Computer Vision and Image Understanding **79,** 25–46 (2000)
doi:10.1006/cviu.2000.0843, available online at http://www.idealibrary.com on IDEAL®

Optical and Sonar Image Classification: Wavelet Packet Transform vs Fourier Transform

Xiaoou Tang

Department of Information Engineering, The Chinese University of Hong Kong, Shatin, Hong Kong, China

and

W. Kenneth Stewart

South Florida Water Management District, West Palm Beach, Florida 33416

Received March 26, 1999; accepted February 4, 2000

To develop a noise-insensitive texture classification algorithm for both optical and underwater sidescan sonar images, we study the multichannel texture classification algorithm that uses the wavelet packet transform and Fourier transform. The approach uses a multilevel dominant eigenvector estimation algorithm and statistical distance measures to combine and select frequency channel features of greater discriminatory power. Consistently better performance of the higher level wavelet packet decompositions over those of lower levels suggests that the Fourier transform features, which may be considered as one of the highest possible levels of multichannel decomposition, may contain more texture information for classification than the wavelet transform features. Classification performance comparisons using a set of sixteen Vistex texture images with several level of white noise added and two sets of sidescan sonar images support this conclusion. The new dominant Fourier transform features are also shown to perform much better than the traditional power spectrum method. © 2000 Academic Press

1. INTRODUCTION

Sidescan sonar has been an important tool for seafloor survey over the past few decades. Due to the highly textured appearance of sonar images, texture analysis techniques become natural choices for sidescan sonar image analysis.

Texture analysis of sidescan sonar imagery can be applied to various geological feature recognitions. Pace and Dyer applied the co-occurrence features to distinguish the physical properties of sedimentary bottoms using sidescan sonar [23]. Reut *et al.* conducted an analysis of one-dimensional image spectra for the classification of six classes of homogeneous sediment type, including mud, clay, sand, gravel, cobbles, and boulders [26]. Pace and Gao continued this work and developed an alternative approach by extracting 1-D spectral features from sidescan data [24]. Reed and Hussong reported an application of the co-occurrence features to the classification of submarine lava flows, as well as the segmentation of lithified and nonlithified sedimentary formations [25]. Stewart *et al.* applied a neural network classifier to several traditional statistical texture features for the classification of terrain types of seafloor [27].

In this paper, we use two sidescan sonar data sets for texture classification experiments. They are the Arctic under-ice canopy data and the seafloor data. The first dataset is a sidescan sonar survey of an Arctic under-ice canopy. As the geophysical and economic importance of the polar seas becomes more widely recognized, the need for intensive study of these regions becomes more apparent. The underwater sidescan sonar imagery offers a unique perspective on the morphology of the under side of the ice, which is important to the study of frictional coupling between the ice cover and water and to research on the scattering of low-frequency acoustic energy by an ice canopy in an underwater communication channel.

Since thermal processes tend to level out rough elements much more slowly on the under-ice surface than on the top, the under-ice canopy tends to preserve many more features for the analysis of ice floe, ridge, and crack formation than the upper surface. Figure 1 shows a sidescan sonar image where ice cracks and ridges are clearly shown. This image is part

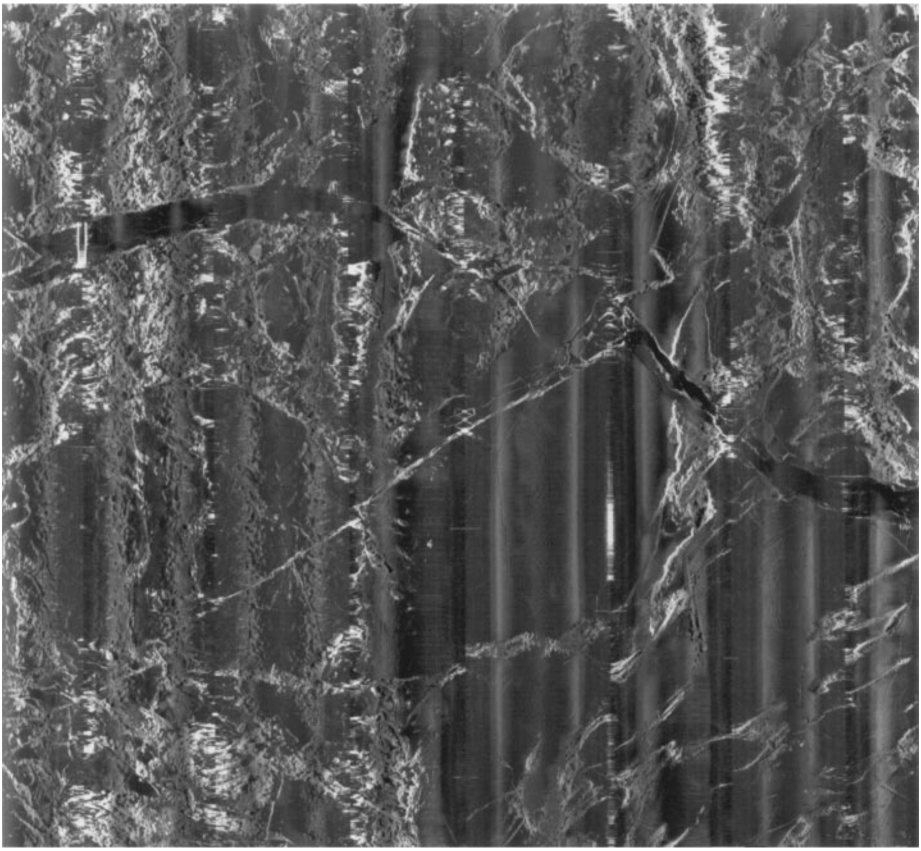

FIG. 1. Sidescan sonar image of an Arctic under-ice canopy.

of a novel set of sidescan sonar imagery of the Arctic under-ice canopy, obtained by an upward-looking sidescan sonar aboard a navy submarine. Using navigation data, several tracks of sonar data are mosaiced together to form the image. In this work, we study the classification of three types of ice textures in the image: first-year young ice (thin ice); multiyear undeformed ice (thick ice); and multiyear deformed ice (ice ridge and crack).

The second dataset is a sidescan sonar survey of the seafloor of a midocean ridge area. The region is characterized by hydrothermal venting and black smokers, geologic structures that vent extremely hot water and dissolved particulates from the seabed into the water column. In particular, our classification experiment focuses on three distinct geoacoustic provinces within the general area: a flat sediment pond, a lightly sedimented, constant-slope ridge flank, and an axial-valley segment.

Most research on texture analysis of sonar images use traditional statistical texture features directly. Recent development of the more efficient multichannel texture features has not been considered. Since sonar images tend to be noisier than optical images, we focus on the multichannel texture analysis approach, which is less sensitive to noise. Especially, we study methods using wavelet packet transformations and eventually arrive at a more straightforward method of feature extraction from the texture Fourier power spectrum, which is shown to be more efficient for both optical and sonar texture images.

Advances in wavelet theory [4, 7, 18–20] provide a good framework for multichannel texture analysis. The texture research community has devoted considerable effort to wavelet applications in texture analysis. An early study of a wavelet transform for texture analysis was described by Mallat [18]. Later, Henke-Reed and Cheng [12] applied a wavelet transform to texture images, using the energy ratios between frequency channels as features. Gabor filters have been used to extract texture features by Bigun *et al.* [2], du Buf *et al.* [8], and Jain *et al.* [13]. Chang and Kuo [3] developed a tree-structured wavelet transform algorithm for texture classification, which is similar to the wavelet-packet best-basis-selection algorithm of Coifman and Wickerhauser [4]. Laine and Fan [15] used both the standard wavelet and the wavelet-packet energy measures directly as texture features in their texture classification work.

These researchers have demonstrated that the wavelet transform is a valuable tool for texture analysis. However, a common problem with these approaches is that they are direct applications of existing wavelet processing algorithms, which were originally developed for signal representation or compression instead of signal classification. To fully utilize the power of a wavelet-packet transform, techniques tailored for extracting features of greater discriminatory ability must be considered. In this paper we demonstrate the use of the multilevel dominant eigenvector estimation algorithm and statistical distance measures to combine and select frequency-channel features that give improved classification performance.

Since the Fourier transform can be considered as one of the highest possible levels of multichannel decomposition, it is reasonable to apply the same feature selection algorithm to the Fourier transform spectrum. Just as the ideal tool for nonstationary signal analysis is a wavelet transform, the ideal tool for stationary signal analysis is a Fourier transform. Because texture signals are mostly stationary, we suspect that the Fourier transform features may generate better results.

