HELSINKI UNIVERSITY OF TECHNOLOGY
Department of Computer Science and Engineering

Independent component analysis of image data

Jarmo Hurri

The thesis belongs to the special subject of information science
Supervisor Professor Erkki Oja
Instructor Doctor Juha Karhunen

Espoo, Finland 17.3.1997

HELSINKI UNIVERSITY OF TECHNOLOGY **ABSTRACT OF THE MASTER'S THESIS**

Author and name of the thesis:
Jarmo Hurri
Independent component analysis of image data

Date: 17.3.1997 **Number of pages:** 78

Department:	Department of Computer Science and Engineering
Professorship:	Tik-61 Computer and Information Science
Supervisor:	Professor Erkki Oja
Instructor:	Doctor Juha Karhunen

The objective of independent component analysis is to represent a set of multidimensional measurement vectors in a basis where the components are statistically independent or as independent as possible. This redundancy reduction between components relates independent component analysis to many other theoretical and practical areas of science that are concerned with information, such as information theory, compression, sensory coding and pattern recognition.

One of the first cases where the principle of redundancy reduction was presented was the analysis of the human visual system. Ever since the first time this idea was brought up, there has existed a need to study the reduction of statistical dependencies in images.

A set of new algorithms for independent component analysis has been developed during the 1990's. Some of these have been applied to image data, but no larger investigation has been performed to determine the properties and limitations of the method to solve this problem.

In this master's thesis the applicability of independent component analysis to analyze statistical dependencies in images has been explored. This has been done by applying some of the new algorithms to image data. The relevant properties of the resulting components have been studied, as well as their connections with closely connected areas of information science.

Keywords: independent component analysis, redundancy reduction, image processing, feature extraction

TEKNILLINEN KORKEAKOULU **DIPLOMITYÖN TIIVISTELMÄ**

Tekijä ja työn nimi:

Jarmo Hurri
Independent component analysis of image data
(Kuvainformaation analysointi riippumattomien komponenttien menetelmällä)

Päivämäärä: 17.3.1997 **Sivumäärä:** 78

Osasto:	Tietotekniikan osasto
Professuuri:	Tik-61 Informaatiotekniikka
Valvoja:	Professori Erkki Oja
Ohjaaja:	Tekniikan tohtori Juha Karhunen

Riippumattomien komponenttien menetelmässä joukko moniulotteisia mittausvektoreita pyritään esittämään kannassa, jossa komponentit ovat tilastollisesti riippumattomia tai mahdollisimman riippumattomia toisistaan. Komponenttien välisten keskinäisten riippuvuuksien vähentäminen on keskeistä myös monissa muissa teoreettisissa ja käytännönläheisissä informaatiotekniikan osa-alueissa, kuten informaatioteoriassa, tiedon tiivistämisessä, aisti-informaation käsittelyperiaatteissa ja hahmontunnistuksessa.

Ihmisen näköjärjestelmän tutkimus oli yksi ensimmäisistä osa-alueista, jossa keskinäisten riippuvuuksien vähentämisen periaate esitettiin. Kuvainformaation tilastollisten riippuvuuksien vähentämistä on tutkittu siitä lähtien kun ajatus ensimmäisen kerran esitettiin.

1990-luvulla on kehitetty joukko algoritmeja riippumattomien komponenttien menetelmän toteuttamiseksi. Joitain näistä algoritmeista on sovellettu kuvainformaatioon, mutta suurempaa tutkimusta ei ole tehty menetelmän ominaisuuksien ja rajoitusten selvittämiseksi.

Tässä diplomityössä on tutkittu riippumattomien komponenttien menetelmän soveltuvuutta kuvien sisältämien tilastollisten riippuvuuksien tutkimiseen. Pääasiassa tähän on pyrittu soveltamalla joitain uusista algoritmeista kuvadataan. Syntyneiden komponenttien olennaisia ominaisuuksia on tarkasteltu eri menetelmin. Lisäksi on pyritty hahmottamaan menetelmän yhteyksiä muihin lähellä oleviin informaatiotekniikan osa-alueisiin.

Avainsanat: riippumattomien komponenttien menetelmä, riippuvuuksien vähentäminen, kuvankäsittely, piirteenirrotus

Preface

Independent component analysis is one of the main areas of research in the Laboratory of Computer and Information Science at Helsinki University of Technology. In spring 1996 I had the pleasure to join the independent component analysis group in order to make my own contribution to the subject.

I would like to thank the Laboratory of Computer and Information Science and the Neural Network Research Center for providing an inspiring environment and good facilities for the work. Especially I would like to thank professor Erkki Oja, who provided me with the opportunity to prepare my thesis on this extremely interesting subject, and also supervised the thesis. I would also like to thank Dr Juha Karhunen for instructing this thesis. Many other persons have been of great help to me, but my thanks are directed in particular to MSc. Aapo Hyvärinen, MSc. Harri Lappalainen, and MSc. Petteri Pajunen for their invaluable advices. The set of natural images used in the experiment was photographed by Mr Janne Sinkkonen, to whom I am greatly indebted for providing these excellent works of art.

In Espoo, 17th March, 1996

Jarmo Hurri

Contents

Abbreviations and notation

DCT	discrete cosine transform
DFT	discrete Fourier transform
ICA	independent component analysis
PCA	principal component analysis
$\bar{0}$	a vector of zeros
$\bar{1}$	a vector of ones
$\alpha, \beta, \gamma, \ldots$	scalars
λ, λ_i	Lagrange multipliers
ξ, ξ_i	eigenvalues
$a, \ldots, e$	vectors
e, e_i	eigenvectors
a_i	ith component of vector a or ith column vector of matrix A
s	source vector
s_i	one source
v	whitened data vector
x, y	random variables or random vectors
x_i, y_i	random variables or components of random vectors
$A, B, C, \ldots$	matrices or constants
A	ICA mixing matrix
B	whitened ICA mixing matrix
E	a matrix of eigenvectors
D	a diagonal matrix of eigenvalues
V	whitening matrix
W	ICA separating matrix
$\mathcal{L}(R^m, R^n)$	linear transformations from R^m to R^n ($n \times m$ matrices)
$i, \ldots, n$	natural numbers
$a(k)$	kth iteration of a
m	the dimension of mixture vector
n	the number of sources
f, g, h	functions
f_x	density function of random variable x
f_{xy}	the joint density function of random variables x and y
$\mathrm{E}\{\cdot\}$	expected value
$\mathrm{var}\{\cdot\}$	variance
$\mathrm{cov}\{\cdot\}$	covariance matrix

CHAPTER 1

Introduction

The visual information that can be obtained from all around us has certain characteristics as a measured signal. It contains for example edges, lines and different textures, all of which are related to objects and their properties. An efficient information processing system that uses visual information and is designed to operate in this environment should utilize these characteristics as efficiently as possible.

During the evolution the human visual system has had the opportunity to develop such an efficient image processing strategy. In the 1960's Barlow [**3**] suggested that the goal of the low level visual system is to use the a priori knowledge about the visual environment to reduce redundancy in the data. The redundancy is present in the images as statistical dependencies. If we see a part of the visual scene we will be able to predict some nearby areas better than when we have not seen the scene. Here "better" means better in the statistical sense. This is illustrated in Figure 1. If we take all the scenes in the world containing the two lines and guess for each such scene that the missing part in the circle contains a line connecting the two other lines, we will on the average be right more often than if we make the same guess for an unknown scene. Thus there is a statistical dependence between the middle line and the surrounding lines.

The previous was an example of simple linear statistical dependence or correlation between two phenomena in visual information. What this means is that the existence of one phenomenon is simply a positive or negative sign of the other one: no matter how strong or weak the two separate lines are, presumably the middle line is there. The reduction of such dependencies is relatively easy to solve. The situation is different for nonlinear dependencies. In these the magnitude and sign of the effect may vary in different subranges of measured values. Reducing such dependencies in a data set is not an easy problem to solve. In fact, there are currently many different areas of research in which this is a central problem.

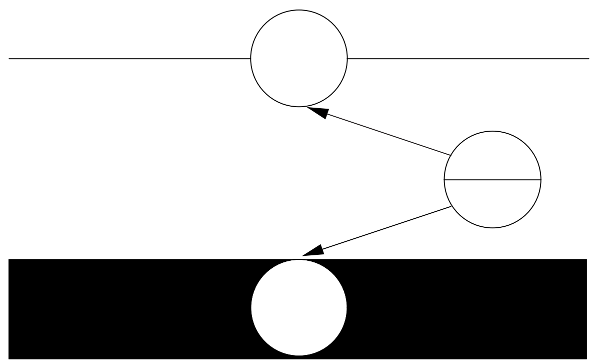


FIGURE 1. An illustration of statistical dependence in images.

Independent component analysis is one such area. In independent component analysis the purpose is to transform the signal by relatively simple (linear) operations so that statistical dependencies vanish or are at a minimum. Even though the set of situations in which this is possible is limited, it seems reasonable to try to apply the method to image data.

The purpose of this thesis is to apply independent component analysis to natural images. The work that has motivated the thesis is that of Bell and Sejnowski [**6**], in which the authors used their algorithm to extract the independent components from images. As a result they found out that the independent components or statistically independent constituents of images are short lines and edges. Interestingly similar properties have been observed in the human visual system. The experiments of Bell and Sejnowski were however performed with just one algorithm which has its shortcomings. So there still exists a need to examine the subject experimentally by applying different algorithms and assessing their applicability to the problem.

The thesis consists of two parts. The first part is an introduction into independent component analysis. It contains sufficient information for the reader to understand the basics of the applied algorithms. This is necessary in order to be able to interpret the results obtained in the second part, which contains the descriptions and analyses of the actual experiments.

Part 1

An introduction to independent component analysis

CHAPTER 2

Basic concepts

1. Principal component analysis

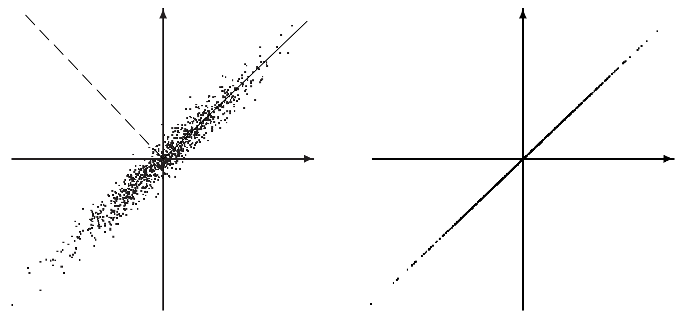
Principal component analysis (for example [**24**]) is a method for reducing the dimensionality of data, that is, representing a set of m dimensional vectors with $n < m$ components for each vector so that information is lost "as little as possible." In terms of linear algebra the problem of dimensionality reduction consists of finding a new basis for the data so that if we drop out or zero some of the components in the new basis, the reconstruction error is as small as possible.

Example 2.1. For illustrational purposes consider the simple example of Figure 1. Here in Figure 1(a) we have a two dimensional data set in which there is a significant correlation (linear dependence) between the components. If we project the data onto the subspace indicated by the solid line in the figure — that is, represent the data with just one number, the value of projection onto the subspace — we get the data set of Figure 1(b).

In practice the most easily computable norm for establishing an optimality criterion for the conservation of information is the mean square norm. If random vector x represents the original data and the estimate in the new basis is $\hat{x}$, the reconstruction error random vector $e = x - \hat{x}$. Then the goal is to minimize $\mathrm{E}\left\{||e||^2\right\}$. Assume that $\mathrm{E}\{x\} = \bar{0}$. It is proven in standard references (for example [**43, 24**]) that the optimal new basis consists of the eigenvectors of the covariance matrix of x

$$\mathrm{cov}\{x\} = \mathrm{E}\left\{xx^T\right\},$$

and that the eigenspaces that retain the most significant amount of information are those that correspond to the largest eigenvalues. These eigenvalues also represent the variances of projections onto the subspaces, and the PCA directions are optima of



(a) A data set and its principal axes.

(b) Reduction of the data to one dimension.

Figure 1. A simple example of principal component analysis.

variances of projections. It can be easily seen in Figure 1(a) that the PCA directions represent the minimum and maximum directions of variance. We also know that the PCA basis is orthogonal, because for an invertible real symmetric matrix — like $\operatorname{cov}\{x\}$ in the nondegenerate case — there exists a set of orthogonal eigenvectors.

Let us denote the set of eigenvector/eigenvalue pairs of $\operatorname{cov}\{x\}$ by $\{(e_1,\xi_1),\ldots,(e_n,\xi_n)\}$ and let $E = [e_1 \ \cdots \ e_n]$ and $D = \operatorname{diag}(\xi_1,\ldots,\xi_n)$ (a matrix with elements $\xi_1,\ldots,\xi_n$ on diagonal). If the data is projected onto the new basis, the new components are uncorrelated. This can be seen by considering the components in the new basis, given by $x_{\mathrm{PCA}} = E^T x$, for which

$$\begin{aligned}\operatorname{cov}\{x_{\mathrm{PCA}}\} &= \mathrm{E}\left\{x_{\mathrm{PCA}}x_{\mathrm{PCA}}^T\right\} = \mathrm{E}\left\{E^T x x^T E\right\} \\ &= E^T \operatorname{cov}\{x\} E = \underbrace{E^T E}_{I} D \underbrace{E^T E}_{I} = D.\end{aligned}$$

Here we have used the eigenvalue decomposition of the covariance matrix, $\operatorname{cov}\{x\} = EDE^T$, and the fact that E is orthogonal because the eigenvectors form an orthonormal vector set.

PCA can be used in many different ways in statistical signal processing. Often small eigenvalues can be associated with noise, and the eigenvalue spread can be used to estimate the dimension of the data. Noise may be removed by reducing the dimension of the data.

