Noise suppression in seismic data with sparse coding

Sam T. Kaplan and Tadeusz J. Ulrych* University of British Columbia

Abstract

Independent and principal component analysis are used to find a series representation of data. Principal component analysis finds a compact representation of the data, while independent component analysis finds a sparse representation. The data are corrupted with additive random noise, and the coefficients in the series expansion are filtered, removing the portion of the series that is more indicative of noise than signal. For the series representation of the data found using principal component analysis, the filter consists of a simple truncation of the series. For independent component analysis, a Bayesian filter is applied to the coefficients of the series expansion; thus, allowing for prior knowledge to be applied to the distribution of the coefficients.

Introduction

Consider the series expansion

$$\boldsymbol{x}=\sum_{i=1}^{m} y_i \boldsymbol{p}_i = \boldsymbol{P}\boldsymbol{y} \qquad (1)$$

where $\boldsymbol{x}\in\Re^m$ and $\boldsymbol{y}^T=[y_1 \;\; y_2 \;\; \cdots \;\; y_m]$ are random vectors (y_i are random variables), and $\boldsymbol{p}_i\in\Re^m$ is the i^{th} column of $\boldsymbol{P}\in\Re^{m\times m}$. Here, $\boldsymbol{p}_i$ are the basis vectors of the expansion, while y_i are the corresponding coefficients. In other words, the vectors $\boldsymbol{p}_i$ span a subspace containing $\boldsymbol{x}$, and $\boldsymbol{y}$ is the projection of $\boldsymbol{x}$ onto the basis vectors $\boldsymbol{p}_i$. In this paper, both y_i and $\boldsymbol{p}_i$ are computed from $\boldsymbol{x}$ using either principal or independent component analysis (PCA or ICA), with PCA providing a compact representation of $\boldsymbol{x}$, and ICA providing a sparse representation.

For the purpose of noise suppression, $\boldsymbol{y}$ is filtered. When $\boldsymbol{y}$ is computed using ICA, a Bayesian filter is applied, and when $\boldsymbol{y}$ is computed using PCA, the filter simply truncates the series in (1). That is, the methods of PCA and ICA give a prior indication for the distribution of $\boldsymbol{y}$, and filters are designed to use this prior information to *shrink* the coefficients y_i, thereby removing (or suppressing) the portions of the subspace that are more indicative of noise than signal. These noise suppression techniques are applied to a toy seismic example. Other authors have applied the technique to noise attenuation in natural scenes (e.g. Hoyer, 1999; Hyvärinen et al., 2003; Olshausen and Field, 1996)[1].

The Model

The series in (1) requires data for the realizations of $\boldsymbol{x}$. From $\boldsymbol{x}$, both $\boldsymbol{P}$ and the realizations of $\boldsymbol{y}$ are computed.

We form several realizations of $\boldsymbol{x}$ by extracting small (e.g. 16×16 pixel) image patches $\boldsymbol{I}$ from a seismic gather. Presently, we let

$$\boldsymbol{I}=\sum_{i=1}^{m} y_i \boldsymbol{P}_i \qquad (2)$$

be the expansion of $\boldsymbol{I}$ onto the basis patches $\boldsymbol{p}_i$, and for convenience, we coax (2) into the form of (1). In particular,

- An image patch $\boldsymbol{I}$, from (2), is mapped, via lexicographic reordering, to one realization of $\boldsymbol{x}$ in (1).
- The basis patches $\boldsymbol{P}_i$, from (2), are mapped, via a lexicographic reordering, to the basis patches $\boldsymbol{p}_i$ in (1).
- The coefficients y_i, from (2), are simply, the elements in one realization of $\boldsymbol{y}$ in (1).

For example, consider image patches taken from the seismic gather in Figure 1a such that $\boldsymbol{I}\in\Re^{16\times16}$ (i.e. 16×16 pixels). In this case, both $\boldsymbol{x}$ and $\boldsymbol{y}$ would have 256 dimensions. If several such image patches are collected, then both $\boldsymbol{x}$ and $\boldsymbol{y}$ will have several realizations, one for each patch.

