Contents lists available at ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

Texture image retrieval based on non-tensor product wavelet filter banks

Zhenyu He [a], Xinge You [a,*], Yuan Yuan [b]

[a] Department of Electronics and Information Engineering, Huazhong University of Science and Technology, China
[b] School of Engineering and Applied Science, Aston University, Birmingham B4 7ET, UK

ARTICLE INFO

Article history:
Received 30 October 2008
Received in revised form
28 December 2008
Accepted 30 January 2009
Available online 10 February 2009

Keywords:
Texture image retrieval
Wavelet
Non-tensor product wavelet filter banks
Generalized Gaussian density

ABSTRACT

In this paper, we present a novel method, which uses non-separable wavelet filter banks, to extract the features of texture images for texture image retrieval. Compared to traditional tensor product wavelets (such as db wavelets), our new method can capture more direction and edge information of texture images, which is highly valuable to reflect the essential properties of the texture images. Experiments show that the proposed method is satisfactory and can achieve better retrieval accuracies than db wavelets.

1. Introduction

Nowadays, more and more images are collected in digital image libraries. To efficiently make use of the image information in the digital image libraries, we have the desire to develop effective and precise methods to browse, search, and retrieval the image collections. Since the 1970s, image retrieval has been an active research area, driven by two major research communities—database management and computer vision, which study image retrieval from text-based and visual-based techniques, respectively. The text-based image retrieval is simple while it relies on the manual annotation of all images. Manual annotation is not only a time-consuming work but also sensitive to human's subjectivity. Content-based image retrieval can overcome the shortcomings of text-based image retrieval. That is, instead of using manual annotation, images would be objectively indexed by computer according to their own visual contents [1]. Texture is an important feature to characterize the content of one image and has been widely used in applications of security control, such as writer identification, iris identification, face identification.

It is very hard to clearly define the concept of "Texture". Based on personal perceptions or driven by special applications, vision researches had presented a number of different definitions of "Texture". It is an innate property of virtually all surfaces, including clouds, trees, bricks, hair, fabric, etc. It contains important information about the structural arrangement of surfaces and their relationship to the surrounding environment. Because texture is an important and useful property in image process and computer vision, rich research works in this field have been done in the past.

The statistical methods are early proposed methods for texture retrieval, which utilize the distribution information of gray value of images. Among the statistical methods, gray level co-occurrence matrix (GLCM) and the autocorrelation function are the most well known and widely used methods to extract texture feature [2]. GLCM estimates image properties related to second-order statistics. For an image I of size $N \times N$, which is denoted as $I(x,y)_{0 \leqslant x,y \leqslant N-1}$, the GLCM M_D for a displacement

[*] Corresponding author.
E-mail addresses: terry.z.y.he@gmail.com (Z. He), xyou@comp.hkbu.edu.hk (X. You), yuany1@aston.ac.uk (Y. Yuan).

0165-1684/$ - see front matter © 2009 Published by Elsevier B.V.
doi:10.1016/j.sigpro.2009.01.021

$D = (D_x, D_y)$ is defined as

$$M_D(i,j) = |\{((x_1,y_1),(x_2,y_2) : I(x_1,y_1) = i, I(x_2,y_2) = j)\}|, \quad (1)$$

where the entry (i,j) of M_D is the number of occurrences of the pair of gray levels i and j, which are a distance D apart. $(x_1,y_1),(x_2,y_2) \in N \times N$, $(x_2,y_2) = (x_1 + D_x, y_1 + D_y)$, $|\cdot|$ is the cardinality of a set. However, the GLCM cannot be used directly because of its large size. Instead, measurements such as inertia, energy, entropy, contrast, local homogeneity, correlation are computed from the GLCM and used as features.

The autocorrelation function is another important statistical method to extract texture features. The autocorrelation function of an image can be used to assess the amount of regularity as well as the fineness/coarseness of the texture present in the image. An important property of many textures is the repetitive nature of the placement of texture elements in the image. For an image $I(x,y)_{0\leqslant x,y\leqslant N-1}$, the autocorrelation function is defined as follows:

$$P(x,y) = \frac{\sum_{u=0}^{N-1}\sum_{v=0}^{N-1} I(u,v)I(u+x,v+y)}{\sum_{u=0}^{N-1}\sum_{v=0}^{N-1} I^2(u,v)}. \quad (2)$$

This function is related to the size of the texture primitive (i.e., the fineness of the texture). If the texture is coarse, then the autocorrelation function will drop off slowly. Otherwise, it will drop off very rapidly. For regular textures, the autocorrelation function will exhibit peaks and valleys.

The geometrical methods for texture retrieval regard that the texture is composed of texture elements or primitives. And these methods usually depend upon the geometric properties of these texture elements. For example, Tuceryan and Jain proposed the extraction of texture tokens by using the properties of the Voronoi tessellation of the given images [2]. Voronoi tessellation was proposed because of its desirable properties in defining local spatial neighborhoods and because the local spatial distributions of tokens are reflected in the shapes of the Voronoi polygons. Moments of area of the Voronoi polygons serve as a useful set of features that reflect both the spatial distribution and shapes of the tokens in the textured image. The $(p+q)$th moment over an image region R is given by the formula

$$m_{pq} = \sum_{(x,y)\in R} (x-x_0)^p (y-y_0)^q I(x,y). \quad (3)$$

Different combination of these moments can reflect different physical features of texture.

Model-based methods for texture retrieval construct some proper models to describe the texture and regard the model parameters as the texture features. Markov random fields (MRFs), which can capture the local (spatial) contextual information in an image, have been popularly used to model texture images. These models are also based on several important assumptions: Markov assumption: each pixel's intensity only depends on the intensities of the neighboring pixels; stationary assumption: the pixel's intensity relationship is independent of the pixel's position in the image. There are a number of ways to model the image using MRF or its counterpoint Gibbs random field (GRF), such as Derin–Elliot model [3] and auto-binomial model [4], both of which were defined by considering only the single pixel and pairwise pixel cliques in the second-order neighbors of a site.

In addition to the MRF and GRF, fractal dimension is another important model-based method to extract texture features [2]. Texture surface has a statistical quality of roughness and self-similarity at different scales. Fractal dimensions are very useful and have become popular in modelling these properties. The concept of fractal dimension is defined as follows: given a bounded set A in a Euclidean n-space, the set A is said to be self-similar when A is the union of N distinct (non-overlapping) copies of itself, each of which has been scaled down by a ratio of r . The fractal dimension D is related to the number N and the ratio r as

$$D = \frac{\log(N)}{\log(1/r)}. \quad (4)$$

The fractal dimension gives a measure of the roughness of a surface. Intuitively, the larger the fractal dimension, the rougher the texture is. Most natural surfaces and in particular texture surfaces are not deterministic as described above but have a statistical variation. This makes the computation of fractal dimension more difficult. There are a number of methods proposed for estimating the fractal dimension D, among which, box fractal dimension is the most well known and widely used.

