Received: 14 July 2020 | Revised: 6 March 2021 | Accepted: 18 March 2021

IET Image Processing

DOI: 10.1049/ipr2.12220

The Institution of Engineering and Technology WILEY

ORIGINAL RESEARCH PAPER

Graph wavelet transform for image texture classification

Yu-Long Qiao[1] | Yue Zhao[1] | Chun-Yan Song[2] | Kai-Ge Zhang[3] | Xue-Zhi Xiang[1]

[1] College of Information and Communications Engineering, Harbin Engineering University, Harbin, Heilongjiang, PR China

[2] College of Mechanical and Electrical Engineering, Northeast Forestry University, Harbin, Heilongjiang, PR China

[3] Department of Computer Science, Utah State University, Logan, Utah, USA

Correspondence
Yue Zhao, College of Information and Communications Engineering, Harbin Engineering University, 145 Nantong Street, Nangang District, Harbin, Heilongjiang, PR China.
Email: zhaoyue1994@hrbeu.edu.cn

Funding information
National Natural Science Foundation of China, Grant/Award Number: 61871142

Abstract

Graph is a data structure that can represent complex relationships among data. Graph signal processing, unlike traditional signal processing, explicitly considers the structure and relationship among the signal samples. Graph wavelet transform can provide a multiscale analysis for the graph signal. It is well known that texture is a region property in an image, which is characterized with the intensity and relationship among pixels. In this context of the graph signal processing framework, an image texture can be considered as the signal on the graph. Therefore, a texture classification method based on graph wavelet transform is proposed. Specifically, image textures are decomposed into multiscale components by using two-channel graph wavelet filter banks. Then the local singular value decomposition is applied to each subband. In order to improve the noise-resistant ability, the maximum, mean and median values of the local singular values of graph-wavelet transformation coefficients are extracted. Finally, the Weibull distributions are used to model those extracted values to describe the image textures. The experiments on the benchmark texture datasets are conducted to demonstrate the effectiveness of the proposed method.

1 | INTRODUCTION

Texture is an important property of the visual scene and provides a power cue for identification and categorization of different objects and natural scenes in the human visual system [1]. In the field of image processing and computer vision, texture classification refers to as assigning an unknown texture sample to one of a set of predefined texture categories, which has attracted a lot of attention [2]. It has been applied in many fields, such as medical diagnosis [3,4], forest species recognition [5] and face recognition [6].

The core issue of texture classification is to describe the image texture with effective features. Tuceryan and Jain [1] classified the various texture feature extraction methods before 1993 into five categories: statistical methods, geometric methods, structural methods, mathematical model methods and signal processing methods [7].

The wavelet transform [8,9], as a classical signal processing technique, possesses better time-frequency localization and multiresolution analysis properties. Thus, various wavelet transform based methods have been proved for the image texture description. Mallat [8] firstly applied the discrete wavelet transform (DWT) in the texture feature extraction. Because DWT is a translation-variant transform, Unser [10] proposed the wavelet frame transform based texture classification. Qiao et al. [11] made use of the complex wavelet frame transform, and combined the magnitude and phase of the complex coefficients into a real measure to characterize the texture. Due to the approximately shift invariance and good directional selectivity of the dual-tree complex wavelet transform (DT-CWT), Celik and Tjahjadi [12] suggested that the statistical measures of magnitude and phase of the DT-CWT coefficients are better than the DWT based method. Ji et al. [13] applied the multi-fractal spectrum (MFS) analysis on wavelet coefficients to extract robust texture descriptors for texture classification. Wang et al. [14] extracted the texture feature based on the fractal dimensions and barycentric coordinates of the bit planes of the wavelet coefficients in both the detail wavelet subbands and approximate wavelet subbands. Yang et al. [15] obtained the image pyramid by applying dual-tree complex wavelet transform (DTCWT) on the original image, and then generated the local binary patterns (LBP) in DTCWT domain to extract features. In recent years, Rieze wavelets [16] and steerable circular harmonic wavelets [17] have shown great potential in texture classification.

In addition, researchers have exploited the traditional texture features, Gray Level Co-occurrence Matrix(GLCM) [18,19], Madras Rubber Factory model [20] and Local Binary Patterns (LBP) method [21] in the wavelet domain.

Graph signal processing is one of the emerging fields in signal processing. In past years, Crovella and Kolaczyk [22] designed wavelet-like functions on graphs which are localized in space and time. In 2011, Hammond et al. [23] constructed a class of wavelet operators in the graph spectral domain, which leads to the research on applying the concepts of filter banks from classical signal processing to graph signals. Narang and Ortega [24] constructed critically sampled wavelet filter banks on graph for analyzing graph-signals defined on any arbitrary finite weighted graph. Then the biorthogonal wavelet filter banks on graph [25] have been introduced. Tanaka and Sakiyama [26] proposed M-channel oversampled filter banks for graph signals. Jiang et al. [27] designed biorthogonal graph filter banks with the lifting-based method. Teke et al. [28] extended the classical multirate signal processing theory to graph signals. Nicolas and Pierre [29] leveraged the graph wavelet to complete the multiscale detection of communities in networks. In recent years, graph wavelets are combined with the convolutional neural network (CNN), and the graph wavelet neural network is proposed in the works [30,31].

Texture is a region property in an image, which is characterized with the intensities and relationships among pixels. Therefore, the texture features, that describe the relationships among the pixel intensities have been demonstrated power discriminative performance in texture classification, such as GLCM, Markov random field, LBP, some variants of the LBP [32], contourlet transform [33,34] and wavelet transforms. Although the traditional wavelet transforms have been successfully applied in texture feature extraction, it only implicitly considers the local structure of a texture by using the filter banks. Unlike traditional signal processing, graph signal processing explicitly considers the structure and relationship among the signal samples. In this context, an image texture can be considered to be the graph signal with very special structures. The graph wavelet transform (wavelet transform on graph) combines the advantages of both the wavelet transform and graph representation of the signal. It can be reasonably applied to describe the image texture. Therefore, in this paper, we extract the image texture feature in the graph wavelet transform domain.

It is well known that the singular value decomposition (SVD that is related with Principal Component Analysis) is a matrix decomposition method, in which larger singular values point to the major information of the matrix. The truncation of lower singular values also results in very good approximation of the original matrix, which can be considered as extracting the major information or denoising. Thus, the SVD (or PCA) based texture features are designed to be tolerance to image blur and noise corruption [35]. Ramakrishnan et al. [36] proposed a new approach based on multiwavelets transformation and singular value decomposition, in which the energy, entropy, local homogeneity and max–min ratio are extracted from the selected singular values as image texture features. Selvan et al. [37] modeled the singular value distribution of wavelet transform coefficients with an exponential function. The estimated six model parameters are taken as image texture features. However, Kim and So [38] demonstrated that it was hardly expected that a six-dimensional feature vector would yield a state-of-the-art performance, and then they proposed a modified texture feature based Haar wavelet transform and SVD.

For a texture image, the singular value decomposition of the whole coefficient matrix of a wavelet subband only captures the major information of that subband. It is difficult to characterize the local property of the texture. Meanwhile, the exponential distribution model parameters of those singular values hardly describe a texture. Therefore, in this paper, we divide the graph wavelet subband into overlapping regions and apply SVD in each region. Then the maximum values, mean values and median values of each subband are modeled with Weibull distributions, respectively. The model parameters are accumulated into the feature vector for texture classification.