2. The concept of independence

Two events $\mathcal{A}$ and $\mathcal{B}$ in an event space of an experiment are called *independent* if

$$\mathrm{P}(\mathcal{AB}) = \mathrm{P}(\mathcal{A})\,\mathrm{P}(\mathcal{B}).$$

Noting that the conditional probability $\mathrm{P}(\mathcal{B}|\mathcal{A})$ is given by

$$\mathrm{P}(\mathcal{B}|\mathcal{A}) = \frac{\mathrm{P}(\mathcal{AB})}{\mathrm{P}(\mathcal{A})}, \tag{1}$$

we can see that independence implies $\mathrm{P}(\mathcal{B}|\mathcal{A}) = \mathrm{P}(\mathcal{B})$, if $\mathrm{P}(\mathcal{A}) \neq 0$. Actually this condition is equivalent to independence, since the implication holds in the other direction by (1). Assuming that $\mathrm{P}(\mathcal{B}) \neq 0$ a corresponding equivalence $\mathrm{P}(\mathcal{A}|\mathcal{B}) = \mathrm{P}(\mathcal{A}) \Leftrightarrow \mathrm{P}(\mathcal{AB}) = \mathrm{P}(\mathcal{A})\,\mathrm{P}(\mathcal{B})$ can be derived by a similar procedure.

Equations $\mathrm{P}(\mathcal{B}|\mathcal{A}) = \mathrm{P}(\mathcal{B})$ and $\mathrm{P}(\mathcal{A}|\mathcal{B}) = \mathrm{P}(\mathcal{A})$ have an appealing intuitive interpretation: if one event happens, it does not give us any *additional information* about the probability of the other event. It can be proven that if $\mathcal{A}$ and $\mathcal{B}$ are independent, then all pairs $\{\mathcal{A},\bar{\mathcal{A}}\} \times \{\mathcal{B},\bar{\mathcal{B}}\}$ are independent [**48**], so independence holds for presence and absence of the event. When events are concerned, the presence or absence of events is the only information we can get from an experiment. In terms of information, independence of two events could be described so that information about one event gives no additional information about the other one.

In the case of random variables independence has a very similar interpretation. The definition of independence of two random variables x and y is

$$\mathrm{P}(x \in A, y \in B) = \mathrm{P}(x \in A)\,\mathrm{P}(y \in B),$$

where $A, B \subset R$. This is equivalent to [**48**]

$$f_{xy}(x,y) = f_x(x)\,f_y(y), \tag{2}$$

assuming that the densities exist. The conditional density

$$f_y(y|x) = \frac{f_{xy}(x,y)}{f_x(x)}$$

becomes $f_y(y|x) = f_y(y)$. In this case information about the *value* of one random variable gives us no information about the value of the other one.

The significance of independence is due to the fact that the independent components can often be processed separately. Consider a system that is trying to learn some aspects of the incoming data. If the data can be separated into independent components, then the components have "nothing to do with each other." We may infer properties by examining characteristics of the individual components and combine them in the end to form some view of the data.

EXAMPLE 2.2. As a straightforward example consider the estimation of the joint density of independent random variables x and y. By the definition of independence we can estimate the marginal densities separately, getting estimates $\hat{f}_x(x)$ and $\hat{f}_y(y)$ and then multiply them to form an estimate for the joint density $\hat{f}_{xy}(x,y) = \hat{f}_x(x)\hat{f}_y(y)$.

However, not all statistical quantities behave nicely even for independent random variables.

EXAMPLE 2.3. Let $h(x,y)$ be a function of two variables. Then h is called *separable,* if it can be written in form $h(x,y) = h_1(x)h_2(y)$. Let x and y be two independent random variables and $g(x,y)$ be any nonseparable function, such as $g(x,y) = \sqrt{x+y}$. Then the expectation of g is given by

$$\begin{aligned} \mathrm{E}\{g(x,y)\} &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x,y) f_{xy}(x,y)\,dydx \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x,y) f_x(x)\,dx f_y(y)\,dy, \end{aligned}$$

but because of the nonseparability of g this equation cannot be broken further.

It is also often thought that independent component analysis could be used for data analysis, because the separated independent components of a complex data set could describe some fundamental underlying factors of the process. There are many problems with this approach, including the restrictions set by current linear ICA model, but one fundamental restriction is that independence is a very strong condition.

EXAMPLE 2.4. As a nonmathematical example from the performing arts, let us consider the "chaotic" theater experiment from Denmark. There the behavior of a group of actors was governed by the activity of ants living on the other side of the world. The director of the experiment stated that except for the control mechanism the two systems were independent of each other. However, understanding that the director may have meant that the actions of the other organisms do not effect those of the other ones, there exists a significant "statistical" dependence between the behavior of ants and actors: both groups of living organisms are highly dependent of the day-night cycle. So during the night in Denmark we may expect daytime activity from the ant community.

EXAMPLE 2.5. Exploring vast masses of financial data is a central problem in economics. Finding the inner driving processes would be very valuable for this task. But what could we even imagine to find, i.e., what is truly independent in economics? This does not mean that examinations would be condemned to failure, but care must be taken to assess the feasibility, reliability and meaning of the results.

3. Linear model of independent component analysis

Let us assume that we have some phenomenon which manifests itself through a set of n independent random variables. We shall denote the combination of these variables with random vector $s = [s_1\ s_2\ \dots\ s_n]^T$. Components $s_1, s_2, \dots s_n$ are called *sources* and s is called the *source vector.* This name is implied by independence: the sources are thought to be independent sources of information.

Now suppose that the original independent source components are observed via a linear process. Denote the observed random vector by x_{obs}. Since the process is assumed linear, the relation between s and x_{obs} can be modeled as

$$x_{\text{obs}} = As. \tag{3}$$

Here matrix $A \in \mathcal{L}(R^n, R^m)$ is referred to as a *mixing matrix,* since it mixes the independent sources. Its companion is the *separating matrix* $W \in \mathcal{L}(R^m, R^n)$, given by equation

$$s = Wx_{\text{obs}}. \tag{4}$$

Note that all cases $m < n$, $m = n$ and $m > n$ are possible. A reader familiar with the theory of linear transformations might already suspect that these cases differ significantly from each other.

Because of a need to identify the original sources or because of the importance of independence as an information theoretic concept, we might be interested in extracting the sources from the observed vector. One might also want to know the structure of the mixing matrix or separating matrix for describing the mixing process. The estimation of sources and/or the estimation of mixing or separating matrices is referred to as *independent component analysis* or *ICA.*

In this work we are primarily concerned with the estimation of mixing and separation matrices. Looking at equation (3) we can see that the observed random vector x_{obs} is a linear mixture of column vectors of matrix A, the coefficients of each vector being given by the corresponding independent source. These column vectors $a_i \in R^m$, $i = 1, \dots, n$ are called *ICA basis vectors.* Correspondingly we can see that in (4) each independent component is given by taking the inner product of the observed sample vector with a row vector of matrix W or — in other words — *filtering* the sample with one of the rows. For this reason the rows of matrix W are called *ICA filters.*

From equation (4) note that if we can find a transform W giving the independent components, then we can scale, negate and permute the independent sources with permutation matrix P and scaling matrix S to give us $s_{\text{PS}} = SPs = SPWx$, and the new components remain independent, so matrix SPW is also a separating matrix. This means that independent sources can only be recovered up to *scale*, *sign* and *permutation* [**13**]. In the rest of this work we assume that scaling is done so that each independent source has unit variance, that is, $\operatorname{var}\{s_i\} = 1$, $i = 1, \dots, n$.

4. Independence as a computable criterion

Assuming that we would want to extract the independent sources from their mixture, we would certainly like to create algorithms that can accomplish it. In general we want our procedures to be data driven, i.e., because we usually do not have a solid, useful mathematical description of quantities involved, we want to base our algorithms on information given in the form of data. For example in PCA, if one was to know all second order densities of vector components, the covariance matrix might be calculated explicitly. In practice, however, this is very seldom the case, so the covariance matrix has to be based on the given data. This is an estimation problem.

Example 2.6. If we have two sources, then equation (2) gives a mathematical description of independence, but it does not lend itself easily to forming computational procedures. Mathematically we might use some distance function d on the function space and try to minimize $d\left(f_{xy}, f_x f_y\right)$. But here we have the extremely difficult problem of finding density functions f_x, f_y and f_{xy}. Assume that we have two original random variables w and z and that $[x\ y]^T = A\,[w\ z]^T$, where A is a nonsingular linear transformation. Then the joint density of x and y can be expressed in terms of the joint density of w and z as [**48**]

$$f_{xy}\left(x, y\right) = \frac{1}{\left|\det\left(A\right)\right|} f_{wz}\left(A^{-1}\left[x\ y\right]^T\right).$$

But this requires knowledge of f_{wz}, which we can only get by estimation. We could use parametric estimation, if we had sufficient a priori knowledge, but as was noted above, this is not our approach. Nonparametric estimation of densities gives us non-explicit forms for estimates, and using these to optimize the distance with respect to A is extremely cumbersome and unstable. The dimensionality of real world problems is often high, as in our case, which causes many additional problems. All in all, this approach leads to insurmountable analytical and computational problems.

If we are to measure independence in an analytically tractable and algorithmically efficient way we have to look for more sophisticated ways of doing this. Fortunately computationally efficient criteria can be devised under suitable assumptions. But first we have to discuss an important preprocessing step.

5. Removing correlations

Assume that our data has zero mean, that is, $\mathrm{E}\left\{x\right\} = \bar{0}$. If we can find a linear transformation giving relation (4), the independent components of s have zero mean as well. In what follows, we assume that the data has this property, or that is has been *centered* by removing its mean.

The covariance between two components s_i and s_j of vector s, $i, j \in \{1, \ldots, m\}$ is

$$\begin{aligned} \mathrm{E}\left\{s_i s_j\right\} &= \operatorname{var}\left\{s_i\right\} = 1, \text{ if } i = j \\ \mathrm{E}\left\{s_i s_j\right\} &= \mathrm{E}\left\{s_i\right\}\mathrm{E}\left\{s_j\right\} = 0, \text{ when } i \neq j. \end{aligned}$$

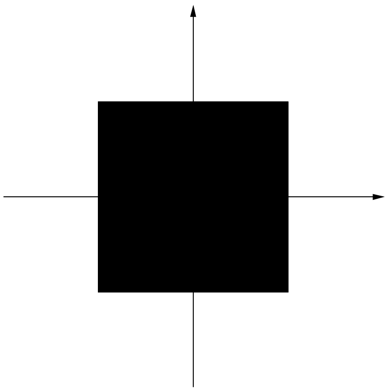
So the covariance matrix of s is $\operatorname{cov}\left\{s\right\} = I$, and components of s are uncorrelated. This means that independence implies uncorrelatedness in the case of zero mean variables, or that uncorrelatedness is a necessary requirement for independence. However, it is not a sufficient condition.

Example 2.7. The uniform distribution of two variables depicted in Figure 2(a) has been rotated by an orthogonal matrix in Figure 2(b). The original random variables are independent and uncorrelated. As we can see from the image, the rotated data is not independent, but it is uncorrelated, since uncorrelatedness is preserved under orthogonal transformations. Let x be an uncorrelated random vector. Then for an orthogonal matrix P

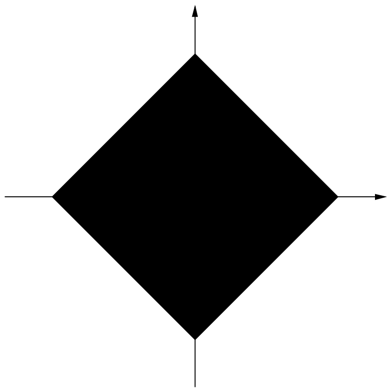
$$\operatorname{cov}\left\{Px\right\} = P \operatorname{cov}\left\{x\right\} P^T = I,$$

so Px is also uncorrelated.

By the previous discussion one might conjecture that if we first decorrelate the components of the observed variables, we shall be "closer" to our final objective of independence. We can accomplish uncorrelatedness by transforming x so that its covariance matrix will be diagonal. If in addition all components have unit variance



(a) Uniform distribution.



(b) Rotated distribution.

FIGURE 2. An example of independence and uncorrelatedness.

