Journal of Mathematical Imaging and Vision, 5, 337–358 (1995)

Local Discriminant Bases and Their Applications*

NAOKI SAITO
Schlumberger-Doll Research, Old Quarry Road, Ridgefield, CT 06877

RONALD R. COIFMAN
Department of Mathematics, Yale University, New Haven, CT 06520

Abstract. We describe an extension to the "best-basis" method to select an orthonormal basis suitable for signal/image classification problems from a large collection of orthonormal bases consisting of wavelet packets or local trigonometric bases. The original best-basis algorithm selects a basis minimizing entropy from such a "library of orthonormal bases" whereas the proposed algorithm selects a basis maximizing a certain discriminant measure (e.g., relative entropy) among classes. Once such a basis is selected, a small number of most significant coordinates (features) are fed into a traditional classifier such as Linear Discriminant Analysis (LDA) or Classification and Regression Tree (CART$^{\mathrm{TM}}$). The performance of these statistical methods is enhanced since the proposed methods reduce the dimensionality of the problem at hand without losing important information for that problem. Here, the basis functions which are well-localized in the time-frequency plane are used as feature extractors. We applied our method to two signal classification problems and an image texture classification problem. These experiments show the superiority of our method over the direct application of these classifiers on the input signals. As a further application, we also describe a method to extract signal component from data consisting of signal and textured background.

Keywords: wavelet packets, local trigonometric transforms, feature extraction, classification, dimensionality reduction

1 Introduction

In analyzing and interpreting signals such as musical recordings, seismic signals, or stock market fluctuations, or images such as mammograms or satellite images, extracting relevant features from them is of vital importance. Often, the important features for signal analysis, such as edges, spikes, transients, or textures, are characterized by local information either in the time (or space) domain or in the frequency (or spatial frequency/wave number) domain or in both:[1] for example, to discriminate seismic signals caused by nuclear explosions from the ones caused by natural earthquakes, the frequency characteristics of the primary waves, which arrive in a short and specific time window, may be a key factor; to distinguish benign and malignant tissues in mammograms, the sharpness of the edges of masses may be of critical importance.

In this paper, we explore how to extract relevant features from signals/images and discard irrelevant information for signal/image classification problems. In particular, we propose a fast algorithm to select an efficient basis (or coordinate system) from a large collection of orthonormal bases (consisting of wavelet packets and local trigonometric bases) to enhance the performance of a few classification schemes. This algorithm reduces the dimensionality of the problems by using these basis functions (which are well-localized in the time-frequency plane) as feature extractors. Since this basis illuminates the differences among classes, it can also be used to extract signal component from data consisting of signal and textured background.

The organization of this paper is as follows. In Section 2, we formulate the problem of feature extraction and classification and briefly review some pattern classification schemes used in our study. Then, in Section 3, we review the "best-basis paradigm" and a dictionary and a library of orthonormal bases which play a critical role for local feature extraction. Section 4 is a core material of this paper: we describe a fast algorithm for constructing a good local basis for classification problems. This is immediately followed by signal classification examples in Section 5 and an image texture classification problem in Section 6. In Section 7, we discuss a method of

*This research was supported by Schlumberger-Doll Research and by ARPA ATR program.

signal/"background" separation as a further application of such a basis.

We note that a concise version of this paper was announced earlier in [1] which also contains an algorithm for constructing a local basis for regression problems, and was presented in the SPIE conference [2]. The other aspects of our proposed method, including its applications to regression problems and examples using real datasets, can be found in [3–5].

2 Problem Formulation and Review of Pattern Classifiers

2.1 Formulation of a Signal Classification Problem

Let us first define appropriate spaces of input signals (or patterns), extracted features, outputs (or responses), and mapping functions among them. Let $\mathcal{X} \subset \mathbb{R}^n$ denote a *signal space* (or a *pattern space*) which is a subset of the standard n-dimensional vector space and which contains all signals (or samples/patterns) under consideration. In this case, the *dimensionality* of the signal space or equivalently the length of each signal is n. Let $\mathcal{Y} = \{1, 2, \ldots, C\}$ be a set of the class or category names to which the input signals belong. We call this space a *response space*. Signal classification can be considered as a mapping function (usually many-to-one) $d\colon \mathcal{X} \to \mathcal{Y}$ between these two spaces. Direct manipulation of signals in the signal space for classification is prohibitive because: 1) the signal space normally has very high dimensionality (e.g., $n \approx 1000$ for a typical exploration seismic record per receiver, and for a typical CT scanner image, $n = 512 \times 512 = 262, 144$), and 2) the existence of noise or undesired components (whether random or not) in signals makes classification difficult. On the other hand, the signal space is overly redundant compared to the response space. Therefore, it is extremely important to reduce the dimensionality of the problem, i.e., extract only relevant features for the problem at hand and discard all irrelevant information. If we succeed in doing this, we can greatly improve classification performance both in its accuracy and efficiency. For this purpose, we set up a *feature space* $\mathcal{F} \subset \mathbb{R}^k$ where $k \leq n$ between the signal space and the response space. A *feature extractor* is defined as a map $f\colon \mathcal{X} \to \mathcal{F}$, and a *classifier* (or *predictor*) as a map $g\colon \mathcal{F} \to \mathcal{Y}$. Let $\mathcal{T} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{N} \subset \mathcal{X} \times \mathcal{Y}$ be a *training* (or *learning*) dataset with N pairs of signals $\boldsymbol{x}_i$ and responses (class names) y_i. This is the dataset to be used to construct a feature extractor f. Let N_c be the number of signals belonging to class c so that we have $N = N_1 + \cdots + N_C$. Also, let us denote a set of class c signals by $\{\boldsymbol{x}_i^{(c)}\}_{i=1}^{N_c} = \{\boldsymbol{x}_i\}_{i \in I_c}$ where $I_c \subset \{1, \ldots, N\}$ is a set of indices for class c signals in the training dataset with $|I_c| = N_c$.

Preferably, the performance of the whole process should be measured by the misclassification rate using a *test* dataset $\mathcal{T}' = \{(y_i', \boldsymbol{x}_i')\}_{i=1}^{N'}$ (which has not been used to construct the feature extractors and classifiers) as $(1/N') \sum_{i=1}^{N'} \delta(y_i' - d(\boldsymbol{x}_i'))$, where $\delta(r \neq 0) = 1$ and $\delta(0) = 0$. If we use the *resubstitution* (or *apparent*) error rates (i.e., the misclassification rates computed on the training dataset), we obviously have overly optimistic figures.

In this paper, we focus on the feature extractors of the form

$$f = \Theta^{(k)} \circ \Psi,$$

where $\Theta^{(k)}\colon \mathcal{X} \to \mathcal{F}$ represents the selection rule (e.g., picking most important k coordinates from n coordinates), and $\Psi \in O(n)$, i.e., an n-dimensional orthogonal matrix. In particular, we consider matrices representing the orthonormal bases in the basis library (consisting of wavelet packets or local trigonometric bases) as candidates for Ψ. As a classifier g, we adopt Linear Discriminant Analysis (LDA) of R.A. Fisher [6] and Classification and Regression Trees (CART$^{\text{TM}}$) [7].

In the following, we briefly review these two classification schemes. We note that other classifiers such as k-nearest neighbor (k-NN) [8], or artificial neural networks (ANN) [9] are all possible to use in our algorithm. The reader interested in comparisons of different classifiers is referred to the excellent review article of Ripley [9]. The useful information on pattern classifiers in general can be found in the books [10–13].

2.2 Linear Discriminant Analysis

Fisher's LDA first tries to do its own feature extraction by a linear map $A^T\colon \mathcal{X} \to \mathcal{F}$ (in this case not necessarily orthogonal matrix). This map A simultaneously minimizes the scatter of sample vectors (signals) within each class and maximizes the scatter of mean vectors of classes around the total mean vector. To be more precise, let $\boldsymbol{m}_c \stackrel{\Delta}{=} (1/N_c) \sum_{i=1}^{N_c} \boldsymbol{x}_i$ be a mean vector of class c signals.[2] Then the total mean vector $\boldsymbol{m}$ can be defined as $\boldsymbol{m} \stackrel{\Delta}{=} \sum_{c=1}^{C} \pi_c \boldsymbol{m}_c$, where π_c is the prior probability of class c (which can be set to N_c/N without the knowledge on the true prior probability). The scatter of samples within each class can be measured by the *within-class covariance matrix* $\Sigma_w \triangleq \sum_{c=1}^{C} \pi_c \Sigma_c$, where Σ_c is the *sample covariance matrix* of class c: $\Sigma_c \triangleq (1/N_c) \sum_{i=1}^{N_c} (\boldsymbol{x}_i^{(c)} - \boldsymbol{m}_c)(\boldsymbol{x}_i^{(c)} - \boldsymbol{m}_c)^T$. The scatter of mean vectors around the total mean can be measured by the *between-class covariance matrix* $\Sigma_b \triangleq \sum_{c=1}^{C} \pi_c (\boldsymbol{m}_c - \boldsymbol{m})(\boldsymbol{m}_c - \boldsymbol{m})^T$. Then, LDA maximizes a class separability index $J(A) \triangleq \mathrm{tr}[(A^T \Sigma_b A)^{-1}(A^T \Sigma_w A)]$, which measures how much these classes are separated in the feature space. This requires solving the so-called generalized (or pencil-type) eigenvalue problem $\Sigma_b A = \Sigma_w A \Lambda$, where Λ is a diagonal matrix containing the eigenvalues. Once the map A is obtained (normally $k = C - 1$), then the feature vector $A^T \boldsymbol{x}_i$ is computed for each i, and finally it is assigned to the class which has the mean vector closest to this feature vector in the Euclidean distance in the feature space. This is equivalent to bisecting the feature space $\mathcal{F}$ by hyperplanes. In this paper we regard LDA as a classifier although, as explained, it also includes its own feature extractor A^T. LDA is the optimal strategy if all classes of signals obey multivariate normal distributions with different mean vectors and an equal covariance matrix [10, 12]. In reality, however, it is hard to assume this condition. Moreover, since it relies on solving the eigensystem, LDA can only extract global features (or squeezes all discriminant information into a few $[C - 1]$ basis vectors) so that the interpretation of the extracted features becomes difficult, it is sensitive to outliers and noise, and it requires $O(n^3)$ calculations.

2.3 Classification and Regression Trees

Another popular classification/regression scheme, CART [7] is a nonparametric method which recursively splits the input signal space *along* the coordinate axes and generates a partition of the input signal space into disjoint blocks so that the process can be conveniently described as a binary tree where nodes represent blocks. Such a tree for classification problems is called a *classification tree* (CT). At each node in a CT, a class label is assigned by the majority vote at that node. Then, candidate splits are evaluated by the "information gain" or the quantity called *deviance* and the most "informative" split is selected. The popular measure as the deviance for the classification is entropy of a node which is defined as $-\sum_{c=1}^{C} p_c \log_2 p_c$, where p_c is the proportion of class c samples over the whole samples at that node. (From now on, we use "log" for the logarithm of base 2, unless mentioned otherwise.) Thus, the best split amounts to maximally reducing the entropy of that node. Once the best split is determined, all the input signals belonging to that node is split into two groups (children nodes). Splitting is continued recursively until nodes become "pure", i.e., they contain only one class of signals, or become "sparse", i.e., they contain only a few signals.[3] Finally, the pruning process to eliminate unimportant branches is usually applied after growing the initial tree to avoid the "overtraining". We refer the reader to [7] for the details of splitting, stopping, and pruning rules. For pruning methods using information-theoretic criteria, see [17, Chapter 7], [18, 19], and [3, 4].

CART requires searching and sorting all the coordinates of training signals for the best splits. Thus, it is computationally expensive for the problem of high dimensionality. This is more emphasized if we want to split the signal space "obliquely" by taking linear combinations of the coordinates to generate a tree.

3 The Best-Basis Paradigm and a Library of Bases

3.1 The Best-Basis Paradigm

The approach to the feature extraction for classification explored in this paper is guided by the so-called *best-basis paradigm* [20, 21], [3]. This paradigm consists of three main steps:

1. select a "best" basis (or coordinate system) for the problem at hand from a *library of bases* (a fixed yet flexible set of bases such as wavelets and their relatives, i.e., wavelet packets and local trigonometric bases),
2. sort the coordinates (features) by "importance" for the problem at hand and discard "unimportant" coordinates, and
3. use the surviving coordinates to solve the problem at hand.