The remaining structure of this paper is as follows. Section 2 mainly introduces the graph signal processing and graph wavelet filter banks. The scheme of this paper including graph wavelet transform for image textures, local singular value decomposition and Weibull distribution model of wavelet coefficients are proposed in Section 3. In Section 4, we conduct the texture classification experiments and demonstrate the performance of the proposed method. Finally, our work is concluded in Section 5.

2 | GRAPH WAVELET TRANSFORM

A graph represents a set of elements and a set of pairwise relationships between those elements, in which the element is called vertex, and the relationship is called edge. Formally, a graph G is defined as $G = (V, E)$, where V represents the set of vertices, and the E represents the set of edges. In this paper, we only consider the undirected graph without self-loops. The connectivity of a graph can be represented by an adjacency matrix A, which is defined as

$$A(i,j) = \begin{cases} w_{i,j}, & \text{vertices } i \text{ and } j \text{ are connected} \\ 0, & \text{otherwise} \end{cases}, \quad (1)$$

$w_{i,j}$ is the weight that describes the relationship between vertices i and j. A graph signal $f: V \to R$ is defined on the graph, which is indexed by the vertices and may be represented by a vector. When the image is represented as a graph signal, the grey values of pixels and the relationships between different pixels can be considered.

The degree matrix D is a diagonal matrix, whose the i^{th} diagonal element is d_i that represents the degree of vertex i, whose value is the sum of the weight of all edges connected to node i. The unnormalized graph Laplacian matrix is defined as $L = D - A$. The symmetric normalized graph Laplacian matrix is $\mathbb{L} = I - D^{-1/2}AD^{-1/2}$, where I is the identity matrix. It is a symmetric positive semi-definite matrix and has a set of real eigenvalues $0 \le \lambda_1 \le \lambda_2 \le \cdots \cdots \le \lambda_N \le 2$. Recently, researchers build a graph signal processing framework from the graph spectral theory, such as graph Fourier transform, graph

filtering and graph wavelet filter banks, which provides potential transforms for image texture analysis.

2.1 | Graph filtering

The spectral representation of the graph signal is related to the graph Laplacian matrix L or the normalized version $\mathbb{L}$, which is a symmetric positive semi-definite matrix. We follow the work [25,26], the matrix L (or $\mathbb{L}$) can be decomposed with the eigenvalue decomposition as follows,

$$L = U\Lambda U^T = \sum_{i=1}^{N} \lambda_i u_i u_i^T, \tag{2}$$

where $u_1, u_2, \cdots u_{N-1}$, and u_N are the eigenvectors corresponding to the eigenvalues $\lambda_1, \lambda_2, \cdots \lambda_{N-1}$, and λ_N, respectively. Similar to the traditional Fourier transform, the eigenvalues are considered as the graph frequencies and eigenvectors as the graph Fourier transform basis. Therefore, for a graph signal $f \in R^N$, the graph Fourier transform is defined as

$$\hat{f}(k) = u_k, f = \sum_{i=1}^{N} f(i)\, u_k(i). \tag{3}$$

By using the eigenspace projection matrix

$$P_{\lambda_i} = \sum_{\lambda_i} u_i u_i^T. \tag{4}$$

The graph filter for graph signal in the spectral domain is defined as follows,

$$H = \sum_{\lambda_i \in \sigma(L)} h(\lambda_i) P_{\lambda_i}, \tag{5}$$

where $\sigma(L)$ is the set of eigenvalues of the graph Laplacian matrix L (or $\mathbb{L}$). The filter H acts on a given graph signal **f** by modulating each Fourier mode as

$$Hf = \sum_{k=1}^{N} h(\lambda_k)\, \bar{f}(k)\, u_k. \tag{6}$$

Due to the high complexity of the eigenvalue decomposition, the Chebyshev polynomial approximation is used to implement the graph signal filtering [23].

2.2 | Graph wavelet filter banks

The critically sampled two-channel wavelet filter banks for the bipartite graph signal is shown in Figure 1. It is assumed that the signal f is defined on the bipartite graph G = (L, H, E). The graph filters H_0 and H_1 are analysis filters, and G_0 and G_1 are synthesis filters, which can be defined as Equation (3)

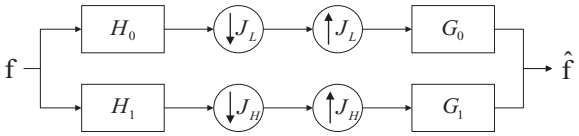

FIGURE 1 Two-channel graph wavelet filter bank

with different graph filter kernels $h_0(\lambda)$, $h_1(\lambda)$, $g_0(\lambda)$ and $g_1(\lambda)$. The operators $\downarrow J_L$ and $\uparrow J_L$ are downsampling and upsampling functions in the first channel, respectively. In order to design critically sampled wavelet filter banks, Narang [24] defines the diagonal downsampling matrix $J_L = diag\{J_L(n)\}$, in which

$$J_L(n) = \begin{cases} +1, & if\ n \in L \\ -1, & if\ n \in H \end{cases}. \tag{7}$$

The downsampling and then upsampling operator can be represented as

$$L_{DU} = \frac{1}{2}\left(I + J_L\right), \tag{8}$$

where I is the identity matrix. In the second channel, the diagonal downsampling matrix is $J_H = -J_L$. Based on the perfect reconstruction condition,

$$\frac{1}{2}G_0\left(I + J_L\right)H_0 + \frac{1}{2}G_1\left(I + J_H\right)H_1 = I. \tag{9}$$

Narang et al. [24] gives the necessary and sufficient condition for the perfect reconstruction condition in bipartite graphs filter banks as follows,

$$\begin{aligned} g_0(\lambda)h_0(\lambda) + g_1(\lambda)h_1(\lambda) &= c^2 \\ g_0(\lambda)h_0(2-\lambda) - g_1(\lambda)h_1(2-\lambda) &= 0 \end{aligned}. \tag{10}$$

They also extend the well-known quadrature mirror filter (QMF) solution to the case of bipartite graphs, such as the ideal kernel and Meyer kernel functions.

2.3 | 2D graph wavelet transform

A digital image is a regular 2D signal, which can be considered as a graph signal by connecting each pixel in the image to its adjacent pixels (vertices) and defining the pixel value as the value of the graph signal. Formally, we define an image texture as a signal on the undirected graph G = (V, E), in which V is the set of pixels (vertices), and E is the set of links (edges) of pixels. Three typical graphs [24] for an image are shown in Figure 2. The graph in Figure 2a is 8-connected image graph that is 4-partite, in which each pixel is a vertex on the graph and each vertex is connected to its eight adjacent vertices.