Both $\boldsymbol{p}_i$ and $\boldsymbol{y}$ are computed using a linear transformation of $\boldsymbol{x}$,

$$y_i=\boldsymbol{b}_i^T\boldsymbol{x} \qquad \boldsymbol{y}=\boldsymbol{B}x \qquad (3)$$

where $\boldsymbol{b}_i^T$ is a row of $\boldsymbol{B}$, and is chosen to satisfy some statistical criteria for y_i. In the case of PCA, the variance of y_i is maximized providing a compact representation of $\boldsymbol{x}$, and in the case of ICA, the entropy of y_i is minimized providing a sparse representation of $\boldsymbol{x}$.

Compact Representation of $\boldsymbol{x}$

As already mentioned, a compact representation of $\boldsymbol{x}$ is achieved using PCA, and is often used to weed out redundancies, or elicit similarities, from data (e.g. Freire and Ulrych, 1988; Pentland and Turk, 1991; Ready and

1 The work of Olshausen and Field (1996) stems from research involving the mammalian visual cortex (e.g. Field, 1996; Bell and Sejnowski, 1997). Originally a statistical model of the cortex was built using a compact representation. However, a sparse representation proved more suitable.

Wintz, 1973). Here, PCA computes the components of the desired expansion (1) through the computation of principal components

$$y_i^{(pc)}=\left(\boldsymbol{b}_i^{(pc)}\right)^T\boldsymbol{x} \qquad \boldsymbol{y}^{(pc)}=\boldsymbol{B}^{(pc)}\boldsymbol{x}$$

where $\left(\boldsymbol{b}_i^{(pc)}\right)^T$ is the i^{th} row of $\boldsymbol{B}^{(pc)}$, and is chosen to explain the variance of $\boldsymbol{x}$ with as few dimensions (principal components) as possible. PCA finds $\boldsymbol{b}_i^{(pc)}$ which are mutually orthogonal; thus, once $\boldsymbol{b}_i^{(pc)}$ are computed, the basis vectors $\boldsymbol{p}_i^{(pc)}$ in (1) are trivially found. For the remainder of this section, the superscript (pc) is dropped; however, it should be remembered that the solution discussed is the principal component solution.

For the principal component solution, $\boldsymbol{b}_i$ are found such that $\mathrm{var}(y_i)$ are maximized subject to some constraints. Namely that $\boldsymbol{b}_i^T\boldsymbol{b}_i=1$, that the second principal component is uncorrelated with the first, the third uncorrelated with both the first and the second, and so on. The first of these constraints is built explicitly into the cost function using a Lagrange multiplier. Constraining the principal components to be uncorrelated is implicit in the formulation and, as will be shown, falls nicely out of the mathematics. Hence, the appropriate cost functions (for maximization) are

$$\begin{aligned}\phi(\boldsymbol{b}_i) &= \mathrm{var}(y_i)+\lambda_i\left(1-\boldsymbol{b}_i^T\boldsymbol{b}_i\right)\\ &= \mathrm{E}(y_i^2)+\lambda_i\left(1-\boldsymbol{b}_i^T\boldsymbol{b}_i\right)\\ &= \mathrm{E}\left[\left(\boldsymbol{b}_i^T\boldsymbol{x}\right)\left(\boldsymbol{b}_i^T\boldsymbol{x}\right)^T\right]+\lambda\left(1-\boldsymbol{b}_i^T\boldsymbol{b}_i\right)\\ &= \mathrm{E}\left(\boldsymbol{b}_i^T\boldsymbol{x}\boldsymbol{x}^T\boldsymbol{b}_i\right)+\lambda_i\left(1-\boldsymbol{b}_i^T\boldsymbol{b}_i\right)\\ &= \boldsymbol{b}_i^T\boldsymbol{C}_x\boldsymbol{b}_i+\lambda_i\left(1-\boldsymbol{b}_i^T\boldsymbol{b}_i\right)\end{aligned} \qquad (4)\text{–}(5)$$

