passing through the fourth submodule, the size of the feature map becomes $\left(8C, \frac{K}{8}, \frac{K}{8}\right)$.

First, we preserve the feature maps after processing each Bottleneck Residual Block. Next, we use convolutional operations to transform the shape of all feature maps into $\left(8C, \frac{K}{8}, \frac{K}{8}\right)$, as shown in the size transformation process in Equation (1):

$$F_{i,j} = \sum_{m=0}^{F=1} \sum_{n=0}^{F=1} f_{(i*S+m),(j*S+n)} \cdot K_{m,n} \quad (1)$$

where $F_{i,j}$ represents the pixel value at position (i, j) in the output feature map; $f_{(i*S+m),(j*S+n)}$ represents the pixel values at positions $(i * S + m)$ and $(j * S + n)$ in the input feature map; $K_{m,n}$ represents the convolutional kernel weights.

Finally, we concatenate all reshaped feature maps along the channel dimension to obtain a new feature of shape $\left(32C, \frac{K}{8}, \frac{K}{8}\right)$, which has the same receptive field (spatially) but different receptive fields across channels. The formula for concatenating the new feature is shown in Equation (2):

$$F' = concat(F_1, F_2, F_3, F_4, dim = 1) \tag{2}$$

where F' represents the new feature obtained after feature extraction by MRF-Resnet; F_1, F_2, F_3, and F_4, respectively, represent the feature maps after dimension transformation by each of the four submodules; $concat(dim = 1)$ represents the concatenation along the channel dimension.

3.1.2. Concurrent Attention

In traditional feature extraction networks, all features have equal weights. However, for wood-texture-similarity matching tasks, the focus of feature extraction should be on texture features rather than background features. To enhance the focus on texture features in MRF-Resnet, the CBAM attention mechanism is introduced to enhance attention to textures during global feature extraction. Furthermore, edge textures in wood play a crucial role in subsequent tasks like joining, as they directly affect the consistency of textures at the joints. Therefore, during texture feature extraction, attention needs to be focused on edge textures. However, the serial structure of attention mechanisms may interfere with each other, leading to diminished effectiveness. To address this issue, a concurrent attention mechanism is designed to more effectively extract texture and edge features, as shown in Figure 5.

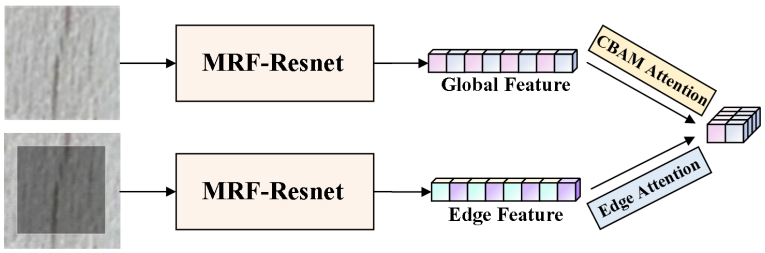

Figure 5. Concurrent attention.

In the concurrent attention mechanism, MRF-ResNet is used to extract features from both complete wood texture patterns and wood patterns containing only edge textures. Obtaining global features and edge features simultaneously ensures that these features are structurally consistent. Additionally, since MRF-ResNet only includes the CBAM attention module, it can preserve complete global texture features during global feature extraction. In contrast, during edge feature extraction, only edge texture patterns are used, ensuring that the extracted features contain only edge texture information without interference from other region features. By integrating global features and edge features, the weight of edge features can be enhanced while preserving global features as much as possible.

CBAM (Convolutional Attention Module) is an attention mechanism that combines channel attention and spatial attention, as shown in Figure 6. Using channel attention (Figure 6a) adjusts the importance of each channel in the feature map, allowing the network to focus more on the most relevant and important channels for texture feature extraction. Using spatial attention (Figure 6b) increases the weights of regions containing rich texture information while decreasing the weights of regions with less texture information.

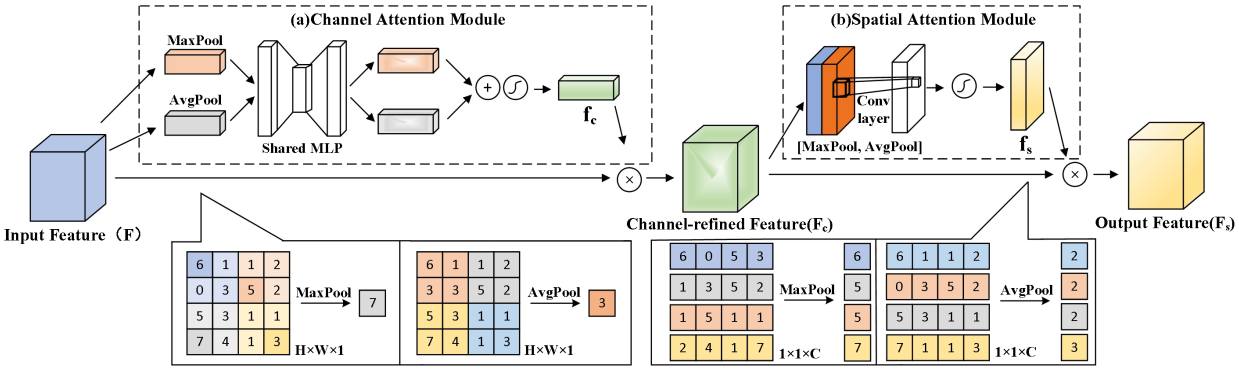

Figure 6. CBAM attention.

The formula for the channel attention module is shown in Equation (3):

$$f_c = \sigma(MLP(AvgPool(F)) + MLP(MaxPool(F))) \tag{3}$$

where F represents the input feature map; $AvgPool$ and $MaxPool$ denote global average pooling and global max pooling, respectively; MLP represents a multi-layer perceptron; σ denotes the $Sigmoid$ function.

The formula for the spatial attention module is given in Equation (4):

$$f_s = \sigma\left(c^{7\times7}([AvgPool(F_c); MaxPool(F_c)])\right) \tag{4}$$

where F_c represents the feature map after applying channel attention; $c^{7\times7}$ denotes a convolution operation with a kernel size of 7×7. The choice of this kernel size is due to the application of CBAM directly on the original wood texture patterns in the paper, where the main texture spans a large area within the pattern. To more comprehensively capture global texture information and the relative positions between main textures, we use a larger 7×7 convolution kernel. This not only helps capture more global texture features but also allows the neural network to focus more on the main textures, reducing interference from minor textures in the model; $[AvgPool(F_c); MaxPool(F_c)]$ represents the concatenation of average pooling and max pooling structures along the channel dimension.

