Texture Analysis and Classification with Tree-Structured Wavelet Transform

Tianhorng Chang and C.-C. Jay Kuo, *Senior Member, IEEE*

Abstract—One difficulty of texture analysis in the past was the lack of adequate tools to characterize different scales of textures effectively. Recent developments in multiresolution analysis such as the Gabor and wavelet transforms help to overcome this difficulty. In this research, we propose a multiresolution approach based on a modified wavelet transform called the tree-structured wavelet transform or wavelet packets for texture analysis and classification. The development of this new transform is motivated by the observation that a large class of natural textures can be modeled as quasi-periodic signals whose dominant frequencies are located in the middle frequency channels. With the transform, we are able to zoom into any desired frequency channels for further decomposition. In contrast, the conventional pyramid-structured wavelet transform performs further decomposition only in low frequency channels. We develop a progressive texture classification algorithm which is not only computationally attractive but also has excellent performance. The performance of our new method is compared with that of several other methods using the DCT, DST, DHT, pyramid-structured wavelet transforms, Gabor filters, and Laws filters.

I. INTRODUCTION

TEXTURES provide important characteristics for surface and object identification from aerial or satellite photographs, biomedical images, and many other types of images. Their analysis is fundamental to many applications such as industrial monitoring of product quality, remote sensing of earth resources, and medical diagnosis with computer tomography. Much research work has been done on texture analysis, classification, and segmentation for last three decades. Despite the effort, texture analysis is still considered an interesting but difficult problem in image processing.

Early research work [13], [25], [32], [33], [44] such as SGLDM [26] and correlation [7], [21] was based on the second-order statistics of textures. In the 1980's, researchers developed the Gaussian Markov random field (GMRF) [6], [9], [14], [20], [34], [35], [51] and Gibbs distribution [18], [19], [24], [37] texture models, where the gray levels between nearest neighboring pixels are characterized by a certain stochastic relationship. All the above methods share one common weakness. That is, they primarily focus on the coupling between image pixels on a single scale. Laws [38] proposed a simple scheme which used local linear transformations and

energy computation to extract texture features. The simple scheme often gives reasonably good performance, and it has been studied and improved by many researchers [11], [28], [29], [47], [48]. More recently, methods based on multichannel or multiresolution analysis have received a lot of attention [2], [4], [49]. They often outperform traditional methods based on the second-order statistics or the GMRF model. The Laws method turns out to be a multichannel method, which explains its good performance. This suggests that the main difficulty of traditional texture analysis is due to the lack of an adequate tool that characterizes different scales of textures effectively. Recent developments in spatial/frequency analysis such as the Gabor transform, Wigner distribution, [42] and wavelet transform provide good multiresolution analytical tools and should help to overcome this difficulty.

A large class of natural textures can be modeled as a quasi-periodic pattern and detected by highly concentrated spatial frequencies and orientations. Recent study of the human vision system indicates that the spatial/frequency representation [8], [10], [42], [50], which preserves both global and local information, is adequate for quasi-periodic signals. This observation has motivated researchers to develop multiresolution texture models. New algorithms such as methods with the Gabor filters and Wigner distribution have been proposed, and successful results have been reported in [3], [4], [22], [31], [42], [46]. A new spatial/scale analysis known as wavelet theory has been under intensive study during the last five years [15], [16], [27], [39], [40], [43], [45]. More recently, Coifman, Meyer, and Wickhauser [12] have generalized the wavelet basis function to include a library of modulated waveform orthonormal bases called wavelet packets. Since the wavelet and wavelet packet transform can be implemented efficiently with pyramid- and tree-structured algorithms, we call them pyramid- and tree-structured wavelet transforms, respectively. In this research, we apply the tree-structured wavelet transform to texture analysis and classification.

This paper is organized as follows. In Section II, we briefly review theory of wavelets and wavelet packets. The application of the tree-structured wavelet transform to texture analysis is described in Section II. Two texture classification algorithms are described in Section IV. The first algorithm uses a fixed number of features, whereas the second algorithm uses a minimum number of features sufficient for classification. Experimental results of texture classification are presented in Section V. Several different techniques such as DCT, DST, DHT, Gabor, Laws, and the pyramid-structured wavelet

transform are compared. Concluding remarks are given in Section VI.

II. REVIEW OF WAVELET TRANSFORM AND WAVELET PACKETS

Wavelet Transform

By wavelet transform, we mean the decomposition of a signal with a family of real orthonormal bases $\psi_{m,n}(x)$ obtained through translation and dilation of a kernel function $\psi(x)$ known as the *mother wavelet*, i.e.,

$$\psi_{m,n}(x) = 2^{-m/2}\psi(2^{-m}x - n) \qquad (1)$$

where m and n are integers. Due to the orthonormal property, the wavelet coefficients of a signal $f(x)$ can be easily computed via

$$c_{m,n} = \int_{-\infty}^{+\infty} f(x)\psi_{m,n}(x)dx$$

and the synthesis formula

$$f(x) = \sum_{m,n} c_{m,n}\psi_{m,n}(x)$$

can be used to recover $f(x)$ from its wavelet coefficients.

To construct the mother wavelet $\psi(x)$, we may first determine a *scaling function* $\phi(x)$, which satisfies the two-scale difference equation [15], [45]

$$\phi(x) = \sqrt{2}\sum_k h(k)\phi(2x - k). \qquad (2)$$

Then, the wavelet kernel $\psi(x)$ is related to the scaling function via

$$\psi(x) = \sqrt{2}\sum_k g(k)\phi(2x - k) \qquad (3)$$

where

$$g(k) = (-1)^k h(1 - k). \qquad (4)$$

The coefficients $h(k)$ in (2) have to meet several conditions for the set of basis wavelet functions in (1) to be unique, orthonormal, and have a certain degree of regularity [45]. Several different sets of coefficients $h(k)$ satisfying the above conditions can be found in the wavelet literature [15], [16], [39], [40] (see also Table I).

The coefficients $h(k)$ and $g(k)$ play a very crucial role in a given discrete wavelet transform. To perform the wavelet transform does not require the explicit forms of $\phi(x)$ and $\psi(x)$ but only depends on $h(k)$ and $g(k)$. Consider a J-level wavelet decomposition which can be written as

$$f_0(x) = \sum_k c_{0,k}\phi_{0,k}(x)$$

$$= \sum_k (c_{J+1,k}\phi_{J,k}(x) + \sum_{j=0}^{J} d_{j+1,k}\psi_{j+1,k}(x)) \qquad (5)$$

where coefficients $c_{0,k}$ are given and coefficients $c_{j+1,n}$ and $d_{j+1,n}$ at scale $j+1$ are related to the coefficients $c_{j,k}$ at scale j via

$$\begin{array}{rcl} c_{j+1,n} & = & \sum_k c_{j,k}h(k - 2n) \\ d_{j+1,n} & = & \sum_k c_{j,k}g(k - 2n) \end{array} \qquad (6)$$

where $0 \le j \le J$. Thus, (6) provides a recursive algorithm for wavelet decomposition through $h(k)$ and $g(k)$, and the final outputs include a set of J-level wavelet coefficients $d_{j,n}$, $1 \le j \le J$, and the coefficient $c_{J,n}$ for a low-resolution component $\phi_{J,k}(x)$. By using a similar approach, we can derive a recursive algorithm for function synthesis based on its wavelet coefficients $d_{j,n}$, $1 \le j \le J$, and $c_{J,n}$

$$c_{j,k} = \sum_n c_{j+1,n}h(k - 2n) + \sum_n d_{j+1,n}g(k - 2n). \qquad (7)$$

It is convenient to view the decomposition (6) as passing a signal $c_{j,k}$ through a pair of filters H and G with impulse responses $h(n)$ and $\tilde{g}(n)$ and downsampling the filtered signals by two (dropping every other sample), where $\tilde{h}(n)$ and $\tilde{g}(n)$ are defined as

$$\tilde{h}(n) = h(-n), \tilde{g}(n) = g(-n).$$

The pair of filters H and G correspond to the halfband lowpass and highpass filters, respectively, and are called the *quadrature mirror filters* in the signal processing literature. The reconstruction procedure is implemented by upsampling the subsignals c_{j+1} and d_{j+1} (inserting a zero between neighboring samples) and filtering with $h(n)$ and $g(n)$, respectively, and adding these two filtered signals together. Usually the signal decomposition scheme is performed recursively to the output of the lowpass filter $\tilde{h}$. It leads to the conventional wavelet transform or the so-called pyramid-structured wavelet decomposition.