(the covariance matrix is unity), a random vector is referred to as being *white* and the process of accomplishing this property is called *whitening* or *sphering.*

Whitening can be done using PCA basis vectors and variances along them. From Section 1, let E denote the matrix of principal component basis vectors of random data vector x , i.e., the eigenvectors of $\operatorname{cov}\{x\}$, and $D = \operatorname{diag}(\xi_1, \ldots, \xi_m)$ a diagonal matrix of corresponding eigenvalues. The new whitened data vector v is given by

$$v = D^{-1/2} E^T x. \tag{5}$$

Matrix $V = D^{-1/2} E^T$ is a *whitening matrix.* The fact that v is really white can be seen from

$$\begin{aligned} \operatorname{cov}\{v\} &= \mathrm{E}\left\{D^{-1/2} E^T x x^T E D^{-1/2}\right\} = D^{-1/2} E^T \operatorname{cov}\{x\} E D^{-1/2} \\ &= D^{-1/2} E^T E D E^T E D^{-1/2} = I. \end{aligned}$$

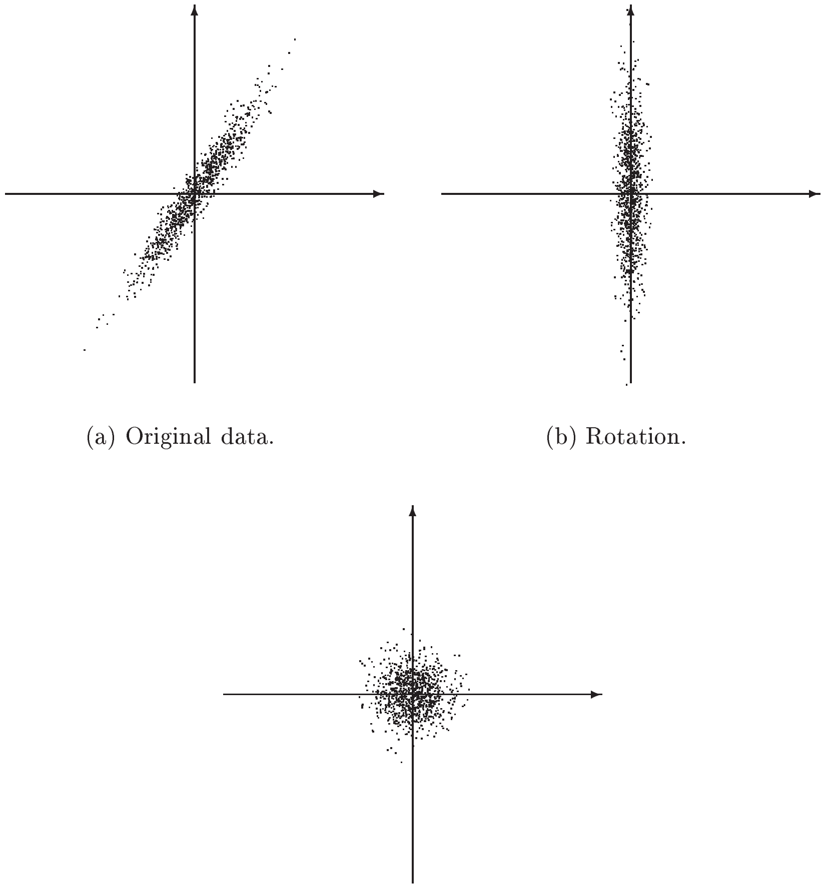
EXAMPLE 2.8. Consider the data set in Figure 3(a). When the data is projected onto its principal axes by multiplying it with matrix E^T, the result shown in Figure 3(b) is uncorrelated. Scaling with inverses of deviations onto the axes or multiplication by matrix $D^{-1/2}$ gives the whitened data of Figure 3(c). Here we can see where the name sphering has originated from.

PCA whitening is by no means the only possible method for whitening. If V is a whitening matrix of a random vector x and P is any orthogonal matrix, then PV is a whitening matrix, because

$$\mathrm{E}\left\{PVx\,(PVx)^T\right\} = P\,\mathrm{E}\left\{Vxx^TV^T\right\}P^T = PIP^T = I.$$

PCA whitening however has the advantage that optimal (in the mean square sense) dimensionality reduction can be combined with the whitening operation, as will be seen later on in Section 8.

In ICA problems whitening orthogonalizes the ICA basis vectors. Suppose that $n = m$, so that we have as many measured components as we have independent sources (we shall later see that when $n < m$, the situation can be reduced to the same form), and that the data is not restricted into a subspace of R^m (which is possible even



(a) Original data. (b) Rotation.

(c) Scaling.

FIGURE 3. Whitening.

if $n \geq m$). Then PCA whitening matrix $V = D^{-1/2}E^T$ exists and is invertible. Premultiplying equation (3) with V gives

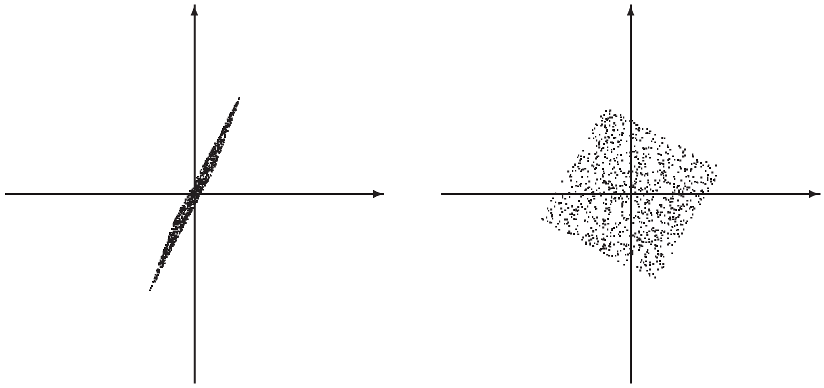
$$v = Vx = VAs = Bs. \tag{6}$$

This is the *whitened ICA problem* corresponding to problem (3). Vector v is called the *whitened data vector* and matrix $B \in \mathcal{L}(R^n, R^n)$ is the *whitened mixing matrix.* Noting that both v and s are white,

$$I = \mathrm{E}\left\{vv^T\right\} = \mathrm{E}\left\{Bss^TB^T\right\} = B\mathrm{E}\left\{ss^T\right\}B^T = BB^T,$$

so B is orthogonal.

EXAMPLE 2.9. We shall now examine our first ICA example. Consider the data set in Figure 4(a), which has been created by mixing data from a uniform two-dimensional distribution with a linear transform. In Figure 4(b) the data of Figure 4(a) has been whitened, and you can see that the independent directions are now orthogonal, or that the independent components can be recovered with a rotation of the coordinates.



(a) Mixed data. (b) Whitened data.

FIGURE 4. Orthogonalization of sources.

6. The central limit theorem and nonnormality

One way to approach the ICA problem is to try to form an optimization problem that has as its solutions the independent components. We shall call such objective functions *contrast functions.*

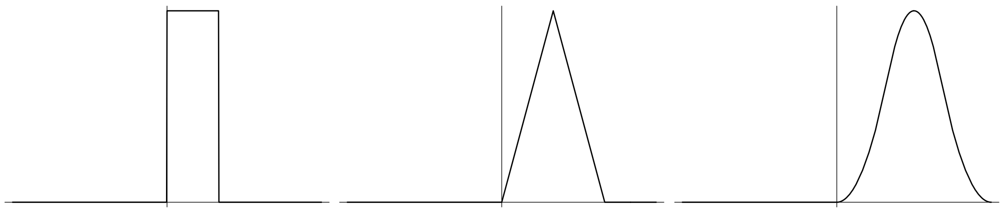
Let us approach the problem of finding contrast functions from the point of view of the central limit theorem. The central limit theorem [**48**] states that if we have n *independent* random variables $s_1, \ldots, s_n$, the distribution of their sum $y_n = \sum_{i=1}^{n} s_i$ approaches a normal distribution as $n \to \infty$. The "speed" with which the sum approaches a Gaussian is naturally dependent of the nature of the original variables. For smooth densities even relatively small values of n may give quite normal-like densities [**48**].

EXAMPLE 2.10. This effect is illustrated in Figure 5. The first density in Figure 5(a) is that of a uniformly distributed random variable. In Figures 5(b) and 5(c) the densities are those of a sum of two and three independent uniformly distributed variables, respectively. The density of the sum starts to approach the bell-shaped Gaussian density quite fast.

Assume that we have two nonnormal independent random variables s_1 and s_2 and $\beta \in R$, $0 \leq \beta \leq 1$. Knowing that the sum of independent random variables approaches a normal distribution, let us make the following imprecise conjecture: with a small but nonzero β the random variable $(1-\beta)s_1 + \beta s_2$ is "more normal" than s_1.

This is the intuitive idea behind contrast functions measuring deviation from a normal distribution. If the previous conjecture holds, then the original independent variables are *optimum points of nonnormality in all linear mixtures of the variables,* because if we start to mix one variable with any of the other variables we get something closer to normal. In other words, we could in principle solve the ICA problem by finding linear combinations or projection directions that give at least locally "maximally nonnormal" random variables.

In order to quantify these ideas we introduce a measure of nonnormality called *kurtosis.* Typically in statistics the value of kurtosis is described to measure the peakedness of the distribution, with peaked distributions giving positive values of kurtosis and flat

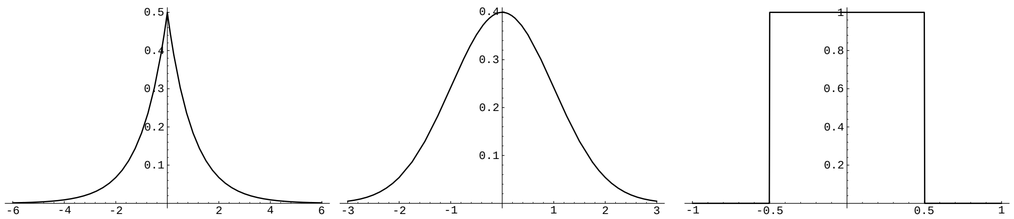


(a) A uniformly distributed variable.

(b) Sum of two uniformly distributed variables.

(c) Sum of three uniformly distributed variables.

FIGURE 5. An illustration of the central limit theorem.



(a) "Peaked" distribution.

(b) Gaussian distribution.

(c) "Flat" distribution.

FIGURE 6. Kurtosis as a descriptive statistic.

distributions negative values. Kurtosis is defined for a random variable y as[1]

$$\kappa_4(y) = \mathrm{E}\left\{y^4\right\} - 3\left(\mathrm{E}\left\{y^2\right\}\right)^2. \tag{7}$$

EXAMPLE 2.11. The plot of Figure 6(a) displays the density of a Laplacian random variable with $f_x(x) = \frac{1}{2}e^{-|x|}$. The kurtosis of the distribution is 12. The Gaussian distribution of Figure 6(b) has a kurtosis of zero. The flat uniform distribution, depicted in Figure 6(c), has a kurtosis of $-\frac{2}{15}$.

The value of kurtosis however depends on the variance of the distribution. Variance should not effect our measure of distance from normality, since it does not effect the flatness or peakedness of the density, and there exists Gaussian distributions of all variances. So in order to use kurtosis as an optimization criterion we should ensure that the optimization is done over mixtures with a fixed variance. The variance of a

[1] The reason for the κ_4 notation is explained in the next section.

projection of a random vector with mean zero is

$$\operatorname{var}\left\{w^T x\right\} = \operatorname{E}\left\{(w^T x)^2\right\} = w^T \operatorname{cov}\{x\} w.$$

This means that all vectors w should have a constant length in the norm induced by the positive definite matrix $\operatorname{cov}\{x\}$,

$$||w||_{\operatorname{cov}\{x\}} = w^T \operatorname{cov}\{x\} w.$$

An easy way to ensure that this is fixed for all w is to have $\operatorname{cov}\{x\} = I$ (use the Euclidean norm) and $||w|| = 1$ for all w (normalize the vectors in this norm). In other words

- x should be white (here we should note that we did not need the uncorrelatedness but rather the sphering property of whitened data) and
- we should keep $||w|| = 1$ while optimizing the function (optimize the function over the unit sphere).

Indeed using such constraints the optimization of kurtosis usually solves the ICA problem. Assuming that we have the whitened situation described above, from (6) we can see that the independent components are given by $s = B^T v$, or $s_i = b_i^T v$, $i = 1, \dots, n$, where $B = [b_1 \ \cdots \ b_n]$. The orthogonal columns of B span R^n, so if $w \in R^n$, then $w = \sum_{i=1}^n a_i b_i$, $a_i \in R$, $i = 1, \dots, n$. If s_1 and s_2 are two *independent* random variables and $\beta \in R$, then by direct calculation the following very important properties hold for kurtosis

$$\begin{aligned} \kappa_4(s_1 + s_2) &= \kappa_4(s_1) + \kappa_4(s_2) \\ \kappa_4(\beta s_1) &= \beta^4 \kappa_4(s_1). \end{aligned}$$

Define $k_i = \kappa_4(s_i)$, $i = 1, \dots, n$, and $a = [a_1 \ \cdots \ a_n]^T$. Then

$$\kappa_4\left(w^T v\right) = \kappa_4\left(a^T B^T v\right) = \sum_{i=1}^n \kappa_4(a_i s_i) = \sum_{i=1}^n a_i^4 k_i. \tag{8}$$

For kurtosis to be a contrast function it would have to be optimized at one of the columns of B. In Appendix A it is proven that

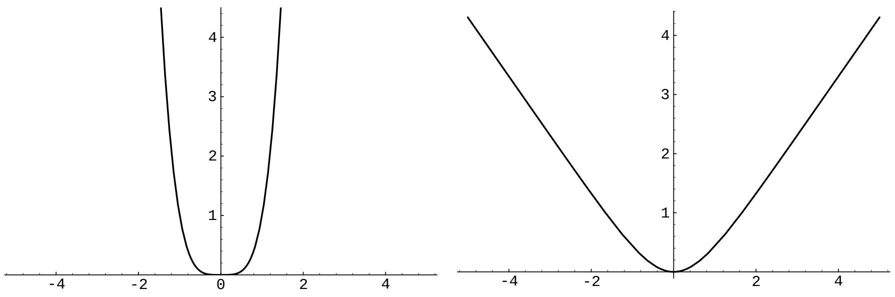
$$f(a) = \sum_{i=1}^n a_i^4 k_i$$

fulfills this condition, that is, the optima of f are given by such unit vectors a in which exactly one component differs from 0 and is 1 or -1. This proves that kurtosis is indeed a contrast.

Although kurtosis can be used to solve the ICA problem, it has some undesirable properties. From the definition of kurtosis (7) we can see that it is very sensitive to variations in data since it is based on the fourth moment. This makes the approach sensitive to errors and outliers, because the algorithms have to estimate the statistical quantities involved. That is why we sometimes use other contrast functions to solve the ICA problem.

The following class of contrast functions is introduced in [**25**]. Let G be a twice differentiable nonquadratic function, x a zero mean random variable and ν a normally distributed random variable with the same mean and variance as x. Then a contrast function based on G is

$$J_G(x) = \operatorname{E}\{G(x)\} - \operatorname{E}\{G(\nu)\}. \tag{9}$$



(a) $G(t) = t^4$. (b) $G(t) = \ln\cosh(t)$

FIGURE 7. Two different nonlinearities for a contrast function.