Signal processing methods for texture retrieval can be divided into two types. One refers to methods in spatial domain. The other refers to methods in frequency domain. Spatial domain filters are the most direct way to capture image texture properties in spatial domain. Early spatial filters were designed to measure the edge density per unit area because of one property: fine textures tend to have a higher density of edges per unit area than the coarser textures. Commonly, Robert operator and Laplacian operator are used. Instead of measuring the edge density, some spatial filters are based on the spatial moments. In fact, to compute the moments around each pixel is equivalent to filter the image by a set of spatial masks. The spatial masks can be computed by a window of size $W \times W$ and a local coordinate system centered within the window [2]. Psychophysical works reveal that human brain will make a frequency analysis when it perceives one image [5]. To simulate human brain, some texture analysis technologies in frequency domain were proposed. The early frequency methods were carried out in the Fourier domain. Later, with the development of multichannel filtering technology, especially with the foundation of wavelet theory, more and more frequency methods utilize the wavelet or wavelet-like transforms rather than the Fourier transform.

Besides these traditional methods, some new methods for image retrieval are proposed in recent publications. For example, Tao et al. made a deep analysis to find which features are beneficial in relevance feedback schemes for image retrieval and accordingly proposed an orthogonal complement component analysis method [6]. Li et al. proposed a multi-training SVM for image retrieval by combining the merits of the cotraining technique and a random sampling method in the feature space [7]. Li et al. proposed a manifold learning technique called discriminant locally linear embedding (DLLE) for visual recognition [8].

In this paper, we propose a novel method based on non-tensor product wavelet filter banks for texture image retrieval. The non-tensor product wavelet filter banks are constructed in our previous work [20], which own a more powerful ability to characterize the properties of textures than the separable wavelets.

The rest of the paper is organized as follows. In Section 2, we briefly review wavelet-based approaches for texture retrieval and the non-tensor product filter banks previously proposed by us. In Section 3, an algorithm combining the non-tensor product filter banks and generalized Gaussian density (GGD) model for texture image retrieval is proposed. The retrieval experiment using this new algorithm and related discussions are offered in Section 4. Finally, the conclusion is made in Section 5.

2. Review of wavelet-based approaches for texture retrieval and non-tensor product wavelet filter banks

2.1. Wavelet-based approaches for texture retrieval

Gabor model is one classic wavelet method for multichannel analysis. The Gabor function is of similar shape as the receptive fields of simple cells in the primary visual cortex. It is localized in both space and frequency domains and has the shape of plane waves restricted by a Gaussian function. The computational model of the 2-D Gabor filters proposed to analyze the image is given as follows [9,10]:

$$h_e(x,y)=g(x,y)\cos[2\pi f(x\cos\theta+y\sin\theta)] \tag{5}$$

and

$$h_o(x,y)=g(x,y)\sin[2\pi f(x\cos\theta+y\sin\theta)], \tag{6}$$

where h_e and h_o denote the so-called even- and odd-symmetric Gabor filters, and $g(x,y)$ is an isotropic Gaussian function. The mean value and the standard derivation of the coefficients of Gabor subbands are selected as features to represent texture features. Afterwards, weighted Euclidean distance (WED) is applied to measure similarity of the features of different texture images.

Though Gabor filter is a good model for texture image retrieval, it still suffers from some inherent shortcomings, which greatly weaken its practicability. One is the Gabor basic function is not orthogonal, as inevitably leads to redundancy. The other is intensively computational cost of Gabor filters because they need to re-convolute the whole image when one orientation or frequency parameter changes. While both of these two defects can be overcome in real tensor product wavelet-based models.

As a multi-resolution analysis method, the real wavelet transform is an atomic decomposition that represents a 1-D signal $f(t)$ in terms of shifted and dilated versions of a prototype band-pass wavelet function $\psi(t)$, and shifted versions of a low-pass scaling function $\phi(t)$ [11,12], that is,

$$f(t)=\sum_b C_k\phi_{a_0,b}(t)+\sum_{a=-\infty}^{a_0}\sum_b W_{a,b}\psi_{a,b}(t), \tag{7}$$

with the coefficients calculated by

$$C_k=\int_{-\infty}^{\infty} f(t)\overline{\phi_{a_0,b}(t)}\,dt \tag{8}$$

and

$$W_{a,b}=\int_{-\infty}^{\infty} f(t)\overline{\psi_{a,b}(t)}\,dt, \tag{9}$$

where $\psi_{a,b}(t)=2^{-a/2}\psi(2^{-a}t-b)$, $\phi_{a,b}(t)=2^{-a/2}\phi(2^{-a}t-b)$, $a,b\in Z$. And a indexes the scale or resolution of analysis, and b indexes the spatial location of analysis. 1-D wavelet can be easily extended to 2-D wavelet by tensor product,

$$\psi^h(x,y)=\phi(x)\psi(y), \tag{10}$$

$$\psi^v(x,y)=\psi(x)\phi(y), \tag{11}$$

$$\psi^d(x,y)=\psi(x)\psi(y). \tag{12}$$

Early real wavelet-based methods for texture retrieval measure the energy or weighted energy signature of coefficients in wavelet subbands as texture features. The basic assumption of these approaches is that the energy distribution in the frequency domain identifies a texture [13]. Generally, L^1-norm and L^2-norm are selected as measurement of energy. In addition, mean and standard derivation are commonly used as energy features. Since most relevant texture information has been removed by iterative low-pass filtering, the energy of the lowest resolution is generally not considered as a texture feature. The advantage of energy-based models is that only a few parameters are needed to describe a texture. While the energy-based model is not sufficient to capture essential properties of textures. It has already been shown that there may be perceptually very different textures that have very similar energy features. A natural extension of the energy method is to model the image by the marginal densities of real wavelet subband coefficients.

Experimental works revealed that histograms of coefficients of natural images in real wavelet high-pass subbands are highly non-Gaussian [14,22]. The non-Gaussian histogram in a subband is surprisingly well fitted by a two-parameter generalized Gaussian model (GGD), which is defined as

$$p(x;\alpha,\beta)=\frac{\beta}{2\alpha\Gamma(1/\beta)}\exp^{-1(|x|/\alpha)^\beta}, \tag{13}$$

where $\Gamma(\cdot)$ is the gamma function, i.e., $\Gamma(\cdot)=\int_0^\infty \exp^{-t}t^{Z-1}\,dt$, $Z>0$. The normalization constant is $Z(\alpha,\beta)=2(\alpha/\beta)\Gamma(1/\beta)$. Besides the GGD model, the Gaussian mixture model (GMM) is also a good model to approximate the statistical distribution of coefficient histogram of real wavelet high-pass subbands.

