mathematics

Article

Local Directional Difference and Relational Descriptor for Texture Classification

Weidan Yan and Yongsheng Dong *

School of Information Engineering, Henan University of Science and Technology, Luoyang 471023, China; 220320040401@stu.haust.edu.cn

* Correspondence: ysdong@haust.edu.cn

Abstract: The local binary pattern (LBP) has been widely used for extracting texture features. However, the LBP and most of its variants tend to focus on pixel units within small neighborhoods, neglecting differences in direction and relationships among different directions. To alleviate this issue, in this paper, we propose a novel local directional difference and relational descriptor (LDDRD) for texture classification. Our proposed LDDRD utilizes information from multiple pixels along the radial direction. Specifically, a directional difference pattern (DDP) is first extracted by performing binary encoding on the differences between the central pixel and multiple neighboring pixels along the radial direction. Furthermore, by taking the central pixel as a reference, we extract the directional relation pattern (DRP) by comparing binary encodings representing different directions. Finally, we fuse the above DDP and DRP to form the LDDRD feature vector. Experimental results on six texture datasets reveal that our proposed LDDRD is effective and outperforms eight representative methods.

Keywords: texture classification; local binary pattern; local feature; multi-direction; noise

MSC: 68T10

check for updates

Citation: Yan, W.; Dong, Y. Local Directional Difference and Relational Descriptor for Texture Classification. *Mathematics* **2024**, *12*, 3432. https://doi.org/10.3390/math12213432

Academic Editors: Vladimir Balan and Iraklis Varlamis

Received: 31 August 2024
Revised: 30 September 2024
Accepted: 30 October 2024
Published: 1 November 2024

1. Introduction

Texture stands out as a key visual element in images, indicative of the basic properties inherent in object surfaces. Texture analysis plays a pivotal role in various fields, including palm-print recognition [1], remote sensing image processing [2,3], and expression recognition [4]. However, since texture features can vary significantly among different objects or even the same object under different conditions, the effective classification of textures is necessary. The local binary pattern (LBP) was proposed by Ojala et al. [5]. Due to its robustness to illumination changes, ease of computation, and efficient feature discrimination, it has garnered significant attention in texture classification tasks and has found widespread use in dynamic texture recognition [6,7], color texture classification [8,9], medical image analysis [10], and image retrieval [11], among other fields.

Since the LBP was proposed, various variants have been developed to enhance the accuracy, rotational invariance, and noise robustness of texture features. To extract richer local features, the completed LBP (CLBP) [12] utilizes local difference signs, magnitudes, and center pixel information to construct a 3D joint histogram for texture representation, effectively capturing texture characteristics. Inspired by the complementary nature of the CLBP operator, the extended LBP (ELBP) [13] extends the LBP by extracting central information, neighborhood differences, and radial differences. In addition, the pairwise rotation-invariant co-occurrence LBP (PRICoLBP) [14] and 2D-LBP [15] incorporate the spatial context information of corresponding LBP groups into texture representation. The partial local gravity feature (PLGF) [16], inspired by the law of gravity, extracts the magnitude and angle features of local gravity to improve recognition rates. The attractive center-symmetric local binary pattern (ACS-LBP) [17] captures both micro and macro texture structures by introducing local and global grayscale information. The adaptive binarized magnitude vector (ABMV) [18] adaptively calculates an average vector threshold to replace the scalar threshold, enabling more precise extraction of magnitude features in different local spaces.

Some variants introduce robust patterns to achieve rotation invariance. For example, Ojala et al. [5] proposed the uniform pattern descriptor, which classifies images based on the number of transitions in binary encodings. Additionally, the completed local binary count (CLBC) [19] considers contrast information by encoding local differences, specifically by calculating the number of '1' bits present in the binary strings. Another class of variants focuses on finding robust permutations. The dominant rotation-invariant LBP (DRLBP) [20] rotates encoding weights based on the local dominant direction. The normalized difference vector (NDV) [21] sorts the binary strings and circularly shifts all orientations, selecting the descriptor with the smallest index as the unique NDV. Furthermore, the local gradient pattern (LGP) [22] employs a global threshold that does not depend on the rotation angle of the image, enabling rotation invariance in related image processing algorithms. Song et al. [23] proposed the locally encoded transformed rotational invariant scale-space texture histogram (LETRIST), which utilizes Gaussian derivative filters to establish rotation-invariant features.

To enhance noise robustness, the local ternary pattern (LTP) [24] extends binary encoding to ternary encoding, enabling a greater reduction in noise interference. The binary rotation-invariant and noise-tolerant (BRINT) descriptor [25] effectively enhances robustness during preprocessing by averaging image pixels. Following this, the median robust extended local binary patterns (MRELBP) [26] incorporates a median filter in the preprocessing stage to boost robustness. SFB-OCPS [27] adopts an adaptive threshold quantization edge-selection strategy to re-screen robust center pixels, while SPALBP [28] utilizes gradient values to obtain adaptive sampling radii in eight directions for robust neighborhood sampling. Both methods achieve noise robustness by converting non-uniform patterns into uniform patterns. Song et al. [29] created the LGONBP descriptor, which encodes neighborhood pixel intensity order and applies a mean filter to integrate local and global information for enhanced noise robustness.

Recently, texture recognition methods based on deep convolutional neural networks (CNNs) have also garnered significant attention. Cimpoi et al. [30] combined the Fisher vector with the deep convolutional activation feature (DeCAF) to describe textures in the wild. DeepTEN [31] integrates dictionary learning and residual encoding into its network layers, constructing an end-to-end framework. The feature aggregation module CLASS [32] models CNN feature maps from different layers and incorporates statistical self-similarity to capture spatial texture features more effectively. Compared to traditionally handcrafted descriptors, deep features demonstrate superior representational capabilities for complex texture images. However, the training process of deep models is time-consuming and demands high computational resources. Moreover, on small-scale datasets with limited data, deep models are prone to overfitting, which restricts their application scope to a certain extent. In contrast, traditional handcrafted descriptors still hold potential in specific scenarios.

The traditional LBP is limited to extracting pixels from a narrow neighborhood within the texture image to form local symbolic structures. Furthermore, the LBP ignores directional amplitude variations and focuses solely on the relationship between one neighborhood sample and the center point, restricting its ability to fully describe local directional texture. Additionally, as the scale increases, the required dimensionality grows exponentially. To address these issues, this paper proposes a novel, highly discriminative texture descriptor called the local directional difference and relational descriptor (LDDRD). Gradient information reflects the direction and intensity of pixel value changes in an image and consists of both radial and tangential components. Extracting both radial and tangential information simultaneously can improve the descriptive power of the descriptor for texture features [33]. Therefore, the LDDRD employs simple arithmetic operations to fuse the grayscale differences between multiple neighboring points and the central point along the local radial direction, thereby generating directional differences. Following this, the CLBP encoding strategy is applied to extract the directional difference pattern (DDP). To further extract directional texture features, the encoded differences from the comparisons between directions are processed to create a directional relation pattern (DRP). Finally, the LDDRD is constructed by combining the DDP and DRP, incorporating central pixel information along with directional differences and inter-directional relationship information. It achieves good performance in texture classification.

