Least statistically-dependent basis and its application to image modeling

Naoki Saito

Department of Mathematics, University of California, Davis, CA 95616 USA

ABSTRACT

Statistical independence is one of the most desirable properties for a coordinate system for representing and modeling images. In reality, however, truly independent coordinates may not exist for a given set of images, or it may be computationally too difficult to obtain such coordinates. Therefore, it makes sense to obtain the least statistically dependent coordinate system efficiently. This basis—we call it *Least Statistically-Dependent Basis* (LSDB)—can be rapidly computed by minimizing the sum of the differential entropy of each coordinate in the basis library. This criterion is quite different from the Joint Best Basis (JBB) proposed by Wickerhauser. We demonstrate the use of the LSDB for image modeling and compare its performance with JBB and Karhunen-Loève Basis (KLB).

Keywords: Statistical independence, principal component analysis, independent component analysis, best basis, dimension reduction, image modeling

1. INTRODUCTION

Suppose we are given a set of similar images such as human faces (or a set of finger prints or a set of mammograms). And suppose we want to *learn* the characteristics of those images, i.e., to represent them efficiently, analyze certain features, and build a probabilistic model that can generate new images that are similar to those given images. What should we do, then? The best possible scenario would be to find a *statistically independent* coordinate system (basis) of that class of images. With this coordinate system we could achieve optimal compression of the images in that class by transmitting each coordinate (feature) separately using quantization scheme depending on the marginal distribution of each coordinate. Moreover, a complete probabilistic description of an image class would be made possible by simply characterizing the probability distributions of each coordinate. We could *sample* as many new images from this probability model to generate "typical" images in this class and to examine how they look like. This would be a great tool for image diagnostics; one typically wants to know how the typical images that got the same diagnostics look like. In reality, however, it may not be possible to obtain truly independent coordinates because 1) the data may not be composed of truly independent features in the first place, and 2) even if the images consist of independent features, it may be too difficult to construct a feasible algorithm to extract such features because of the high dimensionality of the problem (imagine a database consisting of 512 by 512 pixel images).

Let us briefly trace a history in the signal and image processing field to see how people attacked these problems. The importance of the independent coordinates has long been recognized by several researchers in this field. In the seminal paper by Satosi Watanabe[1] about the Karhunen-Loève (KL) expansion—also known as Principal Component Analysis (PCA)—and its application to pattern recognition, he argued the justification of the use of the KL coordinates for "feature compression" as follows:

> It would be desirable, from the viewpoint that information compression means elimination of redundancy, to use variables which are statistically independent, but in the absence of such variables, statistically uncorrelated variables may be the next best.

Then, he went on to characterize the KL basis (KLB) is the minimum entropy basis among all the orthonormal bases in $\mathbb{R}^n$, where n is a number of pixels in images under consideration. This was a great achievement around 1965, and in fact, KLB was probably the best available feature extraction tool around that time. However, KLB-PCA only provides us with the *decorrelated* coordinates, and only takes care of the second order statistics. Of course, if the underlying data obeys the multivariate Gaussian distribution, decorrelation implies independence. But in general, the natural images such as faces are far from Gaussian (see e.g.,[2]). Moreover, KLB-PCA has other drawbacks such as

Further author information: E-mail: saito@math.ucdavis.edu; WWW: http://math.ucdavis.edu/∼saito

high computational cost and inaccuracy of sample estimate of covariance matrices which will be described in detail in Section 2.

More recently, the concept called Independent Component Analysis (ICA) has become popular, in particular, in the field of signal processing[3] and computational neuroscience.[4] ICA considers higher order statistics than KLB-PCA. The spirit of ICA is great; it tries to obtain the statistically independent coordinate system more directly than KLB-PCA. It is very difficult, however, to compute it numerically, in particular for high dimensional data, since they rely on the higher order cumulants.

Thirty years since Watanabe's work has changed the landscape. We have now a *library* of local bases, which consists of various *dictionaries* of bases such as wavelet packet bases and local Fourier bases, at our disposal as feature extraction tools. These are adaptable and flexible set of bases that can be tailored to one's needs very efficiently. They have been increasingly popular in various feature extraction business such as denoising,[5–7] classification and regression.[8–10] The author and his colleagues, in particular, R. R. Coifman and M. V. Wickerhauser, have been advocating the use of the so-called "best basis paradigm" consisting of the following three steps: 1) Select a best possible basis from a dictionary or library of bases by optimizing a certain functional that quickly evaluates the efficacy of each basis in the dictionary/library for the problem at hand, 2) Discard the unimportant coordinates from the selected basis, and 3) Use the survived coordinates for the problem. Depending on the problem at hand, we need to use a different efficacy measure for the basis evaluation, and it is of critical importance to choose an appropriate measure.

Wickerhauser proposed the so-called "joint best basis" (JBB) which tried to alleviate some of the drawbacks of the KLB-PCA.[11] Independently from Watanabe, he proposed to find a basis from a dictionary that minimizes entropy of the energy distribution over its coordinates. Watanabe's argument is that KLB is a best basis over all possible orthonormal bases of $\mathbb{R}^n$ with respect to the minimum entropy criterion whereas Wickerhauser's algorithm can quickly compute an approximate KLB that is a best basis over all possible bases in the dictionary or library of orthonormal bases. Thus, JBB corresponds to KLB-PCA, but not to ICA. It does not address the statistical independence of the coordinates explicitly.

In this paper, we propose yet another best basis aiming more directly to the statistical independence than KLB or JBB. Since there is no guarantee that the images under consideration consist of truly independent coordinates, a compromised but efficient strategy is to extract the *least statistically dependent basis* (LSDB) from a dictionary or library of bases quickly.

This paper is organized as follows. In Section 2, we set up our notation and briefly review the KLB-PCA, ICA, and JBB using our notation. In Section 3, we consider a measure of statistical dependence of a given basis and propose the LSDB algorithm. In Section 4, we apply LSDB to signal and image modeling and compare this with KLB and JBB. We end this paper with discussion of the relation of the LSDB to the other methods and describe some of our ongoing and future work in Section 5.