Hidden Markov tree (HMT) model is a good method to measure the joint statistical distribution of coefficients of real wavelet subband, which makes use of two properties of the real wavelet coefficients: clustering and persistence across scale [11,15–17]. Clustering means if the value of one real wavelet coefficient is large/small, those wavelet coefficients nearby this coefficient have a large possibility to be also large/small. Persistence across scale means the large/small values of wavelet coefficients tend to propagate across scales. The HMT parameter set $\theta = \{P_{S_1}(m), \varepsilon^{mr}_{i,p(i)}, \sigma_{im}\}$ can be regarded as the features of texture images. Generally, Monte-Carlo method is applied to measure the similarity between two HMT models. To reduce the computational cost and improve the retrieval efficiency, Do offered a new measurement method by computing the upper boundary of Kullback–Leibler distance (KLD) between two HMT models [18].

Further, to overcome the drawbacks of real wavelet, such as shift sensitivity, complex wavelet was proposed in [26–28]. The 1-D complex wavelet decomposition is same as Eq. (7), while here $\phi_{a_0,b}(t)$ and $\psi_{a,b}(t)$ are complex. That is,

$$\phi_{a_0,b} = \phi^r_{a_0,b} + \sqrt{-1}\phi^i_{a_0,b} \tag{14}$$

and

$$\psi_{a,b} = \psi^r_{a,b} + \sqrt{-1}\psi^i_{a,b}, \tag{15}$$

where $\psi^r_{a,b}$ and $\psi^i_{a,b}$ are real wavelets. 2-D complex wavelets are also extended from 1-D complex wavelets via tensor product. As a result, though complex wavelet is an improvement of real complex wavelet, it still suffers from the inherent limitation of tensor product wavelets: inability to capture full direction information of one image.

2.2. Non-tensor product wavelet filter banks

In digital image processing, we hope analysis filter banks have the properties of multi-resolution and multi-direction. Generally, the analysis filter banks used in image processing are bivariate filter banks and wavelets. But most of the current bivariate filter banks and wavelets are constructed by the tensor products of univariate filters, which means the convolution of bivariate filter banks or wavelets and image is implemented by first convoluting the univariate filters with the rows of the image and then with the columns of the image. A result brought by construction of tensor product is the bivariate filter banks and wavelets can only capture limited image direction information, namely, vertical, horizontal and diagonal directions. Therefore, non-tensor product approaches for construction of bivariate filter banks and wavelets are desirable in order to develop the multi-direction analysis filter banks.

In our previous work [19,20], we proposed such non-tensor product wavelet filter banks that are expressed as

$$m_0(\xi,\eta) = \frac{1}{4}(1, e^{-i\xi}, e^{-i\eta}, e^{-i(\xi+\eta)}) \times \left(\prod_{k=1}^{N} B_{(\mu_k,v_k)} D(2\xi, 2\eta) B^T_{(\mu_k,v_k)}\right) V_0, \quad (\xi,\eta) \in R^2 \tag{16}$$

and

$$\begin{aligned} m_j(\xi,\eta) = {} & \frac{1}{4}(1, e^{-i\xi}, e^{-i\eta}, e^{-i(\xi+\eta)}) \\ & \times \left(\prod_{k=1}^{N} B_{(\mu_k,v_k)} D(2\xi, 2\eta) B^T_{(\mu_k,v_k)}\right) V_j, \\ & j = 1,2,3, \ (\xi,\eta) \in R^2, \end{aligned} \tag{17}$$

where m_0 is wavelet low-pass filter and $m_j, j = 1,2,3$, are wavelet high-pass filters.

In the following, we would like to provide two concrete examples of the non-tensor product wavelet filter banks. Setting the parameters $N = 1$ and $\{\mu = \pi/2, v = 0\}$, we get the following filter banks:

$$\begin{cases} m_0(x,y) = 1/8(1 + x + y - xy + x^2 + y^2 + x^2y + xy^2 - x^2y^2, \\ \qquad - x^3 - y^3 + x^3y + xy^3 + x^3y^2 + x^2y^3 + x^3y^3), \\ m_1(x,y) = 1/8(1 + x + y - xy + x^2 - y^2 + x^2y - xy^2 + x^2y^2, \\ \qquad - x^3 + y^3 + x^3y - xy^3 - x^3y^2 - x^2y^3 - x^3y^3), \\ m_2(x,y) = 1/8(1 + x + y - xy - x^2 + y^2 - x^2y + xy^2 + x^2y^2, \\ \qquad + x^3 - y^3 - x^3y + xy^3 - x^3y^2 - x^2y^3 - x^3y^3), \\ m_3(x,y) = 1/8(-1 - x - y + xy - x^2 + y^2 + x^2y + xy^2 + x^2y^2, \\ \qquad - x^3 - y^3 + x^3y + xy^3 - x^3y^2 - x^2y^3 - x^3y^3). \end{cases}$$

The filter banks can be represented in matrix form as follows:

$$m_0 = \frac{1}{8}\begin{pmatrix} 1 & 1 & 1 & -1 \\ 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ -1 & 1 & 1 & 1 \end{pmatrix},$$

$$m_1 = \frac{1}{8}\begin{pmatrix} 1 & 1 & 1 & -1 \\ 1 & -1 & 1 & 1 \\ -1 & -1 & 1 & -1 \\ 1 & -1 & -1 & -1 \end{pmatrix},$$

$$m_2 = \frac{1}{8}\begin{pmatrix} 1 & 1 & -1 & 1 \\ 1 & -1 & -1 & -1 \\ 1 & 1 & 1 & -1 \\ -1 & 1 & -1 & -1 \end{pmatrix},$$

$$m_3 = \frac{1}{8}\begin{pmatrix} -1 & -1 & 1 & -1 \\ -1 & 1 & 1 & 1 \\ 1 & 1 & 1 & -1 \\ -1 & 1 & -1 & -1 \end{pmatrix}.$$

