ELSEVIER

Signal Processing 81 (2001) 1337–1356

SIGNAL PROCESSING

www.elsevier.nl/locate/sigpro

An adaptive approach to unsupervised texture segmentation using M-Band wavelet transform

Mausumi Acharyya, Malay K. Kundu*

Machine Intelligence Unit, Indian Statistical Institute, 203 Barrackpore Trunk Road, Calcutta 700 035, India

Received 9 December 1999; received in revised form 5 December 2000

Abstract

The M-band wavelet decomposition, which is a direct generalization of the standard 2-band wavelet decomposition is applied to the problem of an unsupervised segmentation of two texture images. Orthogonal and linear phase M-band wavelet transform is used to decompose the image into $M \times M$ channels. Various combinations of these bandpass sections are taken to obtain different scales and orientations in the frequency plane. Texture features are obtained by subjecting each bandpass section to a nonlinear transformation and computing the measure of energy in a window around each pixel of the filtered texture images. The window size in turn is adaptively selected depending on the frequency content of the images. Unsupervised texture segmentation is obtained by simple K-means clustering. Statistical tests are used to evaluate the average performance of features extracted from the decomposed subbands.

Keywords: M-band wavelets; Texture segmentation; Feature extraction; Multiscale representation

1. Introduction

Most natural surfaces exhibit texture. The characterization of texture plays an important part of many computer vision system. Texture analysis has wide range of applications like medical diagnosis, content-based-image retrieval, satellite imaging and many others. Image segmentation is a difficult yet very important task in image analysis and many computer vision applications. The problem of segmenting an image into meaningful regions based on textural cue is referred to as texture segmentation problem. A large number of techniques for analyzing textured image have been proposed in the past [13] and in a recent review, Tuceryan and Jain [34] have discussed some of those techniques.

In this paper we focus on a particular approach to texture (image) analysis which is referred to as multichannel filtering approach. This approach for texture analysis is intuitively appealing because it allows us to exploit differences in dominant sizes and orientations of different textures. In several papers the successful applications of multichannel filtering for texture segmentation were reported [10,16,3] using various filtering techniques, such as isotropic filters [5] discrete cosine transform (DCT) [23] and Gabor filters. The reason for the popularity of Gabor filters is due to their joint optimum

* Corresponding author. Tel.: +91-33-577-4906; fax: +91-33-577-6680.

E-mail addresses: res9522@isical.ac.in (M. Acharyya), malay@isical.ac.in (M. K. Kundu).

0165-1684/01/$ - see front matter
PII: S0165-1684(00)00278-4

resolution in time and frequency. Randen and Husϕy [28] have examined the performance of multichannel segmentation schemes based on a more general class of filters including Gabor filters. However a large combination of parameters makes texture discrimination using Gabor filters computationally expensive.

Recent development in wavelet theory has provided a promising alternative through multichannel filter banks that have several potential advantages over Gabor filters namely,

(i) Wavelet filters cover exactly the complete frequency domain.
(ii) Fast algorithms are readily available to facilitate computation.

More recently, studies on successful application of wavelet theory on texture analysis have been reported using the multiresolution signal decomposition developed by Mallat [22]. He used quadrature mirror filters to relate information at different scales of decomposition of the embedded subspace representation.

Unser [36] used overcomplete wavelet decomposition of the standard wavelet and characterized the texture by a set of channel variances estimated at the output of the filter bank. The standard or the octave band wavelet decomposition imply finer frequency resolution in the low-frequency region than in the high-frequency region. The work of Chang and Kuo [3] indicates that the texture features are more prevalent in the intermediate frequency band. Laine and Fan [20] carried out studies on texture analysis based on this indication. They used multichannel wavelet frames for feature extraction. Representations obtained from both standard wavelets and wavelet packets were evaluated.

One of the drawbacks of standard wavelets is that they are not suitable for the analysis of high-frequency signals with relatively narrow bandwidth. So the main motivation of the present work is to use the decomposition scheme based on M-band wavelets, which yield improved segmentation accuracies. Unlike the standard wavelet decomposition which gives a logarithmic frequency resolution, the M-band decomposition gives a mixture of a logarithmic and linear frequency resolution. Further as an additional advantage, M-band wavelet decompositions yield a large number of subbands which is required for good quality segmentation.

The analysis of textures using an M-channel wavelet approach was investigated by Greiver et al. [12]. They used a 3-channel extension of 2-channel biorthogonal wavelets. Relevant texture features were used to design a 3-channel biorthogonal wavelet, that were subsequently used in the decomposition and classification of textures. Recently Chitre and Dhawan [4] have used the M-band wavelets for texture classification. Randen et al. [26] have proposed a class of optimally designed filter banks for texture segmentation purpose. Several recent techniques, other than multichannel filtering method for unsupervised texture segmentation can also be found in the literature [14,24].

Here we investigate the problem of segmentation of two texture images by using a generalization of the wavelet transforms to the M-band case. We conjecture that the M-band wavelet transform has the potential to perform multiscale, multidirectional filtering of the images, since it is a tool for viewing signals at different scales and decomposes a signal by projecting it onto a family of functions generated from a single-wavelet basis via its dilations and translations [1,31].

A typical system set-up for the texture segmentation algorithm is illustrated in Fig. 1. The image is first wavelet transformed into $M \times M$ channels by applying the M-band transform in a separable manner, without downsampling which gives an overcomplete representation of the image. Then various combinations of these bandpass sections are taken to obtain different scales and orientations in the frequency plane. In the next step, a local energy estimator consisting of a nonlinear operation (which basically rectifies the filter response) and a smoothing filter, is applied to the various combinations of these bandpass sections. The size of the smoothing window is determined adaptively based on the spectral frequency content of the images. These operations give the texture features that can be classified successfully. The use of overcomplete wavelet representation alleviates the problem of inaccurate edge localization of the texture elements and discrepancies in detection of boundaries of different texture classes. The ability of the filters

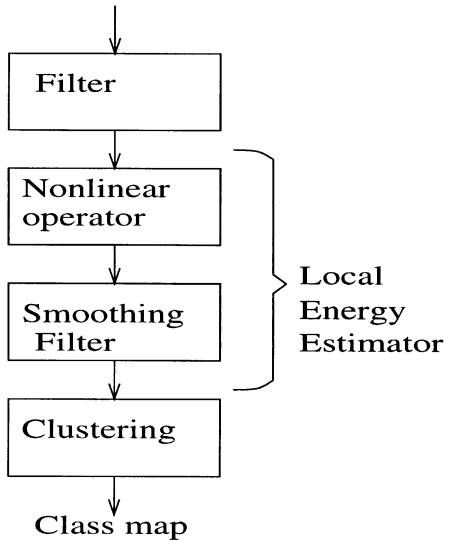


Fig. 1. A typical system setup for texture segmentation.

to exploit the characteristics like difference in frequency (size) and orientation of different textured regions is the key to multichannel approach to texture analysis.

The organization of the paper is as follows. Section 2 briefly overviews the wavelet transform and M-band wavelet transform. Section 3 presents the analysis of the multichannel filtering technique used in the proposed texture segmentation scheme, and extraction of features and further discusses the integration of these extracted features. Section 4 gives experimental results and finally in Section 5 we give critical comments of the present work and conclude our study.

2. Wavelet transform and M-band wavelet transform

In this section the wavelet transform and M-band wavelet transform are described briefly. An exhaustive mathematical treatment is available in [7,31].

2.1. Discrete wavelet transform

The *wavelet transform* is a signal decomposition onto a set of basis functions called wavelets. The wavelets are obtained from a single-prototype function by scalings and shifts [6,22,30]. This is the standard 2-band wavelet transform. The wavelet transform of a 1-D signal $f(x)$ is defined as

$$Wf_a(b) = \int f(x)\psi^*_{a,b}(x)\,\mathrm{d}x, \tag{1}$$

where ψ is the mother wavelet and a and b are dilation and translation parameters respectively. The wavelet decomposition can be interpreted as signal decomposition in a set of independent, spatially oriented frequency channels. Under these constraints an efficient real space implementation of the transform using quadrature mirror filter exists [22]. The full discrete wavelet expansion of a signal $f(x) \in l_2$ (l_2 is the space of square summable functions) is given as

$$f(x) = \sum_{k\in z} s_{J,k}\phi_{J,k}(x) + \sum_{j=1}^{J}\sum_{k\in z} d_{j,k}\psi_{j,k}(x), \tag{2}$$

where ϕ and ψ are the scaling and wavelet functions, respectively and are associated with the analyzing (or synthesizing) filters h and g. $d_{j,k}$'s are the wavelet coefficients and $s_{J,k}$'s are the expansion coefficients of the coarser signal approximation of $f(x)$. It also follows from this construction that the family of sequences $\{\phi_{J,k}, \psi_{1,k}, \psi_{2,k}, \ldots, \psi_{J,k}\}_{k\in Z}$ constitutes an orthonormal basis. The discrete normalized basis functions are defined as

$$\phi_{j,k}(x) = 2^{j/2}h_j(2^j x - k), \tag{3}$$

$$\psi_{j,k}(x) = 2^{j/2}g_j(2^j x - k), \tag{4}$$

where j and k are the scale and translation indices, respectively and the factor $2^{j/2}$ is an inner product normalization.

We denote V_j as $V_j = \mathrm{Span}_k\{\phi_{j,k}(x)_{k\in Z}\}$ which is the subspace spanned by the scaling function at resolution j, (Z corresponds to all integers) and consider the sequence of nested subspaces $l_2 = V_0 \subset V_1 \subset V_2 \ldots \subset V_J$. We also introduce the subspaces W_j ($j = 1, \ldots, J$), where W_j is the residue (or detail) space at resolution j and is defined as the orthogonal complement of V_j with respect to V_{j+1} (i.e. $V_{j+1} = V_j \oplus W_j$ and $V_j \perp W_j$). It can be shown by induction that the families of sequences $\{\phi_{j,k}\}_{k\in Z}$ and $\{\psi_{j,k}\}_{k\in Z}$ provide orthogonal bases of V_j and W_j, respectively [2].