2. FEATURE EXTRACTION AND BASIS SEARCH

Let $\mathcal{X}$ be a set of all input images of a particular class under consideration. We call $\mathcal{X} \in \mathbb{R}^n$ an input image space, where n is a number of pixels in each image. For most of the images we are interested in, the dimension n is rather large. For example, in medical X-ray tomography, we typically have $n \approx 512^2$. Suppose we are given N training (sample) images. $\mathcal{T} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\} \subset \mathcal{X}$. Let us assume that these images are N realizations of some stochastic process, and let $\boldsymbol{X} \in \mathcal{X}$ be a random sample from this process and let $p_{\boldsymbol{X}}(\boldsymbol{x})$ be its probability density function (pdf). The ultimate characterization of a given class of images entails estimating $p_{\boldsymbol{X}}$ from the available training dataset. Estimating empirical pdf from available samples, however, is very difficult because of the high dimensionality of the input space $\mathcal{X}$ (*curse of dimensionality*); we need a huge number of training samples to get a reliable estimate of $p_{\boldsymbol{X}}$, which we normally cannot access. In a typical situation in learning a class of images, the dimension of the problem is much larger than the available number of samples, i.e., $n \gg N$. Therefore, we need to reduce the dimensionality of the problem without losing important information for characterization.

As Scott mentions in his book,[12] this strategy is also supported by the empirical observation that multivariate data in $\mathbb{R}^n$ are almost never n-dimensional and there often exist lower dimensional structures of data. In other words, a class of images often has an *intrinsic dimension* $m < n$ (often $m \ll n$). Therefore, it would be much more efficient and effective to analyze the data in the smaller dimensional subspace $\mathcal{F}$ of $\mathcal{X}$, if possible. We call $\mathcal{F}$ a *feature space*,

and a map $f : \mathcal{X} \to \mathcal{F}$ a *feature extractor*. Then, the key is how to construct this "good" feature space $\mathcal{F}$ consisting of important features and to design the corresponding feature extractor f.

In this paper, we restrict our attention to the specific type of feature extractors. A feature extractor here consists of a change of the coordinates (basis) in $\mathcal{X}$ followed by a selection of m coordinates. Let B is any basis spanning $\mathcal{X} \subset \mathbb{R}^n$. We also view B as a matrix whose columns are the basis vectors in $\mathcal{X}$. Let $\mathcal{E}(B \,|\, \mathcal{T})$ be a certain functional measuring the efficacy of the basis B for our problem at hand given a training dataset $\mathcal{T}$. Then, we seek the best coordinates B_*

$$B_* = \arg\max_{B \in \mathcal{L}} \mathcal{E}(B \,|\, \mathcal{T}), \tag{1}$$

where $\mathcal{L}$ is a set of all possible bases under consideration. Whether we constrain our search by restricting $\mathcal{L}$ or not makes a big difference as we will see soon. This basis selection process is followed by the selection of the basis vectors spanning the feature space.

$$\mathcal{F} = \mathrm{span}\{\boldsymbol{w}_{j_1}, \boldsymbol{w}_{j_2}, \ldots, \boldsymbol{w}_{j_m}\}, \tag{2}$$

where $\boldsymbol{w}_{j_k} \in B_*$, a subset of m basis vectors selected from B_*.

Now we discuss the previously proposed feature extraction techniques under this unifying perspective of the basis selection strategy.

2.1. Karhunen-Loève Basis–Principal Component Analysis

As explained briefly in Introduction, KLB-PCA provides us with a decorrelated coordinate system. The KLB vectors are the eigenvectors of the autocorrelation (or covariance) matrix of the process obeying $p_{\boldsymbol{X}}$. KLB satisfies a number of optimality criteria, and in particular, it is *the minimum entropy basis* among all the orthonormal bases $\mathrm{O}(n)$, i.e., all the rotations of the coordinates in $\mathbb{R}^n$.[1] Let $B = (\boldsymbol{w}_1, \ldots, \boldsymbol{w}_n) \in \mathrm{O}(n)$ is any orthonormal basis in $\mathbb{R}^n$, and let $Z_i = \boldsymbol{w}_i^T \boldsymbol{X}$, the ith coordinate of the image $\boldsymbol{X}$ relative to the basis B. Entropy of the energy distribution over the coordinate axes can be considered as inefficiency of the coordinate systems for a class of images (i.e., the larger the entropy, the less efficient for feature compression). Let us now define the *entropy function* as

$$h(\boldsymbol{\gamma}[B]) \stackrel{\Delta}{=} -\sum_{i=1}^{n} \gamma_i[B] \log \gamma_i[B], \tag{3}$$

where $\gamma_i[B]$ is a normalized energy (or variance) of the ith coordinate of B, i.e., $\gamma_i[B] = E[Z_i^2] / \sum_{j=1}^n E[Z_j^2]$, or $\gamma_i[B] = \mathrm{Var}[Z_i] / \sum_{j=1}^n \mathrm{Var}[Z_j]$. In practice, we need to use the sample estimates $\widehat{\gamma}_i[B]$ of $\gamma_i[B]$ using the training set $\mathcal{T}$. Then, KLB is characterized by (1) with the following specific efficiency criterion.

$$B_{KLB} = \arg\max_{B \in \mathrm{O}(n)} \mathcal{E}_{KLB}(B \,|\, \mathcal{T}) = \arg\max_{B \in \mathrm{O}(n)} -h(\widehat{\boldsymbol{\gamma}}[B]) = \arg\min_{B \in \mathrm{O}(n)} h(\widehat{\boldsymbol{\gamma}}[B]). \tag{4}$$

On the other hand, KLB has several drawbacks. First of all, the criterion (4) does not measure the statistical independence of the coordinates. KLB only provides us with the decorrelated coordinates, i.e., "the next best" coordinates, as Watanabe put it. Therefore, KLB is only optimal for the multivariate Gaussian data. The next serious problem is an inaccuracy of the sample estimate of the autocorrelation or covariance matrices of the underlying process $p_{\boldsymbol{X}}$. In general, we do not know these matrices a priori, therefore, we need to estimate them using the available training samples. This inaccuracy is particularly severe for large n (dimension of the problem) with small N (the number of training samples). This entangles with the computational complexity as follows. Suppose the singular value decomposition of the data matrix $X = (\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N) \in \mathbb{R}^{n \times N}$ is $X = U \Sigma V^T$. (There is no need to perform full SVD in practice. This is just for an explanation of the KLB computation.) Since the rank of X is $\min(n, N)$, if $n < N$ (this is a classical situation in statistics where the dimensionality is small and the large number of samples are available), $B_{KLB} = U \in \mathrm{O}(n)$ and its computational cost is $O(n^3)$ for solving the eigenvalue problem, $XX^T U = U \Sigma \Sigma^T$. Now if $n > N$ (most of our problems of interest are under this category), the column vectors of XV are the first N eigenvectors of the sample autocorrelation matrix XX^T because $XX^T XV = XV\Sigma^T\Sigma$. We then need to solve the eigenvalue problem $X^T XV = V\Sigma^T\Sigma$ which is simply an $N \times N$ problem, i.e., requires $O(N^3)$. In summary, the KLB computation costs $O(\min(n, N)^3)$. Note that having a small N is advantageous only

for computational speed, not for the statistical accuracy as mentioned above. the resulting vectors are essentially useless; they do not truly capture the statistical structure of the distribution $p_{\boldsymbol{X}}$. On the other hand, if N increases, then the computational cost increases cubically. This is a dilemma of the KLB computation.