Keeping the same μ and v, and setting $N = 2$, then the filter banks

$$\begin{cases} m_0(x,y) = 1/8(1 + x + y - xy + x^4 + y^4 + x^4y + xy^4 - x^4y^4, \\ \qquad - x^5 - y^5 + x^5y + xy^5 + x^5y^4 + x^4y^5 + x^5y^5), \\ m_1(x,y) = 1/8(1 + x + y - xy + x^4 - y^4 + x^4y - xy^4 + x^4y^4, \\ \qquad - x^5 + y^5 + x^5y - xy^5 - x^5y^4 - x^4y^5 - x^5y^5), \\ m_2(x,y) = 1/8(1 + x + y - xy - x^4 + y^4 - x^4y + xy^4 + x^4y^4, \\ \qquad + x^5 - y^5 - x^5y + xy^5 - x^5y^4 - x^4y^5 - x^5y^5), \\ m_3(x,y) = 1/8(-1 - x - y + xy - x^4 + y^4 - x^4y + xy^4 + x^4y^4, \\ \qquad + x^5 - y^5 - x^5y + xy^5 - x^5y^4 - x^4y^5 - x^5y^5). \end{cases}$$

The filter banks can be represented in matrix form as follows:

$$m_0=\frac{1}{8}\begin{pmatrix}1&1&0&0&1&-1\\1&-1&0&0&1&1\\0&0&0&0&0&0\\0&0&0&0&0&0\\1&1&0&0&-1&1\\-1&1&0&0&1&1\end{pmatrix},$$

$$m_1=\frac{1}{8}\begin{pmatrix}1&1&0&0&1&-1\\1&-1&0&0&1&1\\0&0&0&0&0&0\\0&0&0&0&0&0\\-1&-1&0&0&1&-1\\1&-1&0&0&-1&-1\end{pmatrix},$$

$$m_2=\frac{1}{8}\begin{pmatrix}1&1&0&0&-1&1\\1&-1&0&0&-1&-1\\0&0&0&0&0&0\\0&0&0&0&0&0\\1&1&0&0&1&-1\\-1&1&0&0&-1&-1\end{pmatrix},$$

$$m_3=\frac{1}{8}\begin{pmatrix}-1&-1&0&0&1&-1\\-1&1&0&0&1&1\\0&0&0&0&0&0\\0&0&0&0&0&0\\1&1&0&0&1&-1\\-1&1&0&0&-1&-1\end{pmatrix}.$$

Afterwards, we denote the non-tensor product wavelet filter banks as "NTPWFB".

By inputting different $\{\mu, v\}$, the NTPWFB can reflect different directions. Another property of NTPWFB is that it can better describe the features of one image than db wavelets. One example to prove this property of NTPWFB is given in Fig. 1. In Fig. 1, the Barbara image is decomposed by NTPWFB with $N=1$ and $\{\mu=0.78, v=1.05\}$, and db4 wavelet, respectively, at one level. From NTPWFB subbands, we can clearly see the facial features (such as the eyes, the nose, the mouse) of Barbara, and the edges, corners of the table, all of which reflect the essential content of this image and thus can play a determinate role to discriminate this image from other images. On the contrary, these features are blur in the db wavelet subbands. This fact means the NTPWFB subbands are capable of capturing more valuable features which are useful to discriminate different images.

3. Our algorithm for texture image retrieval

GGD model is one most efficient model for image retrieval with a few of parameters [21]. It has been proved by a large number of experimental works that the coefficients of wavelet high-pass subbands of texture images can be closely fitted by the GGD model [22]. But do the NTPWFB coefficients of texture images also own the same property? To answer this question, we decompose 40 texture images in MIT Vision Texture (VisTex) database using the NTPWFB with different parameters at varied levels and empirically find that the histograms of coefficients of NTPWFB high-pass subbands of texture images also satisfy the GGD distribution. Therefore, we can utilize the NTPWFB-based GGD model for texture image retrieval.

Our algorithm for texture image retrieval consists of two main steps—feature extraction: extracting features to fully represent one texture image using the NTPWFB-based GGD model; similarity measurement: using KLD to measure the similarity between extracted features of the query texture image and that of those texture images stored in the database.

The basic idea of feature extraction is to use the GGD model to approximate the distribution of coefficients in one NTPWFB subband and then take the parameter pair $\{\alpha, \beta\}$ of the GGD model as the features to represent this subband. The parameter pair $\{\alpha, \beta\}$ of all chosen NTPWFB subbands can be regarded as the features of one texture image. GGD model is given in Eq. (13).

We assume the coefficients of one NTPWFB high-pass subband Y compose one sequence X. According to Bayes rule, which is optimal in terms of retrieval error

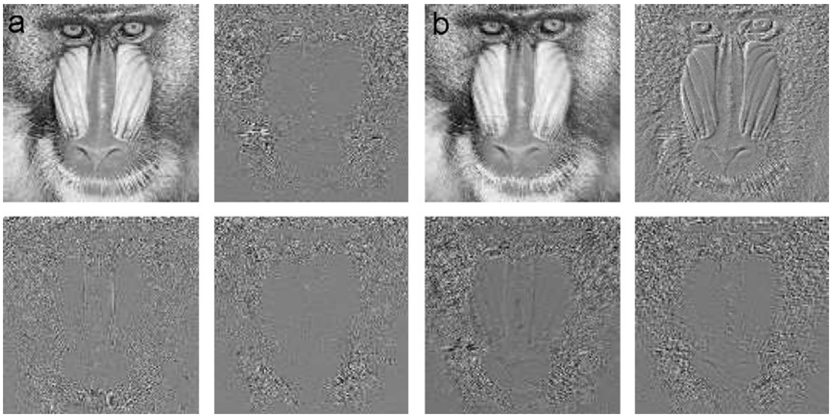

Fig. 1. Decomposing Baboon image using db4 wavelet and NTPWFB, respectively. (a) Decomposed subbands using db4 wavelet. (b) Decomposed subbands using NTPWFB.

probability, the estimated parameters $\{\hat{\alpha},\hat{\beta}\}$ must be chosen such that they maximize $P((\alpha,\beta)|X)$. Similarly, Bayes theorem dictates that it is equivalent to set $\{\hat{\alpha},\hat{\beta}\} = \text{argmax}_{\{\alpha,\beta\}} P((\alpha,\beta)|X)$. This is the maximum likelihood estimation (MLE) rule. Here, such an assumption is made: given an image I, the distribution of coefficients of a given NTPWFB subband is supposed to be independent from the distribution of coefficients of other NTPWFB subbands.