First, G is constrained to be nonquadratic because otherwise J_G would be identically zero. Second, we can immediately see that J_G is zero when x is Gaussian. Note that if $G(t) = t^4$, (9) is reduced to the definition of kurtosis. It is shown in [**25**] that J_G is at least locally a contrast function for whitened data (the independent directions are local optima of J_G). One especially important choice for G is $G(t) = \ln\cosh(t)$. It is much less sensitive to errors in the data. This is illustrated in Figure 7, where the fast growth of the fourth power is contrasted with the nearly linear behavior of $\ln\cosh$. It is shown in [**28**] that this contrast has good behavior with respect to asymptotic variance and robustness.

7. Cumulants and cumulant matrices

Cumulants are higher-order statistics that have become increasingly popular in various signal processing tasks [**39, 42**]. The use of cumulants is usually contrasted with that of second-order statistics: whereas correlation and power spectrum (Fourier transform of autocorrelation) are standard tools in signal processing, cumulants and polyspectra (Fourier transforms of cumulants) can be seen as corresponding statistical tools of order higher than two.

The properties of cumulants and advantages for using them are not easy to illustrate. Here we will first justify superficially the use of cumulants by giving a few of their properties even before we give their definition. Let $\mathrm{cum}(x_1, \dots, x_n)$ denote a cumulant of random variables $x_1, \dots, x_n$.

1. Cumulants are linear in each entry (*multilinear*) , that is, for a random variable y and scalar α

$$\begin{aligned} \mathrm{cum}(x_1, \dots, x_i + y, \dots, x_n) &= \mathrm{cum}(x_1, \dots, x_i, \dots, x_n) \\ &\quad + \mathrm{cum}(x_1, \dots, y, \dots, x_n) \\ \mathrm{cum}(x_1, \dots, \alpha x_i, \dots, x_n) &= \alpha\, \mathrm{cum}(x_1, \dots, x_i, \dots, x_n) \end{aligned}$$

2. If a subset of random variables $x_1, \dots, x_n$ is independent of the others, then $\mathrm{cum}(x_1, \dots, x_n) = 0$.

3. If $x_1, \ldots, x_n$ are Gaussian random variables and $n \geq 3$, then $\mathrm{cum}\,(x_1, \ldots, x_n) = 0$.

The first and second property enable and ease the use of cumulants as operators. The third property makes cumulants "immune" to Gaussian noise and enables their use as measures of nonnormality. For a comprehensive presentation of these and other properties of cumulants see [**39, 42**].

Cumulants are defined as the coefficients of the Taylor series of the second characteristic function of $x = [x_1 \ \cdots \ x_n]^T$ at 0, that is,

$$\mathrm{cum}\,(x_1, \ldots, x_n) = \frac{d}{d\omega_1} \cdots \frac{d}{d\omega_n} \mathbf{\Psi}\,(\omega_1, \ldots, \omega_n)\Big|_{\omega_1=0,\ldots,\omega_n=0},$$

where

$$\mathbf{\Psi}\,(\omega_1, \ldots, \omega_n) = \ln \mathrm{E}\left\{e^{\sum_i \omega_i x_i}\right\}.$$

The class of cumulants involving just one random variable is denoted by the letter κ, so that

$$\kappa_{\mathrm{i}}(x) = \mathrm{cum}(\underbrace{x, \ldots, x}_{i \text{ times}}).$$

Example 2.12. Kurtosis discussed in the previous section is in fact a cumulant, namely the fourth order cumulant $\kappa_4\,(x)$. It can be seen that the properties of kurtosis are just special cases of the more general cumulant properties introduced above.

One concept which has been influential in ICA is that of *cumulant matrices* [**9**]. A cumulant matrix $Q_x(M)$ is defined for a real n-dimensional random vector x and an $n \times n$ matrix M as

$$Q_x(M)_{ij} = \sum_{k=1}^{n} \sum_{l=1}^{n} \mathrm{cum}\,(x_i, x_j, x_k, x_l)\, m_{lk},\ \ i, j \in \{1, \ldots, n\}.$$

We shall denote the set of all cumulant matrices of random vector x by $\mathcal{C}_x$. The usefulness of cumulant matrices stems from the fact that for a whitened random variable v fulfilling the ICA model (6) we have

$$Q_v(M) = B \Lambda_M B^T,$$

where Λ_M is a diagonal matrix

$$\Lambda_M = \mathrm{diag}\left(\kappa_4\,(s_1)\, b_1^T M b_1, \ldots, \kappa_4\,(s_n)\, b_n^T M b_n\right).$$

This means that columns of matrix B are eigenvectors of matrix $Q_v(M)$ for *any matrix* M. Letting $\beta_i = \kappa_4\,(s_i)\, b_i^T M b_i$, $i = 1, \ldots, n$, we can see that as long as $Q_v(M)$ does not have identical eigenvalues, i.e., $\beta_i \neq \beta_j$ for $i \neq j$, columns of matrix B can be determined up to a scalar multiplier by solving the eigenvalue problem. Then the eigenspaces are one dimensional, and the indeterminacy in the result is identical to the sign/magnitude indetermination in ICA.

The approach is computationally simple but has poor statistical performance [**9**]. Below we shall see and use a variant of this method that works by diagonalizing many cumulant matrices simultaneously, thus trying to improve the performance.

8. Over- and underdetermined ICA models

Assume that matrix B has been determined by some ICA algorithm, and that we want to determine the mixing matrix A and the separating matrix W. If $n = m$, then we also know that V in (6) is invertible, and ICA basis vectors and filters are given by $A = V^{-1}B$ and $W = B^T V$. The situation gets more complicated when $n \neq m$.

If the number of independent sources is smaller than the dimension of measured data, i.e., $n < m$, we can use PCA to reduce our problem to a lower dimension. From (3) we note that if $n < m$, then the measured data lies in an at most n dimensional subspace spanned by columns of A. Hence $n-m$ PCA dimensions contain no data, and PCA can be used to constrain the data to a subspace without any loss of information. Naturally in real world problems data contains noise, so the data is not contained exactly in the subspace, but PCA is a good tool for such situations as well.

Let x_{obs} denote the observed data vector and D be the diagonal matrix of n largest eigenvalues of $\operatorname{cov}\{x_{\text{obs}}\}$ and E the matrix of corresponding eigenvectors. So $D \in \mathcal{L}(R^n, R^n)$ and $E \in \mathcal{L}(R^n, R^m)$. Then dimensionality reduction is performed with

$$x = EE^T x_{\text{obs}}. \tag{10}$$

When PCA whitening is used, equation (6) gives $v = D^{-1/2}E^T EE^T x_{\text{obs}} = D^{-1/2}E^T x_{\text{obs}}$, since columns of E are orthogonal ($E^T E = I$).

After dimensionality reduction matrix B is calculated for the reduced system. However, usually we are interested in original basis vectors of equation (3) or filters of (4), so B is not our final aim. Matrix A has to be solved from equation

$$VA = B \tag{11}$$

obtained from (6). This equation can be viewed as consisting of n equations

$$Va_i = b_i, \quad i = 1, \dots, n, \tag{12}$$

where a_i and b_i are columns of A and B. This is equivalent to solving

$$E^T a_i = D^{1/2} b_i, \quad i = 1, \dots, n.$$

If $n < m$, that is, the number of sources is smaller than the number of observed components, then since $E^T \in \mathcal{L}(R^m, R^n)$, the equations are underdetermined and each problem has an infinite number of solutions. However, multiplication performed in operation VA can be viewed as whitening of the column vectors of A. Hence the solution of (11) can be viewed as dewhitening of column vectors of B. Since whitening using PCA consists of projection onto PCA subspaces and scaling with corresponding deviation, the "obvious" counterpart for dewhitening is rescaling and summation back into R^m, which gives us

$$A_{\text{norm}} = ED^{1/2}B. \tag{13}$$

Note that A_{norm} gives one solution to problem (11).

Equation (13) can also be derived in another way. If $a_{\text{part},i}$ is one solution to problem (12), then any solution a_i can be given in form

$$a_i = a_{\text{part},i} + a_{\text{null}},$$

where $a_{\text{null}} \in \mathcal{N}(V) = \{a \mid Va = 0\}$. One particular choice for $a_{\text{part},i}$ is the one with minimum Euclidean norm, given by **[37]**

$$\begin{aligned} a_{\text{norm},i} &= V^T(VV^T)^{-1}b_i = (D^{-1/2}E^T)^T[D^{-1/2}E^T(D^{-1/2}E^T)^T]^{-1}b_i \\ &= ED^{1/2}b_i, \end{aligned}$$

which gives us (13). Solution $a_{\text{norm},i}$ also has the property that it is orthogonal to all vectors in $\mathcal{N}(V)$, which might be described so, that it does not contain any information from this uncertain part of the solution.

ICA basis vectors are closely related to ICA filters. After whitening we have from (6)

$$s = B^{-1}Vx = B^T Vx,$$

since B is orthogonal. Using (5) we get

$$W = B^T D^{-1/2} E^T. \tag{14}$$

On the other hand, from (13) we can see that

$$A_{\text{norm}}^T = B^T D^{1/2} E^T.$$

It should be noted that ICA filters of (14) automatically perform dimensionality reduction when applied to the original data vector x_{obs}, because

$$\begin{aligned} Wx_{\text{obs}} &= B^T D^{-1/2} E^T x_{\text{obs}} = B^T D^{-1/2} E^T E E^T x_{\text{obs}} \\ &= B^T D^{-1/2} E^T x = Wx. \end{aligned}$$

When the number of sources is greater than the number of measured components ($n > m$), matrix B can not in general be solved, but can only be determined in special cases [**7**]. Assuming that we have found B, the methods for determining A are as above, but the determination of W is more complicated now. From (6) we get $Vx = BWx$, so W can be calculated by solving $BW = V$, but in this case B is not orthogonal or even square. This set of equations has again an infinite number of solutions, and like above the solution with minimum norm is chosen, giving

$$W = B^T(BB^T)V.$$

To summarize the results, in the case $n = m$, both A and W may be resolved accurately. In the case $n > m$, A may be solved accurately, but W contains uncertainty, because it has to be chosen from a set of possible solutions. The situation is reversed when $n < m$. The degree of freedom for the choice of the uncertain solution in both cases is $|n - m|$.

CHAPTER 3

Connections with other disciplines

1. Overview

Redundancy and its reduction are central areas of research in many fields of information processing. The theory of such concepts is handled in information theory. Practical approaches to redundancy reduction are needed for example in pattern recognition and data compression.

Here we will discuss some of the related areas that are relevant to this work. These are information theory, sparse coding, wavelets and some basic concepts of sensory coding in biology.

One very closely related area that is certainly worth mentioning but has not been given a full section of its own is *projection pursuit.* It is a branch of statistics in which the objective is to find meaningful and interesting projections of high dimensional data into lower dimensions [**30**]. Typically measures of nonnormality and information theoretic measures are used to quantify what is interesting. This is an important idea from our point of view, because if and when the ICA model does not hold for image data, this approach may give useful interpretations of the results.

2. Information theory

2.1. General. Information theory (for example, [**38, 48**]) is concerned with the study of information and uncertainty in probabilistic systems. This branch of mathematics is central to the study of representation and transmission of signals containing information. Measuring the degree of independence or interdependence is one of the basic problems in information theory, so the concepts and ideas of information theory are widely used in the context of ICA.

One of the fundamental concepts in information theory is *differential entropy,* which is defined for a continuous random variable x as

$$H(x) = -\int_{-\infty}^{\infty} f_x(x) \log f_x(x)\ dx.$$

The interpretation of this quantity must not be confused with that of ordinary entropy, which is defined for a discrete random variable x_d as

$$H(x_d) = -\sum_i p_i \log p_i,$$

where $p_i = \mathrm{P}(x_d = x_i)$, and i ranges over the set of indices of possible values of x_d. Whereas ordinary entropy measures the uncertainty or information contained in a random variable, differential entropy — as its name implies — is used only to measure changes in uncertainty or information [**48**]. *Conditional differential entropy* is defined as

$$H(x|y) = -\int_{-\infty}^{\infty} f_y(y) \int_{-\infty}^{\infty} f_x(x|y) \log f_x(x|y)\ dx\, dy,$$

and it is used to measure changes in uncertainty about x after y has been observed.

To relate information contents between random variables, we introduce *mutual information.* If x and y are random variables, then their mutual information is

$$\begin{aligned} I(x,y) &= H(x) + H(y) - H(x,y) && (15) \\ &= H(x) - H(x|y) = H(y) - H(y|x). && (16) \end{aligned}$$

Intuitively this is interpreted as the amount of uncertainty that is removed from one of two random variables as the other one is observed. Another form is given by

$$I(x,y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f_{xy}\left(x,y\right) \log \frac{f_{xy}\left(x,y\right)}{f_x\left(x\right) f_y\left(y\right)}\, dx\, dy,$$

which is also called the *Kullback-Leibler distance* between f_{xy} and $f_x f_y$. This measure is always nonnegative and is zero if and only if $f_{xy} = f_x f_y$.

The previous concepts can be generalized for random vector $x = [x_1 \ \cdots \ x_n]^T$. The differential entropy of x is

$$H(x) = -\int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} f_{x_1 \cdots x_n}(x_1, \ldots, x_n) \log f_{x_1 \cdots x_n}(x_1, \ldots, x_n) dx_1 \cdots dx_n.$$

Guided by (16) we define the mutual information of x to be

$$\begin{aligned} I(x) &= \sum_{i=1}^{n} H(x_i) - H(x_1, \ldots, x_n) \\ &= \int_{-\infty}^{\infty} f_x\left(x\right) \log \frac{f_x\left(x\right)}{\prod_{i=1}^{n} f_{x_i}\left(x_i\right)}\, dx, \end{aligned}$$

where $\int_{-\infty}^{\infty} dx$ means integration with respect to all components.