We note that the recent work of Donoho, Mallat, and von Sachs tries to address this estimation of covariance matrices with a few number of samples.[13]

2.2. Independent Component Analysis

To overcome the limitation of the PCA to the second order statistics, Comon[3] and the others proposed the so-called Independent Component Analysis (ICA). Bell and Sejnowski discussed the closely related concept of "information maximization" and its neural network implementation.[4]

Given a training dataset $\mathcal{T}$, ICA tries to find a linear transformation (a full rank $m \times n$ matrix) that minimizes the statistical dependence among its coordinates. In our notation, ICA integrates (1) with (2) and can be written as

$$B_{ICA} = \arg \max_{B \in \mathbb{R}^{n \times m}} \mathcal{E}_{ICA}(B \,|\, \mathcal{T}),$$

where $\mathcal{E}_{ICA}(B \,|\, \mathcal{T})$ measures the degree of statistical independence of the coordinate system B for the training dataset $\mathcal{T}$. Let us now define differential entropy $H(p_{\boldsymbol{X}})$ of the process obeying $p_{\boldsymbol{X}}$.

$$H(p_{\boldsymbol{X}}) \stackrel{\Delta}{=} - \int p_{\boldsymbol{X}}(\boldsymbol{x}) \, \log p_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}. \tag{5}$$

A convenient measure to quantify the statistical dependency of the coordinates of $\boldsymbol{X}$ is the so-called *mutual information*:

$$\begin{aligned} I(p_{\boldsymbol{X}}) &= \int p_{\boldsymbol{X}}(x_1, \ldots, x_n) \log \frac{p_{\boldsymbol{X}}(x_1, \ldots, x_n)}{\prod_{i=1}^{n} p_{X_i}(x_i)} \mathrm{d}x_1 \cdots \mathrm{d}x_n & (6) \\ &= -H(p_{\boldsymbol{X}}) + \sum_{i=1}^{n} H(p_{X_i}), & (7) \end{aligned}$$

which is simply relative entropy between $p_{\boldsymbol{X}}$ and the product of the marginals $\{p_{X_i}\}$. We note that $I(p_{\boldsymbol{X}}) = 0$ if and only if the components of $\boldsymbol{X}$ are mutually independent. Now, we can write the efficiency of the coordinate system B as

$$\mathcal{E}_{ICA}(B \,|\, \mathcal{T}) = -I(\widehat{p}_{\boldsymbol{Z}}) = H(\widehat{p}_{\boldsymbol{Z}}) - \sum_{i=1}^{m} H(\widehat{p}_{Z_i}), \tag{8}$$

where $\boldsymbol{Z} \in \mathbb{R}^m$ is the process represented in the new coordinates B, and $\widehat{p}_{\boldsymbol{Z}}$ and $\widehat{p}_{Z_i}$ are the empirical pdf's (epdf's), i.e., sample estimates of the corresponding pdf's using the data $\mathcal{T}$. As mentioned earlier, it is extremely difficult to have a good estimate $\widehat{p}_{\boldsymbol{X}}$ (or $\widehat{p}_{\boldsymbol{Z}}$) for large n, and even the case with $n > 3$ is difficult in practice. Therefore, Comon proposed to approximate (8) using the Edgeworth expansion of $p_{\boldsymbol{Z}}$ around the multivariate normal distribution with the same mean and variance as the original process, and this amounts to using the higher order cumulants of $\boldsymbol{Z}$. This computational procedure is even more complicated and expensive than KLB; it costs $O(n^{2.5}N)$. In other words, the ICA of Comon is not feasible for the problems with very high dimensions, $n \gg N$.

2.3. Joint Best Basis

In the meantime, Wickerhauser proposed the JBB that is the minimum entropy basis among all the bases in the specified dictionary of orthonormal bases.[11] The JBB criterion is simply written as:

$$B_{JBB} = \arg \max_{B \in \mathcal{D}} \mathcal{E}_{KLB}(B \,|\, \mathcal{T}) = \arg \min_{B \in \mathcal{D}} h(\widehat{\boldsymbol{\gamma}}[B]). \tag{9}$$

A key difference from (4) is that B is searched within a specified dictionary of orthonormal bases $\mathcal{D}$ instead of all possible rotations $\mathrm{O}(n)$. Therefore, its computational complexity is reduced to $O(n[\log n]^p)$, where $p = 1$ for wavelet packet dictionaries, and or $p = 2$ for local Fourier dictionaries. Recall that a dictionary $\mathcal{D}$ contains more than 2^n different orthonormal bases.[14] Moreover, since each feature is localized both in the space and spatial frequency domains, analysis and interpretation of the images become easier and more intuitive.

3. LEAST STATISTICALLY-DEPENDENT BASIS

Faced with the difficulty of ICA, it makes sense to find a basis from a dictionary or library of orthonormal bases that minimizes the statistical dependency among the coordinates. To do this, let us consider a change of the basis of $\boldsymbol{X}$ in the definition of the differential entropy (5). We can easily get

$$H(p_{\boldsymbol{Z}}) = H(p_{B^T\boldsymbol{X}}) = H(p_{\boldsymbol{X}}) + \log|\det(B)|. \tag{10}$$

Therefore, if B is a volume-preserving linear transformation, or more specifically, an orthonormal basis, then the differential entropy $H(p_{\boldsymbol{X}})$ is *invariant* under such transformations.

$$H(p_{\boldsymbol{Z}}) = H(p_{B^T\boldsymbol{X}}) = H(p_{\boldsymbol{X}}). \tag{11}$$

This invariance property is the key for our LSDB algorithm. As long as we deal with orthonormal bases, we do not need to compute or estimate $H(\widehat{p}_{\boldsymbol{Z}})$ in (8). The efficacy—degree of the statistical independence among the coordinates—of an orthonormal basis can be quantified by only considering the second term in (8) i.e., the sum of the differential negentropy of the individual coordinates. Now, we can state the selection criterion of our *Least Statistically-Dependent Basis* (LSDB):

$$B_{LSDB} = \arg\max_{B\in\mathcal{D}} \mathcal{E}_{LSDB}(B\,|\,\mathcal{T}) = \arg\max_{B\in\mathcal{D}} \left(-\sum_{i=1}^{n} H(\widehat{p}_{Z_i})\right) = \arg\min_{B\in\mathcal{D}} \sum_{i=1}^{n} H(\widehat{p}_{Z_i}). \tag{12}$$

LSDB is thus obtained by minimizing the sum of the coordinate-wise differential entropy among all possible orthonormal bases in a specified dictionary of orthonormal bases $\mathcal{D}$. We note that the basis search in (12) is fast since the sum of the coordinate-wise differential entropy is an additive measure.[14] So, as long as we use relatively inexpensive pdf estimators such as ASH,[12] the computational complexity of the entire algorithm is dominated by the cost of expanding input images in a dictionary of bases, which costs $O(n[\log n]^p)$.

