Bayesian texture classification and retrieval based on multiscale feature vector

Turgay Celik, Tardi Tjahjadi
School of Engineering, University of Warwick,
Gibbet Hill Road, Coventry, CV4 7AL, United Kingdom.
Email: t.tjahjadi@warwick.ac.uk

Abstract

This paper proposes a supervised multiscale Bayesian texture classifier. The classifier exploits the dual-tree complex wavelet transform (DT-CWT) to obtain complex-valued multiscale representations of training texture samples for each texture class. The high-pass subbands of DT-CWT decomposition of a texture image are used to form a multiscale feature vector representing magnitude and phase features. For computational efficiency, the dimensionality of feature vectors is reduced using principal component analysis (PCA). The class conditional probability density function of low-dimensional feature vectors for each texture class is then estimated by using Parzen-window estimate with identical Gaussian kernels and is used to represent the texture class. A query texture image is classified as the corresponding texture class with the highest a posteriori probability according to a Bayesian inferencing. The superior performance and robustness of the proposed classifier is demonstrated for classifying texture images from image databases. The proposed multiscale texture feature vector extracted from both magnitude and phase of DT-CWT subbands of a query image is also shown to be effective for texture retrieval.

Keywords:
Texture classification, texture retrieval, multiscale analysis, feature extractiondiscrete wavelet transform, dual-tree complex wavelet transform, Parzen-window estimate, Bayesian classification, principal components analysis, dimensionality reduction.

1. Introduction

There has been significant accumulation of visual information in large digital databases in the past decades, making digital image libraries widely used. To improve the management of these collections, it is necessary to have effective and efficient methods to search for specific images. For this purpose, content-based image retrieval (CBIR) from unannotated image databases has been gaining interest of the research community.

There are two main processes in a CBIR system: feature extraction and similarity measurement. In the first process, a set of features such as shape, texture and colour, which constitutes the image signature is generated to represent the content of a given image. The set has to be much smaller in size than the

original image while capturing as much of the image information as possible. In the present work, texture information is used as a feature for representing the content of an image. The second process requires a distance measure to determine how similar each image in the database is to a query image.

The most important function of a CBIR system, where texture is used to represent the content of an image, is the ability to classify texture. For texture classification, Gabor filters, wavelet transforms and finite impulse response filters have been widely used. Gabor filters are appealing because of their simplicity and support from neurophysiological experiments (Faugeras, 1978). They have been used for texture segmentation despite being based on texture reconstruction (Jain and Farrokhnia, 1991), (Arivazhagan et al., 2006b). A general filter bank is often too large since it is designed to capture general texture properties. However, textures can be classified using a small set of filters, which gives rise to the filter selection problem. For example, a neural network system has been used to select a minimum set of Gabor filters for texture discrimination while keeping the classification at an acceptable level compared to the case without filter selection (Jain and Karu, 1996). In these filtering methods, texture images are usually decomposed into several feature images through projection via a set of selected filters. These filters are often based on representation such that textures are reconstructed with the minimum information loss. Our proposed approach extracts features that maximise the separation or discrimination among different textures. The wavelet based texture classifiers are similar to Gabor based methods with the Gabor filters replaced by the Discrete Wavelet Transform (DWT) (Arivazhagan and Ganesan, 2003a), (Arivazhagan and Ganesan, 2003b), (Muneeswarana et al., 2005), (Kim and Kang, 2007), (Kokare et al., 2007), (Hiremath and Shivashankar, 2008). Since DWT is shift variant, a shift in the signal degrades the performance of DWT based classifiers. An extensive set of texture features extracted from the ridgelet transform (Arivazhagan et al., 2006a) has also been used for texture classification achieving a higher classification performance than using features from DWT (Arivazhagan and Ganesan, 2003a). Apart from the mean and standard deviation of the ridgelet transform subbands, co-occurrence features were extracted in order to increase the correct classification rate (Arivazhagan et al., 2006a).

DWT has also been applied for texture retrieval. An energy and co-occurrence (Haralick et al., 1973) based signature was used for texture retrieval in (Wouwer et al., 1999). Statistics of the wavelet coefficients are used to extract two feature sets: (1) the wavelet histogram signatures which capture all first order statistics using a model based approach; and (2) the wavelet co-occurrence signatures which reflect the second-order statistics of the coefficients. The generalized Gaussian distribution (GGD) signature was used in (Do and Vetterli, 2002). The GGD and Kullback-Leibler distance metrics have been used in the DWT domain. The similarity measure and feature extraction are jointly considered for the estimation and detection in a maximum likelihood framework, providing a definition for similarity measurement using Kullback-Leibler divergence (KLD). The performance of the texture retrieval system then depends on modelling the marginal distribution of wavelet coefficients using GGD and on the existence of a closed form for the KLD between

GGDs. Recently, two-dimensional (2-D) rotated wavelet filters that are non-separable and oriented were used to improve the texture retrieval performance of the standard DWT (Kokare et al., 2007), via improved characterisation of diagonally oriented textures. However, DWT is not invariant to shift and lacks direction selectivity (Kingsbury, 1999), which is a major obstacle for robust feature representation. As a result Gabor wavelets have been designed to be directionally selective. They are invariant to shift since they are non-decimated, but they are therefore overcomplete and hence computationally expensive. The dual-tree complex wavelet transform (DT-CWT) has been shown to be approximately shift invariant and has limited redundancy (Kingsbury, 1999).

The afore-mentioned advantages of DT-CWT and its multiscale structure make it appealing for texture classification and retrieval. In our recent work, we used DT-CWT subbands to design a multiscale texture classifier (Celik and Tjahjadi, 2009). The classifier uses simple statistical features of the mean and standard deviation of the magnitude of DT-CWT subbands in different scales. The multiscale feature vector extracted from a query images is used together with simple distance measure to perform the final classification. In this paper, we improve the performance of our previous work by using both magnitude and phase of the complex subbands of DT-CWT, and a Bayesian inferencing. The extra information provided by the phase combined with the magnitude of the complex subbands results in more discriminative feature vectors. The standard deviation and energy of DT-CWT subbands are used to create a multiscale feature vector for each texture image which consists of features extracted from phase and magnitude of DT-CWT subband coefficients. The dimensionality of multiscale feature vectors is reduced using principal component analysis (PCA) to benefit from computational efficiency. The class conditional probability density function of the low-dimensional feature vectors is estimated by using the Parzen-window estimate with identical Gaussian kernels (Fukunaga and Hayes, 1989). A query texture image is classified as the corresponding texture class with the highest a posteriori probability according to the Bayesian inferencing. The new classifier is applied to supervised texture classification. The proposed multiscale texture feature vector extracted from both magnitude and phase of DT-CWT subbands of a query image is also used in texture retrieval.