From the definition and properties of the Kullback-Leibler distance one can immediately deduce that mutual information between random variables is zero if and only if they are independent. So analytically mutual information is a good criterion for ICA. Unfortunately, as with the original definition of independence, the need to approximate the densities involved makes the approach difficult to apply in practice.

One more concept, which has been useful for providing additional insights into ICA is *negentropy.* It can also be seen as a measure of deviation from normal distribution. Let x be a random variable and x_G a Gaussian random variable with the same mean and variance as x (these fix the distribution of x_G). Then the negentropy of x is

$$J(x) = H(x_G) - H(x).$$

It is always nonnegative, since in a class of distributions with fixed mean and variance the random variable with a Gaussian distribution has the largest differential entropy [**24**].

Without getting into the details here, we state that for a random vector x of uncorrelated components mutual information can be expressed as [**13**]

$$I(x) = J(x) - \sum_{i=1}^{n} J(x_i). \tag{17}$$

Remember that whitened data and its orthogonal transformations are uncorrelated. Negentropy is invariant under linear invertible transformations [**13**], a property which is very useful here and not enjoyed by entropy. This property is useful because it implies that when in ICA we want to minimize (17), then since $J(x)$ is constant, we can actually maximize the sum of negentropies of individual components. We can even optimize individual negentropies directly, since these are not interrelated.

2.2. Approximating negentropy. So far the information-theoretic approach to ICA presented here has been quite exact and analytic. Unfortunately, in order to develop the ideas further we have to resort to somewhat crude approximations — or at least this is what has happened in research in this area so far. The actual calculations are quite sophisticated, and since we are only outlining the theory and its relations with ICA, they will not be presented here. For details, see [**13, 30**].

For example, approximating the density function of a nearly normal distribution using a polynomial expansion results in [**30, 26**]

$$J(x) \approx a\left(\left(\kappa_3\left(x\right)\right)^2 + \frac{\left(\kappa_4\left(x\right)\right)^2}{4}\right) + b,$$

where $a, b \in R$ are constants and $a > 0$. If the distribution of x is symmetric, then the third cumulant $\kappa_3\left(x\right)$ or *skewness* of the distribution is zero, and we can optimize directly the square of kurtosis. This result is very similar to the one obtained earlier in the context of ICA.

Jones and Sibson note in their article [**30**] that "this is a largely formal rather than analytical argument: the 'approximation' will be used in cases as non-normal as possible, so the real question is whether the balance between $\kappa_3\left(x\right)^2$ and $\kappa_4\left(x\right)^2$ terms which it suggests is effective in practice." Actually this is not just a question of balance — we may question whether it is reasonable to use just these cumulants for the description of the distribution. The authors claim that one does not often find real-world data in which the third and fourth cumulants were small and higher cumulants larger. In the same paper Marriott notes that for multimodal distributions this is not rare — the same property is pointed out in [**26**].

A similar procedure to the one described above is performed in [**13**]. There another expansion is used for the density function, yielding

$$J(x) \approx \frac{1}{12}\left(\kappa_3\left(x\right)\right)^2 + \frac{1}{48}\left(\kappa_4\left(x\right)\right)^2 + \frac{7}{48}\left(\kappa_3\left(x\right)\right)^3 - \frac{1}{8}\left(\kappa_3\left(x\right)\right)^2 \kappa_4\left(x\right).$$

For a symmetric distribution this is again reduced to optimization of the square of kurtosis.

2.3. Entropy maximization. This section introduces the basic principle behind one method connecting information theory, ICA and neural networks. This method is the maximization of entropy in a two-layer neural network. The structure of the network is presented in Figure 1. Here the inputs are mapped linearly into a hidden layer, and each output of the network is determined by applying a nonlinear function to the corresponding output of the hidden layer. These nonlinear functions are constrained to be bounded.

Intuitively the idea of the method is to map the set of inputs into outputs that are uniformly distributed and hence have maximum entropy or uncertainty. If the input consists of a random vector that is a linear mixture of some underlying independent sources, this can be accomplished by inverting the linear process in the linear layer and then mapping each component into the output nonlinearly by its distribution function. This is why such a network can be used to perform ICA in the middle layer.

EXAMPLE 3.1. If the random variable following a Gaussian distribution like in Figure 2(a) is mapped with its distribution function of Figure 2(b) the result is a uniform distribution.

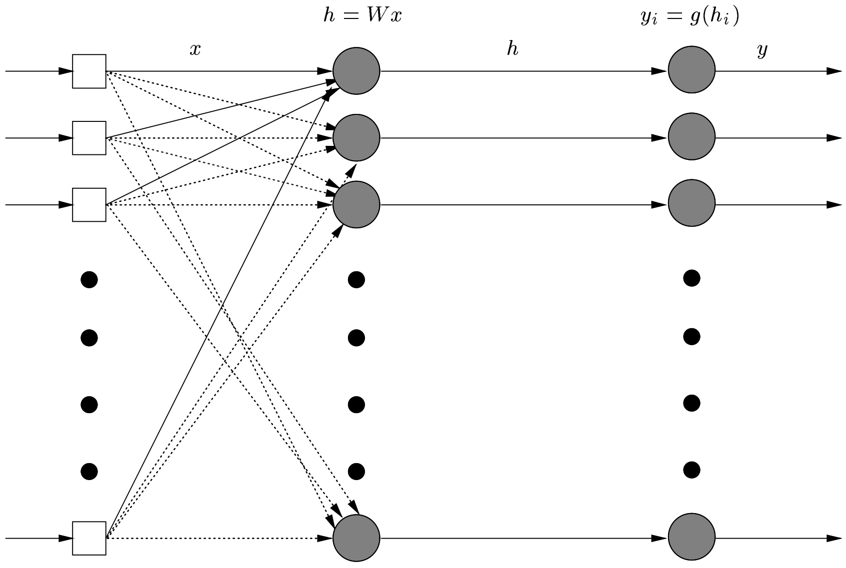
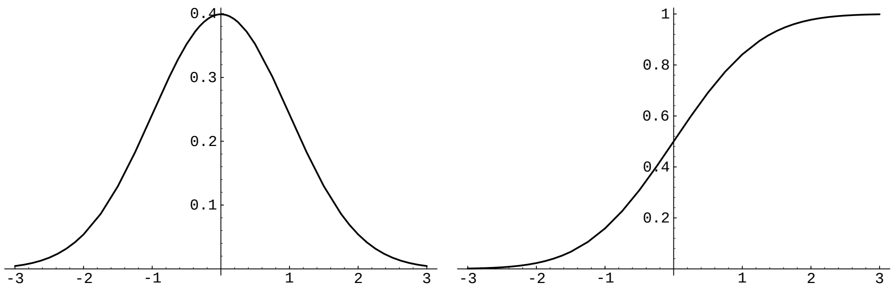


FIGURE 1. Entropy maximization network.



(a) Gaussian density function. (b) Gaussian distribution function.

FIGURE 2. Mapping a random variable with its distribution function.

Intuitively at areas with high probability masses the mapping function has a high derivative, so it spreads the data into a larger area.

Let us denote the outputs of the linear layer by $h = [h_1 \cdots h_n]^T$ and the corresponding nonlinear transfer functions by g_i, $i = 1, \dots, n$. It turns out [**41**] that when there is no noise in the system the maximization of the entropy of the outputs is equivalent to the minimization of the mutual information or Kullback-Leibler distance between f_h and $\prod_{i=1}^n g_i'$, that is

$$d\left(f_h, \prod_{i=1}^n g_i'\right) = \int_{-\infty}^{\infty} f_h(h) \log \frac{f_h(h)}{\prod_{i=1}^n g_i'(h_i)}\, dh.$$

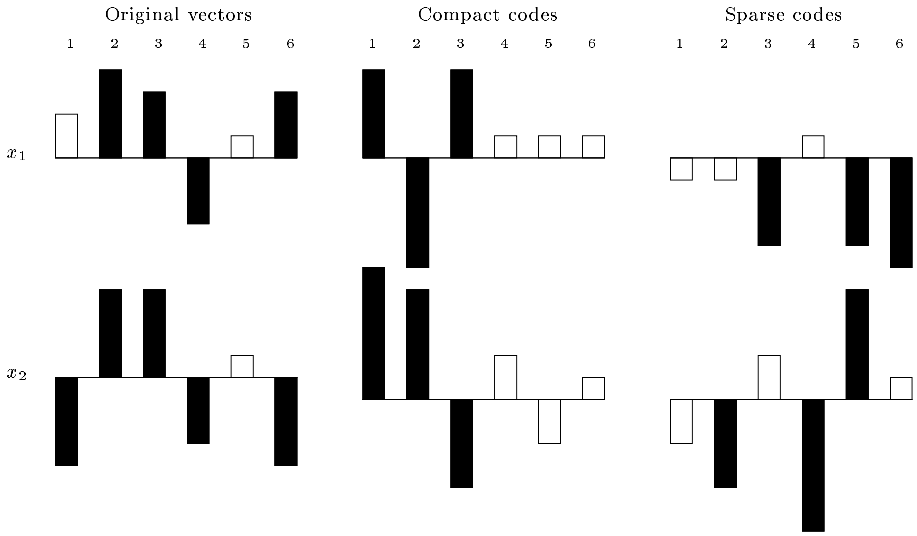


FIGURE 3. Sparse codes and compact codes for two vectors x_1 and x_2.

This distance between the two functions is definitely at a minimum if $f_h(h) = \prod_{i=1}^{n} g_i'(h_i)$, in which case h is said to be *factorial.* This is equivalent to the components of h being independent. (This can be seen easily by integrating out $n-1$ variables, resulting in $f_{h_i}(h_i) = g_i'(h_i)$.) So if the linear layer can separate the independent components, the overall entropy is maximized by selecting the nonlinear function to be the cumulative distribution function of the corresponding variable.

3. Sparse coding

In Section 1 of Chapter 2 we saw that we can exploit the redundancy present in the data to compress it to a lower dimensional subspace. This is accomplished efficiently in PCA by removing linear statistical dependencies, i.e., correlations, from the data and dropping out the subspaces with smallest variances. In other words, we find and remove subspaces in which the data vectors have small values for the most of the time, and preserve subspaces that contain large values for the most of the time. Such a transformation of the data is called *compact coding* [**17**].

Instead of concentrating the redundancy into just a few "disposable" subspaces we can use it in another way by distributing it evenly between all components. Whereas in the previous approach we always dropped out or zeroed the same components with low variance, in this method the zeroed components vary. Put it another way, the number of components that have significant activity levels is small, like in PCA, but the components that are active vary from one coded vector to another. This is called *sparse coding* [**17**].

EXAMPLE 3.2. The difference between compact coding and sparse coding is illustrated in Figure 3, where two example vectors of a larger data set have been coded using these methods. Both the compact code and sparse code use the redundancy of the data to increase the number of "insignificant components" (white bars). In compact coding these components are at the same positions, but in sparse coding the positions vary.

Examining one single component of a very sparse code, the component typically has a lot of insignificant values near zero and fewer large values. The values of the component are small for the most of the time, so the distribution is *peaked.* Remembering from Section 2.6 that kurtosis can be used to measure the peakedness of the distribution, we have a connection with ICA. One way to find sparse codes is to maximize kurtosis to create very peaked distributions [**17**]. In fact, if we have a linear mixture of positively kurtotic or sparse source variables, finding the maxima of kurtosis solves both ICA and the sparse coding problem simultaneously.

There are also significant differences between ICA and sparse coding. Negatively kurtotic directions are useless in sparse coding, whereas in ICA they may represent sources. ICA can only be efficiently solved when the measurements are a linear mixture of independent sources — at least at the present time — but some form of sparse coding can be performed for any data set.

Maximization of kurtosis is just one way to accomplish sparse coding. One of the works that inspired this thesis was [**45**], in which the authors examined sparse coding of natural images. Relating their work to PCA, from Section 1 of Chapter 2 we know that PCA uses a linear model (linear operations) in both finding the coefficients in the new basis and reconstructing the data in the original basis. In [**45**] Olshausen and Field use a nonlinear method to select the basis and the coefficients, but the reconstruction is linear as in PCA. The basis and coefficients are chosen so that the reconstruction error is as small as possible, but the resulting code is sparse.

To be more precise, let us denote the set of vectors to be "sparsified" by $\mathcal{X}$. Then first the basis of the linear reconstruction $\{y_1, \dots, y_n\}$ (in the case of images this is a set of matrices) is chosen by minimizing function

$$f(y_1, \dots, y_n) = \sum_{x \in \mathcal{X}} \min_a \Bigg(\underbrace{\Bigg\| x - \sum_{i=1}^{n} a_i y_i \Bigg\|^2}_{\text{reconstruction}} + \rho \underbrace{\sum_{i=1}^{n} S_\sigma(a_i)}_{\text{sparseness}} \Bigg). \tag{18}$$

Here $a = [a_1 \ \cdots \ a_n]^T$ is a vector containing the actual codes for each x. In the term minimized for each sample vector x the first term is a reconstruction error, and the second term is a penalty term of sparseness, with $\rho \in R$ being a penalty coefficient and S_σ a scaled measure of sparseness. Some possible forms for S_σ are $S_\sigma(t) = -e^{-(t/\sigma)^2}$, $S_\sigma(t) = \ln(1 + \left(\frac{t}{\sigma}\right)^2)$, and $S_\sigma(t) = \left|\frac{t}{\sigma}\right|$ [**45**]. After the basis has been chosen, the coefficients of a vector x in the new basis are chosen by performing just the inner optimization in (18). The actual optimizations employ the gradient descent method.

4. Wavelets

A fundamental problem in signal processing is to find a suitable representation for a signal. Usually different signal representations are based on linear transformations of the signals onto different bases. For example, one way to represent a time signal is to consider its Fourier transformation, which can be seen as projecting the signal onto a basis of functions. Each basis function consists of just one frequency.

