The Least Statistically-Dependent Basis and Its Applications

Naoki Saito
Department of Mathematics
University of California
Davis, CA 95616-8633
saito@math.ucdavis.edu

Abstract

Statistical independence is one of the most desirable properties of a coordinate system for representing and modeling images. In this paper, we propose an algorithm to rapidly construct a coordinate system "closest" to the statistically independent one from a dictionary of bases such as the wavelet packets and local Fourier bases. The criterion is to minimize the sum of the coordinate-wise differential entropy and is quite different from the Joint Best Basis (JBB) of Wickerhauser. We demonstrate the use of the LSDB for image approximation and modeling, and compare its performance with Karhunen-Loève Basis (KLB) and JBB.

1. Introduction

Suppose we are given a set of similar images such as human faces and we want to *learn* the characteristics of those images, i.e., to represent them efficiently 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. This would allow us to *sample* or *simulate* as many new images from this probability model as we want so that we could examine "typical" images in this class and how they look like. This would be a great tool for image diagnostics. 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 independent features faithfully because of the high dimensionality of the problem. Therefore, it makes sense to devise an algorithm to rapidly compute a good coordinate system which is "closest" to the statistically independent one, and to examine how much we can achieve in approximation and probabilistic modeling using such coordinates by assuming that they are in fact independent.

Let us first review a few coordinate systems proposed previously for such applications.

2. PCA, ICA, JBB, and All That

Let $\mathcal{X} \in \mathbb{R}^n$ be an input image space, i.e., a set of all images of a particular class under consideration, where n is a number of pixels in each image. Suppose we are given N training (sample) images. $\mathcal{T} = \{\boldsymbol{X}_1, \ldots, \boldsymbol{X}_N\} \subset \mathcal{X}$, and let us assume that these images are N independent realizations of some stochastic process whose probability density function (pdf) is $f_{\boldsymbol{X}}$. The ultimate characterization of a given image class entails estimating $f_{\boldsymbol{X}}$ from the available training dataset $\mathcal{T}$. Estimating the 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 $f_{\boldsymbol{X}}$, which we normally cannot access.

Now, let us consider what is a good coordinate system to represent and model these images. 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 $\mathcal{C}(B \mid \mathcal{T})$ be a certain functional measuring the cost or inefficiency of the basis B for our problem at hand given a training dataset $\mathcal{T}$. Then, we seek the best coordinates B_*

$$B_* = \arg\min_{B \in \mathcal{L}} \mathcal{C}(B \mid \mathcal{T}),$$

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.

2.1. Karhunen-Loève Basis–Principal Component Analysis

The Karhunen-Loève basis (KLB) a.k.a. Principal Component Analysis (PCA) provides us with a decorrelated coordinate system. The KLB vectors are the eigenvectors of the autocorrelation (or covariance) matrix of the process obeying $f_{\boldsymbol{X}}$. The 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 $\mathbb{R}^n$ [8]. Let B be any basis (not necessarily orthonormal) in $\mathbb{R}^n$, i.e., $B \in \mathrm{GL}(n, \mathbb{R})$, and let $\boldsymbol{Y}$ be the coordinates of the image $\boldsymbol{X}$ relative to the basis B, i.e., $\boldsymbol{Y} = B^{-1}\boldsymbol{X}$. Entropy of the energy distribution over the coordinate axes can be considered as the inefficiency of that coordinate system (i.e., the larger the entropy, the less efficient for feature compression). Let us now define the *entropy function* as

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

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

$$B_{KLB} = \arg \min_{B \in \mathrm{O}(n)} h(\widehat{\boldsymbol{\gamma}}[B]). \tag{1}$$

On the other hand, KLB has several drawbacks. First of all, the criterion (1) does not measure the statistical independence of the coordinates. The KLB only takes care of the second order statistics, i.e., it does "decorrelation." Therefore, the KLB is only optimal for the multivariate Gaussian data since the decorrelation implies the independence for Gaussian data. The next serious problem is an inaccuracy of the sample estimate of the autocorrelation or covariance matrices of the underlying process $f_{\boldsymbol{X}}$. 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). We will demonstrate this problem in Section 4. The KLB computation costs $O(\min(n, N)^3)$ (see more detailed explanation in [7]). Note that having a small N is advantageous only for computational speed, not for the statistical accuracy. On the other hand, if N increases, then the computational cost increases cubically. This is a dilemma of the KLB computation.