What is "best" and "important" clearly depends on the problem. For signal compression, a basis which provides only a few large components in the coordinate vectors should be used since we can then discard the other components without much signal degradation. Thus, to measure the efficiency of the coordinate system for compression, an information cost such as *entropy* may be appropriate since entropy measures the number of significant coordinates in a vector. For classification, a basis through which we can "view" classes as maximally-separated point clouds in the n-dimensional space is a choice. In this case, the class separability index or "distances" among classes should be used as a measure of the efficiency of the coordinate system, which we consider in detail in Section 4.

One may ask why we use wavelets and their relatives as library members. The main reason is that they, as a library, provide flexible and efficient coordinate systems to capture and represent local features in the time-frequency plane. Moreover, they can be obtained in a computationally efficient manner: to find a good coordinate system for one's problem, it costs $O(n[\log n]^p)$, where $p = 0, 1, 2$ depending on the basis type. This paradigm leads us to a vastly more efficient representation, processing, and analysis of signals, compared with strategies of confining ourselves to a single basis, or of seeking the absolutely best solution without restricting the library.

Throughout this paper, we only consider a library of *orthonormal* bases mainly because of their computational efficiency and simplicity in implementation of numerical algorithms. For a library of "non-orthogonal" bases and its applications, see e.g., [3, Chapter 7], [22].

3.2 A Dictionary of Orthonormal Bases

We now define a dictionary of orthonormal bases more precisely. But, first, let us briefly review the wavelet and wavelet packet transforms. The detailed properties of these transforms and bases can be found in the literature, most notably, in [23–25]. These essentially partition the frequency axis *smoothly* and analyze each segment with a resolution matched to its scale. An input signal is first decomposed into low and high frequency bands by the convolution-subsampling operations with the pair consisting of a "lowpass" filter $\{h_k\}_{k=0}^{L-1}$ and a "highpass" filter $\{g_k\}_{k=0}^{L-1}$ directly on the discrete time domain. Let H and G be the convolution-subsampling operators using these filters. These are called (*perfect reconstruction*) *quadrature mirror filters* (QMFs) if they satisfy the following orthogonality (or perfect reconstruction) conditions:

$$HG^* = GH^* = 0 \quad \text{and} \quad H^*H + G^*G = I,$$

where I is the identity operator. Various design criteria (concerning regularity, symmetry etc.) on the lowpass filter coefficients $\{h_k\}$ can be found in [23]. Once $\{h_k\}$ is fixed, we can have QMFs by setting $g_k = (-1)^k h_{L-1-k}$.

In the *wavelet transform*, this decomposition (also known as expansion or analysis) process is iterated only on the low frequency bands and each time the high frequency coefficients are retained intact. In other words, let $x = \{x_k\}_{k=0}^{n-1} \in \mathbb{R}^n$ with $n = 2^{n_0}$ be a vector to be expanded. Then, the convolution-subsampling operations transform the vector x into two subsequences Hx and Gx of lengths $n/2$. Next, the same operations are applied to the vector of the lower frequency band Hx to obtain H^2x and GHx of lengths $n/4$. If the process is iterated J ($\leq n_0$) times, we have the discrete wavelet coefficients $(Gx, GHx, GH^2x, \ldots, GH^Jx, H^{J+1}x)$ of length n. At the last iteration, both low and high frequency coefficients are kept. As a result, the wavelet transform analyzes the data by partitioning its frequency content dyadically finer and finer toward the low frequency region (i.e., coarser and coarser in the original time domain). The reconstruction (or synthesis) process is also very simple thanks to the perfect reconstruction conditions: starting from the lowest two frequency bands $H^{J+1}x$ and GH^Jx, the adjoint operations are applied and added to obtain $H^Jx = H^*H^{J+1}x + G^*GH^Jx$. This process is iterated to reconstruct the original vector x. The computational complexity of the decomposition and reconstruction process is $O(n)$ in both cases as easily seen.

On the other hand, the *wavelet packet transform* decomposes even the high frequency bands which are kept intact in the wavelet transform. Thus, the wavelet packet transform is more suitable for analyzing oscillatory signals such as acoustic signals or textured images. The first level decomposition generates Hx and Gx just like in the wavelet transform. The second level generates four subsequences, H^2x, GHx, HGx, G^2x. If we repeat this process for J times, we end up having Jn expansion coefficients. It is easily seen that the computational cost of this whole process is about $O(Jn) \leq O(n \log n)$.

Because of the perfect reconstruction condition on H and G, each decomposition step is also considered as a decomposition of the vector space into mutually orthogonal subspaces. Let $\Omega_{0,0}$ denote the standard vector space $\mathbb{R}^n$. Let $\Omega_{1,0}$ and $\Omega_{1,1}$ be mutually orthogonal subspaces generated by the application of the projection operators H and G respectively to the parent space $\Omega_{0,0}$, i.e., $\Omega_{0,0} = \Omega_{1,0} \oplus \Omega_{1,1}$. The iterative decomposition process in the wavelet packet transform naturally generates subspaces of $\mathbb{R}^n$ of a binary tree structure where the nodes of the tree represent subspaces with different frequency localization

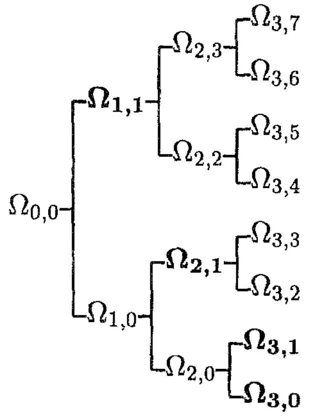


Fig. 1. A decomposition of $\Omega_{0,0}$ into the tree-structured subspaces using the wavelet packet transform (with $J = 3$). This tree includes the subspaces used in the wavelet transform which are indicated by bold fonts.

characteristics with $\Omega_{0,0}$ as the root node. The node $\Omega_{j,k}$ splits into the two orthogonal subspaces $\Omega_{j+1,2k}$ and $\Omega_{j+1,2k+1}$ by the operators H and G, respectively:

$$\Omega_{j,k} = \Omega_{j+1,2k} \oplus \Omega_{j+1,2k+1} \quad \text{for } j = 0, 1, \ldots, J,\ k = 0, \ldots, 2^j - 1.$$

The above figure shows the binary tree of the subspaces of $\Omega_{0,0}$:

As shown in Fig. 1, this binary tree includes the wavelet basis as a special case. Clearly, we have a redundant set of subspaces in the binary tree. In fact, it is easily proved that there are more than $2^{2^{(J-1)}}$ possible orthonormal bases in this binary tree; see e.g., [25, Chapter 7]. This binary tree is our main tool in this paper:

DEFINITION 1. A *dictionary of orthonormal bases* $\mathfrak{D}$ for $\mathbb{R}^n$ is a binary tree if it satisfies:

(a) Subsets of basis vectors can be identified with subintervals of $I = [0, n)$ of the form $I_{j,k} = [2^{n_0-j}k, 2^{n_0-j}(k+1))$, for $j = 0, 1, \ldots, J$, $k = 0, 1, \ldots, 2^j - 1$, where $J \leq n_0$.

(b) Each basis in the dictionary corresponds to a disjoint cover of I by intervals $I_{j,k}$.

(c) If $\Omega_{j,k}$ is the subspace identified with $I_{j,k}$, then $\Omega_{j,k} = \Omega_{j+1,2k} \oplus \Omega_{j+1,2k+1}$.

Each subspace $\Omega_{j,k}$ is spanned by 2^{n_0-j} basis vectors $\{w_{j,k,l}\}_{l=0}^{2^{n_0-j}-1}$. In the wavelet packet dictionary, the parameters k and l roughly indicate frequency bands and the location of the center of wiggles, respectively:[4] the vector $w_{j,k,l}$ is roughly centered at $2^j l$, has length of support $\approx 2^j$, and oscillates $\approx k$ times. By specifying a pair of QMFs, we obtain one dictionary which contains a large number of orthonormal bases of $\mathbb{R}^n$. In other words, we now have a large number of coordinate systems to "view our signals" at our disposal. An important question is how to select the best coordinate system efficiently for the problem at hand from this dictionary.

The *local trigonometric transforms* [26, 27, 21, 25] or lapped orthogonal transforms [28, 29] also form dictionaries of orthonormal bases. These can be considered as conjugates of the wavelet packet transforms: they partition the time axis smoothly and perform frequency analysis in each segment. In fact, Coifman and Meyer [26] showed that it is possible to partition the real-line into any disjoint intervals smoothly and construct orthonormal bases on each interval. Each basis function on an interval uses the signal values on the interval itself and on the adjacent intervals; hence it is named the "lapped" orthogonal transform. Since it partitions the time axis smoothly, these local cosine and sine transforms (LCT/LST) have less edge (or blocking) effects than the conventional discrete cosine/sine transforms (DCT/DST). A natural way to partition a given interval hierarchically is to segment it into dyadic subintervals recursively. This segmentation makes the number of signal samples contained in each subinterval a dyadic number (2^{n_0-j} at step j) if the length of the original signal is also a dyadic number (2^{n_0}). This enables one to utilize the fast DCT/DST-IV algorithm [30]. By this segmentation, the original interval $I = [0, n)$ is split into $[0, n/2)$ and $[n/2, n)$, and each subinterval is further split into half in a recursive manner. Let us set $I_{0,0} = I$ and let $I_{j,k}$ be a subinterval of I after jth iteration of the splitting process. Then we have a familiar relation

$$I_{j,k} = I_{j+1,2k} \cup I_{j+1,2k+1} \quad \text{for } j = 0, 1, \ldots, J,\ k = 0, 1, \ldots, 2^j - 1.$$

Now we can consider the subspaces $\Omega_{j,k}$ associated with the interval $I_{j,k}$. Then we obtain a binary tree of the subspaces with the same structure as the one shown in Fig. 1. Each subspace is spanned by the basis vectors $\{w_{j,k,l}\}_{l=0}^{2^{n_0-j}-1}$ where the triplet (j, k, l) now corresponds to scale, location (or window index) and frequency, respectively. For $j = 0$, this reduces to a simple DCT/DST. Hence, we can obtain two additional dictionaries of orthonormal bases using LCT/LST. The computational complexity to obtain this dictionary (or expanding a signal into this dictionary) is about $O(n[\log n]^2)$; see e.g., [25, Chapter 4].

For higher dimensional versions of wavelets, wavelet packets, and local trigonometric bases, see e.g., [23, Chapter 10], [31, 32], [25, Chapter 9], [33].

Once we obtain a dictionary of orthonormal bases, the question is how to select the best possible basis for the problem at hand from this collection of bases. Next, we review the best-basis algorithm of Coifman-Wickerhauser for signal compression since this contrasts very well with the basis selection algorithm for signal classification in Section 4.

3.3 Selection of a "Best Basis" from a Dictionary of Orthonormal Bases

An efficient coordinate system for representing a signal should give large magnitudes along a few axes and negligible magnitudes along most axes when the signal is expanded into the associated basis. We then need a measure to evaluate and compare the efficiency of many bases. Let $\mathfrak{J}$ denote this measure which is often called "information cost" function. There are several choices for $\mathfrak{J}$; see e.g., [25, Chapter 8], [10, Chapter 9], [34]. All of them essentially measure the "energy concentration" of the coordinate vector. A natural choice for this measure is the *Shannon entropy* of the coordinate vector [35]. Let us define the entropy of a nonnegative sequence $\boldsymbol{p} = \{p_i\}$ with $\sum_i p_i = 1$ by

$$H(\boldsymbol{p}) \overset{\Delta}{=} -\sum_i p_i \log p_i, \tag{1}$$

with the convention $0 \cdot \log 0 = 0$. For a signal $\boldsymbol{x}$, we set $p_i = (|x_i|/\|\boldsymbol{x}\|_r)^r$ where $\|\cdot\|_r$ is the ℓ^r norm and $1 \le r < \infty$ and define

$$H_r(\boldsymbol{x}) \overset{\Delta}{=} -\sum_i \frac{|x_i|^r}{\|\boldsymbol{x}\|_r^r} \log \frac{|x_i|^r}{\|\boldsymbol{x}\|_r^r}. \tag{2}$$

Often $r = 1$ or $r = 2$ is used. In this paper, we always use $r = 2$.