In this study, we also compare the new Fourier features with the traditional power spectrum method (PSM), first studied by Bajcsy [1], Weszka *et al.* [31], and Jernigan *et al.* [14]. Historically, the texture classification performance of the PSM has been ranked fairly low among most texture analysis techniques [5, 31], resulting in limited application of this approach. Criticism of the PSM have been of the Fourier transform rather than of the way that texture features are computed from the power spectrum [31]. In this paper, we show that by using appropriate feature extraction algorithms, the discriminatory power of the Fourier transform features can be significantly improved.

Section 2 of this paper describes the texture feature extraction techniques, including the computation of wavelet features, the Fourier transform features, and the traditional PSM features. The texture classification experimental results on optical and sonar texture images are presented in Section 3. We summarize the conclusions in Section 4.

2. TEXTURE FEATURE EXTRACTION

2.1. Wavelet and Wavelet-Packet Transforms

For simplicity, a one-dimensional discrete signal $f(k)$ of length $n = 2^{n_0}$, where n_0 is an integer, is used for discussion in this section. The standard wavelet transform can be thought of as a smooth partition of the signal frequency axis. First, a lowpass filter $h(m)$ and a highpass filter $g(m)$, both of length M, are used to decompose the signal into two subbands, which are then downsampled by a factor of 2. Let H and G be the convolution-downsampling operators defined as

$$Hf(k) = \sum_{m=0}^{M-1} h(m) f(2k+m), \tag{1}$$

$$Gf(k) = \sum_{m=0}^{M-1} g(m) f(2k+m). \tag{2}$$

H and G are called perfect reconstruction quadrature mirror filters (QMFs) if they satisfy the orthogonality conditions

$$HG^* = GH^* = 0, \tag{3}$$

$$H^*H + G^*G = I, \tag{4}$$

where H^* and G^* are the adjoint (i.e., upsampling-anticonvolution) operators of H and G, respectively, and I is the identity operator.

This filtering and downsampling process is applied iteratively to the low-frequency subbands. At each level of the process, the high-frequency subband is preserved. When the process reaches the highest decomposition level, both the low- and high-frequency bands are kept. If the maximum processing level is L, the discrete wavelet coefficients of signal $f(k)$ are then $\{Gf, GHf, GH^2 f, \ldots, GH^L f, H^{L+1} f\}$ with the same length n as the original signal. Because of the orthogonality conditions of H and G, each level of transformation can be considered as a decomposition of the vector space into two mutually orthogonal subspaces. Let $V_{0,0}$ denote the original vector space R^n, and $V_{1,0}$ and $V_{1,1}$ be the mutually orthogonal subspaces generated by applying H and G to $V_{0,0}$. Then, the lth level of decomposition can be written as

$$V_{l,0} = V_{l+1,0} \oplus V_{l+1,1}, \tag{5}$$

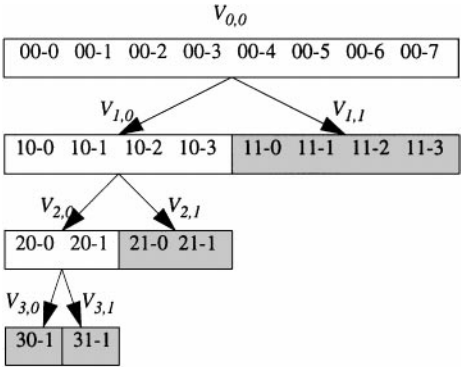

FIG. 2. Standard wavelet transform binary tree.

for $l = 0, 1, \ldots, L$. Figure 2 shows such a decomposition process. Each subspace $V_{l,b}$ with $b = 0$ or 1 is spanned by 2^{n_0-1} wavelet basis vectors $\{\psi_{l,b,c}\}_{c=0}^{2^{n0-l}-1}$, which can be derived from H, G, and their adjoint operators. From the above iterative filtering operations, we can see that the wavelet transform partitions the frequency axis finely toward the lower frequency region. This is appropriate for a smooth signal containing primarily low-frequency energy but not necessarily so for other more general types of signals, such as textures.

A more generalized form of the standard wavelet transform is the wavelet packet transform, which decomposes both the high- and low-frequency bands at each iteration. As with the wavelet transform, two subbands, Hf and Gf, are generated at the first level of decomposition. However, the second-level process generates four subbands, $H^2 f$, GHf, HGf, and $G^2 f$, instead of two bands, $H^2 f$ and GHf, as in the wavelet transform. If the process is repeated L times, Ln wavelet packet coefficients are obtained. In orthogonal subspace representation, the lth level of decomposition is

$$V_{l,b} = V_{l+1,2b} \oplus V_{l+1,2b+1}, \tag{6}$$

where $l = 0, 1, \ldots, L$ is the level index and $b = 0, \ldots, 2^l - 1$ is the channel block index in each level. Figure 3 illustrates the wavelet packet decomposition of the original vector

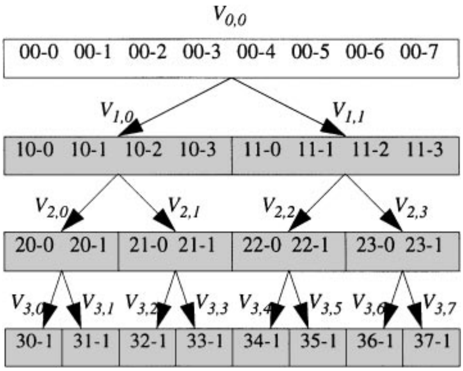

FIG. 3. Wavelet packet transform binary tree.

space $V_{0,0}$. Again, each subspace $V_{l,b}$ is spanned by 2^{n_0-l} basis vectors $\{W_{l,b,c}\}_{c=0}^{2^{n0-l}-1}$. For $b = 0$ and 1, W can be identified with ψ.

For two-dimensional images, the wavelet or wavelet packet basis function can be expressed as the tensor product of two one-dimensional basis functions in the horizontal and vertical directions. The corresponding 2-D filters are thus

$$h_{HH}(m, n) = h(m)h(n), \tag{7}$$

$$h_{HG}(m, n) = h(m)g(n), \tag{8}$$

$$h_{GH}(m, n) = g(m)h(n), \tag{9}$$

$$h_{GG}(m, n) = g(m)g(n). \tag{10}$$

In Fig. 4, we show three sample textures and their Daubechies minimum-support least asymmetric wavelet packet coefficients for levels 1–4.

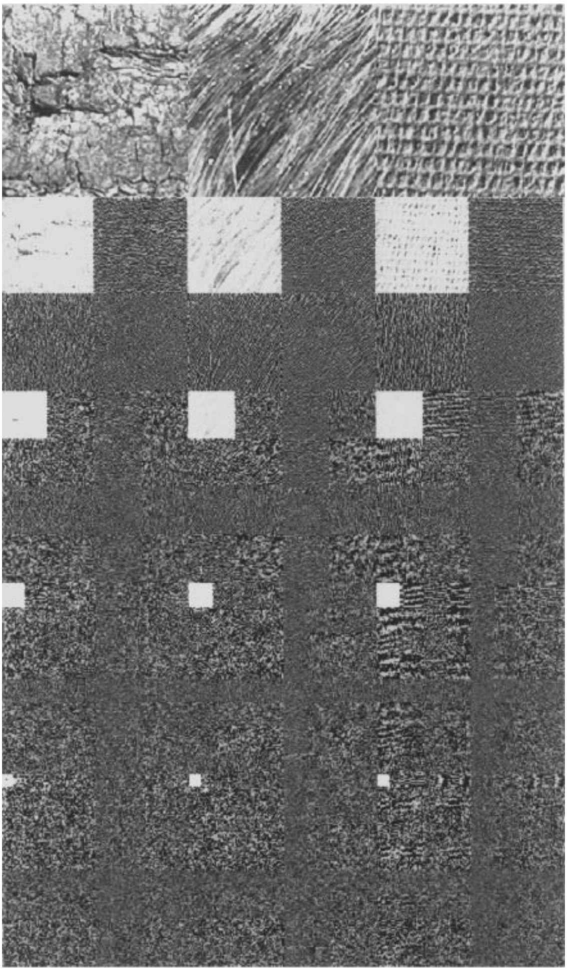

FIG. 4. Three sample textures (row 1) and their Daubechies minimum-support least asymmetric wavelet packet coefficients at decomposition levels 1, 2, 3, and 4 (rows 2–5).