2.2. Independent Component Analysis

To overcome the limitation of the PCA to the second order statistics, Comon [1] proposed the so-called Independent Component Analysis (ICA). Given a training dataset $\mathcal{T}$, ICA tries to find a invertible linear transformation that minimizes the statistical dependence among its coordinates. In our notation, ICA can be written as

$$B_{ICA} = \arg \min_{B \in \mathrm{GL}(n,\mathbb{R})} \mathcal{C}_{ICA}(B \mid \mathcal{T}),$$

where $\mathcal{C}_{ICA}(B \mid \mathcal{T})$ measures the degree of statistical dependence of the coordinate system B for the training dataset $\mathcal{T}$. Let us now define differential entropy $H(\boldsymbol{X})$ of the process obeying $f_{\boldsymbol{X}}$.

$$H(\boldsymbol{X}) \stackrel{\Delta}{=} -\int f_{\boldsymbol{X}}(\boldsymbol{x}) \log f_{\boldsymbol{X}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}. \tag{2}$$

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

$$\begin{aligned} I(\boldsymbol{X}) &= \int f_{\boldsymbol{X}}(\boldsymbol{x}) \log \frac{f_{\boldsymbol{X}}(\boldsymbol{x})}{\prod_{i=1}^{n} f_{X_i}(x_i)} \mathrm{d}\boldsymbol{x} \\ &= -H(\boldsymbol{X}) + \sum_{i=1}^{n} H(X_i), \end{aligned}$$

which is simply relative entropy between $f_{\boldsymbol{X}}$ and the product of the marginals $\{f_{X_i}\}$. We note that $I(\boldsymbol{X}) = 0$ if and only if the components $X_1, \ldots, X_n$ are mutually independent. Now, we can write the inefficiency of the coordinate system B as

$$\mathcal{C}_{ICA}(B \mid \mathcal{T}) = I(\boldsymbol{Y}) = -H(\boldsymbol{Y}) + \sum_{i=1}^{n} H(Y_i), \tag{3}$$

where $\boldsymbol{Y} = B^{-1}\boldsymbol{X}$. As mentioned earlier, it is extremely difficult to have a good estimate $\widehat{f}_{\boldsymbol{X}}$ (or $\widehat{f}_{\boldsymbol{Y}}$) for large n, and even the case with $n > 3$ is difficult in practice. Therefore, Comon proposed to approximate (3) using the Edgeworth expansion of $f_{\boldsymbol{Y}}$ 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 $\boldsymbol{Y}$. This computational procedure is even more complicated and expensive than KLB; it costs $O(n^{2.5}N)$. Therefore, 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 a JBB that is the minimum entropy basis among all the bases in the specified dictionary of orthonormal bases [9, Chap. 11]. The JBB criterion is simply written as:

$$B_{JBB} = \arg \min_{B \in \mathcal{D}} h(\widehat{\boldsymbol{\gamma}}[B]).$$

A key difference from (1) 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 or $p = 2$ for local Fourier dictionaries. Recall that a dictionary $\mathcal{D}$ contains more than 2^n different orthonormal bases [9]. 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 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 (2). We can easily get

$$H(\boldsymbol{Y}) = H(B^{-1}\boldsymbol{X}) = H(\boldsymbol{X}) + \log|\det(B^{-1})|.$$