The "best-basis" algorithm of Coifman and Wickerhauser [20] was the first realization of the "best-basis paradigm" mainly aimed at signal compression. This method first expands a given *single* signal into a specified dictionary of orthonormal bases. Then a complete basis called a *best basis* (BB) which minimizes a certain information cost function such as entropy (2) is searched in this binary tree using the divide-and-conquer algorithm. More precisely, let $B_{j,k}$ denote a set of basis vectors belonging to the subspace $\Omega_{j,k}$ arranged as a matrix

$$B_{j,k} = \left(\boldsymbol{w}_{j,k,0}, \dots, \boldsymbol{w}_{j,k,2^{n_0-j}-1}\right)^T. \tag{3}$$

Now let $A_{j,k}$ be the best basis for the signal $\boldsymbol{x}$ restricted to the span of $B_{j,k}$ and let $\mathfrak{J}$ be an information cost function measuring the goodness of nodes (subspaces) for compression. The following best-basis algorithm essentially "prunes" this binary tree by comparing efficiency of each parent node with that of its two children nodes:

Algorithm 1 (The Best-Basis Algorithm [20]). *Given a vector $\boldsymbol{x}$,*

Step 0: *Choose a dictionary of orthonormal bases $\mathfrak{D}$ (i.e., specify QMFs for a wavelet packet dictionary or decide to use either the local cosine dictionary or the local sine dictionary) and specify the maximum depth of decomposition J and an information cost $\mathfrak{J}$.*

Step 1: *Expand $\boldsymbol{x}$ into the dictionary $\mathfrak{D}$ and obtain coefficients $\{B_{j,k}\boldsymbol{x}\}_{0\le j\le J,\ 0\le k\le 2^j-1}$.*

Step 2: *Set $A_{J,k} = B_{J,k}$ for $k = 0, \dots, 2^J - 1$.*

Step 3: *Determine the best subspace $A_{j,k}$ for $j = J - 1, \dots, 0,\ k = 0, \dots, 2^j - 1$ by*

$$A_{j,k} = \begin{cases} B_{j,k} \\ \quad \textit{if } \mathfrak{J}(B_{j,k}\boldsymbol{x}) \le \mathfrak{J}(A_{j+1,2k}\boldsymbol{x} \cup A_{j+1,2k+1}\boldsymbol{x}), \\ A_{j+1,2k} \oplus A_{j+1,2k+1} \\ \quad \textit{otherwise.} \end{cases} \tag{4}$$

To make this algorithm fast, the cost functional $\mathfrak{J}$ needs to be *additive*:

Definition 2. A map $\mathfrak{J}$ from sequences $\{x_i\}$ to $\mathbb{R}$ is said to be *additive* if $\mathfrak{J}(0) = 0$ and $\mathfrak{J}(\{x_i\}) = \sum_i \mathfrak{J}(x_i)$.

Thus, if $\mathfrak{J}$ is additive, then in (4) we have

$$\begin{aligned}\mathfrak{J}(A_{j+1,2k}\boldsymbol{x} \cup A_{j+1,2k+1}\boldsymbol{x}) \\ = \mathfrak{J}(A_{j+1,2k}\boldsymbol{x}) + \mathfrak{J}(A_{j+1,2k+1}\boldsymbol{x}).\end{aligned}$$

This implies that a simple addition suffices instead of computing the cost of union of the nodes. Although (1) is additive with respect to $\boldsymbol{p}$, $H_r(\boldsymbol{x})$ is not additive with respect to $\boldsymbol{x}$ in general. But it is easy to show that minimizing the additive measure

$$h_r(\boldsymbol{x}) \overset{\Delta}{=} -\sum_i |x_i|^r \log |x_i|^r \tag{5}$$

implies minimizing $H_r(\boldsymbol{x})$ since $H_r(\boldsymbol{x}) = h_r(\boldsymbol{x})/\|\boldsymbol{x}\|_r^r + \log \|\boldsymbol{x}\|_r^r$.

With the additive information cost function, we have the following proposition:

PROPOSITION 1 (Coifman & Wickerhauser [20]). *Algorithm 1 yields the best basis relative to $\mathfrak{D}$ and $\mathfrak{J}$ if $\mathfrak{J}$ is additive.*

See [20], [25, Chapter 8] for the proof.

The computational complexity of computing the best basis from a dictionary is $O(n \log n)$ for a wavelet packet dictionary and $O(n[\log n]^2)$ for a local trigonometric dictionary; it is dominated by the expansion of a signal into the dictionary and the cost for searching the best basis is about $O(n)$ because of the use of the divide-and-conquer algorithm. The reconstruction of the original vector from the best-basis coefficients has the same computational complexity.

3.4 *The Joint Best Basis*

To compress a given set of signals $\{x_i\}_{i=1}^N \subset \mathcal{X} \subset \mathbb{R}^n$ rather than a single signal, one of the well-known traditional methods is the *Karhunen-Loève transform* (KLT). Although it gives the minimum entropy basis among all possible orthonormal bases of $\mathbb{R}^n$ [36], there are several drawbacks; the main problem of the KLT is its computational cost $O(n^3)$ for diagonalizing the sample autocorrelation matrix of the signal set. In fact, its dependence on the eigenvalue system creates more problems: the sensitivity to the alignment of the signals and difficulty in capturing local features in the signals.

In [37] Wickerhauser proposed a method to overcome these problems of the KLT using the "best-basis paradigm," which is an extension to the best-basis method. Let us fix a dictionary $\mathfrak{D}$. Then, the idea is to use the energy distribution of the set of the signals to the coordinate axes in $\mathfrak{D}$ by computing $\sum_{i=1}^N (w_{j,k,l}^T x_i)^2$ for each (j, k, l) and organize them into a binary tree so that the divide-and-conquer algorithm can search a basis minimizing the entropy of the energy distribution from the tree-structured subspaces. Such a best basis is called the *joint best basis* (JBB) for $\{x_i\}_{i=1}^N$. In this paper, we will also use the term "best basis" as a joint best basis for simplicity. See [37], [25, Chapter 11] for the details of the algorithm and its computational cost.

3.5 *A Library of Orthonormal Bases*

We now consider a "meta" algorithm for the best-basis selection.

DEFINITION 3. A *library of orthonormal bases* for $\mathbb{R}^n$ is a collection of the dictionaries of orthonormal bases for $\mathbb{R}^n$.

This library of bases is more adaptable and versatile for representing various transient signals than a single dictionary of bases is. For example, if the signal consists of blocky functions such as acoustic impedance profiles of subsurface structure, the Haar-Walsh dictionary captures those discontinuous features both accurately and efficiently. If the signal consists of piecewise polynomial functions of order p, then the Daubechies wavelets/wavelet packets with filter length $L \geq 2(p + 1)$ or the coiflets with filter length $L \geq 3(p + 1)$ would be efficient because of the vanishing moment property (see e.g., [23] for the details of this property). If the signal has a sinusoidal shape or highly oscillating characteristics, the local trigonometric bases would do the job. Moreover, computational efficiency of this library is also attractive; the most expensive expansion in this library, i.e., the local trigonometric expansion, costs about $O(n[\log n]^2)$.

How can we choose the best dictionary from this library? The strategy of Coifman and Majid [38] is very simple: pick the one giving the minimum entropy among them.[5] More precisely, let $\mathfrak{L} = \{\mathfrak{D}_1, \ldots, \mathfrak{D}_M\}$ denote a library of orthonormal bases where $\mathfrak{D}_m$ represents a dictionary of orthonormal bases. For each dictionary $\mathfrak{D}_m$, the best basis $\mathfrak{B}_m$ of the signal x is computed by Algorithm 1. This generates M different sets of the expansion coefficients $\{\alpha_m\}_{m=1}^M$ of the signal. For each expansion coefficient set, entropy $h_2(\alpha_m)$ defined in (5) is computed and then the basis which gives the minimum entropy among M entropy values is selected as the "best of the best bases."

4 Construction of Local Discriminant Basis

In this section, we describe a fast algorithm to construct a good feature extractor. In particular, we follow the "best-basis paradigm" discussed in the previous section which permits a rapid [e.g., $O(n \log n)$] search among a library of orthonormal bases for the problem at hand. We first select basis functions which are well-localized in the time-frequency plane and which most discriminate given classes, and then the coordinates (expansion coefficients) of these basis functions are fed into LDA or CART. In order to fully utilize these classifiers, we must supply them *good* features (preferably just a few) and throw out useless part of the data. This improves both accuracy and speed of these classifiers.

4.1 Discriminant Measures

Recall that in the best-basis algorithm of Coifman and Wickerhauser, the efficiency of each subspace in the dictionary/library is measured by the Shannon entropy (1). This quantity measures the flatness of the energy distribution of the signal so that minimizing this leads to an efficient representation (or coordinate system) for the signal. For the classification problems, however, we need a measure to evaluate the power of discrimination of each subspace in the tree-structured subspaces rather than the efficiency in representation. Once the discriminant measure (or discriminant information function) is specified, we can compare the goodness of each node (subspace) for the classification problem to that of union of the two children nodes and can judge whether we should keep the children nodes or not, in the same manner as the best-basis search algorithm.

There are many choices for the discriminant measure (see e.g., [40, 41]); all of them essentially measure "statistical distances" among classes. For simplicity, let us first consider the two-class case. Let $\boldsymbol{p} = \{p_i\}_{i=1}^n, \boldsymbol{q} = \{q_i\}_{i=1}^n$ be two nonnegative sequences with $\sum p_i = \sum q_i = 1$ (which can be viewed as normalized energy distributions of signals belonging to class 1 and class 2, respectively). The discriminant information function $\mathcal{D}(\boldsymbol{p}, \boldsymbol{q})$ between these two sequences should measure how differently $\boldsymbol{p}$ and $\boldsymbol{q}$ are distributed. One natural choice for $\mathcal{D}$ is the so-called *relative entropy* (also known as *cross entropy*, *Kullback-Leibler distance*, or *I-divergence*) [42]:

$$I(\boldsymbol{p}, \boldsymbol{q}) \triangleq \sum_{i=1}^{n} p_i \log \frac{p_i}{q_i}, \tag{6}$$

with the convention, $\log 0 = -\infty$, $\log(x/0) = +\infty$ for $x \underset{\neq}{>} 0$, $0 \cdot (\pm\infty) = 0$. It is clear that $I(\boldsymbol{p}, \boldsymbol{q}) \geq 0$ and equality holds iff $\boldsymbol{p} \equiv \boldsymbol{q}$. This quantity is not a metric since it is not symmetric and does not satisfy the triangle inequality. But it measures the discrepancy of $\boldsymbol{p}$ from $\boldsymbol{q}$. Note that if $q_i = 1/n$ for all i, i.e., q_is are distributed uniformly, then $I(\boldsymbol{p}, \boldsymbol{q}) = -H(\boldsymbol{p})$, the negative of the entropy of the sequence $\boldsymbol{p}$ itself.

The relative entropy (6) is asymmetric in $\boldsymbol{p}$ and $\boldsymbol{q}$. For certain applications the asymmetry is preferred (see e.g., Section 7). However, if a symmetric quantity is preferred, one should use the *J-divergence* between $\boldsymbol{p}$ and $\boldsymbol{q}$ [42]:

$$J(\boldsymbol{p}, \boldsymbol{q}) \triangleq I(\boldsymbol{p}, \boldsymbol{q}) + I(\boldsymbol{q}, \boldsymbol{p}). \tag{7}$$

Another possibility of the measure $\mathcal{D}$ is a ℓ^2 analogue of $I(\boldsymbol{p}, \boldsymbol{q})$ [13]:

$$W(\boldsymbol{p}, \boldsymbol{q}) \triangleq \|\boldsymbol{p} - \boldsymbol{q}\|^2 = \sum_{i=1}^{n} (p_i - q_i)^2. \tag{8}$$

Clearly, ℓ^p $(p \geq 1)$ versions of this measure are all possible to use as $\mathcal{D}$.

To obtain a fast computational algorithm, the measure $\mathcal{D}$ should be *additive*:

DEFINITION 4. The discriminant measure $\mathcal{D}(\boldsymbol{p}, \boldsymbol{q})$ is said to be *additive* if

$$\mathcal{D}\left(\{p_i\}_{i=1}^n, \{q_i\}_{i=1}^n\right) = \sum_{i=1}^{n} \mathcal{D}(p_i, q_i) \tag{9}$$

The measures (6) (subsequently (7) as well) and (8) are both additive.

