RIPPLE-FREE LOCAL BASES BY DESIGN

John Lewis, Zhenyao Mo, and Ulrich Neumann

Computer Graphics and Immersive Technology Lab
University of Southern California

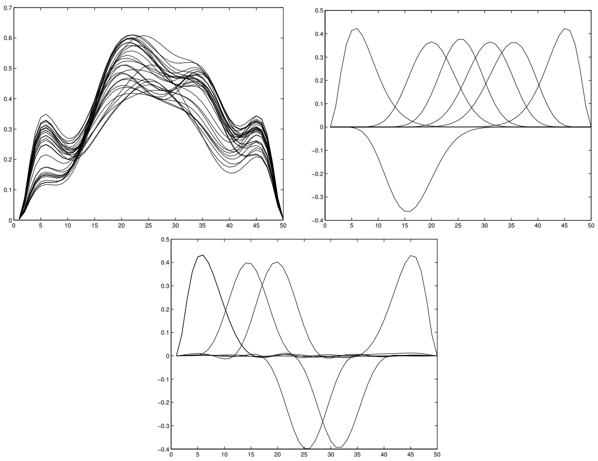

Fig. 1: A synthetic data set (top left) created by mixing the local basis vectors (top right) with strengths determined by random walks. Bottom, local basis vectors blindly recovered from the mixture using our algorithm.

ABSTRACT

In some applications a local or 'parts based' representation is preferable to global basis functions such as those used in Fourier and principal component analysis. In applications that require human understanding and editing of the data, it is also desirable that the basis functions be in some sense as "simple" as possible. This means, for example, that the basis functions should not have Gabor-like ripples if such ripples are not a prominent feature of the data to be represented. This paper introduces a direct local basis construction. Specifically, we show that local bases result from maximizing an appropriate redefinition of pairwise orthogonality while maintaining the ability to represent the data. The resulting basis functions are competitive with (and for some applications superior to) those obtained from existing algorithms, and the construction does not require that the basis coefficients be statistically non-Gaussian or independent, as would be the case with an independent component analysis approach.

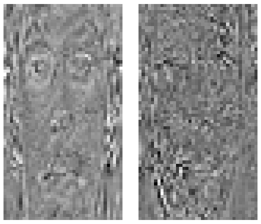

Fig. 2: The 100th and 150th eigenvectors of a collection of faces. It is difficult to intuitively estimate the needed contribution of these images in a task such as enlarging the nose on a given face.

1. INTRODUCTION

Given a set of data vectors $\mathbf{x}_t$ we may wish to represent those vectors as linear combinations of a set of basis vectors (columns of A):

$$\mathbf{x}_t = A\,\mathbf{s}_t$$

It is evident that there is considerable freedom in choosing A. For example, supposing that A has been found, any conforming non-singular matrix W and its inverse can be inserted between A and $\mathbf{s}$, producing another decomposition with basis matrix $\tilde{A} = AW$ and coefficients $\tilde{\mathbf{s}}_t = W^{-1}\mathbf{s}_t$.

Since there are many linear bases that can represent a class of data, some additional criterion is needed to select a particular one among them. While economy of representation is an often-selected criterion, other characteristics of a representation are more useful in some applications. In particular, in applications where the representation must be understood and edited by humans, *locality and sparsity of representation* is desirable. A local basis is one where each basis vector is significantly non-zero over a limited and compact region, and the representation is sparse when a feature of the data is represented by one or a few basis vectors rather than an overlapping combination of many vectors.

Locality and sparsity of representation can be motivated by the police "identi-kit" application in which an operator produces an image of a remembered face. Suppose that the application uses an "eigenface" representation and the operator produces the desired face image by adjusting coefficients of the eigenvectors of a face database. In this representation each eigenvector affects many parts of the face (the representation is not local), and each area of a reconstructed face reflects the contribution of many eigenvectors

Proc. Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP) 2004, p. 684-688, to appear

(the representation is not sparse). When faced with a simple editing task, such as making the nose bigger, it is nearly impossible for a human operator to know by how much the contributions of individual eigenvectors should be changed (Fig. 2). On the other hand, if the basis vectors are local, each region can only be affected by a small number of basis functions, and it is easier to determine the coefficients need to be adjusted.