The graph wavelet filter banks are implemented on bipartite graphs. Therefore 2D graph wavelet transform on an image can be performed by decomposing an arbitrary graph into bipartite subgraphs. For example, the 8-connected graph can be

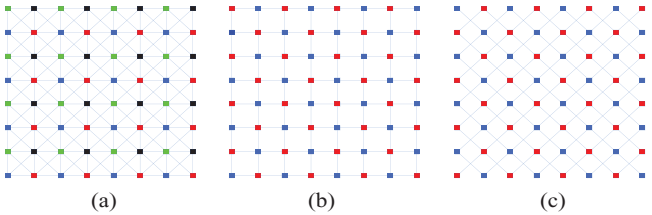

FIGURE 2 Graph-formulation of 2D images. (a) 8-connected image-graph G, (b) 4-connected rectangular image-graph, (c) 4-connected diagonal image graph

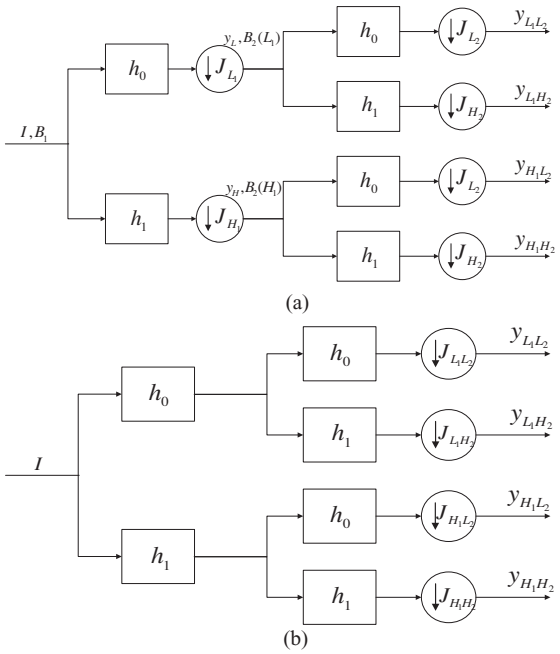

FIGURE 3 2D separable two-channel filter bank. (a) Graph wavelet decomposition on an image. (b) Equivalent implementation of graph wavelet decomposition

decomposed into two 4-connected bipartite graphs $B_1 = (L_1, H_1, E_1)$ and $B_2 = (L_2, H_2, E_2)$ as shown in Figure 2b,c [24]. Similar to the traditional 2D separable wavelet transform, two-channel graph wavelet filter banks are applied in a cascaded manner, by filtering along a series of bipartite subgraphs of the original graph. Specifically, the graph analysis filters are firstly applied on the first bipartite graph B_1(shown in Figure 2b). After downsampling the filtering signals, the graph analysis filtering and downsampling are operated on the signals on two disjoint graphs $B_2(L_1)$ and $B_2(H_1)$, separately, which is shown in Figure 3a. Narang [24] has also proven that the graph wavelet decomposition on bipartite graphs in Figure 3a is the same as the implementation in Figure 3b. In Figure 3, the graph filter kernels h_0 and h_1 are used to construct the graph filters with Equation (3) on the corresponding bipartite subgraphs. The downsampling operator $\downarrow J_{ij}$ ($i = L_1, H_1$ and $j = L_2, H_2$) is to preserve those vertices corresponding to $J_i(\cdot) = 1$ and $J_j(\cdot) = 1$.

3 | FEATURE EXTRACTION AND CLASSIFICATION

It is well known that the traditional wavelet transform possesses better time-frequency localization and multiscale analysis properties. The wavelet transform based texture feature extraction method explores those advantages and provides excellent texture descriptors. However, the wavelet transform does not explicitly consider the local relationship of a texture. Therefore, in this paper, an image texture is represented as a graph signal. Graph wavelet transform combines the advantages of wavelet transform and graph signal processing. It provides a multiscale analysis way for the graph signal. This new transform is used to obtain the multiscale representation of an image texture. The second key component of our method is the local singular value decomposition, because the truncation of lower singular values can be considered as extracting the major information or denoising. Unlike the general truncation method, we take the median, maximum and mean values of the local singular values in each local region. Finally, we use the Weibull distributions to characterize their statistical properties. The model parameters serve as the texture feature. The image texture feature extraction method proposed in this paper is summarized in Figure 5, which includes the following aspects: 1) graph wavelet transform of texture images; 2) local singular value decomposition; 3) Weibull distribution modeling and parameter estimation.

3.1 | Graph wavelet transform

One of the advantages of the critically sampled two-channel graph-based wavelet transform is that the number of transform coefficients is the same as the number of image pixels. Thus this critically sampled transform has been used in image compression [39]. However, as mentioned in the work [10], the overcomplete wavelet decomposition is much better than the complete wavelet transformation for texture analysis. Therefore, we combine the overcomplete transform with critically sampled transform as shown in Figure 4.

Compared with the critically sampled graph wavelet transform shown in Figure 3b, our graph wavelet transform omits the downsampling operation in the LH, HL and HH channels

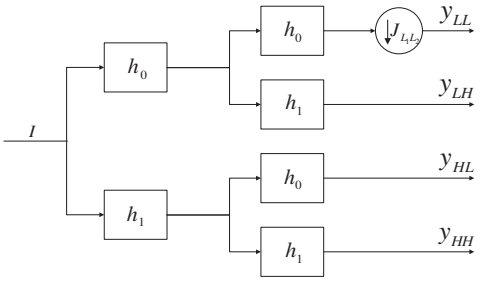

FIGURE 4 Overcomplete two-channel graph filter bank decomposition of an image texture. Only the LL channel is subjected to the downsampling operation and used for the next level decomposition

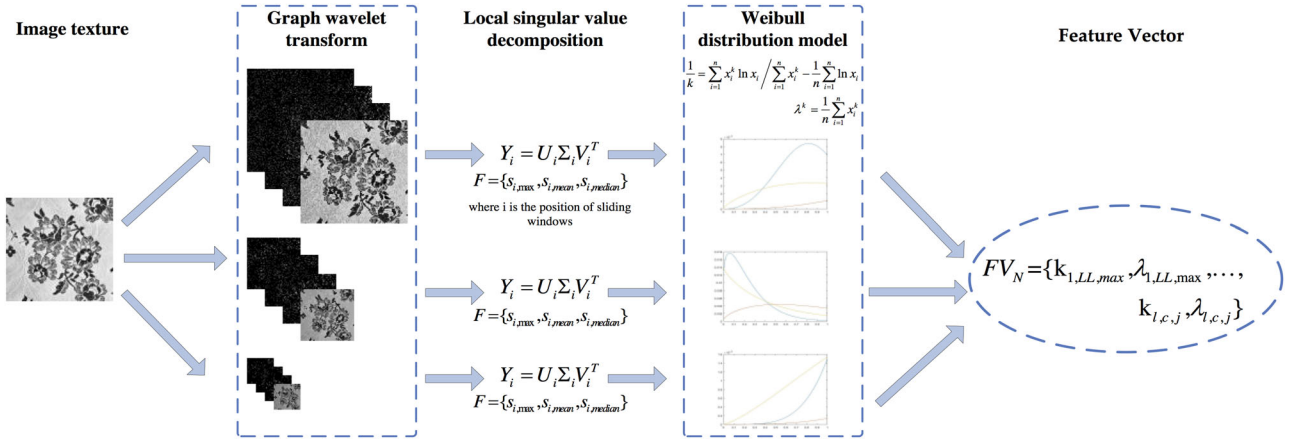

FIGURE 5 The proposed image texture feature extraction method

and only preserves the downsampling operator in the LL channel. That is to say, after one level decomposition, the sizes of graph wavelet coefficients of the LH, HL and HH channels equal to the size of the original image texture, and the size of the LL channel is one fourth of the size of the original image. Then the next level decomposition is performed on the LL channel.