Since $X = (x_1,\ldots,x_L)$ is an independent and identically distributed (i.i.d) sequence, the likelihood function of GGD distribution is defined as

$$L(X;\alpha,\beta) = \log\prod_{i=1}^{L} p(x_i;\alpha,\beta). \tag{18}$$

According to the Lagrange optimization, the following likelihood equations are obtained [10,13]:

$$\sum_{1}^{L}\frac{\beta|x_i|^{\beta}a^{-\beta}}{\alpha} - \frac{L}{\alpha} = 0, \tag{19}$$

$$\frac{L\Psi(1/\beta)}{\beta^2} - \sum_{i=1}^{L}\left(\frac{|x_i|}{\alpha}\right)\log\left(\frac{|x_i|}{\alpha}\right) - \frac{L}{\beta} = 0, \tag{20}$$

where $\Psi(z) = \Gamma'^{z}/\Gamma(z)$. The roots of the two above likelihood equations, α and β, can be calculated out by iterative deduction using EM algorithm [23].

Texture image retrieval is a multiple hypothesis problem to find the N images to maximize $P(I_q|\theta_j)$, $1\leqslant j\leqslant N$, I_q is the query image, and θ_j is the hypothesis parameters. This problem is equivalent to minimizing the KLD between the two probability density functions (PDFs) $P(X|\theta_q)$ and $P(X|\theta_j)$ [21]. The KLD between two PDFs is defined as [13]

$$D(P(X|\theta_q)||P(X|\theta_j)) = \int P(x|\theta_q)\log\frac{P(x|\theta_q)}{P(x|\theta_i)}\,dx. \tag{21}$$

The model parameter set θ in GGD model is $\theta = \{\alpha,\beta\}$. The KLD between two GGDs can be calculated out by substituting (13) into (21). The KLD between two texture image I_1, I_2 is the sum of all the KLDs across all selected NTPWFB subbands.

4. Experiment

To fairly compare our method with other methods, we use the same texture images as those used in [13]. That is, 40 gray-scale textures with size 512 × 512 pixels in the MIT Vision Texture (VisTex) database (http://vismod.www.media.mit.edu) are used in our experiments. These texture images are listed in Fig. 2. Each texture image is divided into 16 non-overlapping subimages with size 128 × 128 pixels. Thus, totally 640 texture subimages are involved in our experiments. For the number of decomposition levels, our experiments agree with that the size of the smallest subimages should not be less than 16 × 16 pixels so that estimated energy values or model parameters would be robust, as was pointed out in [24]. Hence for the input image size of 128 × 128 pixels, a maximum of three levels of decomposition is chosen. In our experiment, we decompose the texture using the NTPWFB and db wavelet at 1–3 level. Our experimental works show that the retrieval result of 1 level decomposition is the worst, 2 level decomposition and 3 level decomposition achieve the nearly same result. The retrieval results shown in the following all are in the case of 2 level decomposition.

From a single texture subimage of size 128 × 128 pixels in the database, two GGD parameters $\{\alpha,\beta\}$ are estimated from each wavelet high-pass subband and each NTPWFB high-pass subband using the MLE described in Section 3. Thus, in the case of two level decomposition, totally 12 parameters are obtained, including six $\{\alpha\}$ and six $\{\beta\}$.

Following the evaluation criteria in [25], for each query subimage, the top S matches are retrieved. If $S<15$, the retrieval rate is the percentage of correct retrievals (the other 15 subimages of the same original texture) among the top S matches. Otherwise, the retrieval percentage is the ratio of the number of correct retrievals within the top S retrieval results to 15. For instance, in the case of $S = 5$, if four correct retrievals are ranked at the top five matches, then the retrieval rate equals to $\frac{4}{5}\times 100\% = 80\%$. In the case of $S = 15$, the retrieval rate is $\frac{10}{15}\times 100\% = 66.67\%$ if 10 correct retrievals are at the top 15 matches.

There are several methods (including wavelet-based energy, wavelet-based GGD, wavelet-based HMT, etc.) for texture image retrieval. But among them, which is the best one? Do and Vetterli [13] and Commowick et al. [21] offered answers to this question. In [13], Do et al. evaluated the performance of three kinds of wavelet-based energy method (L^1, L^2, L^1+L^2) and two kinds of wavelet-based GGD model (*GGD* and *KLD*, *GGD* and *ED*), and then concluded that the *GGD* and *KLD* always outperformed other four methods. In [21], Commowick tested energy-based method, GGD method, WD-HMT method on VisTex database, and found that the GGD method is the most efficient. Here, L^1 and L^2 stand for two energy measurement functions, and *ED* stands for the Euclidean distance function. Refer to [9,13,21] for the expressions of these functions.

Based on their conclusions, to show the efficiency of our method, here we only compared our method with the wavelet-based GGD and KLD method. The comparison of retrieval result is given in Table 1.

In the first row of Table 1, db wavelet means the db wavelet-based GGD and KLD method, $[\pi/2,0]$ means the NTPWFB with parameters $[\mu=\pi/2,\nu=0]$, $[\pi/2,0]$ and $[-\pi/2,0]$ means two NTPWFBs with parameters $[\mu=\pi/2,\nu=0]$ and $[\mu=-\pi/2,\nu=0]$, separately, which are combined to retrieve the texture image. Other parameters in the first row have the similar meanings. From this table, we can see that combining more NTPWFBs can achieve better retrievals, and the retrieval results of NTPWFBs with different parameters also differ. NTPWFB $[\pi/2,0]$ is somewhat better than NTPWFB $[\pi/6,2\pi/7]$ on retrieval.

Our observations in the experiments are given as follows.

(1) NTPWFB-based GGD model always outperforms the db wavelet-based GGD model. This is consistent with our expectation since NTPWFB can capture more