The signal processing and neural information processing literature has demonstrated that a variety of local basis schemes can be derived by requiring locality and orthogonality (wavelets) or independence (and/or sparsity) of the coefficients (independent component analysis (ICA)). Many of these schemes produce basis functions with ringing, kinks, or ripples. Although a machine can find a combination of functions that sum (if needed) to a ripple-free result, for a human this can be a difficult task. In this paper we demonstrate that ripple-free local representations can also be constructed by maximizing an appropriate redefinition of "orthogonality". This construction produces attractive, smooth shapes and makes no assumptions of the data; in particular it does not require non-Gaussian statistics. Unfortunately, in so doing we must give up strict orthogonality.

Observation: locality, smoothness, and orthogonality cannot be simultaneously achieved in a ripple-free (positive-only) basis with full coverage. ("Full coverage" here means that at any point in the domain, there is at least one basis function with a non-zero value.) If a local basis function is smooth, it will necessarily overlap other functions as it transitions from its maximum to zero (Figure 3). If the basis is also orthogonal, the positive contribution of this overlapping region must be somewhere cancelled by a negative region (ripple).

2. DIRECTLY OPTIMIZING FOR BASIS LOCALITY

Principal component analysis provides the decomposition

$$X = OS$$

of a data matrix $X_{d\times n}$ containing n (zero mean) data vectors $\mathbf{x}_t$ of dimension d into a weighted combination of $k \leq d$ orthogonal basis vectors (columns of $O_{d\times k}$; the columns of $S_{k\times n}$ contain the sequence of corresponding weight vectors). This is a linear decomposition that reproduces that data (if $k = n$), but with basis vectors that are global and unintuitive (Fig. 2).

We seek a different set of linear, not necessarily orthogonal, basis vectors that also represent the data. It is evident that each desired basis vector is necessarily some unknown sum of the columns of O.

With this in mind the decomposition can be rewritten

$$X = \tilde{O}\tilde{S} = (OW^{-1})\,(WS)$$

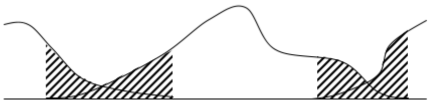

Fig. 3: At every point in the domain at least one basis function must have a significant (non-zero) value. When the basis functions are also smooth each non-zero region of a basis function will have "tails" on either side that necessarily overlap with other basis functions. The area of these overlaps is minimized when each function has only one "hump".

with an invertible matrix W. The matrix W can be chosen either

1. to produce new coefficients $\tilde{S} = WS$ with some desired qualities, or
2. to design new bases $\tilde{O} = OW^{-1}$ with some desired qualities.

The ICA literature emphasizes properties of the coefficients such as independence and sparseness (albeit not using this matrix formulation), with the properties of basis being determined indirectly as a side effect of adjusting the coefficients.

In this paper we instead directly specify the desired basis properties by choosing $R = W^{-1}$. The matrix R "recombines" the orthogonal basis functions in the columns of O so as to produce new, more local, basis function in the columns of $\tilde{O}$. How should R be chosen?

Consider first choosing R so as to minimize a pairwise "absolute value orthogonality"

$$(f, g) = \int |f(t)||g(t)|dt \tag{1}$$

summed across all pairs of basis vectors $f, g \in \tilde{O}$. If $f(t)$ is significant at a particular t, minimizing (1) requires all other vectors to have small values at the same t. R must not be singular, however, and thus OR by definition spans the data. Given these constraints it may not be possible for all other vectors to be strictly zero at t. Instead, the other vectors are as small as possible while still representing the data.

Note that since the eigenvectors are a finite, linear combination of the original data, if the data vectors are smooth the eigenvectors will also be smooth. The new basis vectors OR will be smooth as well, since they are also finite linear combinations of smooth signals. The combined factors that 1) the new basis vectors must be small or zero at regions of the domain that are represented by other vectors, and 2) the new basis vectors are smooth, can only be satisfied when the basis vectors are localized (Fig. 3).

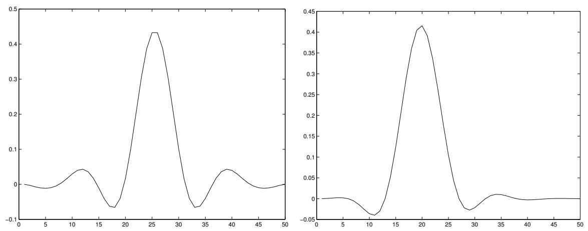

Fig. 4: When the combined "power" of the factors under the integral in Eq. 2 is greater than one the recovered basis functions "ring". Left, $p = 2$ (combined "power" is 4); Right $p = 1$.