Remark 3.1. We can contrast our LSDB with KLB and JBB now. If we use the population version of (12), then we have

$$\sum_{i=1}^{n} H(p_{Z_i}) = \sum_{i=1}^{n} E\left[\log\frac{1}{p_{Z_i}}\right].$$

On the other hand, for KLB and JBB assuming that $\sum_{j=1}^{n} E[Z_j^2] = 1$, we have

$$\sum_{i=1}^{n} h(E[Z_i^2]) \geq \sum_{i=1}^{n} E[h(Z_i^2)] = \sum_{i=1}^{n} E\left[\log\frac{1}{Z_i^{2Z_i^2}}\right],$$

where we used Jensen's inequality. We can easily see that the criterion used in KLB and JBB is not suitable for measuring dependency among the coordinates in a basis.

4. IMAGE MODELING BY LSDB

Image and texture modeling is an important application area where LSDB may contribute. Below, we propose two probabilistic models of an image class using the LSDB coordinates.

4.1. Image Models with LSDB as independent coordinates

We start with the simplest model. This model assumes that the LSDB coordinates are really statistically independent, i.e., a probabilistic description of a class of images is a product of empirical marginal pdf's of the LSDB coordinates. Then this model can be written as

$$\boldsymbol{Z} = B_{LSDB}^T\boldsymbol{X} \sim p_{\boldsymbol{Z}}(\boldsymbol{z}) \approx \prod_{i=1}^{n} \widehat{p}_{Z_i}(z_i). \tag{13}$$

In words,

Image Model = Description of the LSDB + Statistics of each LSDB coordinate.

Here, the description of the LSDB consists of the specification of the dictionary used and the specification of the LSDB basis vectors obtained via (12) in that dictionary. The statistics of each LSDB coordinate means either its epdf or empirical cumulative distribution function (ecdf). Sampling *typical* images from this model is easy. We use the inverse cdf of each coordinate to sample a typical coefficient of that coordinate (the inverse of the cdf can be approximated by a simple interpolation of the ecdf or by integration of the epdf). Let $F_{N,i}(z)$ be an ecdf of the ith LSDB coordinate and $F_i(z)$ be an interpolated version of $F_{N,i}$ so that the inverse exists. If $U \sim \text{unif}(0,1)$, then $F_i^{-1}(U)$ obeys F_i, i.e., $Z_i \sim F_i^{-1}(U)$ since $\Pr\{F_i^{-1}(U) < z\} = \Pr\{U < F_i(z)\} = F_i(z)$. Once we sample all the coefficients of the LSDB coordinates to get $\boldsymbol{Z}_{new}$, then we can synthesize an typical image by the inverse transform $\boldsymbol{X}_{new} = B_{LSDB}\boldsymbol{Z}_{new}$. This procedure is summarized as follows.

Step 0 (Optional) Subtract the mean image from each image

Step 1 Search the LSDB coordinates from a specified dictionary of orthonormal bases via (12)

Step 2 Compute the ecdf of each coordinate

Step 3 Sample typical coefficients by the inverse cdf method

Step 4 Apply the inverse transform

Step 5 (Optional) Add the mean image back to the result

If the images of the class contain noise and the noise model is known a priori, e.g., additive white Gaussian noise (WGN), then we can set up a better model including denoising as follows:

Image Model = Description of the LSDB + Statistics of the top m coordinates + (Statistics of $(n-m)$ coordinates)

Here, the m coordinates to be kept as a part of the signal (meaningful) component can be selected via a certain criterion such as the MDL criterion developed in.[6] The last $(n-m)$ terms correspond to noise. So if we do not want to include noise in the model, we can throw away this part. Note that if the noise is WGN, it suffices to record means and variances of each of these $(n-m)$ components instead of keeping their epdf's.

We first demonstrate the LSDB method and compare this with the KLB and JBB using the geophysical acoustic waveforms. For the detailed background of this dataset, see.[10] Here, we want to model the acoustic waveforms (recorded in a borehole with 256 time samples per waveform) propagated through sandstone layers in the subsurface. We have 201 such "sand waveforms" in the training dataset. We used the local cosine dictionary which is easier to deal with the time information than the wavelet packet dictionaries. Figure 1 shows 10 actual waveforms from the dataset, the top 10 vectors of KLB, JBB, and LSDB. Figure 2 shows 10 synthesized waveforms by assuming that the KLB, JBB, LSDB coordinates are all statistically independent, sampling each coefficient separately, and reconstructing them. For each case, we used all 256 coordinates. As we can see from Figure 2, the synthesized waveforms using LSDB visually look more similar to the original waveforms than those using KLB and JBB. In particular, the synthesis by JBB created spurious small oscillations on the interval of time from 0 msec to 0.7 msec. This is due to the inappropriate basis vectors over that interval in the JBB. On the other hand, the LSDB is free from those spurious oscillations.

Now, we tackle a more challenging problem involving images. We use a set of face images, so-called "Rogues' Gallery Problem" to demonstrate our ideas. This dataset consists of digitized pictures of faces of 143 people. These 143 people are a specific group of people; Caucasian students (and some faculty) at Brown University, without glasses, mustache, beard. The dataset was provided to us by Prof. L. Sirovich at Brown University via Prof. M. V. Wickerhauser of Washington University. For more detailed description of these images, see.[15] Figure 3 displays some samples of this dataset. We note that horizontal dilation has been applied so that the pupils are placed on two fixed points if necessary. Now the question is how to build a probabilistic model of this class of images and how to *sample a typical or representative face* of this group.