The paper is organized as follows. Section 2 presents DT-CWT. The Bayesian classification is presented in Section 3. Section 4 presents the proposed multiscale texture classifier, the learning and classification of texture features for different texture classes, and the process of texture retrieval. The experimental results and discussions are presented in Section 5. Finally, Section 6 concludes the paper.

2. Dual-tree complex wavelet transform

The DWT is not shift invariant due to the decimation during the transform. A small shift in the input signal generates very different wavelet coefficients. The DT-CWT (Kingsbury, 1999) exhibits approximate shift invariance and improved directional resolution. It achieves perfect reconstruction and good frequency

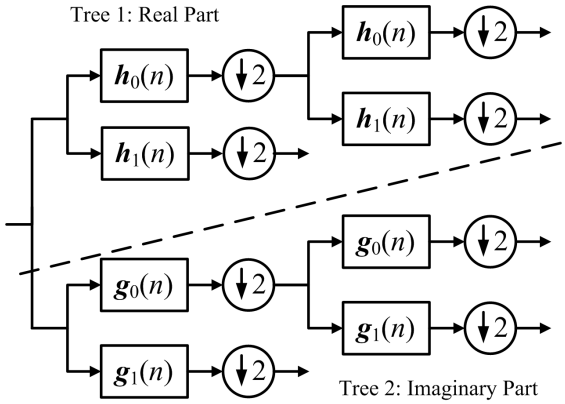


Figure 1: Implementation of the two levels 1-D DT-CWT using two filter banks operating as two parallel trees on the same data, where $h_i(n)$ and $g_i(n)$ are wavelet filters. The outputs from the upper and lower trees are interpreted as the real and imaginary parts of the DT-CWT coefficients, respectively.

characteristics using two parallel fully decimated trees with real coefficients. The one-dimensional (1-D) DT-CWT decomposes a signal $f(x)$ in terms of a shifted and dilated complex mother wavelet $\psi(x)$ and scaling function $\phi(x)$, i.e.,

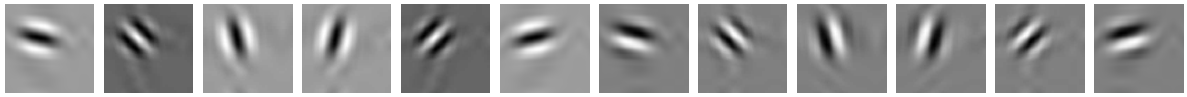
$$f(x) = \sum_{l \in Z} s_{j_0,l}\phi_{j_0,l}(x) + \sum_{j \geq j_0} \sum_{l \in Z} c_{j,l}\psi_{j,l}(x), \tag{1}$$

where $s_{j_0,l}$ is a scaling coefficient and $c_{j,l}$ is a complex wavelet coefficient with $\phi_{j_0,l}(x) = \phi^r_{j_0,l}(x) + \sqrt{-1}\phi^i_{j_0,l}(x)$, and $\psi_{j,l}(x) = \psi^r_{j,l}(x) + \sqrt{-1}\psi^i_{j,l}(x)$. The complex wavelet transform is a combination of two real wavelet transforms. In 1-D the $\{\phi^r_{j_0,l}, \phi^i_{j_0,l}, \psi^r_{j_0,l}, \psi^i_{j_0,l}\}$ form a tight wavelet frame with two times redundant. The real and imaginary parts of the DT-CWT are computed using separate filter bank structures (operating as two parallel trees) with wavelet filters h_0 and h_1 for the real part, and g_0 and g_1 for the imaginary part. 1-D DT-CWT is implemented using two filter banks in parallel operating on the same data as illustrated in Figure 1 (Kingsbury, 1999). The outputs from the two trees are interpreted as the real and imaginary parts of the DT-CWT coefficients.

Similar to 1-D DT-CWT, 2-D DT-CWT decomposes a 2-D image $f(x, y)$ using dilation and translation of a complex scaling function and six complex wavelet functions ψ^θ, i.e.,

$$f(x,y) = \sum_{l \in Z^2} s_{j_0,l}\phi_{j_0,l}(x,y) + \sum_{b \in \theta} \sum_{j \geq j_0} \sum_{l \in Z^2} c^\theta_{j,l}\psi^\theta_{j,l}(x,y), \tag{2}$$

where $\theta = \{\pm 15^\circ, \pm 45^\circ, \pm 75^\circ\}$ refer to the directionality of complex wavelet subbands. The impulse responses of the six complex wavelets associated with 2-D DT-CWT are illustrated in Figure 2. The frequency-partition of the DT-CWT resulting from two levels decomposition is shown in Figure 3. It shows that the DT-CWT can discriminate between features at positive and negative frequencies. Hence, there are six subbands characterising features along lines at angles of $\theta = \{\pm 15^\circ, \pm 45^\circ, \pm 75^\circ\}$.



(a) (b) (c) (d) (e) (f) (g) (h) (i) (j) (k) (l)

Figure 2: The real and imaginary parts of the impulse response of the DT-CWT filters for the 6 directional subbands: (a) real, $-15°$; (b) real, $-45°$; (c) real, $-75°$; (d) real, $75°$; (e) real, $45°$; (f) real, $15°$; (g) imaginary, $-15°$; (h) imaginary, $-45°$; (i) imaginary, $-75°$; (j) imaginary, $75°$; (k) imaginary, $45°$; and (l) imaginary, $15°$.

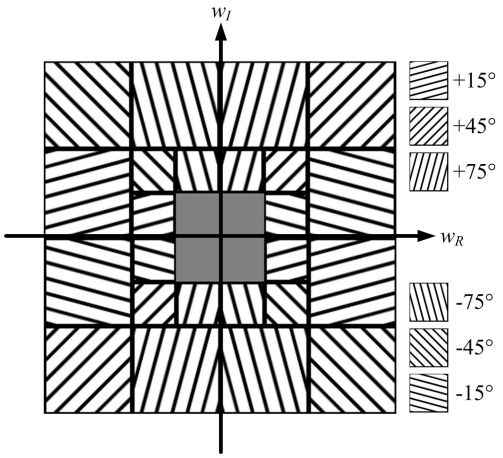


Figure 3: Frequency-domain partition resulted from a two-level 2-D DT-CWT decomposition, where w_R and w_I are the real axis and the imaginary axis of the complex frequency domain, respectively.