Our main contributions are as follows:

- We propose a novel local directional difference and relational descriptor (LDDRD) for texture classification. Our proposed LDDRD captures directional differences and directional relationships.
- We build two complementary operators: a directional difference pattern (DDP) and a directional relation pattern (DRP). The DDP integrates difference information obtained by contrasting multiple neighboring pixels radially with the central pixel, enabling fine-grained encoding of local directional difference information. The DRP compares the direction codes corresponding to different directions to capture directional differences, revealing the relationships between different directions.
- Experimental results on five well-known texture datasets show that our proposed LDDRD is effective and robust against different noise disturbances.

The structure of this paper is as follows. Section 2 provides the foundational background for this paper, including the LBP, CLBP, and NLBP. In Section 3, we introduce the LDDRD in detail. Section 4 summarizes the results of comparing the proposed descriptor with different LBP variants across multiple databases. Lastly, Section 5 provides the concluding remarks of this paper.

2. Related Works

The local binary pattern (LBP) [5] analyzes intensity differences by comparing the central pixel with its surrounding adjacent pixels. To reduce dimensionality, Ojala et al. [5] introduced the uniform pattern "riu" into the LBP, resulting in $\mathrm{LBP}^{\mathrm{riu}}$. $\mathrm{LBP}^{\mathrm{riu}}$ categorizes patterns with no more than two transitions from 0 to 1 or from 1 to 0 in the binary string as uniform patterns, classifying the rest as non-uniform patterns. This reduces the dimensionality of the LBP to P + 2.

To enhance the discrimination capability of the original LBP, Guo et al. [12] supplemented the sign information of the LBP by incorporating complementary magnitude information, denoted as CLBP_M. The central pixel also plays a crucial role in describing features. In the complete LBP (CLBP), the symbolic information is encoded as CLBP_S, and the central pixel is encoded as CLBP_C. By paralleling these three operators, the CLBP can be obtained.

Additionally, in our previous work, we proposed the MCDR [34], JRLP [35], and SMDCP [36] descriptors, all of which are improvements on the LBP. The MCDR captures robust texture features by combining local count vectors, taking into account multi-scale characteristics simultaneously. The JRLP focuses on the differences between adjacent pixels within local regions. Furthermore, the SMDCP, building upon the JRLP, enhances the representation by incorporating the extraction of directional and contour information. To capture global information more comprehensively, Song et al. [29] proposed the NLBP. The NLBP processes the grayscale values of all pixels in an image by arranging them in ascending order. The sorted vector is then uniformly divided to obtain the average grayscale values, which serve as anchor points to capture structural changes in local regions relative to the whole image.

3. Proposed Method

Although the CLBP [12] effectively enhances classification accuracy by integrating local sign and magnitude information, its ability to capture local directional information remains limited during the comparison of selected neighboring pixels. In our work, the MCDR [34] considers expanding the neighborhood range but ignores directional information. In contrast, both the JRLP [35] and SMDCP [36] focus only on the relationships between symmetric pixels along the horizontal and vertical directions. Therefore, we introduce a novel approach named the local direction texture descriptor (LDDRD).

In this section, we provide the details of the proposed texture descriptor, the local directional difference and relational descriptor (LDDRD). First, in Section 3.1, we introduce the directional difference pattern (DDP), which obtains richer directional difference information. Then, in Section 3.2, we introduce the directional relation pattern (DRP), which captures the associations between different directions, thereby complementing the DDP. Finally, we integrate the DDP and DRP to represent the LDDRD in the form of a joint histogram.

3.1. Directional Difference Pattern (DDP)

We propose the directional difference pattern (DDP), which represents local directional differences by capturing the grayscale difference information between multiple pixels and the central pixel in the direction of the image. Since the local binary pattern (LBP) focuses only on the information exchange between a single neighborhood and the central pixel, this limitation renders it insufficient for adequately representing the texture characteristics in a given direction. Moreover, the LBP solely utilizes the sign information obtained from subtracting two pixels, neglecting the magnitude information present in the differences. Furthermore, as the sampling neighborhood increases, the LBP fails to consider the combined effect of multiple pixel differences within the same direction. These limitations hinder its accuracy in processing local directional difference information. Although the SMDCP takes directional information into account, it only extracts the differences in symmetric pixels along the horizontal and vertical directions. Therefore, the DDP supplements directional difference information through the operations outlined below.

First, we consider expanding the selection area of sampling neighborhoods based on the LBP to acquire more neighborhood pixels in various directions. Figure 1 illustrates the sampling structure of the LDDRD. For simplicity, the figure only shows a sampling structure with a window size of 5 × 5. In this structure, we define g_c as the central pixel. Simultaneously, on each circle with a different radius R, the LDDRD employs bilinear interpolation to obtain P neighboring pixels, denoted as $g_{p,n}$ (where P ranges from 0 to 7). Each direction, indicated by the line connecting a neighborhood pixel to the central pixel, is considered an orientation. Compared to the original LBP neighborhood, this design allows for capturing more neighboring pixels in the same direction. This enhancement not only enriches the descriptor but also improves its ability to express image texture features. As a result, the LDDRD is capable of acquiring richer and more detailed directional information.

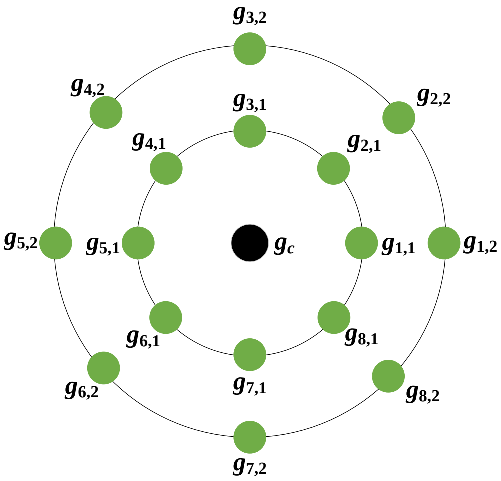

Figure 1. The sampling structure of the DDP (w = 5).