In the traditional wavelet frame transform [10], the LL channel is not downsampled. But the analysis filters must be upsampled by inserting a zero between two filter taps. For graph filters, they are defined in the graph spectral domain. We cannot simply unsampled the graph filters for the next level decomposition in the vertex domain. Therefore, in order to obtain the multiscale overcomplete graph wavelet decomposition, we apply the downsampling operator to the LL channel.

3.2 | Local singular value decomposition

Singular value decomposition is a very powerful tool, which is successfully applied in the dimensionality reduction, solving linear equations and noise reduction [36]. Let Y be a matrix of size $m \times n$, its SVD is defined as

$$Y = U\Sigma V^T, \tag{11}$$

where Σ is a diagonal $m \times n$ matrix with non-negative real diagonal elements. The diagonal entries of Σ are singular values. The left singular vectors and right singular vectors form the orthogonal matrices U and V, respectively. The superscript T denotes the matrix transpose.

Selvan et al. [37] showed that the singular value decomposition of wavelet transform coefficients can be used for texture classification. Kim and So [38] modified the SVD based texture feature in the Haar wavelet domain, and demonstrated the improved classification performance. Nevertheless, it is well known that texture is a region property. The singular value decomposition of the whole coefficient matrix of a wavelet subband only capture the major information of that subband. It is difficult to characterize the local property of the texture. Meanwhile, the number of non-zero singular values equals to the rank

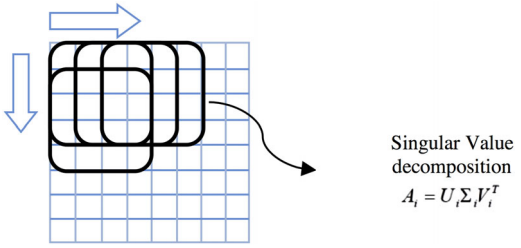

FIGURE 6 Local singular value decomposition. We choose 4×4 window to calculate local singular values

of a coefficient matrix. When the size of the texture sample is small, it is hard to obtain a good probability distribution model for fitting the singular values, especially in the critically sampled wavelet domain, in which the size of each subband at next level decomposition is one quarter of the size of the subband at the current level. Therefore, in this paper, we make use of local SVD of graph wavelet subbands to extract the texture feature.

As for a graph wavelet subband, the local SVD is shown in the Figure 6. We perform the singular value decomposition on the region within the window. Then the window is moved to the next coefficient to perform local singular value decomposition on the windowed region. That is to say, we use the SVD to extract the major information in the local region. Selecting the size of the window plays an important role in texture feature extraction. It can directly affect the performance of texture classification. At present, there is no general way to determine the optimal size of the window without supervision. Considering the computational complexity and ensuring a good texture classification, we empirically choose a window with the size of 4×4.

It is well known that low singular value truncation is considered as a noise reduction method. This manipulation can reduce the noise influence in texture classification when the image texture is corrupted by noise. There are many methods for low singular value truncation in the literature. For example, it can define a measure called index of separability [36] as

$$\alpha = \frac{\sigma_1^2}{\sum_{i=1}^{m} \sigma_i^2}, \tag{12}$$

to select the significant singular values. This truncation method works well when number of singular values is large. However, in the local SVD with 4×4 window, there are 4 singular values. It seems that the general truncation does not suitable for texture description.

In order to extract the texture features and make the proposed method to be noise-resistant to some extent, we take the median, maximum and mean values of the local singular values in each window, and form three sets respectively, which will be modeled with Weibull distributions later. It is assumed that there are N local regions in a subband, and the singular values of the i^{th} region form a vector S_i. We can obtain three sets F_1, F_2 and F_3,

$$\begin{aligned} F_1 &= \{\max(S_1), \max(S_2), \dots, \\ &\qquad \max(S_i), \dots \max(S_N)\} \\ F_2 &= \{\text{mean}(S_1), \text{mean}(S_2), \dots, \\ &\qquad \text{mean}(S_i), \dots \text{mean}(S_N)\} \\ F_3 &= \{\text{median}(S_1), \text{median}(S_2), \dots, \\ &\qquad \text{median}(S_i), \dots \text{median}(S_N)\} \end{aligned} \tag{13}$$

which are composed of the median, maximum and mean values of every vector S_i, respectively.

3.3 | Weibull distribution model

The Weibull distribution has been widely used in various applications, such as reliability analysis and weather forecasting. In general, there are one-parameter Weibull distribution, two-parameter Weibull distribution, three-parameter Weibull distribution and hybrid Weibull distribution. From the perspective of probability and statistics, the probability density function of the two-parameter Weibull distribution [40] is defined as follows,

$$f(x;\lambda,k) = \begin{cases} \frac{k}{\lambda}\left(\frac{x}{\lambda}\right)^{k-1} e^{-(x/\lambda)^k} & x \geq 0 \\ 0 & x < 0 \end{cases}, \tag{14}$$

where x is a random variable, $\lambda > 0$ is the proportional parameter, $k > 0$ is the shape parameter. Obviously, the Weibull distribution function becomes an exponential distribution, if the shape parameter k equals to 1. When $k = 2$, it is a Rayleigh distribution. If $k = 3.458$, it is most similar to a normal distribution. Thus the Weibull distribution possesses great potential in the data modeling.

When we model the extracted sets F1, F2 and F3 in a subband, the key issue is to estimate the distribution parameters. Without loss of generality, let the elements of the set F1 to be $0 \leq x_1 \leq x_2 \leq \dots \leq x_N$, it can be determined the likelihood function from the Weibull probability density function as,

$$L(x;\lambda,k) = \prod_{i=1}^{N} \left(\frac{k}{\lambda}\right)\left(\frac{x_i}{\lambda}\right)^{k-1} \exp\left[-\left(\frac{x_i}{\lambda}\right)^k\right]. \tag{15}$$

Taking the partial derivative of the logarithm of the likelihood function to be zero, we obtain the following equations,

$$\begin{aligned} \frac{1}{k} &= \sum_{i=1}^{N} x_i^k \ln x_i \Big/ \sum_{i=1}^{N} x_i^k - \frac{1}{N}\sum_{i=1}^{N} \ln x_i \\ \lambda^k &= \frac{1}{N}\sum_{i=1}^{N} x_i^k, \end{aligned} \tag{16}$$

from which the parameters can be estimated.

In order to verify the fitting performance of Weibull distribution, we choose a Brodatz image texture, and model the maximum, mean and median values of local singular values in graph wavelet subbands. The fitting results on LL, LH channels of the first level graph wavelet decomposition are shown in Figure 7. It can be seen that the Weibull distribution works well in the local singular value fitting experiments.