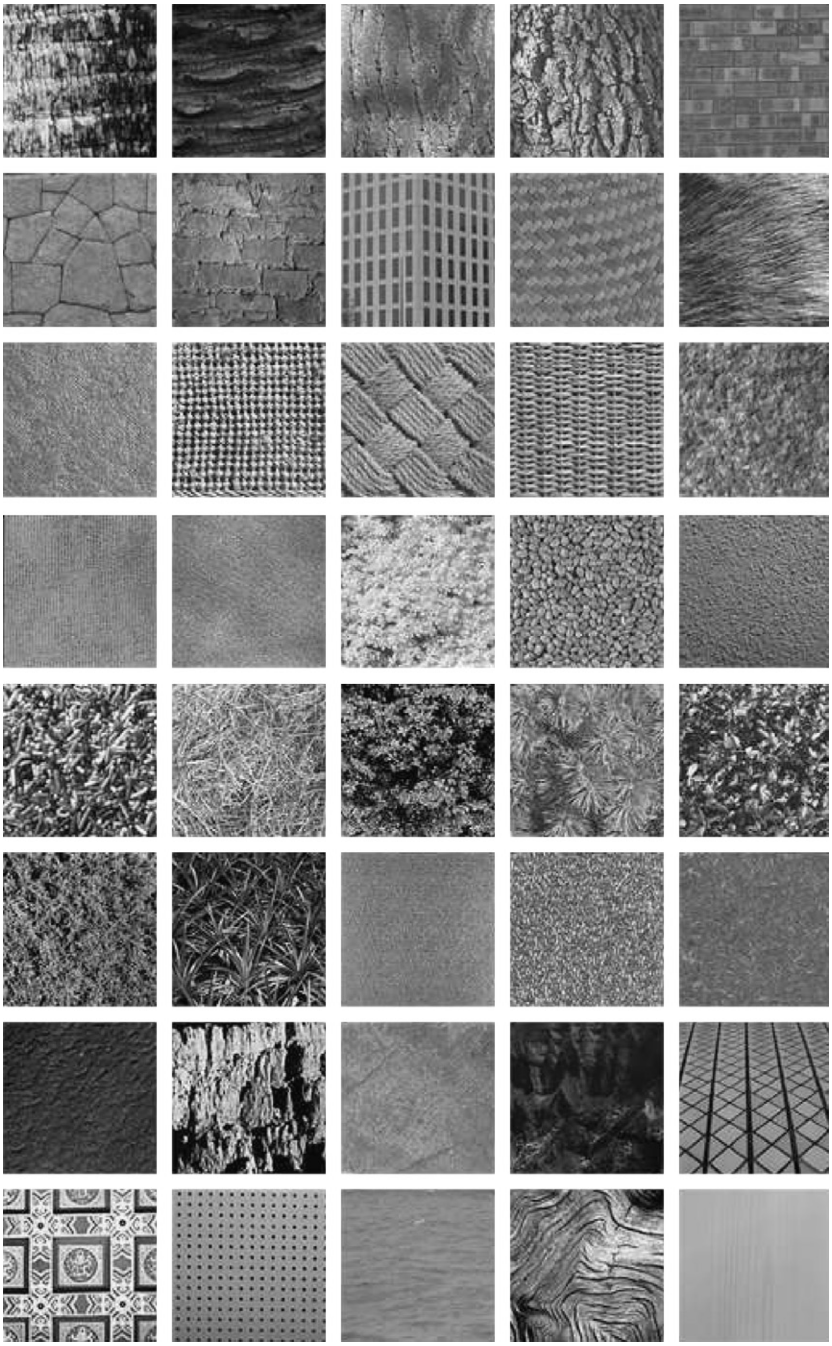

Fig. 2. Gray scale texture images from VisTex database used in our experiments. From left to right and top to bottom: Bark0, Bark6, Bark8, Bark9, Brick1; Brick4, Brick5, Building9, Fabric0, Fabric4; Fabric7, Fabric9, Fabric11, Fabric14, Fabric15; Fabric17, Fabric18, Flowers5, Food0, Food5; Food8, Grass1, Leaves8, Leaves10, Leaves11; Leave12, Leave16, Metal0, Metal2, Misc2; Sand0, Stone1, Stone4, Terrain10, Tile1; Tile4, Tile7, Water5, Wood1, Wood2.

direction information of texture image. As we know, orientation information is the essential feature of texture image, and many methods for texture image retrieval are based on the idea to capture the direction information.

(2) In our database, most of the texture discrimination information exist in the first two scales of NTPWFB decomposition since there is little improvement in retrieval rates when we increased from two to three levels of decomposition. Db wavelet also has the same property.

(3) Combining NTPWFBs with different parameters can achieve a better retrieval accuracy while at the same time increase the computing cost. Therefore, it is wise

Table 1
Texture image average retrieval rate (%).

Number of top matches	Db wavelet	$[\pi/2,0]$	$[\pi/6,2\pi/7]$	$[\pi/2,0]$ and $[-\pi/2,0]$	$[\pi/2,0]$ and $[-\pi/2,0]$ and $[\pi/6,2\pi/7]$ and $[2\pi/7,2\pi/3]$
3	90.8	95.2	93.9	96.8	97.6
5	87.6	92.6	90.5	94.4	95.0
7	84.6	88.3	85.8	90.2	91.6
10	80.5	83.3	81.0	85.3	86.2
15	71.7	75.2	72.8	77.5	79.5
20	80.2	82.7	81.1	84.3	86.1
50	91.7	94.5	92.6	95.2	96.3

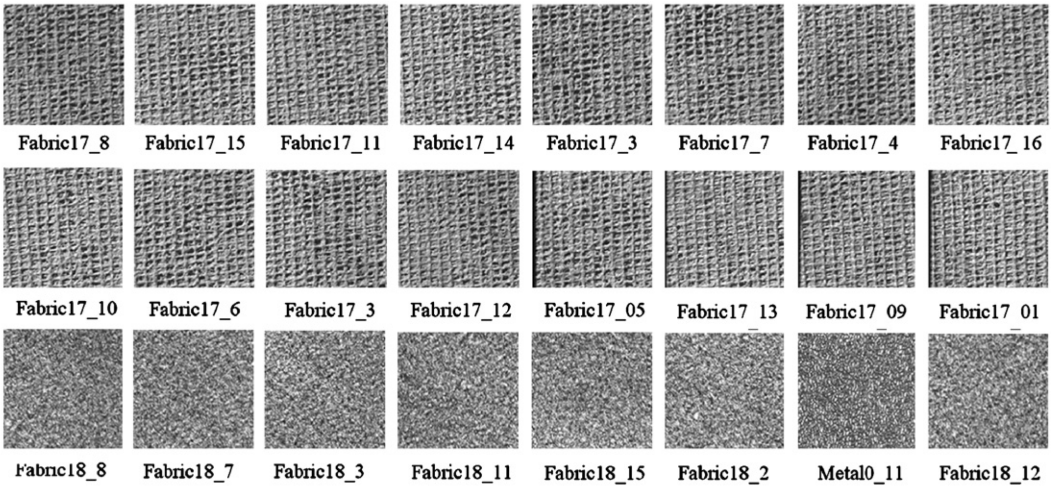

Fig. 3. Example 1 of retrieval result using NTPWFB-based GGD model. The query image "*Fabric*17_8" is on the top left corner and retrieved images are ranked from left to right, from top to bottom.