3. Bayesian Classification

Bayesian classification and decision making are based on probability theory and the principle of choosing the most probable or the lowest risk option (Duda et al., 2000). Let $\mathbf{x} = [x_1, x_2, \ldots, x_D]$ be a feature vector of dimensionality D. The probability (or posteriori probability) that a feature vector $\mathbf{x}$ belongs to texture class w_k is $P(w_k|\mathbf{x})$. The posterior probabilities can be computed using the a priori probabilities in the framework of Bayes formula, i.e.,

$$P(w_k|\mathbf{x}) = \frac{p(\mathbf{x}|w_k)\,P(w_k)}{p(\mathbf{x})}, \tag{3}$$

where $p(\mathbf{x}|w_k)$ and $P(w_k)$ are the probability density function and a priori probability of class w_k, respectively. The factor in the denominator is a normalization factor used to ensure that the weighted sum of $p(\mathbf{x}|w_k)$'s for a training database is one, i.e.,

$$p(\mathbf{x}) = \sum_{k=1}^{K} p(\mathbf{x}|w_k)\,P(w_k)\,, \tag{4}$$

where K is the total number of different texture classes.

The feature vector $\mathbf{x}$ is assigned to the texture class w_k with the highest a posteriori probability which produces the minimum error probability (Duda et al., 2000), i.e., $P(w_k|\mathbf{x}) > P(w_i|\mathbf{x}), \forall i \in \{1, 2, \ldots, K\}, i \neq k$ which can be reformulated using (3) as follows:

$$\begin{aligned} w_k &= \underset{w_i=\{w_1,w_2,\ldots,w_K\}}{\operatorname{argmax}} P(w_i|\mathbf{x}) \\ &= \underset{w_i=\{w_1,w_2,\ldots,w_K\}}{\operatorname{argmax}} \frac{p(\mathbf{x}|w_i)\,P(w_i)}{p(\mathbf{x})} \\ &= \underset{w_i=\{w_1,w_2,\ldots,w_K\}}{\operatorname{argmax}} p(\mathbf{x}|w_i)\,P(w_i)\,. \end{aligned} \tag{5}$$

The classification problem reduces to the estimation of $p(\mathbf{x}|w_k)$ and $P(w_k)$ for each texture class w_k. Without losing generality, one can assume that the a priori probabilities of the texture classes are equal, which further simplifies the problem to the estimation of the class conditional probability density function $p(\mathbf{x}|w_k)$ of each texture class w_k in the training set.

4. Proposed method

The DT-CWT decomposition of an $W \times H$ image I results in a decimated dyadic decomposition into $s = 1, 2, \ldots, S$ scales, where each scale is of size $W/2^s \times H/2^s$. Each decimated scale has a set C_s of 6 subbands of complex coefficients, denoted as $C_s = \{\alpha_1^{(s)} e^{i\theta_1^{(s)}}, \ldots, \alpha_6^{(s)} e^{i\theta_6^{(s)}}\}$, that correspond to responses of the 6 subbands respectively orientated at $-15°$, $-45°$, $-75°$, $15°$, $45°$, and $75°$. For each subband orientation at scale s there are two responses: magnitude ($\alpha_i^{(s)}$) and phase ($\theta_i^{(s)}$) responses, $i = 1, \ldots, 6$. Each response is a 2-D data of size $W/2^s \times H/2^s$.

4.1. Feature extraction

Texture features are extracted using image statistics. A feature vector is formed using the energy and standard deviation of every subband. The basic assumption of using energy as a feature for texture discrimination is that the energy distribution in the frequency domain identifies a texture. An energy based approach is partly supported by physiological studies of the visual cortex (Daugman, 1980).

The standard deviation $M_1(r)$ and energy $M_2(r)$ of wavelet subband $r, r \in \{\alpha_i^{(s)}, \theta_i^{(s)}\}$, are defined as follows:

$$M_1(r) = \sqrt{\frac{2^{2s}}{HW} \sum_{y=1}^{H/2^s} \sum_{x=1}^{W/2^s} \left(r\left(x,y\right) - \mu\left(r\right)\right)^2} \tag{6}$$

$$M_2(r) = \frac{2^{2s}}{HW} \sum_{y=1}^{H/2^s} \sum_{x=1}^{W/2^s} \left|r\left(x,y\right)\right| \tag{7}$$

where

$$\mu(r) = \frac{2^{2s}}{HW} \sum_{y=1}^{H/2^s} \sum_{x=1}^{W/2^s} r\left(x,y\right). \tag{8}$$

Using $M_k(r)$, $k \in \{1,2\}$, the following feature vectors are defined for scale s using the set C_s:

$$\mathbf{F}_{M_k,\alpha,s} = \left[M_k(\alpha_1^{(s)}) M_k(\alpha_2^{(s)}) \ldots M_k(\alpha_5^{(s)}) M_k(\alpha_6^{(s)})\right] \tag{9}$$

$$\mathbf{F}_{M_k,\theta,s} = \left[M_k(\theta_1^{(s)}) M_k(\theta_2^{(s)}) \ldots M_k(\theta_5^{(s)}) M_k(\theta_6^{(s)})\right] \tag{10}$$

$$\mathbf{F}_{\alpha,s} = \left[\frac{\mathbf{F}_{M_1,\alpha,s}}{\|\mathbf{F}_{M_1,\alpha,s}\|} \frac{\mathbf{F}_{M_2,\alpha,s}}{\|\mathbf{F}_{M_2,\alpha,s}\|}\right] \tag{11}$$

$$\mathbf{F}_{\theta,s} = \left[\frac{\mathbf{F}_{M_1,\theta,s}}{\|\mathbf{F}_{M_1,\theta,s}\|} \frac{\mathbf{F}_{M_2,\theta,s}}{\|\mathbf{F}_{M_2,\theta,s}\|}\right] \tag{12}$$

$$\mathbf{F}_{s} = \left[\frac{\mathbf{F}_{\alpha,s}}{\|\mathbf{F}_{\alpha,s}\|} \frac{\mathbf{F}_{\theta,s}}{\|\mathbf{F}_{\theta,s}\|}\right]. \tag{13}$$

where $\|.\|$ is the second norm (i.e., the signal energy). Since the numerical ranges of the feature vectors extracted from magnitude and phase responses are not equal, the feature vectors $\mathbf{F}_{\alpha,s}$, $\mathbf{F}_{\theta,s}$ and $\mathbf{F}_s$ are normalised as in (11), (12) and (13), respectively to ensure that the contributions of the magnitude and phase responses to the feature vectors are equally weighted.

Given an image I, its multiscale feature vector is extracted by combining different realizations of (13) for different values of scale s, i.e.,

$$\begin{aligned} \mathbf{F}_S^I &= \frac{[\mathbf{F}_1\ \mathbf{F}_2\ \cdots\ \mathbf{F}_S]}{\|[\mathbf{F}_1\ \mathbf{F}_2\ \cdots\ \mathbf{F}_S]\|} \\ &= \left[\ f_{S,1}^I\ f_{S,2}^I\ f_{S,3}^I\ \cdots\ f_{S,24S-2}^I\ f_{S,24S-1}^I\ f_{S,24S}^I\ \right], \end{aligned} \tag{14}$$

where $\mathbf{F}_S^I$ is a vector of $24S$ elements.