At, even $J = -\infty$, $l_2 = \ldots \oplus W_{-2} \oplus W_{-1} \oplus W_0 \oplus W_1 \oplus \ldots \oplus V_0$ (Fig. 2a).

2.2. M-band wavelets

M-band wavelets are a direct generalization of the conventional wavelets [6,22,30]. The standard dyadic wavelets are not suitable for the analysis of high-frequency signals with relatively narrow bandwidth [31]. To resolve this problem M-band orthonormal wavelets were developed as a direct generalization of the 2-band orthogonal wavelets of Daubechies [6]. These M-band wavelets are able to zoom in onto narrowband high-frequency components of a signal and have been found to give better energy compaction than 2-band wavelets [37].

An M-band wavelet system form a tight frame for the set of square integrable functions defined over the set of real numbers $L^2(R)$ [2]. There are $M-1$ wavelets, $\psi_i(x)$, $i = 1, \ldots, M-1$ associated with the scaling function. For any function $f(x) \in L^2(R)$, it can be shown that

$$f(x) = \sum_{i=1}^{i=M-1} \sum_{j \in Z} \sum_{k \in Z} \langle f(x), \psi_{i,j,k}(x) \rangle \psi_{i,j,k}(x), \tag{5}$$

where Z represents the set of integers and $\langle , \rangle$ is the inner product operator. The $\psi_{i,j,k}(x)$ functions are obtained by scaling and shifting the corresponding wavelet $\psi_i(x)$:

$$\psi_{i,j,k}(x) = M^{j/2} \psi_i(M^j x - k),$$
$$i = 1, \ldots, M-1,\ k \in Z,\ j \in Z. \tag{6}$$

Given a scaling function $\psi_0(x)$ in $L^2(R)$, the wavelet functions are defined as

$$\psi_i(x) = \sqrt{M} \sum_{k=0}^{k=N-1} h_i(k) \psi_0(Mx - k),$$
$$i = 1, \ldots, M-1. \tag{7}$$

The scaling function satisfies the recursive equation and is compactly supported in $[0, (N-1)/(M-1)]$,

$$\psi_0(x) = \sqrt{M} \sum_{k=0}^{k=N-1} h_0(k) \psi_0(Mx - k), \tag{8}$$

where the sequence h_0 is the scaling filter [2] and satisfies the following linear and quadratic constraints:

$$\sum_{k=0}^{k=N-1} h_0(k) = \sqrt{M}, \tag{9}$$

$$\sum_{k=0}^{k=N-1} h_0(k) h_0(k + Ml) = \delta l. \tag{10}$$

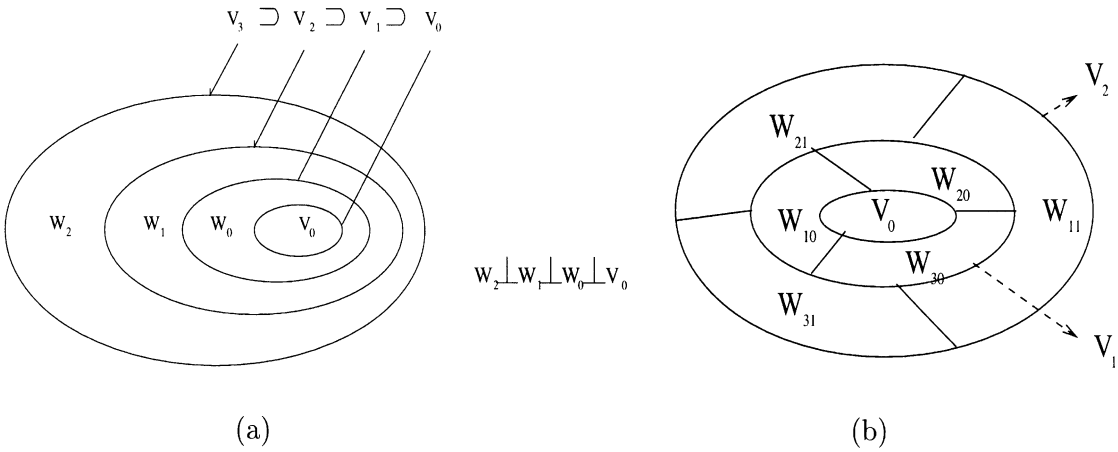


Fig. 2. Nested vector spaces spanned by the scaling and wavelet functions in (a) standard wavelet and (b) M-band wavelet.

The $(M-1)h_i$ vectors called the wavelet filters satisfy the equation

$$\sum_{k=0}^{k=N-1} h_i(k)h_j(k+Ml) = \delta(l)\delta(i-j). \tag{11}$$

2.3. Multiresolution analysis

The scaling function and the $M-1$ wavelet functions also define a multiresolution analysis [2]. A multiresolution analysis is a sequence of approximation spaces for $L^2(R)$. If the space spanned by the translates of $\psi_i(x)$ for fixed j and $k \in Z$ is defined by $W_{i,j} = \mathrm{Span}\{\psi_{i,j,k}\}$, then it can be shown [2] that

$$W_{0,j} = \bigoplus_{i=0}^{M-1} W_{i,j-1}, \tag{12}$$

$$\lim_{j\to\infty} W_{0,j} = L^2(R). \tag{13}$$

Thus the $W_{0,j}$ spaces form a multiresolution space for $L^2(R)$. An important aspect of M-band wavelets is that a given scaling filter h_0 specifies a unique $\psi_0(x)$ and consequently a unique multiresolution analysis. For example, with $M=4$,

$$V_1 = V_0 \oplus W_{10} \oplus W_{20} \oplus W_{30},$$

$$V_2 = V_1 \oplus W_{11} \oplus W_{21} \oplus W_{31},$$

where V_j's are the spaces spanned by the scaling function at different resolution and W_j's are the spaces spanned by the wavelet functions. It is pictorially illustrated in Fig. 2b.

The scale-space tiling for the standard wavelet ($M=2$) and for M-band wavelet ($M=4$) are depicted in Fig. 3. The figure clearly shows that the frequency tilings in the standard wavelet decomposition are logarithmic (octave band) while the M-band decomposition gives a mixture of logarithmic as well as linear frequency tilings.

There is also a close relationship between M-band wavelets and M-channel filter banks [2]. A typical M-channel filter bank is shown in Fig. 4.

3. Computing texture features

An important aspect of texture analysis is to develop a set of texture measures (features) that can successfully discriminate arbitrary textures. The feature extraction scheme that we have used has a multichannel filtering, and a subsequent nonlinear stage followed by a smoothing filter (both these constitute the local energy estimator) as shown in Fig. 1. The objectives of the filtering and that of the local energy estimator are to transform the edges between textures into detectable discontinuities.

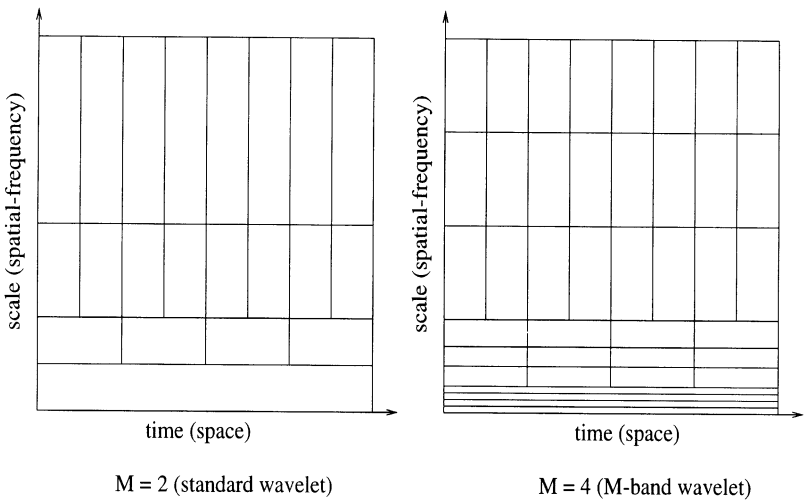


Fig. 3. 2-band and 4-band basis tilings.

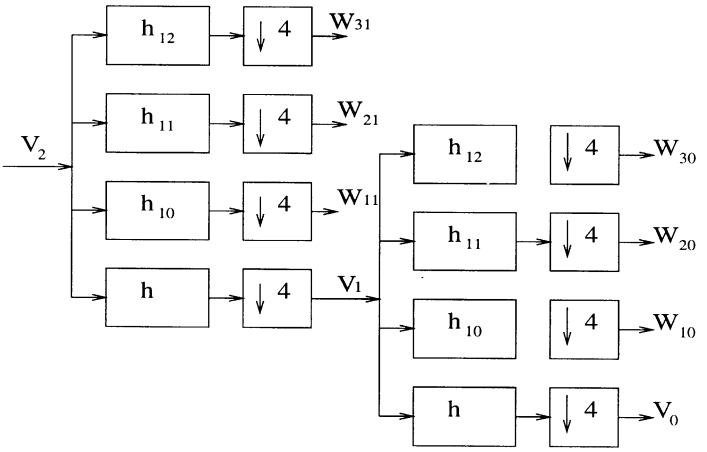


Fig. 4. An M-channel filter bank structure ($M = 4$).

3.1. M-band wavelet filters

The filter bank in essence is a set of bandpass filters with frequency and orientation selective properties. In the filtering stage we make use of orthogonal and linear phase M-band wavelet transform [1] to decompose the texture images into $M \times M$-channels, corresponding to different direction and resolutions. The 1-D $M(=4)$-band wavelet filter impulse responses are given by ψ_i and their corresponding transfer functions are denoted by H_i for $i = 1, \ldots, 4$. ψ_1 is the scaling function (lowpass filter) and the other ψ_i's correspond to the wavelet functions (bandpass filters). In this work we have obtained the M^2-channel 2-D separable transform by the tensor product of M-band 1-D wavelet filters but without downsampling, which are denoted by $\psi_{i,j}$, for $i,j = 1,2,3,4$ with $M = 4$. The i,jth resolution cell is obtained via the filtering step $H_{i,j} = \psi_{i,j}\psi^*_{i,j}$ for $i,j = 1,2,3,4$ with $M = 4$. The decomposition of the image into $M \times M(=16)$ channels is illustrated in Fig. 5a.

