PERGAMON

Pattern Recognition 35 (2002) 2841–2852

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

Discriminant feature extraction using empirical probability density estimation and a local basis library

Naoki Saito[a,*], Ronald R. Coifman[b,c], Frank B. Geshwind[b,c], Fred Warner[b,c]

[a]Department of Mathematics, University of California, One Shields Avenue, Davis, CA 95616, USA
[b]Department of Mathematics, Yale University, New Haven, CT 06520, USA
[c]Fast Mathematical Algorithms and Hardware Corp., Hamden, CT 06514, USA

Received 18 December 2000; accepted 20 November 2001

Abstract

The authors previously developed the so-called local discriminant basis (LDB) method for signal and image classification problems. The original LDB method relies on differences in the time–frequency energy distribution of each class: it selects the subspaces where these energy distributions are well separated by some measure such as the Kullback–Leibler divergence. Through our experience and experiments on various datasets, however, we realized that the time–frequency energy distribution is not always the best quantity to analyze for classification. In this paper, we propose to use the discrimination of coordinates based, instead, on empirical probability densities. That is, we estimate the probability density of each class in each coordinate in the wavelet packet/local trigonometric bases after expanding signals into such bases. We then evaluate a power of discrimination of each subspace by selecting the m most discriminant coordinates in terms of the "distance" among the corresponding densities (e.g., by the Kullback–Leibler divergence among the densities). This information is then used for selecting a basis for classification. We will demonstrate the capability of this algorithm using both synthetic and real datasets. © 2002 Pattern Recognition Society. Published by Elsevier Science Ltd. All rights reserved.

Keywords: Local feature extraction; Pattern classification; Density estimation; Kullback–Leibler divergence; Hellinger distance

1. Introduction

Recently, the authors introduced the concept of the so-called *local discriminant basis* (LDB) for signal and image classification problems [1, 2 (Chapter 4), 3, 4]. This method first decomposes available training signals in a *time–frequency dictionary* (also known as a *dictionary of orthonormal bases*), which is a large collection of basis functions (such as wavelet packets and local trigonometric functions) localized both in time and in frequency. Then, signal energies at the basis coordinates are accumulated for each signal class separately to form a time–frequency energy distribution per class. Based on the differences among these energy distributions (measured by a certain "distance" functional), a complete orthonormal basis called LDB, which "can see" the distinguishing signal features among signal classes, is selected from the dictionary. After the basis is determined, the expansion coefficients in the most important several coordinates (features) are fed into a traditional classifier such as linear discriminant analysis (LDA) or a classification tree (CT). Finally, the corresponding coefficients of test signals are computed and fed to the classifier to predict their classes.

This LDB concept has become increasingly popular and has been applied to a variety of classification problems including geophysical acoustic waveform classification [5,6], radar signal classification [7], and classification of neuron firing patterns of monkeys to different stimuli [8]. Through

[*] Corresponding author. Tel.: +530-754-2121; fax: +530-752-6635.

E-mail addresses: saito@math.ucdavis.edu (N. Saito), coifman@math.yale.edu (R.R. Coifman), geshwind@math.yale.edu (F.B. Geshwind), warner@math.yale.edu (F. Warner).

0031-3203/02/$22.00 © 2002 Pattern Recognition Society. Published by Elsevier Science Ltd. All rights reserved.
PII: S0031-3203(02)00019-5

these studies, we have found that the criterion used in the original LDB algorithm—the one based on the differences of the time–frequency energy distributions among signal classes—is not always the best one to use. Consider an artificial problem as follows. Suppose one class of signals consists of a single basis function in a time–frequency dictionary with its amplitude 10 and they are embedded in white Gaussian noise (WGN) with zero mean and unit variance. The other class of signals consists of the same basis function but with its amplitude -10 again embedded in the WGN process. Their time–frequency energy distributions are clearly identical. Therefore, we cannot select the right basis function as a discriminator. This simple counterexample suggests that we should also consider the differences of the distributions of the expansion coefficients in each basis coordinate. In this example, all coordinates except the one corresponding to the single basis function have the same Gaussian distribution. The probability density function (PDF) of the projection of input signals onto this one basis function should reveal twin peaks around ± 10.

In this paper we propose a new LDB algorithm based on the differences among coordinate-wise PDFs as a basis selection criterion and we explain similarities and differences among the original LDB algorithm and the new LDB algorithm.

2. Steps toward the original LDB

In this section, we review various feature extraction strategies and the original LDB algorithm. Before proceeding further, let us set up some notation. Let $\mathscr{X} \times \mathscr{Y}$ be the set of all pairs of input signals and their corresponding class labels $(\boldsymbol{x}, y)$. We call $\mathscr{X} \subseteq \mathbb{R}^n$ an *input signal space* and $\mathscr{Y} = \{1, \ldots, K\}$ an *output response space* which is simply a set of possible class labels (names). For most of the signal classification problems in which we are interested, the dimension n is rather large. For example, in medical X-ray tomography, we typically have $n \approx 512^2$, and in seismic signals, $n \approx 4000$. A *classifier* is a function $d : \mathscr{X} \to \mathscr{Y}$ which assigns a class name to each input signal $\boldsymbol{x} \in \mathscr{X}$. Note that if we define $A_y \triangleq \{\boldsymbol{x} \in \mathscr{X} : d(\boldsymbol{x}) = y\}$, then the A_y's are disjoint and $\mathscr{X} = \bigcup_{y=1}^{K} A_y$. A *learning* (or *training*) dataset $\mathscr{T}$ consists of N pairs of input signals and the corresponding class names: $\mathscr{T} = \{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_N, y_N)\}$. Let N_y be a number of class y signals in $\mathscr{T}$, and, hence, $N = \sum_{y=1}^{K} N_y$. We often write $\boldsymbol{x}_i^{(y)}$ to emphasize membership in class y. We also assume the availability of another dataset $\mathscr{T}'$ which is independent of $\mathscr{T}$ and sampled from the same probability model. This is called a *test* dataset and is used for evaluation of classifiers. Let us now introduce a probability model. Let $P(A, y)$ be a probability on $\mathscr{X} \times \mathscr{Y} (A \subset \mathscr{X}, y \in \mathscr{Y})$. Let $(\boldsymbol{X}, Y) \in \mathscr{X} \times \mathscr{Y}$ be a random sample from this probability distribution

$$P(A, y) \triangleq P(\boldsymbol{X} \in A, Y = y) = \pi_y P(\boldsymbol{X} \in A \mid Y = y),$$

where $\pi_y \triangleq P(Y = y)$ is a prior probability of class y. In practice, we often assume $\pi_y = N_y / N$.

The first and most direct approach to the classification problem is

Approach 0. *Construct the best possible classifier using information of $P(A, y)$.*

This naturally leads to the concept of the *Bayes classifier*, but we quickly realize that it is extremely difficult to obtain in practice.