We first removed the "average face" from each face to make "caricatures," as Kirby and Sirovich put it.[15] We then computed the LSDB with the 2D local cosine dictionary. Figure 4 shows how the LSDB splits these face images

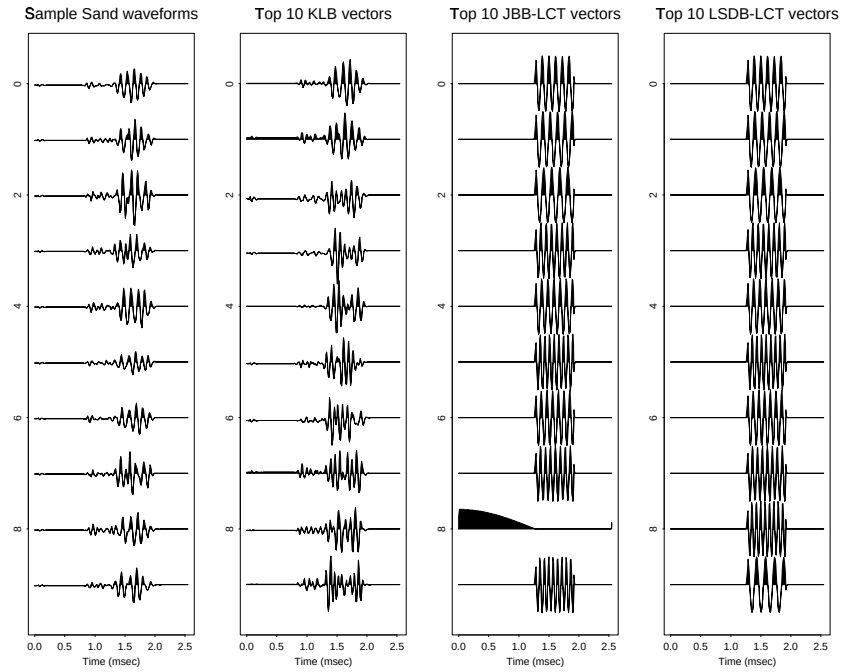

Figure 1. Ten sample sand waveforms, top 10 KLB vectors, top 10 JBB vectors, and top 10 LSDB vectors.

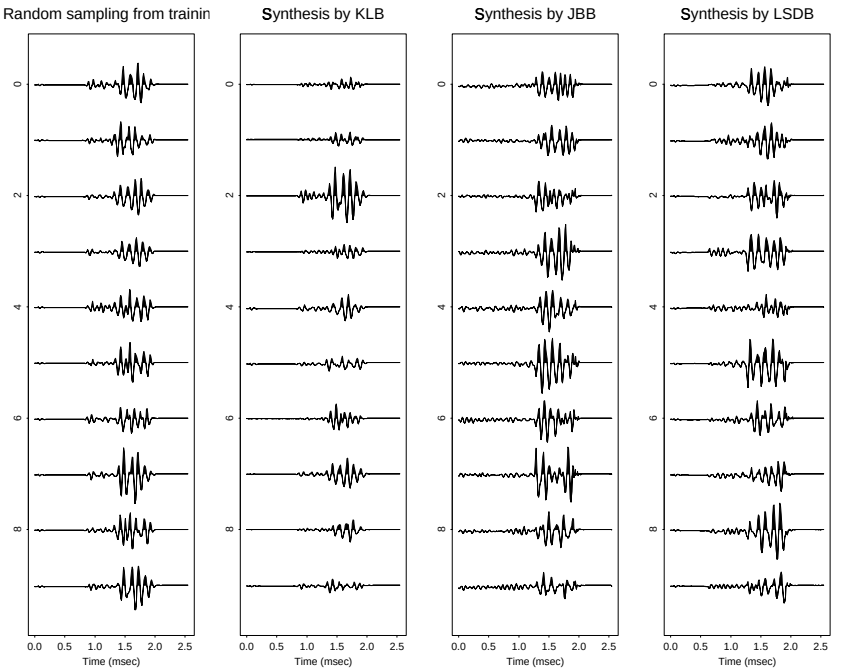

Figure 2. 10 example sand waveforms, 10 synthesized waveforms using KLB, JBB, LSDB, respectively.

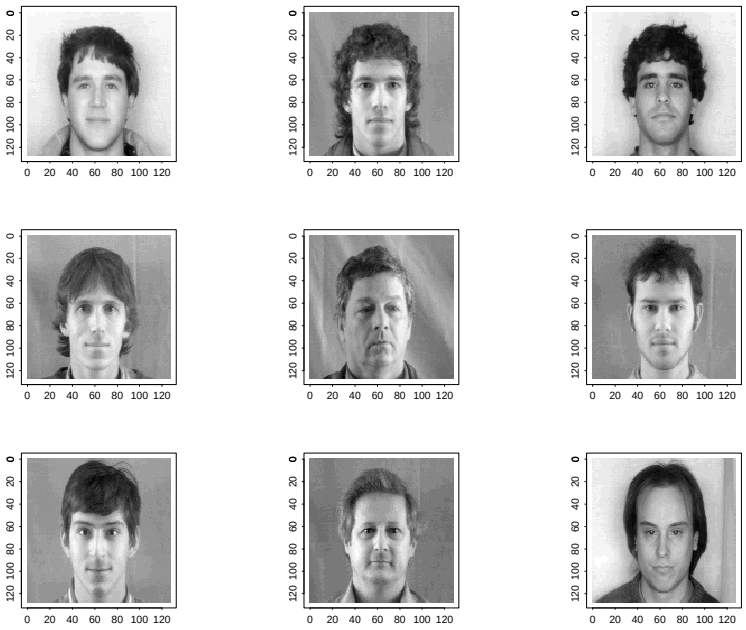

Figure 3. Nine random samples from the Rogue's gallery dataset.

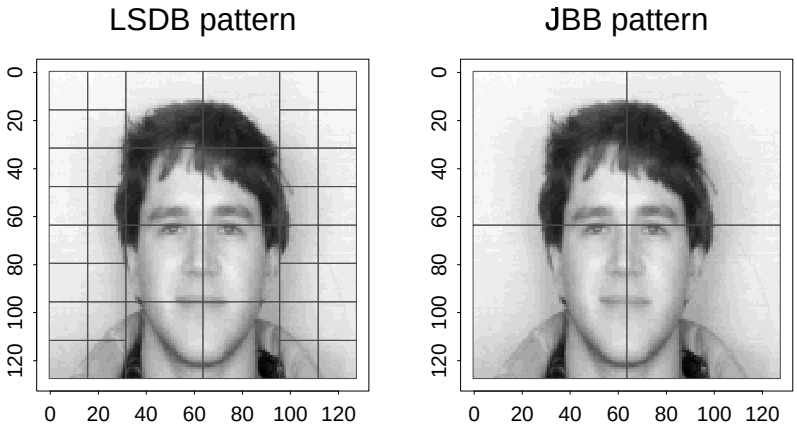

Figure 4. Spatial Partitioning of the images by LSDB and by JBB overlaid on one sample face.