Assuming the size of the DDP sampling window is w × w, we calculate the differences between the neighborhood pixels of different scales along the same radial direction and the central pixel, resulting in N pixel difference values corresponding to each of the P different radial directions. The total difference value is obtained by summing the differences along the same radial direction, and the difference information is decomposed into sign information DDP_S and magnitude information DDP_M according to the CLBP. The sign information of the DDP can be defined as follows:

$$DDP\_S = \sum_{p=0}^{\mathrm{P}-1} s\left(\sum_{n=1}^{\mathrm{N}} (g_{n,p} - g_c)\right) 2^p, s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases}, \tag{1}$$

where $g_{n,p}$ represents the n^{th} neighborhood pixel in the p^{th} direction, g_c corresponds to the central pixel, and s(x) is the binarization function, as shown in Equation (1). The dimension R^P of the original LBP is determined by the radius R and the number of directions for neighboring pixel sampling P. Excessively large values of R and P can lead to a sharp increase in the dimension. Therefore, we fix P = 8 to construct the feature image. When the window size changes, the resulting feature vector remains constant at 2^p compared to other variants. When N = 1, DDP_S only considers the differences between the central pixel and its nearest neighboring pixels in the p^{th} direction. At this point, DDP_S is identical to CLBP_S under the R = 1 setting. Therefore, the DDP inherits the advantage of illumination invariance from the CLBP to a certain extent. The expansion of the sampling neighborhood increases the amount of information in each direction, which can compensate for the lack of macroscopic information in those directions. Similarly, DDP_M is defined as

$$DDP\_M = \sum_{p=0}^{\mathrm{P}-1} s\left(\sum_{n=1}^{N} m_{n,p} - c_p\right) 2^p, \tag{2}$$

where $m_{n,p}$ represents the value of $|g_{n,p} - g_c|$ and $s(x)$ is defined in the same way as in Equation (1). As mentioned above, P = 8 represents the number of directions for sampling neighboring pixels. c_p is the average grayscale of $\sum_{n=1}^{N} m_{n,p}$ in the whole image, which is defined as

$$c_p = \frac{\sum_{i=1}^{K} \sum_{j=1}^{H} \left(\sum_{n=1}^{N} m_{n,p}(i,j)\right)}{K \times H} \tag{3}$$

where $m_{n,p}$ represents the value of $|g_{n,p} - g_c|$ and $K \times H$ is the size of the entire image.

During the amplitude binarization calculation, we add up the absolute values of the differences. This approach avoids the potential loss of critical information due to the cancellation of positive and negative differences. By preserving and emphasizing the absolute values of these differences, we can maximize the integrity of the features in each direction. When N = 1, DDP_M is identical to CLBP_M under the R = 1 setting. Both DDP_S and DDP_M employ the “riu” mode. As N gradually increases (N > 1), the two-dimensional histogram generated by the combination of DDP_S and DDP_M consistently maintains a size of 100 bins, significantly reducing the feature dimensionality of the descriptor. In contrast, methods like the LBP may lead to computational complexity due to their excessively high dimensionality. According to our experiments, the DDP achieves the best classification performance when N = 3. Therefore, in this method, the window size w is set to (2N − 1). Figure 2 provides a specific implementation example of the DDP within a 5 × 5 window.

Next, after encoding the symbolic information and intensity information into binary form, they are concatenated in parallel with the central pixel information to obtain the descriptor. In [29], the NLBP mechanism ingeniously leverages multiple anchors to effectively encode the intensity differences between adjacent sampling points and these anchors, thereby capturing abundant non-local interaction information.

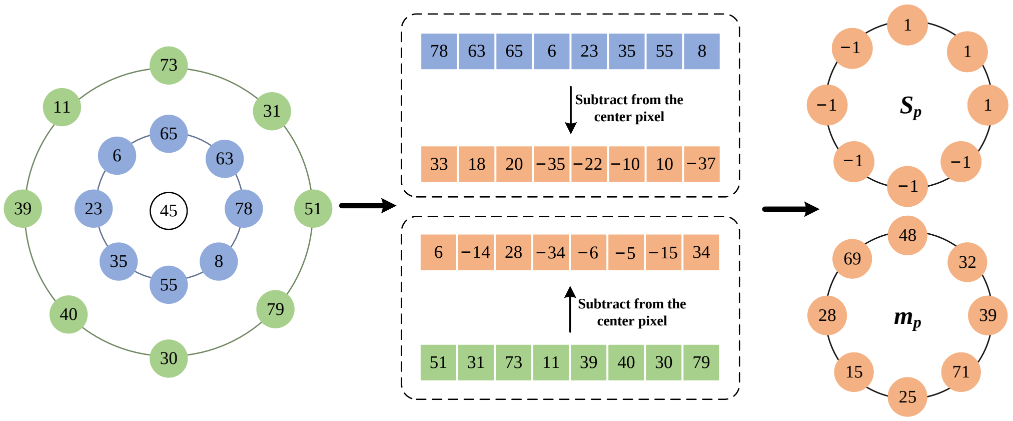

Figure 2. Illustration of an example of the specific implementation of the DDP within a 5 × 5 local neighborhood window.

Drawing upon this, to better delve into the details of the global information C and smoothly capture a broader range of global dependency relationships, we introduce the absolute grayscale information of pixels into DDP_C. Through the distribution of image grayscale, we perform a classification of the central pixels. The central pixels across the entire image are categorized into J distinct levels, each corresponding to a unique threshold. This approach can be regarded as an extension of the CLBP_C descriptor, which segments the image based on multiple thresholds. Consequently, the central pixel information DDP_C can be defined as

$$DDP\_C = \sum_{j=0}^{J-1} s\left(g_c - c_j\right), \tag{4}$$

where J represents the number of thresholds and c_j is the j^{th} threshold. During the specific calculation process, the pixels of the image are first sorted in ascending order of their grayscale values. Afterward, the sorted pixels are divided into J equal parts. Finally, the mean value of the j^{th} partition is calculated to obtain c_j.

DDP_C retains the advantages of the original descriptor while significantly enhancing its ability to capture and describe global information by introducing grayscale levels. Since the thresholds are calculated based on the average pixel grayscale, DDP_C exhibits a certain degree of illumination invariance and noise robustness.

Finally, we adopt the joint histogram of DDP_S, DDP_M, and DDP_C as the directional difference pattern. Unlike the CLBP, we consider extending the range of the sampling neighborhood, which makes it easier for the DDP to capture more favorable directional features. At the same time, by accumulating the absolute values of differences in the same direction, we effectively fuse information between different pixels in that direction, allowing for a more natural handling of textural features in terms of direction. Furthermore, during the extraction process of the local central pixels, we refer to the anchor setting concept in the NLBP, dividing the set of image pixels into several subsets to further refine the extracted directional features. This enhances the sensitivity of features to image details. The DDP inherits the advantages of the CLBP, making it robust against rotation and scale variations in images.