In edge feature extraction, the element-wise multiplication of the wood texture image with the mask matrix obtains the edge texture pattern. The mask formula is shown as Equation (5):

$$Edge\_Pattern_{i,j} = Wood\_Texture_{i,j} \odot M_{i,j} \tag{5}$$

where $Edge\_Pattern$ represents the edge texture image; $Wood\_Texture$ denotes the input wood texture image; M represents the mask matrix with the same shape as $Wood\_Texture$, and the form of M is shown as Equation (6):

$$\begin{bmatrix} 1 & 1 & 1 & \dots & 1 & 1 & 1 \\ 1 & \ddots & \vdots & & \vdots & \ddots & 1 \\ 1 & \dots & 1 & \dots & 1 & \dots & 1 \\ 1 & \dots & 0 & \dots & 0 & \dots & 1 \\ \vdots & & \vdots & \ddots & \vdots & & \vdots \\ 1 & \dots & 0 & \dots & 0 & \dots & 1 \\ 1 & \dots & 1 & \dots & 1 & \dots & 1 \\ 1 & \ddots & \vdots & & \vdots & \ddots & 1 \\ 1 & \dots & 1 & \dots & 1 & \dots & 1 \end{bmatrix} \tag{6}$$

3.2. Texture Feature Aggregation and Matching Module

The texture feature aggregation-matching module consists of two parts: the Feature Fusion Module and the Similarity Computation Module. Since the texture feature extraction module in this study returns both global texture features and edge texture features, the traditional fusion method of element-wise subtraction and absolute value leads to feature confusion, failing to effectively capture the potential correlations and interactions between different feature dimensions and resulting in a lack of richness and accuracy in the fused information representation. To address this, we propose an improved feature fusion method from the perspective of feature correlation, which, while preserving the internal structure of the features, dynamically adjusts the fusion weights to promote the model's ability to explore the relationships between feature dimensions. Meanwhile, depthwise separable convolutions are employed to optimize the similarity computation module, reducing the model's parameter count and improving computational efficiency while maintaining performance. The improved texture feature aggregation module is shown in Figure 7.

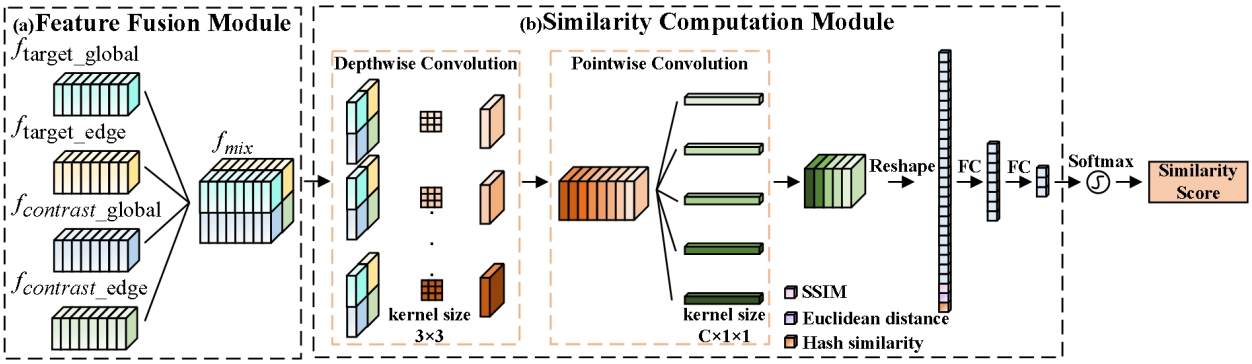

Figure 7. Texture feature aggregation and matching module.

3.2.1. Feature Fusion Module

The role of the Feature Fusion Module is to integrate the global features of the target image, x_1, obtained from the texture feature extraction module, the edge features of the target image, x_1, the global features of the reference image, x_2, and the edge features of the reference image, x_2. Since these four features differ in their sources and types, the meaning represented at each corresponding position is also different. Therefore, directly using the method of subtracting the absolute value pair-wise is not suitable for feature fusion. However, these four features all come from the same feature extraction network, so they share commonalities in structure; they have the same shape and similar receptive fields in the same channels. Based on these characteristics, a feature fusion method based on feature correlation is designed. The specific fusion method is as follows:

Firstly, we concatenate the global features and edge features of the target image along the width (W) direction. Secondly, we concatenate the global features and edge features of the reference image in the same manner. The concatenation formula is shown in Equation (7):

$$f_{target(contrast)} = concat\left(f_{global}, f_{edge}, dim = 2\right) \tag{7}$$

where $f_{target(contrast)}$ represents the fused features of the target or reference image after fusion. f_{global} and f_{edge} denote the respective global features and edge features. $concat(dim = 2)$ signifies the concatenation operation along the width dimension.

Finally, we concatenate the already concatenated features of the target image and reference image along the height (H) dimension. The concatenation formula is shown in Equation (8):

$$f_{mix} = concat\left(f_{target}, f_{contrast}, dim = 3\right) \tag{8}$$

where f_{target} and $f_{contrast}$ represent the features of the target image and reference image obtained in the previous step; $concat(dim = 3)$ denotes the concatenation operation along the height

dimension; f_{mix} represents the newly fused features. By using this method of feature fusion, it is possible to maximize the preservation of the internal relationships of the original features. Additionally, the new features combine all the features of the reference image and the comparison image spatially while still having different receptive fields in the channels.

3.2.2. Similarity Computation Module

The original similarity metric module directly flattens the fused features into a one-dimensional vector and calculates similarity by using a fully connected network, which ignores the internal relationships among features, making it difficult to accurately fit the mapping relationship between features and similarity by not analyzing the relationship between the target and comparison images from the spatial and channel dimensions. Furthermore, the feature maps (after extraction) are large, and directly flattening them for use in a fully connected network increases the parameter count to millions. To address this, we designed an optimized similarity computation module that utilizes depthwise separable convolutions to improve similarity calculations. Additionally, to enhance the accuracy of similarity computation, we incorporate the structural similarity, Euclidean distance, and hash similarity of the target and comparison images as extra feature inputs to the network. The specific steps for similarity computation are as follows:

Depthwise separable convolution consists of depthwise convolution and pointwise convolution. First, depthwise convolution is applied to each channel of the feature map separately, integrating global and edge features and learning the relationship between the target image and comparison image under the same receptive field. Next, pointwise convolution is performed on features from different channels using a 1×1 convolution kernel to learn the relationships between features under different receptive fields. Finally, the feature vectors after pointwise convolution are flattened, and the structural similarity (SSIM), Euclidean distance, and hash similarity of the target and comparison images are added at the end, enhancing the dimensionality of features during similarity computation. This enhances the feature dimensionality of the model during similarity computation.