2.1. The Bayes classifier

If we know the true probability distribution $P(A, y)$, then the optimal classifier is the so-called *Bayes classifier* d_B and is defined as

$$P(d_B(\boldsymbol{X}) \neq Y) \leqslant P(d(\boldsymbol{X}) \neq Y),$$

for any other classifier $d(\boldsymbol{X})$. Then, the *Bayes* misclassification rate $R_B \triangleq P(d_B(\boldsymbol{X}) \neq Y)$ is clearly the minimum among the rates obtained by all possible classifiers using information of $P(A, y)$. In fact, assuming the existence of the conditional PDF $p(\boldsymbol{x} \mid y)$ such that $P(A \mid y) = \int_A p(\boldsymbol{x} \mid y)\,\mathrm{d}\boldsymbol{x}$, it is well known (see e.g., Ref. [9, p.14]) that the Bayes classifier is to *assign $\boldsymbol{x}$ to class k if $\boldsymbol{x} \in A_k^*$*, where $A_k^* = \{\boldsymbol{x} \in \mathscr{X} : p(\boldsymbol{x} \mid k)\pi_k = \max_{y \in \mathscr{Y}} p(\boldsymbol{x} \mid y)\pi_y\}$. This Bayes classifier, however, is impossible to construct. First of all, $p(\boldsymbol{x} \mid y)$ is not known in practice. It is even difficult to estimate $p(\boldsymbol{x} \mid y)$ computationally using available training samples because of the high dimensionality of the input space $\mathscr{X}$ (*curse of dimensionality*); we need a huge number of training samples to get a reliable estimate of $p(\boldsymbol{x} \mid y)$. Therefore, we need to reduce the dimensionality of the problem without losing important information for classification.

2.2. Feature extraction, dimension reduction, and projection pursuit

Faced with the curse of dimensionality and having such difficulty in constructing Bayes classifiers, the extraction of important features becomes essential. We want to keep only important information and discard the irrelevant information for classification purposes. Humans perform this kind of "feature compression" on a daily basis when facing classification and discrimination tasks. As Scott mentions in his book [10, Chapter 7], this strategy is also supported by the empirical observation that multivariate data in $\mathbb{R}^n$ are almost never n-dimensional, and lower dimensional structures often exist within the data. In other words, a signal classification problem often has an *intrinsic dimension* $m < n$. (Note that this problem is clearly different from that of signal compression, even for the same dataset. In this sense, the intrinsic dimension is goal-dependent.) Therefore, it would be much more efficient and effective to analyze the data and build classifiers in this smaller dimensional subspace $\mathscr{F}$ of $\mathscr{X}$, if possible. We call $\mathscr{F}$ a *feature space*, and a map f: $\mathscr{X} \rightarrow \mathscr{F}$ a *feature extractor*. Then, the key is how to construct this "good" feature space $\mathscr{F}$ consisting of discriminant features and design the corresponding feature extractor f. If we precisely know the underlying physical and mathematical models of the problem, then we can design a mechanism to extract specific features relevant for that problem and may obtain its intrinsic dimension. It is often difficult, however, to set up exact mathematical models for the problem we are interested in, such as medical and geophysical diagnostics. Therefore, we want to adopt exploratory approaches; we want to find discriminant features by automatic procedures and examine their effectiveness. In turn, this may shed light on the underlying physics of the problem.

Based on the above observations, the next approach to signal classification problems can be stated as follows.

Approach 1. *Extract the best possible features from the signals and construct the best possible classifier on these features.*

This approach can be symbolically written as

$$d = g \circ f, \tag{1}$$

where f is a feature extractor and $g : \mathscr{F} \rightarrow \mathscr{Y}$ is a classifier using signal features. Ideally, mapping input signals to $\mathscr{F}$ should reveal separate clusters of points corresponding to signal classes and ease the subsequent classification tasks. In general, f can be nonlinear, e.g., neural network feature extractors such as self-organizing maps [11]. Although these types of feature extractors have interesting capabilities, they almost always require nonlinear optimization procedures which are computationally expensive and have difficulty in finding global optima. What about using some linear transforms as f? This question naturally leads to the concept of projection pursuit (PP).

Projection pursuit (PP), which was originally proposed by Kruskal [12] and was implemented, experimented upon, and named by Friedman and Tukey [13], is one of the few techniques which does dimensionality reduction. The original purpose of PP was to pick "interesting" low-dimensional projections of high-dimensional point clouds automatically by numerically optimizing a certain objective function or *projection index*. One can find a sequence of best 1D projections by optimizing the projection index, then removing the structure that makes this direction interesting, and iterating. As one can see, the idea of PP is very general and was extended for various purposes including density estimation, regression, classification, and clustering (see excellent expository papers by Huber [14] and by Jones and Sibson [15]). In particular, by changing the projection index appropriately, many of the classical multivariate data analysis techniques, such as principal component analysis (PCA) and linear discriminant analysis (LDA), are shown to be the special cases of PP. Therefore, PP with a projection index measuring discriminant power of projections (or coordinates) seems an attractive approach for feature extraction and dimension reduction. The problem, however, still exists: (1) a straightforward application of PP may still be computationally too expensive for high-dimensional problems; (2) sequentially obtaining the best 1D projections may be too "greedy". It may miss the important 2D structures. After all, "the best two measurements are not the two best" [16]; (3) interpretation of the results may also become difficult because the best 1D projection squeezes all discriminant features in the signals—however separated in time and frequency domains—into a single feature vector.

2.3. The original LDB method

Faced with the above difficulties, one may wonder how to approach the problem. One point of view involves the use of a so-called *time–frequency dictionary*. This is a large collection of specific basis vectors in $\mathscr{X}$ organized in a hierarchical binary tree structure. This dictionary is redundant and contains up to $n(1 + \log_2 n)$ basis vectors (also called *time–frequency atoms*) which have specific time–frequency localization properties. Examples include wavelet packets and local trigonometric functions of Coifman and Meyer; see [17–19] and references therein for the detailed properties of these basis functions. This dictionary of bases is huge, containing more than 2^n different complete orthonormal bases [18, p. 256]. This dictionary, however, offers us at least two things: (1) efficiency in representing features localized both in time and in frequency; (2) efficiency in computing such features. The former is particularly useful for representing nonstationary signals or discontinuous signals. Moreover, this property makes interpretation of classification results far easier than does either Approach 0 or Approach 1. In terms of computational efficiency, expanding a signal of length n into such a tree-structured bases is fast, i.e., $O(n \log n)$ for a wavelet packet dictionary and $O(n[\log n]^2)$ for a local trigonometric dictionary. Moreover, the Coifman–Wickerhauser best-basis algorithm [18, (Chapter 8)] provides an $O(n)$ search for a basis tailored to some specific need.