The coefficients $h(k)$, $0 \le k \le 15$, of the Battle-Lemarié and the 16-tap Daubechies wavelet transforms are listed in Table I. The coefficients $h_b(k)$ of the Battle-Lemarié wavelet are symmetric, i.e., $h_b(k) = h_b(-k)$. Although $h_b(k)$ is an infinite duration sequence, it decays exponentially for large k. In contrast, the coefficients $h_d(k)$ of the 16-tap Daubechies wavelet have a compact support of length 16, i.e., $h_d(k) = 0$ if $k < 0$ or $k > 15$.

Wavelet Packets

The pyramid-structured wavelet transform decomposes a signal into a set of frequency channels that have narrower bandwidths in the lower frequency region. The transform is suitable for signals consisting primarily of smooth components so that their information is concentrated in the low frequency regions. However, it may not be suitable for quasi-periodic signals such as speech signals whose dominant frequency channels are located in the middle frequency region. To analyze quasi-periodic signals, the concept of wavelet bases has been generalized to include a library of modulated waveform orthonormal bases, called wavelet packet bases [12] or simply wavelet packets. The library of wavelet packet basis functions

TABLE I
WAVELET TRANSFORM FILTER COEFFICIENTS

	Battle-Lemarié	16-tap Daubechies
$h(0)$	0.766130	0.054416
$h(1)$	0.433923	0.312872
$h(2)$	-0.050202	0.675631
$h(3)$	-0.110037	0.585355
$h(4)$	0.032081	-0.015829
$h(5)$	0.042068	-0.284016
$h(6)$	-0.017176	0.000472
$h(7)$	-0.017982	0.128747
$h(8)$	0.008685	-0.017369
$h(9)$	0.008201	-0.044088
$h(10)$	-0.004354	0.013981
$h(11)$	-0.003882	0.008746
$h(12)$	0.002187	-0.004870
$h(13)$	0.001882	-0.000392
$h(14)$	-0.001104	0.000675
$h(15)$	-0.000927	-0.000117

$\{W_n\}_{n=0}^{\infty}$ can be generated from a given function W_0 as follows:

$$W_{2n}(x) = \sqrt{2} \sum_k h(k) W_n(2x - k)$$

$$W_{2n+1}(x) = \sqrt{2} \sum_k g(k) W_n(2x - k)$$

where the function $W_0(x)$ can be identified with the scaling function ϕ and W_1 with the mother wavelet ψ. Then, the library of wavelet packet bases can be defined to be the collection of orthonormal bases composed of functions of the form $W_n(2^l x - k)$, where $l, k \in Z, n \in N$. Each element of the library is determined by a subset of the indices: a scaling parameter l, a localization parameter k, and an oscillation parameter n. The tree-structured wavelet transform introduced in Section III in fact provides an algorithmic approach to represent a function in terms of a certain wavelet packet basis.

The 2-D wavelet (or wavelet packet) basis functions can be expressed by the tensor product of two 1-D wavelet (or wavelet packet) basis functions along the horizontal and vertical directions. The corresponding 2-D filter coefficients can be expressed as

$$h_{LL}(k,l) = h(k)h(l), \quad h_{LH}(k,l) = h(k)g(l),$$

$$h_{HL}(k,l) = g(k)h(l), \quad h_{HH}(k,l) = g(k)g(l)$$

where the first and second subscripts denote the lowpass and highpass filtering characteristics in the x- and y-directions, respectively.

III. TEXTURE ANALYSIS WITH TREE-STRUCTURED WAVELET TRANSFORM

The traditional pyramid-type wavelet transform recursively decomposes subsignals in the low frequency channels. However, since the most significant information of a texture often appears in the middle frequency channels, further decomposition just in the lower frequency region, such as the conventional wavelet transform does, may not help much for
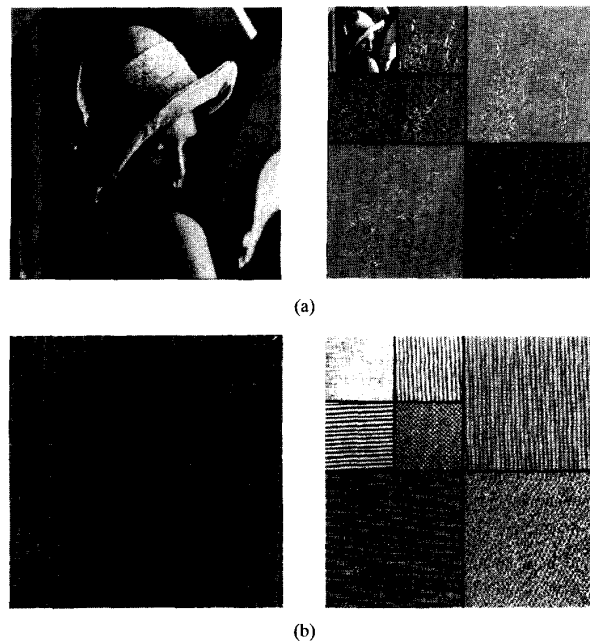

(a)

(b)

Fig. 1. Pyramid-structured wavelet transforms. (a) Lena. (b) French Canvas.

the purpose of classification. This concept can be illustrated in Fig. 1 where the pyramid-structured wavelet transform is applied to two different kinds of images. We use the Lena image as a representative for an ordinary image. The image and its pyramid-structured wavelet transform are shown in Fig. 1(a). The textured image French Canvas (D21 from the Brodatz's album [5]) and its pyramid-structured wavelet transform are shown in Fig. 1(b) for comparison. By examining the wavelet-transformed image in Fig. 1(a), we recognize the Lena image clearly from its low frequency channel (the upper left corner). In contrast, we are not able to recognize a similar texture pattern in the same low frequency channel for Fig. 1(b). Instead, we observe some horizontal and vertical line patterns clearly in the middle frequency region. The simple experiment implies that the low frequency region of textures may not necessarily contain significant information. Thus, an appropriate way to perform the wavelet transform for textures is to detect the significant frequency channels and then to decompose them further.

The above idea leads naturally to a new type wavelet transform called the tree-structured wavelet transform. The key difference between this algorithm and the traditional pyramid algorithm is that the decomposition is no longer simply applied to the low frequency subsignals recursively. Instead, it can be applied to the output of any filter h_{LL}, h_{LH}, h_{HL}, or h_{HH}. Note that it is usually unnecessary and expensive to decompose all subsignals in each scale to achieve a full decomposition. To avoid a full decomposition, we may consider a criterion to decide whether a decomposition is needed for a particular output. We use the averaged l_1-norm

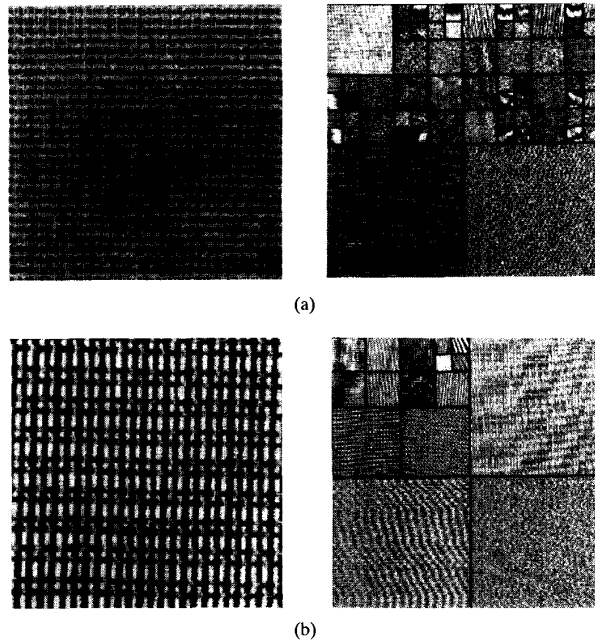$$e(\mathbf{x}) = \frac{1}{N} \|\mathbf{x}\|_1 = \frac{1}{N} \sum_{i=1}^{N} |x_i| \tag{8}$$

(a)

(b)

Fig. 2.   Tree-structured wavelet transforms. (a) French Canvas (D21).
(b) Oriental Straw Cloth (D53).