For measuring discrepancies among C distributions, $\boldsymbol{p}^{(1)}, \ldots, \boldsymbol{p}^{(C)}$, one may take $\binom{C}{2}$ pairwise combinations of $\mathcal{D}$:

$$\mathcal{D}\left(\{\boldsymbol{p}^{(c)}\}_{c=1}^C\right) \triangleq \sum_{i=1}^{C-1} \sum_{j=i+1}^{C} \mathcal{D}(\boldsymbol{p}^{(i)}, \boldsymbol{p}^{(j)}). \tag{10}$$

4.2 The Local Discriminant Basis Algorithm

The first step of our strategy for classification is to select a basis which attains the maximum discriminant information for given classes from a library of orthonormal bases. Let us first consider the selection of such a basis from a dictionary of orthonormal bases in the library. Given an additive discriminant measure $\mathcal{D}$, what quantity should be supplied to $\mathcal{D}$ to evaluate the discrimination power of each subspace $\Omega_{j,k}$ in the binary-tree-structured subspaces in the dictionary? In order to fully utilize the time-frequency localization characteristics of our dictionary of bases, we compute the following *time-frequency energy map* for each class and supply them to $\mathcal{D}$:

DEFINITION 5. Let $\{\boldsymbol{x}_i^{(c)}\}_{i=1}^{N_c}$ be a set of training signals belonging to class c. Then the *time-frequency energy map* of class c, denoted by Γ_c, is a table of real numbers specified by the triplet (j, k, l) as

$$\Gamma_c(j, k, l) \triangleq \sum_{i=1}^{N_c} \left(\boldsymbol{w}_{j,k,l}^T \boldsymbol{x}_i^{(c)}\right)^2 \Big/ \sum_{i=1}^{N_c} \|\boldsymbol{x}_i^{(c)}\|^2, \tag{11}$$

for $j = 0, \dots, J,\ \ k = 0, \dots, 2^j - 1,\ \ l = 0, \dots, 2^{n_0 - j} - 1$.

In other words, Γ_c is computed by accumulating the squares of expansion coefficients of the signals at each position in the binary tree followed by the normalization by the total energy of the signals belonging to class c. (This normalization may be important especially if there is significant differences in number of samples among classes.) In the following, we use the notation:

$$\mathcal{D}\big(\{\Gamma_c(j,k,\cdot)\}_{c=1}^{C}\big) = \sum_{l=0}^{2^{n_0-j}-1} \mathcal{D}(\Gamma_1(j,k,l), \dots, \Gamma_C(j,k,l)).$$

Here is an algorithm to select an orthonormal basis (from the dictionary) which maximizes the discriminant measure on the time-frequency energy distributions of classes. We call this a *local discriminant basis* (LDB). Similarly to the best-basis algorithm, let $B_{j,k}$ denote a set of basis vectors at the subspace $\Omega_{j,k}$ as defined in (3). Let $A_{j,k}$ represent the LDB (which we are after) restricted to the span of $B_{j,k}$. Also, let $\boldsymbol{\Delta}_{j,k}$ be a work array containing the discriminant measure of the subspace $\Omega_{j,k}$.

ALGORITHM 2 (The Local Discriminant Basis Selection Algorithm). *Given a training dataset $\mathcal{T}$ consisting of C classes of signals $\{\{\boldsymbol{x}_i^{(c)}\}_{i=1}^{N_c}\}_{c=1}^{C}$,*

```
Step 0: Choose a dictionary of orthonormal bases $\mathfrak{D}$ (i.e., specify QMFs for a wavelet packet dictionary or decide to use either the local cosine dictionary or the local sine dictionary) and specify the maximum depth of decomposition $J$ and an additive discriminant measure $\mathcal{D}$.
Step 1: Construct time-frequency energy maps $\Gamma_c$ for $c = 1, \dots, C$.
Step 2: Set
        $A_{J,k} = B_{J,k}$ and $\Delta_{J,k} = \mathcal{D}(\{\Gamma_c(J,k,\cdot)\}_{c=1}^{C})$ for $k = 0, \dots, 2^J - 1$.
Step 3: Determine the best subspace $A_{j,k}$ for $j = J-1, \dots, 0,\ k = 0, \dots, 2^j - 1$ by the following rule:
          Set $\Delta_{j,k} = \mathcal{D}(\{\Gamma_c(j,k,\cdot)\}_{c=1}^{C})$.
           If $\Delta_{j,k} \ge \Delta_{j+1,2k} + \Delta_{j+1,2k+1}$,
         then $A_{j,k} = B_{j,k}$,
         else $A_{j,k} = A_{j+1,2k} \oplus A_{j+1,2k+1}$ and $\Delta_{j,k} = \Delta_{j+1,2k} + \Delta_{j+1,2k+1}$.
Step 4: Order the basis functions by their power of discrimination (see below).
Step 5: Use k ($\le n$) most discriminant basis functions for constructing classifiers.
```

The selection (or pruning) process in Step 3 is fast, i.e., $O(n)$ since the measure $\mathcal{D}$ is additive. After this step, we have a complete orthonormal basis LDB. We now have the following proposition:

PROPOSITION 2. *The basis obtained by Step* 3 *of Algorithm* 2 *maximizes the additive discriminant measure $\mathcal{D}$ on the time-frequency energy distributions among all the bases in the dictionary $\mathfrak{D}$ obtainable by the divide-and-conquer algorithm.*

See [4] or [3] for the proof.

Once the LDB is selected, we can use all expansion coefficients of signals in this basis as features; however, if we want to reduce the dimensionality of the problem, Steps 4 and 5 are still necessary. In Step 4, there are several choices as a measure of discriminant power of an individual basis function. For simplicity in notation, let $\lambda = (j,k,l) \in \mathbb{Z}^3$ be a triplet specifying one of the LDB functions selected in Step 3, and let $\alpha_{\lambda,i}^{(c)} = \boldsymbol{w}_\lambda^T \boldsymbol{x}_i^{(c)}$, i.e., an expansion coefficient of $\boldsymbol{x}_i^{(c)}$ in the basis vector $\boldsymbol{w}_\lambda$. In the following, we list a few candidates for that measure:

(a) the discriminant measure of a single basis function $\boldsymbol{w}_\lambda$:

$$\mathcal{D}(\Gamma_1(\lambda), \dots, \Gamma_C(\lambda)). \tag{12}$$

(b) Fisher's class separability of the expansion coefficients in the basis function $\boldsymbol{w}_\lambda$:

$$\frac{\sum_{c=1}^{C} \pi_c \big(\mathrm{mean}_i(\alpha_{\lambda,i}^{(c)}) - \mathrm{mean}_c(\mathrm{mean}_i(\alpha_{\lambda,i}^{(c)}))\big)^2}{\sum_{c=1}^{C} \pi_c \mathrm{var}_i(\alpha_{\lambda,i}^{(c)})}, \tag{13}$$

where $\mathrm{mean}_i(\cdot)$ and $\mathrm{var}_i(\cdot)$ are operations to take the sample mean and variance with respect to the samples indexed by i, respectively.

(c) the robust version of (b):

$$\frac{\sum_{c=1}^{C} \pi_c \big|\mathrm{med}_i(\alpha_{\lambda,i}^{(c)}) - \mathrm{med}_c(\mathrm{med}_i(\alpha_{\lambda,i}^{(c)}))\big|}{\sum_{c=1}^{C} \pi_c\, \mathrm{mad}_i(\alpha_{\lambda,i}^{(c)})}, \tag{14}$$

where $\mathrm{med}_i(\cdot)$ and $\mathrm{mad}_i(\cdot)$ are operations to take the sample median and median absolute deviation with respect to the samples indexed by i, respectively.

See [40, 43] for more examples. We note that this step can also be viewed as a restricted version of the projection pursuit algorithm [43].

Step 5 reduces the dimensionality of the problem from n to k without losing the important discriminant information in terms of time-frequency energy distributions among classes. Thus many interesting statistical techniques which are usually computationally too expensive for n dimensional problems become feasible. How to select the best k is a tough interesting question. One possibility is to use model selection methods such as the minimum description length (MDL) criterion [17] (see also [3, Chapter 3]).

We can easily extend Algorithm 2 to a library of orthonormal bases. Let $\mathfrak{L} = \{\mathfrak{D}_1, \dots, \mathfrak{D}_M\}$ denote a library. Let $\mathfrak{B}_m$ be the LDB selected from the dictionary $\mathfrak{D}_m$. Each LDB $\mathfrak{B}_m$ is associated with the maximum value of a discriminant measure on the time-frequency energy distributions (relative to $\mathfrak{D}_m$) as shown in Proposition 2. Let Δ_m^* denote this maximum value. Then we can simply pick the basis giving the maximum value among $\{\Delta_m^*\}$: the "best" of the LDBs, $\mathfrak{B}_{m^*}$, is given by choosing m^* such that $\Delta_{m^*}^* = \max_{1 \le m \le M} \Delta_m^*$.

An extension of our algorithm to image classification problems is quite straightforward; one has only to replace the one-dimensional bases in the library by their two-dimensional versions. See e.g., [23, Chapter 10], [25, Chapter 9] for the details of the higher dimensional bases.

Remark. Our LDB method can be used for certain regression problems which are closely related to the classification problems. Let a training dataset $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$ consist of C classes of samples $\{\{(\boldsymbol{x}_i, y_i)\}_{i \in I_c}\}_{c=1}^C$ as before. Let us assume that the response y_i for $i \in I_c$ is now a real number conditioned as $y_i \in R_c = [a_c, b_c]$ and that $\cap_{c=1}^C R_c \neq \emptyset$. Under this assumption, suppose one wants to estimate the response y_i for a given input signal $\boldsymbol{x}_i$ rather than its class label or assignment. This situation is not really special; we often encounter this type of regression problems in medical and geological sciences where the objects are made in the course of nature. In [3, Chapter 6], [5], one can find real-life examples from the field of geophysical prospecting using the algorithms described in this paper.

5 Signal Classification Examples

To demonstrate the capability of the LDB method, we conducted two classification experiments using synthetic signals. In both cases, we specified three classes of signals by analytic formulas. For each class, we generated 100 training signals and 1000 test signals. We first constructed LDA-based classifiers and CTs using the training signals represented in the original coordinate (i.e., standard Euclidean) system. Then we fed the test signals into these classifiers. Next, we computed the LDB (using (10) as a discriminant measure and (12) for ordering the individual basis functions) on the training signals. Then we selected a small number of most discriminant basis functions, say about 10% of the dimensionality of the signals, and used these coordinates to construct LDA-based classifier and CTs. Finally the test signals were projected onto these selected LDB functions and then fed into these classifiers. For each method, we computed the misclassification rates on the training dataset and the test dataset. We repeated this procedure (including dataset generation) 10 times to get the average misclassification rates.

Example 5.1 (Triangular Waveform Classification). This is an example for classification originally examined in [7]. The dimensionality of the signal was extended from 21 in [7] to 32 for the dyadic dimensionality requirement of the bases under consideration. Three classes of signals were generated by the following formulas:

$$\begin{aligned} x^{(1)}(i) &= uh_1(i) + (1-u)h_2(i) + \epsilon(i) \quad \text{for Class 1,} \\ x^{(2)}(i) &= uh_1(i) + (1-u)h_3(i) + \epsilon(i) \quad \text{for Class 2,} \\ x^{(3)}(i) &= uh_2(i) + (1-u)h_3(i) + \epsilon(i) \quad \text{for Class 3,} \end{aligned}$$

where $i = 1, \dots, 32$, $h_1(i) = \max(6 - |i-7|, 0)$, $h_2(i) = h_1(i-8)$, $h_3(i) = h_1(i-4)$, u is a uniform random variable on the interval $(0, 1)$, and $\epsilon(i)$ are the standard normal variates. Figure 2 shows five sample waveforms from each class.

The LDB was computed from the wavelet packet coefficients with the 6-tap coiflet filter [23]. Then the five most discriminant coordinates were selected. In Fig. 3, we compare the top five vectors from LDA and LDB. Only the top two vectors were useful in LDA in this case. The top five LDB vectors look similar to the functions h_j or their derivatives whereas it is difficult to interpret the LDA vectors.

The misclassification rates are given in Table 1.[6] The best result so far was obtained by applying LDA to the top 5 LDB coordinates. We would like to note that according to Breiman et al. [7], the Bayes error of this example is about 14%.