2.2. *Wavelet Texture Feature Extraction*

After the wavelet packet coefficients are computed, we develop the algorithm by addressing the three main issues of multichannel texture classification: feature extraction within each channel, channel selection, and feature combination among channels.

Since the wavelet coefficients are shift variant, they are not suitable for direct use as texture features, which must be shift-invariant. We compute the shift-invariant measures

$$\mu_e = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N} e(i,\, j), \tag{11}$$

$$MNT_k = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N} (e(i,\, j) - \mu_e)^k, \tag{12}$$

$$ENT = -\sum_{i=1}^{M}\sum_{j=1}^{N} \frac{e(i,\, j)^2}{\|e\|_2^2}\log\left(\frac{e(i,\, j)^2}{\|e\|_2^2}\right), \tag{13}$$

where $e(i,\, j)$ denotes an element of the wavelet packet coefficient matrix e in each channel, μ_e is the mean of the matrix elements in the channel, and M and N represent the vertical and horizontal size of the matrix e whose norm is $\|e\|_2$. To make our algorithm less vulnerable to the nonuniform illumination of images, the texture sample mean is removed before a feature vector is computed. Thus, the mean feature in Eq. (11) becomes zero. The four features we use in the experiment are (1) variance feature VAR with $k = 2$ in (12), (2) the entropy feature ENT in Eq. (13), (3) the third moment MNT_3, and (4) the fourth moment MNT_4.

Because of the orthogonality condition on the wavelet transform, for the variance feature the following relation holds for any decomposition node and its four child nodes:

$$\mathrm{VAR}_{l,b} = \frac{1}{4}\sum_{j=0}^{3}(\mathrm{VAR}_{l+1,4b+j}). \tag{14}$$

The effect of this linear relationship on the classification accuracy of overcomplete wavelet packet features can be seen in later experiments.

After the features are computed within each channel, the second issue is how to select good features among channels. One possible approach is to apply a statistical distance measure to each feature and to select those features with large distance measures. However, there are two drawbacks with this approach. The first is that neighborhood channel features tend to correlate with each other; thus, they contain similar information. If one has a large distance measure, the other will also, and both will be selected. Therefore, similar features will usually be selected. The second problem is that for some very small energy channels, a small amount of unexpected noise may cause the distance measure to be large and the channel to be selected. To avoid these problems, we combine the channel selection step with the channel combination step in one feature selection step, using principal component analysis and statistical distance measures.

The widely used Karhunen–Loeve transform (KLT) is an appropriate feature reduction and selection procedure for our algorithm [10, 29]. The KLT decorrelates neighborhood channel features, and its energy-packing property removes noisy channels and compacts useful information into a few dominant features. However, for a large feature vector, such as a vector comprising features of a higher level wavelet packet decomposition or a Fourier transform, the computation of the eigenvectors of the covariance matrix can be prohibitively expensive. We use the multilevel dominant eigenvector estimation (MDEE) method [28] to overcome this problem. The following is a brief description of the MDEE method.

To compute the Karhunen–Loeve transform, let x_i be a feature vector sample and form an n by m matrix,

$$A = \begin{bmatrix} x_1(1) & x_2(1) & \dots & x_m(1) \\ x_1(2) & x_2(2) & \dots & x_m(2) \\ \dots & \dots & \dots & \dots \\ x_1(n) & x_2(n) & \dots & x_m(n) \end{bmatrix}, \tag{15}$$

where n is the feature vector length and m is the number of training samples. The sample covariance matrix is computed by

$$W = \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu)(x_i - \mu)^T = \frac{1}{m}AA^T, \tag{16}$$

where μ is the mean vector. For the standard KLT, the eigenvalues and eigenvectors are computed directly from W. However, with the feature vector formed by the higher level wavelet-packet decomposition features or the scan-line vectorized Fourier transform matrix, n is a large number. For a neighborhood of 64×64, n can reach a maximum of 4096, resulting in a covariance matrix of size 4096×4096. Direct computation of the eigenvalues and eigenvectors becomes impractical.

To alleviate this problem, we use the multilevel dominant eigenvector estimation method. After breaking the long feature vector into $g = n/k$ groups of small feature vectors of length k,

$$A = \begin{bmatrix} B_1 \left\{ \begin{bmatrix} x_1(1) & x_2(1) & \dots & \dots & x_m(1) \\ \dots & \dots & \dots & \dots & \dots \\ x_1(k) & x_2(k) & \dots & \dots & x_m(k) \end{bmatrix} \right\} \\ B_2 \left\{ \begin{bmatrix} x_1(k+1) & x_2(k+1) & \dots & \dots & x_m(k+1) \\ \dots & \dots & \dots & \dots & \dots \\ x_1(2k) & x_2(2k) & \dots & \dots & x_m(2k) \end{bmatrix} \right\} \\ \begin{matrix} \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \end{matrix} \\ B_g \left\{ \begin{bmatrix} x_1((g-1)k+1) & x_2((g-1)k+1) & \dots & \dots & x_m((g-1)k+1) \\ \dots & \dots & \dots & \dots & \dots \\ x_1(n) & x_2(n) & \dots & \dots & x_m(n) \end{bmatrix} \right\} \end{bmatrix} \tag{17}$$

we apply the KLT to each of the g group short feature vector set B_i. Then a new feature vector is constructed by the first few dominant eigenfeatures of each group. The final eigenvectors are computed by applying the KLT to this new feature vector.

The difference between the KLT and MDEE is the information that is thrown away in the second step, where only the dominant eigenfeatures in each group are kept. The discarded information is contained in three groups of covariance matrices. They are the covariance matrices of the removed small eigenfeatures within each group, the cross-covarianace matrices between the removed small eigenfeatures of each group, and the cross-covariance matrices between the small eigenfeatures in one group and the dominant eigenfeatures in another group. Because of the energy packing property of KLT, the information in the first two types of matrices should be negligibly small.

We can also argue that the information in the third type of matrices cannot be large either. If two feature groups are fairly uncorrelated with each other, then any cross-covariance matrices between the two groups will be very small. On the other hand, if the two groups are strongly correlated with each other, the dominant eigenfeatures of the two group will be very similar. Therefore the matrix of cross-covariance between the dominant features in one group and the minor features in another group will be similar to the matrix of cross-covariance between the dominant features and the minor features within the same group, which is zero due to the decorrelation property of the first step KLT transform. Since all three types of matrix are small, we conclude that the information that is discarded in the second step of MDEE is insignificant. The experiments also confirm that the MDEE method is indeed a close approximation of the standard KLT.

Significant reduction of computational time over the standard KLT can be achieved by the MDEE. For example, if a feature vector of length $n = 1000$ is broken into 10 vector groups of length 100, and 10% of the eigenfeatures in each group are saved for the second-level eigenvalue computation, the computational complexity for the MDEE is $11(n/10)^3$, which is nearly two orders of magnitude faster than the KLT's complexity of n^3. Furthermore, the algorithm offers a structure for parallel computation. If all individual group KLTs are computed in parallel, a near three-order-of-magnitude speed increase can be achieved for this example.

However, as optimal representation features, the KLT selected features may not be the best for classification. Additional feature class separability measures are needed to rank the discriminatory ability of KLT decorrelated features. We use the Bhattacharyya distance in this study, because it has a direct relation with the error bound of the Gaussian classifier and has a simple form for features with normal distributions. As shown in [10], for a two-class problem we have

$$\varepsilon_{(c_1,c_2)} \leq [P(c_1)P(c_2)]^{1/2} \exp[-\beta_{d(c_1,c_2)}], \tag{18}$$

where $P(c_i)$ is the prior probability of class c_i, ε is the probability of error for a Gaussian classifier, and β_d is the Bhattacharyya distance. Because its inverse gives the upper bound on the probability of error, β_d can be an effective measure of class separability. For a normal distribution, β_d has the analytical form

$$\beta_{d(c_1,c_2)} = \frac{1}{8}(\mu_1 - \mu_2)^T \left(\frac{W_1 + W_2}{2}\right)^{-1} (\mu_1 - \mu_2) + \frac{1}{2} \ln \frac{\left|\frac{1}{2}(W_1 + W_2)\right|}{|W_1|^{1/2}|W_2|^{1/2}}. \tag{19}$$

where μ_1, μ_2 and W_1, W_2 are the mean vectors and covariance matrices of the two class distributions. The many possible combinations of multiple features and the possibility of covariance matrix singularity make it impractical to compute the Bhattacharyya distance for multiple features at the same time. The one-at-a-time method is adopted instead. The formula is the same as Eq. (19), only with the covariance matrix W replaced by the variance and the mean vector μ replaced by the class mean. Apparently, a large difference in mean values between two clusters results in a large value for the first term of Eq. (19). When the two clusters differentiate in variances, the large value of the second term of Eq. (19) gives a large output of β_d. In the situation that both means and variances are different for the two classes, β_d is at its largest. In all three cases, a Gaussian classifier is expected to give good performance.