Therefore, if B is a volume-preserving linear transformation, or more specifically, $B \in \mathrm{SL}(n, \mathbb{R})$, then the differential entropy is *invariant* under such transformations.

$$H(\boldsymbol{Y}) = H(B^{-1}\boldsymbol{X}) = H(\boldsymbol{X}).$$

This invariance property is the key for our LSDB algorithm. As long as we deal with the bases in $\mathrm{SL}(n, \mathbb{R})$, we do not need to compute or estimate $H(\boldsymbol{X})$ in (3). The degree of the statistical dependence among the coordinates in a basis can be quantified by only considering the second term in (3) i.e., the sum of the differential entropy of the individual coordinates. Our recent discussion with J. O. Strömberg clarified that $\mathrm{SL}(n, \mathbb{R})$ contains not only $\mathrm{O}(n)$ including all the orthonormal wavelet packet dictionaries and local cosine/sine dictionaries, but also all the *biorthogonal* wavelet packet dictionaries via the fast rotation algorithms [3, Chap. 2]. These biorthogonal dictionaries significantly increase our "vocabulary."

Now, we can state the selection criterion of our *Least Statistically-Dependent Basis* (LSDB):

$$B_{LSDB} = \arg\min_{B \in \mathcal{D}} \sum_{i=1}^{n} H(Y_i). \tag{4}$$

The LSDB is thus obtained by minimizing the sum of the coordinate-wise differential entropy among all possible (bi)orthogonal bases in a specified dictionary of (bi)orthogonal bases $\mathcal{D}$. We note that the basis search in (4) is fast since the sum of the coordinate-wise differential entropy is an additive measure. In practice, as Hall and Morton [4] suggests, the entropy $H(Y_i)$ can be estimated by

$$\widehat{H}(Y_i) = -\frac{1}{N}\sum_{k=1}^{N} \log \widehat{f}_{Y_i}(Y_{i,k}),$$

where $\widehat{f}_{Y_i}$ is a histogram estimator of the pdf f_{Y_i}, and $Y_{i,k}$ is the ith expansion coefficient (relative to B) of the training vector $\boldsymbol{X}_k$, $k = 1, \ldots, N$. Since the histogram computation is relatively cheap, 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. In the LSDB criterion (4), we have

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

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

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

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. Applications

In this section, we discuss two applications of the LSDB. 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 L. Sirovich via M. V. Wickerhauser. For more detailed description of these images, see [5]. We note that horizontal dilation has been applied so that the pupils are placed on two fixed points if necessary. We first removed the "average face" from each face to make "caricatures," as Kirby and Sirovich put it [5]. All of our basis computations and processing are based on these caricatures, and the average face is added back to the results when they are displayed in the figures. For all the JBB and LSDB computations, we use the multiple folding 2D local cosine dictionary (with DCT IV) [2]. We note that the preliminary version of the experiments using the multiple folding LCT with DCT II-III implementation were also reported in [7]. In the following experiments, we use the training dataset $\mathcal{T}$ containing 72 face images randomly selected from the total 143 faces.

4.1. Compression–Approximation

An obvious application of the LSDB is image compression/approximation. Since the redundancy is reduced *explicitly* using the criteria (4), our strategy is simple: sort the

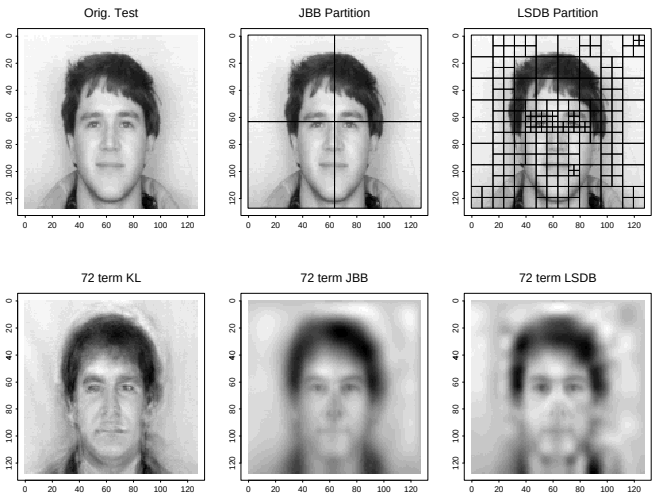

Figure 1. Comparison of KLB, JBB, and LSDB. The original data was not in the training dataset.