The multidimensional features provided by SSIM (such as brightness, contrast, and structural information) can enrich the input features of neural networks, helping the model assess wood texture similarity from multiple aspects. The structural similarity calculation formula is shown in Equation (9):

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{\left(\mu_x^2 + \mu_y^2 + c_1\right)\left(\sigma_x^2 + \sigma_y^2 + c_2\right)} \tag{9}$$

where x and y, respectively, denote the reference texture image and the comparison texture image; μ_x and μ_y represent the local means of x and y; σ_x and σ_y denote the local standard deviations of x and y; σ_{xy} represents the local covariance of x and y; c_1 and c_2 represent two constants.

Combining Euclidean distance with other features can supplement more complex features. The formula for calculating the Euclidean distance is shown in Equation (10):

$$d(x,y) = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}\left(x_{(i,j)} - y_{(i,j)}\right)^2} \tag{10}$$

where $x_{(i,j)}$ and $y_{(i,j)}$, respectively, represent the pixel values of the reference texture image and the comparison texture image at position (i,j).

Hash similarity exhibits robustness against minor changes in images (such as rotation, scaling, slight noise, etc.), helping the model accurately determine similarity even when facing subtle variations in wood texture. The hash similarity calculation formula is shown in Equation (11):

$$similarity = 1 - \frac{\sum_{i=1}^{n}(H_1(i) \odot H_2(i))}{n} \tag{11}$$

where $H_1(i)$ and $H_2(i)$, respectively, represent the hash codes of the reference texture image and the comparison texture image; n denotes the length of the hash code; $\odot$ represents the bitwise XOR operation.

Finally, we establish the mapping relationship between texture features and texture similarity using a fully connected network, and we output the texture similarity score through a sigmoid function.

4. Dataset Introduction

4.1. Dataset Design Ideas

Due to the lack of public datasets in the field of wood-texture-similarity matching, this study created a custom wood-texture-similarity-matching dataset. The dataset is designed to simulate human visual perception using machine vision, enabling the model to better recognize and assess the similarity of wood images. The classification of the labels in the dataset is based on human visual perception of color and texture:

1. Class 0 label: Color consistent and texture similar, representing human visual recognition of the same material and surface characteristics;
2. Class 1 label: Color consistent but texture dissimilar, aiming to simulate how humans recognize different textures under the same color;
3. Class 2 label: Significant color difference, intended to examine whether the model can differentiate based on texture features under varying color conditions.

This labeling classification method helps the model learn more dimensions of features, particularly the ability to distinguish between similar and dissimilar factors under different conditions, thereby enhancing the model's generalization capability. Furthermore, in the formulation of Class 0 and Class 1 labels, this study also considers the impact of overall visual similarity and content-based similarity on the experimental results. If the dataset is labeled solely based on overall visual similarity, the images returned by the model may have colors similar to the target image but may not be similar in texture details such as stripe direction and spacing. Conversely, if the dataset is labeled solely based on content, the results would be the opposite. Based on these two comparisons, the dataset design clearly distinguishes Class 0 (where both color and texture are similar) from Class 1 (where color is consistent but texture differs), helping the model accurately identify wood images that truly possess similar texture features as required for the experiments. This distinction allows the model to better learn the differences between texture structure and overall visual during training, thereby improving the classification accuracy and performance of the model.

4.2. Labeling Strategy

When humans perform a comparison of wood texture similarity, they not only evaluate aspects such as the direction, arrangement, and distribution patterns of the texture but also consider multiple factors like texture coarseness, details, color, tone, surface gloss, and tactile feel. This allows the human eye to form an intuitive perception during wood-texture-similarity matching. However, there is no single metric that can meet all these complex requirements, and any single metric cannot fully and accurately reflect human subjective perception. Therefore, this dataset is labeled manually to better simulate human visual judgment logic. This annotation method is also a common method for label creation regarding datasets in the field of perceptual similarity computation. During the labeling process, we required annotators to evaluate similarity from two perspectives: texture characteristics and overall color. For texture, we focused on whether the primary texture's direction, quantity, coarseness, and relative positions were generally consistent. For overall color, we assessed whether the base color of the wood, the color of the texture, and the depth of the texture color were generally consistent. If both texture and color met the required criteria, the image pair was categorized as Class 0 (highly similar texture). If only color was consistent, the pair was classified as Class 1. All other cases were classified as Class 2.

Moreover, although subjectivity is inevitably present in the labeling process, this does not reduce the validity of the experiment; on the contrary, it helps the model more accurately simulate human visual perception. To reduce the potential subjective bias introduced during manual labeling, we adopted the "multiple raters averaging" method. Specifically, we invited multiple annotators to rate the same image pair, and the average rating was used as the final label. This method not only effectively reduces individual differences among annotators but also improves label consistency and reliability, thereby enhancing the dataset's accuracy and ensuring the validity of the experimental results.

4.3. Dataset Production

The dataset used in this paper is a self-made wood-texture-similarity matching dataset. The creation of this dataset involves three steps: wood texture image collection, texture image preprocessing, and label creation, as detailed below:

During the wood texture image collection stage, to ensure uniform lighting on the wood surface and reduce the impact of factors such as exposure on image quality, supplementary lighting was used in the shooting environment. Subsequently, an OscarF810C industrial camera was employed to capture 3000 original wood texture images with a resolution of 2048 × 2048 pixels.

During the texture image preprocessing phase, 1000 images were randomly selected from the original set of 3000 wood texture images with a resolution of 2048 × 2048 pixels, and these images were randomly rotated. The rotation angles included 0°, 90°, 180°, and 270°, thereby expanding the dataset to 4000 images. Subsequently, 500 images were randomly selected from the 4000-image dataset and subjected to random translations. The translation range was ±10% in both the horizontal and vertical directions. This operation simulated minor positional deviations that may occur during the acquisition process, further expanding the dataset to 5000 images. Next, Perlin noise was applied to fine-tune and simulate the 5000 wood texture images, generating 5000 new images consistent with natural texture characteristics. This introduced natural variations, increasing the diversity of the dataset. Finally, a sliding window method was employed to segment each 2048 × 2048-pixel original image into 176 × 176-pixel sub-images in a non-overlapping manner, with a sliding step size of 400 × 400 to prevent excessive similar images from biasing model training. The segmented images underwent a quality evaluation to select images with clear textures, complete boundaries, and distinct features. Ultimately, 10,000 images were retained for the construction of the wood texture similarity dataset.