where λ_i are Lagrange multipliers and $\boldsymbol{C}_x$ is the covariance matrix of $\boldsymbol{x}$. In (4), we assume $\mathrm{E}(y_i)=0$; that is, $\mathrm{var}(y_i)=\mathrm{E}(y_i^2)-\mathrm{E}(y_i)^2$. This assumption is trivial since the mean of $\boldsymbol{x}$ is easily set to zero and $\mathrm{E}(y_i)=\mathrm{E}\left(\boldsymbol{b}_i^T\boldsymbol{x}\right)=\boldsymbol{b}_i^T\mathrm{E}(\boldsymbol{x})$. Taking the gradient of (5) gives

$$\nabla\phi(\boldsymbol{b}_i)=2\boldsymbol{C}_x\boldsymbol{b}_i-2\lambda_i\boldsymbol{b}_i,$$

and setting this result to zero yield the extrema of the cost functions,

$$\boldsymbol{C}_x\boldsymbol{b}_i=\lambda_i\boldsymbol{b}_i. \qquad (6)$$

In (6), we recognize an eigen problem where $\boldsymbol{b}_i$ are the eigenvectors of the symmetric matrix $\boldsymbol{C}_x$, and are therefore mutually orthonormal. Hence,

$$\mathrm{E}(y_iy_j)=\boldsymbol{b}_i^T\boldsymbol{C}_x\boldsymbol{b}_j=\lambda_j\boldsymbol{b}_i^T\boldsymbol{b}_j=\begin{cases}0 & , \quad i\neq j\\ \lambda_j & , \quad i=j\end{cases}. \qquad (7)$$

In (7), we illustrate two ideas. First, we confirm that the principal components are uncorrelated; and second, we demonstrates that the variance of the i^{th} principal component $y_i=\boldsymbol{b}_i^T\boldsymbol{x}$ is the i^{th} eigenvalue λ_i. Hence, ordering the pairs of eigenvectors and eigenvalues in the usual fashion so that $\lambda_1>\lambda_2>\cdots>\lambda_m$ completes the solution.

Since $\boldsymbol{b}_i^T\boldsymbol{b}_i=1$, $\boldsymbol{B}$ is orthogonal and

$$\boldsymbol{x}=\boldsymbol{P}\boldsymbol{y}=\boldsymbol{P}\boldsymbol{B}\boldsymbol{x}\Rightarrow\boldsymbol{P}\boldsymbol{B}=\boldsymbol{I}\Rightarrow\boldsymbol{P}\boldsymbol{B}\boldsymbol{B}^T=\boldsymbol{B}^T=\boldsymbol{P}$$

Therefore, $\boldsymbol{p}_i=\boldsymbol{b}_i^T$, and PCA provides a means for computing both $\boldsymbol{P}$ and $\boldsymbol{y}$ in (1).

Sparse Representation of $\boldsymbol{x}$

A sparse representation of $\boldsymbol{x}$ is achieved using ICA, and in the literature, is known as sparse coding (e.g. Hoyer, 1999). ICA is commonly used for source separation (e.g. Common, 1994), and its description is most readily understood from this point of view. However, in this paper, we are only interested in finding the series expansion in (1), and a full description of ICA, in terms of source separation, is left to Kaplan (2003) and the interested reader. Analogous to PCA, ICA computes the components of the desired expansion (1) through the computation of independent components,

$$y_i^{(ic)}=\left(\boldsymbol{b}_i^{(ic)}\right)^T\boldsymbol{x} \qquad \boldsymbol{y}^{(ic)}=\boldsymbol{B}^{(ic)}\boldsymbol{x}$$

where $\left(\boldsymbol{b}_i^{(ic)}\right)^T$ is the i^{th} row of $\boldsymbol{B}^{(ic)}$, and is chosen such that the independent components $y_i^{(ic)}$ are, indeed, statistically independent and have minimum entropy. As is the case for PCA, there is a simple relation connecting $\boldsymbol{b}_i^{(ic)}$ and the basis vectors $\boldsymbol{p}_i^{(ic)}$ required for the expansion in (1). To simplify notation, we, again, drop the superscript (ic) for the remainder of this section, while keeping in mind that we are computing the independent component solution.