The best-basis algorithm of Coifman and Wickerhauser, however, was developed mainly for signal compression problems. The LDB algorithm was developed to fully utilize these properties of the time–frequency dictionaries and the best-basis algorithm for signal classification and discrimination problems [1,2 (Chapter 4), 3,4]. It is much more specific than Approach 0 and 1, but it offers a computationally efficient dimension reduction and extraction of local signal features. It is also "modest" in the sense that it picks a set of good coordinates from a finite collection of orthonormal bases rather than a sequence of the absolutely best 1D projections without constraint. This philosophy can be phrased as

Approach 2. *Extract the most discriminant features from a time–frequency dictionary, construct several* *classifiers on these features, and pick the best one among them.*

For a specific classifier g (which can be any conventional classifier of one's choice, such as LDA, CT, or neural networks, etc.), this approach can be written as

$$d = g \circ \Theta_m \circ \Psi. \tag{2}$$

The feature extractor here consists of $\Psi \in \mathrm{O}(n)$, an n-dimensional orthogonal matrix (i.e., an orthonormal basis) selected from a time–frequency dictionary, and a *feature selector* Θ_m which selects the most important $m(<n)$ coordinates (features) from n-dimensional coordinates. Most statistical literature focuses on the performance and statistical properties of various classifiers g used in Eqs. (1) and (2). Some literature discusses the feature selector Θ_m given a set of features. On the other hand, both the original and the new LDB methods focus on f, in particular, how to select Ψ from a finite collection of bases.

Let $\mathscr{D}$ be this time–frequency dictionary consisting of a collection of basis vectors $\{\boldsymbol{w}_i\}$, $i = 1,\ldots,n(1+\log_2 n)$. $\mathscr{D}$ can also be expressed as a list of all possible orthonormal bases $\{B_j\}$, $j = 1,\ldots,M$ with $M > 2^n$. Let $B_j = \{\boldsymbol{w}_{j_1},\ldots,\boldsymbol{w}_{j_n}\}$ and let $\mathscr{E}(B_j)$ be a measure of efficacy of the basis B_j for classification tasks. Then, both the original and the new LDB are selected rapidly by the best-basis algorithm using the following criterion:

$$\Psi = \arg\max_{B_j \in \mathscr{D}} \mathscr{E}(B_j). \tag{3}$$

The heart of the matter is this measure of efficacy $\mathscr{E}$. Here we describe the one adopted in the original LDB method. In the next section, we consider the various possibilities of this measure and propose the new LDB method.

Let $\Gamma^{(y)}(\boldsymbol{w}_j)$ be a normalized total energy of class y signals along the direction $\boldsymbol{w}_j$:

$$\Gamma^{(y)}(\boldsymbol{w}_j) \triangleq \frac{\sum_{i=1}^{N_y} |\boldsymbol{w}_j \cdot \boldsymbol{x}_i^{(y)}|^2}{\sum_{i=1}^{N_y} \|\boldsymbol{x}_i^{(y)}\|^2}, \tag{4}$$

where $\cdot$ denotes the standard inner product in $\mathbb{R}^n$. We refer to the tree-structured set of normalized energies $\{\Gamma^{(y)}(\boldsymbol{w}_j) : \boldsymbol{w}_j \in \mathscr{D}\}$ as the *normalized time–frequency energy map* of class y. Let $\boldsymbol{\Gamma}^{(y)}(B_j) = (\Gamma^{(y)}(\boldsymbol{w}_{j_1}), \ldots, \Gamma^{(y)}(\boldsymbol{w}_{j_n}))$ be a vector of such normalized energies corresponding to the basis B_j which is extracted from the time–frequency energy map. In the original LDB algorithm, we examined several functionals $\mathscr{E}$ measuring "distances" among K vectors, $\boldsymbol{\Gamma}^{(1)}(B_j),\ldots,\boldsymbol{\Gamma}^{(K)}(B_j)$, such as relative entropy, Hellinger distances, and simple ℓ^2 distances. See Eqs. (2)–(5) in Ref. [6] where we compared their classification performance. We note that the functionals such as relative entropy and Hellinger distance were originally created to measure the "distances" or deviations among PDFs [20], and in the original LDB algorithm, the normalized time–frequency energy maps are viewed as PDFs and the distances among them are computed for basis evaluation.

3. The new local discriminant bases

In this section, we use the probability model to reinterpret the original LDB algorithm and then propose the new basis selection criteria.

3.1. Discriminant measures for 1D projections

Let us first reconsider what is a good 1D projection for classification and discrimination in general. The more specific issues for the LDB methods are treated in the next subsection.

If we project an input signal $\boldsymbol{X} \in \mathscr{X}$ onto a unit vector $\boldsymbol{w}_i \in \mathscr{D}$, then its projection (or coordinate) $Z_i \triangleq \boldsymbol{w}_i \cdot \boldsymbol{X}$ is a random variable. We use the notation $Z_i^{(y)}$ if we want to emphasize the projection of class y signals. We are interested to know how Z_i is distributed for each signal class so that we can quantify the efficacy of the direction $\boldsymbol{w}_i$ for classification. We refer to such a projection index as a *discriminant measure*. We also use the term *discriminant power* of $\boldsymbol{w}_i$ which is the actual value of such a measure evaluated at $\boldsymbol{w}_i$. We can think of four possibilities of such a measure:

Type I: A measure based on the differences of derived quantities from projection Z_i, such as mean class energies or cumulants.

Type II: A measure based on the differences among the PDFs of Z_i.

Type III: A measure based on the differences among the cumulative distribution functions (CDFs) of Z_i.

Type IV: A measure based on the actual classification performance (e.g., a rate of correct classification) using the projection of the available training signals.

Type I measure is the one used in the original LDB method. As we will show below, this is a special case of a Type II measure, and the method adopted for the new LDB algorithm is also of Type II. This requires the estimation of the PDFs of Z_i for each class and we will describe the details below. Type III and IV measures are currently under investigation. Buckheit suggested using Type III measures since computing an empirical CDF is simpler and easier than estimating a PDF. This includes the Anderson–Darling distance which he used for measuring non-Gaussianity of projections in his paper [21]. As for Type IV measures, there are two approaches. One is to estimate the 1D PDFs, assume them as the "true" 1D PDFs, and invoke the Bayes classifier to get the misclassification rate. The other is to use any simple classifier (e.g., LDA, CT, k-nearest neighbor, etc.) or a combination of them to get the least misclassification rate. The Type IV strategy is essentially the same as the local regression basis proposed by the authors [1,2 (Chapter 5)]. We note that the misclassification rate in this context is simply used to evaluate the direction $\boldsymbol{w}_i$, and is different from the final misclassification rate obtained by the entire system d.

In the following, we focus on Type II measures and their relationship to Type I measures. If we knew the “true” conditional PDF $p(\boldsymbol{x} \mid y)$, then the PDF of Z_i for class y signals would be

$$q(z \mid y, \boldsymbol{w}_i) \triangleq \int_{\boldsymbol{w}_i \cdot \boldsymbol{x}=z} p(\boldsymbol{x} \mid y)\,\mathrm{d}\boldsymbol{x}. \tag{5}$$

For notational convenience, we write $q_i^{(y)}(z)$ instead of $q(z \mid y, \boldsymbol{w}_i)$. In practice, however, Eq. (5) cannot be computed since $p(\boldsymbol{x} \mid y)$ is not available as mentioned in the previous section. Therefore, we must estimate $q_i^{(y)}(z)$ empirically using the available training dataset. Let $\hat{q}_i^{(y)}(z)$ denote such an estimate. There are many empirical density estimation techniques as described in Ref. [10]. In fact, PDF estimation using wavelets and their relatives is a quite interesting subject itself [22]. In Section 5, we use a particular estimation method called *averaged shifted histograms* (ASH) [10, Chapter 5] which is computationally fast and has an interesting connection with the “spin cycle” algorithm of Coifman and Donoho [23] for signal denoising problems.