During the label creation stage, two images were randomly selected from the preprocessed set of 10,000 wood texture images to form an image pair, with one serving as the target image and the other as the comparison image. Subsequently, each image pair was manually classified based on similarity. The classifications were as follows: Label 0 indicated that the two images were highly similar in both color and texture; Label 1 indicated similarity in color but significant differences in texture; Label 2 indicated significant differences in both color and texture. Considering the continuity characteristics of wood texture, areas close to each other on the same board often exhibit high texture similarity. However, differences may arise in the texture of the same type of wood across different boards, even when the color is similar. Similarly, color differences may exist between different types of wood. Therefore, during the manual labeling process, these texture characteristics were taken into account, and multiple reviews were conducted to ensure the accuracy and consistency of the labels while minimizing significant imbalances in the number of data points across different categories. In total, 7588 image pairs were created, with each pair assigned a corresponding similarity label. These pairs were used to train and test the wood texture similarity dataset. The similarity classification criteria are shown in Table 1:

Table 1. Dataset labels.

Label	Meaning
0	The image pair has consistent colors and similar textures.
1	The image pair has consistent colors but dissimilar textures.
2	The image pair has significant color differences.

The dataset labeling is shown in Figure 8.

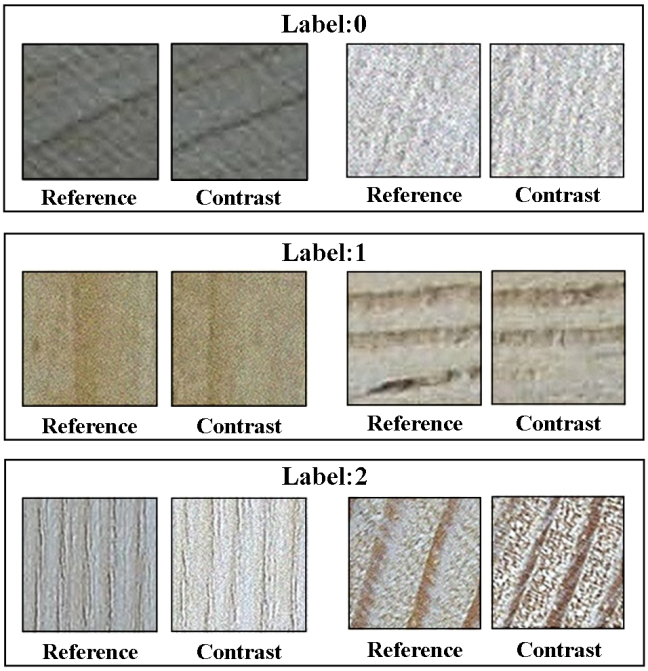

Figure 8. Sample dataset.

5. Experiment and Result Analysis

5.1. Experimental Introduction

In the experiment, the dataset was randomly split into training and test sets in a 4:3 ratio, where the training set was used for model training and the test set for model evaluation. Moreover, during the training process, it was ensured that all models were fully trained until convergence to the optimal state. Furthermore, WTSM-SiameseNet, which was designed in this study, utilizes the PyTorch deep learning framework. During training, it employs the Adam optimizer with a learning rate of 1×10^{-3} and a batch size of 10. The remaining experimental hardware devices and software platforms are shown in Table 2.

Table 2. Hardware equipment and software platform.

Name	Configuration Instruction
Image Acquisition Equipment	AVT Oscar F-810C
GPU	NVIDIA RTX 3080/NVIDIA A40 48 G
CPU	Intel Core i7-12700K
Operating System	Windows10/Ubuntu 23.0
Deep Learning Framework	Pytorch 1.13.0 + cu117
Version of Python	3.10.13

5.1.1. Training Process

To train the model, we first need to set the training parameters, including choosing the Adam optimizer and setting the epoch to 200 and the Batch Size to 10. Next, we input

the texture image pairs and their corresponding similarity labels. Then, we used WTSM-SiameseNet to calculate the similarity between the two images. Finally, we performed a cross-entropy loss operation on the similarity categories and the similarity labels of the image pairs to obtain the training loss, and we optimized the parameters in WTSM-SiameseNet using backpropagation to enable the model to more accurately predict the similarity of the input image pairs. The calculation of the cross-entropy loss function is shown in Equation (12):

$$Loss(y, \hat{y}) = -\sum_{i=1}^{3} y_i \log(\hat{y}_i) \tag{12}$$

where y represents the similarity label, with only the elements in the corresponding class being 1 and the others being 0; $\hat{y}$ represents the model's prediction value, which is a vector representing the probability distribution of the predicted classes by the model; y_i is the ith element in y; $\hat{y}_i$ is the ith element in $\hat{y}$.

5.1.2. Evaluation Metrics

The model's performance was evaluated using accuracy, precision, recall, and F1-score. Accuracy, precision, and recall are calculated based on common confusion matrix metrics (such as TP, TN, FP, and FN). The F1-score combines precision and recall. The formulas for calculating accuracy, precision, recall, and F1-score are shown in Equations (13)–(16):

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \tag{13}$$

$$Precision = \frac{TP}{TP + FP} \tag{14}$$

$$Recall = \frac{TP}{TP + FN} \tag{15}$$

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{16}$$

5.2. Comparative Experiment

5.2.1. SiameseNet Comparative

To validate the performance of WTSM-SiameseNet in the wood-texture-similarity matching task, comparative experiments were conducted using SiameseNet, SE-SiameseNet, Res-SiameseNet, and CS-SiameseNet. Among them, SiameseNet is the basic Siamese network model, with a nine-layer convolutional network for feature extraction and a two-layer fully connected network for similarity computation; SE-SiameseNet is an improved Siamese network that uses SENet50 as the backbone, as proposed by Yu et al. in 2020 [31]; Res-SiameseNet, also proposed by Yu et al. in 2020, uses ResNet50 as the backbone and incorporates attention mechanisms to further improve the Siamese network; CS-SiameseNet is the model proposed by Yan et al. in 2023 [26], which improves the performance of the Siamese network using CBAM attention with VGGNet as the backbone. The minimum training loss and the number of training epochs to obtain the best model for each group are shown in Table 3:

Table 3. Best loss and optimal model for model training.