Since the spectral response to edges of an image is strongest in direction perpendicular to the edge, while it decreases as the look direction of the filter approaches that of the edge. Therefore we can perform edge detection by using 2-D filtering as follows:

- *Horizontal edges*: are detected by highpass filtering on columns and lowpass filtering on rows.
- *Vertical edges*: are detected by lowpass filtering on columns and highpass filtering on rows.
- *Diagonal edges*: are detected by highpass filtering on columns and highpass filtering on rows.
- *Horizontal–diagonal edges*: are detected by highpass filtering on columns and lowpass filtering on rows.
- *Vertical–diagonal edges*: are detected by lowpass filtering on columns and highpass filtering on rows.

A typical edge detection filter corresponding to a particular direction covers a certain region in the 2-D spatial frequency domain, this is illustrated in Fig. 5b where, f_x and f_y are the horizontal and vertical frequencies. Based on this concept several wavelet decomposition filters are possible which are given by, $\sum_{Reg} H_{i,j}$, where Reg denotes the frequency sector corresponding to a certain direction and scale.

The filter system we are using here is orthogonal and has quadrature mirror filter (QMF) structure, that is $\sum_{i=1}^{M}\sum_{j=1}^{M} \psi_{i,j}\psi^*_{i,j} = 1$. The resulting 2-D filters treats all the frequencies in a resolution cell as equally as possible. The number of channels as well as the number of possible filter combinations depend on the number of bands (M). The decomposition filters $\sum_{Reg} H_{i,j}$ are formed as follows for different directions in increasing level of resolutions.

- *Horizontal direction*:

$$\mathrm{Filt}_{\mathrm{hor}_1} = H_{12},$$

$$\mathrm{Filt}_{\mathrm{hor}_2} = H_{12} + H_{13},$$

$$\mathrm{Filt}_{\mathrm{hor}_3} = H_{12} + H_{13} + H_{14} + H_{24}.$$

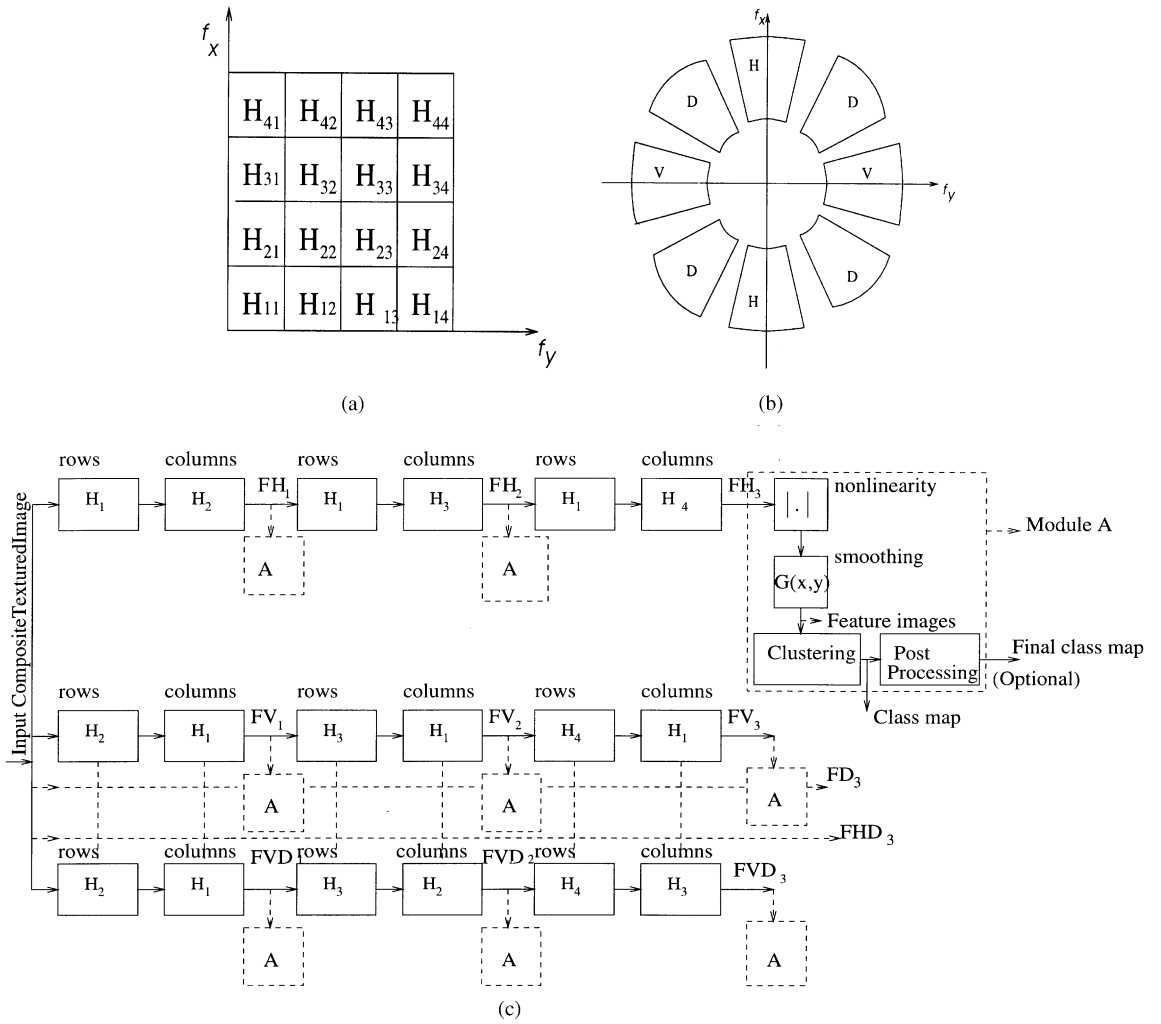


Fig. 5. (a) Frequency bands corresponding to decomposition filters. (b) Frequency sector representations for filtering in horizontal (H), vertical (V) and diagonal (D) directions. (c) Block diagram of the algorithm of our method.

- *Vertical direction*:

$$\mathrm{Filt}_{\mathrm{ver}_1} = H_{21},$$

$$\mathrm{Filt}_{\mathrm{ver}_2} = H_{21} + H_{31},$$

$$\mathrm{Filt}_{\mathrm{ver}_3} = H_{21} + H_{31} + H_{41} + H_{42}.$$

- *Diagonal direction*:

$$\mathrm{Filt}_{\mathrm{diag}_1} = H_{22},$$

$$\mathrm{Filt}_{\mathrm{diag}_2} = H_{22} + H_{33},$$

$$\mathrm{Filt}_{\mathrm{diag}_3} = H_{22} + H_{33} + H_{44}.$$

- *Horizontal–diagonal direction*:

$$\mathrm{Filt}_{\mathrm{hdiag}_1} = H_{12},$$

$$\mathrm{Filt}_{\mathrm{hdiag}_2} = H_{12} + H_{23},$$

$$\mathrm{Filt}_{\mathrm{hdiag}_3} = H_{12} + H_{23} + H_{34}.$$

- *Vertical–diagonal direction*:

$$\mathrm{Filt}_{\mathrm{vdiag}_1} = H_{21},$$

$$\mathrm{Filt}_{\mathrm{vdiag}_2} = H_{21} + H_{32},$$

$$\mathrm{Filt}_{\mathrm{vdiag}_3} = H_{21} + H_{32} + H_{43}.$$

These filter outputs basically give a measure of signal energies at different directions and scales, the corresponding filtered images are denoted by F_{H_i}, F_{V_i} etc. for $i = 1,2,3$ as shown in Fig. 5c.

3.2. Local energy estimator

The objective of the local energy estimator is to estimate the energy of the filter output in a local region. The purpose of the estimator is to transmit the strong bandpass frequency components resulting in a high-constant gray value and weaker frequency components into a low-constant gray value. Now, in essence accurate edge preservation and accurate energy preservation conflicts each other. This is because high spatial resolution calls for accurate edge localization, while high spatial frequency resolution is required for accurate energy estimation. Although energy is usually defined in terms of a squaring nonlinearity, but in a generalized energy function, other alternatives are also in use. We have used the most popular magnitude operation $|\cdot|$. One reason for choosing this nonlinear operator is that it is parameter free, meaning it is independent of the dynamic range of the input image and also of the filter amplification.

The nonlinear transform is succeeded by a Gaussian low-pass (smoothing) filter of the form

$$h_G(\dot{x}, y) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(1/2\sigma^2)(x^2+y^2)},$$

where, σ defines the spatial extent of the averaging filter. Formally, the feature image $\mathrm{Feat}_k(x, y)$ corresponding to filtered image $F_k(x, y)$ is given by,

$$\mathrm{Feat}_k(x, y) = \sum_{(a,b)\in G_{xy}} \Gamma(F_k(a, b)h_G(x - a, y - b)), \quad (14)$$

where $k = H_i, V_i$, etc., $\Gamma(\cdot)$ is the nonlinear function and G_{xy} is a $G \times G$ window centered at pixel with coordinates (x, y). The size G of the smoothing or the averaging window in Eq. (14) is an important parameter. More reliable measurement of texture feature demands larger window sizes. On the other hand, more accurate localization of region boundaries requires smaller windows. Another important aspect is that, Gaussian weighted windows are naturally preferable over unweighted windows, because, the former are likely to result in more accurate localization of texture boundaries, since averaging blurs the boundaries between textured regions.

3.2.1. Choice of σ of the smoothing filter

The choice of the space constant σ of the averaging filter is very crucial. The problem is how to determine the size of the smoothing filter. If we want to estimate the local energy of an image with low spatial frequency the smoothing filter must have a wide unit impulse response, while narrower filter is to be used for higher-frequency content image.