Having estimated PDFs, $\hat{q}_i^{(1)}, \ldots, \hat{q}_i^{(K)}$, what kind of discriminant measure $\mathscr{E}$ should we use to evaluate the direction $\boldsymbol{w}_i$? The worst direction is the one which makes all the PDFs look identical and for this direction the value of $\mathscr{E}$ should be 0. The best direction is the one which makes all the PDFs look most different or “distant” from one another which in turn should ease the subsequent classification tasks. Therefore, $\mathscr{E}$ should attain a maximum positive value in that direction. Also, we should have $\mathscr{E}(\hat{q}_i^{(1)}, \ldots, \hat{q}_i^{(K)}) \geqslant 0$ and the equality should hold if and only if $\hat{q}_i^{(1)} \equiv \cdots \equiv \hat{q}_i^{(K)}$. These conditions indicate that $\mathscr{E}$ should be a “distance-like” quantity among PDFs. There are many “distance” measures to quantify the difference or discrepancy of PDFs; see e.g., Ref. [20]. Among them, two popular distance measures (between two PDFs p and q) are:

- Relative entropy (also known as cross entropy and Kullback–Leibler divergence):
$$D(p,q) \triangleq \int p(x) \log \frac{p(x)}{q(x)}\,\mathrm{d}x. \tag{6}$$
- Hellinger distance:
$$H(p,q) \triangleq \int (\sqrt{p(x)} - \sqrt{q(x)})^2\,\mathrm{d}x. \tag{7}$$

We also note that relative entropy (6) is not a metric since it satisfies neither symmetry nor triangle inequality. For measuring discrepancies among K PDFs, $p^{(1)}, \ldots, p^{(K)}$, the simplest approach is to take $\binom{K}{2}$ pairwise combinations of $\mathscr{E}$:

$$\mathscr{E}(p^{(1)}, \ldots, p^{(K)}) \triangleq \sum_{i=1}^{K-1} \sum_{j=i+1}^{K} \mathscr{E}(p^{(i)}, p^{(j)}). \tag{8}$$

Remark 3.1. For a small number of classes, say 2–4, this approach may be sufficient. The more signal classes one has in a problem, the more obscure the meaning of Eq. (8) becomes; a large value of Eq. (8) may be due to a few significant terms with negligible majority (a favorable case) or to the accumulation of many terms with relatively small values (an unfavorable case). For such a situation, we can take a different approach suggested by Watanabe and Kaminuma [24]. Instead of constructing a single classifier for the entire problem, consider K sets of two-class problems by splitting the training dataset into class y and *non*class y. Then construct a classifier for each two-class problem. Suppose the probability of a signal $\boldsymbol{x}$ being classified to class y using the yth classifier is $p^{(y)}(y \mid \boldsymbol{x})$. Then we assign $\boldsymbol{x}$ to class k where $k = \arg\max_{y \in \mathscr{Y}} p^{(y)}(y \mid \boldsymbol{x})$. We are currently investigating this approach in the LDB setting and comparing with the one based on Eq. (8).

Now, let us consider a particular Type I measure, the normalized energy (or the normalized second moment) of signals along the direction $\boldsymbol{w}_i$. This quantity for class y signals can be written as

$$V_i^{(y)} \triangleq \frac{E[Z_i^2 \mid Y = y]}{\sum_{i=1}^{n} E[Z_i^2 \mid Y = y]}. \tag{9}$$

Therefore, once we get the estimate $\hat{q}_i^{(y)}$, we can estimate the energy as well:

$$\hat{V}_i^{(y)} = \frac{\int z^2 \hat{q}_i^{(y)}(z)\,\mathrm{d}z}{\sum_{i=1}^{n} \int z^2 \hat{q}_i^{(y)}(z)\,\mathrm{d}z}. \tag{10}$$

This shows that this Type I measure can be derived once we get an estimate of the PDFs. This is true for other derived quantities such as cumulants. Now, how are Eqs. (9) and (10) related with the normalized time–frequency energy map used in the original LDB algorithm? With the available finite samples in the training dataset, we have

$$E[Z_i^2 \mid Y = y] \approx \frac{1}{N_y} \sum_{j=1}^{N_y} |\boldsymbol{w}_i \cdot \boldsymbol{x}_j^{(y)}|^2 = \int z^2 \hat{q}_i^{(y)}(z)\,\mathrm{d}z.$$

Also, because of the orthonormality of $\{\boldsymbol{w}_i\}$,

$$\sum_{i=1}^{n} E[Z_i^2 \mid Y = y] = E[\|\boldsymbol{X}\|^2 \mid Y = y] \approx \frac{1}{N_y} \sum_{j=1}^{N_y} \|\boldsymbol{x}_j^{(y)}\|^2.$$

Thus from Eq. (4), we conclude

$$\Gamma^{(y)}(\boldsymbol{w}_i) = \hat{V}_i^{(y)}.$$

In other words, the normalized time–frequency energies used in the original LDB algorithm turn out to be finite sample versions of the Type I measure (9) in the probabilistic context. For a fixed $\boldsymbol{w}_i$, both $V_i^{(y)}$ and $\Gamma_i^{(y)}$ are point estimates, and the same is true of the discriminant measure $\mathscr{E}(\Gamma^{(1)}(\boldsymbol{w}_i), \ldots, \Gamma^{(K)}(\boldsymbol{w}_i))$. On the other hand, the discriminant measure $\mathscr{E}(\hat{q}_i^{(1)}, \ldots, \hat{q}_i^{(K)})$ makes use of the full probability distribution. Therefore, Type II measures using

PDFs are able to capture more subtle discriminant features, such as *phase* information, than can the Type I measures based on Eqs. (9) and (4).

Remark 3.2. Once it is noticed that the quantity (9) is essentially a point estimate of the nonlinear function (square function in this case) of a random variable Z_i, we may further improve the original LDB by the use of other nonlinear transformations of the coordinate values. That is, instead of using Z_i^2 in Eq. (9), we can use other nonlinear functions $\eta(Z_i)$, e.g., $\arctan Z_i$, $\log(1+|Z_i|)$, and so on. We can also use this idea for the discriminant measure $\mathscr{E}$ by estimating the PDF of $\eta(Z_i)$ instead of that of Z_i.