For multiclass problems, the overall probability of error can be bounded by

$$\varepsilon \leq \sum_{i>j}^{K} \sum_{j=1}^{K} \varepsilon_{(c_i,c_j)}, \tag{20}$$

where ε and $\varepsilon_{(c_i,c_j)}$ $(i, j = 1, 2, \ldots, K)$ are the probability of overall error and the pairwise error between class i and j, respectively. From Eqs. (18) and (20) we select features according to the minimum total upper error bound. Because the test data size is the same for all classes in the experiment, the prior probabilities $P(c_i)$ are equal for all classes. Thus, we select features with small values of

$$S_b = \sum_{i>j}^{K} \sum_{j=1}^{K} \exp[-\beta_{d(c_i,c_j)}]. \tag{21}$$

Throughout the experiment, we select the first 30 features with largest eigenvalues, rank these MDEE-decorrelated features by their S_b values, and use the first n features with the smallest S_b for classification. We run the feature length n from 1 to 30 to select the one that gives the best performance as the final feature vector length. This is apparently not an optimal searching approach, since a combination of the first n best individual features may not be the best length n feature vector. However, the experimental results suggest it to be a close approximation. Since all features are first decorrelated by the MDEE transform, as we increase the feature length each additional feature brings in new uncorrelated information and noise. When their S_b values increase to a certain point, the new features start to bring in more noise than information, suggesting that a suboptimal feature length is reached. The experiments show that most best feature lengths are from 10 to 20.

In the experiment, we test our algorithms on the following group of wavelet packet features: (1) Level 1: VAR, ENT, MNT$_3$, MNT$_4$, ALL, (2) Level 2: VAR, ENT, MNT$_3$, MNT$_4$, ALL, (3) Level 3: VAR, ENT, MNT$_3$, MNT$_4$, ALL, (4) Level 4: VAR, ALL, (5) Level 1&2: VAR, ALL, (6) Level 1&2&3: VAR, ALL, (7) Level 1&2&3&4: VAR, (8) Standard Wavelet: VAR, ALL, where "ALL" represents the combination of all the four types of features. Our goals are to test the discriminatory power of each feature type in each individual level, the effects of overcomplete representation, and the classification power of the standard wavelet transform.

Another important experiment compares the texture classification performance of different wavelet filter designs and filter lengths. Most previous texture analysis studies have only tested one particular type of wavelet filter design without any specific justification [3, 15]. In this paper, we demonstrate experimentally that the wavelet filter types and lengths do not significantly affect texture classification performance.

2.3. Fourier Transform Features

If we consider the Fourier transform as the highest possible level of multichannel decomposition, we can treat each Fourier frequency component as a frequency channel.

Accordingly, we can treat the Fourier transform magnitude as an energy feature. Instead of painstakingly designing various multichannel filters, we can take the maximum number of filter channels that can be obtained and then let the MDEE transform and the Bhattacharyya distance measure determine which channels to keep and how much of the energy in those channels to keep. The resulting coefficients, which represent the magnitude of each frequency channel's contribution, form a designed filter. We call this approach the dominant spectrum method (DSM). Only half the spectrum matrix is used because of the symmetric property of the Fourier transform of real functions. Since most texture images are generally stationary processes, which decompose canonically into a linear combination of sine and cosine waves in the same way that nonstationary signals decompose into linear combinations of wavelets, we expect DSM features to perform at least as well as the wavelet features.

Our Fourier transform DSM features should not be confused with the traditional power spectrum method (PSM). Early studies of PSM by Bajcsy [1] and Weszka *et al.* [31] concentrate on features computed by the summed spectral energy within circular or wedge-shaped frequency regions. In smooth images, for example, features of the form

$$\mathrm{PSM}_r = \sum_{r_l^2 \le u^2+v^2 < r_2^2} P(u, v) \tag{22}$$

have high values for small r because the smooth images have more energy at lower frequencies, where $P(u, v)$ is the power spectrum. For a rough texture, the high-frequency energy dominates, resulting in high PSM_r for large r. For the same reason, features of the form

$$\mathrm{PSM}_\theta = \sum_{\theta_1 \le \mathrm{atan}(v/u) < \theta_2} P(u, v) \tag{23}$$

give a good measure of directional information. In general, the PSM has been shown to be much less efficient than most other texture analysis methods [5, 31]. Although Jernigan *et al.* [14, 17] propose to use entropy, peak, and shape measures to extract more texture features from the power spectrum, the performance improvement is limited, and the method is not widely accepted as an efficient texture analysis algorithm. Liu and Jernigan [17] gave an extensive summary of features that can be extracted by the PSM. In this paper, we compare 20 features defined in [17].

Criticisms of the PSM have focused on the use of the Fourier transform rather than on the way that texture features are computed from the power spectrum [31]. We believe that the fundamental problem with the PSM features is that the feature extraction functions are, for the most part, *ad hoc*. The features are based on intuitive reasoning through human observation of the power spectrum shape. Instead of trying to develop more *ad hoc* features, we use the MDEE transform and Bhattacharyya distance as feature extraction algorithms to compute texture features from the Fourier transform spectrum matrix. All information in the spectrum is preserved and extracted in an optimal way.

Another problem addressed in this paper is the so-called phase dilemma. In most previous research pertaining to PSM, researchers have tried to use the rich information contained in the Fourier transform phase [8, 9, 15, 30]. This is mainly attributed to successes in the study of image reconstruction from partial Fourier transform information, where the Fourier phase has been shown to contain more information than the Fourier magnitude [6, 11, 21, 22]. So far, however, all results in texture analysis research seem to show that the textural content of the phase information is low.

In fact, the answer to this contradiction is embedded in the basic property of the Fourier transform. The Fourier transform phase carries vital information representing the relative position of the harmonic basis functions essential for the image reconstruction. In most image reconstruction work, the images studied are all natural images with such large, smooth areas as sky or black background. The general shapes of their Fourier transform magnitudes are quite similar, with most energy concentrated in the low-frequency area. When only the phase information is used for reconstruction, usually an average of the Fourier magnitudes of many other irrelevant images is used as the initial magnitude. Thus, except for the small differences in the magnitude of some frequency components, most overall structural information, i.e., the positions of all basis functions, are still there with small changes in magnitude. So the images can be mostly reconstructed with only gray scale changes in certain places.

For texture classification, the situation is completely different. Phase information is a special property of each individual texture sample, as important as that of the natural images used in reconstruction studies. However, since individual texture samples are delineated by an arbitrary grid definition within any class of image, statistically, the phase signals for all texture samples are essentially random regardless of texture classes and thus offer no discrimination power for texture classification. In fact, the phase signal is exactly the kind of noise signal we try to remove.

Although the absolute phase values are noise signals, the phase differences between different frequencies reflect the relative positions of the harmonic functions and thus may offer useful texture information. However, another important but overlooked property of the phase signal prevents the extraction of this information. No matter how small the energy of a frequency component is, its phase value can be anything in a period of 2π. Even though this phase value may be controlled largely by random noise, it still has a value comparable with the phase value of the frequency component having the largest energy. This equal value property essentially renders the phase information useless for natural texture classification, because it makes the extraction of relative phase information impossible. Only for very structured synthetic textures may phase information from Gabor filters be extracted [8].

To confirm this analysis experimentally, we extract texture features by applying the MDEE transform directly on the original texture images. This is equivalent to using both the Fourier magnitude and phase information while, at the same time, avoiding the problem of the equal phase value property.