Whereas the usual signal representation is well localized in time — or in the case of images, in space — the Fourier representation of the signal is well localized in frequency. It has been observed that for signals with time varying spectra, i.e., signals in which the frequency contents do not stay constant over time, a representation of the signal localized both in time and in frequency would be very useful for many purposes. One method used to solve the problem is the short time Fourier transformation (STFT), where we restrict our attention around a certain time point of the original signal by multiplying it with a window function (e.g., a Gaussian function) and Fourier transforming this modified signal. The problem with STFT is the constant window size, which is difficult to choose and fixed for all frequencies [**52**].

Wavelets are one method developed to solve this problem [**14**]. Wavelets are families of basis functions, each family being generated by scaling and translating a 'model function' called *mother wavelet.* From a mathematical point of view this restricted form of functions is an essential property because it makes the theoretical analysis feasible. But it may not be crucial from the point of view of applications. An important property of wavelets is that they are localized both in time and frequency [**52**]. This is also typical of STFT, but the localization of wavelets is specialized so that wavelets that respond to low frequencies are more frequency selective, i.e., localized in the frequency domain, but also more spread in time than wavelets which respond to high frequencies. So wavelets have variable window size, and the window size is connected to the frequency response of the wavelet. These qualities will be the basis of our examination of the similarities between ICA and wavelets.

Wavelets represent a compromise between good time resolution and good frequency resolution, a trade-off being forced by the linear approach [**52**] (for other, nonlinear methods see [**12**]). Some applications in which they have proven to be useful are feature detection, compression, noise removal, computer vision and graphics, and time-frequency description of signals [**50**].

In order to mathematically analyze the connections between ICA and the previously mentioned properties of wavelets we introduce some useful concepts from time-frequency analysis [**12**] and generalize these to two dimensions. The techniques we shall use are called *time and frequency distributions.* Consider the energy of a time signal at each moment, that is, the square of the signal. Assume that we normalize the total energy of a finite energy signal to be 1. Then we can consider this energy function to be an *energy density* [**12**]. This density can be used in a similar manner as a probability density to calculate properties of the signal. For example for a 1-D signal $s(t)$ which has been normalized to have an energy of 1, the *mean time* of the signal is calculated as $\langle t \rangle = \int t \, |s(t)|^2 \, dt$. The *duration* of the signal — or variance in time — is calculated as $\left\langle (t - \langle t \rangle)^2 \right\rangle$. Similar concepts can be defined in the frequency domain, giving us the *mean frequency* and *bandwidth* (or variance in frequency).

Now consider generalizing these definitions for image data. Then the energy densities in both spatial and frequency domain are functions of two variables — in the spatial domain we have the image function $s(x, y)$ and in the frequency domain the corresponding cosine transform $S(u, v)$ — so the densities are similar to joint probability densities of two random variables. The *mean location* and mean frequency of an image are calculated as for usual random variables. The result is a two-dimensional vector. To calculate the spatial variance, which we shall call *spread*, we add the variances of x- and y-directions together. The bandwidth of the image is defined similarly. These concepts will be used in the experimental part of the thesis to analyze the connections between the two methods.

Wavelets are also closely related to sparse coding. From the point of view of sparse coding, the use of wavelets in data compression has been successful because their outputs are sparse. On the other hand, wavelet-like bases emerge as a result of applying a sparse coding method to natural images [**45**].

5. Biological considerations

There has been an enormous amount of research on the human visual system and sensory coding. In the scope of this thesis it is only possible to briefly mention some concepts and results that are relevant to our research. The short and greatly oversimplified discussion presented here is based on material from [**32**], [**17**] and [**18**].

Even the low level parts of the mammalian visual system are extremely complex, but for practical purposes we shall now describe crudely their properties with just a few concepts. First of all, the system consists of different *layers,* each layer containing *neurons* that have some common task. The first layer is the *retina,* which collects the signals from the outermost visual sensors, *photoreceptors.* Each neuron in the retina has a corresponding *receptive field,* which is an area in the set of photoreceptors from which the neuron receives its incoming connections. The second layer in the system is the *primary visual cortex,* which in receives has its input connections from retina (in reality there are some layers between these two).

The functionality of different layers has been examined with the help of *activity patterns.* An activity pattern of a neuron is a (visual) stimulus to which the neuron responds by activating its output. It has been found out, that neurons in the retina respond to "on-center off-surround" or "off-center on-surround" kinds of responses, that is, responses that correspond to white spots on dark background or the other way around. Interestingly enough, some studies have claimed that the major functions of the retina are decorrelation of the input information and suppression of noise [**2**]. From our point of view this is of course interesting since whitening is a major preprocessing step in many ICA algorithms.

However, we are more interested in the combination of the first two layers of our idealized visual system — the retina and the primary visual cortex. This is of major concern to us, because it has for a long time been suggested that a major function of these layers is to reduce statistical redundancy in input. This is ICA in the larger sense of the concept: producing outputs that are as independent as possible.

The activity patterns of the primary visual cortex are meaningful from an image processing point of view, because they can be thought to represent *features* of natural images, and so the neurons may be seen as *feature detectors.* The neurons of this layer respond to various basic image structures like edges, lines and bars. These have been used for obvious reasons in various image processing tasks, but now it may be possible to provide a statistical or information theoretic justification for their use, with possible enhancements of the existing methods.

Some words of warning must be given at this point. This classification of neurons as feature detectors and the corresponding statistical and information theoretical justifications have been rightly criticized in [**18**] for the following reasons.

- The neurons in the visual system do respond to a larger collection of stimuli than just one feature.
- The current experimental method of "stimulus-response" for determining the activity patterns of neurons cannot be used to determine the responses to all possible stimuli.
- The effect of evolutionary processes has not been taken adequately into account while examining the resulting visual systems.

Regardless of this criticism, the results of applying ICA to image data should be evaluated with respect to our knowledge of the visual system — with all its connections to statistical independence and image processing — even if this knowledge is somewhat inaccurate and partial.

CHAPTER 4

ICA algorithms

1. Overview

There exists a wide variety of ICA algorithms. The objective of this work is to examine the union of ICA and image information. Because the number of different experiments, needed implementations and combinations of image data and algorithms grows fast with the number of used algorithms, a relatively small set of algorithms will have to be chosen.

To honor those algorithms which are not described or used here we indicate their creators and the major references to them. The author is sorry for any omissions.

- Jutten and Herault [**31**]
- Oja, Ogawa and Wangviwattana [**44**], and Karhunen and Joutsensalo [**33**]
- Comon [**13**]
- Cichocki, Kasprzak and Amari [**10**]
- Delfosse and Loubaton [**16**]
- Deco and Brauer [**15**]
- Moreau and Macchi [**40**]
- Pajunen, Hyvärinen and Karhunen [**47**]
- Karhunen and Pajunen [**35**]
- Amari, Cichocki and Yang [**1**]
- Cardoso and Laheld [**8**]
- Cichocki and Unbehauen [**11**]
- Karhunen, Oja, Wang, Vigário and Joutsensalo [**34**]
- Karhunen and Pajunen [**36**]
- Girolami and Fyfe [**20, 21, 23, 22**]
- Pearlmutter and Parra [**49**]

The justifications for the selection of algorithms used in these experiments are depicted in Table 1. There the different types of algorithms that we have selected to use are classified with respect to two dimensions: the mathematical approach of solution and the actual algorithmic method of implementing the solution.

- The fixed point algorithms (Section 3) are efficient iterative methods that search for the stationary points of the Lagrangian of the optimization problem. The original fixed point algorithm optimizes kurtosis, and the generalized ones optimize a measure of nonnormality determined by selecting a nonlinearity.
- `JADE` (Section 4) works by diagonalizing a set of cumulant matrices.
- The bigradient algorithm (Section 5) is a gradient descent algorithm that optimizes kurtosis or other contrast functions and uses a penalty term to impose orthogonality. In our experiments we shall only use other contrast functions than kurtosis for bigradient.

One especially important class of algorithms missing from this list is the set of algorithms with foundations in information theory. In Section 2 we will introduce an

Table 1 A classification of used ICA algorithms.

Mathematical approach	**Method of solution** Diagonalization	Fixed point	Gradient method
Fourth order cumulants	`JADE`	Original fixed point	Bigradient
Contrasts based on other nonlinearities	—	Generalized fixed point	Bigradient

entropy maximization algorithm of Bell and Sejnowski. It will not be used in our experiments, because it has already been applied to the problem.

2. Entropy maximization

As we saw in Section 2.3 of Chapter 3, it is possible to perform ICA by maximizing the entropy of the outputs in a two-layer neural network. However, it is not easy to form a computational procedure based directly on the general theory presented there. This is because we are unable to determine the density functions of the unknown components.

The approach Bell and Sejnowski have taken to overcome this problem is to constrain the form of the nonlinear functions in the second layer [**5**]. They use the *logistic* transfer function

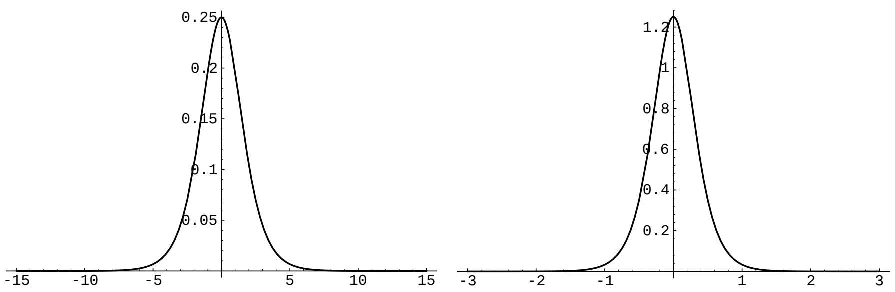
$$g(x) = \frac{1}{1 + e^{wx+w_0}},$$

where w controls the steepness of the function and w_0 is a bias parameter. Actually parameter w is embedded in the linear part of the network (the magnitude of an independent component remains unknown). Bell and Sejnowski derive a gradient ascent algorithm for changing the weights of the first (linear) layer and the biases of the second layer to maximize the entropy of the outputs.

Based on the discussion of the general theory in Section 2.3 of Chapter 3, we could assume that the performance of the algorithm depends on how well the nonlinear transfer functions approximate the distribution functions of the independent components. The derivative of the logistic function with $w = -1$ is shown in Figure 1(a). Here we can see that the probability distribution described by this density is symmetric and unimodal. Thinking of parameter w in the transfer function not as the steepness of the function but as a scaling coefficient of the distribution we may see that those distributions that can be shifted and scaled to the one described by Figure 1(a) should be handled well by the algorithm. The scaling changes only the variance of the signal, not the shape of the density.

Example 4.1. The derivative of the logistic function with $w = -1$ and $w = -5$ is depicted in Figure 1. Here we can see that the shape of the function stays the same regardless of the scaling coefficient.

The shape of the derivative is peaked, so the function could be used to approximate a positively kurtotic density. In fact Bell and Sejnowski themselves suggest that the algorithm may not work for negatively kurtotic sources. This is one very important point in which our work differs from theirs.



(a) $w = -1$ (b) $w = -5$

FIGURE 1. Derivatives of the logistic function.

3. Fixed point algorithms

The basic idea of the fixed point algorithms of Hyvärinen and Oja [**29, 27**] is that of optimization of a contrast function (see Section 6 in Chapter 2).

We have shown above that kurtosis can be used as an optimization criterion for ICA. Now consider the optimization of kurtosis of projection of a zero mean whitened random variable v onto vector w

$$f(w) = \mathrm{E}\left\{(w^T v)^4\right\} - 3\,||w||^4$$

with constraint

$$h(w) = ||w||^2 - 1 = 0.$$

Necessary conditions for an optimum are given by the method of Lagrange multipliers [**37**]

$$4\mathrm{E}\left\{(w^T v)^3 v\right\} - 12\,||w||^2\, w + 2\lambda' w = \bar{0}$$

or, by noting that $||w||^2 = 1$ and setting $\lambda = -\frac{1}{2}\lambda'$,

$$\lambda w = \mathrm{E}\left\{(w^T v)^3 v\right\} - 3w. \tag{19}$$

Equation (19) shows that at an optimum w the right hand side of the equation is parallel with w, or that the direction of w remains fixed under the iteration

$$w(k+1) = \mathrm{E}\left\{(w(k)^T v)^3 v\right\} - 3w(k).$$

It can actually be shown [**29**] that starting from a random initial point the computation given in Algorithm 1 converges to one of the desired optima if the ICA model holds. In fact, the algorithm does even better than the optimality conditions might imply. In Appendix A it is shown that the optimization of kurtosis solves the problem if there is at least one positively and one negatively kurtotic source. The fixed point algorithm works even without this assumption.

In the algorithm vectors w and w_{old} are normalized because we are only interested in their directions. It should also be noted that on line 7 we test for both directions

```
funct w = fp0(v, ϵ)
    /* Returns one ICA basis vector (column of B). */
    w := rand()/* A random initial starting point. */
    w := w/ ||w||
    w_old := 0̄
    /* Need only be parallel, check both directions. */
    while ||w − w_old|| > ϵ ∧ ||w + w_old|| > ϵ
            w_old := w
            w := E{(w^T v)^3 v} − 3w
            w := w/ ||w|| /* We are only interested in the direction. */
    end
.
```

Algorithm 1: A fixed point algorithm.

while checking for convergence since in equation (19) λ may be either positive or negative.

Algorithm 1 is guaranteed to find one source. Under some special circumstances this may be adequate for our purposes.

- We only want to find one source from which we have a priori knowledge and the first source is the desired one. The probability of convergence to such a source depends on the basin of attraction of the fixed point.
- We have two sources, $\dim(w) = 2$. Since we know that the sources are orthogonal, the second source is found immediately by taking the one dimensional orthogonal subspace.