(a)

(b)

Fig. 3.   (a) Channel decomposition. (b) Quadtree representation of Oriental
Straw Cloth (D53).

where $\mathbf{x} = (x_1, \cdots, x_N)$, as the energy function to locate dominant frequency channels. Although there exist other norms such as the l_2-norm for the energy function, we find that the l_1 and l_2 norms make little difference in the final results. The energy function (8) is chosen due to its simplicity. The tree-structured wavelet transform is given below.
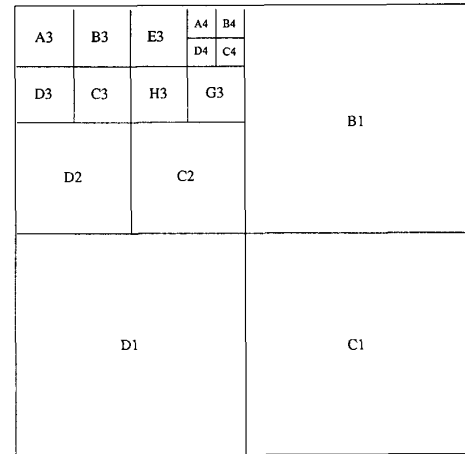
Algorithm 1: Tree-Structured Wavelet Transform

1) Decompose a given textured image with 2-D two-scale wavelet transform into 4 subimages, which can be viewed as the parent and children nodes in a tree.

2) Calculate the energy of each decomposed image (children node). That is, if the decomposed image is $x(m, n)$, with $1 \leq m \leq M$ and $1 \leq n \leq N$, the energy is

$$e = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} |x(m, n)|.$$

3) If the energy of a subimage is significantly smaller than others, we stop the decomposition in this region since it contains less information. This step can be achieved by comparing the energy with the largest energy value in the same scale. That is, if $e < C e_{max}$, stop decomposing this region where C is a constant less than 1.

4) If the energy of a subimage is significantly larger, we apply the above decomposition procedure to the subimage.

Practically, the size of the smallest subimages should be used as a stopping criterion for further decomposition. If the decomposed channel has a very narrow size, the location and the energy value of the feature may vary widely from sample
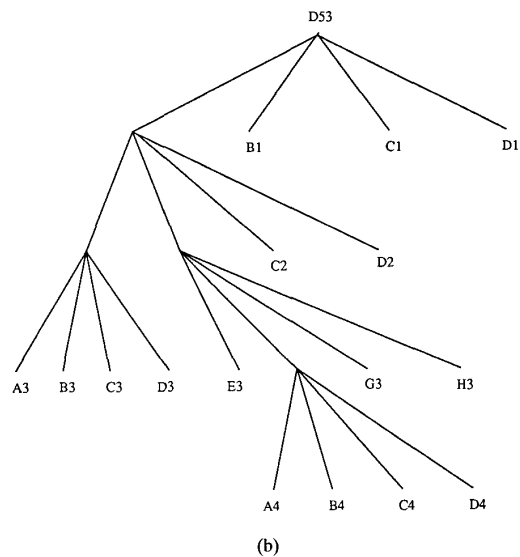
to sample so that the feature may not be robust. According to our experience, the size of the smallest subimages should not be less than 16×16. Consequently, if the input image size is 256×256 (or 64×64), a 4-level (or 2-level) tree-structured wavelet transform is appropriate. It is also worthwhile to point out that the above tree-structured wavelet transform provides a nonredundant representation, and it takes no more space to store the wavelet coefficients than it does to store the original image.

Two textured images of size 256×256 and their 4-level tree-structured wavelet decompositions with $C = 0.3$ are shown in Fig. 2. With respect to each tree-structured wavelet transform, we calculate the energy at its leaves, and obtain an energy function defined on the spatial/frequency domain called the *energy map*, which will be used for texture classification. Fig. 3 shows the energy map of texture D53. Note that the energy
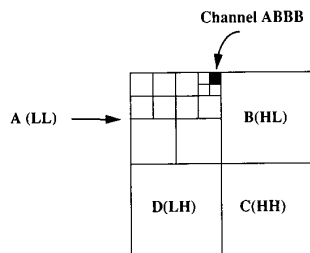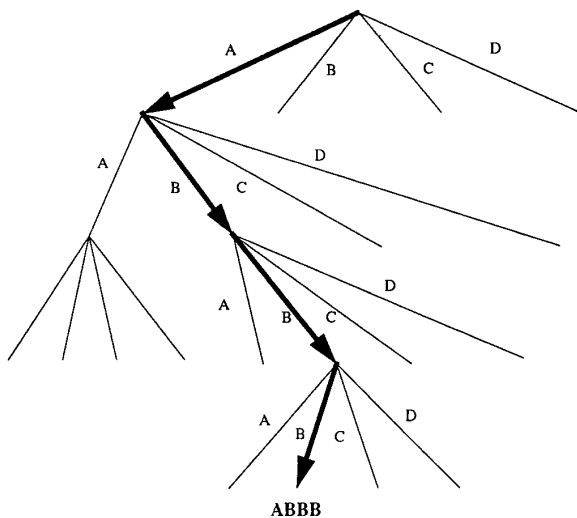
Fig. 4. The meaning of Channel ABBB in (a) tree-structured wavelet transform domain and (b) quadtree representation.

TABLE II
THE FIRST 5 DOMINANT CHANNELS FOR TEXTURE D53

type	number of occurrence	dominant frequency channels				
		first	second	third	fourth	fifth
1	49	ABBB	ABAB	BBBB	AABB	AACC
2	29	ABBB	ABAB	BBBB	AABB	AABC
3	14	ABBB	ABAB	BBBB	AACC	AABB
4	5	ABBB	ABAB	BBBB	AABC	AABB
5	3	ABBB	ABAB	BBBB	AACC	AABC

TABLE III
THE FIRST 5 DOMINANT CHANNELS FOR TEXTURE D21

type	number of occurrence	dominant frequency channels				
		first	second	third	fourth	fifth
1	38	ADAD	ADDD	BABA	BBBA	ABAB
2	20	ADAD	ADDD	BABA	BBBA	ABBB
3	17	ADDD	ADAD	BABA	BBBA	ABAB
4	8	ADDD	ADAD	BABA	BBBA	ABAB
5	7	ADAD	ADDD	BBBA	BABA	ABAB
6	5	ADDD	ADAD	BBBA	BABA	ABAB
7	4	ADAD	ADDD	BBBA	BABA	ABBB
8	1	ADDD	ADAD	BBBA	BABA	ABBB

map is in fact represented by the well known *quadtree* structure [30].

To investigate the robustness of the tree-structured wavelet decomposition, we applied the transform to 100 samples of French Canvas (D21) and Oriental Straw Cloth (D53) with $C = 0.15$. It turns out that all samples of texture D53 have exactly the same 4-level tree structure as given in Fig. 2(b). In contrast, there exist two 3-level tree structures occurring 84 and 16 times, respectively, for texture D21. With a further decomposition, we observe sixteen 4-level tree structures. Actually, the number of different tree structures depends highly on the constant C. For example, by decreasing the value of C, the tree-structure of texture D53 will not be the same for all samples. This observation suggests that the structure of the tree may not serve as a good candidate for texture representation.

However, if we focus on dominant frequency channels with large energy value, the representation becomes much more robust. Table II shows different kinds of energy maps with the first 5 dominant channels for 100 samples of texture D53 with threshold $C = 0.15$. The meaning of the frequency channel ABBB is described in Fig. 4. We see from Table II that the first 3 dominant channels are the same for all samples. Although

the 4th and 5th dominant frequency channels may differ, only three channels (AABB, AACC, and AABC) appear.

For texture D21 with $C = 0.15$, we can classify its energy maps into 8 different types according to the first 5 dominant channels as shown in Table III. Note that there are only six different channels appearing in the table. The first two dominant channels are either ADDD or ADAD, and the 3rd and 4th dominant channels are either BABA or BBBA.

The tree-structured wavelet transform generates a multiresolution/multichannel texture representation with complete basis functions which have a "reasonably well controlled"[12] spatial/frequency localization property. It is worthwhile to comment on the difference between the tree-structured wavelet transform and other transform-based texture analysis methods. The local linear transform methods such as DCT, DST, DHT, and the filtering-based methods such as Laws and Gabor filters use a fixed number of filter masks with predetermined frequencies and bandwidths. The set of Laws filter masks is basically determined by extensive experiments and may vary for different textures. In contrast, the tree-structured wavelet transform determines important channels dynamically according to energy calculation and can be viewed as an adaptive multichannel method.