Example 5.2 (Signal Shape Classification). The second example is a signal shape classification problem. In

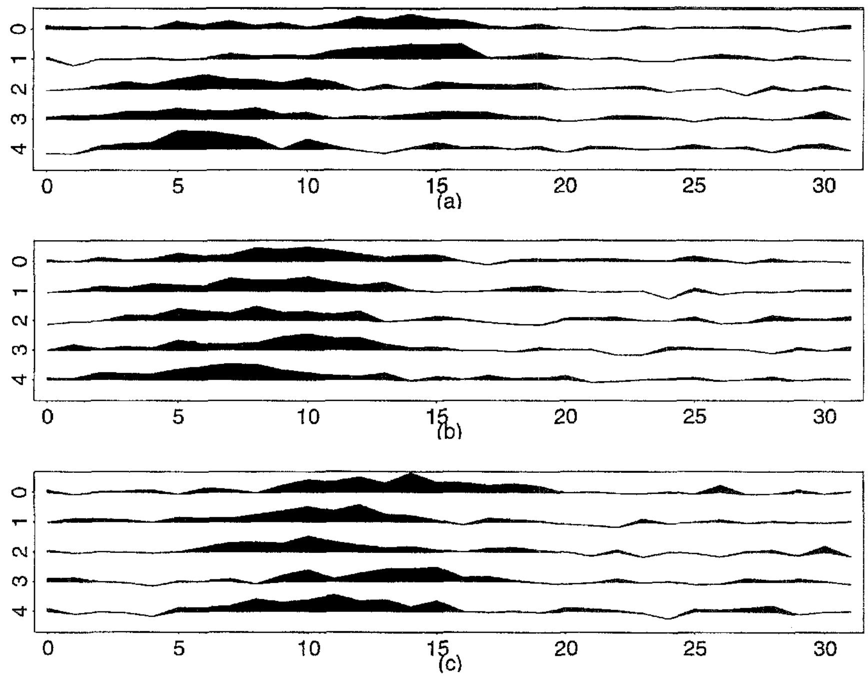


Fig. 2. Sample waveforms of Example 5.1: (a) Class 1, (b) Class 2, and (c) Class 3.

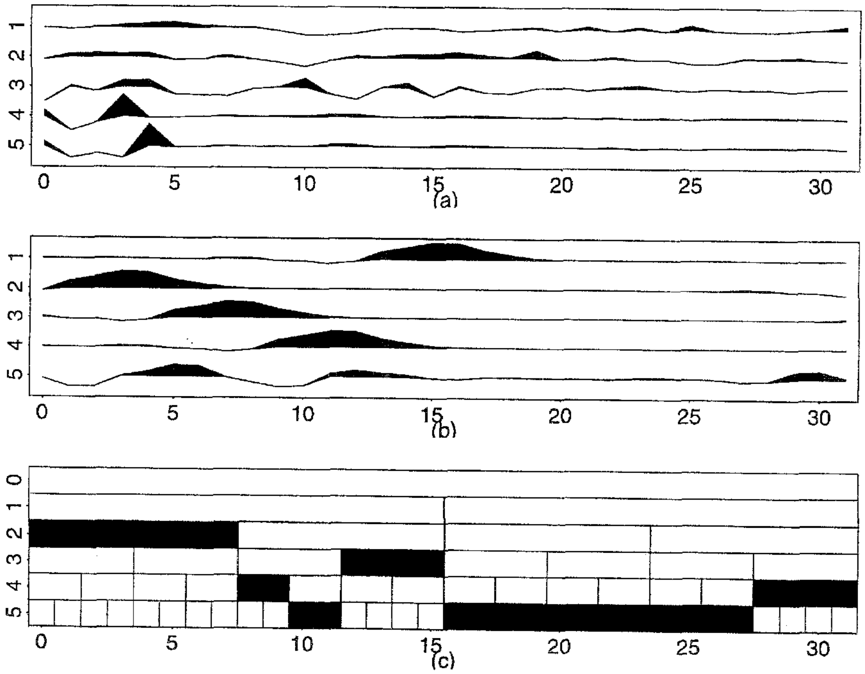


Fig. 3. Comparison of LDA and LDB vectors of Example 5.1: (a) Top five LDA vectors. (b) Top 5 LDB vectors. (c) The subspaces selected as the LDB.

Table 1. Misclassification rates (averages over 10 simulations) of Example 5.1.

	Error rate (%)	
Method	Training	Test
LDA on STD	12.00	22.65
CT on STD	7.03	29.31
LDA on LDB5	14.13	**16.16**
CT on LDB5	8.13	21.86
CT on LDB	6.30	23.78

Notation: STD = the standard Euclidean coordinates; LDB5 = the top 5 LDB coordinates; LDB = all the LDB coordinates. The smallest error on the test dataset is shown in bold font.

this example, we try to classify synthetic noisy signals with various shapes, amplitudes, lengths, and positions into three possible classes. More precisely, sample signals of the three classes were generated by:

$$\begin{aligned} c(i) &= (6+\eta)\cdot\chi_{[a,b]}(i)+\epsilon(i) && \text{for "cylinder" class,}\\ b(i) &= (6+\eta)\cdot\chi_{[a,b]}(i) \\ &\quad\cdot(i-a)/(b-a)+\epsilon(i) && \text{for "bell" class,}\\ f(i) &= (6+\eta)\cdot\chi_{[a,b]}(i) \\ &\quad\cdot(b-i)/(b-a)+\epsilon(i) && \text{for "funnel" class,} \end{aligned}$$

where $i = 1,\dots,128$, a is an integer-valued uniform random variable on the interval $[16, 32]$, $b - a$ also obeys an integer-valued uniform distribution on $[32, 96]$, η and $\epsilon(i)$ are the standard normal variates, and $\chi_{[a,b]}(i)$ is the characteristic function on the interval $[a, b]$. Figure 4 shows five sample waveforms from each class. If there is no noise, we can characterize the "cylinder" signals by two step edges and constant values around the center, the "bell" signals by one ramp and one step edge in this order and positive slopes around the center, and the "funnel" signals by one step edge and one ramp in this order and negative slopes around the center.

The 12-tap coiflet filter [23] was used for the LDB selection. Then the 10 most important coordinates were selected. In Fig. 5, we compare the top 10 LDA and LDB vectors. Again, only the top two vectors were used for classification in LDA case. These LDA vectors are very noisy and it is difficult to interpret what information they captured. On the other hand, we can observe that the top 10 LDB vectors are located around the edges or the centers of the signals. Also note that some of the vectors work as a smoother (low pass filter) and the others work as a edge detector (band pass filter), so that the resulting expansion coefficients carry the information on the edge positions and types.

The misclassification rates in this case are displayed in Table 2.

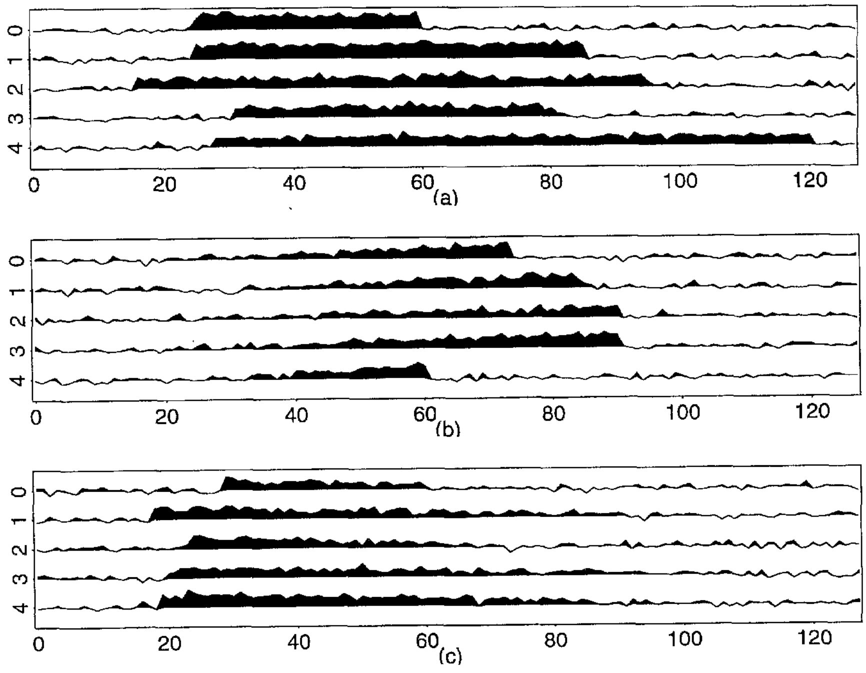


Fig. 4. Sample waveforms of Example 5.2: (a) "cylinder" class, (b) "bell" class, and (c) "funnel" class.

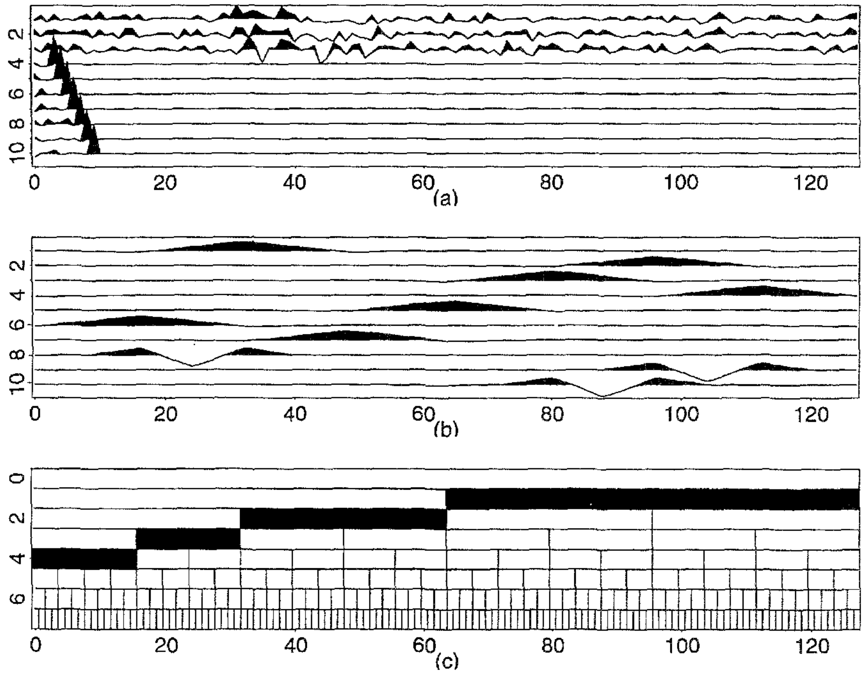


Fig. 5. Comparison of LDA and LDB vectors of Example 5.2: (a) Top 10 LDA vectors. (b) Top 10 LDB vectors. (c) The subspaces selected as the LDB.

As expected, LDA applied to the original coordinate system was almost perfect with respect to the training data, but it adapted too much to the features specific to the training data, and lost its generalization power; when applied to the new test dataset, it did not work well. The best result was obtained using the CT on the top 10 LDB coordinates in this example. The misclassification rates of the training data and the test data are very close here; that is, the algorithm really "learned" the structures of signals. This best CT is shown in Fig. 6.

If the tree-based classification is combined with the coordinate system capturing local information in the time-frequency plane, the interpretation of the result becomes so explicit and easy: in Fig. 6 we find that the LDB coordinate #1 is checked first. If this is less than 10.0275, it is immediately classified as "bell." From Fig. 5(b), we observe that the LDB function #1 is located around $i = 30$ which, in fact, coincides with the starting position (the parameter a in the formulas) of various signals. Around this region, both the cylinder and the funnel signals have sharp step edges. On the other hand, the bell signals start off linearly. Thus CART algorithm found that the LDB function #1 is the most important coordinate in this example. Separating the cylinder class from the funnel class turned out to be more difficult because of the large variability of the ending positions. This resulted in the more complicated structure of the right branch from the root node. But we can still obtain the intuitive interpretation: the first node in the right branch (with "cylinder" label) from the root node is split into either "funnel" or "cylinder" depending on the LDB coordinate #5 which is located around the middle of the axis ($i = 64$). If there were no noise, the cylinder signals would have constant values around this area whereas the funnel signals would decrease linearly here; the LDB obtained this important information by removing the noise. (See also [4], [3, Chapter 4] for the details of the relation between LDB and denoising.) One can continue the

Table 2. Misclassification rates (averages over 10 simulations) of Example 5.2.