3.2. Discriminant power of bases in dictionaries

Instead of just a single direction, suppose we are given a basis $B=(\boldsymbol{w}_1,\ldots,\boldsymbol{w}_n)$ in a time–frequency dictionary. Because it is difficult to estimate the n-dimensional PDF directly for a large n, we evaluate each basis vector separately and sum up their discriminant powers. For notational convenience, let $\delta_i \triangleq \mathscr{E}(\hat{q}_i^{(1)},\ldots,\hat{q}_i^{(K)})$, i.e., the discriminant power of a single direction $\boldsymbol{w}_i$. Then, a simple-minded measure of the discriminant power of B may be

$$\mathscr{E}_n(B) \triangleq \sum_{i=1}^{n} \delta_i. \tag{11}$$

This coincides with the true discriminant power of the n-dimensional PDF *if all the coordinates in the basis B are statistically independent.* In general, this is not the case, i.e., there exists some dependency among the basis coordinates. Therefore, Eq. (11) is simply an approximation to the true discriminant power of the basis B.

Once we start comparing the discriminant power of various bases in the dictionary, we quickly realize the shortcoming of Eq. (11): many small δ_i's may add up to a large discriminant power for the basis, as mentioned in Remark 3.1. Therefore, we want to sum only the $k(<n)$ largest terms, i.e.

$$\mathscr{E}_k(B) \triangleq \sum_{i=1}^{k} \delta_{(i)}. \tag{12}$$

where $\{\delta_{(i)}\}$ is the decreasing rearrangement of $\{\delta_i\}$. In fact, it may be better not to take the most useless (i.e., nondiscriminant) vectors into consideration. However, automating the choice of k is not necessarily easy and needs further research.

Another possibility is to measure only the discriminant powers of the directions that carry signal energies larger than a certain threshold $t>0$

$$\mathscr{E}'_t(B) \triangleq \sum_{i=1}^{n} \varepsilon_i \delta_i, \tag{13}$$

where $\varepsilon_i = 1$ if $E[Z_i^2] > t$ and $=0$ otherwise. The selection of t should be done carefully. It should be large enough to remove all the noisy coordinates, but also should be small enough not to discard subtle discriminant features which are not noise. Unfortunately, there is no automatic procedure of threshold selection in this case unlike in simple denoising problems as described by Donoho and Johnstone [25].

Remark 3.3. It is interesting to use joint distributions of pairs of projections $(\boldsymbol{w}_i,\boldsymbol{w}_j)$ instead of 1D projections because hidden structures (such as holes) may be captured by a 2D projection but not by any 1D projection. Since it is too expensive to estimate all possible combinations, i.e., $\binom{n}{2}$ joint PDFs, for each class, we select the most discriminant $k(<n)$ coordinates in B using the 1D projections, then deal with $\binom{k}{2}$ joint PDFs. 2D joint PDFs are indispensable for feature extraction using a dictionary of local Fourier bases. There, each expansion coefficient is a complex number. Taking magnitudes of the coefficients in this case clearly ignores important *phase* information. We are currently investigating this approach.

3.3. The new LDB selection algorithm

We summarize the new procedure here with its computational cost.

Step 0: Expand each training signal $\boldsymbol{x}_j$ into a specified time–frequency dictionary $\mathscr{D}$. $[O(n\log_2 n)$ or $O(n(\log_2 n)^2)]$.

Step 1: For each vector $\boldsymbol{w}_i$ in the dictionary, estimate the PDFs $\hat{q}_i^{(y)}$ of the projection for $y=1,\ldots,K$. $[O(n(1+\log_2 n))$ using ASH estimates].

Step 2: For each vector $\boldsymbol{w}_i$ in the dictionary, compute its discriminant power δ_i. $[O(n\log_2 n)]$.

Step 3: Evaluate $\mathscr{E}(B)$ for each basis B in the dictionary and obtain the best basis via

$$\Psi = \arg\max_{B\in\mathscr{D}} \mathscr{E}(B),$$

where $\mathscr{E}$ is either $\mathscr{E}_k$ in Eq. (12) or $\mathscr{E}'_t$ in Eq. (13). $[O(n)]$.

Step 4: Select m vectors from Ψ corresponding to the m largest δ_i.

Step 5: Construct classifiers g with features derived from the m vectors.

We note that it is not necessary to have $k=m$ in general, although we set $k=m$ in the experiments in Section 5.

4. "Spin cycle": a data stabilization procedure for time–frequency dictionaries

Our bases in the dictionaries, i.e., wavelets, wavelet packets, and local trigonometric bases, are not translation invariant. If we shift the original signal, its expansion coefficients change (significantly in some cases) from those of the original signal so that one cannot tell the amount of shift from the coefficients. Depending on the application, this lack of translation invariance may be problematic. For example, for image texture classification, we are not concerned with the locations of individual edges or transients which form texture elements. Thus, it is preferable to make the analysis and classification processes more insensitive to translations in such problems. On the other hand, if the time delay of a certain waveform is an important discriminant feature in one's problem, we should not use translation invariant features.

To compensate for the lack of translation invariance, we use the so-called *spin cycle* procedure: increase the number of training signals in the training and the test datasets by creating their translated versions. More precisely, we shift each $\boldsymbol{x}_i$ in the training and the test datasets in a circular manner by $-\tau, -\tau+1, \dots, -1, 1, \dots, \tau$, where $\tau \in \mathbb{N}$ and $\tau < n$. Then we have $2\tau+1$ signals for each original signal (counting itself) all of which share the same class assignment y_i. Next, we construct the LDB using this increased training dataset, extract the top m features, and build a classifier. Then we feed all the signals to the classifier and predict the class labels. Finally, for each original signal, we take the majority vote on the $2\tau+1$ predicted class labels. This "spin cycle" procedure also plays an important role for other applications such as denoising [2 (Chapter 3), 23].

Remark 4.1. It turns out that increasing the number of sample signals by the spin cycle procedure is, in spirit, very similar to the "bagging" (*b*ootstrap *agg*regat*ing*) procedure proposed by Breiman [26]. This method tries to stabilize certain classifiers by: generating multiple versions of the training dataset by the bootstrap method [27], constructing a classifier for each training dataset, and predicting the class of test samples by majority vote on the predictions by all the classifiers.

5. Examples

In this section, we analyze two classification problems. The first one is the famous "waveform" classification problem described in Ref. [9]. The second problem is a discrimination of geophysical acoustic waveforms propagated through different media [5,6].

Example 5.1. *Triangular waveform classification*: This is a three-class classification problem using synthetic triangular waveform data. The dimensionality of the signal was extended from 21 in Ref. [9] to 32 for the dyadic dimensionality requirement of the bases under consideration. We generated 100 training signals and 1000 test signals for each class by the following formula:

$$x^{(1)}(i) = uh_1(i) + (1-u)h_2(i) + \varepsilon(i) \quad \text{for Class 1,}$$

$$x^{(2)}(i) = uh_1(i) + (1-u)h_3(i) + \varepsilon(i) \quad \text{for Class 2,}$$

$$x^{(3)}(i) = uh_2(i) + (1-u)h_3(i) + \varepsilon(i) \quad \text{for Class 3,}$$

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 $\varepsilon(i)$ are standard normal variates. Fig. 1 shows five sample waveforms from each class.