One advantage of the proposed tree-structured wavelet transform over the conventional wavelet transform can be illustrated by an example given in Fig. 5, where texture D532 is a synthesized texture obtained from D53 by exchanging its tree-structured wavelet transform coefficients in channel ABB and those in channel ABD. These two textures are quite different to human eyes. However, the total energy in the dominant channel AB remains the same, so that the pyramid-structured wavelet transforms cannot distinguish these two textures. We need higher resolution in channel AB for their
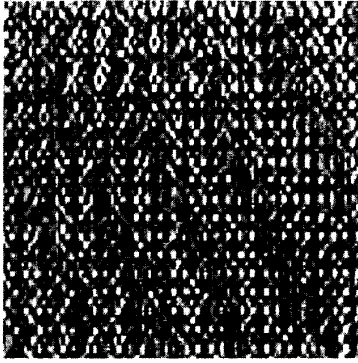
Fig. 5. A synthesized texture (D532) obtained from D53.

discrimination. This can be easily achieved by using the tree-structured wavelet transform. The method using Gabor filters with the constant-Q property [31] is in fact very close to the pyramid-structured wavelet transform in decomposing signals in the frequency domain. For detailed experimental comparison, we refer to Section V-B.

IV. TEXTURE CLASSIFICATION ALGORITHMS

Classification with Fixed Number of Features

A simple texture classification algorithm follows directly from Algorithm 1. The process is detailed as follows:

Algorithm 2: Classification Algorithm with J Features.

- *Learning phase*

  1) Given m samples obtained from the same texture, decompose each sample with the tree-structured wavelet transform and calculate the normalized energy at its leaves which defines an energy function on the spatial/frequency domain known as the energy map.
  2) Generate a representative energy map for each texture by averaging the energy maps over all m samples.
  3) Repeat the process for all textures.

- *Classification phase*

  1) Decompose an unknown texture with the tree-structured wavelet transform and construct its energy map.
  2) Pick up the first J dominant channels which are the leaf nodes in the energy map with the largest energy values as features. Denote this feature set by $\mathbf{x} = (x_1, \ldots, x_J)$.
  3) For texture i in the database, pick up the energy values in the same channels and denote the energy value by $\mathbf{m}_i = (m_{i,1}, \ldots, m_{i,J})$.
  4) Calculate the discrimination function for textures in the candidate list by

$$D_i = \text{distance}(\mathbf{x}, \mathbf{m}_i). \tag{9}$$

TABLE IV
FOUR DISTANCE FUNCTIONS FOR CLASSIFICATION

Euclidean Distance:	$D_{1,i} = \sum_{j=1}^{J} (x_j - m_{i,j})^2$
Bayes Distance:	$D_{2,i} = (x - m_i)^T C_i^{-1}(x - m_i) + \ln|C_i|$
Mahalanobis Distance:	$D_{3,i} = (x - m_i)^T C_i^{-1}(x - m_i)$
Simplified Mahalanobis Distance:	$D_{4,i} = \sum_{j=1}^{J} \frac{(x_j - m_{i,j})^2}{c_{i,j}}$

5) Assign the unknown texture to texture i if $D_i < D_j$ for all $j \neq i$.

Note that if the associated leaf node does not exist in the energy map of texture i in step 3, we may have the following two choices. Since the energy value of the parent node is the averaged energy value of its children nodes with the l_2 norm, we may use the energy value of its parent node as an approximation. It also implies that texture i is unlikely to be the unknown texture because the energy value of its corresponding channel is too small. Thus, another alternative is simply to discard texture i from the candidate list and test the next texture in the database.

The above algorithm uses the energy values at the J most dominant channels as features for classification. Thus, it implicitly assumes that high energy means better discriminability. Although this may not be necessarily true in general, the energy criterion has been widely used by research work in texture classification and segmentation. It also works well for all textures tested in our experiments given in Section V.

Several distance functions can be used in (9). In particular, we consider four such functions and list them in Table IV, where C_i is the covariance matrix of the feature set for texture i.

The Mahalanobis distance is a useful measure of similarity if some statistical properties of textures are known. In particular, if C_i is a diagonal matrix or, equivalently, features are independent of each other, the Mahalanobis distance reduces to the form

$$D_{4,i} = \sum_{j=1}^{J} \frac{(x_j - m_{i,j})^2}{c_{i,j}} \tag{10}$$

where $c_{i,j}$ is the variance of feature j and class i. For the case that C_i is not diagonal but diagonally dominant, (10) is still a good choice for distance measure. One special feature of (10) is that it can be computed recursively. That is, the distance $D_{4,i}^{(J+1)}$ using $J + 1$ features can be expressed in terms of the partial sum $D_{4,i}^{(J)}$ and the new feature x_{J+1} as

$$D_{4,i}^{(J+1)} = \sum_{j=1}^{J+1} \frac{(x_j - m_{i,j})^2}{c_{i,j}}$$
$$= D_{4,i}^{(J)} + \frac{(x_{J+1} - m_{i,J+1})^2}{c_{i,J+1}}. \tag{11}$$

Furthermore, if the joint density function of features is available, more sophisticated distance measures such as the one based on the Bayes decision rule can be used. By the law of large numbers, we may assume that the density function

of features is Gaussian so that the Bayes decision function assumes the form

$$D_2(x, m_i, C_i) = (x - m_i)^T C_i^{-1}(x - m_i) + \ln|C_i|.$$

Note that the Bayes distance is similar to the Mahalanobis distance for features with Gaussian distributions except for the addition of the second term known as the covariance difference.

Progressive Classification Algorithm

The above classification algorithm is effective for classifying textures with special dominant frequency channels. The experiment shows that textures of this type can be classified using only one or two features (see Section V). However, for the situation where textures have no significant dominant frequency channels such as random textures or where multiple textures have similar dominant frequency channels, we need more features for classification. It is difficult to determine the number of features required a priori. For example, if its first J dominant channels are close to several textures in the database, more than J features may be needed for its classification. This problem can be solved by a progressive classification algorithm where we start with one feature which eliminates very unlikely candidates and then another feature is added for further elimination. The procedure is repeated until there is only one texture left in the candidate list. The algorithm is detailed as follows:

Algorithm 3: Progressive classification algorithm

- *Learning phase*   (The same as Algorithm 2.)
- *Classification phase*

  1) Use the tree-structured wavelet transform to decompose an unknown texture and construct its energy map.
  2) Use the channels with large energy values x_j, $j = 1, 2, \cdots$, as features and arrange them in order, i.e., $x_1 > x_2 > \cdots > x_{j-1} > x_j > x_{j+1} > \cdots$.
  3) Order textures in the database into a candidate list, and perform the following iteration from the first feature (i.e., x_k, $k = 1$).

     i. Remove textures from the candidate list if they do not have the same leaf node as the kth dominant channel of the unknown texture.
     ii. For the remaining textures, denote the energy value and the variance in this channel by $m_{i,j}$ and $c_{i,j}$ for texture i and feature j, where $j \leq k$, and calculate the simplified Mahalanobis distance

     $$D_{4,i} = \sum_{j=1}^{k} \frac{(x_j - m_{i,j})^2}{c_{i,j}}.$$

     Let $D_{\min} = \min D_{4,i}$. If $D_{4,i} > K D_{\min}$, where K is a constant greater than one, remove texture i from the list.

TABLE V
TEXTURES USED IN THE EXPERIMENTS

Texture ID	Texture Description	Texture ID	Texture Description
D3	Reptile skin	D57	Handmade paper
D4	Pressed cork	D65	Handwoven oriental rattan
D6	Woven aluminum wire	D68	Wood grain
D9	Grass lawn	D74	Coffee beans
D11	Homespun woolen cloth	D77	Cotton canvas
D16	Herringbone	D78	Oriental straw cloth 3
D19	Woolen cloth with soft tufts	D79	Oriental grass fiber cloth
D21	French canvas	D82	Oriental straw cloth 4
D24	Pressed calf leather	D83	Woven matting
D29	Beach sand	D84	Raffia looped to a high pile
D34	Netting	D92	Pigskin
D36	Lizard skin	D95	Brick wall
D52	Oriental straw cloth 1	D102	Cane
D53	Oriental straw cloth 2	D103	Loose burlap
D55	Straw matting	D105	Cheesecloth

  iii. If there is only one texture left, assign the unknown texture to this texture. Otherwise, perform the next iteration by increasing the value of k by one.

In step 3-i, we discard the very unlikely textures because of poor similarity. The algorithm usually removes most irrelevant textures from the list in only one or two iterations, and all remaining textures are visually similar to the unknown texture. Although the number of features used for classifying a particular texture may be large, the computational complexity is in fact much lower than the one using a fixed number of features. The constant K in step 2-ii is a controllable parameter which serves as a threshold for eliminating irrelevant textures. Empirically, $K = 10$ gives a very satisfactory result. We can always decrease the false rejections by increasing the value of K. This will lead to a higher computational cost but more reliable result.