4.2. Texture learning and classification

In the supervised texture learning stage, each texture class w_k is modelled using a probability density function $p(\mathbf{x}|w_k)$. For this, texture features extracted from texture samples using (14) are used.

Let us assume that each training set for texture class w_k consists of N texture samples. N feature vectors for a texture class w_k are used to create a class conditional probability density function $p(\mathbf{x}|w_k)$ using the Parzen-window estimate (Fukunaga and Hayes, 1989). The dimensionality of the texture feature vector $\mathbf{F}_S^{I^{(n)}}$ extracted from the n^{th} texture sample $I^{(n)}$ is $24S$, and increases with an increase in S. In order to benefit from low computational cost, the dimensionality of each feature vector $\mathbf{F}_S^{I^{(n)}}$ is reduced from $24S$ to D by using PCA. For ease of mathematical notation, $\mathbf{x}_n$ is used to denote the vector $\mathbf{F}_S^{I^{(n)}}$.

The set of vectors $\{\mathbf{x}_n\}_{n=1}^N$ is used to create an eigenvector space using PCA (Gonzalez and Woods, 2006). The average vector of the set is defined by

$$\mathbf{\Psi} = \frac{1}{N}\sum_{n=1}^{N}\mathbf{x}_n. \tag{15}$$

Each feature vector differs from the average vector by the vector $\mathbf{\Delta}^{(n)} = \mathbf{x}_n - \mathbf{\Psi}$. PCA is applied to the feature vector set $\mathbf{\Delta}^{(n)}$ to determine a set of $24S$ orthonormal vectors $\mathbf{e}_d$ and their associated scalars λ_d, $d = 1, 2, \ldots, 24S$, which best describe the distribution of the data. The vectors $\mathbf{e}_d$ and corresponding scalars λ_d are the eigenvectors and eigenvalues, respectively, of the covariance matrix

$$\mathbf{C} = \frac{1}{N}\sum_{n=1}^{N}\mathbf{\Delta}^{(n)^T}\mathbf{\Delta}^{(n)}, \tag{16}$$

where T is the transpose operator. The matrix $\mathbf{C} \in \mathbb{R}^{24S\times 24S}$ determines $24S$ eigenvectors and eigenvalues, where the eigenvectors are sorted in descending order with respect to their eigenvalues, i.e., $\lambda_d \geq \lambda_{d+1}$.

The dimensionality of the feature vector space is then reduced by projecting each feature vector $\mathbf{x}_n$ onto the eigenvector space spanned by the D largest eigenvectors of (16), i.e., the feature vector $\mathbf{x}_n \in \mathbb{R}^{24S}$ is transformed into $\hat{\mathbf{x}}_n \in \mathbb{R}^D$ as

$$\hat{\mathbf{x}}_n = (\mathbf{x}_n - \mathbf{\Psi})\,\mathbf{E}, \tag{17}$$

where $\mathbf{E} = [\mathbf{e}_1\ \mathbf{e}_2\ \cdots\ \mathbf{e}_D] \in \mathbb{R}^{24S\times D}$ is an eigenvector matrix. In this work, $D = 10$ is used.

The Parzen-window density estimate (Fukunaga and Hayes, 1989) is used for estimating $p(\mathbf{x}|w_k)$ of texture class w_k from given texture features $\hat{\mathbf{x}}_n$. It involves the superposition of a normalized window centred on a set of samples, i.e.,

$$p(\mathbf{x}|w_k) = \frac{1}{N}\sum_{n=1}^{N}\varphi(\mathbf{x} - \mathbf{x}_n, h) \tag{18}$$

where $\varphi(\cdot,\cdot)$ is the kernel (window), and h is width of the window. In this work, it is found experimentally that $h = 20$ gives satisfactory results on different databases. The Gaussian kernel with the covariance matrix

$\mathbf{\Sigma} = \mathbf{I}$ ($\mathbf{I}$ is the identity matrix) is used as a kernel, i.e.,

$$\varphi(\mathbf{y}, h) = \frac{1}{(2\pi)^{D/2} h^D |\mathbf{\Sigma}|^{1/2}} \exp\left(-\frac{\mathbf{y}\mathbf{\Sigma}^{-1}\mathbf{y}^T}{2h^2}\right), \tag{19}$$

where $|\cdot|$ is determinant.

After the class conditional probability density function $p(\mathbf{x}|w_k)$ has been learned for each texture class w_k, the feature vector $\mathbf{x}_u$ extracted from a query image I_u is used in its classification. The feature vector $\mathbf{x}_u$ is projected onto each texture class w_k according to (17) to create a low dimensional projection vector $\hat{\mathbf{x}}_u$. The projected feature vectors are used in classifying I_u into one of the texture classes with the maximum likelihood, according to the Bayesian classification of (5), i.e.,

$$w_i = \underset{w_i=\{w_1,w_2,\ldots,w_K\}}{\operatorname{argmax}} p(\hat{\mathbf{x}}_u|w_i), \tag{20}$$

where it is assumed that all texture classes have the same a priori probabilities.

4.3. Texture retrieval

Texture retrieval is viewed as a search for the best N images, i.e., the N images most similar to a given query image I_q from a database with M images, $I_m, m = 1, 2, \cdots, M$. For this purpose, each image is represented by a feature vector as in (14). The similarity between two images is measured by the distance between the corresponding feature vectors. The goal is to select among the M possible distances the images with the N smallest distances in a ranked order, and thus the N images that are most similar to I_q.

Given two images I_q and I_m, let $\mathbf{F}_S^{I_q}$ and $\mathbf{F}_S^{I_m}$ represent the corresponding feature vectors extracted using S level DT-CWT decomposition according to (14). We define the distance measure $\Delta(\mathbf{F}_S^{I_q}, \mathbf{F}_S^{I_m})$, between two feature vectors, $\mathbf{F}_S^{I_q}$ and $\mathbf{F}_S^{I_m}$ as

$$\Delta(\mathbf{F}_S^{I_q}, \mathbf{F}_S^{I_m}) = \sum_{i=1}^{24S} \frac{|f_{S,i}^{I_q} - f_{S,i}^{I_m}|}{\sigma_i}, \tag{21}$$

where σ_i is defined for the feature vectors of the images in the database, i.e.,

$$\sigma_i = \sqrt{\frac{1}{(M-1)} \sum_{m=1}^{M} \left(f_{S,i}^{I_m} - \mu_i\right)^2}, \tag{22}$$

and $\mu_i = \frac{1}{M} \sum_{m=1}^{M} f_{S,i}^{I_m}$.