It can be shown, via the central limit theorem, that if y_i are found so that they are maximally non-Gaussian, then they will also be independent. Here, a measure central to information theory, called differential entropy (Shannon, 1984), is considered. It is well known that if only the mean and variance of a continuous random variable y are given, then y has maximum differential entropy exactly when it has Gaussian statistics. Conversely, it is maximally non-Gaussian when it has minimum entropy. Therefore y will be an independent component exactly when it has minimum entropy.

Differential entropy is defined,

$$h(p_Y)=-\int_{-\infty}^{\infty}p_Y(y)\ln p_Y(y)\,dy$$

where in the context of ICA, y is an independent component. The subscript i is dropped for the sake of clarity. Next, we consider an estimate of a measure related to entropy, called negentropy $J(p_Y(y))$ such that (Hyvärinen, 2001)

$$J(p_Y(y))=h(p_\xi(\xi))-h(p_Y(y))\approx\frac{1}{2}\sum_{i=3}^{l}c_i^2 \qquad (8)$$

where $c_i=\mathrm{E}(r_i(y))$, r_i and $p_\xi(\xi)$ is a Normal distribution with the same mean and variance as $p_Y(y)$.

Hence, negentropy measures the distance from a Gaussian random variable, and minimizing the entropy of y is equivalent to maximizing its negentropy. Therefore, independent components correspond to the maxima of (8).

Due to the relation between negentropy and independent components, the ICA problem is reduced to one in optimization with an associated cost function measuring negentropy. Here, this optimization problem is discussed, and the utility of PCA in terms of ICA is explained.

PCA, used as pre-processor, allows for the derivation of much needed constraints for the optimization problem. Given zero mean mixtures $\boldsymbol{x}$, let

$$\boldsymbol{z}=\Gamma x$$

where $\boldsymbol{z}^T=[z_1 \;\; z_2 \;\; \cdots \;\; z_m]$ are whitened mixtures such that $\mathrm{E}(\boldsymbol{z})=\boldsymbol{0}$, $\mathrm{E}(\boldsymbol{z}z^T)=\boldsymbol{I}$ and $\boldsymbol{I}$ is the identity matrix. That is, the random variables z_i, $i=1\ldots m$ are mutually uncorrelated. The utility of $\boldsymbol{z}$ is illustrated by understanding the relation between *uncorrelated* and *independent.* Namely, that independent implies uncorrelated. Consider, for example, two random variables, y_1 and y_2 that follow the bivariate pdf $p_{Y_1Y_2}(y_1,y_2)$, with marginal pdfs $p_{Y_1}(y_1)$ and $p_{Y_2}(y_2)$. Also, let $g_1(y_1)$ and $g_2(y_2)$ be arbitrarily defined functions. The random variables, y_1 and y_2 are said to be uncorrelated if

$$\mathrm{E}(y_1y_2)=\mathrm{E}(y_1)\mathrm{E}(y_2).$$

Further, if y_1 and y_2 are independent, then $p_{Y_1Y_2}(y_1,y_2)=p_{Y_1}(y_1)p_{Y_2}(y_2)$. Thus,

$$\begin{aligned}\mathrm{E}[g_1(y_1)g_2(y_2)] &= \iint_{-\infty}^{\infty} g_1(y_1)g_2(y_2)\,p_{Y_1Y_2}\,dy_1\,dy_2 \\ &= \int_{-\infty}^{\infty} g_1 p_{Y_1}\,dy_1 \int_{-\infty}^{\infty} g_2 p_{Y_2}\,dy_2 \\ &= \mathrm{E}[g_1(y_1)]\mathrm{E}[g_2(y_2)]\end{aligned} \tag{9}$$

Therefore, uncorrelated is a special case of independent where $g_1(y_1)=y_1$ and $g_2(y_2)=y_2$; and hence, independent implied uncorrelated but uncorrelated does not imply independent. Since the goal of ICA is to produce components that are independent, they are also uncorrelated and under orthogonal transformations they stay that way. Therefore, an appropriately chosen rotation transforms uncorrelated components into independent components. This immediately drops the degrees of freedom in the optimization problem by one.