In the present work we set the smoothing filter size based on the measure of the spectral content of the image. *Spectral flatness measure* (SFM) gives a measure of the overall image activity. The spectral flatness of a digital image is defined as the ratio of the arithmetic and the geometric mean of the Fourier coefficients [18]. For two-dimensional digital image this can be expressed as

$$\mathrm{SFM} = \frac{1/(MN)\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}|\hat{F}(i,j)|^2}{[\prod_{i=0}^{M-1}\prod_{j=0}^{N-1}|\hat{F}(i,j)|^2]^{1/MN}}. \quad (15)$$

$\hat{F}(i,j)$ is the (i,j)th Fourier coefficient of the two-dimensional image. SFM has a dynamic range of [0,1].

Highly active image means SFM close to 1, then the image has many edges or has predominantly high frequencies. So the image requires a smaller window for smoothing. Moderately active image has SFM of somewhat moderate value within 0 and 1. That means the image contains moderate range of frequencies and require a moderate window size for good feature extraction. Finally an image with low SFM is lowly active and has low spectral content. It is evident that this type of image would require larger window size of the smoothing filter.

We have found experimentally that the spatial extent of the windows, for these three categories of image activities range from 11×11 to 31×31.

With these choices we have worked successfully on all the test images that have been experimented. Thus we can adaptively select the size of the averaging window depending on the spectral content of the image. Our scheme is adaptive in the sense that

we do not use any fixed windowing operation and hence it can accommodate diverse set of textured images as input.

3.3. Integrating the feature images

Having obtained the feature images, the main task is to integrate these feature images to produce a segmentation. Let us assume that there are K texture categories. $C_1, \ldots, C_K$, present in the image. If our texture features are capable of discriminating these categories then the patterns belonging to each category will form a cluster in the feature space which is compact and isolated from clusters corresponding to other texture categories. Pattern clustering algorithms are ideal modes for forming such clusters in the feature space. Segmentation algorithm accepts a set of features as input and assign a class for each pixel. Fundamentally this can be considered as a multidimensional data clustering problem. Texture segmentation algorithms can be divided into two categories: supervised and unsupervised segmentation [33]. We emphasize on the feature extraction (representation) part in this work. So we have used a traditional *K-means* clustering algorithm. An overview of the unsupervised *K-means* clustering algorithm is depicted below.

K-means ($x[1:N,1:M]$, NC)
$x[1:N,1:M]$: array of structure containing vectors
$N \rightarrow$ Number of feature elements in a feature vector
$M \rightarrow$ Data size (number of pixels in the image)
$NC \rightarrow$ Total number of classes
begin
  begin (Initialization)
    Select NC number of vectors arbitrarily from the array $x[1:N,1:M]$ and then each of these are assigned a class, these form the initial class centers C_k's.
  end

  begin
    Euclidean distance between each of the M vectors and the selected NC vectors are found out taking one out of M vectors at a time. A vector is assigned to the class k if it is closest to C_k. Recompute the class centers C_k by taking mean of the vectors assigned to class k. Repeat until there is no change in the class centers.
  end
end

3.4. Post processing

After the class maps are obtained, segmentation results can be improved by post processing. The simple *K-means* clustering algorithm labels each pixel independently and does not take into account the high correlation between neighboring pixels. A more sophisticated algorithm should incorporate some neighborhood constraint into the segmentation process, such as relaxation labelling. So we have used median filtering to simulate the benefit of a local constraint.

3.5. Algorithm

The texture segmentation algorithm based on the M-band wavelet decomposition is illustrated in the block diagram (Fig. 5c).

This algorithm consist of the following steps:

- The input image is first decomposed into $M \times M$ channels by wavelet analysis without downsampling as referred in Section 4.1. In this work we have used an eight tap 4-band wavelet [1], so in all we get 16 decomposition channels as discussed in Section 3.1, which means the feature set comprises of 16 feature elements. Out of these 16 features we ignore the low-frequency channel feature corresponding to H_{11} and FHD_1 and FVD_1 since these are nothing but FH_1 and FV_1, respectively. Now we are left with 13 features.
- These outputs are subjected to the nonlinear operation followed by smoothing as discussed in Section 3.2, which then form the feature images Feat_k.
- We have a matrix of $N \times M$, where N is the number of feature elements in each vector (13 in this case) and M is the total data size (the total number of pixels in the input image). The features are normalized between [0,1] along each column of the feature matrix and subjected to the clustering algorithm. This step gives us the class map corresponding to the composite texture image.

3.6. Statistical significance of features

The performance of the different extracted features have been compared using a statistical significance test. We test whether two distributions have significantly different variances. The F-test examines the hypothesis that two samples have different variances by trying to reject the null hypothesis that their variances are actually consistent [25]. The statistics F is the ratio of one variance to the other, so value either $\ll 1$ or $\gg 1$ will indicate very significant difference. The F-test routine returns the value of f and its significance as prob. Small value of prob indicate that the two samples have significantly different variance.

$$\text{prob} = 2\text{betai}\left(0.5 * df2, 0.5 * df1, \frac{df2}{(df2 + df1 * f)}\right)$$

where,

$$f = \text{var1}/\text{var2} \quad \text{if var1} > \text{var2},$$
$$df1 = n1 - 1, \qquad df2 = n2 - 1$$

or

$$f = \text{var2}/\text{var1} \quad \text{if var2} > \text{var1},$$
$$df1 = n2 - 1, \qquad df2 = n1 - 1.$$

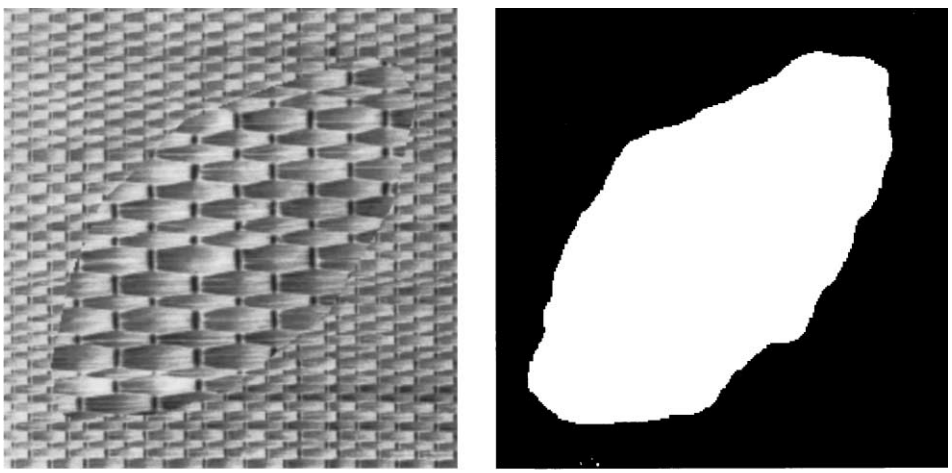
var1 and var2 are the variances of the samples and $n1$ and $n2$ their sizes respectively. betai is the incomplete beta function.

We have tested the set of features corresponding to the two textures in each image for significance.

4. Experimental results

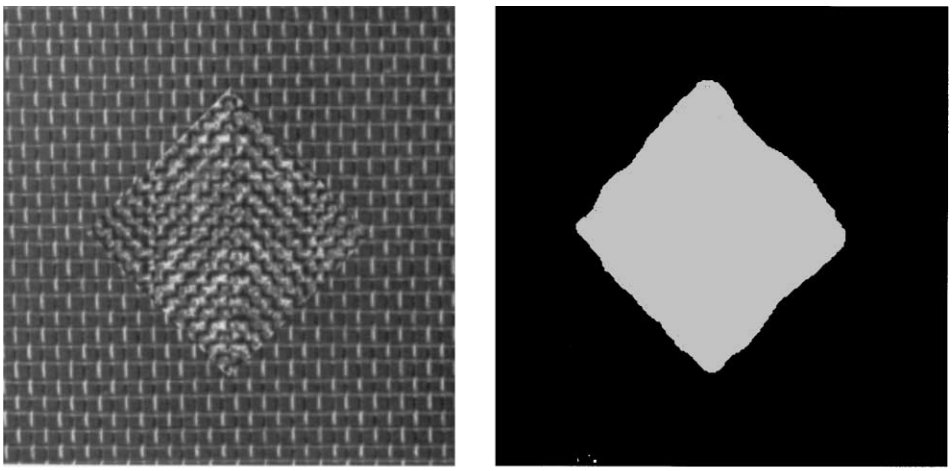
We have applied our texture segmentation algorithm to several two texture images, in order to demonstrate the performance of our algorithm. These images are created by collaging subimages extracted from various natural texture from images.

It has already been mentioned in Section 3.5 that out of the 16 features possible for $M = 4$, we considered only 13. The number of features could even be reduced in many texture images. In case of Nat2 and Nat3 (Figs. 6 and 7) of size 256×256 texture pairs, only 5 features were sufficient for successful segmentation of the images. It is to be noted that the results shown are without any post processing of the class maps. Randen et al. [26] proposed a technique for unsupervised optimal feature extraction and segmentation of texture images. The image was first divided into cells of equal sizes and similarity measure on the autocorrelation function for the cells were estimated. The similarity



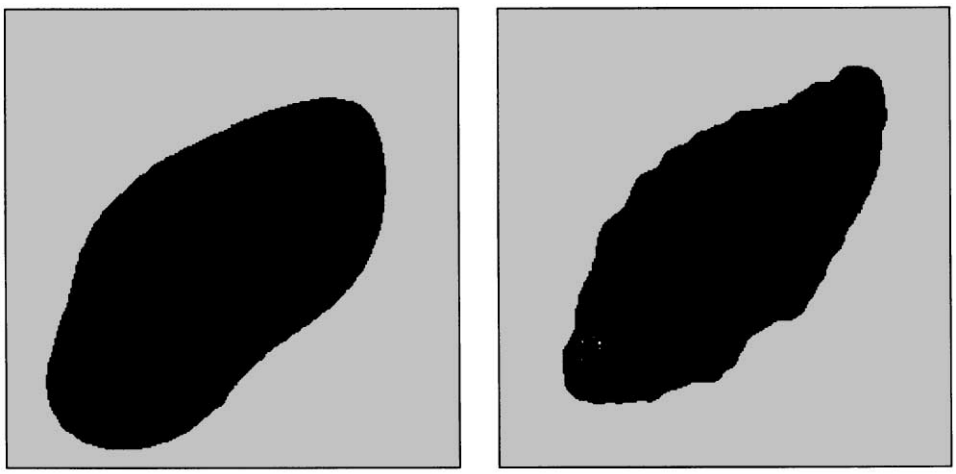
(a) (b)

Fig. 6. (a) Texture nat2. (b) Corresponding class map.