5. Experimental results

5.1. Test dataset

The effectiveness of the proposed texture feature extraction approach to texture classification is evaluated by performing supervised classification of several test images with varying texture complexities from two

commonly-used natural texture image databases: 128 monochrome images from MIT VisTex colour image database (MITVisTex, 1998) and 111 monochrome images from Brodatz album (Brodatz, 1966). Each texture image has a size of 512×512, with 256 grey levels. Each image is globally histogram equalized to ensure that the textures are not trivially discriminable simply based on the local mean or local variance. Different portions of the input patterns of each texture class are selected and used for training the texture classifier. We avoid using the texture patterns on a texture border for training because these patterns are not representative of the texture.

Each texture image is divided into two non-overlapping parts of size 256×512, one for training and one for testing. Overlapped samples are generated from the training texture images using a sliding window of size $K \times K$ which is moved with shifts of Δ in both the horizontal and vertical directions. The number of test samples varies with the value of Δ. The value of Δ is set to 8 to give a reasonable overlap between two test samples, thus a total of 1152 texture samples are used in training for each class. After training, another 1152 samples are used to evaluate the performance of the texture classifier.

In texture retrieval tests, MIT VisTex database and Brodatz album are used together. Each texture image is uniformly sampled with 288 sub-images of size 128×128, and thus a database of size $239 \times 288 = 68832$ is constructed. The query image is searched in this large database, and the top matches for the query image are used in performance evaluations.

5.2. Performance evaluation metrics

The success of texture classification is measured using the classification gain (G) in percentage, i.e.,

$$G\,(\%) = \frac{C}{T} \times 100\% \tag{23}$$

where C is the number of sub-images correctly classified and T is the total number of sub-images, derived from each texture image, i.e., $T = 1152$ in our case.

Texture retrieval performance is measured using precision-recall curve. Recall signifies the relevant images in the database that are retrieved in response to a query. Precision is the proportion of the retrieved images that are relevant to the query. More precisely, let A and B be the set of relevant items and the set of retrieved items, respectively. Let us further assume that a, b, and c be the retrieved-relevant images, retrieved-irrelevant images, and unretrieved-relevant images, respectively. Recall and precision are then defined as the following conditional probabilities (Smith and Chang, 1996):

$$\begin{aligned} recall &= P\,(B|A) = \frac{P\,(B \cap A)}{P\,(A)} = \frac{a}{a+c}, \\ precision &= P\,(A|B) = \frac{P\,(A \cap B)}{P\,(B)} = \frac{a}{a+b}. \end{aligned} \tag{24}$$

With these conditions, the image retrieval capability of a method is said to be more effective than that of another method if its precision values at the same recall values are higher than those of the other method.

5.3. Implementations of methods

In the experiments for texture classification, in addition to the proposed method, the methods in (Celik and Tjahjadi, 2009) and (Kokare et al., 2007) are implemented. In the implementations, three level decompositions are used for wavelet decompositions of the texture samples, i.e., $S = 3$, to achieve good performances. We also implemented the Haralick texture classification method (Haralick et al., 1973) based on grey-level co-occurrence matrix (GLCM). Only a subset of the 14 Haralick features (i.e., energy, contrast, correlation, entropy, homogeneity, cluster shade and cluster prominence) representing the most commonly chosen ones are used in our study. In the experiments for GLCM method, quantization is applied to the texture datasets considered in this paper. Different quantization levels are tested, and it is empirically found that 32 levels of quantization provide the best performance for GLCM method. More details on the extraction of Haralick features can be found in (Haralick et al., 1973). We also present the texture classification results of the method in (Arivazhagan et al., 2006a).

In order to make fair comparisons, Bayesian classification method are used in GLCM method, the methods in (Kokare et al., 2007) and (Celik and Tjahjadi, 2009), and the proposed method in the final stage of classifying extracted features into the texture classes.

5.4. Experiments on texture classification

The texture classification performances, in terms of classification gain, of different methods on VisTex and Brodatz images are evaluated and compared. The results, Table 1 and Table 2, show that the proposed method consistently and significantly outperforms all other methods.

The average classification gains of the different methods are shown in Table 3. The proposed method achieves an average classification gain of 99% on both databases. The performance of GLCM is the worst. This is mainly because the co-occurrence matrices derived from the images in both databases do not result in discriminative features for texture classification. Furthermore, the monochrome grey-level distributions of many of the texture images are similar to one other, thus degrading the classification performance. The method in (Arivazhagan et al., 2006a) achieves the second best performance mainly due to the extensive set of statistical texture features obtained from ridgelet decompositions of texture images. The performances of the methods in (Kokare et al., 2007) and (Celik and Tjahjadi, 2009) are very similar. The method in (Kokare et al., 2007) employs statistical features extracted from both DWT and rotated wavelet transform (RWT). For each level of texture decomposition, features are extracted from subbands of both DWT and RWT. Thus, at each level of decomposition, similar to (Celik and Tjahjadi, 2009), the feature vectors are extracted from 6 subbands, which are equally distributed between 0-90 degree directions. Since, the direction representation in DT-CWT transform is better than DWT+RWT, the texture classification performance of the method in (Celik and Tjahjadi, 2009) is better than the method in (Kokare et al., 2007) by 3% on average for both sets of images.

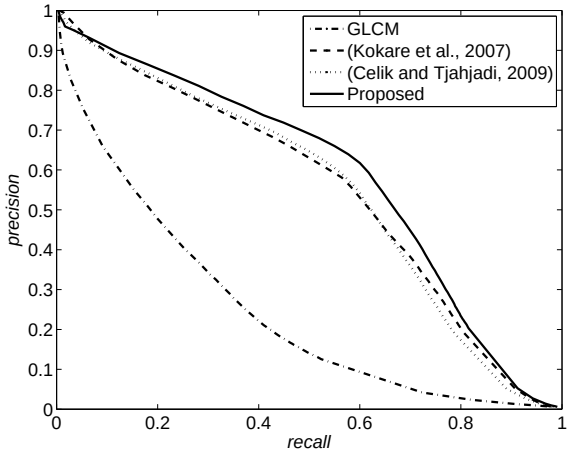


Figure 4: Texture retrieval performances of different methods as a precision-recall curve on a database with 68832 images.

The texture classification performance of the proposed method is much higher (about 18% higher) than the method in (Celik and Tjahjadi, 2009) due to two reasons. First, using both magnitude and phase information extracted from the subbands of DT-CWT decompositions results in a more discriminative feature vector than when only magnitude information is used as in (Celik and Tjahjadi, 2009). Second, the Bayesian framework and dimensionality reduction using PCA provide better feature representation.

5.5. Experiments on texture retrieval

The effectiveness of an image retrieval system is measured using a precision-recall graph. In constructing such a graph, the top matching T_r texture images for a query texture image are retrieved from a database, and for each retrieval, the *precision* and *recall* are calculated according to (24).