V. EXPERIMENTAL RESULTS

We used 30 textures obtained from the Brodatz's texture album[5] and have listed them in Table V. Each texture is scanned with 150 dpi resolution, and each image is of size 512×512 pixels with 256 gray levels. The mean of each image is removed before the processing.

A. Performance of Classification Methods with Tree-Structured Wavelet Transform

In this subsection, we demonstrate the performance of the two texture classification algorithms discussed in Section IV. One hundred sample images of size 256×256 are randomly chosen from the original image and used in training and classification phases. The mean and the covariance of the samples are calculated with the leave-one-out algorithm [23], [36] in classification.

TABLE VI
CLASSIFICATION RESULTS WITH FOUR DISTANCE FUNCTIONS

Texture ID	Correct Classification Rate			
	D_1	D_2	D_3	D_4
D3	88%	100%	100%	100%
D4	58%	96%	97%	95%
D6	98%	100%	100%	100%
D9	96%	97%	96%	91%
D11	100%	100%	100%	100%
D16	100%	100%	100%	100%
D19	100%	100%	100%	100%
D21	100%	100%	100%	100%
D24	100%	100%	100%	99%
D29	92%	100%	100%	99%
D34	76%	100%	100%	100%
D36	100%	100%	100%	100%
D52	96%	100%	100%	100%
D53	100%	100%	100%	100%
D55	99%	100%	100%	100%
D57	98%	100%	100%	100%
D65	100%	100%	100%	97%
D68	97%	99%	99%	98%
D74	100%	100%	89%	96%
D77	100%	100%	100%	100%
D78	82%	98%	100%	93%
D79	100%	100%	100%	100%
D82	100%	100%	100%	100%
D83	86%	99%	100%	98%
D84	100%	100%	100%	100%
D92	100%	100%	100%	100%
D95	98%	100%	100%	100%
D102	96%	100%	100%	100%
D103	100%	100%	100%	100%
D105	100%	100%	100%	100%
Overall	93.5%	99.6%	99.4%	98.9%

TABLE VII
AVERAGE NUMBER OF FEATURES USED IN PROGRESSIVE CLASSIFICATION

Texture ID	Numbers	Texture ID	Numbers	Texture ID	Numbers
D3	2.78	D34	4.08	D78	5.20
D4	7.78	D36	1.19	D79	3.14
D6	3.58	D52	2.65	D82	3.99
D9	8.96	D53	1.00	D83	5.68
D11	2.27	D55	2.87	D84	1.88
D16	1.00	D57	4.08	D92	5.50
D19	5.82	D65	5.98	D95	4.16
D21	1.00	D68	8.46	D102	1.59
D24	6.37	D74	9.68	D103	1.05
D29	6.68	D77	2.00	D105	1.31

coefficients $h_b(k)$, $-8 \leq k \leq 8$ is also observed. This indicates that the classification algorithm is insensitive to different wavelet bases in use except for the Haar basis.

Progressive Classification Although the covariance matrix of features is only diagonally dominant, the simplified Mahalanobis distance D_4 performs reasonably well according to Fig. 6. Since it can be computed recursively from the partial sum in the previous iteration as indicated in (11), the Mahalanobis distance D_4 with the independent feature assumption is used in the progressive classification algorithm. We list the average number of features in the progressive classification algorithm in Table VII with the Battle-Lemarié cubic spline wavelet basis. The 100% correct classification rate can be achieved with only 4.06 features on the average of all samples.

Textures that cannot be discriminated easily using 5 features in the previous experiment require more features in the progressive algorithm as well, since they are similar to other textures whose dominant frequencies are in the low frequency region. Despite more features being needed, the computational complexity is not as high as in the non-progressive algorithm. After two or three iterations, only a few textures which are similar to the unknown texture in the dominant frequencies remain in the candidate list. Thus, the overall computational complexity is still low. For textures significantly different from the others, only one or two features are needed.

We expect that the average number of features in the progressive algorithm will increase if we have more texture samples in the database, since the chance of several textures with similar dominant frequencies becomes higher.

B. Comparison With Other Transform-Based Methods

With data reported in this and next subsections, we performed experiments by partitioning each 512×512 texture image into 225 subimages of size 64×64 so that horizontally (or vertically) neighboring subimages overlap with each other with 32 columns (or rows).

We compared the tree-structured wavelet transform (TSWT) classification algorithm with other transform-based and filtering-based techniques such as DCT [30], [41], [47], DST [30], [41], [47], DHT [30], [47], pyramid-structured wavelet transform (PSWT) [15], [16], [27], [39], [40], [43], [45], Gabor filters [4], [31],[22], [46], and Laws filters [38]. For each
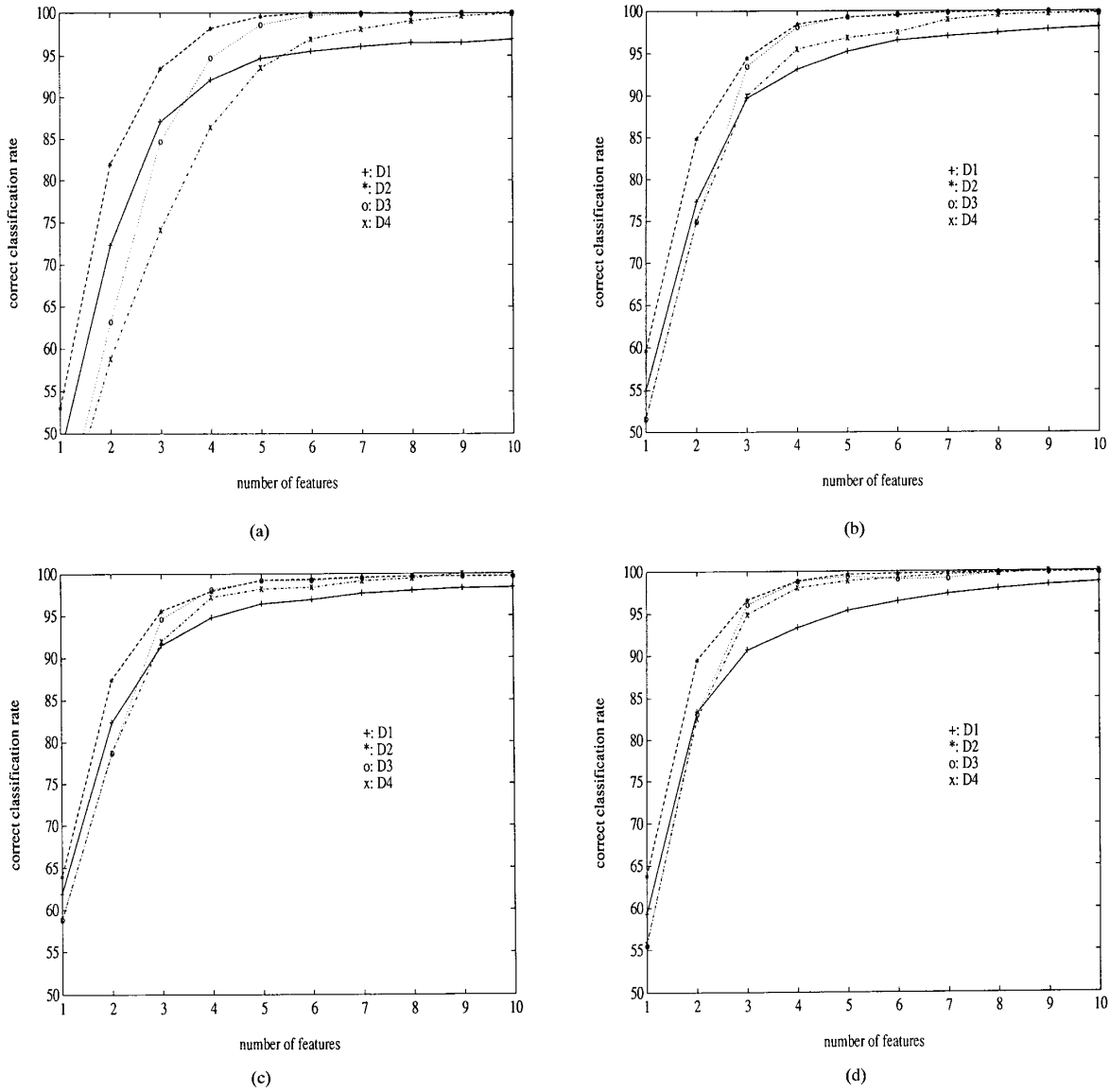
Classification with Fixed Number of Features We used the largest 5 dominant channels of the unknown texture as features, and classify the texture with 4 distance functions summarized in Table IV. Experimental results with Battle-Lemarié cubic spline wavelet basis functions are given in Table VI, where the average of classification rates of the 30 tested textures is called the overall correct classification rate and listed in the last row of the table.