There is one problem with this scheme, however (see Fig. 4): the resulting basis vectors resemble the sinc function: they "ring". The reason for this is most easily described by considering the case where f is (at some stage in the optimization for R) nearly equal to some other basis vector g. In this case, with $f(t) \approx g(t)$, the largest value under the integral in Eq. 1 is near the peak of f, where it has the value $f(t)g(t) \approx f(t)^2$. Attempting to minimizing a squared quantity will reduce the peaks at the expense of other regions, resulting in sinc-like ringing in this case. The fact that the two functions f, g in Eq. 1 appear with a combined square power is the problem.

Fractional power orthogonality

Instead, we will choose R to minimize the "square-root orthogonality"

$$(f, g) = \int |f(t)|^p |g(t)|^p dt \qquad (2)$$

(again, summed over all pairs of basis functions), with $p = 0.5$. The combined "power" of the two functions under the integral is now one, so the minimization does not prefer large values over small values. Note that this principle resembles motivations for using the l_1 rather than l_2 norm to produce sparse coefficients [1, 2].

R can be updated from an initial non-singular random matrix using a gradient descent procedure. The objective to be minimized is

$$\begin{aligned} E &= \sum_k \sum_{m \neq k} (f_k, f_m) \\ &= \frac{1}{2}[(\tilde{O}^t, \tilde{O}) - \mathrm{tr}(\tilde{O}^t, \tilde{O})] \end{aligned}$$

where $\tilde{O} = OR$ is the matrix of new (local) basis vectors f_k. The gradient of E with respect to $R_{a,b}$ is (R_k denotes the k-th column of R):

$$\begin{aligned} \frac{\partial E}{\partial R_{a,b}} &= \frac{1}{2}[\frac{\partial}{\partial R_{a,b}}|OR_1|^P|OR_2|^P + \frac{\partial}{\partial R_{a,b}}|OR_1|^P|OR_3|^P + \cdots \\ &+ \frac{\partial}{\partial R_{a,b}}|OR_2|^P|OR_1|^P + \frac{\partial}{\partial R_{a,b}}|OR_2|^P|OR_3|^P + \cdots] \end{aligned}$$

the terms above will be zero except for those corresponding to column b so

$$\begin{aligned} \frac{\partial E}{\partial R_{a,b}} &= \frac{1}{2}[\frac{\partial}{\partial R_{a,b}}|OR_1|^P|OR_b|^P + \frac{\partial}{\partial R_{a,b}}|OR_2|^P|OR_b|^P + \cdots \\ & \frac{\partial}{\partial R_{a,b}}|OR_b|^P|OR_1|^P + \frac{\partial}{\partial R_{a,b}}|OR_b|^P|OR_2|^P + \cdots] \\ &= \sum_{k \neq b} \frac{\partial}{\partial R_{a,b}}|OR_b|^P|OR_k|^P \end{aligned}$$

Looking at one term of this,

$$\begin{aligned} \frac{\partial}{\partial R_{a,b}}|OR_b|^P|OR_k|^P &= \sum_t \frac{\partial}{\partial R_{a,b}}(f_b(t), f_k(t)) \\ &= \sum_t \frac{\partial}{\partial R_{a,b}}|\sum_c O_{t,c}R_{c,b}|^P|\sum_c O_{t,c}R_{c,k}|^P \end{aligned}$$

these terms will be zero except when $c = a$ (we know $k \neq b$ already), so

$$\begin{aligned} &= \sum_t \frac{\partial}{\partial R_{a,b}}|O_{t,a}R_{a,b}|^P|\sum_c O_{t,c}R_{c,k}|^P \\ &= \sum_b \sum_{k>b} \sum_t \frac{pO_{t,a}|\sum_c O_{t,c}R_{c,b}|^P|\sum_c O_{t,c}R_{c,k}|^P}{|\sum_c O_{t,c}R_{c,b}|} \mathrm{sign}(\sum_c O_{t,c}R_{c,b}) \end{aligned}$$

Implementation Trick

While this is the most direct approach to minimizing E, a much simpler implementation is possible. Observe that if the desired f_k are known, then $\tilde{O}$ is known and R can be obtained by $R = O^{-1}\tilde{O}$. Thus, a simpler gradient descent is to take a small step adjusting each f_k (rather than R) to reduce E and then solve for the resulting R at the end of each iteration. In this case the gradient of Eq. 2 with respect to a particular sample f[k] of a digital basis vector is simply

$$\frac{d}{df[k]} f[k]^p g[k]^p = p\frac{f[k]^p g[k]^p}{f[k]} = p\frac{g[k]^p}{f[k]^{1-p}}$$