	SiameseNet	**SE-SiameseNet**	**Res-SiameseNet**	**CS-SiameseNet**	**WTSM-SiameseNet**
Cross-entropy loss	0.1050	0.0860	0.0839	0.0558	0.0461
Best model (epoch)	79	157	152	169	154

The changes in training loss for the comparative experiment models are shown in Figure 9:

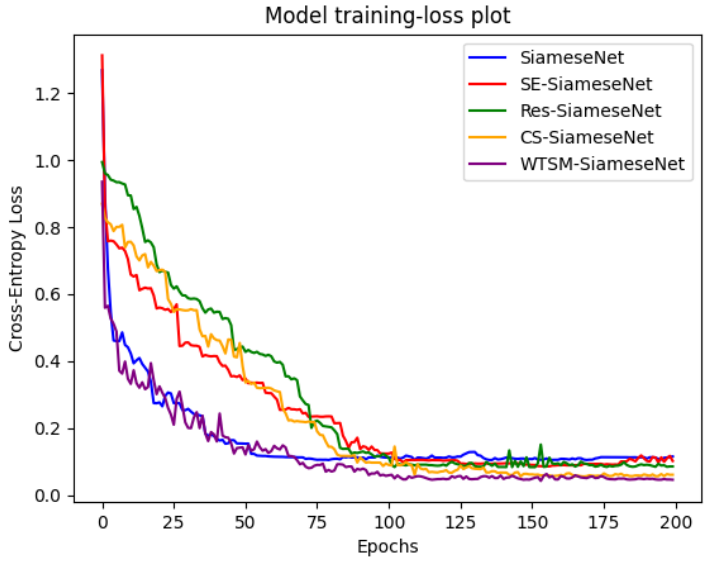

Figure 9. Training loss.

As shown in the figure, the original SiameseNet model has the fastest fitting speed, with the loss tending to flatten after the 50th epoch, while the losses of the other models tend to flatten after the 110th epoch. This is because the SiameseNet network structure is relatively simple. In terms of training model loss, our proposed WTSM-SiameseNet performs better than the other models.

To further compare the generalization abilities of each model, experiments were conducted on the test set, and the comparative experimental results are shown in Table 4:

Table 4. Comparison of experimental results.

Model	Params	Accuracy/%	Precision/%	Recall/%	F1-Score
SiameseNet	7.3M	83.76	83.03	85.15	0.840
SE-SiameseNet	27.1M	87.85	88.39	86.83	0.867
Res-SiameseNet	25.6M	89.08	88.45	86.66	0.875
CS-SiameseNet	17.2M	93.81	90.66	91.78	0.912
WTSM-SiameseNet	20.7M	96.67	97.24	96.81	0.970

The results in Table 4 indicate that, as a baseline model, SiameseNet performs relatively poorly. This may be due to its relatively simple structure, which uses only a nine-layer convolutional network for feature extraction and lacks the ability to learn complex texture features. Additionally, using a two-layer fully connected network for similarity computation may lead to information loss, affecting classification performance. SE-SiameseNet introduces SENet50 as the backbone, significantly improving model performance, with an increase of 4.09% in accuracy and 5.36% in precision. This indicates that the SE (Squeeze-and-Excitation) mechanism effectively enhances the model's focus on important features, improving both accuracy and precision. Compared to SiameseNet, SE-SiameseNet is better at capturing subtle differences in texture features. Res-SiameseNet uses ResNet50 as the backbone, utilizing residual connections to improve gradient propagation and enhance model training effectiveness. Although there is a significant improvement over SiameseNet, its performance is similar to SE-SiameseNet, with only a 1.23% increase in accuracy and a 0.06% increase in precision. This may be because the attention mechanism in Res-SiameseNet may not have fully played its role in comparing similarities, leading to a relatively small improvement. CS-SiameseNet adopts VGGNet with the CBAM attention mechanism as its backbone, greatly improving model performance. CBAM enhances important features and suppresses irrelevant ones, making the model's judgments on texture

similarity more precise; this results in significant increases in precision and recall of 4.73% and 2.21% over Res-SiameseNet, respectively. WTSM-SiameseNet, which was designed in this study, improves the texture feature extraction module and the texture feature aggregation matching module, making the model more suited for wood-texture-similarity matching tasks. In the test set, WTSM-SiameseNet achieved an accuracy of 96.67%, which is an increase of 2.86% over CSNet. At the same time, the model achieved a precision of 97.24%, significantly higher than that of other models. Moreover, WTSM-SiameseNet achieves high precision performance with a moderate number of parameters, indicating the efficiency of its structural design.

5.2.2. Pseudo-SiameseNet Comparative

To verify whether the pseudo-Siamese network can further improve the performance of wood-texture-similarity matching, three comparison experiments were designed in this study; the experimental results are shown in Table 5, and the experimental setup is as follows:

1. Experiment 1 utilized the existing pseudo-Siamese network architecture [32] for wood-texture-similarity matching. The specific approach involves using two weight-independent feature extraction networks to extract features from the target and contrast images and then compute the texture similarity.
2. Experiment 2 is based on the Siamese network architecture proposed in this study. Weight-independent feature extraction networks are used to extract features from the target and contrast texture patterns, and the corresponding mask patterns are extracted using the same feature extraction network as the original images.
3. Experiment 3 is also based on the Siamese network architecture proposed in this study. However, the same feature extraction network is used for the target and contrast texture patterns, and a separate weight-independent feature extraction network is used for their corresponding mask texture patterns.

Table 5. Pseudo-SiameseNet comparative experiment.

Group	Accuracy/%	Precision/%	Recall/%	F1-Score
1	72.53	93.62	68.81	0.793
2	74.23	79.68	75.17	0.773
3	98.09	96.41	97.52	0.969
Our Method	96.67	97.24	96.81	0.970

According to the experimental results, the performance differences between Experiment 1 and Experiment 2 primarily arise from the different feature extraction methods. Experiment 1 used two weight-independent feature extraction networks, which may have led to insufficient alignment of texture features between the target image and the contrast image, affecting the texture similarity calculation and resulting in a lower recall (68.81%). Although Experiment 2 adopted an improved Siamese network architecture, it still used the same feature extraction network to extract features for the mask pattern from the original image, which failed to effectively capture the detailed texture patterns, resulting in lower precision (79.68%). Thus, the results of Experiment 1 and Experiment 2 indicate that the different branch weights in the pseudo-Siamese network may lead to the output of different features from the two branches, affecting the similarity calculation and reducing the matching accuracy. Although the pseudo-Siamese network can effectively match images of the same object in different modalities, in the wood-texture-matching task, the primary comparison involves the similarity of different textures within the same modality. Therefore, a unified feature extraction method may be more suitable. In contrast, Experiment 3 improved performance significantly by optimizing the feature extraction module. It used the same feature extraction network for both target and contrast textures while independently processing the mask textures, achieving the following: an accuracy of 98.09%, a precision of