3. CLASSIFICATION EXPERIMENTS

3.1. Classification Algorithm

The classification algorithm used in this study is the Gaussian classifier. There are two reasons for this choice. First, it agrees with the above error bound defined by the Bhattacharyya distance. Second, with our focus on feature extraction we choose the simplest classification algorithm available. Let the class mean and covariance matrix of the feature vector x for each class i be μ_i and W_i, respectively. The distance measure is defined as [29]

$$D_i = (x - \mu_i)^T W_i^{-1}(x - \mu_i) + \ln|W_i|, \tag{24}$$

where the first term on the right of the equation is actually the Mahalanobis distance. The decision rule is

$$x \in C_L \quad \text{when } D_L = \min\{D_i\}. \tag{25}$$

3.2. Data Description

In addition to the sonar images, we use another set of standard optical texture data, which includes 16 types of natural optical images obtained from the MIT Media Lab Vistex texture data base. They are shown in Fig. 5. The original 512 × 512 color images are converted to the same size gray-scale images. Adaptive histogram equalization is applied so that all images have similar flattened histograms. We conduct most of the testing on the Vistex textures because of their large number of classes. Since our main concern for sonar image is the noise, we test the sensitivity of the algorithms to noise by adding several levels of white Gaussian noise to the optical data. To save computational time, the first round of extensive tests is conducted on the first eight image classes, with three levels of noise added to the images. The signal-to-noise ratios (SNR) are 15 dB, 5 dB, and 1 dB, respectively.

Then relatively more efficient methods selected from the initial comparison are applied and further compared on all 16 classes of Vistex images and on the sidescan sonar images.

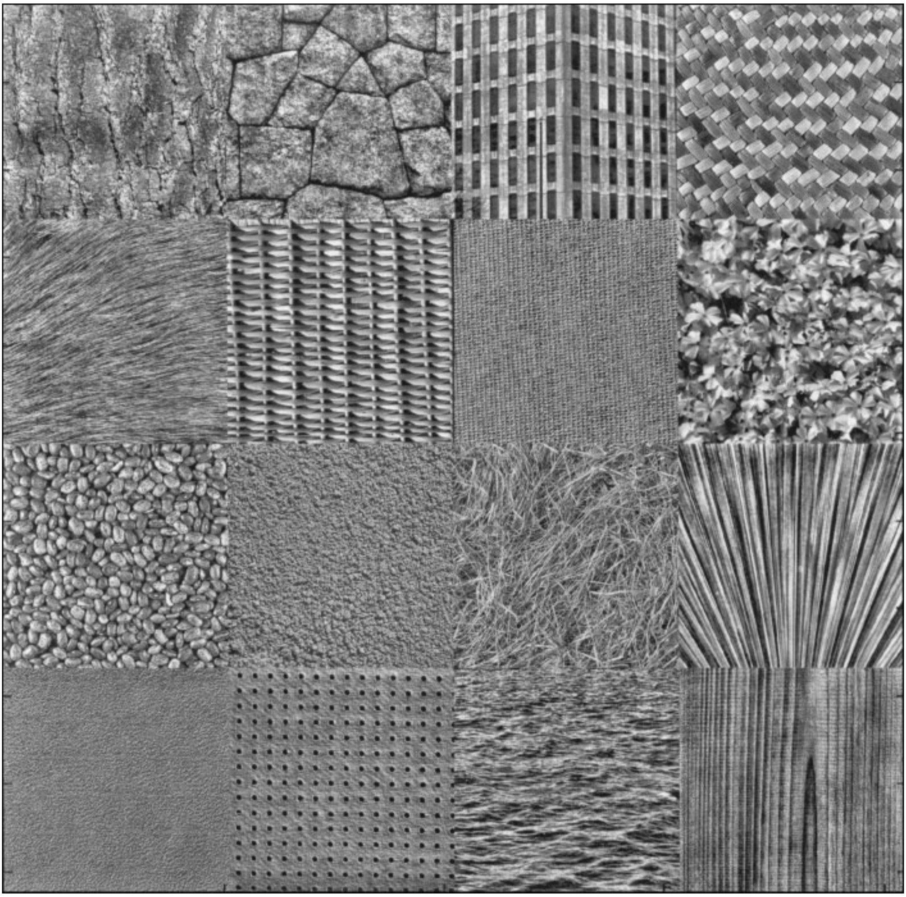

FIG. 5. Sixteen Vistex textures. The eight images in the top two rows are used in the main experiment.

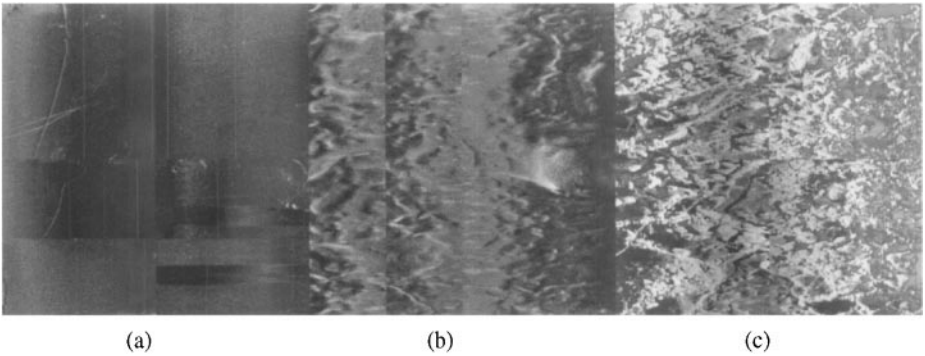

FIG. 6. Three types of sidescan sonar image samples of an Arctic under-ice canopy: (a) first-year ice, (b) multiyear underformed ice, and (c) multiyear deformed ice.

The three classes of sidescan sonar images of an Arctic under-ice canopy are shown in Fig. 6. These are first-year young ice, multiyear undeformed ice, and multiyear deformed ice. The three classes of seafloor sonar images are shown in Fig. 7. For all data sets except the seafloor sonar data, each image class is divided into 225 half-overlapping samples of dimension 64×64, of which 60 samples are used for training. Therefore, the total data sample number is 1800 for the first eight Vistex images, 3600 for all the sixteen Vistex images, and 675 for the sidescan sonar data set, with 480, 960, and 180 samples for training, respectively. For the seafloor data, since the textures are fairly distinctive, to increase the difficulty, we divide each image class into 256 nonoverlapping samples of dimension 32×32, of which 64 samples are used for training.

3.3. Comparison of Wavelet Features with Fourier Transform Features

Table 1 shows the complete test results from the eight class of Vistex images. It is difficult to draw conclusions directly from the large table, so we use a couple of figures to illustrate. Figure 8 shows a comparison of the four types of features and their combinations on the first three decomposition levels. The MNT_3 feature is the worst for all levels and for all data sets and is apparently not a useful measure. Entropy also gives less satisfactory results

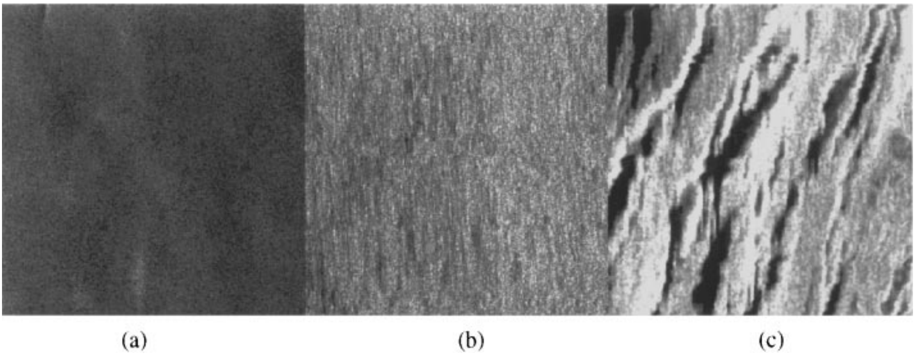

FIG. 7. Three types of seafloor sonar image samples: (a) flat sediment pond; (b) constant-slope ridge flank; and (c) axial-valley segment.