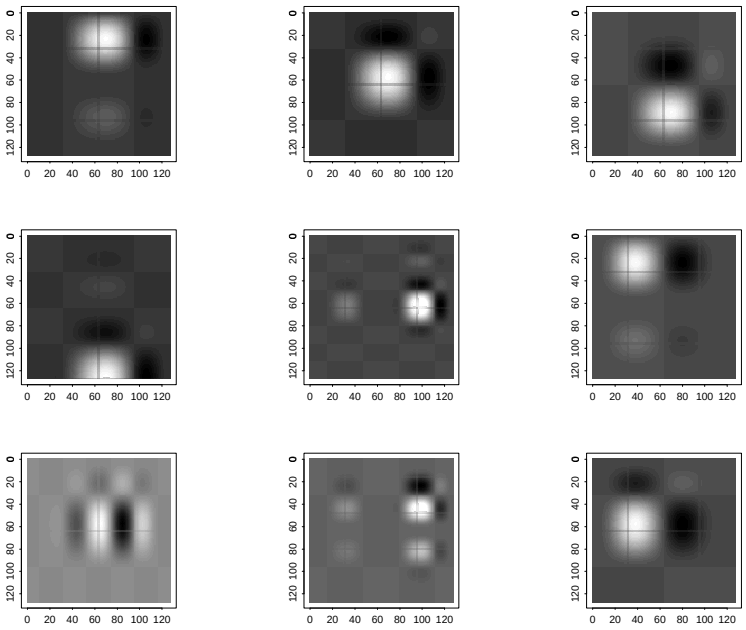

Figure 5. Top 9 most energetic LSDB basis vectors. The underlying basis dictionary is the 2D local cosine dictionary with multiple folding.

spatially and compares it with that of the JBB. Note that the LSDB nicely split the faces into the regions more or less corresponding to hairs, foreheads, eyes and cheeks, chins, and backgrounds. It is interesting to note that Kirby and Sirovich carefully segmented out the oval-shaped portion of the faces containing the eyes, noses, and mouths and removed all the background portion and most of the hair portion for their compression analysis since "it significantly reduced the accuracy of the expression".[15] We note that this natural splitting was done automatically in our case.

On the other hand, JBB simply splits images into four quadrants. We computed the energy distribution of the training images over the LSDB, and Figure 5 shows the top 9 most energetic basis vectors. We sampled all the LSDB coefficients from the ecdf's of each LSDB coordinate separately, applied the inverse transform, and added the average face back to get the following "new face" in Figure 6. The synthesized face is far from a representative face of this class of images. This experiment clearly shows that the LSDB coordinates are not really mutually independent for this dataset. Synthesis using the JBB coordinates (not shown) does not work well either. This failure took us to the next level of the modeling.

Remark 4.1. We note that we used the *multiple folding* local cosine dictionary for both LSDB and JBB. The multiple folding was developed by Fang and Séré[16] and enjoys the good frequency localization of the "bell" functions. Essentially, the size of the action region (the region where two adjacent windows interact) depend on the size of the window itself in the multiple folding. Thus, the larger the image segment, the larger its action region. Because of this, even smaller image segment might have quite distant pixels folded in during early stages of the split. On the other hand, the fixed folding local cosine dictionary employs the constant size of the action region regardless of the size of the windows. Therefore, in order to maintain the good frequency localization we cannot split the images into very small segments. This issue of the frequency localization turns out to be very important for the LSDB algorithm. If we used the fixed folding local cosines by recursively splitting the images into rather small segments (4×4 pixels), then the LSDB was the global frequency basis, i.e., it did not split the image into segments at all. If we split the images three times recursively (the smallest segment has 16×16 pixels), then the LSDB partitioning pattern is exactly the same as the one computed by the multiple folding local cosines. We also note that the multiple folding took care of the symmetry of face pictures around the center line *automatically*. This symmetry was explicitly utilized in the KLB computation by Kirby and Sirovich.[15]

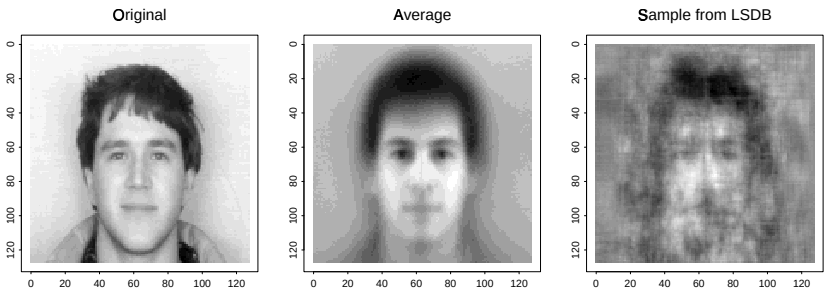

Figure 6. An original image vs. a sampled image from the LSDB coordinates under the independent assumption.

4.2. Image Models with LSDB with KLB

Accepting that the LSDB does not provide us with truly independent coordinates, we form m-dimensional feature space $\mathcal{F}$ by selecting the top m LSDB coordinates (via e.g., MDL criterion), then rotate this feature space coordinates further to have decorrelated coordinates. In other words, we apply KLB-PCA on the top m LSDB coordinates. Now we have the following image model:

Image Model = Description of the LSDB + Description of the KLB of the top m LSDB coordinates + Statistics of these m KLB coordinates +(Statistics of the $(n-m)$ LSDB coordinates)

This can be quite powerful since these m coordinates are already statistically less dependent than the original coordinates and we can compute the m-dimensional KLB rather quickly if $m \ll n$. This procedure can be written as:

Step 0 (Optional) Subtract the mean image from each image

Step 1 Search the LSDB via (12)

Step 2 Select the most energetic m components out of n ($m \ll n$) to form the feature space $\mathcal{F}$