(a) (b)

Fig. 7. (a) Texture nat3. (b) Corresponding class map.

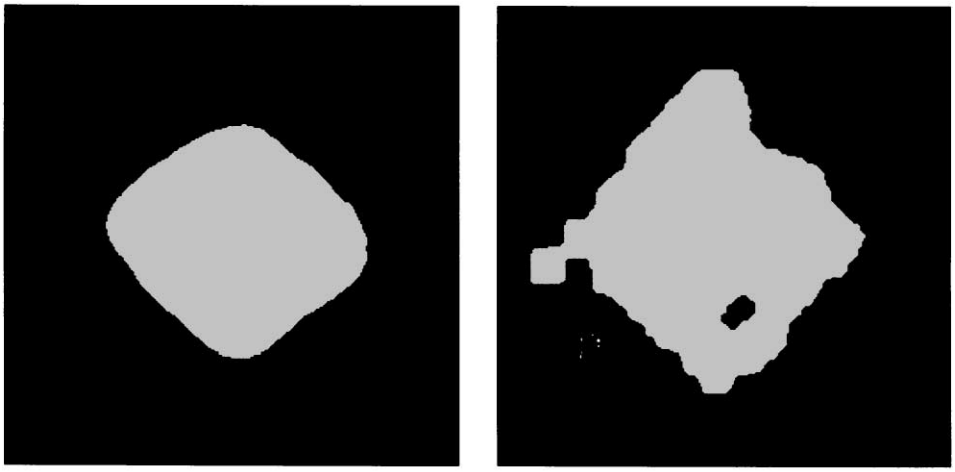


(a) (b)

Fig. 8. Unsupervised segmentation using (a) optimal Gabor filters by Teuner et al. [32] and (b) optimal filters by Randen et al. [26].

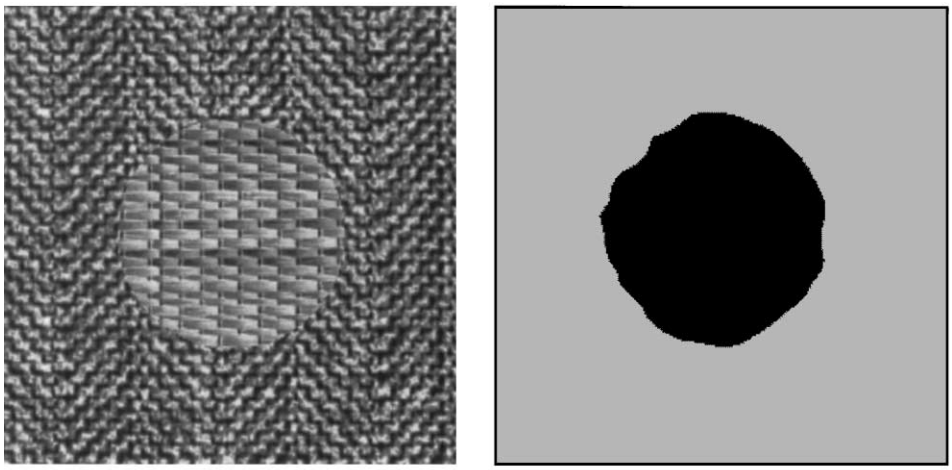
measures were used for clustering the image into clusters of cells with similar textures. Figs. 8 and 9 show the results obtained by Randen et al. [26] applied to the texture images Nat2 and Nat3 and also the results that had been obtained by applying the approach of Teuner et al. [32]. Two examples are by far not enough experimental data to judge which approach is best, but the figures clearly demonstrate that the approach we have proposed is better for these two images.

We have tested our segmentation scheme on several other textured mosaics. A representative set of images are *D*17*D*55, *D*5*D*92, *D*8*D*84, *D*12*D*17, *D*9*D*24. In Fig. 10 for the texture pair *D*17*D*55 of size 256×256 we took only 3 features, corresponding to the horizontal, vertical and diagonal frequencies, at second level of resolution ($\mathrm{Feat}_{\mathrm{H}_2}$, $\mathrm{Feat}_{\mathrm{V}_2}$ and $\mathrm{Feat}_{\mathrm{D}_2}$). The percentage of correctly classified pixels has been used as the segmentation quality measure in this work. We present a comparative



(a) (b)

Fig. 9. Unsupervised segmentation using (a) optimal Gabor filters by Teuner et al. [32] and (b) optimal filters by Randen et al. [26].



(a) (b)

Fig. 10. (a) Texture D17D55. (b) Corresponding class map.

performance evaluation of several approaches of texture segmentation found in the literature so far and that proposed by us with respect to the texture mosaic *D*17*D*55. Several approaches to multichannel filtering for texture segmentation have been proposed [9,8,27]. Farrokhnia [9] used a bank of Gabor filters with even symmetry and an octave band decomposition, the segmentation result of which is presented in Table 1. Although this

Table 1
Performance evaluation for texture mosaic *D*17*D*55

Method	Percent classification
Farrokhnia [9]	98.3%
Dunn [8]	91.6%
Randen [27] (*f*32*d*)	97.4%
Randen [27] (*F*_2_1_09)	97.0%
Proposed method	98.1%

method gives a good result, but it applies a large set of fixed filters, and consequently results in a high computational complexity. Dunn et al. [8] introduced a method for finding the optimum parameters to the Gabor filters to be used for a given pair of textures and achieved good computational savings. Randen Husϕy [27] proposed a set of critically sampled FIR filter '*f*32*d*' [19] and IIR filter '*F*_2_1_09' [15] for the purpose of segmentation. All the segmentation results for texture mosaic *D*17*D*55 are summarised in Table 1.

Although so far the best result is still obtained using the bank of Gabor filters in [9], the result we have obtained is very close to it.

When comparing the performance results, complexity issues should also be taken into account. The method presented in [9] gives good segmentation, but the most important drawback is its high computational complexity. An image of size 256×256 requires 28 feature images of the same size as the input image. The method introduced in [8] gives a large savings in computational cost, but with a modest degradation of performance compared to [9]. Instead of the 28 features used in [9], only 1 feature is required in [8], but this method has its weaknesses,

(i) The computational cost for calculating the optimal Gabor filter parameters is very high.

(ii) To find out the optimal parameters an a priori knowledge about the input textures is required. So for an unsupervised approach to texture segmentation this method cannot be applied.

For critically sampled filtering method in [27], there is a good savings in computational cost. But the feature dimension is 10, and also there is a degradation of the segmentation result. One major advantage of our scheme over other methods is that even though we have made use of overcomplete wavelet representation of images, which imply large feature space (i.e. 13 feature images of the same dimension as the input image), we have experimentally found that 3–5 features suffice for the desired segmentation and hence dimensionality of the feature space can be greatly reduced. So compared to the methods in [9,8,27] our method has reduced feature space and is computationally simple, while still maintaining the segmentation quality comparable to [9].

We present a comparative performance measure on two test images that is reported in [29]. Fig. 11

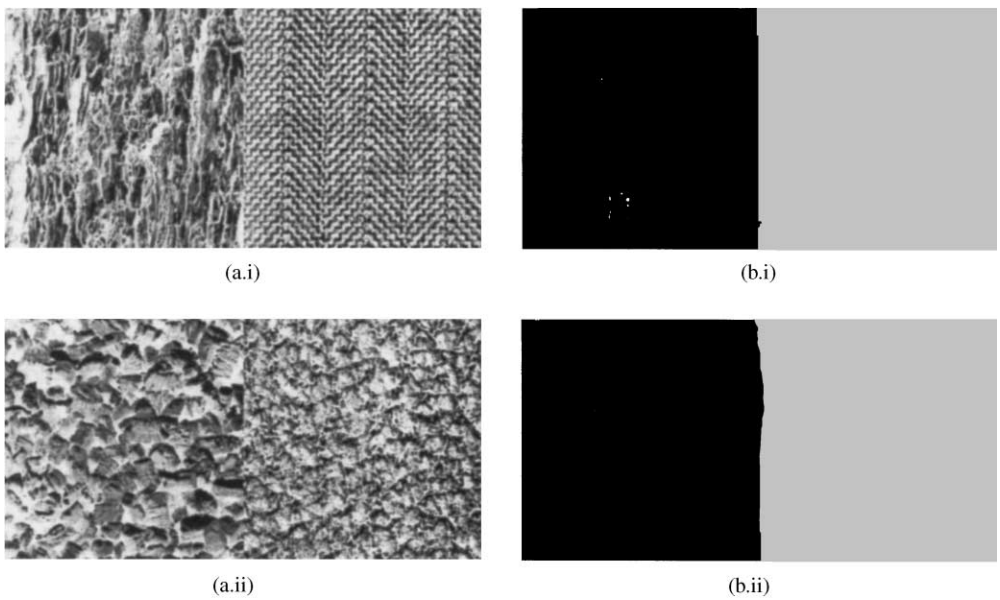


Fig. 11. (a) Texture image (a.i) *D*12*D*17, (a.ii) *D*5*D*92. (b) Class map of (b.i) *D*12*D*17, (b.ii) *D*5*D*92.

shows two of the texture pairs $D12D17$ and $D5D92$ of size 256×512 that we have worked on. Although Fig. 11a (ii) looks simple, it was very difficult to segment using different techniques that are available in the literature.