If the situation is more complicated, we can run the algorithm successively with different starting points. However, sources with large basins of attraction tend to dominate the results. The problem can be alleviated by using the information that the whitened ICA basis vectors are orthogonal, that is, by orthogonalizing the found vector w against the subspace spanned by previously found vectors. This gives Algorithm 2, as suggested in [**29**].

To be exact, there is one flaw in Algorithm 2 if we consider the optimality conditions. We want the termination point to be the fixed point of the iteration, but the orthogonalization step complicates things. When the termination condition is fulfilled in Algorithm 2 w is actually a fixed point of the iteration-orthogonalization-normalization combination. However, such a fixed point fulfills the optimality conditions with an additional constraint that the new vector is orthogonal to the previously found ones. In Algorithm 2 (see lines 12–14) let $-BB^T w(k)^* = \sum_{i=1}^{l} \lambda_i b_i$, where l is the number of

```
funct B = fp(v, r_max, ϵ)
  /* Returns mixing matrix B. */
  B := ∅ /* Start with an empty matrix. */
  i := 1 /* The extracted source. */
  while i ≤ n /* Extract all sources. */
        w := rand()
        w := w − B B^T w /* Gram-Schmidt orthogonalization */
        w := w/ ||w||
        r := 0 /* Current iteration round. */
        while r < r_max /* Maximum number of iterations. */
              w_old := w
              w* := E{(w^T v)^3 v} − 3w
              w† := w* − B B^T w*
              w := w†/ ||w†||
              if ||w − w_old|| < ϵ ∨ ||w + w_old|| < ϵ
                 i := i + 1
                 B := [B w] /* Augment B with the found basis vector. */
                 break /* Break from the while-loop. */
              end
              r := r + 1
        end
  end
.
```

Algorithm 2: Fixed point algorithm with orthogonalization. The acronym `FP` will be used below to refer to this algorithm.

solutions already found. If $w(k)$ is a fixed point of the iteration, we know that

$$
\begin{aligned}
w(k+1) = w(k) \quad &\Leftrightarrow \quad w(k)^{\dagger} = \lambda_0' w(k) \\
&\Leftrightarrow \quad \lambda_0' w(k) = w(k)^{*} + \sum_{i=1}^{l} \lambda_i b_i \\
&\Leftrightarrow \quad \lambda_0' w(k) = \mathrm{E}\left\{(w(k)^T v)^3 v\right\} - 3w(k) + \sum_{i=1}^{l} \lambda_i b_i \\
&\Leftrightarrow \quad \mathrm{E}\left\{(w(k)^T v)^3 v\right\} + \lambda_0 w(k) + \sum_{i=1}^{l} \lambda_i b_i = \bar{0},
\end{aligned}
$$

where $\lambda_0' \in R$ and $\lambda_0 = -3 - \lambda_0'$. These are exactly the first order optimality conditions of the optimization problem with objective function

$$
f(w) = \mathrm{E}\left\{(w^T v)^4\right\} - 3\,||w||^4
$$

and constraints

$$
\begin{aligned}
h_0(w) &= ||w||^2 - 1 = 0 \\
h_i(w) &= b_i^T w = 0, \; i = 1, \ldots, l.
\end{aligned}
$$

Instead of using kurtosis as an optimization criterion we can use the general contrast functions introduced in Section 6. The *local* convergence of such algorithms has been proved in [**27**]. In the algorithm we only need to change the actual fixed point iteration (line 12). Denoting the derivative of the contrast function by g this line becomes

$$w^* := \mathrm{E}\left\{g(w^T v)v\right\} - \mathrm{E}\left\{g'(w^T v)\right\} w.$$

For $g(t) = \tanh(t)$ we have $g'(t) = 1 - \tanh^2(t)$. This is the nonlinearity we shall mostly use (see Section 6 of Chapter 2).

Because we suspect that the ICA model does not hold at least exactly for image data, the Gram-Schmidt approach to orthogonalization used in these algorithms may be too inflexible. If one of the iterations converges to a "poor" fixed point, it may affect too much the convergence of the rest of the results. This is why Algorithm 3 was developed. It searches for all basis vectors of ICA simultaneously and uses symmetric orthogonalization. The orthogonalization step is on line 13. This symmetric orthogonalization changes matrix B minimally in the Frobenius norm sense. Unfortunately, for this algorithm the effect of orthogonalization on optimality conditions is not easy to analyze.

```
1  funct B = symfp(v, g, r_max, ε)
2    B := rand()
3    B := orth(B) /* Initial orthogonalization with Gram-Schmidt. */
4    for i = 1 to n
5        b_i := b_i / ||b_i||
6    end
7    r := 0
8    while r < r_max
9          B_old := B
10         for i = 1 to n
11             b_i := E{g(b_i^T v)v} − E{g'(b_i^T v)} b_i
12         end
13         B := B(B^T B)^(−1/2) /* Symmetric orthogonalization. */
14         if ||B − B_old|| < ε /* Don't take directions into account. */
15            return B
16         end
17         r := r + 1
18   end
19 .
```

Algorithm 3: A symmetric algorithm for finding all sources simultaneously. The acronym `SYMFP` will be used below to refer to this algorithm.

4. Joint approximate diagonalization of eigenmatrices

The `JADE`-algorithm of Cardoso and Souloumiac is based on joint approximate diagonalization of eigenmatrices [**9**]. In Section 7 of Chapter 2 we saw that ICA problem could be solved by computing the eigenvectors of cumulant matrix $Q_v(M)$ for any matrix M. `JADE` is an extension of this idea in that it diagonalizes a set of eigenmatrices representing the whole cumulant matrix set $\mathcal{C}_v$. More precisely, in `JADE` the diagonalization proceeds by maximization of

$$C(B) = \sum_{M \in \mathcal{C}_{v,e}} \left|\left|\mathrm{diag}\left(B^T M B\right)\right|\right|^2,$$

where B is an orthogonal matrix and $\mathcal{C}_{v,e}$ is a set of eigenmatrices of Q_v with nonzero eigenvalues

$$\mathcal{C}_{v,e} = \{\xi M \mid Q_v(M) = \xi M \wedge \xi \neq 0\}.$$

(Here diag $\left(B^T M B\right)$ represents a vector formed by scanning the diagonal elements of $B^T M B$.) It can be shown that $|\mathcal{C}_v| = n$ **[9]**, so that the computations need only be performed over a set of n matrices. When the ICA model holds, the optimization over this eigenset $\mathcal{C}_{v,e}$ is equivalent to optimization over the whole cumulant matrix set $\mathcal{C}_v$, so the computational advantage is significant.

5. The bigradient algorithm

Above in Section 6 of Chapter 2 we have seen that kurtosis can be used as a contrast function. In Appendix A the proof of this shows that when we have positively kurtotic sources we should find the maxima of kurtosis, and for negatively kurtotic sources we should find the minima.

Let us assume that the kurtoses of the sources have the same sign. Then we can in principle solve all ICA basis vectors simultaneously by optimizing the sum of all kurtoses. This gives us a contrast function

$$\begin{aligned} J_{x,\sum\kappa}(B) &= \sum_{i=1}^{n} \kappa_4\left(b_i^T x\right) \\ &= \sum_{i=1}^{n} \mathrm{E}\left\{\left(b_i^T x\right)^4\right\} - 3\left(\mathrm{E}\left\{\left(b_i^T x\right)^2\right\}\right)^2, \end{aligned}$$

where $B = [b_1 \ \cdots \ b_n]^T$. For whitened random vector v we have $\mathrm{E}\left\{\left(b_i^T v\right)^2\right\} = 1$ for all $i \in \{1, \ldots, n\}$, so the second term can be dropped out, and we are left with a simplified objective function

$$J_{v,\sum}(B) = \sum_{i=1}^{n} \mathrm{E}\left\{\left(b_i^T v\right)^4\right\}. \tag{20}$$

To find the ICA basis we should optimize (20) with the constraint that B is orthogonal. This can be solved by using a penalty function approach. The penalized object function is

$$J_{v,\mathrm{penalty}}(B) = \underbrace{J_{v,\sum}(B)}_{\text{original objective}} + \rho \underbrace{\left|\left|I - B^T B\right|\right|_{\mathrm{F}}^2}_{\text{orthogonalization}}, \tag{21}$$

where $\rho \in R$ is a penalty coefficient and $||X||_{\mathrm{F}} = \sum_{i=1}^{n}\sum_{j=1}^{n} x_{ij}^2$ denotes the *Frobenius norm* of matrix X.

The actual optimization is performed with a gradient descent algorithm. Differentiating (21) with respect to B yields

$$J'_{v,\mathrm{penalty}}(B) = 4\mathrm{E}\left\{v\left(v^T B\right)^3\right\} + \rho B(I - B^T B),$$

where $(\cdot)^3$ operates componentwise on a vector. A gradient descent or ascent rule — depending on the signs of kurtoses — for changing B is

$$\begin{aligned} B(k+1) &= B(k) + \mu_k J'_{v,\mathrm{penalty}}(B(k)) \\ &= B(k) + \mu_k \mathrm{E}\left\{v\left(v^T B(k)\right)^3\right\} + \gamma_k B(k)\left[I - B(k)^T B(k)\right], \end{aligned}$$

where $\gamma_k = \mu_k \rho_k$ and notation $B(k)$ indicates time indexing. Taking a stochastic gradient approach by changing the expectation to instantaneous value of input data gives the final form of the bigradient algorithm

$$B(k+1) = B(k) + \mu_k \mathrm{E}\left\{ v(k) \left(v(k)^T B(k) \right)^3 \right\} + \gamma_k B(k) \left[I - B(k)^T B(k) \right].$$

Adaptation parameter μ_k is a small constant or decreases slowly with the number of iterations, and acceptable values of γ_k have been observed to be in the range between 0.5 and 1 [**53**]. Like in the fixed point algorithm, the third power of the algorithm may be changed to another nonlinearity g like the popular $g(t) = \tanh(t)$, yielding the generalized bigradient algorithm

$$B(k+1) = B(k) + \mu_k \mathrm{E}\left\{ v(k)\, g\left(v(k)^T B(k) \right) \right\} + \gamma_k B(k) \left[I - B(k)^T B(k) \right].$$

It should be noted that the bigradient algorithm is only able to separate sources all of which have either positive or negative kurtoses. This is typical of gradient descent or ascent based ICA algorithms in general. We can not assure that image data is a mixture of such similar sources. The use of such algorithms is still justified in our case, because information about any of the “independent” components is valuable to us.

Part 2

Application of ICA to image data

CHAPTER 5

Objectives and preparatory procedures

1. Introduction

There are several goals in the experiments to apply ICA to image data. At least the following objectives can be identified.

1. Verify/assess the applicability of the linear ICA model for image data and consider the limitations of the model.
2. Assess the applicability of the algorithms to the problem at hand.
3. Obtain basic knowledge about the characteristics of the "independent components" of the data.
4. Consider the applicability of the results to various image processing tasks, notably feature extraction and compression.
5. Interpret the results to assess the connections between ICA and other areas of research and the implications of these similarities and differences.

The objectives are presented in the order of importance, so that the successfulness of the first objectives affects the applicability of the other results.

The first objective is *by far* the most important, profound and difficult one. First, it is very difficult to describe how and why the image data violates the model. Second, even if we could verify and quantify this violation, the analysis of the behavior of different algorithms in the case of perturbations from the model is a formidable task. It is most certainly a subject worth of a thesis of its own, and so we are only able to scratch its surface here. We try to evaluate the independence property, but only for pairwise independence. We have to be constantly aware of the consequences of the model and violations of it.

Before we start to describe the experiments, we shall first consider all of the different parameters we have to choose in each experiment. When choosing these parameters we have to keep in mind that we are trying to fulfill the objectives stated above. The parameters include

- image data
- sampling
- data preprocessing
- number of sources (dimensionality reduction)
- the algorithm and its parameters
- estimation of statistical quantities (number of samples)

2. Image data

While choosing the image data we have to pay special attention to objectives 3–5.

Our basic approach is to apply the method to general image data instead of using e.g. textures. This has several reasons. First, it gives us results we can compare to previous work. This problem is also probably the most interesting one, offering simultaneously answers to many of the questions posed beforehand. Additionally, the fact that textures have a very restricted form may pose additional problems in terms of statistical dependencies.

The generality is introduced into the data by using a set of 15 "natural" images (Figures 1 and 2). They were digitized into Canon's PhotoCD format in a professional laboratory. The sizes of the images are either 256×512 or 512×256 pixels.

For completeness we do include one experiment with textures. The textures used (Figure 3) are a subset of the Brodatz texture set used as standard data in image processing experiments. The texture images are all of size 256×256 pixels.

While studying relationships with the other areas of research described in Chapter 3 we should notice that in two cases an extremely relevant property is the amount of variability in image data. These two cases are wavelets and biology. Wavelets are general purpose mathematical tools that work acceptably well with different kinds of data. The visual system has to operate in many different environments, so comparisons with it must also be based on general data.

3. Sampling

In sampling we have to choose the size of the samples (subimages) we are taking from the images and the window function with which the sample windows are taken. (Here sampling does not mean digitization, but obtaining the sample windows.) When we are sampling we should try to avoid excessive artificial effects due to limited window size. Sampling is also connected with objectives 3–5.

The size of the window should be a compromise between two requirements.

- It is large enough to contain sensible visual information even though the image is digitized (as a counterexample consider a subimage of size 2×2) and to retain the spectral properties of the original image.
- It is small enough to introduce generality into the data. To find any general properties in image(s), we have to restrain to local neighborhoods.

The window size we are using here is 12×12 pixels. This size was inspired by [**5**], who used it in similar experiments ([**45**] used a 16×16 window). Same image size facilitates the comparison of their results with those of ours.