An appropriate choice for Γ is easily found such that

$$\boldsymbol{\Gamma}=\boldsymbol{\Sigma}^{-1}\left(\boldsymbol{B}^{(pc)}\right)^T \tag{10}$$

where $\boldsymbol{B}^{(pc)}$ was found previously in our discussion of PCA, $\Sigma=\mathrm{diag}\left(\sqrt{\lambda_1},\sqrt{\lambda_2},\ldots,\sqrt{\lambda_m}\right)$ and λ_i is the variance of the i^{th} principal component (see (7)).

Next, define a matrix $\boldsymbol{Q}$ such that $\boldsymbol{y}=\boldsymbol{Q}\boldsymbol{z}$, $\boldsymbol{q}_i^T$ is the i^{th} row of $\boldsymbol{Q}$ and $y_i=\boldsymbol{Q}_i^T\boldsymbol{z}$ is an independent component exactly when $\boldsymbol{q}_i$ is chosen such that y_i has maximum negentropy. Hence, an appropriate cost function (for minimization) is

$$\phi(\boldsymbol{q}_i)=-\mathcal{J}(p_Y(y_i))=-\mathcal{J}(p_Y(\boldsymbol{q}_i^T\boldsymbol{z})). \tag{11}$$

As already mentioned, whitening the data further constrains the cost function. In particular, recalling that $\mathrm{var}(y_i)=1$, $\mathrm{E}(y_i)=0$ and that independent components are uncorrelated such that $\mathrm{E}(y_iy_j)=0$, $i\neq j$ gives

$$\mathrm{E}(y_iy_j)=\boldsymbol{q}_i^T\boldsymbol{C}_z\boldsymbol{q}_j=\boldsymbol{q}_i^T\boldsymbol{q}_j=\begin{cases}0 & , \quad i\neq j\\ 1 & , \quad i=j\end{cases}.$$

Thus, the cost function need only be considered on the surface defined by $\boldsymbol{q}_i^T\boldsymbol{q}_i=1$, and multiple local minima may be found using orthogonality.

Hyvärinen (1999) presents a method for optimizing (11) which employs Newton steps in an iterative scheme. In particular, (8) is considered using only one term in its series expansion which gives, for minimization,

$$\phi(\boldsymbol{q}_i)=-\frac{1}{2}\left[\mathrm{E}(r(y_i))\right]^2$$

where $y_i=\boldsymbol{q}_i^T\boldsymbol{z}$. Using approximative Newton steps, Hyvärinen (1999) finds an iterative update rule for the i^{th} row of $\boldsymbol{Q}$:

$$\begin{aligned}\boldsymbol{q}_i^{(k+1)} &= \mathrm{E}(\ddot{r}(y_i))\boldsymbol{q}_i^{(k)}-\mathrm{E}(\dot{r}(y_i)\boldsymbol{z}) \\ \frac{\boldsymbol{q}_i^{(k+1)}}{\|\boldsymbol{q}_i^{(k+1)}\|_2} &\rightarrow \boldsymbol{q}_i^{(k+1)}\end{aligned} \tag{12}$$

The projection back onto the unit circle accounts for the constraint $\boldsymbol{q}_i^T\boldsymbol{q}_i=1$. For the algorithm used in this paper, all rows of $\boldsymbol{Q}$ are updated simultaneously. That is, for each iteration of the optimization routine, (i) Each row $\boldsymbol{q}_i^T$ of $\boldsymbol{Q}$ is updated according to (12), and (ii) the rows of $\boldsymbol{Q}$ are made orthogonal using symmetric orthogonalization such that

$$\boldsymbol{Q}(\boldsymbol{Q}^T\boldsymbol{Q})^{-1/2}\rightarrow\boldsymbol{Q}.$$

Once an optimum $\boldsymbol{Q}$ is found, $\boldsymbol{P}$ is readily computed. In particular, noting that $\boldsymbol{Q}$ is orthogonal, $\boldsymbol{y}=\boldsymbol{Q}\boldsymbol{z}$ and $\boldsymbol{z}=\boldsymbol{\Gamma}x$, we find $\boldsymbol{x}=\boldsymbol{\Gamma}^{-1}\boldsymbol{Q}^T\boldsymbol{y}$. Hence, it follows from (1) that $\boldsymbol{P}=\boldsymbol{\Gamma}^{-1}\boldsymbol{Q}^T$.