Table 3 in [29] summarizes the performance of several heuristically designed filter banks used as the feature extractors. If the performance results for the texture mosaics mentioned above are taken into consideration then discrete cosine transform (DCT) appears to have produced the best result. On the other hand if the average performance for several other texture mosaics is considered then the quadrature mirror filter (QMF) and wavelet frame approaches produce better results. Most of the heuristically designed filter banks have been reported to yield successful segmentation on several test images. But most of the filter banks imply large number of features and consequently high computational complexity. Therefore optimization of the filtering operation with respect to some explicit criterion related to texture classification is desirable. Randen and Husϕy [29] made extensive study on the performance of several optimized filter banks like, Gabor filter and finite impulse response (FIR) filters [19]. These were optimized following Mahalanobis and Singh (J_{MS}) [21], Unser (J_U) [35] and Fisher (J_F) [11] criterion. Table 6 in [29] gives the results of various approaches using several optimized filters, while Tables 8 and 9 summarize the performance results using full rate and critically sampled wavelet filters. In Table 2 we summarize the results achieved using the present method and those reported to have given the best results in [29], for a comparative study.

It is quite clear from the above discussion that we have been able to obtain high-quality segmentation.

For the filtering approaches, filtering and classification are the main contributors to the total complexity of the system. The heuristically designed filter (DCT) approach has low-filtering complexity (short separable filter masks with fast implementation schemes) and low feature dimensionality with number of features equaling 8. The optimized filtering approach has low feature count and consequently low computational complexity. But it is not possible for an unsupervised system, because for finding out the optimal parameters a first hand knowledge about the input images have to be known. The QMF filter banks have high filtering complexities. A 40-dimensional feature extractor puts high requirements on the system [29]. Whereas in our method for both of these texture pairs the number of features were limited to 3.

Table 2
Performance evaluation for texture pair $D12D17$ and $D5D92$

Methods/filters	Test figures (%)	
	11.a(i)	11.a(ii)
Heuristic DCT	97.8	97.5
Optimized w.r.t J_U/J_F	97.9	94.9
Full rate QMF $f32d$ (d)	99.5	93.2
Critically sampled QMF $f32d$ (d)	99.1	93.6
Proposed method M-band wavelet	99.7	99.0

Almost all the texture pairs were well discriminated by our algorithm. The spatial extent of the smoothing filter that have been used in our approach ranges from 11×11 to 31×31 depending on the spectral content of the images.

In order to prove the efficacy of our algorithm we have tested it over two other texture pairs of size 256×512 Fig. 12. Fig. 12a(i) $D8D84$ was tested by Jain and Karu [17] by the back propagation designed mask. The result obtained by their method is taken from [29] and given for a comparative study with our result. While $D9D24$ shown in Fig. 12a(ii) is visually difficult to discriminate. The segmentation accuracy for these two images are 99.3 and 99.5% respectively. In Fig. 13 we present the experimental results of a texture pair, which is also difficult to discriminate. We also give the texture feature images that have been used for this particular texture pair and it is quite evident from the figures that the two texture regions are appreciably discernible on the basis of features extracted. This is also reflected from the corresponding graphs which give the features averaged per column of the feature images. Very good results have been achieved in this case also which is evident from the class map. Simple morphological operation like median filtering of size (5×5) was applied to the class maps as a post-processing step.

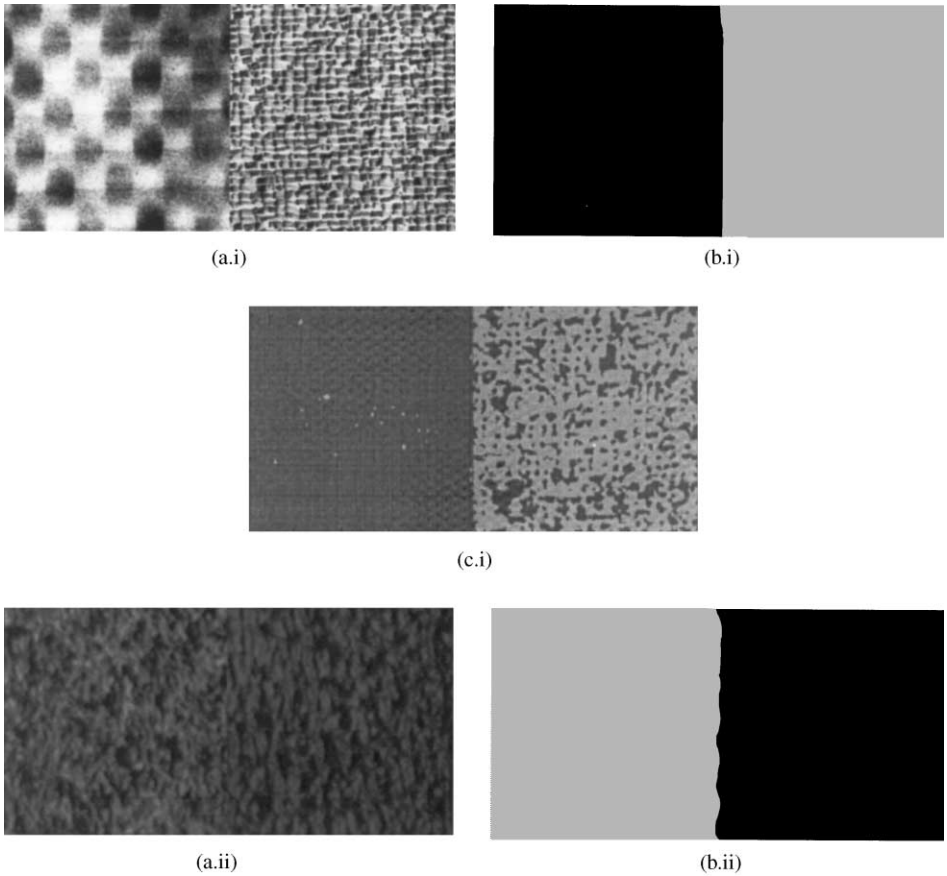


Fig. 12. (a) Texture image (a.i) *D*8*D*84, (a.ii) *D*9*D*24. (b) Class map of (b.i) *D*8*D*84, (b.ii) *D*9*D*24. (c.i) Segmentation result obtained in [17].

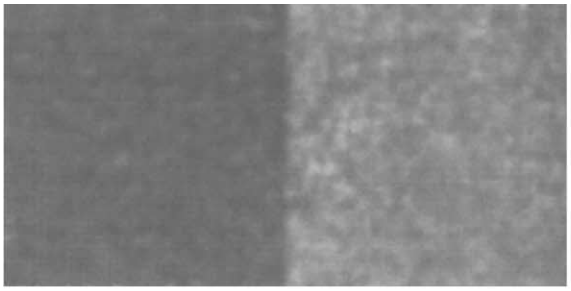
All the test images discussed so far have simple texture boundaries. It is to be noted that the feature extractors extract features using some window or neighborhood, so subsequently the complexity and shape of the boundary is of major interest. The ability to cope with texture boundaries is of course an important feature and the test images should be appropriate in this respect.

Our studies have been extended to images consisting of two textures but having complex boundaries Fig. 14. Performance measure for Fig. 14a(i) has been found to be 98.8% and for Fig. 14a(ii) is 93.8%.

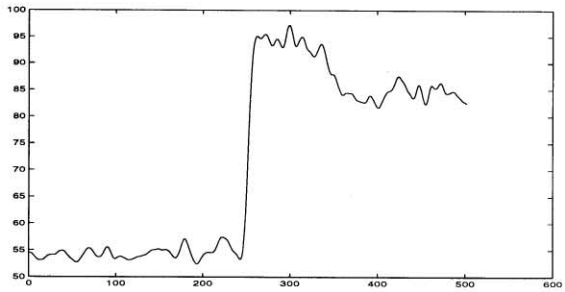
The proposed scheme has been examined over composite textures having rotated versions of the textures Fig. 15. The segmentation result is appealing in the sense that our algorithm is rotation invariant, the segmentation accuracy being 97.1%.

We have also worked on natural scenes, one such example is given in Fig. 16. We have segmented the image considering the two regions of sky and grass/plant as two different textures.

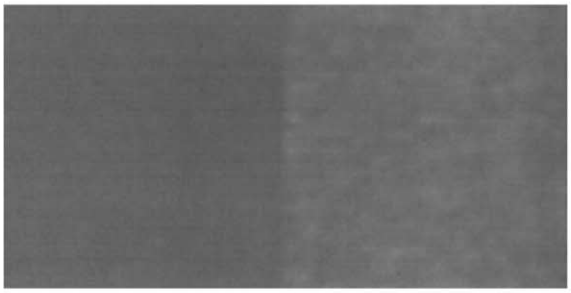
We have particularly focused our studies on two texture images, because there are several real world images comprising of two textures, one such example being document images. But we have also examined our algorithm over composite texture images comprising of a moderate number of textures Fig. 17. Fig. 17a(i) consists of four different texture classes and Fig. 17a(ii) consists of three texture classes with a more or less complex boundary. Classification percentage for these test images



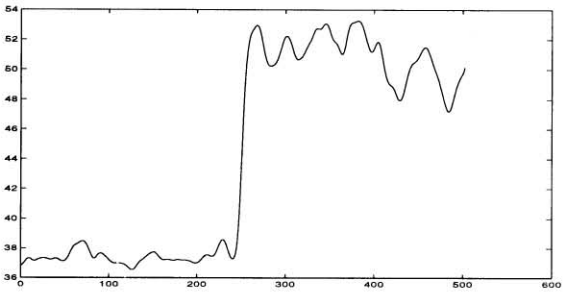
(a.i)



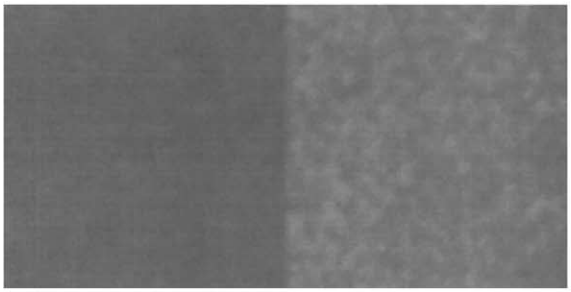
(a.ii)