3.4 | Feature extraction and classification

For an image texture, the feature extraction procedures are summarized as follows,

1) Decompose the image texture with L-level graph wavelet filter banks as shown in Figure 5, and obtain graph wavelet subbands $WS_{l,c}$ ($l = 1, \dots, L$ and c = LL, LH, HL, HH).
2) In each subband $WS_{l,c}$, implement SVD on the local region and form the sets $F_{l,c,j}$, in which $j = \max, \text{mean}, \text{median}$.
3) Model $F_{l,c,j}$ with Weibull distribution, and estimate the parameters $k_{l,c,j}$ and $\lambda_{l,c,j}$.
4) Accumulate all parameters $k_{l,c,j}$ and $\lambda_{l,c,j}$ into a feature vector to describe the texture.

During the texture classification, we need measure the similarity of feature vectors. As for Weibull distributions, the Kullback–Leibler divergence between two distributions $f(x|\lambda_1, k_1)$ and $g(x|\lambda_2, k_2)$ is given by Bauckhage [41] as follows,

$$\begin{aligned} KL(f|g) &= \int_{-\infty}^{\infty} f(x|\lambda_1,k_1)\log\left(\frac{f(x|\lambda_1,k_1)}{g(x|\lambda_2,k_2)}\right)dx \\ &= \log\frac{k_1}{l_1^{k_1}} - \log\frac{k_2}{l_2^{k_2}} + (k_1-k_2)\left[\log l_1 - \frac{\gamma}{k_1}\right] \\ &\quad + \left(\frac{l_1}{l_2}\right)^{k_2}\Gamma\left(\frac{k_2}{k_1}+1\right) - 1. \end{aligned} \tag{17}$$

However, Kullback–Leibler divergence is not symmetric, the symmetrized Kullback–Leibler divergence,

$$\text{KLS}(f,g) = \text{KL}(f|g) + \text{KL}(g|f), \tag{18}$$

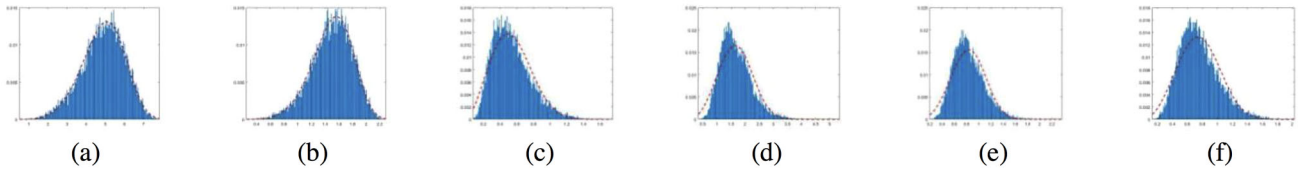

FIGURE 7 The histograms of the max, mean, median and values of local singular values and the corresponding fitting curves for the Brodatz image texture "d004". (a) LL-max, (b) LL-mean, (c) LL-median, (d) LH-max, (e) LH-mean, (f) LH-median

TABLE 1 Selected 30 Brodatz image textures

Brodatz(30) texture ID					
D005	D006	D009	D014	D018	D020
D022	D026	D030	D034	D036	D041
D045	D047	D048	D052	D055	D062
D064	D069	D073	D075	D082	D088
D091	D099	D101	D104	D107	D109

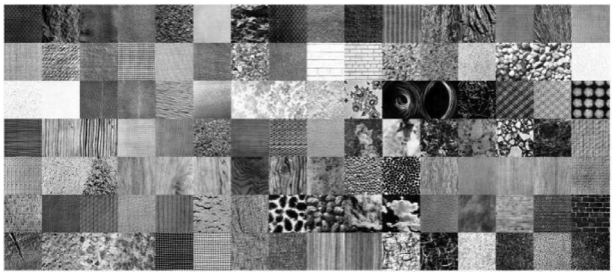

FIGURE 8 Brodatz image texture dataset

is used in texture classification. For two feature vectors FV1 and FV2, the similarity measure for texture classification is defined as

$$D(FV_1, FV_2) = \sum_{l=1}^{L} \sum_{c\in\{LL,LH,HL,HH\}} \sum_{j\in\{\max imum,mean,median\}} KLS_{l,c,j}. \quad (19)$$

4 | EXPERIMENTAL RESULTS AND DISCUSSIONS

In our experiment, five different experimental datasets constructed by four benchmark texture datasets are used to evaluate the proposed method.

The first experimental dataset contains 30 texture images (listed in Table 1) which are selected from the Brodatz texture database as shown in Figure 8 [10]. There are 112 textures in dataset, and the size of each image is 512 × 512. The second experimental dataset contains the whole Brodatz dataset. In our experiments, each texture image is subdivided into 64 non-overlapping sub-images of size 64 × 64. For each texture class, 32 texture samples are used for training, and the rest are served as test samples to verify the performance of the proposed method. Therefore, the first experimental dataset consists of 960 training texture images, with 30 texture classes and 32 images per class. The number of testing textures is the same as that of training samples. In the second experimental dataset, the numbers of training and testing texture samples are 7168, respectively.

TABLE 2 Selected 50 Vistex image textures

Vistex (50) texture ID				
Bark02	Bark03	Bark04	Bark08	Bark09
Brick01	Brick02	Brick04	Build09	Fabric00
Fabric02	Fabric04	Fabric07	Fabric09	Fabric11
Fabric13	Fabric15	Fabric17	Fabric18	Flowers00
Flower02	Flower03	Flower04	Flower05	Food02
Food07	Food08	Food10	Food11	Grass01
Grass02	Leaves00	Leaves02	Leaves05	Leaves10
Leaves13	Leaves15	Misc03	Misc04	Misc06
Stone01	Stone02	Tile03	Tile04	Tile07
Water02	Water04	Water07	WheresWaldo02	Wood02

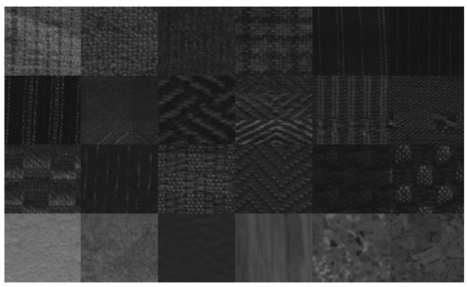

FIGURE 9 Outex_TC_00010 image texture dataset

The third experimental dataset including 50 classes of textures are built from the VisTex database, which are listed in Table 2 [10]. Each texture image is divided into 64 non-overlapping sub-images of size 64 × 64. Therefore, the training dataset and testing dataset all consist of 1600 texture images and 32 images per class.

We choose Outex_TC_00010 database as the fourth dataset in our experiment which contains 24 types of texture images as shown in Figure 9. For each class, there are 20 non-overlapping

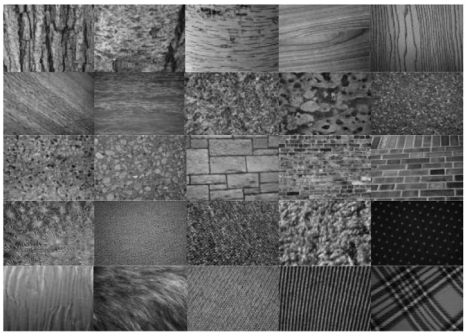

FIGURE 10 UIUC image texture dataset

128 × 128 texture samples in each rotation angle under a given illumination condition. The sample of illumination "inca" and angle 0° in each class are employed to train the classifier [28]. In order to highlight the advantages of the method proposed in this paper, we extract the texture with the size of 64 × 64 in the centre of the image for the experiment. Therefore, there are 24 × 20 training textures and 24 × 8 × 20 testing samples.