96.41%, and a recall of 97.52%. This indicates that by properly combining independent and shared networks, the model can better align texture features and reduce information loss. The success of Experiment 3 suggests that the rational combination of shared and independent feature extraction networks can enhance texture-similarity-matching performance. Compared to the Siamese network in this study (accuracy: 96.67%, precision: 97.24%, and recall: 96.81%), the results of Experiment 3 are similar, with a slight advantage in accuracy. This provides insight for further optimizing the feature extraction network design from this study. Since mask images and original images originate from the same source but exhibit differences in feature representation, extracting features separately from both original texture images and mask texture images, although increasing computational load, may be more beneficial for wood-texture-similarity matching, especially in optimizing mask pattern feature extraction. Overall, these comparative experiments suggest that combining different feature extraction module designs can effectively enhance model performance. In the future, further exploration of the optimization directions for mask pattern feature extraction in different network architectures could be conducted, along with experimenting with more feature fusion methods to improve the model's generalization ability.

5.3. Optimization Experiment

To validate the effectiveness of concurrent attention, this study designed optimization experiments to evaluate the contribution of each attention mechanism to model performance. It is important to note that in experiments without the edge attention mechanism, due to changes in the number of features, feature fusion based on feature correlation cannot be used. Instead, vector subtraction to take the absolute value was used for fusion, while the rest of the modules remained consistent. The concurrent attention optimization experiment is shown in Table 6, where "✓" indicates the use of the corresponding module.

Table 6. Concurrent attention optimization experiment.

CBAM	Edge Att	Accuracy/%	Precision/%	Recall/%	F1-Score
		86.46	84.25	86.04	0.851
✓		94.61	92.58	93.72	0.931
✓	✓	96.67	97.24	96.81	0.970

The data from the table show that from not using any attention mechanism to using only the CBAM attention mechanism, accuracy improved by 8.15%, and the F1-score increased by 0.080. This indicates that the CBAM attention mechanism enhances important features and suppresses noise, helping the model focus on key texture information and thereby improving classification accuracy. When the concurrent attention mechanism was introduced, the model's accuracy further increased to 96.67%, with an F1-score of 0.970. This indicates that the use of the concurrent attention mechanism further optimized model performance in feature extraction and similarity computation. This also demonstrates that the concurrent attention mechanism combines the advantages of CBAM and edge attention, allowing the model to simultaneously focus on global and local features, enhancing its ability to learn complex texture features. This complementary information characteristic enables the model to have a more comprehensive understanding of image content, thus enhancing performance. By employing different types of attention mechanisms, the model may better generalize to unseen samples, reducing overfitting regarding the training data. This capability improves the model's performance on the test set. In summary, the attention mechanism significantly enhances model performance, especially with the introduction of the concurrent attention mechanism, which further strengthens the understanding and classification ability of texture features. By using a combination of different mechanisms, the model can better capture the complex features of wood textures, achieving higher accuracy and F1-scores.

5.4. Ablation Experiment

To verify the impact of improvements to attention (Att), feature extraction network (FEN), the feature fusion module (FFM), and the similarity computation module (SCM) on model performance, ablation experiments were conducted on the four modules. The ablation experiment results are shown in Table 7, where "✓" indicates the use of the corresponding module.

Table 7. Main module ablation experiments.

Att	FEN	FFM	SCM	Accuracy/%	Precision/%	Recall/%	F1-Score
				85.73	84.06	85.77	0.849
✓				82.84	82.02	84.27	0.831
✓	✓			86.86	87.48	85.86	0.866
✓	✓	✓		95.38	94.49	95.58	0.950
✓	✓	✓	✓	96.67	97.24	96.81	0.970

The data from Table 6 indicates that using the concurrent attention mechanism alone leads to a decrease in model performance. This is because using the concurrent attention mechanism generates four texture features, and directly fusing them by taking the absolute value of the vector difference increases feature complexity, making similarity computation more difficult. Secondly, the improved feature extraction network effectively alleviates the limitation of the receptive field by increasing its dimensionality, thus enhancing feature quality. This resulted in an increase of 1.13% in accuracy and 3.42% in precision for the model. Furthermore, the improved feature fusion module can fuse features based on their correlation, effectively reducing the destruction of the original feature structure and thereby enhancing feature expressiveness. The experimental results show that the model's accuracy, precision, recall, and F1-score increased by 8.52%, 7.01%, 9.72%, and 0.084, respectively. Finally, the improved similarity computation module employs depthwise separable convolutions, which, while only yielding a limited increase of 1.29% in model accuracy, reduces the number of parameters by 86.8%.

5.5. Matching Example

To visually observe the matching effect of wood texture similarity using WTSM-SiameseNet, three sets of defect wood repair instances were designed. In the early stages of the project, texture generation techniques [33] were used to study texture generation for defective wood areas. The purpose of this experiment was to use the generated textures as target images to search for similar boards in the texture library for practical wood repair work. The specific steps of the experiment are as follows:

First, locate and extract the defective area image (Defect Image) from the original image of the solid wood board (Original Image). Next, we used MRS-Transformer (the method from reference [33]) to generate a texture for the defective area, obtaining the generating textures (Generating Textures). Then, use WTSM-SiameseNet to compare the generated wood texture with the real textures in the database to select the highest matching texture (Matching Textures). Finally, replace the generated texture with the matching texture to repair the defective area in the original solid wood board. The wood defect texture repair process is shown in Figure 10, where the red box area is the repair area.

From the experiment, it can be seen that WTSM-SiameseNet performed well in the task of wood-texture-similarity matching. Using this method for tasks such as wood defect repair can significantly improve the quality and consistency of the repaired texture, making the texture better meet the subjective perception of the human eye.