	Error rate (%)	
Method	Training	Test
LDA on STD	0.83	12.31
CT on STD	2.83	11.28
LDA on LDB10	7.00	8.37
CT on LDB10	2.67	**5.54**
CT on LDB	2.33	7.60

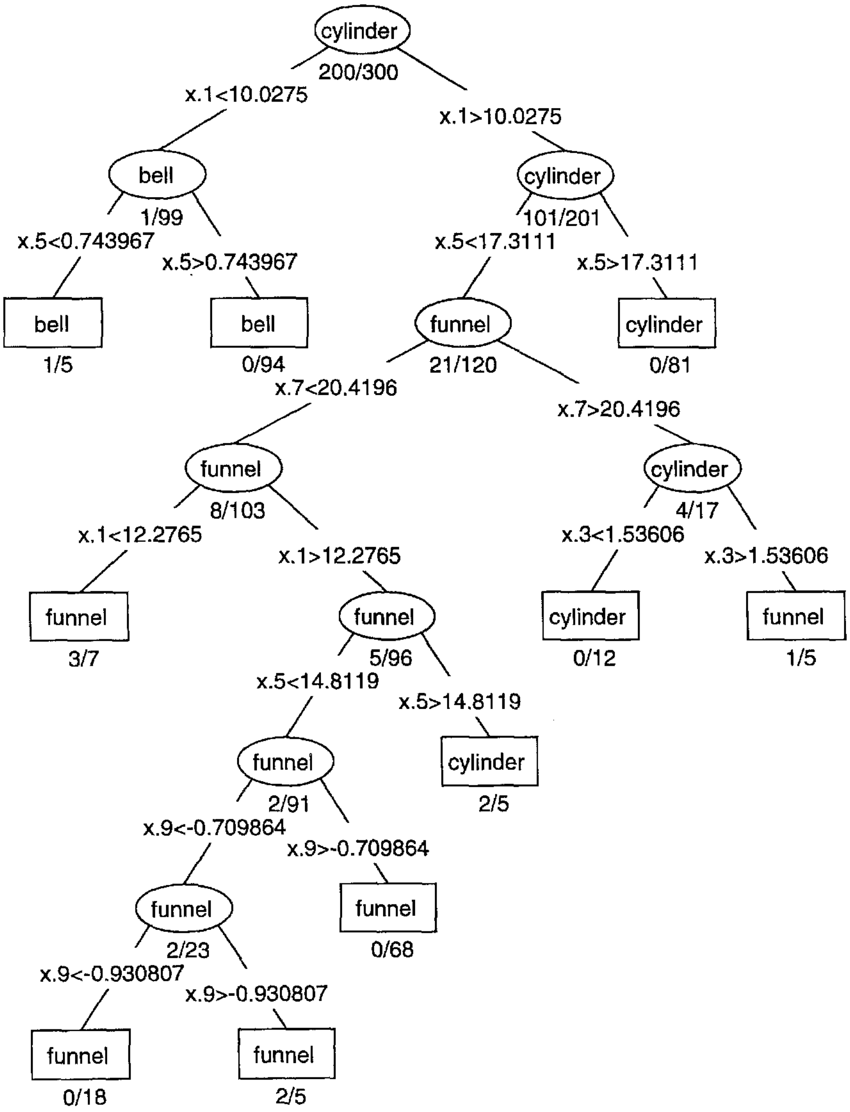


Fig. 6. The best classification tree generated on the LDB coefficients in Example 5.2. Nodes are represented by ellipses (interior nodes) and rectangles (terminal nodes/leaves). The node labels are the predicted class names which are "cylinder," "bell," and "funnel" in this case. The ratio displayed under each node represents the misclassification rate of the cases reached to that node. The splitting rules are displayed on the edges connecting nodes. The rule "x.1 < 10.0275" implies "if the first coordinate value of the input signal is less than 10.0275, then go to this branch."

interpretation in a similar manner for all remaining nodes.

From these examples, we can see that it is more important to select the good features than to select the best possible classifier without supplying the good features; each classifier has its advantages and disadvantages [9], i.e., the best classifier heavily depends on the problem (e.g., LDA was better than CART in Example 5.1 whereas the situation was opposite in Example 5.2.) By supplying a handful of good features, we can greatly enhance the performance of classifiers.

6 Image Texture Classification using LDB

In this section, we describe our preliminary experiments on image texture classification using the two-dimensional version of the LDB algorithm. Image texture analysis, classification, and segmentation are

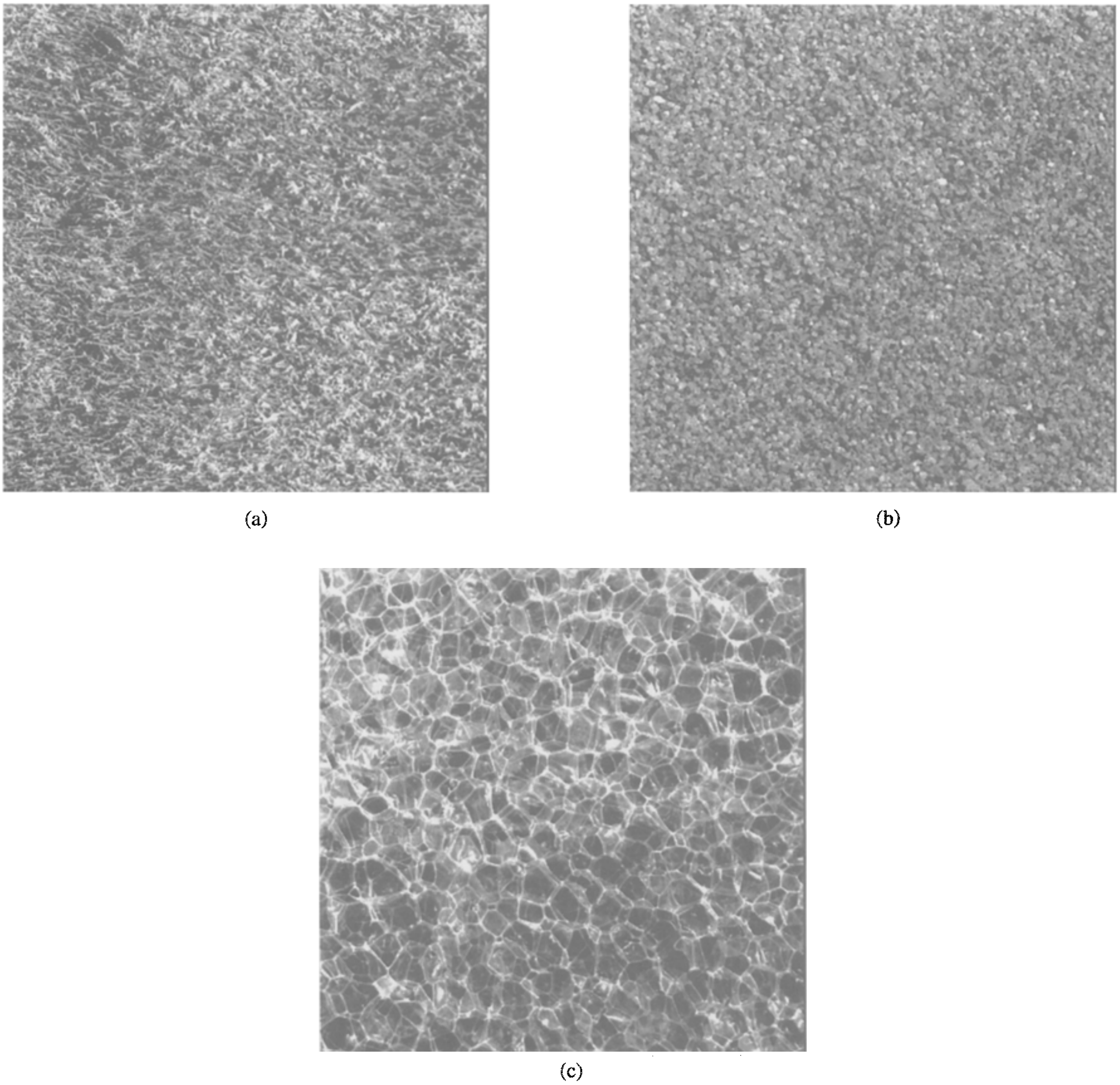


Fig. 7. Three texture images from the Brodatz book [45]: (a) Grass lawn (D9), (b) Beach sand (D29), and (c) Plastic bubbles (D112). Each of them has 512 × 512 pixels with 8-bit gray levels per pixel.

very important area of research. Our experiments here are of simple nature and thorough investigation is in progress. Another approach using the wavelet packets can be found in [44].

6.1 Data Description and Preparation

Figure 7 shows three textured images used in this experiment, i.e., "grass lawn," "beach sand," and "plastic bubbles," digitized from the Brodatz book [45]. Each of these images has 512 by 512 pixels with 8-bit gray levels per pixel.

As a training dataset, we selected 100 subimages of size 128 by 128 pixels from each original image by specifying the upper-left coordinates of the subimages randomly without replacement. For a test dataset, we generated another 100 random upper-left coordinates and selected 100 subimages of the same size. We note that the coordinates for the training and test datasets are mutually exclusive; each subimage in the datasets is unique although there exist subimages which share some common area in the original image. Then, each subimage was normalized to have zero mean and unit variance prior to the LDB computation.

6.2 Training Process Stabilization by "Spin Cycle"

Our bases in the library, i.e., wavelets, wavelet packets, and local trigonometric bases, do not have the translation-invariance property. Since we do want to analyze the texture properties, but do not concern the locations of individual edges or transients which form texture elements (or textons), lack of translation invariance is problematic. Thus, to make the analysis and classification processes more insensitive to translations, we applied the *spin cycle* procedure: increase the number of samples of the training and the test datasets by creating their translated versions. In particular, we used translations along the diagonal direction of images with periodic boundary condition. This "spin cycle" procedure plays an important role for other applications such as denoising [46], [3, Chapter 3], [47].

It turns out that increasing the number of samples by the spin cycle procedure is, in spirit, very similar to the "bagging" (*b*ootstrap *agg*regat*ing*) procedure proposed by Breiman [48]. This method tries to stabilize certain classifiers by: 1) generating multiple versions of training dataset by the bootstrap method [49], 2) constructing a classifier for each training dataset, and 3) predicting the class of test samples by the majority vote on the predictions by all these classifiers.

For the examples studied in Section 5, the spin cycle procedure would not improve the results so much because: 1) in Example 5.1, the locations of the triangles in each class do not change, and 2) in Example 5.2, the signal components in the samples are already shifted randomly (within certain intervals) by definition.

6.3 Experimental Results

In our experiments, we created 10 translated versions of each subimage (i.e., totally 1,100 subimages for each class in the training dataset as well as in the test dataset) by the spin cycle procedure. Then, we computed an energy map on the phase space (a higher dimensional version of the time-frequency plane for space and spatial frequency characterization) of each class using the tensor product of the 12-tap coiflet filters into the full scales (seven decomposition levels). Then, a quadtree representing subspaces was built and associated with these three phase-space energy maps. Using the symmetric relative entropy, a two-dimensional LDB was selected from the quadtree-structured subspaces. Figure 8 shows the partitioning pattern of the

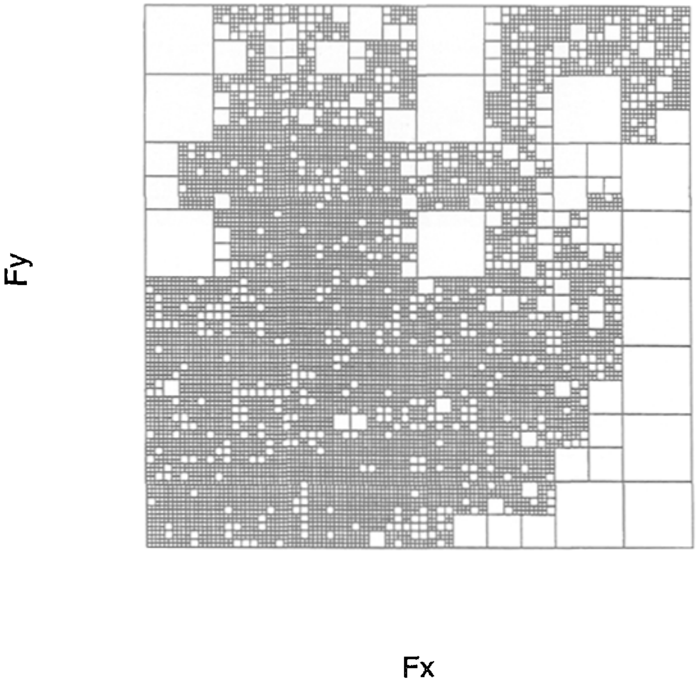


Fig. 8. A partition of the spatial-frequency plane by the two-dimensional LDB for the texture classification experiment. The horizontal and vertical axes correspond to spatial frequencies on x and y directions, respectively. Thus, the DC component is located at the lower-left corner of this figure.

Table 3. Misclassification rates of the texture classification experiment.