The fifth experimental dataset is constructed from UIUC texture database as shown in Figure 10, which involves 25 texture classes and 40 texture images for each class. The resolution of each texture image is 640 × 480 pixels [42]. Similar to the previous database, we reduce the image size to one quarter of the original size. In the experiment, we select the first 20 texture samples to test the method, and the last 20 images for training the classifier.

Furthermore, in order to eliminate the different dynamic ranges in the gray levels of subimages derived from the same original image, each subimage is individually normalized to zero-mean and unit variance. In our experiments, we make use of Nearest Neighbour classifier with the similarity measure Equation (20). The average classification correct rate,

$$\mathrm{ACCR} = \frac{number\ of\ correctly\ classified\ textures}{total\ number\ of\ test\ textures} \times 100\%, \quad (20)$$

are utilized to evaluate the classification performance.

We will conduct the classification experiments by considering the following aspects.

1) Number of graph wavelet decomposition levels. It aims to determine whether the number of decomposition levels influences the discrimination power for texture classification and evaluate the computational complexity of the feature extraction.
2) Different wavelet kernel functions. We will use different graph wavelet kernel functions to test the classification performance of the proposed texture feature.
3) Feature selection. The sequential forward floating selection (SFFS) method is used to further verify the discrimination power of the introduced feature.
4) Comparison with the existing methods. We will compare our experimental results with the state-of-art works to test the performance of the proposed method.
5) Noise resistance performance. In our experiments, the zero mean additive white Gaussian noises are added into the texture samples to test the noise resistance performance of the introduced method.

4.1 | Number of graph wavelet decomposition levels

In order to determine whether the number of decomposition levels influence the discrimination power for texture classification, and choose a reasonable decomposition level for the texture feature extraction, we conduct the texture classification test on five datasets with different levels of graph wavelet decomposition. For better comparison, we list the average time consuming of the feature extraction for each texture image. The ACCRs with the Meyer graph wavelet kernel function and corresponding extraction time are shown in Table 3.

It can be seen that the ACCRs of the texture features extracted from the two-level decomposition is about 1.6% better than that of the one-level decomposition, and the extraction time is about 0.2–0.3s longer than that with the one level decomposition. Meanwhile, except the UIUC database, with the increase of the decomposition level after two levels, the ACCRs will decrease, and the computational complexity is increasing. Especially for the four level decomposition, the ACCRs are much lower (around 4.0–10.2%) than those of the two level decomposition. The reason is that for a texture image with the size of 64 × 64, the size of each subband at the fourth level will be 8 × 8, which will result in 25 values in the median, maximum and mean sets, respectively. It is difficult to effectively estimate the parameters of Weibull distribution with those random samples. Therefore, in the following experiments, we set the decomposition level L to be 2.

4.2 | Different graph wavelet kernel functions

Narang [24] proposed the graph-QMF that is to design a single spectral kernel function $h_0(\lambda)$ and select other spectral kernel functions as

$$\begin{aligned} h_1(\lambda) &= h_0(2-\lambda) \\ g_0(\lambda) &= h_0(\lambda) \\ g_1(\lambda) &= h_1(\lambda) = h_0(2-\lambda) \end{aligned} \quad . \quad (21)$$

This solution extends the well-known quadrature mirror filter (QMF) solution to the case of bipartite graphs. Therefore, as the traditional wavelet kernel, the ideal spectral kernel function $h_0(\lambda)$ is defined as

$$h_0^{ideal}(\lambda) = \begin{cases} c, & \lambda < 1 \\ \dfrac{c}{\sqrt{2}}, & \lambda = 1 \\ 0, & \lambda > 1 \end{cases} . \quad (22)$$

TABLE 3 Experimental results on multi-level graph wavelet decomposition

Level	1	Costs	2	Costs	3	Costs	4	Costs
Brodatz(112)	87.00%	2.23s	88.70%	2.40s	87.75%	2.48s	78.52%	3.02s
Brodatz(30)	93.96%	2.09s	95.52%	2.39s	94.69%	2.43s	91.56%	2.51s
Vitex(50)	87.88%	2.16s	89.44%	2.35s	87.69%	2.43s	80.80%	2.53s
Outex(24)	83.88%	2.08s	85.36%	2.36s	82.13%	2.41s	79.04%	2.49s
UIUC(25)	70.80%	3.24s	74.40%	3.48s	74.80%	3.67s	74.00%	3.90s

TABLE 4 Experimental result with different graph wavelet kernel functions

	Meyer		DB6		Ideal	
Data	KNN	KNN-SFFS	KNN	KNN-SFFS	KNN	KNN-SFFS
Brodatz(30)	95.52%	96.25%	95.52%	96.15%	91.98%	94.98%
Bradatz(112)	88.70%	94.79%	88.76%	94.90%	81.81%	91.04%
Vitex(50)	89.44%	95.94%	89.94%	96.25%	76.94%	91.25%
Outex(24)	85.36%	92.01%	85.81%	92.01%	82.40%	90.31%
UIUC(25)	74.40%	75.60%	75.20%	76.00%	71.60%	73.60%

The Meyer spectral kernel is

$$h_0^{Meyer}(\lambda) = \sqrt{v\left(2 - \frac{3}{2}\lambda\right)}, \tag{23}$$

where

$$v(\lambda) = \begin{cases} 0, & \lambda \le 0 \\ 3\lambda^2 - 2\lambda^3, & 0 \le \lambda \le 1 \\ 1, & \lambda \ge 1 \end{cases}. \tag{24}$$

We can also construct the spectral kernel function with the traditional Daubechies wavelet filters.

For example, the low-pass Daubechies wavelet filter is $h_0 = \{h_0(0), h_0(1), \ldots, h_0(N-1)\}$, then the corresponding spectral kernel function for the graph wavelet can be defined as,

$$h_0^{Db}(\lambda) = \left| \sum_{i=0}^{N-1} h_0(i) \exp\left(\mathrm{j} \cdot \pi \cdot \lambda / 2 \cdot i\right) \right|, \tag{25}$$

where $\mathrm{j} = \sqrt{-1}$ is used to represent the complex value. This spectral kernel can be considered as the scaled amplitude spectrum.

In this section, we decompose the image texture with ideal spectral kernel (Ideal), Meyer spectral kernel (Meyer) and Daubechies wavelet (the length of filter is 6) kernel (DB6) based graph wavelet filters to extract the texture feature, and texture classification experiments are also conducted with feature selection. The results are listed in Table 4. It can be found that three wavelet kernels bring competitive classification performance. For the first experimental dataset (30 Brodatz texture), the results of graph filters based on Daubechies wavelet with filter length 6 are slightly worse than that of Meyer kernel based graph wavelet filters. On the second, third, fourth and fifth experimental datasets, the ACCRs of Daubechies wavelet with filter length 6 based graph wavelet filters are the best one. In order to fairly evaluate our method and compare it with other existing methods, we choose the Meyer spectral kernel for the graph wavelet transform in the following experiments.