The texture retrieval performances of different methods are shown in Figure 4. Similar to the texture classification, GLCM achieves the worst performance in texture retrieval. This is mainly because the database consists of many texture samples with similar grey-level distributions but different shapes. The texture retrieval performance is significantly increased by employing wavelet transform techniques which rely on the high frequency components (edges) of texture images at different resolutions. The performances of the methods in (Celik and Tjahjadi, 2009) and (Kokare et al., 2007) are almost the same. This is mainly because both methods use almost the same directionality at each resolution. By using the phase together with the magnitude information increases the texture retrieval capacity of the proposed method when compared with the methods in (Celik and Tjahjadi, 2009) and (Kokare et al., 2007), thus achieving the best performance.

Table 1: Classification gains of different methods on 128 VisTex images. Method A: GLCM, Method B: (Arivazhagan et al., 2006a), Method C: (Kokare et al., 2007), Method D: (Celik and Tjahjadi, 2009).

Images	Method A	Method B	Method C	Method D	Proposed	Images	Method A	Method B	Method C	Method D	Proposed
Bark.0000	36	70	65	69	100	Grass.0000	55	75	65	65	100
Bark.0001	41	75	69	47	100	Grass.0001	77	85	92	91	99
Bark.0002	45	80	68	50	100	Grass.0002	82	95	91	90	99
Bark.0003	57	80	44	73	100	Leaves.0000	33	95	79	78	100
Bark.0004	54	100	86	90	100	Leaves.0001	92	95	97	97	100
Bark.0005	52	75	74	67	100	Leaves.0002	79	100	99	99	100
Bark.0006	43	85	72	66	100	Leaves.0003	91	100	98	99	100
Bark.0007	40	65	77	72	100	Leaves.0004	53	95	69	74	100
Bark.0008	37	90	72	78	100	Leaves.0005	21	20	34	30	100
Bark.0009	24	55	47	46	100	Leaves.0006	34	45	67	64	100
Bark.0010	27	85	54	57	100	Leaves.0007	27	60	33	35	100
Bark.0011	26	75	62	62	100	Leaves.0008	55	80	96	83	100
Bark.0012	26	65	45	43	99	Leaves.0009	37	75	60	55	100
Brick.0000	73	90	83	82	99	Leaves.0010	25	100	73	66	99
Brick.0001	52	90	85	81	98	Leaves.0011	42	85	70	84	100
Brick.0002	39	100	99	95	97	Leaves.0012	50	100	100	96	100
Brick.0003	33	90	91	81	100	Leaves.0013	80	85	88	90	100
Brick.0004	64	90	100	100	99	Leaves.0014	43	95	67	70	100
Brick.0005	65	100	47	56	100	Leaves.0015	26	70	54	52	100
Brick.0006	35	85	93	84	100	Leaves.0016	16	85	27	30	98
Brick.0007	19	75	27	30	100	Metal.0000	86	100	100	100	100
Brick.0008	10	30	17	19	100	Metal.0001	72	100	100	100	100
Clouds.0000	80	80	70	68	99	Metal.0002	78	100	91	92	100
Clouds.0001	72	80	81	82	99	Metal.0003	47	100	90	83	100
Fabric.0000	79	65	75	83	98	Metal.0004	78	85	82	82	100
Fabric.0001	85	90	93	89	100	Metal.0005	76	95	79	83	100
Fabric.0002	48	85	74	86	100	Misc.0000	52	95	82	69	100
Fabric.0003	56	95	80	99	100	Misc.0001	10	100	93	60	100
Fabric.0004	68	75	80	81	99	Misc.0002	90	100	92	88	100
Fabric.0005	49	75	69	74	99	Misc.0003	60	85	84	80	100
Fabric.0006	82	100	98	100	100	Sand.0000	87	100	100	100	100
Fabric.0007	53	100	85	90	100	Sand.0001	75	95	99	97	100
Fabric.0008	40	75	72	74	100	Sand.0002	73	85	66	74	100
Fabric.0009	90	100	95	98	100	Sand.0003	56	85	66	65	100
Fabric.0010	59	95	85	84	100	Sand.0004	35	55	53	55	100
Fabric.0011	60	100	82	95	100	Sand.0005	39	95	74	79	100
Fabric.0012	36	90	88	86	100	Sand.0006	63	100	93	89	100
Fabric.0013	76	100	100	99	100	Stone.0000	32	70	36	54	100
Fabric.0014	81	100	100	98	100	Stone.0001	36	80	90	86	100
Fabric.0015	44	95	90	81	100	Stone.0002	42	100	98	98	100
Fabric.0016	59	95	90	77	100	Stone.0003	48	90	79	64	100
Fabric.0017	89	100	100	100	100	Stone.0004	54	100	78	80	100
Fabric.0018	86	100	97	94	100	Stone.0005	64	95	76	77	100
Fabric.0019	85	100	95	93	100	Tile.0000	39	95	86	86	100
Flowers.0000	32	85	3	29	100	Tile.0001	21	100	66	70	100
Flowers.0001	49	90	67	64	100	Tile.0002	49	55	56	57	100
Flowers.0002	24	90	39	22	100	Tile.0003	64	60	80	80	99
Flowers.0003	14	95	90	61	100	Tile.0004	25	100	91	93	100
Flowers.0004	56	100	77	82	100	Tile.0005	54	35	81	64	100
Flowers.0005	27	100	76	69	100	Tile.0006	28	55	57	45	100
Flowers.0006	63	95	90	83	100	Tile.0007	51	90	98	93	100
Flowers.0007	53	100	73	70	100	Tile.0008	92	95	95	95	100
Food.0000	5	100	43	59	100	Tile.0009	76	85	96	96	100
Food.0001	85	95	97	97	100	Tile.0010	50	100	67	69	100
Food.0002	32	90	77	80	100	Water.0000	67	65	97	99	100
Food.0003	64	100	98	98	99	Water.0001	69	95	94	91	100
Food.0004	45	85	69	70	100	Water.0002	61	80	60	59	100
Food.0005	39	95	85	90	100	Water.0003	47	90	98	92	100
Food.0006	12	85	40	27	98	Water.0004	29	85	82	80	100
Food.0007	18	85	46	40	98	Water.0005	47	100	97	92	100
Food.0008	33	95	73	63	100	Water.0006	64	95	100	92	100
Food.0009	0	95	0	0	0	Water.0007	38	80	83	69	100
Food.0010	51	90	82	72	100	Wood.0000	31	45	60	47	100
Food.0011	95	100	99	99	100	Wood.0002	94	100	100	100	100

Table 2: Classification gains of different methods on 111 Brodatz images. Method A: GLCM, Method B: (Arivazhagan et al., 2006a), Method C: (Kokare et al., 2007), Method D: (Celik and Tjahjadi, 2009).