3.2. Directional Relation Pattern (DRP)

The directional relation pattern (DRP) is used to capture the differential relationships between directions. Although the DDP effectively encodes the directional difference information within the local neighborhood surrounding each pixel in an image, it fails to capture the intrinsic connections between these local directions, which limits its discriminative ability to a certain extent. Specifically, when performing difference comparisons, the DDP focuses solely on the differences between multiple pixels within a single direction, without considering the relationships among multi-pixel groups across different directions. Information across different directions can represent the tangential information of gradients to a certain extent. To overcome this limitation and enhance the discriminability and robustness of the DDP descriptor, we propose a complementary descriptor named the DRP. The DRP utilizes pixels within multiple radius neighborhoods to obtain a unique identifier value for each direction. The sign differences of these identifier values are then encoded into binary to capture the relationships between directions. Figure 3, with R = 2 as an example, details the specific implementation steps of the DRP.

First, to quantify the relative relationships among pixels in these directions, we compare the magnitude of each pixel within the neighborhood with that of the central pixel g_c and construct a binary string along the direction, which represents the relative information of that direction. In this example, we select circular neighborhoods with two different radii (R_1 and R_2) to acquire the neighboring pixels $g_{p,1}$ (r = 1) and $g_{p,2}$ (r = 2) in these circular neighborhoods. $g_{p,r}$ represents the r^{th} directional neighboring pixel in the p^{th} direction. For R neighboring pixels in one direction, an R-bit binary code is generated. Here, each bit (0 or 1) directly reflects the grayscale value change relative to the central pixel g_c in a specific direction.

Then, to ensure that each direction can be accurately encoded, we need to integrate the difference symbols within the same direction through binary coding. This combination is based on assigning corresponding weights to elements according to their importance, thereby constructing a code (denoted as $\mathit{\Gamma}$) that represents the local directional relationship. In this process, higher weights are assigned to pixels closer to the central pixel. For instance, $g_{p,1}$ should have a higher weight than $g_{p,2}$. Based on this principle, for the local directional characteristic of pixel $g_{p,r}$ in a specific direction p, we can express its local directional information code $\mathit{\Gamma}$ as follows:

$$\mathit{\Gamma}_p = \sum_{r=1}^{R} s(g_{p,r} - g_c) 2^{R-r}, \tag{5}$$

where 2^{R-r} functions as the weighting mechanism for converting the directional binary string into a unique directional relationship code (represented as a decimal value) and $s(x)$ is defined by Equation (1). Employing a weighted method allows for the unique identification of $\mathit{\Gamma}$ for each direction, ensuring the accuracy of the information.

As illustrated in Figure 3, we calculate the differences between neighborhood pixels and the central pixel under two different radii and then convert this difference information into a 2-bit binary string through binarization. This process is repeated for eight different directions, generating unique binary codes representing each direction. To effectively utilize these directional codes for comparison, we need to design local thresholds to convert them into a single numerical form. Meanwhile, it is important to ensure that this conversion process does not introduce additional dimensions.

Next, when selecting the baseline point for local thresholds, we consider that the central pixel can more intuitively reflect local features than the local average. Based on this, we adopt the central pixel as the reference for the local thresholds to explore the correlations between different directions within the local neighborhood. However, it should be noted that the range of the central pixel values (from 0 to 2^P) often does not directly correspond to the range of the directional relationship encodings (from 0 to 2^R). To address this mismatch, we design a conversion formula. This formula aims to map the central pixel values into a numerical range that is compatible with the directional relationship encoding. The specific conversion formula is as follows:

$$\mu = \left\lfloor g_c \times \frac{2^R - 1}{2^P - 1} \right\rfloor, \tag{6}$$

where $\lfloor x \rfloor$ is the floor function. Through the transformation function, a comparative benchmark for the directional relationship can be obtained. In Figure 3, the result of the conversion for central pixel 45 is 0.

Finally, we calculate the difference between Γ_p and μ to obtain the directional relation pattern (DRP). The DRP is a binary pattern that describes the correlation between directions. This process involves encoding the R neighboring pixels within the local neighborhood across p different directions to generate a representative binary code for each direction. These encoded binary codes are then compared with each other to capture the differences between the directions. Consequently, for a pixel (i, j) in an image, its DRP can be represented as

$$DRP_{i,j} = \sum_{p=0}^{P-1} s(\Gamma_p - \mu)2^p, \tag{7}$$

where $s(x)$ represents the weight function, which is defined in Equation (1). Γ_p and μ are defined by Equations (5) and (6), respectively.

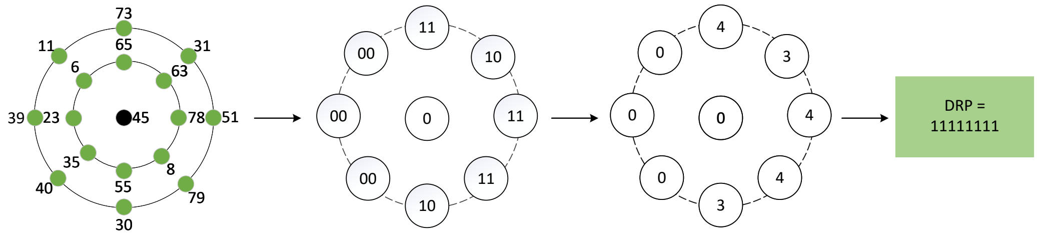

Figure 3. Illustration of an example of the specific implementation of the DRP within a 5 × 5 local neighborhood window.

The distinction between the DRP and DDP lies in the fact that the DRP primarily focuses on capturing the correlation among multiple directions within a local neighborhood. The DRP employs the associative nature of radial pixels to establish clear directional identifiers. These identifiers are compared with reference points and use differential information to describe the relationship between directions, thereby effectively supplementing the description of local directional texture features.

The DDP captures the relationships between adjacent central pixels across multiple scales and different directions, while the DRP gathers the directional relationships between distinct orientations. As a result, these two components display a mutually beneficial relationship, where one complements the other, thereby augmenting their respective capabilities. Ultimately, we concatenate the histograms generated by the DDP and DRP to form a local directional difference and relational descriptor (LDDRD). The overall process is illustrated in Figure 4. LDDRD can be defined as

$$H_{LDDRD} = (H_{DDP}, H_{DRP}), \tag{8}$$

The feature dimension of H_{DDP} is J × 100, where J represents the number of thresholds in DDP_C (J = 3). Meanwhile, the feature dimension of H_{DRP} is 10.

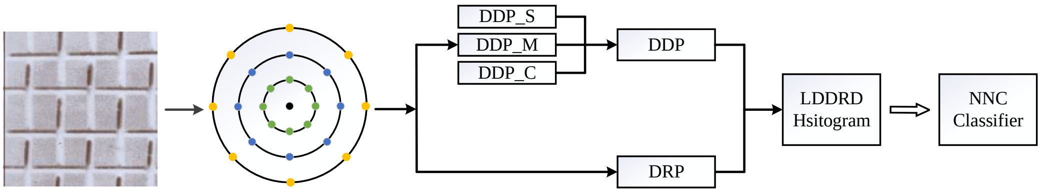

Figure 4. The flowchart of the proposed LDDRD descriptor.