4.3 | Comparison with existing methods

We will evaluate the proposed methods by comparing it with the existing techniques. The following approaches are included in our experiments.

1) GWT+Energy. The energy features, the mean values and standard deviations of amplitude of wavelet coefficients are always the baseline for texture classification. Therefore, we also extract these energy features for comparison.
2) DWT+GLCM. Gray level co-occurrence matrix (GLCM) is another baseline method for texture classification. The texture feature based on GLCM in the traditional wavelet domain is also used to be compared in the experiments.
3) GWT+GLCM. It is obtained by extracting GLCM based feature in the graph wavelet transform domain.
4) MDSV. Ramakrishnan et al. [36] applied the SVD decomposition in multiwavelet subbands. They extracted the energy, entropy, local homogeneity and max–min ratio of the singular values to characterize the texture.
5) CLBC. Zhao et al. [43] proposed the local binary count (LBC) based texture feature. In this framework, each pixel in the local region is turned to binary values by

TABLE 5 Experimental comparison of different methods

Method	Brodatz(30)	Brodatz(112)	Vitex(50)	Outex(24-64 × 64)	UIUC(25)
GWT+Energy	92.71%	81.75%	80.44%	78.13%	64.00%
DWT+GLCM	90.10%	79.19%	76.73%	76.82%	62.80%
GWT+GLCM	90.31%	82.11%	82.31%	77.60%	65.20%
MDSV	91.56%	84.40%	83.63%	78.64%	68.20%
CLBC_S_M/C	86.88%	71.76%	59.19%	67.81%	58.80%
CLBC_S/M	88.96%	79.80%	78.94%	75.55%	56.00%
CLBC_S/M/C	94.79%	86.99%	82.13%	81.28%	72.00%
CLDP	95.41%	87.97%	87.44%	**86.90%**	73.40%
CWT-GGD	92.10%	83.10%	84.60%	—	—
MDCM	91.35%	84.49%	87.19%	82.08%	68.40%
Jet Texton Learning	94.17%	**91.02%**	84.50%	84.94%	71.20%
Proposed method	**95.52%**	**88.70%**	**89.44%**	**85.36%**	**74.40%**

Note. The best result is bolded.

comparing it with the central pixel, and then the number of value 1′s is counted to form the feature. The completed LBC is also introduced to enhance the performance. The features CLBC_S/M, CLBC_S_M/C and CLBC_S/M/C are compared with our method.

6) CLDP. Hu et al. [44] proposed a new texture descriptor, completed local derivative pattern (CLDP). The new component in CLDP, with regarded as the directional derivative pattern, reflects the directional smoothness of local textures without increasing computation complexity.
7) CWT+GGD. Qiao et al. [11] combined the phase information and amplitude information of the complex wavelet coefficients into a real quantity, and then used the real generalized Gaussian distribution (GGD) to extract the feature for texture classification.
8) MDCM. Li et al. [45] developed the marginal distribution covariance model (MDCM) in the multiple wavelet domain. The texture features were extracted by using the orthogonal wavelet transform, Gabor wavelet transform and dual-tree complex wavelet transform for texture classification.
9) Jet Texton Learning. Roy et al. [46] used a new class of jet texton learning to classifying textures, in which a Jet space representation of the image is derived from several filter responses, the jet textons dictionary is learned using K-means clustering algorithm. The feature distribution of jet texton is considered as a model for texture classification.

The experimental results are listed in Table 5. In the existing methods, the best ACCRs for five experimental datasets are 95.41% (CLDP) [44], 91.02% (Jet Texton Learning) [44], 87.44% (CLDP) [44], 86.90% (CLDP) [44] and 73.40%(CLDP), respectively. Meanwhile, the ACCRs of our proposed method are 95.52%, 88.70%, 89.44% 85.36% and 74.40%. It can also be seen from Table 4 that, after using the feature selection with SFFS method, our ACCRs reach to 96.25%, 94.79%, 95.95%, 92.01% and 75.60%. Although the ACCRs of the proposed method on the Outex texture database and Brodatz(112) database are slightly lower than those of CLDP and Jet Texton Learning methods, our accuracy is 5.11% and 3.77% higher than those methods after feature selection. Therefore, compared with the existing methods, the proposed texture classification method shows its potential in texture classification.

TABLE 6 Experimental results of noise resistance

SNR	Brodatz (30)	Brodatz (112)	Vitex (50)	Outex (24)	UIUC (25)
10	89.48%	78.91%	68.56%	73.41%	61.60%
15	91.77%	84.38%	76.50%	78.85%	66.40%
20	93.44%	86.89%	81.75%	81.30%	68.00%
25	94.37%	88.25%	85.88%	82.27%	70.40%
30	95.31%	88.42%	86.94%	83.41%	73.20%

4.4 | Noise resistance performance

As we know, the low singular value truncation has the function of noise reduction. The paper extracts the mean values, the maximum values and the median values of the local singular values, and applies Weibull distributions. In order to test the noise resistance performance of the introduced method, we add the zero mean additive white Gaussian noise (AWGN) to the image textures, and apply the original classification method. The experimental results with different signal-to-noise ratios (SNR) are shown in Table 6.

It can be seen from the Table 6 that when the SNR decrease, the ACCRs on those datasets become lower. If the SNR is greater than 30db, the noise has little effect on the accuracy of texture classification. When SNR is 20db, ACCRs are worse than those without adding AGWN by around 2.48%, 1.81%, 7.69%, 1.95% and 6.40% on the four datasets, respectively. Once SNR reaches to 10 dB, the performance of the pro-

posed method becomes worse severely. Therefore, the proposed method has a noise resistance capability to some extent.

5 | CONCLUSION

This paper introduces an image texture classification method based on graph wavelet filter banks. The image texture is considered as a graph signal. By using graph wavelet transform, the texture image is decomposed into the multi-scale representation. Meanwhile we make use of the local singular value decomposition to extract the local information of the texture image in the graph wavelet domain, and then model them with Weibull distributions to extract the texture feature. The experiments are conducted on four benchmark texture datasets, which includes graph wavelet decomposition levels, different graph spectral kernel functions, comparison with other existing methods and noise resistance performance evaluation. Experimental results show that the proposed texture features have better texture classification performance.

ACKNOWLEDGEMENTS

This work is supported by National Natural Science Foundation of China under Grant 61871142. The authors would like to thank the anonymous reviewers for their valuable revise opinion and suggestions in improving the technical presentation of this paper.