Images	Method A	Method B	Method C	Method D	Proposed	Images	Method A	Method B	Method C	Method D	Proposed
D1	68	100	100	98	100	D57	98	100	100	100	100
D2	66	80	88	68	100	D58	33	100	44	25	99
D3	84	100	100	100	100	D59	67	100	62	74	99
D4	73	100	78	100	100	D60	52	70	76	67	100
D5	63	100	85	88	100	D61	28	85	65	65	99
D6	55	100	100	95	100	D62	46	90	79	73	99
D7	62	90	75	79	100	D63	44	75	68	43	100
D8	82	100	99	97	100	D64	58	100	93	91	100
D9	55	100	100	100	100	D65	90	100	100	100	100
D10	54	95	81	77	100	D66	32	90	52	64	100
D11	62	100	97	97	100	D67	37	80	87	91	100
D12	72	90	95	95	100	D68	87	100	100	98	100
D13	19	90	66	73	100	D69	49	50	53	37	100
D15	97	90	96	98	100	D70	17	75	68	35	100
D16	91	100	100	100	100	D71	46	95	89	83	100
D17	67	100	100	100	100	D72	15	90	75	66	100
D18	86	100	97	98	100	D73	39	95	72	52	100
D19	60	100	96	74	100	D74	34	100	84	71	100
D20	99	100	100	100	100	D75	32	100	96	96	100
D21	100	100	100	100	100	D76	77	100	100	100	100
D22	82	100	95	97	100	D77	79	100	100	100	100
D23	65	95	79	71	100	D78	48	100	100	96	100
D24	79	100	100	100	100	D79	72	100	97	97	100
D25	71	95	92	94	100	D80	48	100	98	98	100
D26	56	100	94	97	100	D81	59	100	97	99	100
D27	40	90	42	56	100	D82	90	100	100	100	100
D28	77	100	97	91	100	D83	76	100	100	100	100
D29	70	100	97	97	100	D84	92	100	100	100	100
D30	50	100	65	59	95	D85	80	100	100	100	100
D31	38	100	56	45	98	D86	60	95	93	96	100
D32	77	100	100	100	100	D87	85	100	99	99	100
D33	75	100	92	82	100	D88	56	95	73	59	99
D34	92	100	100	99	100	D89	23	90	46	45	100
D35	72	100	100	100	100	D90	59	100	75	67	100
D36	38	100	99	100	100	D91	37	80	69	65	95
D37	77	100	100	97	100	D92	65	100	100	65	100
D38	66	100	94	88	100	D93	61	100	98	97	100
D39	57	100	76	62	100	D94	48	100	98	95	100
D40	47	95	68	36	100	D95	64	100	100	100	100
D41	42	100	94	85	100	D96	59	100	94	97	100
D42	19	50	79	72	100	D97	46	60	78	78	100
D43	43	35	54	59	89	D98	31	90	46	34	100
D44	22	65	44	51	96	D99	38	90	49	45	99
D45	9	90	21	26	99	D100	31	95	76	71	100
D46	63	100	99	98	100	D101	56	100	75	84	99
D47	91	100	100	100	100	D102	83	95	76	79	98
D48	95	100	100	98	100	D103	33	95	90	74	100
D49	100	100	100	100	100	D104	90	100	89	81	99
D50	50	90	98	81	100	D105	78	100	93	87	100
D51	81	95	98	93	100	D106	73	90	99	98	100
D52	36	95	80	81	100	D107	63	90	94	88	99
D53	98	100	100	100	100	D108	44	90	69	65	98
D54	28	90	89	85	100	D109	52	90	91	85	100
D55	89	100	100	100	100	D110	88	100	98	94	100
D56	79	100	100	100	100	D111	23	100	96	85	100
						D112	47	90	72	70	100

Table 3: Average classification gains of different methods on VisTex and Brodatz images.

Method	VisTex	Brodatz
GLCM	52	60
(Arivazhagan et al., 2006a)	86	94
(Kokare et al., 2007)	75	83
(Celik and Tjahjadi, 2009)	77	86
Proposed	99	99

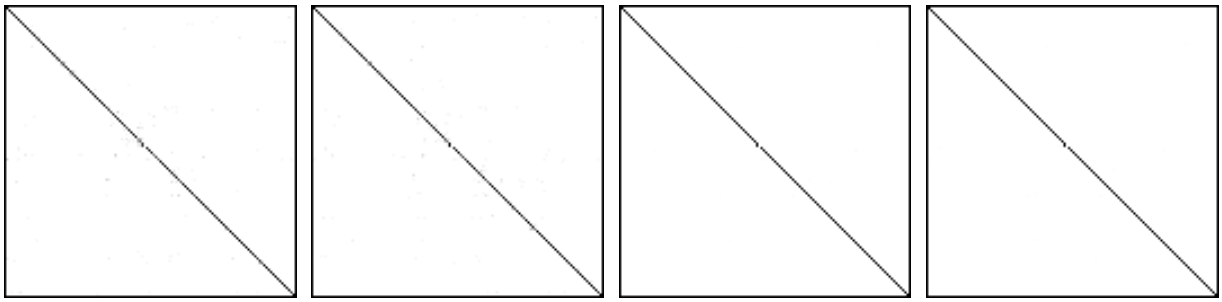
5.6. Effect of number of scales

The number of scales S used in DT-CWT decompositions affect both the texture classification and retrieval performances of the proposed method. In order to observe this effect, the texture classification and retrieval are performed for different values of S.

We employ a confusion matrix **CM** (Kohavi and Provost, 1998) to better represent texture classification performance with different values of S. **CM** is a $L \times L$ matrix for L different texture classes and $\mathbf{CM}(i, j)$, where $\mathbf{CM}(i, j) \in [0, 1]$, refers to the classification rate when samples from class i are identified as class j. When the texture classifier performs well, the confusion matrix is expected to be a diagonal matrix with diagonal values close to 1. In a pictorial representation of a confusion matrix, a white square denotes 0, a black square denotes 1, and values in between 0 and 1 are denoted by squares of varying grey shades. Thus, the more black squares there are in the diagonal of the pictorial representation of a confusion matrix the better is the performance. The effect of S is evaluated using the VisTex database only, since the performance of the proposed method is the same for the two databases.

Figure 5 shows the texture classification presented as confusion matrices. The average classification gain for $S = 1$, $S = 2$, $S = 3$ and $S = 4$ are 91%, 92%, 99% and 99%, respectively. The classification performance increases by 7% for $S \geq 3$. This can also be observed from Figure 5: the confusion matrix scatters around the main diagonal when $S \leq 2$ but converges to an almost diagonal matrix when $S \geq 3$. This increase in performance comes with a higher computational cost with respect to the lower S values. The classification performance does not change for $S \geq 3$.