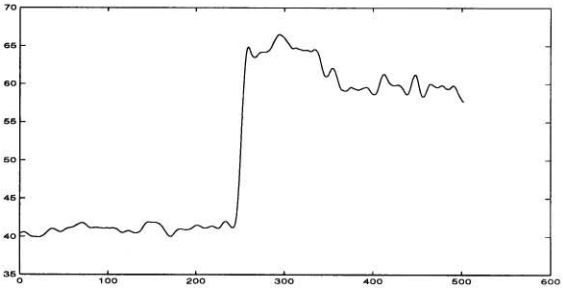
(b.i)



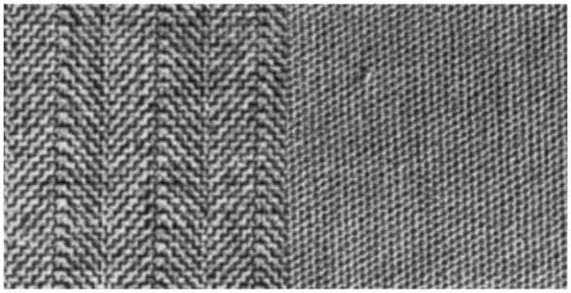
(b.ii)



(c.i)



(c.ii)



(d.i)

(d.ii)

Fig. 13. (a.i) Feature image corresponding to (a.ii) Feat_{H_2}, (b.i) Feat_{HD_2}, (c.i) Feat_{D_2}. Features averaged along columns in (a.ii) Feat_{H_2}, (b.ii) Feat_{HD_2}, (c.ii) Feat_{D_2}. (d.i) Texture image. (ii) Corresponding class map.

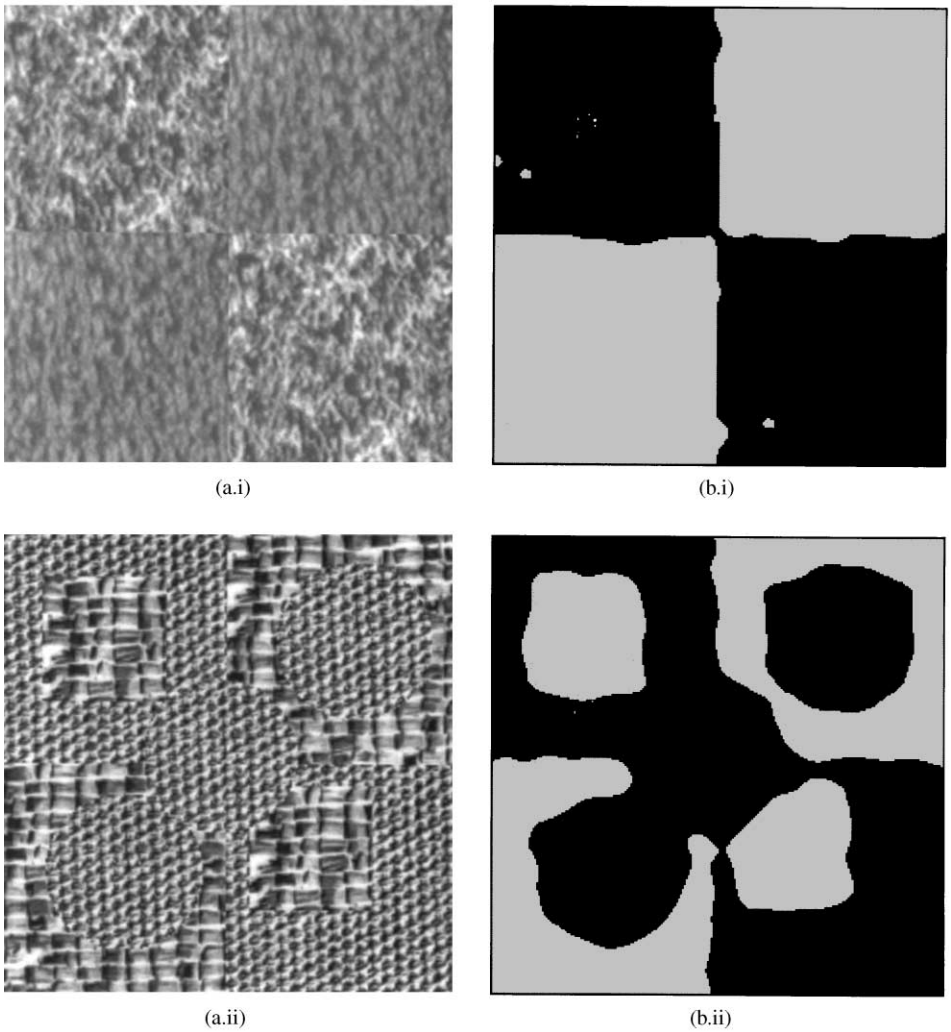


Fig. 14. (a.i)–(a.ii), Composite image consisting of two textures with complex boundaries. (b.i)–(b.ii), Corresponding class maps.

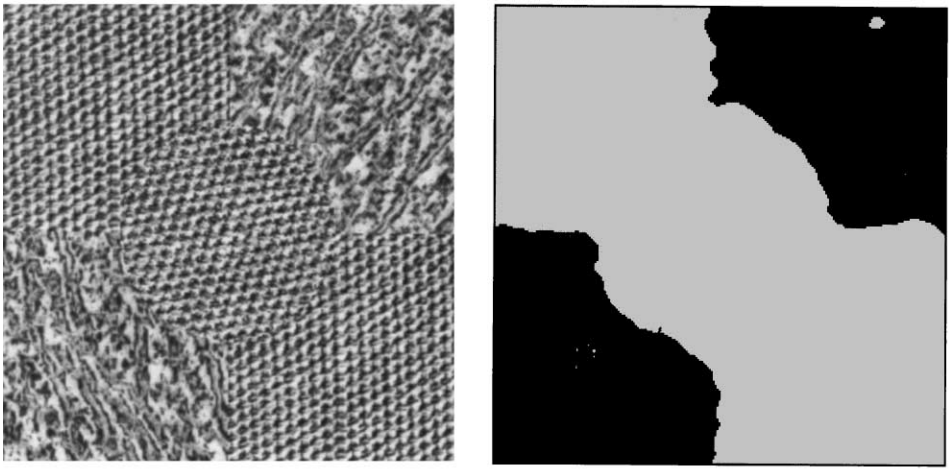
are 97.4 and 94.8%, respectively. This ensures that our algorithm works appreciably well for moderate number of textures also.

It is to be noted that for test Figs. 14–17 we have not done any post processing. While post processing like median filtering or morphological operation like dilation would have definitely improved our classification performance we opted to present our result without it to show how robust our scheme is under various conditions.

Another important point which is mention worthy is that except for the knowledge about the number of classes present in a composite image we otherwise do not have any a priori knowledge regarding the test images. That is we do not have any ground truth data and our scheme is completely unsupervised. While all the other approaches reported in [29] are supervised.

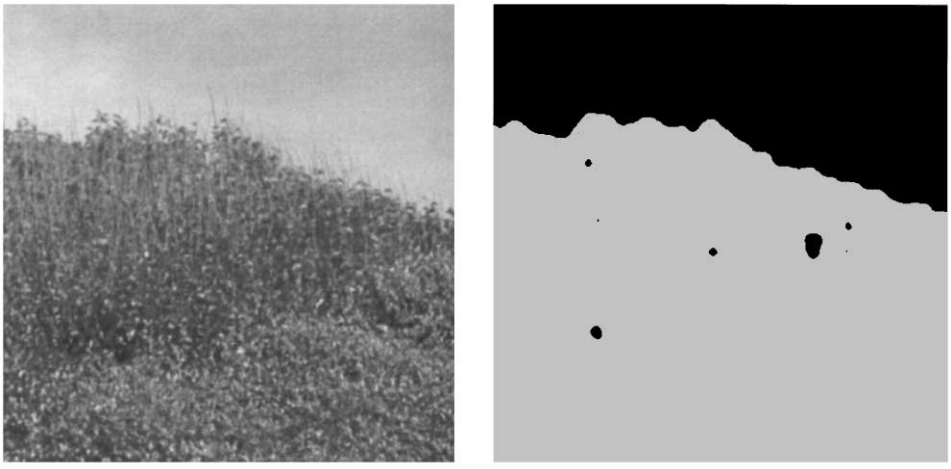
5. Conclusion

For good edge boundary localization, intuitively we would desire the filter to have a compact spatial



(a) (b)

Fig. 15. Composite image consisting of rotated texture and the corresponding class map.



(a) (b)

Fig. 16. A natural scene and the corresponding class map.

domain representation, while for reliable discrimination of different texture frequency contents the filter should have a good frequency response localization and high-stop band attenuation. The QMF filters have significantly compact frequency response. Also symmetry of the filter responses is an important factor. A nonsymmetric filter response would consistently lead to edge detection error and consequently higher classification error. The filter we have used has perfect reconstruction and quadrature mirror filter (PR-QMF) structure and is symmetrical.

In the case of standard dyadic wavelet decomposition, the low-frequency band of an image is split as the level of decomposition increases. Since we expect the textures to have their characteristic features in the higher frequency bands, we have proposed an M-band decomposition of the textured images which splits the lower as well as the higher-frequency bands.