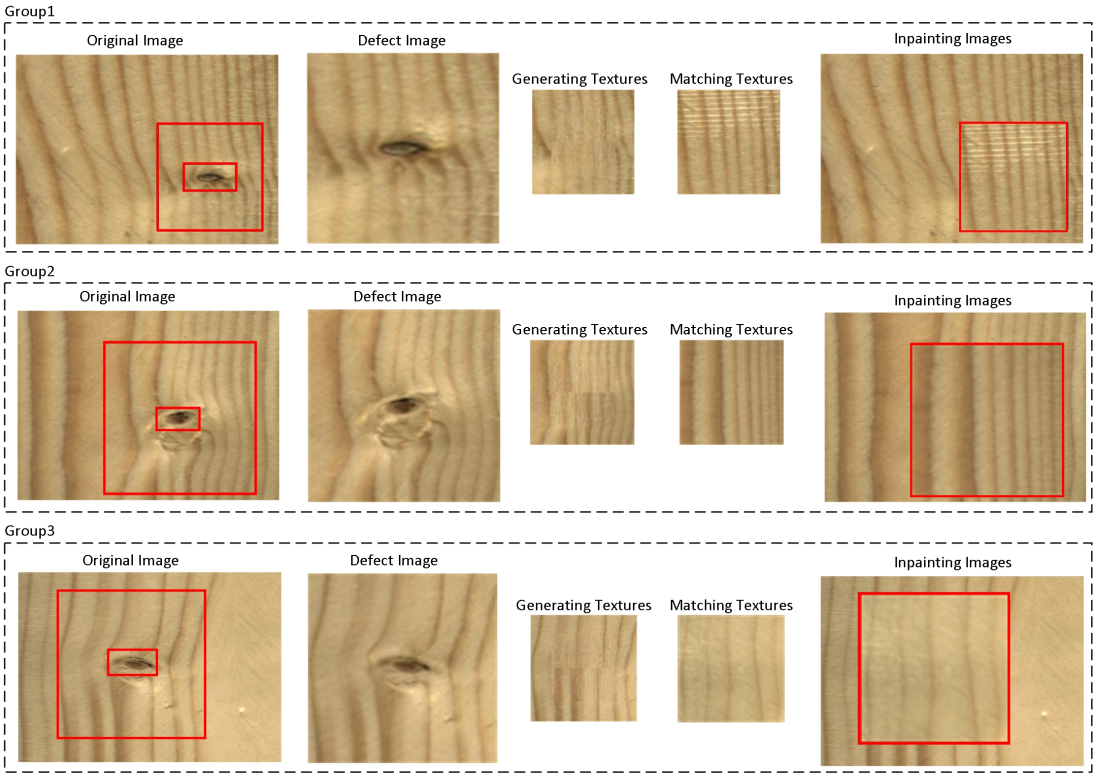

Figure 10. Wood-texture-similarity matching example.

6. Conclusions

In the wood-texture-similarity-matching task, the reliability of the similarity score depends both on the model's ability to extract texture features and on the adaptability of the similarity measure function to the task. WTSM-SiameseNet, which is based on the Siamese network architecture and is studied in this paper, can efficiently extract wood texture features through concurrent attention mechanisms and a multi-receptive field feature extraction network. Based on this, the model improves the similarity measure module using feature fusion methods based on feature correlation and depthwise separable convolutions, enabling it to adaptively generate wood texture similarity scores that align with human subjective perceptions while reducing the computational load of model parameters. The evaluation results on the wood texture similarity dataset indicate that the model's scoring of wood texture similarity aligns with human subjective perception. However, in real production environments, incorrectly predicting dissimilar textures as similar textures can have a more significant negative impact than other prediction errors. Therefore, future research directions will focus on further improving the loss function by incorporating penalty mechanisms that better align with the wood-texture-similarity-matching task, enhancing the model's learning efficiency and capability.

Author Contributions: Conceptualization, Y.Z. and G.W.; methodology, G.W.; software, G.W.; validation, Y.Z., G.W. and H.Y.; formal analysis, Y.Z., H.Y., G.W. and S.S.; investigation, Y.Z. and G.W.; data curation, G.W.; writing—original draft preparation, G.W.; writing—review and editing, Y.Z. and S.S.; supervision, Y.Z.; project administration, Y.Z. and G.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets presented in this study are not readily available because the data are part of an ongoing study. The data presented in this study are available upon request from the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

GLCM	Gray-Level Co-occurrence Matrix
SSIM	Structural Similarity Index Measure
BoVW	Bag of Visual Words
CNNs	Convolutional neural networks
WTSM-SiameseNet	Wood-Texture-Similarity Matching-SiameseNet
MRF-Resnet	Multi-scale Receptive Field-Resnet
CBAM	Convolutional Block Attention Module
TP	True Positive
TN	True Negative
FP	False Positive
FN	False Negative
Att	Attention
FEN	Feature extraction network
FFM	Feature fusion module
SCM	Similarity computation module

References

1. Heidari, M.; Fouladi-Ghaleh, K. Using siamese networks with transfer learning for face recognition on small-samples datasets. In Proceedings of the 2020 International Conference on Machine Vision and Image Processing (MVIP), Qom, Iran, 18–20 February 2020; IEEE: Piscataway Township, NJ, USA, 2020; pp. 1–4.
2. Alrashidi, A.; Alotaibi, A.; Hussain, M.; AlShehri, H.; AboAlSamh, H.A.; Bebis, G. Cross-sensor fingerprint matching using siamese network and adversarial learning. *Sensors* **2021**, *21*, 3657. [CrossRef] [PubMed]
3. He, Z.; Lin, M.; Xu, Z.; Yao, Z.; Chen, H.; Alhudhaif, A.; Alenezi, F. Deconv-transformer (DecT): A histopathological image classification model for breast cancer based on color deconvolution and transformer architecture. *Inf. Sci.* **2022**, *608*, 1093–1112. [CrossRef]
4. Chen, Y.; Lin, M.; He, Z.; Polat, K.; Alhudhaif, A.; Alenezi, F. Consistency-and dependence-guided knowledge distillation for object detection in remote sensing images. *Expert Syst. Appl.* **2023**, *229*, 120519. [CrossRef]
5. Wang, Z.; Zhuang, Z.; Liu, Y.; Ding, F.; Tang, M. Color classification and texture recognition system of solid wood panels. *Forests* **2021**, *12*, 1154. [CrossRef]
6. Chen, X.; Wang, X.; Zhang, K.; Fung, K.M.; Thai, T.C.; Moore, K.; Mannel, R.S.; Liu, H.; Zheng, B.; Qiu, Y. Recent advances and clinical applications of deep learning in medical image analysis. *Med. Image Anal.* **2022**, *79*, 102444. [CrossRef] [PubMed]
7. Zhu, L.; Chen, T.; Yin, J.; See, S.; Liu, J. Learning gabor texture features for fine-grained recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 2–3 October 2023; IEEE: Piscataway, NJ, USA, 2023; pp. 1621–1631.
8. Indra, D.; Fadlillah, H.M.; Ilmawan, L.B. Rice Texture Analysis Using GLCM Features. In Proceedings of the 2021 International Conference on Electrical, Computer and Energy Technologies (ICECET), Cape Town, South Africa, 9–10 December 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1–5.
9. Srivastava, D.; Rajitha, B.; Agarwal, S.; Singh, S. Pattern-based image retrieval using GLCM. *Neural Comput. Appl.* **2020**, *32*, 10819–10832. [CrossRef]
10. Li, N.; Qi, W.; Jiao, J.; Li, A.; Li, L.; Xu, W. SPCC: A superpixel and color clustering based camouflage assessment. *Multimed. Tools Appl.* **2024**, *83*, 26255–26279. [CrossRef]
11. Sun, J. Research on Image Copy-paste Tamper Detection Based on Gray Scale Co-occurrence Matrix and Graph Neural Network. *Int. J. Netw. Secur.* **2023**, *25*, 1002–1009.
12. Ding, K.; Ma, K.; Wang, S.; Simoncelli, E.P. Image quality assessment: Unifying structure and texture similarity. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *44*, 2567–2581. [CrossRef] [PubMed]
13. Ouni, A.; Royer, E.; Chevaldonné, M.; Dhome, M. A hybrid approach for improved image similarity using semantic segmentation. In Proceedings of the 15th International Symposium on Advances in Visual Computing (ISVC 2020), San Diego, CA, USA, 5–7 October 2020; Springer: Berlin/Heidelberg, Germany, 2020; pp. 647–657.
14. Bakheet, S.; Mofaddel, M.; Soliman, E.; Heshmat, M. Content-based image retrieval using BRISK and SURF as bag-of-visual-words for Naïve Bayes classifier. *Sohag J. Sci.* **2023**, *8*, 329–335. [CrossRef]