As we will show in our toy seismic example, contrary to the basis patched computed using PCA, the independent component solution provides basis patches with some semblance of structure.

Bayesian Filtering

The previous two sections provide two methods for computing the series expansion in (1). When $\boldsymbol{x}$ is corrupted with additive Gaussian random noise, (1) becomes

$$\hat{\boldsymbol{x}}=\sum_{i=1}^{m}\hat{y}_i\boldsymbol{p}_i=\boldsymbol{P}\hat{\boldsymbol{y}}$$

where $\hat{\boldsymbol{y}}=\boldsymbol{y}+\boldsymbol{n}$ and $\boldsymbol{n}$ is representative of the noise. To use (1) for noise suppression, we need some technique for filtering $\hat{\boldsymbol{y}}$. In the case of PCA, $\hat{\boldsymbol{y}}^{(pc)}$ is computed to represent the data with as few dimensions as possible; hence, simply setting $\hat{\boldsymbol{y}}^{(pc)}=0$ for $i>n<m$ should suffice. For ICA, we expect $\boldsymbol{y}^{(ic)}$ to be sparse. To use this information, we follow the work of Hoyer (1999) who uses Bayes formula to construct thresholding functions. When applied to $\hat{\boldsymbol{y}}^{(ic)}$, these thresholding functions shrink the coefficients, suppressing the noise portion of the appropriate subspace. In this section, we consider the sparse representation of $\boldsymbol{x}$, and Bayesian filtering.

Consider random variables y and n such that $\hat{y}=y+n$. Further, recall Bayes formula,

$$p(y|\hat{y})=\frac{p(\hat{y}|y)\,p(y)}{p(\hat{y})}$$

where $p(\hat{y})=\int_{-\infty}^{\infty}p(y)\,p(\hat{y}|y)\,dy$ is a constant. Here, we let $p(\hat{y}|y)=p_n(\hat{y}-y)$ where n is random noise. The idea behind Bayesian filtering is to maximize the posterior density $p(y|\hat{y})$ with respect to y. In particular, letting $n\sim N(0,\sigma_n)$ gives

$$\arg\max_y p(y|\hat{y})=\arg\max_y g(y,\hat{y})$$

where

$$g(y,\hat{y})=\frac{1}{\sqrt{2\pi}\sigma_n}\exp\left(-\frac{(\hat{y}-y)^2}{2\sigma_n^2}\right)p(y)$$

It follows that

$$\arg\max_y p(y|\hat{y})=\arg\min_y\left[\frac{1}{2\sigma_n^2}(\hat{y}-y)^2-\ln p(y)\right] \qquad (13)$$

Defining a score function as $f(y)=-\ln p(y)$, we find a solution to (13) when

$$\frac{1}{\sigma_n^2}(y-\hat{y})+\dot{f}(y)=0\Rightarrow y=\hat{y}-\sigma_n^2\dot{f}(y) \qquad (14)$$

For example, consider, as Hoyer (1999) does, the following pdf,

$$p(y)=c\exp(-ay^2/2-b|y|)$$

where c is a constant, and the parameters a and b are adjusted, allowing control over the sparseness of y. In this case, the score function is

$$f(y)=-\ln c+ay^2/2+b|y|. \qquad (15)$$

Taking the derivative of (15) gives

$$\dot{f}(y)=ay+b\frac{d}{dy}e^{\ln y}=ay+b\frac{|y|}{y}=ay+b\,\mathrm{sign}(y)$$

and applying this result to (14) gives

$$y=\frac{\mathrm{sign}(y)}{1+\sigma^2 a}\left(\frac{\hat{y}}{\mathrm{sign}(y)}-\sigma^2 b\right).$$