	Error rate (%)	
Method	Training	Test
LDA on LDB10	28.00	34.67
CT on LDB10	9.00	33.67
LDA on LDB100	1.33	**5.00**
CT on LDB100	4.33	28.67

spatial frequency plane by the LDB. We observe that most of the cells are small except certain regions, in particular, around the right edge of the plane (i.e., regions of high horizontal frequencies). This implies that the most of the texture information in this experiment are of global nature in the space domain and of local nature in the spatial-frequency domain.

After the LDB selection, each LDB coordinate was sorted in terms of its discrimination power using the coordinate-wise symmetric relative entropy (12). Finally, the most important LDB coefficients (after squared for representing energy) were used to construct CT and LDA classifiers. In this experiment, we selected the top 10 and the top 100 coordinates. To compute the misclassification rates for the training dataset, we adopted the "voting" procedure: we assigned each original subimage (not the translated versions) the class label which was the majority of the class predictions of the original itself and its 10 translated versions, and then computed the misclassification rates on these original subimages. In the test stage, we expanded each test subimage into the LDB and the top 10 and 100 coefficients (after squared) were supplied to the CT and LDA classifiers. We also applied the same voting procedure to compute the misclassification rates for the test dataset. Table 3 summarizes the misclassification rates. This table clearly reveals that the top 10 LDB coordinates are not enough to classify these textures. Even with the top 100 coordinates, the CT does not work well, and the resulting CT, in fact, has an extremely complicated structure (many branches and nodes). On the other hand, the LDA constructed from the top 100 coordinates works well with 5% misclassification rate on the test dataset. These strongly suggest that the structure of class distributions in the feature space spanned by the top 100 LDB vectors are at least linearly oriented with respect to the LDB axes (or maybe more complicated).

Figure 9 shows the most important six basis functions for this experiment. We note that they are not the top six of the LDB functions ordered by the coordinate-wise symmetric relative entropy, but the basis functions which corresponds to the top six most energetic components in the first LDA vector of length 100. We are currently investigating a better strategy for ordering the individual LDB functions other than (12) (e.g., (13), (14), and other measures listed in [40], [43], etc.) after the LDB selection since the dimensionality reduction is particularly important for the image processing applications because of their large dimensionality.

We note that the direct application of the CT and LDA on the standard Euclidean coordinate system do not work since 1) the dimensionality of the problem is too high ($128 \times 128 = 16{,}384$), and 2) the local spatial variability on the individual pixel level is too high to yield meaningful results.

7 Signal/Background Separation by LDB

LDB vectors can also be used as a tool for extracting signal component from the data obscured by some unwanted noise or "background" (which may not be random). Let class 1 consist of a signal plus noise or a signal plus "background" and let class 2 consist of a pure noise or "background". Then, by selecting the LDB maximizing $\mathcal{D}$ between class 1 and class 2, we can construct the best basis for denoising arbitrary noise or pulling a signal out of a textured background. In this application, the asymmetric relative entropy (6) makes more sense than the symmetric version (7).

We show one example here. As "background" (class 2), we generated 100 synthetic sinusoid with random phase as $b(k) = \sin(\pi(k/32 + u))$, where $k = 1, \ldots, 128$, and u is a uniform random variable on $(0, 1)$. As class 1 samples, we again generated 100 "backgrounds," and added a small spike (as a "signal" component) for each sample vector randomly between $20 \leq k \leq 60$, i.e., $x(k) = \sin(\pi(k/32 + u)) + 0.01\delta_{k,r}$, where $\delta_{k,r}$ is the Kronecker delta and r is an integer-valued uniform random variable on the interval [20, 60]. Figure 10 shows how these "backgrounds" were removed. figure 10(a) shows 10 sample vectors of class 1. We can hardly see the spikes. Then we transformed both class 1 and 2 samples by the discrete sine transform (DST) into "frequency" domain. Figure 10(b) shows the transformed version of Fig. 10(a). Then these DST coefficients of both classes were supplied to the LDB algorithm of Section 4 using the local sine basis dictionary (which essentially does segmentation in frequency domain). After the LDB was found, the basis vectors were sorted by (12).

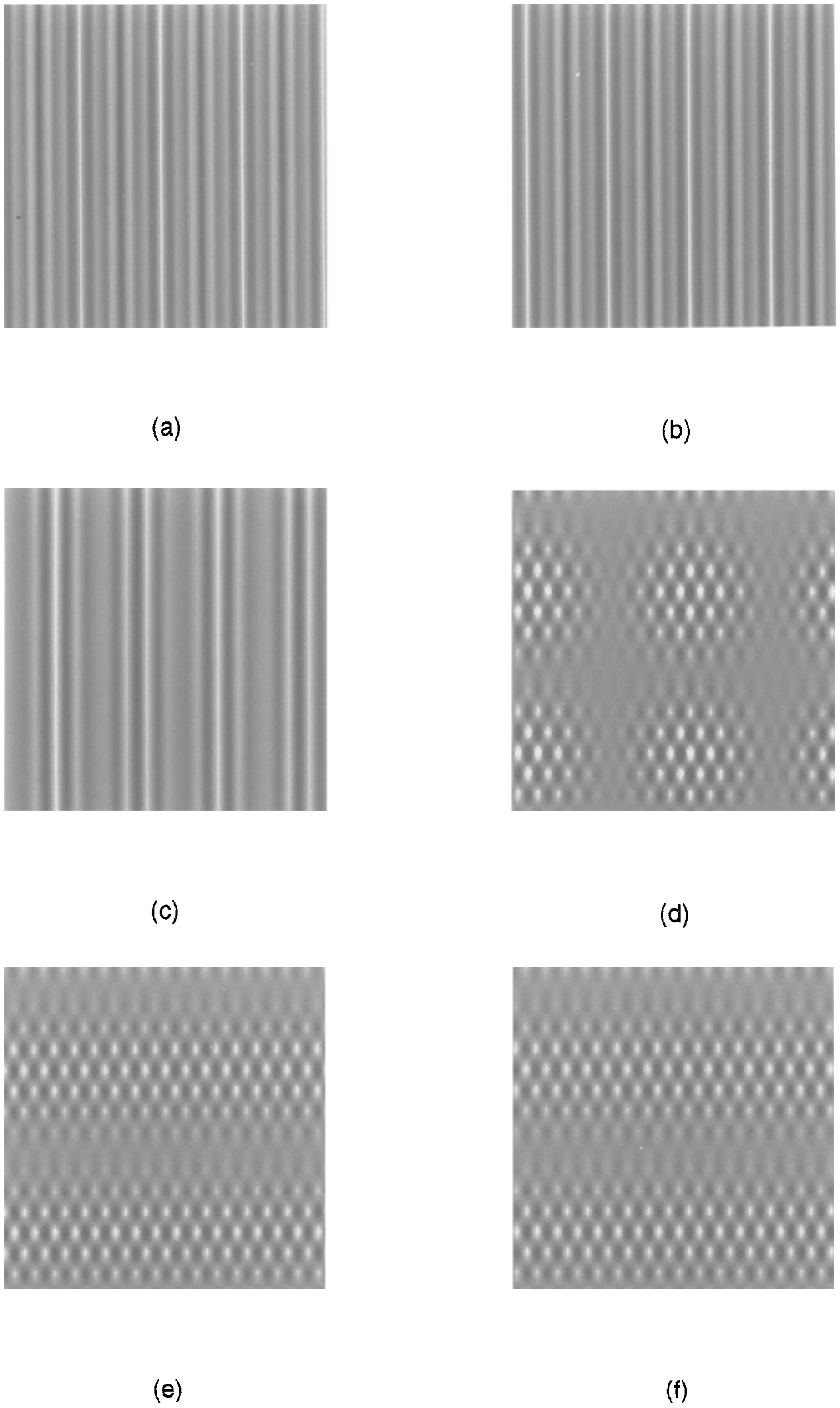


Fig. 9. The most significant 6 LDB functions for the texture classification experiment. Each basis function has 128 by 128 pixels. Observe that a rather high characteristic frequency exists in all basis functions and this roughly corresponds to the grain sizes of the "beach sand."

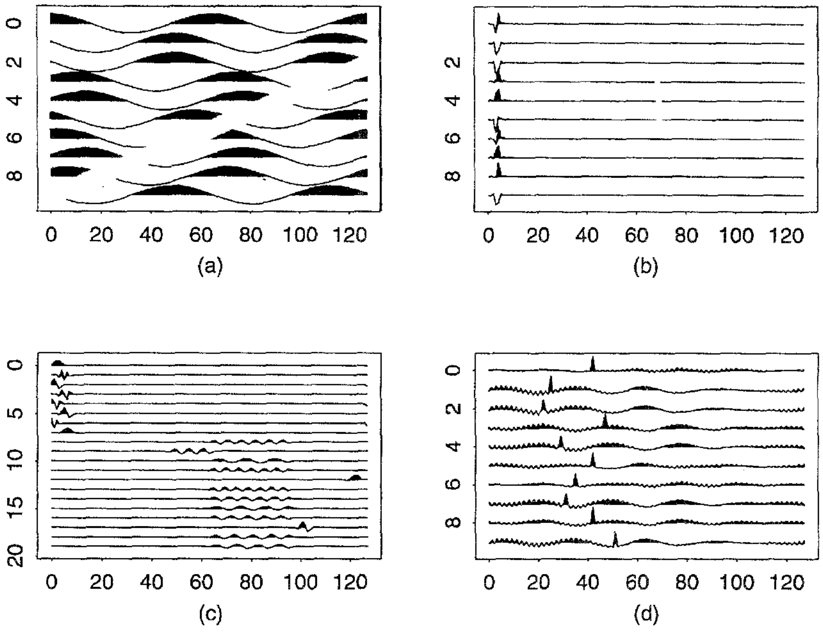


Fig. 10. The signal/"background" separation algorithm in action. (a) Ten samples of Class 1 vectors, i.e., sinusoids plus spikes. (b) DST coefficients of vectors in (a). (c) Top 20 LDB vectors using the local sine dictionary on the frequency domain. (d) Reconstructed spikes after removing the "background".

The top 20 LDB vectors are displayed in Fig. 10(c). We can clearly see that the top eight basis vectors are concentrated around low frequency region and other vectors are located in higher frequency region. We regard the subspace spanned by these eight LDB vectors as "background" using the *a priori* knowledge that the "background" component consists of only low frequency component. The reason why these vectors have large values in (12) is that the "background" parts of class 1 samples are different from class 2 samples in phase, and the DST is not a shift-invariant transform. After removing the component belonging to this "background" subspace, we reconstructed the "signal" component of class 1 samples by inverse DST which are shown in Fig. 10(d). We can clearly see the spikes now. The LDB thus can improve the algorithm of extracting "coherent" component from the data by Coifman, Majid, and Wickerhauser [38], [50] if we know the statistics of the background *a priori* or have actual pure background signals.

A similar idea for multidimensional signals has been proposed by Harlan et al. [51] in the geophysical prospecting field; they considered the problem of removing linear and hyperbolic structures from seismic images using the Radon and the generalized Radon transforms. The key observation is that the structural components (e.g., lines and hyperbolas) in the images can be well-compressed or "focused" in the certain transformed domains (e.g., the Radon, the generalized Radon transformed domains). On the other hand, the unstructured components or backgrounds are "defocused" in these domains. Based on this observation, the thresholding operation in the transformed domain is applied and only the "focused" objects in the transformed domain remain. Then the inverse transform only reconstructs the structural components and eliminates backgrounds. In this sense, the "structure" strongly depends on the transform under consideration. Our philosophy is to use the library of bases in Section 3; we have a large collection of transforms each of which can represent and adapt to many different "structures" in signals. For images or multidimensional signals, it is not simple to determine which basis should be included in the library of bases because: 1) there are many possible two-dimensional bases both separable and nonseparable (see e.g., [23, Chapter 10], [31, 32] for the details), and 2) the computational cost is much higher ($\approx O(n^2 \log_4 n^2)$ for an image of n rows and n columns) compared with the one-dimensional bases. Here is the place to use *a priori* information carefully to restrict the number of bases or dictionaries in the library to achieve both the computational efficiency and the representation power of the bases. Signal/noise separation for images including texture segmentation is our important future project.

8 Conclusion

We have described an algorithm to construct an adaptive local orthonormal basis [*local discriminant basis* (LDB)] for classification problems by selecting a basis from a library of orthonormal bases using a discriminant measure (e.g., relative entropy). The basis functions generated by this algorithm can capture relevant local features (in both time and frequency) in data. LDB provides us with better insight and understanding of relationships between the essential features of the input signals and the corresponding outputs (class names), and enhances the performance of classifiers. We have demonstrated that LDB can also be used for pulling out signal component from the data consisting of signals plus "backgrounds."