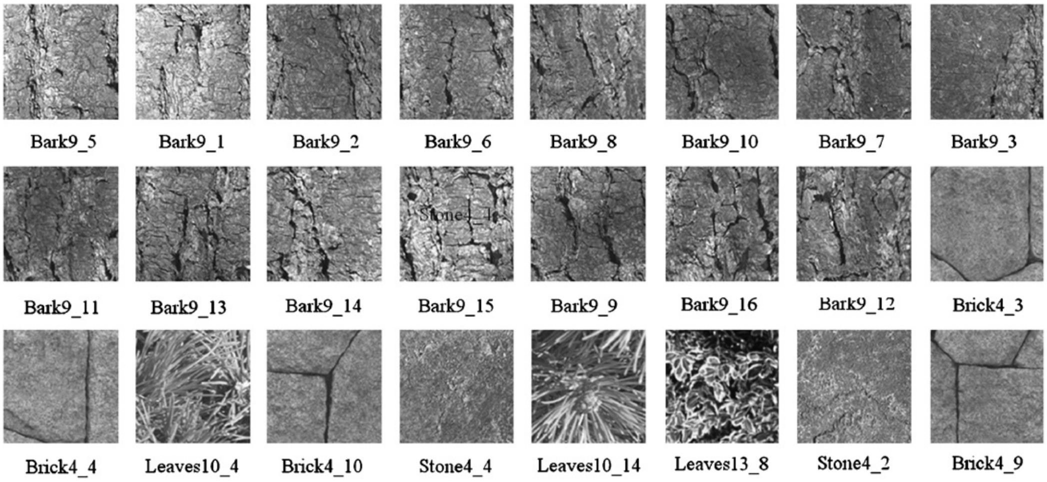

Fig. 4. Example 2 of retrieval result using NTPWFB-based GGD model.

to make a balance between the retrieval accuracy and computing cost.

Two examples of image retrieval using NTPWFB-based GGD model are shown in Figs. 3 and 4. In Fig. 3 the query image is "*Fabric*17_8" (the denotation means this subimage belongs to "*Fabric*17" and its index is 8), and all other 15 subimages of the same texture image "*Fabric*17" are correctly ranked at the top 15, and other retrieved subimages belong to textures "*Metal*0" and "*Fabric*18". It is very clear that these three textures are very similar. In Fig. 4, the query image is "*Bark*9_5", almost all relevant subimages are retrieved at the top 15 except for the subimage "*Bark*9_4". Though this retrieval result is not

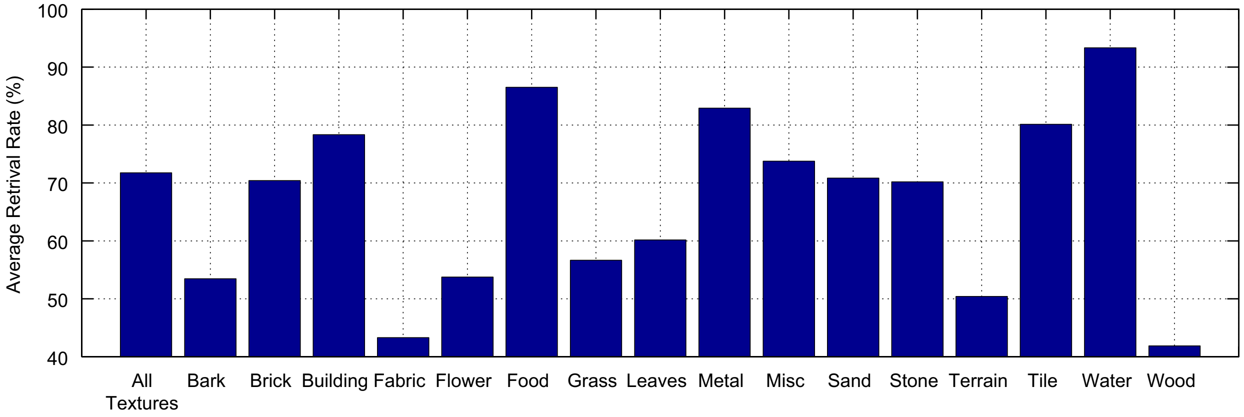

Fig. 5. Average retrieval rates (retrieving top 15 matches) for different texture classes using db wavelet-based GGD model.

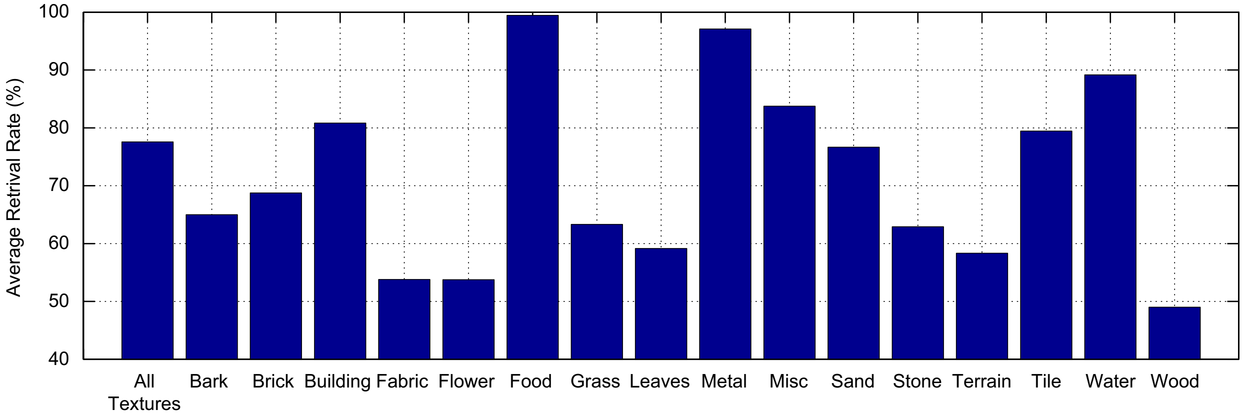

Fig. 6. Average retrieval rates (retrieving top 15 matches) for different texture classes using NTPWFB-based GGD model combining two parameters pairs $\{\pi/2, 0\}$ and $\{-\pi/2, 0\}$.

perfect, it still shows that our retrieval technology is highly effective and satisfactory.

Table 1 only provides the average retrieval rate of all 40 textures. Naturally, the correct retrieval accuracy of different texture classes may be higher or lower than the average accuracy. Figs. 5 and 6 provide the retrieval rate of different kinds of texture using db wavelet-based GGD model and NTPWFB-based GGD model, respectively. For example, the retrieval rate of "*Bark*" is the average of four texture images "*Bark*0", "*Bark*6", "*Bark*8" and "*Bark*9" which belongs to a large class "*Bark*". Based on this figure, we can more clearly see NTPWFB-based GGD model outperforms the db wavelet-based GGD model in almost all texture classes.