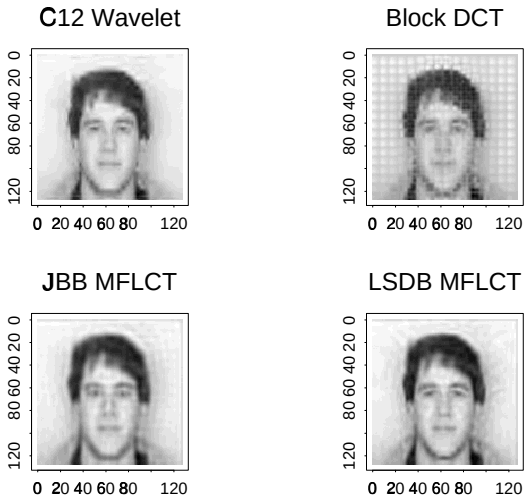

Figure 2. Comparison of the approximations with 800 terms of various bases.

LSDB coordinates in decreasing order in energy, keep only the top m coordinates instead of n, and apply the inverse transform. For the KLB and JBB, we use the same strategy. We now examine how these bases compress/approximate the faces *not in the training dataset.* Figure 1 compares the performance of these three bases. Since the total number of the computable eigenfaces (i.e., the KLB vectors) are 72 for this training dataset, we first compared the KLB approximation with those using the most energetic 72 JBB and LSDB vectors. These 72 vectors were selected by accumulating the energy of the coordinates of the entire 72 training images. If the target image were in the training data, then the KLB approximation would be perfect. However, because the target image is in fact not in the training dataset, the approximation using these 72 KLB vectors is not impressive. In fact, it is not clear whether one can judge whether this approximation represents the same person as the original image. Using the KLB, we cannot do better than this. Now, let us examine the JBB and LSDB approximations. Let us first note that the LSDB nicely split the faces into the regions corresponding to hairs, foreheads, eyes and cheeks, chins, and backgrounds. In particular, the region around the eyes are split into a set of small segments. 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" [5]. We note that this natural splitting was done automatically in our case. On the other hand, JBB simply splits images into four quadrants. The 72 term approximations by the JBB and LSDB shown in Figure 1 are not necessarily better than the one by the KLB. However, they offer much more than the KLB. With the JBB and LSDB, we can use more terms to perform better approximation. With the most energetic 800 terms (i.e., about 6% of the total number of dimensions) instead of 72 terms, we can get the very good approximation as shown in Figure 2. In this figure, we compare the performance of the wavelet basis with the 12-tap Coiflet filter, Block DCT with windows of 8×8 pixels, LSDB with multiple folding LCT, and JBB with multiple folding LCT. We observe that the LSDB approximation perceptually performs best, especially around important signatures such as the eyes, nose, and mouth.

4.2. Probabilistic Modeling

Let us now consider how to build a probabilistic model of this class of images and how to *sample a typical or representative face* of this group.