Figure 6 shows the overall correct classification rate as a function of the number of features for different distance measures and wavelet bases. Distance D_1 has the worst performance for a large number of features, whereas distance D_2 has the best performance for a small number of features. Distances D_2, D_3, and D_4 give a similar performance when the number of features increases. This result implies that the correct classification rate will be higher by incorporating the statistical information of texture features. As far as different wavelet bases are concerned, the Haar basis gives the worst result while the other three bases achieve similar performance. A very similar performance for the 16-tap Daubechies wavelet and the truncated Battle-Lemarié cubic spline wavelet with

Fig. 6. Correct classification rates for different wavelet bases.

sample subimage of size 64×64, twenty-five 5×5 filter masks were used for DCT, DST and Laws filtering method (L5, E5, S5, W5, R5), sixteen 4×4 filters were used for DHT. We also implemented real-valued even-symmetric Gabor filters [31] by convolving a signal with a set of functions of the form

$$g(x, y) = \exp\left[-\frac{1}{2}(\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2})\right]\cos(2\pi ux).$$

Given the half-peak magnitude frequency bandwidth B_r in octaves and the orientation bandwidth B_o in degrees, parameters σ_x, σ_y are chosen by solving the following equations:

$$B_r = \log_2\left[\frac{u + (2\ln 2)^{1/2}\sigma_u}{u - (2\ln 2)^{1/2}\sigma_u}\right]$$

and

$$B_o = 2\tan^{-1}\left[\frac{(2\ln 2)^{1/2}\sigma_v}{u}\right]$$

where $\sigma_u = 1/2\pi\sigma_x$, $\sigma_v = 1/2\pi\sigma_y$, and parameter u is chosen so that the octave scale radial frequency is achieved. Different orientations of the filters can be obtained from a rotation of the x-y axis. Here we used four orientations ($0°$, $45°$, $90°$, and $135°$) and four radial frequencies ($2\sqrt{2}$, $4\sqrt{2}$, $8\sqrt{2}$, and $16\sqrt{2}$ cycles image-width $^{-1}$), where orientation and frequency bandwidths are chosen to be $45°$ and 1 octave, respectively. Thus, we obtained 16 Gabor filters in total. Finally, 3-level pyramid-structured and 2-level tree-structured
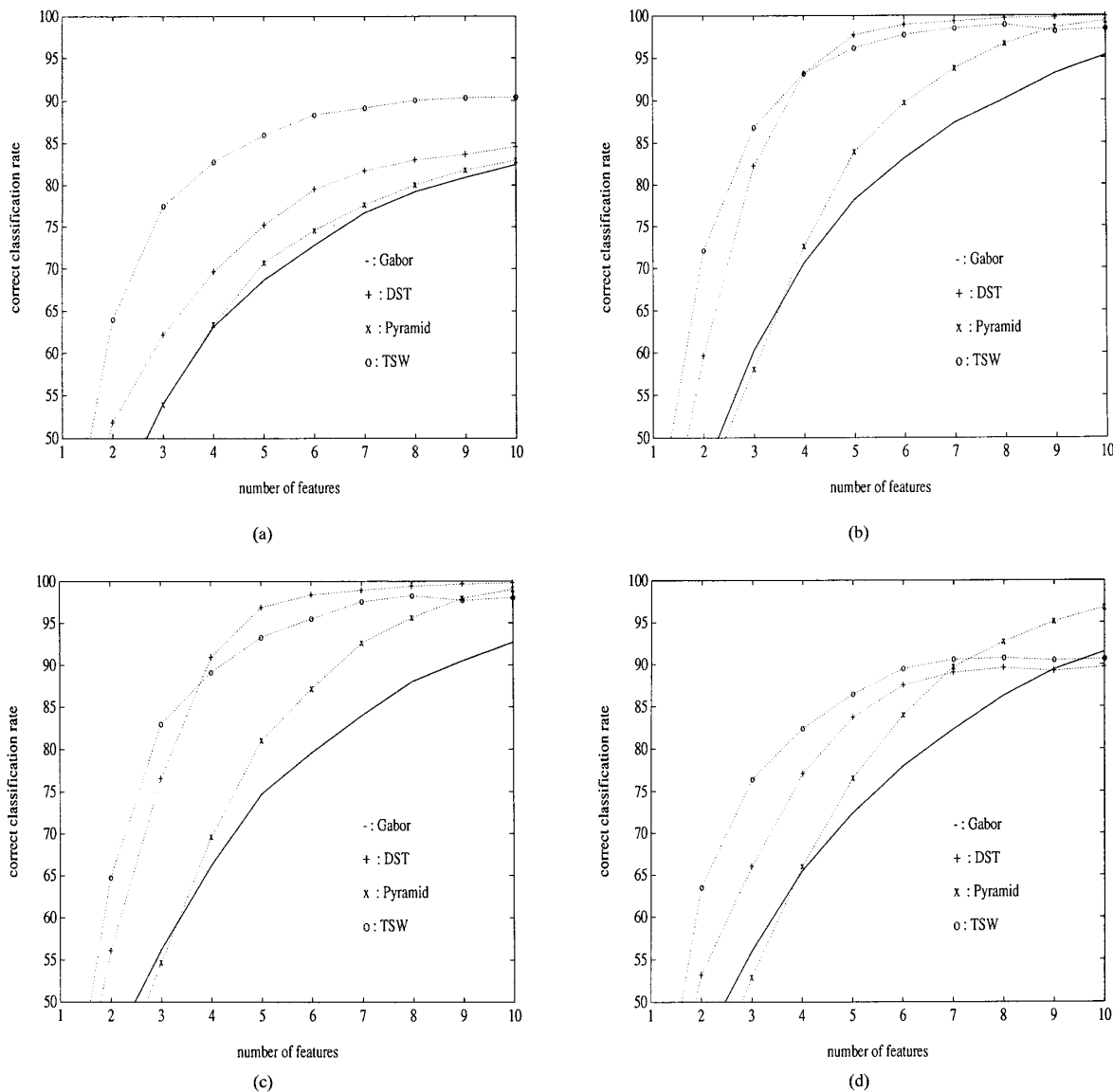
Fig. 7.   Comparison of experimental results using DST, Gabor filters, PSWT, and TSWT.

wavelet transforms were applied, where at least 10 frequency channels can be obtained.

Texture features were determined by computing the average energy in each channel or filtered image over all 225 samples for each texture. In the classification phase, only the largest k ($1 \leq k \leq 10$) features were used. We applied the leave-one-out algorithm and four distance measures to 225 sample subimages for each of 30 textures, and computed the average correct classification for each value of k. The experimental results are shown in Fig. 7. The results of the DCT, DHT, and Laws method are not plotted since they are very close to those of the DST. From these data, we have the following observations: First, the the Bayes distance gives the best performance, the Mahalanobis distance the second, the simplified Mahalanobis

distance the third, and the Euclidean distance the worst. Second, the performance of the Gabor filters in use is very close to that of the pyramid-structured wavelet transform. Their corresponding correct rates are usually the lowest regardless of the type of distance measures for small values of k. This may be explained by the fact that they both have the constant-Q property and suffer from the loss of spectral resolution in higher frequency regions. Third, the DCT, DST, DHT, Laws methods give very similar performance in all four distance measures. They are closely lying with each other. Finally, the tree-structured wavelet transform gives the best classification results for the Euclidean distance or for small values of k ($k \leq 4$). If $k \geq 4$, the performance of DCT, DST, DHT, Laws is similar to that of the tree-structured wavelet transform for

(d68)   (d105)   (d78)   (d79)