4. Experimental Results

In this section, we conduct a series of experimental validations on seven representative datasets, demonstrating the effectiveness of the local directional difference and relational descriptor (LDDRD) for texture classification.

4.1. Databases and Experimental Setup

We compared the LDDRD with LBP variants, specifically focusing on their performance in texture classification across multiple benchmark texture datasets, including Brodatz [37], KTH-TIPS [38], UIUC [39], CUReT [40], and Outex [5]. The methods used for comparison were LBP^{riu} [5], CLBP [12], CLBC [19], MCDR [34], JRLP [35], CRDP [33], LGONBP [29], SMDCP [36], and ABMV [18].

4.1.1. Brodatz Dataset

The Brodatz dataset [37] comprises 112 types of texture images, all of which are derived from real-world texture samples. We chose the grayscale version of Brodatz, where each grayscale image has a size of 640 × 640 pixels. For the purpose of experimentation, each image in the dataset is split into 25 smaller images, each of which has a size of 128 × 128 pixels. Consequently, the dataset comprises 112 texture classes, with 25 images per class, resulting in a total of 2800 images. We selected 11 images per class (N = 11) for training and retained 14 images for testing.

4.1.2. KTH-TIPS Dataset

The KTH-TIPS dataset [38] encompasses 10 diverse texture categories, each containing 81 images. The images reflect realistic textures with nine different scales, three distinct poses, and nine types of lighting conditions. For training purposes, we selected 40 images from each category while reserving the remaining images for testing.

4.1.3. UIUC Dataset

The UIUC dataset [39] contains texture images from 24 different species, with each category comprising 32 images. The challenge of this database lies in the fact that each category contains images with viewpoint changes while also exhibiting substantial differences between categories. Each image has a resolution of 640 × 480 pixels. From each category, half of the images were selected as training images, and the other half were designated as test images.

4.1.4. CUReT Dataset

The CUReT dataset [40] comprises 61 distinct categories and is commonly used for detecting and recognizing the surface materials of objects. Each category contains 92 images of the same scale but with different lighting conditions. The size of the images is 200 × 200 pixels. Specifically, we used 46 × 61 images for training in this paper, with the rest serving as the test set.

4.1.5. Outex Dataset

The Outex dataset [5] contains a total of 17 sub-datasets. We used Outex_TC11 (TC11), Outex_TC12 (TC12), and Outex_TC10 (TC10) for our experiments. They all contain 24 texture categories, with each category represented by three types of illumination ("Inca", "tl84", and "horizon") and nine different rotation angles (0°, 5°, 10°, 15°, 30°, 45°, 60°, 75°, and 90°). TC11 includes images in 100 dpi and 120 dpi resolutions. When constructing the training and testing sets, we adhered to the original recommended segmentation of the Outex dataset. Specifically, we utilized 24 × 20 images under the "Inca" illumination and a 0° angle for training. For the Outex_TC10 dataset, we selected the images from the remaining eight angles for testing. For the Outex_TC12 dataset, all samples captured under different lighting conditions were included in the test set.

For the CUReT, KTH-TIPS, UIUC, and Brodatz datasets, there were no fixed training and testing sets. We conducted 10 experiments on each dataset and calculated the average accuracy as the comparison metric. During image preprocessing, each image underwent normalization with a mean of 0 and a standard deviation of 1. Subsequently, in the noise robustness testing, we added different types of noise to the images. As the density of noise increased, it indicated a higher degree of noise pollution. In our experiment, we set the number of neighborhood orientations P to 8. The parameter configurations of the remaining LBP variants all followed the settings described in their respective papers.

The classifier selects the nearest neighbor classifier (NNC) to categorize texture images. The NNC is the K-nearest neighbor (KNN) classifier when K = 1. All methods employed the NNC classifier in this paper. The NNC uses the cardinality distance to measure the similarity between the texture histograms, specifically comparing those of the test set against those of the training set. We selected the class with the smallest distance to the test image as its classification result. The definition formula is as follows:

$$D_{\chi^2}(H_1, H_2) = \sum\nolimits_K \frac{[H_1(k) - H_2(k)]^2}{H_1(k) + H_2(k)}, \tag{9}$$

where H_1 and H_2 represent the feature histograms of the test images and the training images, and $H_1(k)$ and $H_2(k)$ represent the k^{th} level index of histograms H_1 and H_2.

4.2. Parameter Evaluation

The LDDRD is composed of the DDP and DRP, and both the window size w of the DDP and the scale size r of the DRP have an impact on classification accuracy. To study the effects of w and r, we conducted experiments on the OutexTC10 dataset. For example, in one of our experiments, the local image block size was set to 3 × 3, which corresponds to a window size of w = 3 for the DDP. Figure 5 illustrates the effects of these specific parameter settings on the classification performance of the LDDRD.

It can be observed in Figure 5 that, under noise-free conditions, the classification performance of the LDDRD improved as the values of w and r increased. When w = 7 (assuming a 7 × 7 window size for the DDP), the DDP achieved a relatively high classification accuracy and integrated well with the DRP. Specifically, when w equals 7, the accuracy of the LDDRD was generally higher than that of other w values. For instance, when r was fixed at 1, the accuracies of the LDDRD were 98.30%, 99.06%, 99.24%, and 98.95%, respectively (with w ranging from 3 to 9). When the value of w was set excessively high, the classification performance of the LDDRD showed a declining trend. When w was increased to 9, the overall performance of the LDDRD decreased notably by 0.54% compared to when w = 7. This indicates that there exists an optimal value for w, and after reaching this peak, the classification effectiveness of the LDDRD diminishes. Within the range of $r \in [4, 6]$, the DRP achieved better classification results. Specifically, when $r = 6$, the combination with the DDP achieved the best complementarity, reaching an accuracy of 99.8% on OutexTC10. Therefore, the optimal window size of the DDP was set to 7 × 7, and the scale of the DRP was set to 6.

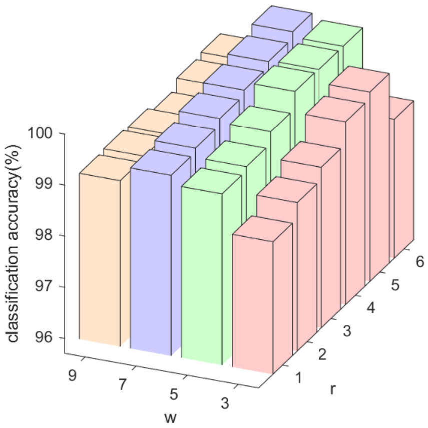

Figure 5. Effects of the DDP sampling window size (w) and the DRP scale size (r) on the classification performance of the LDDRD.