15. Ali, R.; Maheshwari, M. Feature detection and extraction techniques with different similarity measures using bag of features scheme. In *Applications of Mathematical Modeling, Machine Learning, and Intelligent Computing for Industrial Development*; CRC Press: Boca Raton, FL, USA, 2023; pp. 203–222.
16. Bharathi, K.; Mohan, M.C. Bag of Visual Words and Cnn Approaches for Content-Based Image Retrieval Using Hog, Gch And Orb Features. *J. Theor. Appl. Inf. Technol.* **2022**, *100*, 17.
17. Yao, R.; Lin, G.; Xia, S.; Zhao, J.; Zhou, Y. Video object segmentation and tracking: A survey. *ACM Trans. Intell. Syst. Technol. (TIST)* **2020**, *11*, 1–47. [CrossRef]
18. Chen, H.; Song, J.; Han, C.; Xia, J.; Yokoya, N. Changemamba: Remote Sensing Change Detection with Spatio-Temporal State Space Model. Available online: https://arxiv.org/abs/2404.03425 (accessed on 24 April 2024).
19. Ahrabian, K.; BabaAli, B. Usage of autoencoders and Siamese networks for online handwritten signature verification. *Neural Comput. Appl.* **2019**, *31*, 9321–9334. [CrossRef]
20. Tao, C.; Zhu, X.; Su, W.; Huang, G.; Li, B.; Zhou, J.; Qiao, Y.; Wang, X.; Dai, J. Siamese image modeling for self-supervised vision representation learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; IEEE: Piscataway, NJ, USA, 2023; pp. 2132–2141.
21. Figueroa-Mata, G.; Mata-Montero, E. Using a convolutional siamese network for image-based plant species identification with small datasets. *Biomimetics* **2020**, *5*, 8. [CrossRef] [PubMed]
22. Hudec, L.; Bencsova, W. Texture similarity evaluation via siamese convolutional neural network. In Proceedings of the 25th International Conference on Systems, Signals and Image Processing (IWSSIP), Maribor, Slovenia, 20–22 June 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1–5.
23. Gao, Y.; Gan, Y.; Qi, L.; Zhou, H.; Dong, X.; Dong, J. A perception-inspired deep learning framework for predicting perceptual texture similarity. *IEEE Trans. Circuits Syst. Video Technol.* **2019**, *30*, 3714–3726. [CrossRef]
24. Dong, X.; Dong, J.; Chantler, M.J. Perceptual texture similarity estimation: An evaluation of computational features. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *43*, 2429–2448. [CrossRef] [PubMed]
25. Cao, W.; Feng, Z.; Zhang, D.; Huang, Y. Facial expression recognition via a CBAM embedded network. *Procedia Comput. Sci.* **2020**, *174*, 463–477. [CrossRef]
26. Yan, R.; Li, W.; Chen, Y.; Huang, H.; Wang, W.; Song, Y. A Siamese network-based image matching algorithm. *J. NanJing Univ. Natural Sci.* **2023**, *59*, 770–776.
27. Peng, Z.; Huang, W.; Gu, S.; Xie, L.; Wang, Y.; Jiao, J.; Ye, Q. Conformer: Local features coupling global representations for visual recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 367–376.
28. Starovoytov, V.V.; Eldarova, E.E.; Iskakov, K.T. Comparative analysis of the SSIM index and the Pearson coefficient as a criterion for image similarity. *Eurasian J. Math. Comput. Appl.* **2020**, *8*, 76–90. [CrossRef]
29. Alsaqr, A.M. Remarks on the use of Pearson's and Spearman's correlation coefficients in assessing relationships in ophthalmic data. *Afr. Vis. Eye Health* **2021**, *80*, 10. [CrossRef]
30. Hayale, W.; Negi, P.S.; Mahoor, M.H. Deep siamese neural networks for facial expression recognition in the wild. *IEEE Trans. Affect. Comput.* **2021**, *14*, 1148–1158. [CrossRef]
31. Yu, J.; Xie, G.; Li, M.; Hao, X. Retrieval of family members using siamese neural network. In Proceedings of the 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020), Buenos Aires, Argentina, 16–20 November 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 882–886.
32. Hughes, L.H.; Schmitt, M.; Mou, L.; Wang, Y.; Zhu, X.X. Identifying corresponding patches in SAR and optical images with a pseudo-siamese CNN. *IEEE Geosci. Remote Sens. Lett.* **2018**, *15*, 784–788. [CrossRef]
33. Zhang, Y.; Liu, X.; Liu, H.; Yu, H. MRS-Transformer: Texture Splicing Method to Remove Defects in Solid Wood Board. *Appl. Sci.* **2023**, *13*, 7006. [CrossRef]