Finally, we let $\mathrm{sign}(y)=\mathrm{sign}(\hat{y})$, and ensure that the choice of b does not flip the sign of the coefficients such that

$$y=\frac{\mathrm{sign}(\hat{y})}{1+\sigma^2 a}\max(0,|\hat{y}|-\sigma^2 b). \qquad (16)$$

We use (16) to shrink the components of $\hat{\boldsymbol{y}}^{(ic)}$, and, in turn, filter $\hat{\boldsymbol{x}}$.

Toy Seismic Example

To illustrate the methods, we consider the simple and synthetic data in Figure 1a. From this data, ICA and PCA compute the terms in the series expansion in (1). The basis patches $\boldsymbol{p}_i^{(pc)}$ computed using PCA are plotted in Figure 1d, and the basis patches $\boldsymbol{p}_i^{(ic)}$ computed using ICA are plotted in Figure 1e.

To filter the data using PCA, we simply use the first 26 principal components. That is, the series in (1) is truncated after its first 26 terms. The result of this filtering is shown in Figure 1b. To filter the data using ICA, a Bayesian filter is applied to the coefficients of the expansion $\boldsymbol{y}^{(ic)}$. The result of this filtering is show in Figure 1c.

Disscussion

This paper represents a preliminary investigation of the use of sparse coding in seismic data processing with an application to noise suppression. However, perhaps more interesting than the application of noise suppression is the nature of the basis patches produced by sparse coding. Salle and Olshausen (2002) and Olshausen et al. (2001) discuss using these basis functions for a wavelet basis, allowing for scaling and translation of the basis patches; certainly a worthwhile direction for further investigation.

References

Bell, A. J. and Sejnowski, T. J., 1997, The 'independent components' of natural scenes are edge filters: Vision Research, Vol. 37, p287-314, December.

Common, P., 1994, Independent component analysis, a new concept?: Signal Processing, Vol. 36: p287-314.

Cover, T. M., and Thomos, J. A., 1991, Elements of information theory. Ed. Donald L. Schilling: Wiley Series in Telecommunications. John Wiley & Sons, Inc.

Field, D. J., 1988, What is the goal of sensory coding: Neural Computation, Vol. 6: p559-601

Freire, S. L. and Ulrych, T. J., 1988, Applications of singular value decomposition to vertical seismic profiling: Geophysics, Vol. 53, p778-785

Hoyer, P., 1999, Independent component analysis in image denoising: Master's thesis, Helsinki University of