4.3. Classification Performance Comparison

In this subsection, we validate the effectiveness of the LDDRD by comparing it with nine classification methods for texture classification, utilizing six benchmark datasets for a comprehensive evaluation. Among them, the scale (R,P) for LBP^{riu}, CLBP, CLBC, MCDR, JRLP, SMDCP, and ABMV was set to (3,24). For LGONBP, the scale was set to (3,24)+(5,24)+(7,24), and for CRDP, the scale was set to (1,8)+(3,8)+(5,8).

Table 1 presents the classification accuracy in detail, from which the following observations can be made. The proposed LDDRD achieved accuracies of 96.8%, 95.3%, 98.8%, 98.3%, 99.8%, 95.6%, and 99.6%, respectively. Additionally, the standard deviations of the LDDRD on the CUReT, KTH-TIPS, UIUC, and Brodatz datasets were 0.87, 0.96, 0.50, and 0.16, respectively. Compared to the CLBP, the LDDRD achieved classification accuracy improvements of 2.93%, 2.6%, 0.71%, 0.71%, 0.86%, 0.31%, and 2.08% across various datasets. This is primarily attributed to the fact that the LDDRD enhances its ability to capture contrast difference information by extending the sampling range, allowing for the effective acquisition of local directional information. In contrast to the JRLP and SMDCP, the LDDRD demonstrated a more effective capture of texture features through the extraction of information from eight distinct directions. Although the LDDRD holds some advantages over certain LBP variants, CRDP demonstrated superior performance. CRDP, through its multi-scale fusion strategy, particularly the scale combination of (1,8) + (3,8) + (5,8) in conjunction with the support vector machine (SVM) classifier, demonstrated superior performance. In contrast, the LDDRD, utilizing the scale of (3,8), fell short of CRDP in terms of performance across multiple datasets. Notably, when employing the same neighborhood size, CRDP achieved a result of 98.52% on the TC10 dataset. The accuracy of the LDDRD was 99.79%, which was 1.27% higher than that of CRDP. This indicates that the LDDRD is more effective under this condition. Furthermore, the methods used for comparison with the LDDRD primarily consisted of variants of the LBP approach. However, when compared to SPR [41], a texture method based on random projection, SPR demonstrated higher results, achieving an accuracy of 99.05% on the CURET dataset.

Table 1. Classification accuracy (%) of different classification algorithms on eight datasets.

Method	**Texture Databases**						
	KTH-TIPS	**UIUC**	**Brodatz**	**CUReT**	**TC10**	**TC12**	**TC11**
LBP^{riu} [5]	95.07	73.18	90.36	87.45	95.07	85.05	77.71
CLBP [12]	93.90	92.71	98.04	97.58	98.93	95.32	97.5
CLBC [19]	94.63	93.23	97.86	96.89	98.77	94.01	98.54
MCDR [34]	93.17	85.42	87.86	87.13	98.85	95.06	76.67
JRLP [35]	95.61	91.41	95.18	94.90	96.15	92.96	88.54
CRDP [33]	99.37	—	—	99.46	99.92	98.82	—
LGONBP [29]	94.87	90.10	92.14	96.86	99.60	97.89	98.33
SMDCP [36]	90.97	89.06	97.32	96.40	98.33	92.12	98.34
ABMV [18]	95.12	94.79	98.55	96.46	99.42	96.20	97.92
LDDRD	96.83	95.31	98.75	98.29	99.79	95.63	99.58

Figure 6 displays the average classification accuracy when the number of training image samples varied across different datasets. For each class within the dataset, N images were randomly selected as training samples, while the remaining images served as test samples. It can be observed in Figure 6 that as the number of training samples decreased, the classification accuracy of the LDDRD generally declined. This is because fewer training samples usually lead to a decrease in the generalization ability of the descriptor. With a training sample size of N = 20, the LDDRD demonstrated the best performance, achieving classification accuracies of 99.79%, 94.62%, 98.18%, and 94.63% on the respective datasets. Additionally, on the TC_10 dataset, the accuracy did not notably decrease as N decreased, with the drop being maintained at 1.33%. This indicates that the LDDRD possesses a degree of flexibility, allowing it to adjust the value of N based on actual conditions. This flexibility enables the LDDRD to reduce computational time without excessively compromising classification accuracy.

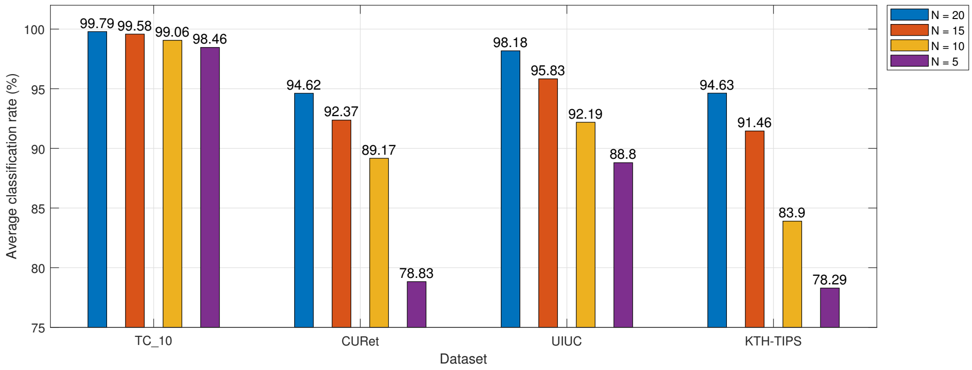

Figure 6. Classification accuracy of LDDRD under different numbers of training samples.

Table 2 shows the running time and feature dimensionality of the LDDRD. We compared the total runtime on the Outex_TC10 dataset under the operational conditions of MATLAB (R2019b) configured with an Intel Core i5-10400 CPU and GTX 1050 Ti. Compared to other texture classification methods, the LDDRD ran faster than LGONBP, while LBP^{riu} had the shortest running time. Overall, the LDDRD maintained a moderate running time while achieving high accuracy. The number of features extracted by the LDDRD was 410. Table 2 shows that LBP^{riu} had the lowest dimensionality, while LGONBP had the highest. Therefore, the LDDRD was able to capture texture with fewer feature dimensions, resulting in a relatively lower computational cost. Furthermore, when the neighborhood size was increased, the LDDRD maintained a constant number of feature dimensions, which did not increase with the increase in r.

Table 2. The running time (s) and feature dimensions of different classification algorithms on the Outex10 dataset.

Method	Time	Size
LBP^{riu} [5]	11.6	10
CLBP [12]	32.8	200
CLBC [19]	23.3	162
MCDR [34]	246.9	462
JRLP [35]	94.6	245
LGONBP [29]	982.1	1404
SMDCP [36]	65.5	243
ABMV [18]	185.8	200
LDDRD	426.5	410

4.4. Robustness Experiment Results

In this subsection, we aim to showcase the robustness of the LDDRD against different types of noise. First, we added noise to the Brodatz benchmark database. Then, we compared the LDDRD with seven representative and robust texture descriptors. Table 3 shows the classification results of the LDDRD on the Brodatz dataset under different levels of salt-and-pepper noise.