3. APPLICATIONS

We will demonstrate the local basis construction with a synthetic one-dimensional basis mixture and the two-dimensional face representation problem. The basis construction is a gradient descent starting from random initialization. As is the case with some previously published basis evolution algorithms, some of the discovered basis functions may be duplicates. In practice such duplication can be detected using either sophisticated [3] or simpler means, and the optimization can be restarted from other locations until a complete basis is generated. (Given a nearly complete collection, the complement to the subspace spanned by the collection contains the remaining vectors, and functions in this region can be tried as starting points).

Synthetic Basis Mixture. As a first test of the direct local basis construction, some analytically generated local

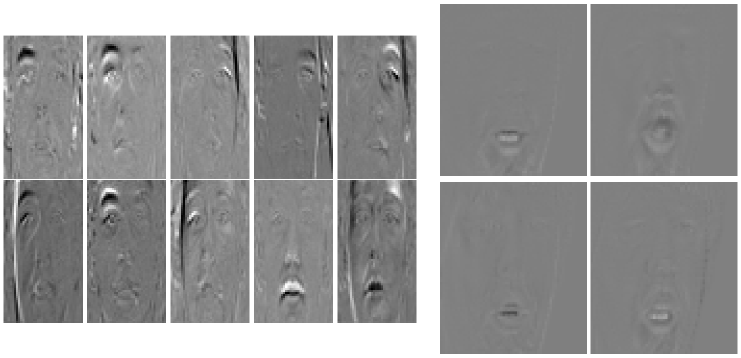

Fig. 5: Two dimensional results: (Left) examples of local basis functions constructed from a collection of registered faces. (Right) example basis functions from approximately registered video frames of a person speaking. (A constant offset has been added to all images to allow visualization.)

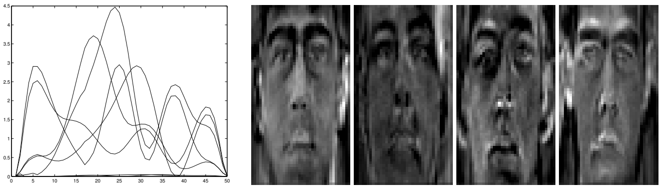

Fig. 6: (For comparison purposes) sample basis vectors discovered by non-negative matrix factorization. The NMF results are more local than eigenvectors, but less so than those shown in Figs. 1,5.

functions (Fig. 1, right) were mixed, generating an evolving "waveform" over time (Fig. 1, left). The weight on each basis is a random walk generated by integrating a uniform density random number generator.

The basis functions recovered using our algorithm are shown in Fig. 1. They are of the same general shape as the original basis vectors up to a change in sign (notice that the leftmost and rightmost functions are teardrop shaped) but are slightly more localized than the original functions.

For comparison, Fig. 6 (left) shows the results of non-negative matrix factorization operating on the same data. The one-dimensional evolution was run for 25K iterations. While the basis vectors resulting from this algorithm are localized in the sense that they usually have a single dominant peak, they also have some intermediate non-local variation.

Face representation results. Fig. 5 shows some of the local basis functions discovered in two-dimensional face representation tasks. For comparison, Fig. 6 (right) shows local basis vectors discovered by non-negative matrix factorization (NMF). The NMF was run for 2160 iterations and was stopped when it did not appear to be changing further. As with the one-dimensional results from NMF the facial basis vectors have some undesirable global detail although they do have dominant local peaks.

4. CONCLUSION

As discussed in the introduction, different applications benefit from different representations. This paper describes an approach for constructing a localized linear basis representing given data. The discovered basis functions are typically smooth and the effects of individual functions are easy to understand due to their locality. The construction also does not require positivity, independence, or non-Gaussian statistics of the data or the basis coefficients. As such it may be applicable to certain practical problems, such as discovering human-editable facial representations from data, that violate the assumptions of independent component analysis.

5. REFERENCES

[1] Scott Shaobing Chen, David L. Donoho, and Michael A. Saunders, "Atomic decomposition by basis pursuit," Tech. Rep., Department of Statistics, Stanford University, February 1996.

[2] B. A. Olshausen and D. J. Field, "Emergence of simple-cell receptive-field properties by learning a sparse code for natural images," *Nature*, vol. 381, no. 6583, pp. 607–609, June 1996.

[3] Jianbo Shi and Jitendra Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.