The effect of the number of scales on texture retrieval is evaluated on both VisTex database and Brodatz album. Figure 6 shows the texture retrieval performance for different values of S. Similar to the texture classification performance, the texture retrieval performance increases with an increase in the value of S. However, the increase in the performance is not significant when $S \geq 3$. Thus, $S = 3$ gives satisfactory results on different databases when the input image size is 128×128.



(a) (b) (c) (d)

Figure 5: Texture classification performances of the proposed method for different values of the number of scales S presented as confusion matrices: (a) $S = 1$; (b) $S = 2$; (c) $S = 3$; and (d) $S = 4$.

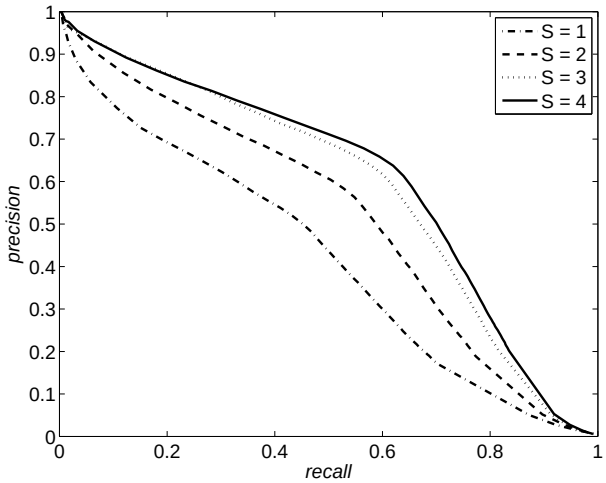


Figure 6: Texture retrieval performances of the proposed method as a precision-recall curve on a database of size 68832 images for different values of the number of scales S.

6. Conclusion

In this paper we propose a new texture classifier structure which uses both the magnitude and phase of DT-CWT subbands. The coarse approximation of the signal is not used for feature extraction since it is more prone to illumination changes. For each texture image, a multiscale texture feature vector is extracted from the magnitude and phase of DT-CWT subbands at different scales. The dimensionality of feature vectors is reduced using PCA, and the class conditional probability density function of a texture class is represented using Parzen-window estimation. Using Bayesian classification of the feature vector of a query image, the query image is assigned to the corresponding texture class with the highest a posteriori probability. The average classification rate of the proposed texture classification algorithm increases when the number of the scales S is increased.

It is shown that for texture classification the proposed classifier outperforms recently proposed texture classifiers. It achieves 99% correct classification rate on both MIT VisTex and Brodatz album databases. It is also shown that the use of phase information together with the magnitude information improves the performance significantly.

It is shown that for texture retrieval the proposed texture retrieval algorithm outperforms recently proposed texture retrieval algorithms. The average retrieval rate of the proposed texture retrieval algorithm increases when the number of the scales S is increased. However, the increase in performance is achieved at the expense of an increase in computational load.

Acknowledgments

The authors would like to thank Warwick University Vice Chancellor Scholarship for providing the funds for this research.

References

Arivazhagan, S., Ganesan, L., 2003a. Texture classification using wavelet transform. Pattern Recognit. Letts. 24, 1513–1521.

Arivazhagan, S., Ganesan, L., 2003b. Texture segmentation using wavelet transform. Pattern Recognit. Letts. 24, 3197–3203.

Arivazhagan, S., Ganesan, L., Kumar, T. S., 2006a. Texture classification using ridgelet transform. Pattern Recognition Letters 27 (16), 1875–1883.

Arivazhagan, S., Ganesan, L., Priyal, S., Dec 2006b. Texture classification using gabor wavelets based rotation invariant features. Pattern Recognit. Letts. 27 (16), 1976–1982.

Brodatz, P., 1966. Textures: A Photographic Album for Artists and Designers. Dover, New York, USA.

Celik, T., Tjahjadi, T., 2009. Multiscale texture classification using dual-tree complex wavelet transform. Pattern Recogn. Letts. 30 (3), 331–339.

Daugman, J. G., 1980. Two-dimensional spectral analysis of cortical receptive field profiles. Vision Research 20 (10), 847–856.

Do, M., Vetterli, M., Feb 2002. Wavelet-based texture retrieval using generalized gaussian density and kullback-leibler distance. IEEE Trans. Image Proc. 11 (2), 146–158.

Duda, R. O., Hart, P. E., Stork, D. G., November 2000. Pattern Classification (2nd Edition), 2nd Edition. Wiley-Interscience.

Faugeras, O., 1978. Texture analysis and classification using a human visual model. In: Proceedings of IEEE International Conference on Pattern Recognition. pp. 549–552.

Fukunaga, K., Hayes, R., Apr 1989. The reduced parzen classifier. IEEE Transactions on Pattern Analysis and Machine Intelligence 11 (4), 423–425.

Gonzalez, R. C., Woods, R. E., 2006. Digital Image Processing (3rd Edition). Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

Haralick, R. M., Shanmugam, K., Dinstein, I., Nov 1973. Textural features for image classification. IEEE Trans. Sys. Man Cybern. 3 (6), 610–621.

Hiremath, P., Shivashankar, S., 2008. Wavelet based co-occurrence histogram features for texture classification with an application to script identification in a document image. Pattern Recognit. Letts. 29, 1182–1189.

Jain, A., Farrokhnia, F., 1991. Unsupervised texture segmentation using gabor filters. Pattern Recognit. 24 (12), 1167–1186.

Jain, A., Karu, K., Feb 1996. Learning texture-discrimination masks. IEEE Trans. Pattern Anal. Mach. Intell. 18 (2), 195–205.

Kim, S., Kang, T., 2007. Texture classification and segmentation usingwavelet packet frame and gaussian mixture model. Pattern Recognit. 40, 1207–1221.

Kingsbury, B. Y. N., 1999. Image processing with complex wavelets. Phil. Trans. Royal Society London A 357, 2543–2560.

Kohavi, R., Provost, F., 1998. Glossary of Terms. Vol. 30. Kluwer Academic Publishers, Hingham, MA, USA.

Kokare, M., Biswas, P., Chatterji, B., 2007. Texture image retrieval using rotated wavelet filters. Pattern Recognit. Letts. 28, 1240–1249.

MITVisTex, 1998. Vision texture database. `http://www.media.mit.edu/vismod/`.

Muneeswarana, K., Ganesanb, L., Arumugamc, S., Soundara, K., 2005. Texture classification with combined rotation and scale invariant wavelet features. Pattern Recognit. 38, 1495–1506.

Smith, J. R., Chang, S.-F., February 1996. Tools and techniques for color image retrieval. In: IS&T/SPIE Symposium on Electronic Imaging: Science and Technology-Storage and Retrieval for Image and Video Databases IV. Vol. 2670. San Jose, CA.

Wouwer, G., Scheunders, P., Dyck, D., Apr 1999. Statistical texture characterization from discrete wavelet representations. IEEE Trans. Image Proc. 8 (4), 592–598.