It is clear from Table 3 that the classification performance of the LDDRD fluctuated relatively little under different levels of salt-and-pepper noise, with the best performance occurring at a noise level of 0.01. The classification accuracies of the LDDRD under different salt-and-pepper noise levels were 98.75%, 98.57%, and 97.21%, respectively. Even when the noise level reached 0.1, its accuracy remained above 97%, which was higher than that of most LBP variants. The SMDCP was more robust against noise levels of 0.1 due to its utilization of texture contour and directional information. As the noise intensity increased, the accuracy of the LDDRD decreased by only 1.25%, while the decreases for the LBP and CLBP were 4.11% and 1.35%, respectively. This indicates that, overall, the LDDRD exhibited robustness to salt-and-pepper noise.

Table 3. Classification accuracy (%) of various methods under different salt-and-pepper noise levels on the Brodatz dataset.

Method	Salt-and-Pepper Noise		
	0.01	0.05	0.1
LBP^{riu} [5]	94.82	92.67	90.71
CLBP [12]	98.39	98.10	97.04
CLBC [19]	97.86	98.10	97.38
MCDR [34]	94.64	92.32	93.92
JRLP [35]	97.31	97.61	96.91
LGONBP [29]	93.03	93.57	94.10
SMDCP [36]	97.14	98.03	97.67
ABMV [18]	98.04	97.75	97.47
LDDRD	98.75	98.57	97.32

Second, Table 4 shows the classification results under different levels of Gaussian noise added to the Brodatz benchmark database. As the level of Gaussian noise increased (from 0.01 to 0.1), the classification performance of all methods decreased. It can be observed in Table 4 that the LDDRD outperformed both the CLBP and SMDCP in terms of classification accuracy under varying levels of Gaussian noise. Specifically, compared to the CLBP, the accuracy of the LDDRD improved by 0.71%, 5.54%, and 5.37% under different noise levels. Similarly, compared to the SMDCP, the improvements were 2.50%, 0.71%, and 1.78%, respectively. This indicates that the proposed LDDRD performed better than other LBP variants under different levels of Gaussian noise. Furthermore, we compared the magnitude of performance degradation under varying levels of Gaussian noise. As the noise level increased from 0.01 to 0.1, the performance decline for the LDDRD was recorded at 7.14%, while other LBP variants, excluding LGONBP, generally suffered decreases of approximately 10%. To a certain extent, this reflects the robustness of the LDDRD in resisting Gaussian noise.

Table 4. Classification accuracy (%) of various methods under different Gaussian noise levels on the Brodatz dataset.

Method	Gaussian Noise		
	0.01	0.05	0.1
LBP^{riu} [5]	86.07	82.86	76.25
CLBP [12]	98.57	90.53	86.77
CLBC [19]	98.57	94.82	89.82
MCDR [34]	86.96	83.04	74.82
JRLP [35]	91.61	89.46	86.25
LGONBP [29]	90.18	89.82	88.93
SMDCP [36]	96.78	95.36	90.36
ABMV [18]	96.25	92.50	87.32
LDDRD	99.28	96.07	92.14

5. Conclusions

In this paper, we propose a local directional difference and relational descriptor (LDDRD) for texture classification. Our proposed LDDRD is built using a directional difference pattern (DDP) and a directional relation pattern (DRP) in the form of histograms to capture different directional texture features of local neighborhood pixels and central pixels. Our DDP integrates multiple differences in the same direction to encode differential information in different directions. However, our DRP is introduced to capture the relative relationships between different directions, complementing the DDP and enhancing its discrimination ability. Our proposed LDDRD requires no training and exhibits a certain degree of invariance to image rotation and illumination changes. Experimental results show that the LDDRD performs well on popular benchmark texture datasets. It outperforms eight other texture classification methods under various levels of salt-and-pepper noise and Gaussian noise, demonstrating robustness to noise.

Author Contributions: Conceptualization, W.Y. and Y.D.; Methodology, W.Y. and Y.D.; Formal analysis, W.Y. and Y.D.; Investigation, W.Y. and Y.D.; Writing—original draft, W.Y.; Writing—review & editing, Y.D.; Visualization, W.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Natural Science Foundation of Henan (grant no. 232300421023).

Data Availability Statement: No new data were created or analyzed in this study.

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

References

1. Zhao, S.; Zhang, B. Learning complete and discriminative direction pattern for robust palmprint recognition. *IEEE Trans. Image Process.* **2020**, *30*, 1001–1014. [CrossRef] [PubMed]
2. Akiva, P.; Purri, M.; Leotta, M. Self-supervised material and texture representation learning for remote sensing tasks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 8203–8215. [CrossRef]
3. Zhang, X.; Shen, J.; Hu, H.; Yang, H. A new instance segmentation model for high-resolution remote sensing images based on edge processing. *Mathematics* **2024**, *12*, 2905. [CrossRef]
4. Li, X.; Yang, W. Illumination removal via gaussian fifference L0 norm model for facial experssion recognition. *Mathematics* **2023**, *11*, 2667. [CrossRef]
5. Ojala, T.; Pietikainen, M.; Maenpaa, T. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 971–987. [CrossRef]
6. Zhao, G.; Pietikainen, M. Dynamic texture recognition using local binary patterns with an application to facial expressions. *IEEE Trans. Pattern Anal. Mach. Intell.* **2007**, *29*, 915–928. [CrossRef]
7. Zhao, X.; Lin, Y.; Liu, L.; Heikkilä, J.; Zheng, W. Dynamic texture classification using unsupervised 3D filter learning and local binary encoding. *IEEE Trans. Multimed.* **2019**, *21*, 1694–1708. [CrossRef]
8. Dong, Y.; Jin, M.; Li, X.; Ma, J.; Liu, Z.; Wang, L.; Zheng, L. Compact interchannel sampling difference descriptor for color texture classification. *IEEE Trans. Circuits Syst. Video Technol.* **2020**, *31*, 1684–1696. [CrossRef]
9. Song, T.; Feng, J.; Wang, S.; Xie, Y. Spatially weighted order binary pattern for color texture classification. *Expert Syst. Appl.* **2020**, *147*, 113167. [CrossRef]
10. Dubey, S.R.; Singh, S.K.; Singh, R.K. Local wavelet pattern: A new feature descriptor for image retrieval in medical CT databases. *IEEE Trans. Image Process.* **2015**, *24*, 5892–5903. [CrossRef]
11. Dubey, S.R.; Singh, S.K.; Singh, R.K. Multichannel decoded local binary patterns for content-based image retrieval. *IEEE Trans. Image Process.* **2016**, *25*, 4018–4032. [CrossRef]
12. Guo, Z.; Zhang, L.; Zhang, D. A completed modeling of local binary pattern operator for texture classification. *IEEE Trans. Image Process.* **2010**, *19*, 1657–1663. [CrossRef] [PubMed]
13. Liu, L.; Zhao, L.; Long, Y.; Kuang, G.; Fieguth, P. Extended local binary patterns for texture classification. *Image Vis. Comput.* **2012**, *30*, 86–99. [CrossRef]
14. Qi, X.; Xiao, R.; Li, C.G.; Qiao, Y.; Guo, J.; Tang, X. Pairwise rotation invariant co-occurrence local binary pattern. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *36*, 2199–2213. [CrossRef] [PubMed]
15. Xiao, B.; Wang, K.; Bi, X.; Li, W.; Han, J. 2D-LBP: An enhanced local binary feature for texture image classification. *IEEE Trans. Circuits Syst. Video Technol.* **2018**, *29*, 2796–2808. [CrossRef]
16. Bhattacharjee, D.; Roy, H. Pattern of local gravitational force (PLGF): A novel local image descriptor. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *43*, 595–607. [CrossRef] [PubMed]
17. Ruichek, Y.; El merabet, Y.; El idrissi, A. Attractive-and-repulsive center-symmetric local binary patterns for texture classification. *Eng. Appl. Artif. Intell.* **2019**, *78*, 158–172. [CrossRef]
18. Hu, S.; Pan, Z.; Dong, J.; Ren, X. A novel adaptively binarizing magnitude vector method in local binary pattern based framework for texture classification. *IEEE Signal Process. Lett.* **2022**, *29*, 852–856. [CrossRef]
19. Zhao, Y.; Huang, D.S.; Jia, W. Completed local binary count for rotation invariant texture classification. *IEEE Trans. Image Process.* **2012**, *21*, 4492–4497. [CrossRef]