REFERENCES

1. Tuceryan, M., Jain, A.K.: Texture analysis. In: Handbood of Pattern Recognition & Computer Vision, pp. 207–248.World Scientific, Singapore (2014)
2. Hayati, S., Ahmadzadeh, M.R.: WIRIF: Wave interference-based rotation invariant feature for texture description. Signal Process. 160–171 (2018)
3. Khademi, A., Krishnan, S.: Medical image texture analysis: A case study with small bowel, retinal and mammogram images. Electr. Comp. Eng. IEEE 1949–1954 (2008)
4. Niwas, S.I., et al.: Analysis of nuclei textures of fine needle aspirated cytology images for breast cancer diagnosis using complex daubechies wavelets. Signal Process. 93(10), 2828–2837 (2013)
5. Filho, P.L.P., et al.: Forest species recognition using color-based features. In: International Conference on Pattern Recognition, pp. 4178–4181.IEEE Computer Society, Los Alamitos (2010)
6. Zhu, Y., et al.: Improved principal component analysis and linear regression classification for face recognition. Signal Process. 145, 175–182 (2018)
7. Cavalin, P., Oliveira, L.S.: A review of texture classification methods and databases. Graph. Patterns Images Tutorials IEEE 1–8 (2018)
8. Mallat, S.G.: A theory for multiresolution signal decomposition: The wavelet representation. IEEE Comp. Soc. 11(7), 674–693 (1989)
9. Sweldens, W.: The lifting scheme: A construction of second generation wavelets. SIAM J. Math. Anal. 29, 511–546 (1997)
10. Unser, M.: Texture classification and segmentation using wavelet frames. IEEE Trans. Image Process. 4(11), 1549–1560 (1995)
11. Qiao, Y.L., et al.: Complex wavelet based texture classification. Neurocomputing 72(16), 3957–3963 (2009)
12. Çelik, T., Tjahjadi, T.: Multiscale texture classification and retrieval based on magnitude and phase features of complex wavelet subbands. Comp. Electr. Eng. 37(5), 729–743 (2011)
13. Ji, H., et al.: Wavelet domain multifractal analysis for static and dynamic texture classification. IEEE Trans. Image Process. 22(1), 286–299 (2013)
14. Wang, S., Wang, G.: Texture classification by multifractal spectrum and barycentric coordinates of bit planes of wavelet coefficients. IET Image Proc. 11(12), 1205–1209 (2017)
15. Yang, P., et al.: Fusing DTCWT and LBP based features for rotation, illumination and scale invariant texture classification. IEEE Access 776(6), 13336–13349 (2018)
16. Depeursinge, A., et al.: Rotation-covariant texture learning using steerable Riesz wavelets. IEEE Trans. Image Process. 23(2), 898–908 (2014)
17. Depeursinge, A., et al.: Steerable wavelet machines (swm): Learning moving frames for texture classification. IEEE Trans. Image Process. 26(4), 1626–1636 (2017)
18. Haraclick, R.M.: Texture features for image classification. IEEE Trans. Syst. Man Cybern. 3(6), 610–621 (1973)
19. Haralick, R.M.: Statistical and structural approaches to texture. Proc. IEEE 67(5), 786–804 (2005)
20. Yuan, Y., et al.: Hyperspectral image classification via multitask joint sparse representation and stepwise mrf optimization. IEEE Trans. Cybern. 46(12), 2966–2977 (2017)
21. Dong, Y., Ma, J.: Wavelet-based image texture classification using local energy histograms. IEEE Signal Process Lett. 18(4), 247–250 (2011)
22. Crovella, M., Kolaczyk, E.: Graph wavelets for spatial traffic analysis. In: Joint Conference of the IEEE Computer and Communications Societies, pp. 1848–1857.IEEE, Piscataway (2002)
23. Hammond, D.K., et al.: Wavelets on graphs via spectral graph theory. Appl. Comput. Harmonic Anal. 30(2), 129–150 (2011)
24. Narang, S.K., Ortega, A.: Perfect reconstruction two-channel wavelet filter banks for graph structured data. IEEE Trans. Signal Process. 60(6), 2786–2799 (2012)
25. Narang, S.K., Ortega, A.: Compact support biorthogonal wavelet filterbanks for arbitrary undirected graphs. IEEE Trans. Signal Process. 61(19), 4673–4685 (2013)
26. Tanaka, Y., Sakiyama, A.: M-channel oversampled graph filter banks. IEEE Trans. Signal Process. 62(14), 3578–3590 (2014)
27. Jiang, J.Z., et al.: Lifting-based design of two-channel biorthogonal graph filter bank. IET Signal Proc. 10(6), 670–675 (2016)
28. Oguzhan, T., Vaidyanathan, P.P.: Extending classical multirate signal processing theory to graphs—Part II: M-channel filter banks. IEEE Trans. Signal Process. 65(2), 423–437 (2017)
29. Tremblay, N., Borgnat, P.: Graph wavelets for multiscale community mining. IEEE Trans. Signal Process. 62(20), 5227–5239 (2014)
30. Xu, B., et al.: Graph wavelet neural network. In: International Conference on Learning Representations (ICLR) (2019)
31. Donnat, C., et al.: Learning structural node embeddings via diffusion wavelets. In: The 24th ACM SIGKDD International Conference. ACM, New York (2018)
32. Zhao, Y., et al.: Local quantization code histogram for texture classification. Neurocomputing 207, 354–364 (2016)
33. Po, D., Do, M.: Directional multiscale modeling of images using the contourlet transform. IEEE Trans. Image Process. 15(6), 1610–1620 (2006)
34. Dong, F., et al.: Directional weight based contourlet transform denoising algorithm for OCT image. Intell. Autom. Soft Comput. 19(4), 525–535 (2013)
35. Ren, J., et al.: Noise-resistant local binary pattern with an embedded error-correction mechanism. IEEE Trans. Image Process. 22(10), 4049–4060 (2013)
36. Ramakrishnan, S.: Multiwavelets domain singular value features for image texture classification. J. Zhejiang University-Sci. A (Appl. Phys. Eng.) 8(4), 538–549 (2007)
37. Selvan, S., Ramakrishnan, S.: SVD-based modeling for image texture classification using wavelet transformation. IEEE Trans. Image Process. 16(11), 2688–2696 (2007)
38. Kim, N.C., So, H.J.: Comments on "SVD-based modeling for image texture classification using wavelet transform". IEEE Trans. Image Process. 22(12), 5408–5408 (2013)
39. Narang, S.K., et al.: Critically sampled graph-based wavelet transforms for image coding. In: Signal and Information Processing Association Summit and Conference, pp. 1–4.IEEE, Piscataway (2013)
40. Yuan, F.: Performance evaluation for maximum likelihood and moment parameter estimation me. In: IEEE International Conference on Industrial Engineering and Engineering Management, pp. 802–806.IEEE, Piscataway (2016)

41. Bauckhage, C.: Computing the Kullback-Leibler divergence between two generalized gamma distributions. Eprint Arxiv (2014)
42. Kazak, N., Koc, M.: Some variants of spiral LBP in texture recognition. IET Image Proc. 12(8), 1388–1393 (2018)
43. Zhao, Y., et al.: Completed local binary count for rotation invariant texture classification. IEEE Trans. Image Process. 21(10), 4492–4497 (2012)
44. Hu, Y., et al.: Completed local derivative pattern for rotation invariant texture classification. In: IEEE International Conference on Image Processing (ICIP), pp. 3548–3552.IEEE, Piscataway (2016)
45. Li, C., et al.: Marginal distribution covariance model in the multiple wavelet domain for texture representation. Pattern Recognit. 92(4), 246–257 (2019)
46. Roy, S.K., et al.: Unconstrained texture classification using efficient jet texton learning. Appl. Soft Comput. 86, 105910 (2020)

How to cite this article: Qiao Y-L, Zhao Y, Song C-Y, Zhang K-G, Xiang X-Z. Graph wavelet transform for image texture classification. IET Image Process. 2021;15:2372–2383.
https://doi.org/10.1049/ipr2.12220