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. We can then easily simulate *typical* images from this model by 1) independently sampling the LSDB coefficients using the standard sampling methods such as the inversion method or the rejection method (see [6] for these sampling methods), and 2) performing the inverse transform from the LSDB coordinates to the pixel coordinates. The upper-left image of Figure 3 is a simulated face obtained this way. It is

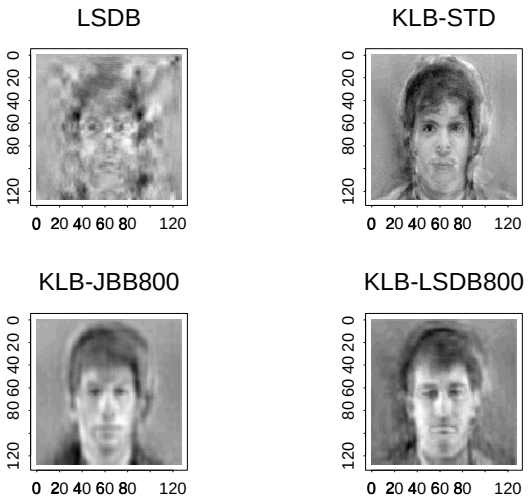

Figure 3. Comparison of the "new faces."

safe to say that this face is far from a representative face of this class of images. This experiment clearly shows that the LSDB coordinates are not yet mutually independent for this dataset. Synthesis using the JBB coordinates (not shown) does not work well either.

Accepting that the LSDB does not provide us with truly independent coordinates, there are two ways to go. One is to model the dependency among the LSDB coordinates, starting from a simple dependency model such as the pairwise dependency model, which we are currently investigating. Here, we report the other method. We form m-dimensional feature space by selecting the top m LSDB coordinates, then further rotate this feature space coordinates by computing the KLB to have decorrelated 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 $m \ll n$.

In Figure 3, we compare the realizations from three different models. The upper-right image is a realization from the model using the 72 eigenfaces computed in the previous subsection. Here, we also assumed that the eigenface coordinates are mutually independent, and sampled the typical coefficients from each coordinate independently. We abbreviate this model as KLB-STD. The lower-left image is a realization of the model using the KLB of the top 800 JBB coordinates. Finally, the lower-right image is a realization of the model KLB-LSDB, i.e., the KLB of 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. Conclusion

We proposed a simple and rapid algorithm to construct a good coordinate system, LSDB, which is "close" to the statistically independent one from a (bi)orthogonal basis dictionaries, clarified the difference between the LSDB and the JBB of Wickerhauser, and demonstrated some of its applications. We are currently investigating 1) how to model the dependency among the LSDB coordinates, 2) how to model textures from a single training image, 3) application to denoising, and 4) incorporation of the biorthogonal and orientation sensitive dictionaries such as the brushlets and the local Fourier dictionary.

References

[1] P. Comon. Independent component analysis, a new concept? *Signal Processing*, 36:287–314, 1994.

[2] X. Fang and E. Séré. Adapted multiple folding local trigonometric transforms and wavelet packets. *Appl. Comput. Harmonic Anal.*, 1:169–179, 1994.

[3] E. Fossgaard. Fast computational algorithms for the discrete wavelet transform and applications of localized orthonormal bases in signal classification. Master's thesis, University of Tromsø, Norway, 1997.

[4] P. Hall and S. C. Morton. On the estimation of entropy. *Ann. Inst. Statist. Math.*, 45(1):69–88, 1993.

[5] 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):103–108, 1990.

[6] B. D. Ripley. *Stochastic Simulation.* John Wiley & Sons, Inc., 1987.

[7] N. Saito. Least statistically-dependent basis and its application to image modeling. In A. F. Laine, M. A. Unser, and A. Aldroubi, editors, *Wavelet Applications in Signal and Image Processing VI*, volume Proc. SPIE 3458, pages 24–37, 1998.

[8] S. Watanabe. Karhunen-Loève expansion and factor analysis: theoretical remarks and applications. In *Trans. 4th Prague Conf. Inform. Theory, Statist. Decision Functions, Random Processes*, pages 635–660, Prague, 1965. Publishing House of the Czechoslovak Academy of Sciences.

[9] M. V. Wickerhauser. *Adapted Wavelet Analysis from Theory to Software.* A K Peters, Ltd., Wellesley, MA, 1994. with diskette.