TABLE 1
Complete Test Results on the Eight Vistex Texture Images

			Original images				SNR 15 dB				SNR 5 dB				SNR 1 dB			
			Training	Testing	All data	Feature num.	Training	Testing	All data	Feature num.	Training	Testing	All data	Feature num.	Training	Testing	All data	Feature num.
1	Level 1	VAR	95.6	95.5	95.5	4	95.2	94.8	94.9	4	89.0	89.2	89.2	4	82.3	81.3	81.6	4
2		ENT	87.7	86.7	87.0	4	86.7	84.3	84.9	4	64.4	58.0	59.7	4	49.8	42.7	44.6	4
3		MNT_3	62.9	58.0	59.3	4	61.5	58.1	59.0	4	49.4	41.3	43.4	4	39.4	32.7	34.4	4
4		MNT_4	89.2	91.1	90.6	4	92.3	91.8	91.9	4	89.2	87.5	87.9	4	81.3	80.7	80.8	4
5		ALL	98.8	96.7	97.2	8	98.5	96.7	97.2	9	97.3	92.5	93.8	11	94.8	82.5	85.8	13
6	Level 2	VAR	96.5	96.9	96.8	6	96.3	95.8	95.9	5	96.7	93.0	94.0	9	97.9	89.8	92.0	12
7		ENT	94.2	90.6	91.6	8	97.7	86.5	89.5	14	97.1	66.7	74.8	20	94.2	48.9	60.9	20
8		MNT_3	80.4	62.3	67.2	9	85.2	66.2	71.3	11	96.5	69.8	76.9	18	95.8	56.5	67.0	20
9		MNT_4	94.2	90.9	91.8	6	93.1	91.6	92.0	6	95.0	87.7	89.7	9	95.6	85.0	87.8	12
10		ALL	98.8	97.8	98.1	12	99.4	97.7	98.1	12	95.8	90.5	91.9	9	97.5	84.1	87.7	16
11	Level 3	VAR	97.7	98.1	98.0	6	97.5	98.0	97.9	7	95.8	95.7	95.7	6	96.7	90.3	92.0	13
12		ENT	97.7	79.2	84.2	17	89.8	75.2	79.1	11	96.9	54.8	66.1	22	97.5	40.5	55.7	26
13		MNT_3	82.7	60.5	66.4	10	85.8	61.6	68.1	12	96.0	53.8	65.1	22	99.6	45.2	59.7	30
14		MNT_4	94.8	92.7	93.2	7	94.2	92.0	92.6	8	95.0	91.1	92.2	9	97.9	83.9	87.7	18
15		ALL	98.5	97.8	98.0	16	97.7	97.3	97.4	6	97.5	93.2	94.3	10	96.5	90.9	92.4	13
16	Level 4	VAR	97.1	97.5	97.4	6	97.5	96.9	97.1	6	96.7	95.9	96.1	6	98.5	94.5	95.6	12
17		ALL	97.7	97.9	97.8	8	97.9	96.4	96.8	9	99.2	93.9	95.3	18	95.4	93.0	93.6	6
18	Level 12	VAR	96.5	96.9	96.8	6	96.3	95.8	95.9	5	96.7	93.0	94.0	9	97.9	89.8	92.0	12
19		ALL	99.4	98.3	98.6	10	99.2	98.2	98.4	10	99.2	93.3	94.8	17	93.3	85.8	87.8	10
20	Level 123	VAR	98.5	98.6	98.6	7	98.3	98.6	98.6	7	96.9	95.3	95.7	9	97.9	90.5	92.4	13
21		ALL	99.8	98.6	98.9	20	99.2	98.2	98.4	11	98.8	94.5	95.7	13	97.7	91.7	93.3	14
22	Level 1234	VAR	99.0	97.2	97.7	14	98.8	96.7	97.3	12	97.9	96.0	96.5	12	98.1	93.7	94.9	11
23	Wavelet	VAR	98.1	97.0	97.3	10	97.9	96.7	97.1	10	97.7	95.0	95.7	11	97.9	94.2	95.2	12
24		ALL	100	96.6	97.5	18	100	96.4	97.4	18	97.9	93.8	94.9	14	98.1	90.2	92.3	19
25	FFT Mag. (DSM)		99.8	98.4	98.8	26	99.4	97.9	98.3	15	99.0	96.4	97.1	10	99.8	95.8	96.8	17

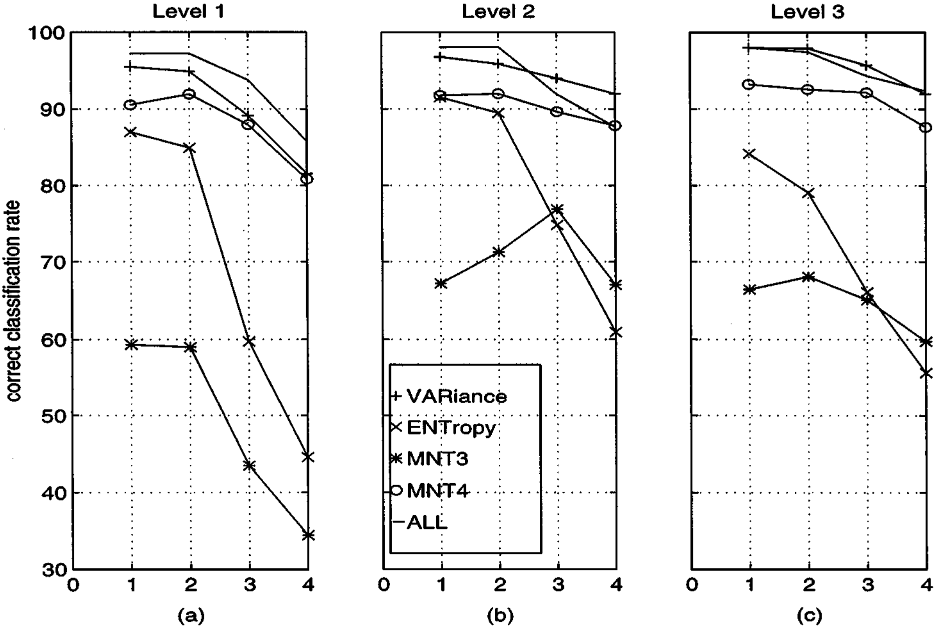

FIG. 8. Comparison of the four types of features in the first three individual decomposition levels. The index of the horizontal axis represents signal-to-noise ratio (SNR) level: (1) original image, (2) SNR 15 dB, (3) SNR 5 dB, (4) SNR 1 dB.

than the variance feature, and the classification accuracy drops sharply for noisy data. The MNT$_4$ feature seems to give better results than the above two features but is still less successful than the variance feature. The performance differences between the MNT$_4$ and the variance are consistent over all data sets and all decomposition levels, because they are very closely correlated features. The observation that variance features perform better than other features is consistent with Laws' [16] experiment with features extracted from empirical frequency channels. By combining all features together, we get improved results for the lower decomposition level. Since the feature length is much smaller on these levels, an added dimension helps more than in the higher level decomposition case.

Now consider in detail the variance measure results shown in Fig. 9. For the individual levels of Fig. 9a, the general trend is that the higher the decomposition level the better the result. This is predictable from Eq. (14), which shows that the lower level variance features are simply the average of their higher level children nodes. A KLT transform will do better than such a simple average operation in terms of extracting maximum information. To confirm this point, compare Figs. 9a and 9b, which show that the following pairs of results are almost identical: level 1&2 vs level 2, level 1&2&3 vs level 3, level 1&2&3&4 vs level 4. This means that lower level features are only a subset of higher level decomposition features and that better discriminatory ability is not added by redundancy. Instead, more discriminatory information is extracted by applying KLT to higher levels of finer channel decomposition, so the channel nodes are combined in an optimal way instead of by simple averaging.

Continuing this thread of analysis, we should expect that the Fourier transform provides even more information with more channels. Figure 9c compares the performances of three levels of wavelet packet decomposition, the standard wavelet transform, and the Fourier

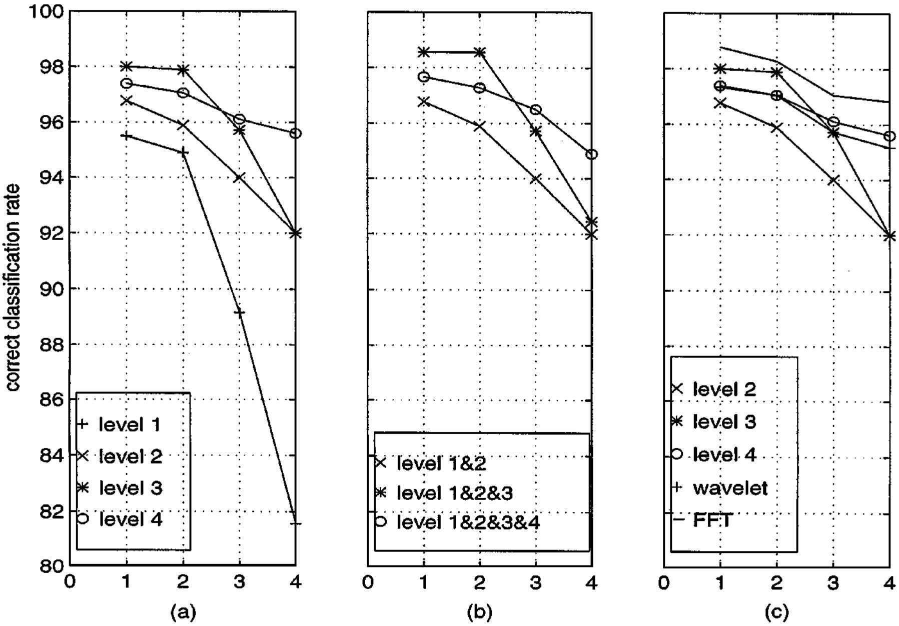

FIG. 9. Comparison of variance features for individual decomposition levels, overcomplete levels, standard wavelet, and Fourier transform. The index of the horizontal axis represents the same SNR as in Fig. 8.