Acknowledgment

The authors would like to thank Prof. Andrew Laine at University of Florida for providing us the digitized photographs of the Brodatz textures.

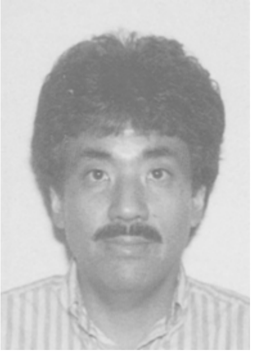
Notes

1. From now on, unless mentioned otherwise, time and frequency also means space and spatial frequency respectively.
2. The sample mean operation $(1/N_c)\Sigma_{i=1}^{N_c}$ in this subsection can be replaced by expectation E_c for general cases; however, in this paper, we focus our attention on the cases of a finite number of samples, so we stay with the sample mean operations.
3. In the tree-based classification module included in the S-PLUS$^{\text{TM}}$ package [14] (the extended version of the statistical language S$^{\text{TM}}$ [15], [16]) which we intensively use to test our algorithm, the split stops by default if either the number of samples belonging to that node becomes less than 10 or the deviance of that node becomes less than 1% of the deviance of the root node.
4. The original binary tree generated by successive applications of H and G is called "Paley ordered" and the frequency band of $\Omega_{j,k}$ is not monotonically increasing as a function of k. This behavior is corrected by the so-called "Gray code" permutation; see [25] for the details.
5. The purpose of [38] is not the compression but the noise removal. Another way of choosing a basis for the noise removal using minimum description length (MDL) criterion can be found in [39], [3, Chapter 3].
6. We do not show the error rates of LDA on all the LDB coordinates in this table since this is the same as the ones of LDA on STD theoretically (and this fact was also confirmed numerically).

References

1. R.R. Coifman and N. Saito, "Constructions of local orthonormal bases for classification and regression," *C. R. Acad. Sci. Paris, Série I*, Vol. 319, pp. 191–196, 1994.
2. N. Saito and R.R. Coifman, "Local discriminant bases," in *Mathematical Imaging: Wavelet Applications in Signal and Image Processing*, A.F. Laine and M.A. Unser (eds.), *Proc. SPIE 2303*, pp. 2–14, 1994.
3. N. Saito, *Local Feature Extraction and Its Applications Using a Library of Bases*, Ph.D. Thesis, Dept. of Mathematics, Yale University, New Haven, CT 06520 USA, Dec. 1994.
4. N. Saito and R.R. Coifman, "Local feature extraction for classification and regression using a library of bases," in preparation.
5. N. Saito and R.R. Coifman, "Extraction of geological information from acoustic well-logging waveforms using time-frequency atoms," *Geophysics*, 1995 (submitted).
6. R.A. Fisher, "The use of multiple measurements in taxonomic problems," *Ann. Eugenics*, Vol. 7, pp. 179–188, 1936.
7. L. Breiman, J.H. Friedman, R.A. Olshen, and C.J. Stone, *Classification and Regression Trees*, Chapman & Hall: New York, 1993. Previously published by Wadsworth & Brooks/Cole in 1984.
8. T.M. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Trans. Inform. Theory*, Vol. IT-13, pp. 21–27, 1967.
9. B.D. Ripley, "Statistical aspects of neural networks," in *Networks and Chaos: Statistical and Probabilistic Aspects*, O.E. Barndorff-Nielsen, J.L. Jensen, D.R. Cox, and W.S. Kendall (eds.), Ch. 2, pp. 40–123, Chapman & Hall: New York, 1993.
10. K. Fukunaga, *Introduction to Statistical Pattern Recognition*, Academic Press: San Diego, CA, second edition, 1990.
11. S.M. Weiss and C.A. Kulikowski, *Computer Systems That Learn: Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning, and Expert Systems*, Morgan Kaufmann: San Francisco, CA, 1991.
12. G.J. McLachlan, *Discriminant Analysis and Statistical Pattern Recognition*, John Wiley & Sons: New York, 1992.
13. S. Watanabe, *Pattern Recognition: Human and Mechanical*, John Wiley & Sons: New York, 1985.
14. StatSci, *S-PLUS Reference Manual, Vol. 1 & 2, version 3.2*, Seattle, WA, Dec. 1993.
15. R.A. Becker, J.M. Chambers, and A.R. Wilks, *The New S Language: A Programming Environment for Data Analysis and Graphics*, Chapman & Hall: New York, 1988.
16. J.M. Chambers and T.R. Hastie, *Statistical Models in S*, Chapman & Hall: New York, 1992.
17. J. Rissanen, *Stochastic Complexity in Statistical Inquiry*, World Scientific: Singapore, 1989.
18. J.R. Quinlan and R.L. Rivest, "Inferring decision trees using the minimum description length principle," *Information and Control*, Vol. 80, pp. 227–248, 1989.
19. C.S. Wallace and J.D. Patrick, "Coding decision trees," *Machine Learning*, Vol. 11, pp. 7–22, 1993.

20. R.R. Coifman and M.V. Wickerhauser, "Entropy-based algorithms for best basis selection," *IEEE Trans. Inform. Theory*, Vol. 38, pp. 713–719, 1992.
21. Y. Meyer, *Wavelets: Algorithms and Applications*, SIAM: Philadelphia, PA, 1993. Translated and revised by R.D. Ryan.
22. N. Saito and G. Beylkin, "Multiresolution representations using the auto-correlation functions of compactly supported wavelets," *IEEE Trans. Signal Processing*, Vol. 41, pp. 3584–3590, 1993.
23. I. Daubechies, *Ten Lectures on Wavelets*, SIAM: Philadelphia, PA, 1992.
24. Y. Meyer, *Wavelets and Operators*, Cambridge University Press: New York, 1993. Translated by D.H. Salinger.
25. M.V. Wickerhauser, *Adapted Wavelet Analysis from Theory to Software*, A.K. Peters: Wellesley, MA, 1994.
26. R.R. Coifman and Y. Meyer, "Remarques sur l'analyse de fourier à fenêtre," *C. R. Acad. Sci. Paris, Série I*, Vol. 312, pp. 259–261, 1991.
27. P. Auscher, G. Weiss, and M.V. Wickerhauser, "Local sine and cosine bases of Coifman and Meyer and the construction of smooth wavelets," in *Wavelets: A Tutorial in Theory and Applications* C.K. Chui (ed.), pp. 237–256, Academic Press: San Diego, CA, 1992.
28. H.S. Malvar, "The LOT: transform coding without blocking effects," *IEEE Trans. Acoust., Speech, Signal Processing*, Vol. 37, pp. 553–559, 1989.
29. H.S. Malvar, "Lapped transforms for efficient transform/subband coding," *IEEE Trans. Acoust., Speech, Signal Processing*, Vol. 38, pp. 969–978, 1990.
30. K.R. Rao and P. Yip, *Discrete Cosine Transform: Algorithms, Advantages, andApplications*, Academic Press: San Diego, CA, 1990.
31. J. Kovačević and M.Vetterli, "Nonseparable multidimensional perfect reconstruction filter banks and wavelet bases for R^n," *IEEE Trans. Inform. Theory*, Vol. 38, pp. 533–555, 1992.
32. K. Gröchenig and W.R. Madych, "Multiresolution analysis, Haar bases, and self-similar tilings of R^n," *IEEE Trans. Inform. Theory*, Vol. 38, pp. 556–568, 1992.
33. M.V. Wickerhauser, "High-resolution still picture compression," *Digital Signal Processing: A Review Journal*, Vol. 2, pp. 204–226, 1992.
34. N. Otsu, "Mathematical studies on feature extraction in pattern recognition," (in Japanese), Researches of the Electrotechnical Laboratory, No. 818, Electrotechnical Laboratory, 1-1-4, Umezono, Sakura-machi, Niihari-gun, Ibaraki, Japan, July 1981.
35. C.E. Shannon and W.Weaver, *The Mathematical Theory of Communication*, The University of Illinois Press: Urbana, IL, 1949.
36. S. Watanabe, "Karhunen-Loève expansion and factor analysis: theoretical remarks and applications," in *Trans. 4th Prague Conf. Inform. Theory, Statist. Decision Functions, Random Processes*, Prague, 1967, pp. 635–660.
37. M.V. Wickerhauser, "Fast approximate factor analysis," in *Curves and Surfaces in Computer Vision and Graphics II, Proc. SPIE* 1610, pp. 23–32, 1991.
38. R.R. Coifman and F. Majid, "Adapted waveform analysis and denoising," in *Progress in Wavelet Analysis and Applications*, Y. Meyer and S. Roques (eds.), pp. 63–76, Editions Frontieres: B.P.33, 91192 Gif-sur-Yvette Cedex, France, 1993.
39. N. Saito, "Simultaneous noise suppression and signal compression using a library of orthonormal bases and the minimum description length criterion," in *Wavelets in Geophysics*, E. Foufoula-Georgiou and P. Kumar (eds.), Ch. XI, pp. 299–324, Academic Press: San Diego, CA, 1994.
40. M. Basseville, "Distance measures for signal processing and pattern recognition," *Signal Processing*, Vol. 18, pp. 349–369, 1989.
41. J.N. Kapur and H.K. Kesavan, *Entropy Optimization Principles with Applications*, Academic Press: San Diego, CA, 1992.
42. S. Kullback and R.A. Leibler, "On information and sufficiency," *Ann. Math. Statist.*, Vol. 22, pp. 79–86, 1951.
43. P.J. Huber, "Projection pursuit (with discussion)," *Ann. Statist.*, Vol. 13, pp. 435–525, 1985.
44. T. Chang and C.-C.J. Kuo, "Texture analysis and classification with tree-structured wavelet transform," *IEEE Trans. Image Processing*, Vol. 2, pp. 429–441, 1993.
45. P. Brodatz, *Textures: A Photographic Album for Artists and Designers*, Dover: New York, 1966.
46. L. Woog, *Wavelet-packet based signal enhancement and denoising algorithms*, Ph.D. Thesis, Dept. of Comput. Sci., Yale University, 1995, in preparation.
47. R.R. Coifman and D. Donoho, "Translation-invariant denoising," in *Wavelets and Statistics*, A. Antoniadis (ed.), Springer-Verlag: New York, 1995.
48. L. Breiman, "Bagging predictors," Dept. of Statistics, Univ. of California, Berkeley, CA, Tech. Rep. 421, Sep. 1994.
49. B. Efron and R.J. Tibshirani, *An Introduction to the Bootstrap*, Chapman & Hall: New York, 1993.
50. R.R. Coifman and M.V. Wickerhauser, "Wavelets and adapted waveform analysis," *Wavelets: Mathematics and Applications*, J. Benedetto and M. Frazier (eds.), Ch. 10, CRC Press: Boca Raton, FL, 1993.
51. W.S. Harlan, J.F. Claerbout, and F. Rocca, "Signal/noise separation and velocity estimation," *Geophysics*, Vol. 49, pp. 1869–1880, 1984.



Naoki Saito received the B. Eng. and the M. Eng. degrees in mathematical engineering from the University of Tokyo, Japan, in 1982 and 1984, respectively, and the Ph.D. degree in applied mathematics from Yale University in 1994. In 1984, he joined Nippon Schlumberger K. K., Fuchinobe, Japan, and in 1986, he transferred to Schlumberger-Doll Research, Ridgefield, CT where he is currently a research scientist. His research interest includes feature extraction, classification, regression, pattern recognition, and statistical signal/image processing.

Dr. Saito is a member of the Institute of Electrical and Electronics Engineers (IEEE), the Institute of Mathematical Statistics (IMS), the Society of Exploration Geophysicists (SEG), and the Society for Industrial and Applied Mathematics (SIAM).

Ronald R. Coifman is a professor of mathematics at Yale University. He received his Ph.D. from the University of Geneva in 1965. Prior to coming to Yale in 1980, he was a professor at Washington University in St. Louis. His recent publications have been in the areas of nonlinear Fourier analysis, wavelet theory, numerical analysis and scattering theory. He is currently leading a research program to develop new wavelet-based mathematical tools for feature extraction, recognition, and denoising.

Dr. Coifman is a member of the American Mathematical Society, and is on the board of governors of the Institute for Mathematics and its Applications. He was chairman of the Yale mathematics department 1986–89. He founded FMA&H, a small technology transfer company specializing in the conversion of theoretical mathematical tools into engineering software for fast computation and processing.