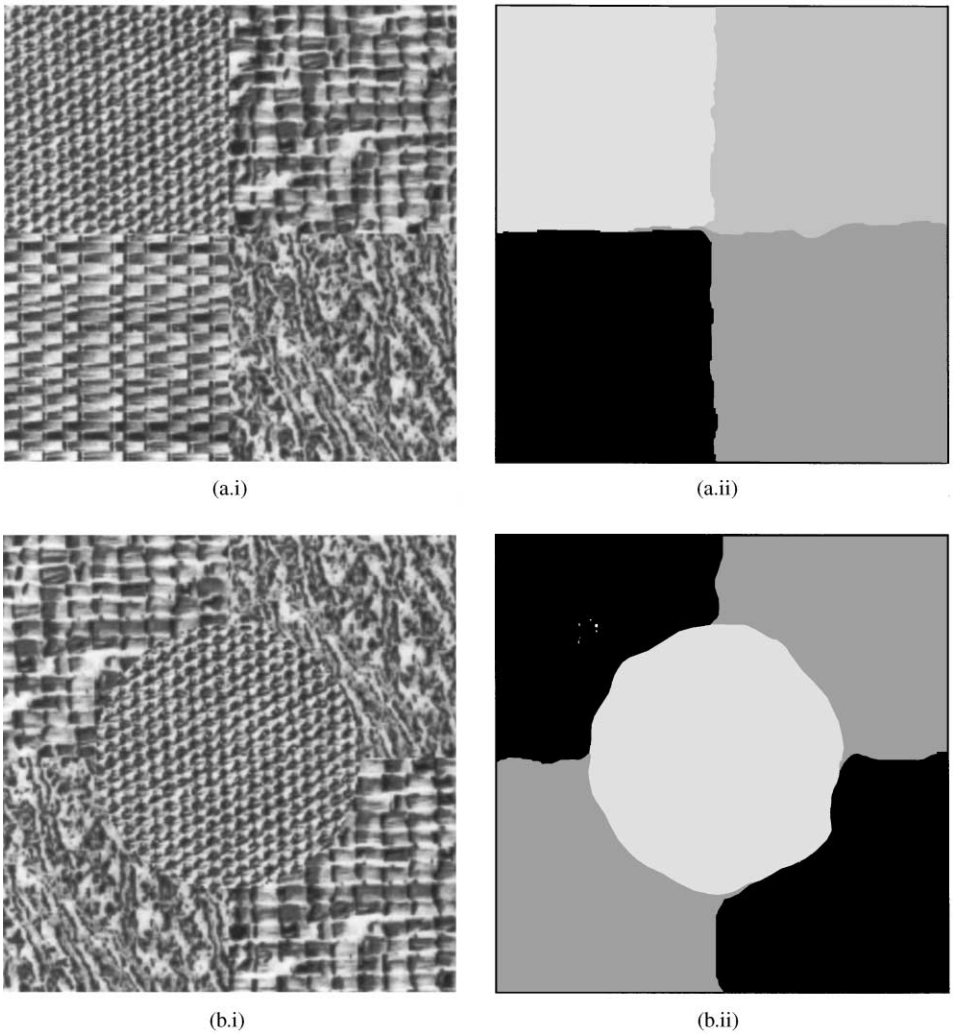


Fig. 17. (a.i) Composite image consisting of four textures; (a.ii) corresponding class maps. (b.i) Composite image consisting of three textures; (b.ii) corresponding class maps.

Here we have presented a multichannel filtering technique for texture segmentation using M-band wavelet transform in this work. It can be inferred that the use of M-band wavelet decomposition of the texture image gives an efficient representation of the image in terms of frequencies in different directions and orientations at different resolutions. This representation thus facilitates an improved segmentation of the different texture regions. The filtering and the feature extraction operations account for most of the required computations, however our method is very simple, computationally less expensive and efficient. It has been experimentally found that 3–5 features out of the 13 features are sufficient for good-quality segmentation. So dimensionality of the feature space is greatly reduced. Also since we have used an overcomplete wavelet representation of the textured images (i.e. without downsampling) translational invariance can also be achieved.

We have also worked on natural scenes the result shown is impressive. Although the major impetus

in this work has been the segmentation of two textured images, we have shown that the algorithm works appreciably well in case of images consisting of a moderate number of textures.

References

[1] O. Alkin, H. Caglar, Design of efficient M-band coders with linear phase and perfect reconstruction properties, IEEE Trans. Signal Process. 43 (7) (1995) 1579–1590.

[2] C.S. Burrus, A. Gopinath, H. Guo, Introduction to Wavelets and Wavelet Transform. A Primer, Prentice-Hall International Editions, Englewood Cliffs, NJ, 1998.

[3] T. Chang, C.C.J. Kuo, Texture analysis and classification with tree structured wavelet transform, IEEE Trans. Image Process. 2 (4) (1993) 42–44.

[4] Y. Chitre, A.P. Dhawan, M-band wavelet discrimination of natural textures, Pattern Recognition 32 (1999) 773–789.

[5] J.M. Coggins, A.K. Jain, A spatial filtering approach to texture analysis, Pattern Recognition Lett. 3 (3) (1985) 195–203.

[6] I. Daubechies, Orthogonal bases for compactly supported wavelets, Commun. Pure Appl. Math. 41 (1988) 909–996.

[7] I. Daubechies, Ten Lectures on Wavelets, Soc. Ind. Applied Math., Philadelphia, 1992.

[8] D.F. Dunn, W.E. Higgins, J. Wakeley, Determining Gabor-filter parameters for texture segmentation, Intell. Robots Comput. Vision XI 1826 (1992) 51–63.

[9] F. Farrokhnia, Multichannel filtering techniques for texture segmentation and surface quality inspection, Ph.D. Thesis, Michigan State university, 1990.

[10] F. Farrokhnia, A.K. Jain, A multichannel filtering approach to texture segmentation, Proceedings of Computer Vision and Pattern Recognition, 1991, pp. 364–370.

[11] K. Fukunaga, Statistical Pattern Recognition, 2nd Edition, Academic Press, New York, 1990.

[12] T. Greiver, J.P. Carel, M. Pandit, Texture analysis with a texture matched M-channel wavelet approach, IEEE International Conference on Accoustics Speech Signal Processing, Vol. 5, 1993, pp. V-129–V-132.

[13] R.M. Haralick, Statistical and structural approaches to texture, Proc. IEEE 67 (1979) 768–804.

[14] T. Hofmann, J. Puzicha, J.M. Buhmann, Unsupervised texture segmentation in a deterministic annealing framework, IEEE Trans. Pattern Anal. Mach. Intell. 20 (1998) 803–818.

[15] J.H. Husϕy, Subband coding of still images and video, Ph.D. Thesis, Norwegian Institute of Technology, 1991.

[16] A.K. Jain, F. Farrokhnia, Unsupervised texture segmentation using Gabor filters, Pattern Recognition 24 (12) (1991) 1167–1186.

[17] A.K. Jain, K. Karu, Learning texture discrimination masks, IEEE Trans. Pattern Anal. Mach. Intell. 18 (1996) 195–205.

[18] N.S. Jayant, P. Noll, Digital Coding of Waveforms: Principles and Applications to Speech and Video, Prentice-Hall Publications, Englewood Cliffs, NJ, 1984.

[19] J.D. Johnston, A filter family designed for use in quadrature mirror filter banks, Proceedings of International Conference on Acoustics Speech, Signal Processing IEEE, New York, 1980, pp. 291–294.

[20] A. Laine, J. Fan, Frame representation for texture segmentation, IEEE Trans. Image Process. 5 (5) (1996) 771–779.

[21] A. Mahalanobis, H. Singh, Application of correlation filters for texture recognition, Appl. Opt. 33 (11) (1994) 2173–2179.

[22] S. Mallat, A theory for multiresolution signal decomposition: the wavelet representation, IEEE Trans. Pattern Anal. Mach. Intell. 11 (7) (1989) 674–693.

[23] I. Ng, T. Tan, J. Kittler, On local linear transform and Gabor filter representation of texture, Proceedings of International Conference on Pattern Recognition, 1992, pp. 627–631.

[24] T. Ojala, M. Pietikainen, Unsupervised texture segmentation using feature distributions, Pattern Recognition 32 (1999) 477–486.

[25] W.H. Press, S.A. Teukolsky, W.T. Vellerling, B.P. Flannery, Numerical Recipes in C, Cambridge University Press, New Delhi, 1995.

[26] T. Randen, V. Alvestad, J.H. Husϕy, Optimal filtering for unsupervised texture feature extraction, Proceedings of Visual Communication and Signal Processing, Orlando, March 1996, pp. 441–452.

[27] T. Randen, J.H. Husϕy, Novel approaches to multichannel filtering for image texture segmentation, Proceedings of Visual Communication and Signal Processing, Boston, November 1993, pp. 626–636.

[28] T. Randen, J.H. Husϕy, Multichannel filtering for image texture segmentation, Opt. Eng. 33 (1994) 2617–2625.

[29] T. Randen, J.H. Husϕy, Filtering for texture classification: a comparative study, IEEE Trans. Pattern Anal. Mach. Intell. 21 (April 1999) 291–310.

[30] O. Rioul, A discrete time multiresolution theory, IEEE Trans. Signal Process. 41 (8) (August 1993) 2591–2606.

[31] P. Steffen, P.N. Heller, R.A. Gopinath, C.S. Burrus, Theory of regular M-band wavelet bases, IEEE Trans. Signal Process. 41 (12) (1993) 3497–3510.

[32] A. Teuner, O. Pichler, B.J. Hosticka, Unsupervised texture segmentation of images using tuned matched Gabor filters, IEEE Trans. Image Process. 4 (June 1995) 863–870.

[33] J.T. Tou, R.C. Gonzales, Pattern Recognition Principles, Addison-Wesley, Reading, 1974.

[34] M. Tuceryan, A.K. Jain, Texture analysis, in: C.H. Chen, L.F. Pau, P.S. Wang (Eds.), Handbook of Pattern Recognition and Computer Vision, World Scientific, Singapore, 1996, pp. 1–40.

[35] M. Unser, Local linear transforms for texture measurements, Signal Processing 11 (1) (1986) 61–79.

[36] M. Unser, Texture classification and segmentation using wavelet frames, IEEE Trans. Image Process. 4 (11) (1995) 1549–1560.

[37] H. Zou, A.H. Tewfik, Discrete orthogonal, M-band wavelet decomposition, Proceedings of ICASSP, Vol. 4, 1992, pp. IV-605–IV-608.