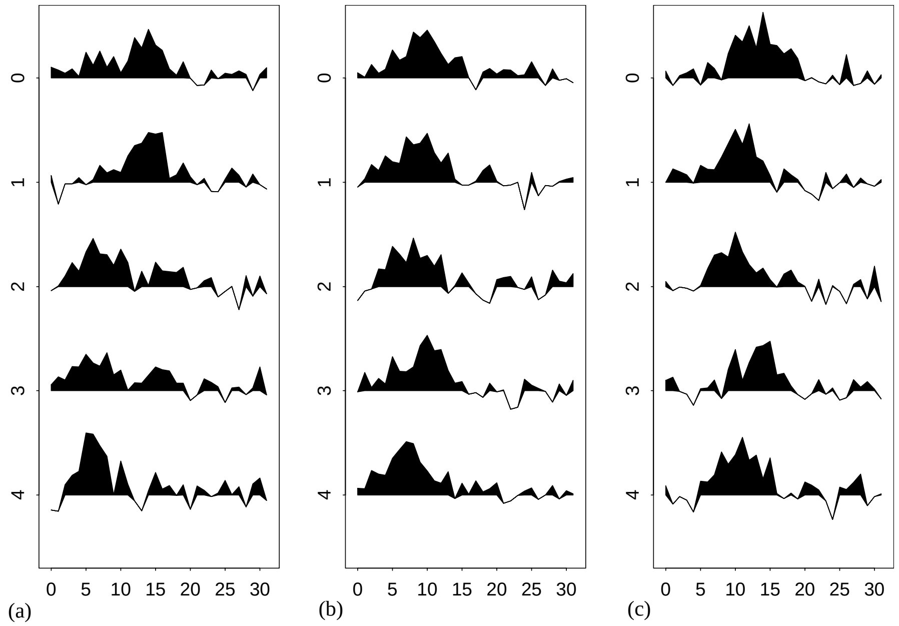

Fig. 1. Five sample waveforms from: (a) Class 1, (b) Class 2, and (c) Class3.

We conducted three sets of different classification experiments.

1. The original LDB method with and without Spin Cycle.
2. The new LDB method with and without Spin Cycle using the simple histograms as the PDF estimation method.
3. The new LDB method with and without Spin Cycle using ASH as the PDF estimation method.

In each experiment, we repeated the whole process 10 times by generating 10 different realizations of the training and test datasets. As a classifier g in Eq. (2), we used LDA and a CT. As a dictionary for LDB, we used the wavelet packet dictionary with the 6-tap Coiflet filter [18, Appendix C]. For the discriminant measure, we adopted the relative entropy (8) for three classes. As for m, the number of most important features to be fed to the classifiers, we set $m = 5$ by the heuristics that $m \approx 0.1n$ or $0.2n$. For comparison, we also directly applied LDA and CT to the signals represented in the standard (or canonical) basis of $\mathbb{R}^{32}$.

The averaged misclassification rates are summarized in Tables 1–3.

We would like to note that according to Breiman et al. [9], the Bayes error of this example is about 14%. From these tables, we observe:

- The best result so far was obtained by applying LDA to NLDB5SC3, i.e., the top 5 LDB coordinates with ASH estimates on the spin cycled data (with three shifts).
- Four of the top 5 original and new LDB vectors are the same except for their order of importance.
- Due to the nature of the problem (three triangles are located at fixed positions), Spin Cycle with too many shifts (five shifts here) degraded the performance.
- Overall the LDA gave much better results than CT.
- For the new LDB algorithm, the spin cycle is critical. Without spin cycle, it is worse than the original LDB.
- Overall ASH-based methods gave better results than the simple histogram-based methods.

Note that the best results NLDB5SC3 essentially used a *double* stabilization procedure: spin cycle on the original signals and ASH on the expansion coefficients. In Fig. 2, we compare the top 5 discriminant vectors from OLDB5 and NLDB5SC3. The fifth vectors differ only by a translation, and the others are the same excepting their order.

Table 1
The average misclassification rates for the waveform classification example over 10 simulations.

Method (coordinates)	Training (%)	Test (%)
LDA on STD	12.0	22.7
LDA on OLDB5	14.1	16.2
LDA on OLDB5SC3	14.1	16.1
LDA on OLDB5SC5	16.2	17.4
CT on STD	7.0	29.3
CT on OLDB5	8.1	21.9
CT on OLDB5SC3	5.8	21.4
CT on OLDB5SC5	7.7	22.5

Note: In theMethod column, STD and OLDB5 represent the standard coordinatesand the top 5 coordinates of the original LDB (based on thetime–frequency energy distributions), respectively. SC3 and SC5refer to the spin cycle with 3 and 5 shifts ($\tau = 1, 2$)

Table 2
The average misclassification rates for the new LDB algorithm using simple histograms as the empirical PDF estimation method (averaged over 10 simulations)

Method (coordinates)	Training (%)	Test (%)
LDA on NLDB5	17.0	19.6
LDA on NLDB5SC3	16.2	18.2
LDA on NLDB5SC5	17.5	18.8
CT on NLDB5	9.9	26.3
CT on NLDB5SC3	6.3	24.5
CT on NLDB5SC5	7.9	23.7

Note: Here, NLDB5xxx means that $m = k = 5$ in Eq. (12).

Table 3
The average misclassification rates with the new LDB algorithm using ASH as the empirical PDF estimation method (averaged over 10 simulations)

Method (coordinates)	Training (%)	Test (%)
LDA on NLDB5	16.0	18.2
CT on NLDB5	8.8	24.5
LDA on NLDB5SC3	14.4	**15.9**
CT on NLDB5SC3	5.5	20.8
LDA on NLDB5SC5	15.9	17.9
CT on NLDB5SC5	7.5	22.8

Example 5.2. *Discrimination of geophysical acoustic waveforms*: For the detailed background of this problem, see Refs. [5,6]. Here, we want to discriminate the waveforms (recorded in a borehole with 256 time samples per waveform) propagated through sandstone layers in the subsurface from the ones through shale layers. We have 201 such "sand waveforms" and 201 "shale waveforms." We used a 10-fold cross-validation procedure to compute the misclassification rates. We used the local sine dictionary which is easier in dealing with time information than the wavelet packet dictionaries. We used relative entropy (6) as a discriminant measure and ASH as the PDF estimator again. In this experiment, we examined the dependence of classification performance on the number of important features m to compare with the results obtained in Ref. [6]. The results for $m = k = 5, \ldots, 100$ in steps of five are summarized in Figs. 3 and 4.