5. Conclusions

Wavelet-based GGD model is one efficient method for texture retrieval while traditional wavelets (tensor product wavelets, such as db wavelets) are not good at capture multi-direction information, which is one most important texture feature. Compared to the traditional wavelet, the NTPWFB can capture more direction information of texture image and thus can better reflect the properties of the texture images. In addition, the NTPWFB can better capture the edges of one image, as is also helpful to discriminate different images. What is more, through experimental results on 640 texture images of 40 classes from the VisTex database, we empirically find that coefficients of NTPWFB high-pass subbands of texture images, similar to the wavelet coefficients, can also be well fitted by the GGD models. Based on this discovery, we propose a novel NTPWFB-based GGD model for texture image retrieval. Experiment results on 640 texture images indicated that our new method significantly improve the retrieval rates. In this paper, we only investigate the global feature of the texture images. In fact, the local features are also important to reflect the properties of texture images, especially for the texture images with regular structure. To well capture more texture features and further improve the retrieval rate, an approach combining the global and local features will be considered in our further work.

Acknowledgments

This work was supported by the Grants 60803056, 60773187 from the NSFC, NCET-07-0338 from the Ministry of Education, and the Grants 2006ABA023, 2007CA011 and 2007ABA036 from the Department of Science and Technology in Hubei province, China. In addition, the authors would like to thank the anonymous reviewers for their perceptive comments, which have significantly improved the paper.

References

[1] Y. Rui, Image retrieval: current techniques, promising directions, and open issues, Journal of Visual Communication and Image Representation 10 (1999) 39–62.

[2] M. Tuceryan, A.K. Jain, The Handbook of Pattern Recognition and Computer Vision, second ed., World Scientific Publishing Co., Singapore, 1998.

[3] H. Derin, H. Elliott, Modelling and segmentation of noisy and textured images using Gibbs random fields, IEEE Transaction on Pattern Recognition and Machine Intelligence 9 (1987) 39–55.

[4] G.C. Cross, A.K. Jain, Markov random field texture models, IEEE Transaction on Pattern Recognition and Machine Intelligence 5 (1983) 25–39.

[5] J. Daugman, Gabor wavelets and statistical pattern recognition, The Handbook of Brain Theory and Neural Networks, second ed., MIT Press, Cambridge, 2002, pp. 457–463.

[6] D. Tao, X. Tang, X. Li, Which components are important for interactive image searching?, IEEE Transactions on Circuits and Systems for Video Technology 18 (1) (2008) 3–11.

[7] J. Li, N. Allinson, D. Tao, X. Li, Multitraining support vector machine for image retrieval, IEEE Transactions on Image Process 15 (11) (2006) 3597–3601.

[8] X. Li, S. Lin, S. Yan, D. Xu, Discriminant locally linear embedding with high-order tensor data, IEEE Transactions on Systems, Man, and Cybernetics Part B 38 (2) (2008) 342–352.

[9] H.E.S. Said, T. Tan, K. Baker, Writer identification based on handwriting, Pattern Recognition 33 (1) (2000) 133–148.

[10] Z. He, X. You, Y.Y. Tang, Writer identification using global wavelet-based features, Neurocomputing 71 (2008) 1832–1841.

[11] S. Mallat, W. Hwang, Singularity detection and processing with wavelets, IEEE Transactions on Information Theory 38 (2) (1992) 617–643.

[12] S. Mallet, A Wavelet Tour of Signal Processing, Academic Press, San Diego, 1999.

[13] M.N. Do, M. Vetterli, Wavelet-based texture retrieval using generalized Gaussian density and Kullback–Leibler distance, IEEE Transactions on Image Processing 11 (2) (2002) 146–158.

[14] P. Moulin, J. Liu, Analysis of multiresolution image denoising schemes using generalized Gaussian and complexity priors, IEEE Transactions on Information Theory 45 (1999) 909–919.

[15] M. Crouse, R.D. Nowak, R.G. Baraniuk, Wavelet-based signal processing using hidden Markov model, IEEE Transactions on Signal Processing (Special Issue on Wavelets and Filter banks) (1998) pp. 886–902.

[16] S. Mallat, S. Zhong, Characteristics of signals from multiscale images, IEEE Transactions on Pattern Analysis and Machine Intelligence 14 (1) (1992) 710–732.

[17] Z. He, X. You, Y.Y. Tang, Writer identification of Chinese handwriting documents using hidden Markov tree model, Pattern Recognition 41 (2008) 1295–1307.

[18] M.N. Do, Fast approximation of Kullback–Leibler distance for dependence trees and hidden Markov models, IEEE Signal Processing Letters 10 (2003) 115–118.

[19] Z. He, X. You, Y.Y. Tang, P. Wang, Y. Xue, Texture image retrieval using novel non-separable filter banks based on centrally symmetric matrices, in: Proceedings of 18th International Conference on Pattern Recognition, August 2006, pp. 161–164.

[20] X. You, Q. Chen, Construction of non-tensor product wavelet and its application, Technical Report, Department of Computer Science, Hong Kong Baptist University, 2007.

[21] O. Commowick, C. Lenglet, C. Louchet, Wavelet-based texture classification and retrieval, Technical Report, April 2003 ⟨http://www.tsi.enst.fr/fsi/enseignement/ressources/mti/classif-textures/ReportFinal.html⟩.

[22] G.V. Wouwer, P. Scheunders, D.V. Dyck, Statistical texture characterization from discrete wavelet representations, IEEE Transactions on Image Processing 8 (4) (1999) 592–598.

[23] R. Redner, H. Walker, Mixture densities, maximum likelihood and the EM algorithm, SIAM Review 26 (2) (1984).

[24] T. Chang, C.C.J. Kuo, Texture analysis and classification with tree-structure wavelet transform, IEEE Transactions on Image Processing 2 (4) (1985) 429–441.

[25] B.S. Manjunath, W.Y. Ma, Texture features for browsing and retrieval of image data, IEEE Transactions on Pattern Recognition and Machine Intelligence 18 (1996) 837–842.

[26] N.G. Kingsbury, Complex wavelets for shift invariant analysis and filtering of signals, Journal of Applied and Computational Harmonic Analysis 10 (3) (2001) 234–253.

[27] M. Kokare, P.K. Biswas, B.N. Chatterji, Texture image retrieval using new rotated complex wavelet filters, IEEE Transaction on Systems, Man, and Cybernetics Part B 35 (6) (2005) 1168–1178.

[28] W.I. Selesnick, R.G. Baraniuk, N.G. Kingsbury, The dual-tree complex wavelet transform, IEEE Signal Processing Magazine 22 (6) (2005) 123–151.