(d77)   (d21)   (d16)   (d53)
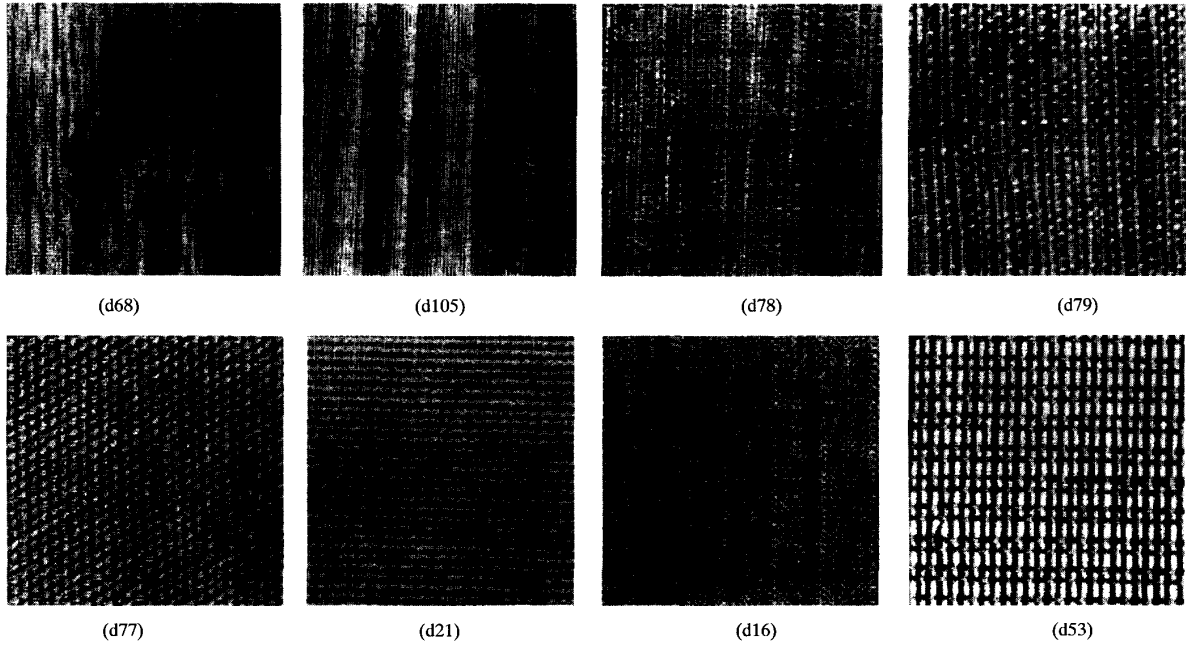
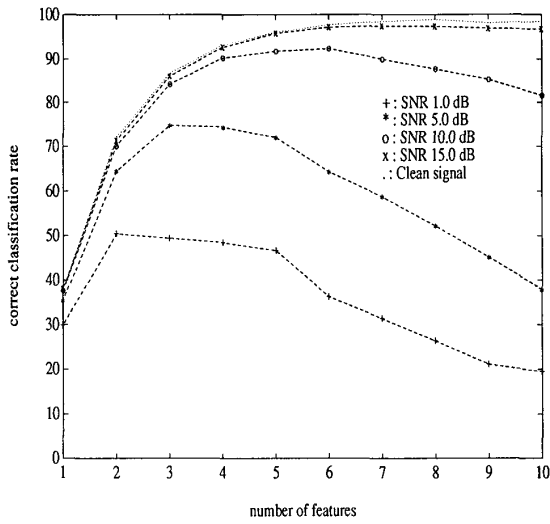Fig. 9.   A set of visually similar test textures.

Fig. 8.   Correct classification rate of TSWT with noisy data.
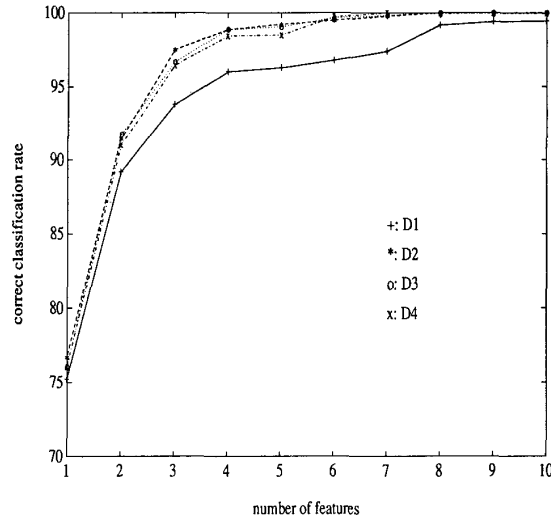

Fig. 10.   Correct classification rate of TSWT using test textures in Fig. 9.

the other three distance measures. We can conclude that the tree-structured wavelet transform in general provides better results over other tested techniques.

C. Sensitivity to Noisy Data

The sensitivity of the tree-structured wavelet transform classification algorithm to noisy data was tested by adding white noise to sample images before classification. The result is shown in Fig. 8. The classification rate is almost unaffected when the SNR is higher than 15 dB. However, for lower SNR,

we see that the correct rate decreases as the number of features increases. To explain this phenomenon, recall that we choose the first k largest average channel energy as features. Since the energy of white noise is spread over the entire spectrum, features from channels with small energy are sensitive to white noise and cause the performance deterioration.

D. Performance of TSWT on the Visually Similar Test Textures

We also examined the performance of our classification algorithm with respect to a set of 8 visually similar textures of

size 256 × 256 as shown in Fig. 9. The experimental results using Daubechies 8-tap bases are shown in Fig. 10. We see that the algorithm still performs quite well for this set of data. It gives a correction rate higher than 97% by using only 3 features with respect to three different distance measures.

VI. CONCLUSIONS AND EXTENSIONS

It is shown in this research that the tree-structured wavelet transform provides a good analytic tool for texture analysis. Although the conventional pyramid-structured wavelet transform or Gabor filters with the constant-Q property are suitable for images with energy concentrated in the low frequency region, the tree-structured wavelet transform is more natural and effective for textures which have dominant middle frequency channels. It is worthwhile to point out several distinct features of our method in comparison with the method using Gabor filters. First, the wavelet transform provides a set of frequency/space localization bases which can exactly reconstruct the original signal from wavelet coefficients, whereas the Gabor filters provide a set of "optimum joint bandwidth" [17] filters which can also be used for representation through the use of auxiliary functions [1]. Second, filters $h(n)$ and $g(n)$ used in the wavelet transform remain the same between two consecutive scales while the Gabor approach requires filters of different parameters. Third, the constant-Q Gabor filters (and the pyramid-structured wavelet transform) may not have sufficient resolution for signal analysis if the signal characteristics are in high frequency regions.

Our current work has so far focused on algorithmic development and experimental justification. More thorough theoretical analysis is expected in the future. The application of the tree-structured wavelet transform to texture segmentation is under our current investigation. The tree-structured wavelet transform (or wavelet packets) is also applicable to speech processing and other interesting problems.

ACKNOWLEDGMENT

REFERENCES

[1] M. Bastiaans, "Gabor's expansion of a signal into Gaussian elementary signals," Proc. IEEE, vol. 68, pp. 538–539, 1980.
[2] C. Bouman and B. Liu, "Multiple resolution segmentation of textured images," IEEE Trans. Pattern Anal. and Machine Intell., vol. 13, pp. 99–113, Feb. 1991.
[3] A. C. Bovik, "Analysis of multichannel narrow-band filters for image texture segmentation," IEEE Trans. Signal Processing, vol. 39, pp. 2025–2043, Sept. 1991.
[4] A. C. Bovik, M. Clark, and W. S. Geisler, "Multichannel texture analysis using localized spatial filters," IEEE Trans. Pattern Anal. and Machine Intell., vol. 12, Jan. 1990.
[5] P. Brodatz, Textures: A Photographic Album for Artists & Designers. New York: Dover, New York, 1966.
[6] R. Chellappa, "Two-dimensional discrete Gaussian Markov random field models for image processing," Pattern Recognition, vol. 2, pp. 79–112, 1985.