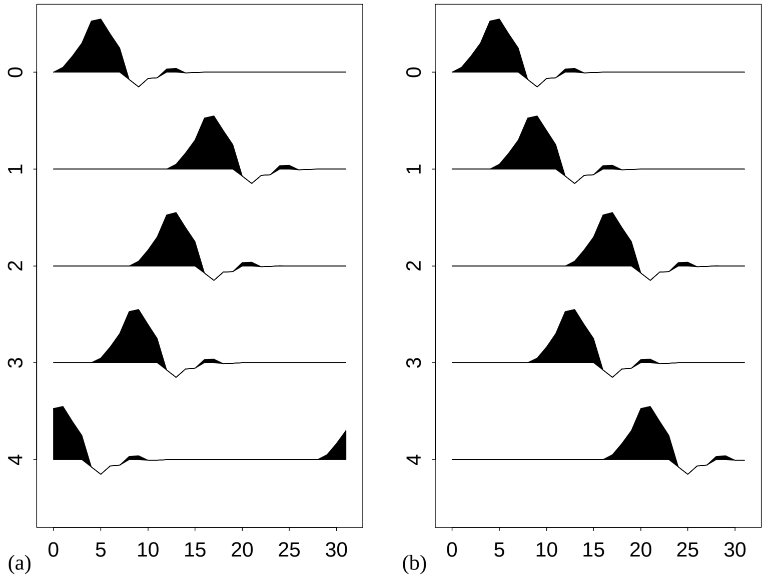

Fig. 2. Top five LDB vectors of the waveform classification example: (a) The original LDB vectors, (b) The new LDB vectors.

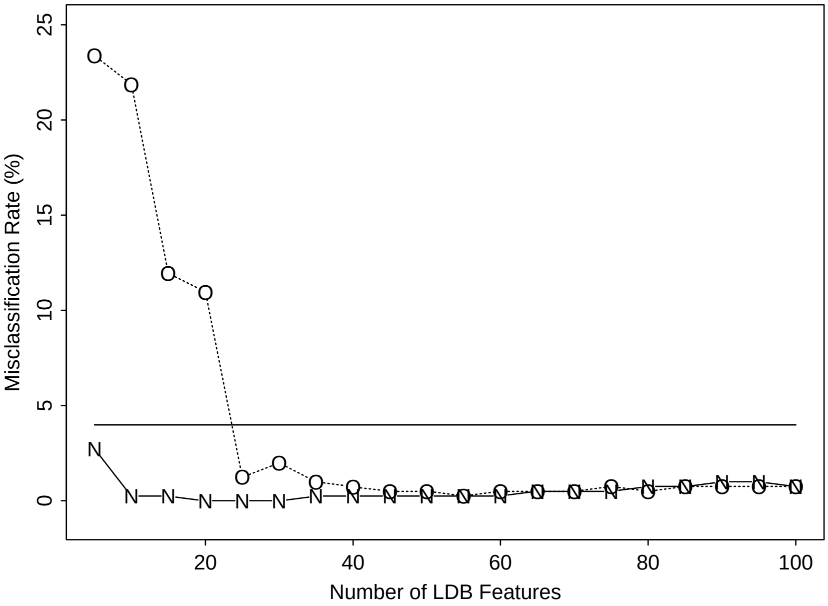

Fig. 3. Misclassification rates using LDA as a classifier versus the number of top LDB features retained. The plots with symbols O and N correspond to the results using the original and the new LDB algorithms, respectively. The constant level line about 4% indicates the performance of the LDA directly applied to the signals represented in the standard coordinate system (of 256 time samples).

From these plots, we observe that

- No misclassification occurs with LDA on the top 20, 25, and 30 new LDB vectors.
- These good features are mainly concentrated in P wave components; see also Refs. [5,6].
- Using LDA with < 40 features, the new LDB outperforms the original LDB. The difference is small for > 45 features.
- Using CT, the original LDB performs better than the new LDB, but the result on the standard basis is even better.

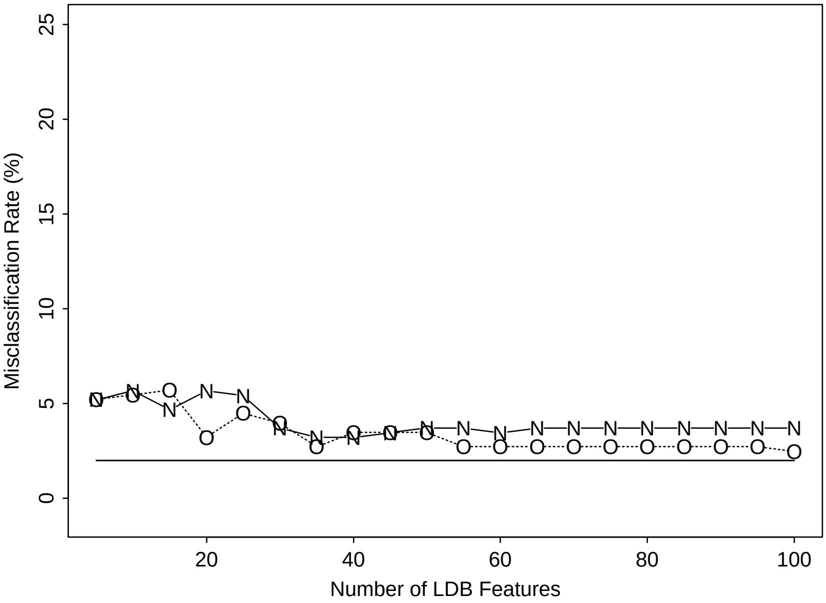

Fig. 4. Misclassification rates using CT as a classifier versus the number of top LDB features retained. The constant level line about 2% indicates the performance of the CT directly applied to the signals represented in the standard coordinate system.

6. Conclusion

We described a new LDB algorithm using "distances" among the estimated PDFs of projections of input signals onto basis vectors in time–frequency dictionaries. Using a probabilistic setting for the new LDB method, the meaning of the original LDB method, which is based on the time–frequency energy distributions of the projections, was clarified. The features derived from the new LDB vectors can be more sensitive to phase shifts than the original LDB vectors. For the examples we showed, the new LDB method performed better than the original one. We are currently investigating the new LDB method for complex-valued features derived from the local Fourier dictionary, where the new method may have a significant advantage over the original one. However, we would like to emphasize that the new algorithm should be considered as an option, not as a better method in all cases than the original one. Depending on the problem, the original LDB method may give sufficient or even better results. In general, one should try both the original and the new LDB methods for any problem at hand.

Acknowledgements

An earlier version of this paper was presented at the 1996 Annual Meeting of the American Statistical Association in Chicago, IL.

The authors would like to thank Dr. Jon Buckheit, formerly at Stanford University, for helpful discussions. This research was partially supported by NSF DMS-9973032, DMS-9978321, ONR YIP N00014-00-1-0469, and Schlumberger-Doll Research.

References

[1] R.R. Coifman, N. Saito, Constructions of local orthonormal bases for classification and regression, Comptes Rendus Acad. Sci. Paris, Série I 319 (1994) 191–196.

[2] N. Saito, Local feature extraction and its applications using a library of bases. Ph.D. Thesis, Department of Mathematics, Yale University, New Haven, CT 06520, USA, December 1994, Available via World Wide Web, http://www.math.yale.edu/pub/wavelets/papers/lfeulb.tar.gz.

[3] N. Saito, R.R. Coifman, Local discriminant bases, Proc. SPIE 2303 (1994) 2–14.