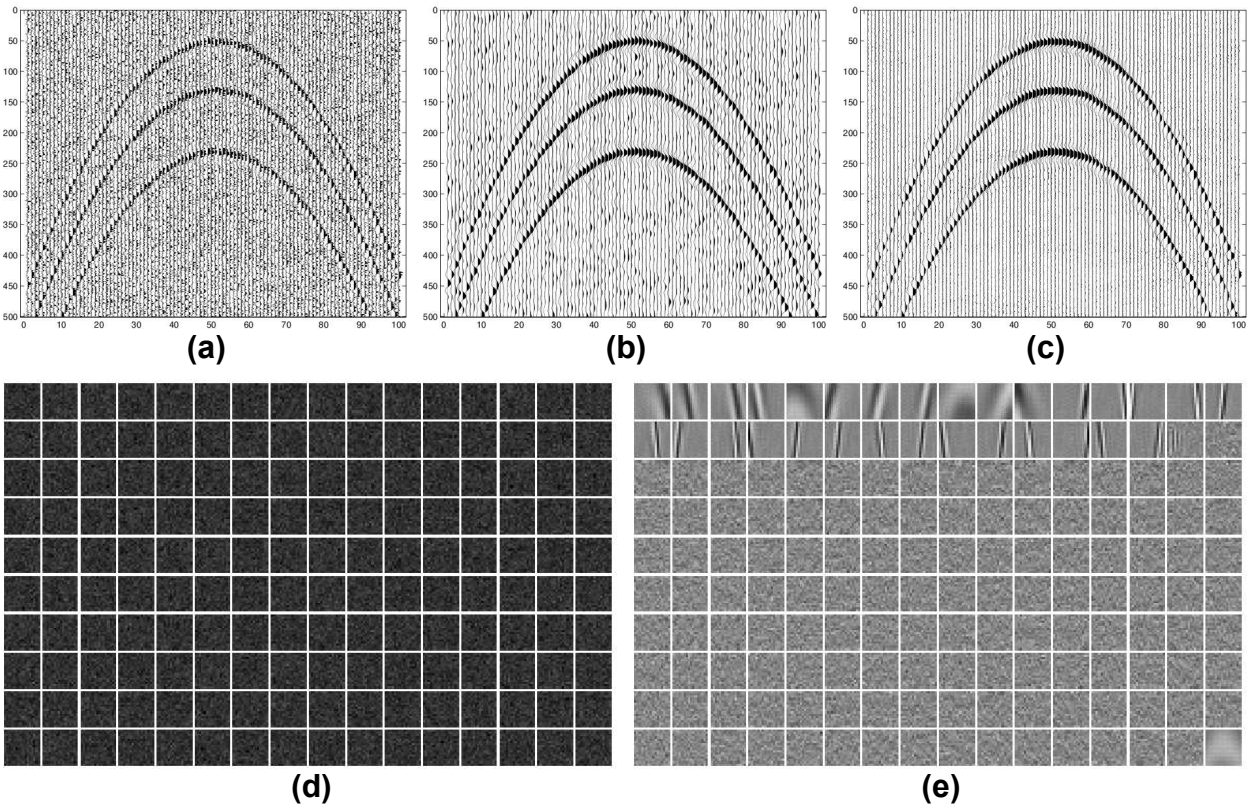

Figure 1: (a) Hyperbolas corrupted with additive Gaussian noise. (b) The data in (a) filtered using PCA. (c) The data in (a) filtered using ICA. (d) The basis patches produced using PCA. (e) The basis patches produced using ICA.

Technology, Department of Compute Science and Engineering.

Hyvärinen, A., Hoyer, P., and Hurri, J., 2003, Extensions of ICA as models of natural images and visual processing: 4th International Symposium on Independent Component Analysis and Blind Signal Separation, p963-974.

Hyvärinen, A., 1999, Fast and robust fixed-point algorithms for independent component analysis: Neural Networks, IEEE Transactions on, Vol. 10, p626-634, May.

Hyvärinen, A., Hoyer, P., and Oja, E., 2001, Image denoising by sparse code shrinkage: Intelligent signal processing. Ed. Simon Haykin and B. Kosko, IEEE Press.

Kaplan, S. T., 2003, Principal and independent component analysis for seismic data: Master's thesis, University of British Columbia.

Olshausen, B. A., and Field, D. J., 1996, Emergence of simple-cell receptive field properties by learning a sparse code for natural images: Nature, Vol. 381, p607-609.

Olshausen, B. A., Sallee, P., and Lewicki, M. S., 2001, Learning sparse image codes using a wavelet pyramid architecture: Advances in Neural Information Processing Systems. Ed. T. K. Leen, T. G. Dietterich, and V. Tresp, MIT Press,

Pentland, A., and Turk, M., 1991, Eigenfaces for recognition: Journal of Cognitive Neuroscience, Vol 3, p71-86, March.

Ready, P. J. and Wintz, P. A., 1973, Information extraction, SNR improvement and data compression in multispectral imagery: IEEE transactions on Communications, p1123-1131, October.

Sallee, P., and Olshausen, B. A., 2002, Learning sparse multiscale image representations: Advances in Neural Information Processing Systems. Ed. Suzanna Becker, Sebastian Thrun, and Klaus Obermayer.

Shannon, C. E., 1948, A mathematical theory of communications: Bell System Technical Journal, Vol. 27, p379-423,623-659, July-October.