transform. The Fourier transform indeed gives consistently better performance than all other feature groups on all levels of noisy data sets. This result should not be surprising, since the wavelet transform is optimal for nonstationary transient signal analysis, whereas the Fourier transform is optimal for stationary signal analysis. Most texture images are stationary periodic signals. Using the MDEE on the Fourier spectrum is in effect a filter design method according to signal statistical distribution, which should be better than the filter channels prefixed by the wavelet transform.

Next, notice in Fig. 9c that the Fourier transform and other higher levels of wavelet packet decomposition are very insensitive to noise. Noise insensitivity is a particular strength of multichannel image analysis. Noise usually has a flat spectrum and, when it is divided into more channels, the noise energy usually stay the same. Yet the energy of signals tends to concentrate in a few channels. Therefore, even when the total energy of the signal and noise are almost the same, as in the case of the testing data of SNR 1 dB, the signal-to-noise ratio is much higher in channels containing the most signal energy. The MDEE tends to condense the coherent signal channels with high SNR into a compact representation of the data, with the incoherent noisy channels neglected. This noise insensitivity property is the reason that we study the multichannel algorithm for sonar image classification, because sonar signals tend to be noisier than optical images.

3.4. Comparison of Different Wavelet Filter Types and Lengths

The wavelet filter type used in the above experiments is the Daubechies minimum-support least asymmetric wavelet of filter length 4 [7]. In this section we compare this particular type and length of wavelet filter with others. The Daubechies minimum-support least asymmetric wavelet and the Coiflets [7] of various lengths are used for the comparison. Table 2 lists the

TABLE 2
Comparison of Different Wavelet Filter Types and Lengths for Eight Vistex Images with SNR 5 dB

		Daubechies minimum-support least asymmetric			Coiflets		
		4	8	16	6	12	18
1	Level 2 VAR	94.0	94.7	95.4	95.7	94.7	95.8
2	Level 3 VAR	95.7	96.7	96.3	96.3	95.7	95.4
3	Level 4 VAR	96.1	95.4	95.3	94.9	94.7	95.9
4	All levels VAR	96.5	95.4	95.6	95.6	95.4	96.3

classification results on the eight Vistex images with noise level of SNR 5 dB. We see that almost all differences in classification accuracy among the different filter types and lengths are within 1%. No particular trend can be observed as the filter length increases, and the overall performance of the two types of filters are almost the same. These results seem to indicate that the conclusions drawn in the previous section are very likely to be similar for other types of wavelets.

3.5. Comparison of the KLT and MDEE

Except for some small feature vectors in the above experiments, we use the MDEE transform in place of the KLT. We did not list the specific grouping parameters in the table

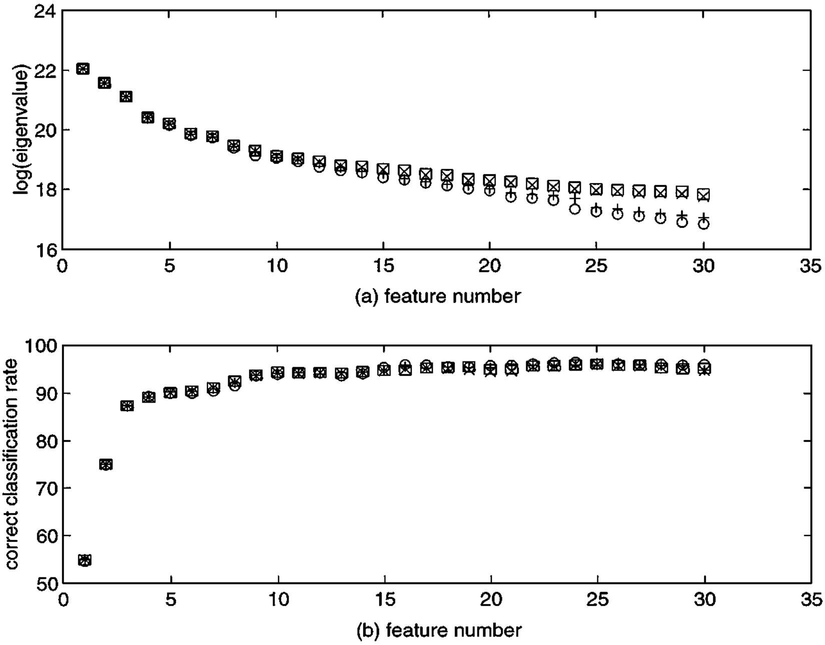

FIG. 10. Comparison of the KLT and MDEE transforms: (a) plot of the top 30 largest eigenvalues, (b) plot of the correct classification rate against number of features used. In both plots, the square symbol is for the KLT and the other three are for MDEE: × for $n = 20$; + for $n = 10$; ○ for $n = 5$.

TABLE 3
Comparison of the DSM, PSM, and Texture Features Extracted Directly from Spatial Domain Images, for Eight Vistex Images with SNR 5 dB

		Training data	Testing data	All data	Feature number
1	Original image	97.1	73.0	79.4	23
2	PSM	91.0	85.7	87.1	7
3	DSM	99.0	96.4	97.1	10

of results, because the numerical differences between the standard KLT and the MDEE transform are negligibly small, as shown in the following experiment.

For this experiment, the data set comprises the original eight images from the previous experiment. The feature vector is formed by 420 frequency components in the center high-frequency region of the Fourier transform magnitude matrix. We use a small feature vector for this experiment so the brute force KLT can be computed within a reasonable time with reasonable memory requirement. For the MDEE transform, the feature vector is broken into seven feature vectors of length 60 each. Then, the first $n = 20$, 10, and 5 dominant features in each group are selected to form the new feature vector of length $7 \cdot n$, from which the final dominant eigenvalues and eigenvectors are computed. Figure 10a shows the results of the top 30 eigenvalues of the standard KLT and the MDEE transforms with the three values of n. We see that when 20 features are kept after the first step eigenvalue computation, the final MDEE eigenvalues are almost the same as the standard KLT. When only 10 or 5 features are kept, the first 15 eigenvalues are still similar to KLT; the remaining eigenvalues start to lose a very small amount of energy. However this does not affect the final classification results at all. Figure 10b shows the classification accuracies using the KLT and the MDEE plotted against the number of features used. All four groups of results overlap with each other almost completely, with the maximum classification accuracy difference being less than 0.5%.

3.6. Comparison of the DSM Features with the PSM Features

A comparison of the DSM features with the PSM [17] features is conducted on the eight Vistex images with noise level of SNR 5 dB. Results are given in Table 3, where the DSM features show nearly 10% better performance than the PSM features. The performance

TABLE 4
Classification Results of the 16 Vistex Images

		Original 16 Vistex images				16 Vistex images of SNR 5 dB			
		Training	Testing	All data	Feature num.	Training	Testing	All data	Feature num.
1	Level 3 VAR	97.6	97.3	97.4	8	97.2	94.0	94.8	11
2	Level 4 VAR	97.8	97.2	97.3	10	99.1	92.8	94.5	19
3	All levels VAR	98.3	97.3	97.6	10	98.9	93.1	94.6	19
4	PSM	83.5	81.4	81.9	5	72.9	71.6	71.9	4
5	DSM	99.1	96.8	97.4	13	99.2	95.8	96.7	16

TABLE 5
Classification Results of the Sidescan Sonar Images

		Seafloor sidescan sonar images				Arctic sidescan sonar images			
		Training	Testing	All data	Feature num.	Training	Testing	All data	Feature num.
1	Level 3 VAR	99.4	99.4	99.4	6	93.3	83.2	85.9	12
2	Level 4 VAR	100	98.8	99.1	13	96.1	81.4	85.3	20
3	All levels VAR	100	98.9	99.2	14	98.3	85.3	88.7	23
4	PSM	98.4	93.4	94.6	16	93.9	89.1	90.4	8
5	DSM	100	99.8	99.9	22	98.9	92.5	94.2	9

discrepancy is further widened when tested on a larger data set, as described in the next section. This demonstrates that the optimal feature selection approach using MDEE and Bhattacharyya distance is better than an *ad hoc* feature extraction method. The first row in the table gives the results of using the original image directly as the feature vector. The poor performance confirms that the phase information is only a noise signal for texture classification.