20. Mehta, R.; Egiazarian, K. Dominant rotated local binary patterns (DRLBP) for texture classification. *Pattern Recognit. Lett.* **2016**, *71*, 16–22. [CrossRef]
21. Zhang, W.; Zhang, W.; Liu, K.; Gu, J. A feature descriptor based on local normalized difference for real-world texture classification. *IEEE Trans. Multimed.* **2017**, *20*, 880–888. [CrossRef]
22. Song, T.; Xin, L.; Gao, C.; Zhang, G.; Zhang, T. Grayscale-inversion and rotation invariant texture description using sorted local gradient pattern. *IEEE Signal Process. Lett.* **2018**, *25*, 625–629. [CrossRef]
23. Song, T.; Li, H.; Meng, F.; Wu, Q.; Cai, J. LETRIST: Locally encoded transform feature histogram for rotation-invariant texture classification. *IEEE Trans. Circuits Syst. Video Technol.* **2017**, *28*, 1565–1579. [CrossRef]
24. Tan, X.; Triggs, B. Enhanced local texture feature sets for face recognition under difficult lighting conditions. *IEEE Trans. Image Process.* **2010**, *19*, 1635–1650. [CrossRef] [PubMed]
25. Liu, L.; Long, Y.; Fieguth, P.W.; Lao, S.; Zhao, G. BRINT: Binary rotation invariant and noise tolerant texture classification. *IEEE Trans. Image Process.* **2014**, *23*, 3071–3084. [CrossRef]
26. Liu, L.; Lao, S.; Fieguth, P.W.; Guo, Y.; Wang, X.; Pietikäinen, M. Median robust extended local binary pattern for texture classification. *IEEE Trans. Image Process.* **2016**, *25*, 1368–1381. [CrossRef]
27. Lan, S.; Fan, H.; Hu, S.; Ren, X.; Liao, X.; Pan, Z. An edge-located uniform pattern recovery mechanism using statistical feature-based optimal center pixel selection strategy for local binary pattern. *Expert Syst. Appl.* **2023**, *221*, 119763. [CrossRef]
28. Hu, S.; Li, J.; Fan, H.; Lan, S.; Pan, Z. Scale and pattern adaptive local binary pattern for texture classification. *Expert Syst. Appl.* **2024**, *240*, 122403. [CrossRef]
29. Song, T.; Feng, J.; Luo, L.; Gao, C.; Li, H. Robust texture description using local grouped order pattern and non-local binary pattern. *IEEE Trans. Circuits Syst. Video Technol.* **2020**, *31*, 189–202. [CrossRef]
30. Cimpoi, M.; Maji, S.; Kokkinos, I.; Mohamed, S.; Vedaldi, A. Describing textures in the wild. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 3606–3613. [CrossRef]
31. Zhang, H.; Xue, J.; Dana, K. DeepTEN: Texture encoding network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 708–717. [CrossRef]
32. Chen, Z.; Li, F.; Quan, Y.; Xu, Y.; Ji, H. Deep texture recognition via exploiting cross-layer statistical self-similarity. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 5231–5240.
33. Wang, K.; Bichot, C.E.; Li, Y.; Li, B. Local binary circumferential and radial derivative pattern for texture classification. *Pattern Recognit.* **2017**, *67*, 213–229. [CrossRef]
34. Dong, Y.; Feng, J.; Yang, C.; Wang, X.; Zheng, L.; Pu, J. Multi-scale counting and difference representation for texture classification. *Vis. Comput.* **2018**, *34*, 1315–1324. [CrossRef]
35. Wang, T.; Dong, Y.; Yang, C.; Wang, L.; Liang, L.; Zheng, L.; Pu, J. Jumping and refined local pattern for texture classification. *IEEE Access* **2018**, *6*, 64416–64426. [CrossRef]
36. Dong, Y.; Zheng, B.; Liu, H.; Zhang, Z.; Fu, Z. Symmetric mean and directional contour pattern for texture classification. *Electron. Lett.* **2021**, *57*, 918–920. [CrossRef]
37. Brodatz, P. *Textures: A Photographic Album for Artists and Designers*; Dover Publications: New York, NY, USA, 1966; Volume 66.
38. Hayman, E.; Caputo, B.; Fritz, M.; Eklundh, J.O. On the significance of real-world conditions for material classification. In Proceedings of the European Conference on Computer Vision, Prague, Czech Republic, 11–14 May 2004; pp. 253–266. [CrossRef]
39. Lazebnik, S.; Schmid, C.; Ponce, J. A sparse texture representation using local affine regions. *IEEE Trans. Pattern Anal. Mach. Intell.* **2005**, *27*, 1265–1278. [CrossRef] [PubMed]
40. Varma, M.; Zisserman, A. A statistical approach to material classification using image patch exemplars. *IEEE Trans. Pattern Anal. Mach. Intell.* **2008**, *31*, 2032–2047. [CrossRef]
41. Liu, L.; Fieguth, P.; Clausi, D.; Kuang, G. Sorted random projections for robust rotation-invariant texture classification. *Pattern Recognit.* **2012**, *45*, 2405–2418. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.