Step 3 Compute the KLB of $\mathcal{F}$ to obtain the decorrelated coordinates

Step 4 Compute the ecdf's of these m coordinates

Step 5 Sample typical coefficients by the inverse cdf method

Step 6 Rotate back these coefficients to the LSDB coordinates

Step 7 Apply the inverse transform

Step 8 (Optional) Add the mean image back to the result

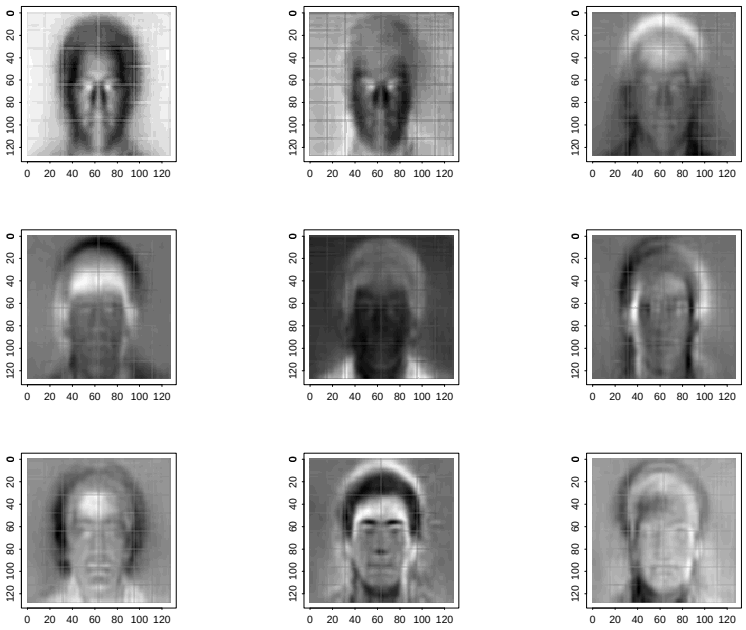

Figure 7. Top 9 KLB vectors computed on the top 800 LSDB coordinates. Compare with Figure 5.

Figure 7 shows the top 9 KLB vectors computed on the top 800 LSDB coordinates. Setting $m = 800$ amounts to using only about 5% of the original information. Each KLB vector—which really looks like a caricature of a face—is a linear combination of the 800 LSDB vectors that are abstract blobs or local oscillations as some of them are shown in Figure 5).

In Figure 8, we compare one of the original image and one realization using the above procedure. Note the clear improvement over Figure 6.

In Figure 9, we compare realizations (samples) from three different models. The images in the first row are realizations of the model (13) with the KLB as the underlying basis. These KLB vectors were computed on the standard coordinates as described in Section 2. So, we abbreviate this model as KLB-STD. The images in the second row are realizations of the model in this subsection but the KLB was computed on the top 800 JBB coordinates instead of the LSDB coordinates. The images in the third row are generated from our proposed model, i.e., the KLB computed on the top 800 LSDB coordinates. These are really "new faces"; they do not exist in the training dataset (of course, in this world, there may exist real people who resemble these faces). The realizations using the KLB-JBB model tend to be more blurry than the other two models. The realizations using the KLB-STD are sharper because they are linear combinations of the original training images, but they look unnatural. The realizations using the KLB-LSDB model, on the other hand, are slightly blurry, but look more natural than the other two models.

5. DISCUSSION

In this section, we discuss the relations of our proposed methods to the others and describe some of our ongoing projects.

5.1. Relation with Local Karhunen-Loève Basis

R. R. Coifman and the author proposed a notion called the local Karhunen-Loève basis (LKLB) in.[17] The idea is to split the signals/images into tree-structured segments by the smooth orthogonal projections,[18] compute the KLB locally within each segment, then prune the tree and merge the segments using a certain criterion to have a basis consisting of a set of localized versions of the KLBs. Computing KLBs locally make sense both computationally and statistically. However, we had difficulty in deciding the basis selection (i.e., tree-pruning) criterion. We examined a few alternative criteria, but all of them were based on the eigenvalues of the autocorrelation matrices computed at

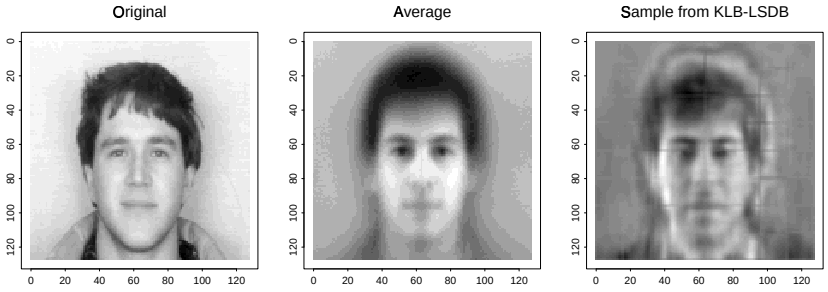

Figure 8. An original image vs. a sampled image from the KLB coordinates computed over the top 800 LSDB coordinates.

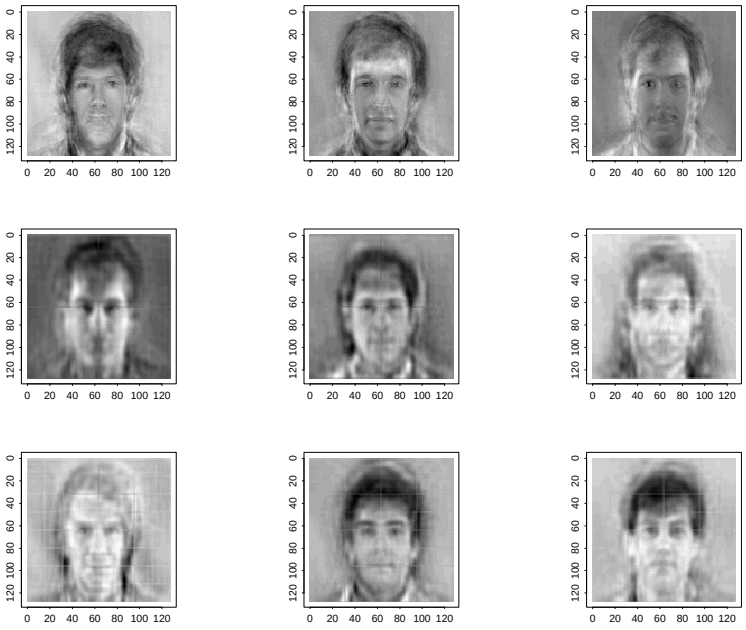

Figure 9. Comparison of the "new faces." Images in the first row are the realizations from KLB-STD, the ones in the second row were from KLB-JBB, and the ones in the third row were from KLB-LSDB.