3.7. Experiments on the Sonar Imagery

Finally, we apply the algorithms on the classification of a larger data set of 16 Vistex images and the sidescan sonar images. Only the feature groups that perform best in the above experiment are used. Table 4 shows the classification results on the Vistex images, which are consistent with the above results. An interesting observation is that the DSM method is very insensitive to noise. With SNR 5 dB noise added to the 16 images, the classification accuracy drops less than 1%. This is exactly the kind of property that is important for the noisy sonar image classification.

For the sidescan sonar images, although the image class number is smaller, each class of images is noisy and nonuniform. From results in Table 5, it is interesting to see that the wavelet features perform better than the PSM features on the seafloor data, while they give worse results on the Arctic ice canopy data. This shows that both methods rely on the combination of frequency channel features through ad hoc linear or nonlinear operations and thus produce unstable results for different data. We can see from both datasets that the DSM texture features are very effective in classification of sidescan sonar images. This again demonstrates that given proper feature extraction approach, more effective texture features can be extracted from the Fourier transformation than from the more complex wavelet transformation.

4. CONCLUSIONS

Based on the above experiments, the following conclusions are drawn:

(1) Variance (energy) measures are much better than entropy and higher order moments. For variance features, overcomplete representation does not add more information than individual level features. Higher levels of decomposition perform better than lower levels. This leads to the conclusion that the Fourier transform magnitude features are better than the more complicated wavelet packet features.

(2) The MDEE is a close approximation of the KLT. The MDEE plus the Bhattacharyya distance measure is shown to be very effective in extracting texture features from both wavelet packet transforms and the Fourier transform.

(3) The Fourier phase information is a noise signal for texture classification. However, the superior performance of DSM over the conventional PSM shows that the Fourier transform magnitudes contain enough texture information for classification, if the right feature extraction algorithm is used.

(4) Multichannel features are insensitive to noise. Features from higher levels are less sensitive than the lower level features. Dominant Fourier transform features are the best in terms of noise insensitivity and are clearly very effective features for both optical and sidescan sonar image classification.

ACKNOWLEDGMENTS

The authors thank Professor W. Eric L. Grimson of MIT AI Lab for his comments and encouragement. The authors are also grateful to the MIT Media Lab for providing the Vistex texture images. This work was supported in part by the Chinese University of Hong Kong through a RGC Earmarked Grant and by the Office of Naval Research of the U.S.

REFERENCES

1. R. Bajcsy, Computer description of textured surfaces, in *Proc. 3rd Int. Joint Conf. Artificial Intelligence, 1973*, pp. 572–579.
2. J. Bigun and J. M. H. du Buf, N-folded symmetries by complex moments in Gabor space and their application to unsupervised texture segmentation, *IEEE Trans. Pattern Anal. Mach. Intell.* **16**(1), 1994, 80–87.
3. T. Chang and C.-C. J. Kuo, Texture analysis and classification with tree-structured wavelet transform, *IEEE Trans. Image Processing* **2**, 1993, 429–441.
4. R. R. Coifman and M. V. Wickerhauser, Entropy-based algorithms for best basis selection, *IEEE Trans. Inform. Theory* **38**, 1992, 713–719.
5. R. W. Conners and C. A. Harlow, A theoretical comparison of texture algorithms, *IEEE Trans. Pattern Anal. Mach. Intell.* **2**(3), 1980, 204–222.
6. S. R. Curtis, A. V. Oppenheim, and J. S. Lim, Signal reconstruction from Fourier transform sign information, *IEEE Trans. Acoust. Speech Signal Process.* **33**, 1985, 643–657.
7. I. Daubechies, *Ten Lectures On Wavelets*, SIAM, Philadelphia, 1992.
8. J. M. H. du Buf and P. Heitkaemper, Texture features based on Gabor phase, *Signal Process.* **23**, 1991, 227–244.
9. J. O. Eklundh, On the use of Fourier phase features for texture discrimination, *Comput. Graphics Image Process.* **9**, 1979, 199–201.
10. K. Fukunaga, *Introduction to Statistical Pattern Recognition*, Academic Press, New York, 1972.
11. M. H. Hayes, J. S. Lim, and A. V. Oppenheim, “Signal reconstruction from phase or magnitude,” *IEEE Trans. Acoust. Speech Signal Process.* **28**, 1980, 672–680.
12. M. B. Henke-Reed and S. N. C. Cheng, Cloth texture classification using the wavelet transform, *J. Imaging Sci. Technol.* **37**, 1993, 610.
13. A. K. Jain and F. Farrokhnia, Unsupervised texture segmentation using Gabor filters, *Pattern Recognition* **24**, 1991, 1167–1186.
14. M. E. Jernigan and F. D’Astous, Entropy-based texture analysis in the spatial frequency domain, *IEEE Trans. Pattern Anal. Mach. Intell.* **6**, 1984, 237–243.
15. A. Laine and J. Fan, Texture classification by wavelet packet signatures, *IEEE Trans. Pattern Anal. Mach. Intell.* **15**, 1993, 1186–1191.

16. K. I. Laws, *Textured Image Segmentation*, Ph.D. dissertation, Image Processing Inst., Univ. of Southern California, 1980.

17. S. Liu and M. E. Jernigan, Texture analysis and discrimination in additive noise, *Comput. Vision Graphics Image Process.* **49**, 1990, 52–67.

18. S. G. Mallat, A theory for multiresolution signal decomposition: The wavelet representation, *IEEE Trans. Pattern Anal. Mach. Intell.* **11**, 1989, 674–693.

19. S. G. Mallat, Multifrequency channel decompositions of images and wavelet models, *IEEE Trans. Acoust. Speech Signal Process.* **37**, 1989, 2091–2110.

20. Y. Meyer, *Wavelets: Algorithms and Applications*, translated and revised by R. D. Ryan, SIAM, Philadelphia, 1993.

21. A. V. Oppenheim, M. H. Hayes, and J. S. Lim, Iterative procedures for signal reconstruction from Fourier transform phase, *Opt. Eng.* **21**, 1982, 122–127.

22. A. V. Oppenheim, J. S. Lim, and S. R. Curtis, Signal synthesis and reconstruction from partial Fourier-domain information, *J. Opt. Soc. Am.* **73**(11), 1983, 1413–1420.

23. N. G. Pace and C. M. Dyer, Machine classification of sedimentary sea bottoms, *IEEE Trans. Geosci. Remote Sensing* **GE-17**, 1979, 52–56.

24. N. G. Pace and H. Gao, Swathe seabed classification, *IEEE J. Oceanic Eng.* **13**(2), 1988, 83–90.

25. T. B. Reed, IV, and D. M. Hussong, Digital image processing techniques for enhancement and classification of SeaMARC II sidescan sonar imagery, *J. Geophys. Res.* **94**, 1989, 7469–7490.

26. Z. Reut, N. G. Pace, and M. J. P. Heaton, Computer classification of sea beds by sonar, *Nature* **314**, 1985, 426–428.

27. W. K. Stewart, M. Jiang, and M. Marra, A neural network approach to classification of sidescan sonar imagery from a midocean ridge area, *IEEE J. Oceanic Eng.* **19**(2), 1994, 214–224.

28. X. Tang, *Transform Image Classification*, Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge, MA, 1996.

29. C. W. Therrien, *Decision Estimation and Classification, An Introduction to Pattern Recognition and Related Topics*, Wiley, New York, 1989.

30. A. E. J. Walford and M. E. Jernigan, The phase spectrum and its contribution to human texture perception, *IEEE 1985 Proc. of the Internat. Conf. on Systems Man and Cybernetics*, pp. 701–704.

31. J. S. Weszka, C. R. Dyer, and A. Rosenfeld, A comparative study of texture measures for terrain classification, *IEEE Trans. Systems Man Cybernet.* **SMC-6**(4), 1976, 269–285.