[7] P. C. Chen and T. Pavlidis, "Segmentation by texture using correlation," IEEE Trans. Pattern Anal. and Machine Intell., vol. 5, pp. 64–69, Jan. 1983.
[8] H.-I. Choi and W. J. Williams, "Improved time-frequency representation of multicomponent signals using exponential kernels," IEEE Trans. Acoust., Speech, and Signal Processing, vol. 37, pp. 862–871, June 1989.
[9] F. S. Cohen and D. B. Cooper, "Simple parallel hierarchical and relaxation algorithms for segmenting noncausal Markovian random fields," IEEE Trans. Pattern Anal. and Machine Intell., vol. 9, pp. 195–219, Mar. 1987.
[10] L. Cohen, "Time-frequency distributions—a review," Proc. IEEE, vol. 77, pp. 941–981, July 1989.
[11] P. Cohen, C. T. Ledinh, and V. Lacasse, "Classification of natural textures by means of two-dimensional orthogonal masks," IEEE Trans. Acoust., Speech, and Signal Processing, vol. 37, pp. 125–128, Jan. 1989.
[12] R. R. Coifman and M. V. Wickerhauser, "Entropy-based algorithms for best basis selection," IEEE Trans. Information Theory, vol. 38, pp. 713–718, Mar. 1992.
[13] R. W. Conners and C. A. Harlow, "A theoretical comparison of texture algorithm," IEEE Trans. Pattern Anal. and Machine Intell., vol. 2, pp. 204–222, May 1980.
[14] G. R. Cross and A. K. Jain, "Markov random field texture models," IEEE Trans. Pattern Anal. and Machine Intell., vol. 5, pp. 25–39, Jan. 1983.
[15] I. Daubechies, "Orthonormal bases of compactly supported wavelets," Communications on Pure and Applied Mathematics, vol. 41, pp. 909–996, Nov. 1988.
[16] I. Daubechies, "The wavelet transform, time-frequency localization and signal analysis," IEEE Trans. Information Theory, vol. 36, pp. 961–1005, Sept. 1990.
[17] J. G. Daugman, "Uncertainty relation for resolution in space," J. Opt. Soc. Am. A, vol. 2, pp. 1160–1169, 1985.
[18] H. Derin, "Segmentation of textured images using Gibbs random fields," Computer, Vision, Graphics, and Image Processing, vol. 35, pp. 72–98, 1986.
[19] H. Derin and H. Elliott, "Modeling and segmentation of noisy and textured images using Gibbs random fields," IEEE Trans. Pattern Anal. and Machine Intell., vol. 9, pp. 39–55, Jan. 1987.
[20] H. Derin, H. Elliott, R. Cristi, and D. Geman, "Bayes smoothing algorithms for segmentation of binary images modeled by Markov random fields," IEEE Trans. Pattern Anal. and Machine Intell., vol. 6, pp. 707–720, Nov. 1984.
[21] O. D. Faugeras and W. K. Pratt, "Decorrelation methods of texture feature extraction," IEEE Trans. Pattern Anal. and Machine Intell., vol. 2, pp. 323–332, July 1980.
[22] I. Fogel and D. Sagi, "Gabor filters as texture discriminator," Biological Cybernetics, vol. 61, pp. 103–113, 1989.
[23] K. Fukunaga, Introduction to Statistical Pattern Recognition. Boston: Academic Press, 1990.
[24] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," IEEE Trans. Pattern Anal. and Machine Intell., vol. 6, pp. 721–741, Nov. 1984.
[25] R. M. Haralick, "Statistical and structural approaches to texture," Proc. IEEE, vol. 67, pp. 786–804, May 1979.
[26] R. M. Haralick, K. Shanmugan, and I. Dinstein, "Textural features for image classification," IEEE Trans. SMC, vol. 8, pp. 610–621, Nov. 1973.
[27] C. E. Heil and D. F. Walnut, "Continuous and discrete wavelet transforms," SIAM Review, vol. 31, pp. 628–666, Dec. 1989.
[28] J. Y. Hsiao and A. A. Sawchuk, "Supervised textured image segmentation using feature smoothing and probabilistic relaxation techniques," IEEE Trans. Pattern Anal. and Machine Intell. , vol. 11, pp. 1279–1292, Dec. 1989.
[29] J. Y. Hsiao and A. A. Sawchuk, "Unsupervised textured image segmentation using feature smoothing and probabilistic relaxation techniques," Computer, Vision, Graphics, and Image Processing, vol. 48, pp. 1–21, 1989.
[30] A. K. Jain, Fundamentals of Digital Image Processing. Englewood Cliffs, NJ: Prentice Hall, 1989.
[31] A. K. Jain and F. Farrokhnia, "Unsupervised texture segmentation using Gabor filters," Pattern Recognition, vol. 24, no. 12, pp. 1167–1186, 1991.
[32] B. Julesz, "Visual pattern discrimination," IRE Transactions on Information Theory, vol. 8, pp. 84–97, Feb. 1962.
[33] B. Julesz, "Experiments in the visual perception of texture," Scientific American, vol. 232, pp. 765–771, June 1977.
[34] R. L. Kashyap and R. Chellappa, "Estimation and choice of neighbors in spatial-interaction models of images," IEEE Trans. Information Theory, vol. 29, pp. 60–72, Jan. 1983.

[35] R. L. Kashyap and R. Chellappa, "Texture synthesis using 2-D non-causal autoregressive models," *IEEE Trans. Acoust., Speech, and Signal Processing*, vol. 33, pp. 194–203, Jan. 1985.

[36] P. A. Lachenbruch, "Estimation of error rates in discriminant analysis," Ph.D. dissertation, Univ. of California, Los Angeles, 1965.

[37] S. Lakshmanan and H. Derin, "Simultaneous parameter estimation and segmentation of Gibbs random fields using simulated annealing," *IEEE Trans. Pattern Anal. and Machine Intell.*, vol. 11, pp. 799–813, Aug. 1989.

[38] K. I. Laws, "Texture image segmentation," Ph.D. dissertation, Image Processing Inst., Univ. of Southern California, 1980.

[39] S. G. Mallat, "Multiresolution approximation and wavelet orthonormal bases of L2," *Trans. Amer. Math. Soc.*, June 1989.

[40] S. G. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE Trans. Pattern Anal. and Machine Intell.*, vol. 11, pp. 674–693, July 1989.

[41] K. Rao and P. Yip, *Discrete Cosine Transform*. New York: Academic Press, 1990.

[42] T. R. Reed and H. Wechsler, "Segmentation of textured images and Gestalt organization using spatial/spatial-frequency representations," *IEEE Trans. Pattern Anal. and Machine Intell.*, vol. 12, pp. 1–12, Jan. 1990.

[43] O. Rioul and M. Vetterli, "Wavelets and signal processing," *IEEE SP Mag.*, pp. 14–38, Oct. 1991.

[44] A. Rosenfeld and L. S. Davis, "Image segmentation and image models," *Proc. IEEE*, vol. 67, pp. 764–773, May 1979.

[45] G. Strang, "Wavelets and dilation equations: a brief introduction," *SIAM Rev.*, vol. 31, pp. 614–627, Dec. 1989.

[46] M. R. Turner, "Texture discrimination by Gabor functions," *Biological Cybernetics*, vol. 55, pp. 71–82, 1986.

[47] M. Unser, "Local linear transforms for texture measurements," *Signal Processing*, vol. 11, pp. 61–79, 1986.

[48] M. Unser, "Sum and difference histograms for texture classification," *IEEE Trans. Pattern Anal. and Machine Intell.*, vol. 8, pp. 118–125, Jan. 1986.

[49] M. Unser and M. Eden, "Multiresolution feature extraction and selection for texture segmentation," *IEEE Trans. Pattern Anal. and Machine Intell.*, vol. 11, pp. 717–728, July 1989.

[50] R. Wilson and G. H. Granlund, "The uncertainty principle in image processing," *IEEE Trans. Pattern Anal. and Machine Intell.*, vol. 6, pp. 758–767, Nov. 1984.

[51] J. W. Woods, S. Dravida, and R. Mediavilla, "Image estimation using doubly stochastic Gaussian random field models," *IEEE Trans. Pattern Anal. and Machine Intell.*, vol. 9, pp. 245–253, Mar. 1987.

Tianhorng Chang received the B.S. degree from National Tsing-Hua University, Hsin-chu, Taiwan, and the M.S. degree from the University of Florida, Gainesville, in 1983 and 1988, respectively. He is currently pursuing the Ph.D. degree in electrical engineering at the Signal and Image Processing Institute at the University of Southern California.

Since 1990 he has been employed by the Signal and Image Processing Institute, University of Southern California, as a Research Assistant. His current research interests include digital signal and image processing, wavelet theory and applications, image analysis, and pattern recognition.

C.-C. Jay Kuo (S'83–M'86–SM'92) received the B.S. degree from the National Taiwan University, Taipei, in 1980 and the M.S. and Ph.D. degrees from the Massachusetts Institute of Technology, Cambridge, in 1985 and 1987, respectively, all in electrical engineering.

From October 1987 to December 1988, he was Computational and Applied Mathematics Research Assistant Professor in the Department of Mathematics at theUniversity of California, Los Angeles. Since January 1989, he has been with the Department of Electrical Engineering–Systems and the Signal and Image Processing Institute at the University of Southern California, where he is currently Assistant Professor. His research interests are in the areas of digital signal and image processing, wavelet theory and applications, video image compression, multimedia and large-scale scientific computing.

Dr. Kuo is a member of Sigma Xi, SIAM, SPIE, and ACM. He received the National Science Foundation Young Investigator Award in 1992,