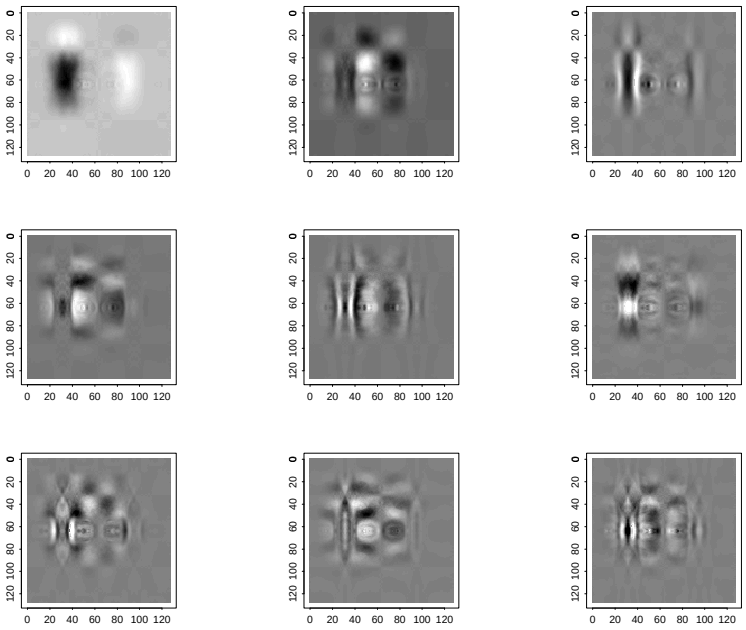

Figure 10. Top 9 "eigen-eyes" computed from the upper left segment of the faces (i.e., the region around right eyes of the faces).

those segments, and in some sense, they were ad hoc. But now, the LSDB offers an optimal (from the basis dictionary point of view) and justifiable (we are selecting a statistically least dependent basis) split of signals and images into segments, we can compute all sorts of bases in each segment. In particular, we can compute KLB in each segment. Figure 10 shows one interesting example. This shows the top 9 KLB vectors computed on the upper left segment shown in Figure 4 corresponding to the right eyes of the faces. The first "eigen-eye" checks the symmetry between the right eyes and left eyes. Remember that we are using the 2D local cosine basis dictionary with multiple folding. That is why the eigen-eyes have activities outside of this upper left segment. Then the subsequent eigen-eyes reveal more detailed structures of eyes as well as the boundary between the face and the background. We are also working on hierarchical image model consisting of the localized KLBs by examining how each segment interact or correlate with the others.

5.2. Image Models with the LSDB with pairwise conditioning

As our experiments showed, the LSDB does not guarantee a truly independent coordinate system in general. So, we considered the KLB of the top m LSDB coordinates as an attempt to make them more independent (in this case only decorrelation is achieved of course). Alternatively, we can examine the dependency among the selected LSDB coordinates more explicitly to some extent. Currently, we are working on algorithms to check the pairwise dependency among the LSDB coordinates and to sample the LSDB coefficients conditionally using some 2D pdf estimation techniques such as ASH2D.[12]

5.3. What if only one training image is available?

If only a single image is available as a training dataset, we need some assumption to compute LSDB. (Note that we can always compute JBB even with a single image since it does not rely on pdf's.) In the case of a homogeneous texture image, it is very natural to consider spatially translated versions of an original image as training images. If we consider all possible translations, then all the epdf's belonging to the same subspace/node are the same. In particular, the epdf's of the root node in the wavelet packet dictionary (corresponding to the pixel coordinates) is the conventional histogram used for common image processing tasks such as histogram normalization to enhance images. This "translation invariant" assumption was also used in the other texture synthesis projects.[19,20]

ACKNOWLEDGMENTS

The author would like to thank Professor L. Sirovich at Brown University and Professor M. V. Wickerhauser at Washington University at St. Louis for providing the digitized face images.

REFERENCES

1. S. Watanabe, "Karhunen-Loève expansion and factor analysis: theoretical remarks and applications," in *Trans. 4th Prague Conf. Inform. Theory, Statist. Decision Functions, Random Processes*, pp. 635–660, Publishing House of the Czechoslovak Academy of Sciences, (Prague), 1965.
2. D. J. Field, "What is the goal of sensory coding?," *Neural Computation* **6**, pp. 559–601, 1994.
3. P. Comon, "Independent component analysis, a new concept?," *Signal Processing* **36**, pp. 287–314, 1994.
4. A. J. Bell and T. J. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," *Neural Computation* **7**, pp. 1129–1159, 1995.
5. R. R. Coifman and D. Donoho, "Translation-invariant de-noising," in *Wavelets and Statistics*, A. Antoniadis and G. Oppenheim, eds., Lecture Notes in Statistics, pp. 125–150, Springer-Verlag, 1995.
6. 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.
7. D. L. Donoho and I. M. Johnstone, "Ideal denoising in an orthonormal basis chosen from a library of bases," *Comptes Rendus Acad. Sci. Paris, Série I* **319**, pp. 1317–1322, 1994.
8. R. R. Coifman and N. Saito, "Constructions of local orthonormal bases for classification and regression," *Comptes Rendus Acad. Sci. Paris, Série I* **319**, pp. 191–196, Jul. 1994.
9. N. Saito and R. R. Coifman, "Local discriminant bases and their applications," *J. Mathematical Imaging and Vision* **5**(4), pp. 337–358, 1995. Invited paper.
10. N. Saito and R. R. Coifman, "Extraction of geological information from acoustic well-logging waveforms using time-frequency wavelets," *Geophysics* **62**(6), pp. 1921–1930, 1997.
11. M. V. Wickerhauser, "Fast approximate factor analysis," in *Curves and Surfaces in Computer Vision and Graphics II*, pp. 23–32, Oct. 1991. Proc. SPIE 1610.
12. D. W. Scott, *Multivariate Density Estimation: Theory, Practice, and Visualization*, John Wiley & Sons, New York, 1992.
13. D. L. Donoho, S. Mallat, and R. von Sachs, "Estimating covariances of locally stationary processes: Rate of convergence of best basis methods," tech. rep., Department of Statistics, Stanford University, 1997.
14. M. V. Wickerhauser, *Adapted Wavelet Analysis from Theory to Software*, A K Peters, Ltd., Wellesley, MA, 1994. with diskette.
15. M. Kirby and L. Sirovich, "Application of the Karhunen-Loève procedure for the characterization of human faces," *IEEE Trans. Pattern Anal. Machine Intell.* **12**(1), pp. 103–108, 1990.
16. X. Fang and E. Séré, "Adapted multiple folding local trigonometric transforms and wavelet packets," *Appl. Comput. Harmonic Anal.* **1**, pp. 169–179, 1994.
17. R. R. Coifman and N. Saito, "The local Karhunen-Loève bases," in *Proc. IEEE-SP Intern. Symp. Time-Frequency and Time-Scale Analysis*, pp. 129–132, IEEE. Jun. 18–21, 1996, Paris, France.
18. M. V. Wickerhauser, "Smooth localized orthonormal bases," *C. R. Acad. Sci. Paris, Série I* **316**, pp. 423–427, 1993.
19. D. J. Heeger and J. R. Bergen, "Pyramid-based texture analysis/synthesis," in *ACM SIGGRAPH*, pp. 229–238, 1995.
20. E. P. Simoncelli, "Statistical models for images: Compression, restoration, and synthesis," in *31st Asilomar Conference on Signals, Systems, and Computers*, IEEE, 1997.