[4] N. Saito, R.R. Coifman, Local discriminant bases and their applications, J. Math. Imaging Vision 5(4) (1995) 337–358. (Invited paper).

[5] N. Saito, R.R. Coifman, Extraction of geological information from acoustic well-logging waveforms using time–frequency wavelets, Geophysics 62 (6) (1997) 1921–1930.

[6] N. Saito, Classification of geophysical acoustic waveforms using time–frequency atoms, in: American Statistical Association 1996 Proceedings of the Statistical Computing Section, pp. 322–327.

[7] R.J.M. Guglielmi, Wavelet feature definition and extraction for classification and image processing, Ph.D. Thesis, Yale University, 1997.

[8] F. Warner, personal communication, 1996.

[9] L. Breiman, J.H. Friedman, R.A. Olshen, C.J. Stone, Classification and Regression Trees, Chapman & Hall, New York, 1993 (previously published by Wadsworth & Brooks/Cole in 1984).

[10] D.W. Scott, Multivariate Density Estimation: Theory, Practice, and Visualization, Wiley, New York, 1992.

[11] T. Kohonen, The self-organizing map, Proc IEEE 78(9) (1990) 1464–1480 (Invited paper).

[12] J.B. Kruskal, Toward a practical method which helps uncover the structure of a set of multivariate observations by finding the linear transformation which optimizes a new 'index of condensation', in: R.C. Milton, J.A. Nelder (Eds.), Statistical Computation, Academic Press, New York, 1969, pp. 427–440.

[13] J.H. Friedman, J.W. Tukey, A projection pursuit algorithm for exploratory data analysis, IEEE Trans. Comput. 23 (1974) 881–890.

[14] P.J. Huber, Projection pursuit (with discussion), Ann. Statist. 13 (2) (1985) 435–525.

[15] M.C. Jones, R. Sibson, What is projection pursuit? (with discussion), J. Roy. Statist. Soc. A 150 (Part 1) (1987) 1–36.

[16] T.M. Cover, The best two independent measurements are not the two best, IEEE Trans. Systems. Man Cybernet. SMC-4 (1974) 116–117.

[17] Y. Meyer, Wavelets: Algorithms and Applications. SIAM, Philadelphia, PA, 1993 (translated and revised by R.D. Ryan).

[18] M.V. Wickerhauser, Adapted Wavelet Analysis from Theory to Software, A K Peters, Wellesley, MA, 1994.

[19] S. Mallat, A Wavelet Tour of Signal Processing, 2nd Edition, Academic Press, San Diego, CA, 1999.

[20] M. Basseville, Distance measures for signal processing and pattern recognition, Signal Process. 18 (4) (1989) 349–369.

[21] J.B. Buckheit, D.L. Donoho, Time–frequency tilings which best expose the non-Gaussian behavior of a stochastic process, in: Proceedings of the International Symposium on Time-Frequency and Time-Scale Analysis, IEEE, Paris, France, June 18–21, 1996, pp. 1–4.

[22] D.L. Donoho, I.M. Johnstone, G. Kerkyacharian, D. Picard, Density estimation by wavelet thresholding, Ann. Statist. 24 (2) (1996) 508–539.

[23] R.R. Coifman, D. Donoho, Translation-invariant de-noising, in: A. Antoniadis, G. Oppenheim (Eds.), Wavelets and Statistics, Lecture Notes in Statistics, Springer, New York, 1995, pp. 125–150.

[24] S. Watanabe, T. Kaminuma, Recent developments of the minimum entropy algorithm, in: Proceedings of the International Conference on Pattern Recognition, IEEE, New York, 1988, pp. 536–540.

[25] D.L. Donoho, I.M. Johnstone, Ideal denoising in an orthonormal basis chosen from a library of bases, Comptes Rendus Acad. Sci. Paris, Sér. I 319 (1994) 1317–1322.

[26] L. Breiman, Bagging predictors, Machine Learning 24 (1996) 123–140.

[27] B. Efron, R.J. Tibshirani, An Introduction to the Bootstrap, Chapman & Hall, New York, 1993

About the Author—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 worked as a research scientist until 1997. In 1997, he joined the Department of Mathematics at the University of California, Davis, where he is currently a professor.

Dr. Saito received the Best Paper Award from SPIE for the wavelet applications in signal and image processing conference in 1994, and the Henri Doll Award, the highest honor for technical papers presented at the annual symposium within the Schlumberger organization in 1997. He also received a Young Investigator Award from the Office of Naval Research, and a Presidential Early Career Award for Scientists and Engineers (PECASE) from the White House, both in 2000.

His research interests include: wavelet theory and its applications, feature extraction, pattern recognition and classification, statistical signal processing and analysis, human and machine vision, and geophysical inverse problems.

He is a member of IEEE, IMS, SEG, and SIAM.

About the Author—RONALD R. COIFMAN is Phillips 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.

Prof. Coifman's recent publications have been in the areas of nonlinear Fourier analysis, wavelet theory, numerical analysis and scattering theory.

Professor Coifman is currently leading a research program to develop new mathematical tools for efficient transcription of physical data, with applications to feature extraction, recognition, and denoising.

He was chairman of the Yale mathematics department from 1986 to 1989. He is a member of the National Academy of Science, American Academy of Arts and Sciences, and the Connecticut Academy of Sciences and Engineering.

He received the DARPA Sustained Excellence Award in 1996, the 1996 Connecticut Science Medal, the 1999 Pioneer Award from the International Society for Industrial and Applied Mathematics, and the 1999 National Medal of Science. He founded Fast Mathematical Algorithms and Hardware (FMA& H), a technology transfer company specializing in the conversion of theoretical mathematical tools into engineering software for fast computation and processing.

About the Author—FRANK B. GESHWIND received a B.A. in Mathematics, summa cum laude, from CUNY Queens College in 1988, and a Ph.D. in Mathematics from the CUNY Graduate School in 1993. He has been a postdoctoral associate at the Yale University Mathematics Department since 1993. In 1994 he joined FMA& H, where he has developed fast wavelet image and video compression and other software and algorithms.

Dr. Geshwind's research interests include: applied and computational harmonic analysis, noiselets, feature extraction, classification, anomaly detection, self organizing analysis, data transcription and compression, and algorithm and implementation issues.

In 2000, Dr. Geshwind co-founded and became the CTO of Streaming Senses Technology, a company that develops streaming media technology for internet and wireless networks.

About the Author—FRED WARNER received a B.A. in Mathematics from the University of Virginia in 1985, and the Ph.D. degree in Mathematics from Yale University in 1992. He was a postdoctoral associate at the Graduate Center of the City University of New York from 1992 to 1995. From 1995 to 1997, he was jointly affiliated with the Graduate Center of CUNY and with Digital Diagnostic Corp. in Hamden, CT as an NSF University-Industry Postdoctoral Fellow. Since 1997, he has been jointly with the Department of Mathematics at Yale and with FMA& H, working on a variety of problems in pattern recognition and signal and image processing. His current interests include feature extraction for regression and classification, clustering methods, and the geometry and analysis of high-dimensional data sets.