The most usual windowing method is the rectangular one, where a sample window of size $N \times N$ with top corner positioned at x_0, y_0 is obtained by multiplying the original image function $i(x, y)$ with window function

$$w(x, y, x_0, y_0) = \begin{cases} 1 & \text{if } x_0 \le x \le x_0 + N - 1 \text{ and } y_0 \le y \le y_0 + N - 1, \\ 0 & \text{otherwise.} \end{cases}$$

After the multiplication the area outside the window is discarded, leaving an $N \times N$ image window.

Although rectangular windowing is intuitively pleasing, it has some unwanted properties. First of all, the abrupt change at the edges of the window introduces high frequencies into the samples. If the window size is small, these will change significantly the overall spectral characteristics when compared with those of the original image [**46**]. This problem can be alleviated by using a window function that decays smoothly. In addition, the rectangular windowing prefers diagonal directions, since when taking a sample we retain pixels from a larger distance in the diagonal directions. As a solution one can use a round window. A solution to both of these problems is a round, smoothly decaying window function, that seems to be biologically plausible. We shall call this sampling *radially smooth*.

(a) img0001

(b) img0003

(c) img0005

(d) img0011

(e) img0014

(f) img0017

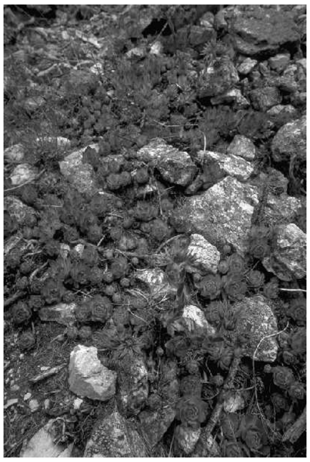
(g) img0019

(h) img0022

FIGURE 1. Natural images

(a) `img0023`

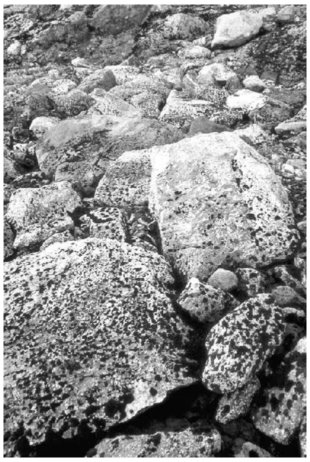
(b) `img0028`



(c) `img0007`

(d) `img0021`

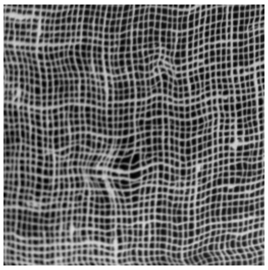
(e) `img0024`



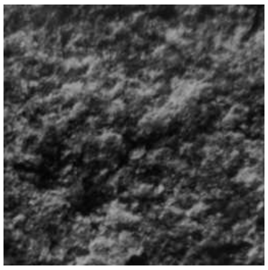
(f) `img0026`

(g) `img0030`

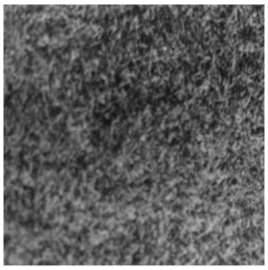
FIGURE 2. Natural images (*continued*)



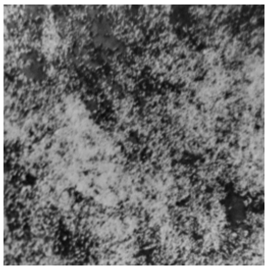
(a) `burlap`



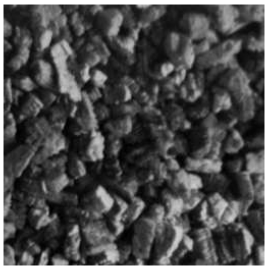
(b) `fieldstone`



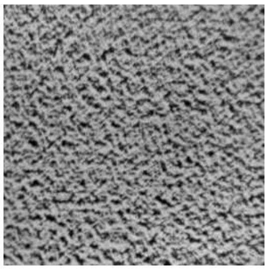
(c) `grass`



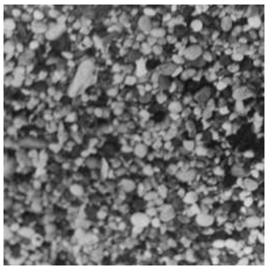
(d) `ice`



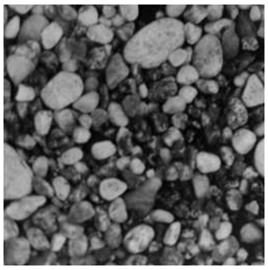
(e) `mica`



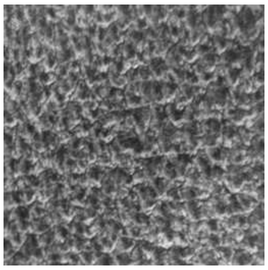
(f) `paper`



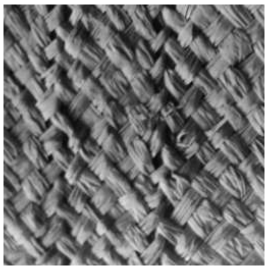
(g) `peb54`



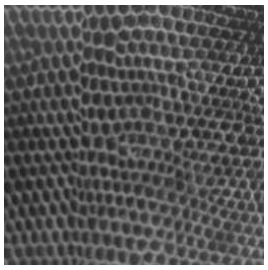
(h) `pebbles23`



(i) `pig`

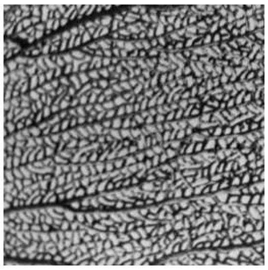


(j) `raffia`

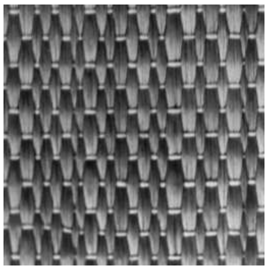


(k) `reptile`

(l) `ricepaper`



(m) `seafan`



(n) `straw`

(o) `tree`

FIGURE 3. Textures

In view of objectives 3–5 we note the following. To obtain basic knowledge we should use sampling that retains those basic properties we are interested in. This might imply that for a small window size we should use smooth windowing if we are interested in spectral properties. In the application area, for feature extraction the windowing method should be chosen to give good results (for example in classification problems). This is not an easy thing to do. In compression we want to preserve the linearity and invertibility of the operations, so here rectangular sampling is justified. This is also the case when we compare the results with other linear transformations, such as PCA or wavelets. Contrasting our findings with the knowledge from the visual system may require radially smooth sampling.

4. Data preprocessing

Preprocessing is used to facilitate the operation of ICA algorithms and to enhance the features we are interested in. In view of our objectives, objective 3 and compression application of objective 4 favor simple preprocessing, feature extraction can use any relatively simple preprocessing, and objective 5 allows any preprocessing that is used or plausible in the comparative areas of research. It should also be noted that working with different preprocessing methods may give valuable information about the algorithms.

Preprocessing stages can be divided into two different categories.

General preprocessing for ICA: ICA algorithms require certain properties from the data to function correctly. Usually they require that the data be zero mean and whitened.

Image data preprocessing: This phase is application dependent and is directed towards bringing out the properties of image data we are interested in and removing peculiarities of natural image data that might interfere with ICA algorithms.

Whitening was discussed above in Section 5 of Chapter 2. The purpose of this section is to consider other preprocessing steps and their implications.

Since our point of view in image processing is on local low-level processing, what we are interested in images are local spatial properties like edges, corners, points or lines. Because these correspond to changes in grayscale values, the local mean is subtracted from each sample to bring out these details and to remove the effect of the uninteresting constant part. Subtracting the local mean gives us a new data vector

$$x_{\text{zero},i} = x_{\text{obs},i} - \frac{1}{m}\sum_{j=1}^{m} x_{\text{obs},j}, \quad i = 1, \ldots, m.$$

This can also be written as

$$x_{\text{zero}} = x_{\text{obs}} - \frac{1}{m} x^T \bar{1}\bar{1}, \tag{22}$$

where $\bar{1} = [1\ 1\ \cdots\ 1]^T$.

Subtraction of local mean has the effect that variance along one PCA dimension becomes zero. This dimension is spanned by vector $\bar{1}$ and is called the *DC (or direct current) component* of the vector. The name comes from the fact that projection onto this dimension gives us the constant component of the vector, the part which has no frequency components. From (22) we can see that

$$x_{\text{zero}} = x_{\text{obs}} - \frac{1}{||\bar{1}||^2} x^T \bar{1}\bar{1},$$

which shows that this operation removes the projection onto $\bar{1}$.

Assuming that we take the samples from images randomly at overlapping positions, the mean of all components is the same, that is, $\mathrm{E}\{x_{\text{obs},i}\} = \mathrm{E}\{x_{\text{obs},j}\}$, $i, j \in \{1, \ldots, m\}$. After subtraction of local mean

$$\begin{aligned} \mathrm{E}\{x_{\text{zero},i}\} &= \mathrm{E}\left\{ x_{\text{obs},i} - \frac{1}{m}\sum_{j=1}^{m} x_{\text{obs},j} \right\} \\ &= \mathrm{E}\{x_{\text{obs},i}\} - \frac{1}{m}\mathrm{E}\left\{ \sum_{j=1}^{m} x_{\text{obs},j} \right\} = 0, \end{aligned}$$

so $\mathrm{E}\{x_{\text{zero}}\} = \bar{0}$.

The algorithms used to extract ICA properties use extensively projections of samples onto ICA basis vectors. If some sample vectors are much longer than others, their projections are significantly larger and they contribute much more to the result. In the case of zero mean samples from images, the square of the length of a sample vector x (realization of a random variable) $||x||^2 = \sum_{i=1}^{m} x_i^2$ is the same as the local variance in the image. This shows that image parts with greater local variance have a greater effect on the outcome of the algorithms, which is not what we want. This is why we equalize the local variance by normalizing the input variable x_{zero}, giving us

$$x_{\text{norm}} = \frac{x_{\text{zero}}}{||x_{\text{zero}}||}.$$

After this all samples have an equal length of one.

The effect of normalization on the mean of samples is not significant. Assume that the density $f_x(x)$ of a zero-mean random variable x is symmetric with respect to the origin, i.e., $f_x(-x) = f_x(x)$. The assumption that the distribution of gray levels around the mean of an image is symmetrical leading to the symmetric property of f is very reasonable. The new expected value is

$$\mathrm{E}\left\{ \frac{x}{||x||} \right\} = \int_{-\infty}^{\infty} \frac{x}{||x||} f_x(x)\, dx$$

(the single integral $\int dx$ means integration with respect to all components). The integrand $\frac{x}{||x||} f_x(x)$ is an antisymmetric function, so $\mathrm{E}\{x_{\text{norm}}\} = \bar{0}$.

To return to our objectives again, we could summarize that the two preprocessing steps introduced here, subtraction of the local mean and normalization, are both simple and fast, so they *can be* used in all investigations. However, when examining connections with biology these operations should be done adaptively for each picture element in a radial neighborhood to justify their biological plausibility.

5. The algorithm and its parameters

As was discussed in Chapter 4, the following algorithms will be used to extract the "independent" components of image data.

- Fixed point algorithms
- `JADE`

- The bigradient algorithm

For the fixed point algorithm the parameters we have to choose are

- the nonlinearity $g(t)$
- seed of the random number generator s_{rand} (effects initial starting points)
- convergence parameter ϵ
- maximum number of iterations

The nonlinearities we will work with are $g_1(t) = t^3$ and $g_2(t) = \tanh(t)$.

In order to examine the dependence of the results on initialization we experiment with different initial starting points. The convergence parameter ϵ is chosen to be as small as possible, but large enough to allow for some errors in the model and estimation of statistical quantities involved and fluctuation in the numerical algorithm. In practice this means that the smallest ϵ with which the algorithm seems to ensure convergence is found, and if this value is "too large", the ICA model or applicability of the algorithm for the data at hand will be questioned. The maximum number of iterations is chosen heuristically.

`JADE` has only one parameter, which is used to determine when the diagonalization has converged. This is chosen automatically in the algorithm written by Cardoso, and the chosen value is based on the number of available samples.

The bigradient algorithm is the most problematic algorithm when parameters are concerned. It has two parameters

- In **[53]** it is adviced that the gradient descent/ascent step size μ should be a small constant or decrease slowly with the number of iterations. We choose to use a small constant, since otherwise we would have to determine the descent rule for the parameter as well. The constant we use is 10^{-3}.
- The same authors write that reasonable values for parameter γ containing the penalty parameter and the penalty parameter have been observed to be between 0.5 and 1. We use 0.5.
- The number of epochs N_{e} is chosen to determine how long the algorithm should be run. In our experiments we have used 20 epochs, each epoch consisting of the 10000 sample vectors.

6. Number of sources

Even if we use a constant window size, the number of "independent" sources that we are trying to find may be nonconstant. If we have simple data, this data may not effectively span all directions in the space. (That is, the covariance matrix may have some very small but nonzero eigenvalues.) Fortunately PCA can be used to discover such situations.

For general image data we have selected to use two different dimensions. The first one contains 90% of the energy of the data, but is still much lower than the dimension of the original data. The other dimension is the full dimensionality of the data, with the exclusion of the removed DC component.

7. Estimation of statistical quantities

The statistical quantities that have to be estimated in the algorithms are fortunately very simple expectations. So forming the estimates is no problem. The only remaining open question is the amount of data needed for these estimates.

The general rule of thumb for the number of samples used in estimation is that we should have 5–10 measurements per each estimated parameter. If we estimate n sources, the number of estimated parameters (elements of matrix B) is n^2. Assuming that we use c window samples for estimation, the number of measurements is cn, and we have c/n measurements for each estimated parameter. In our experiments we have had $c = 10000$ and $c/n > 50$.

8. Assessing results

Since this work is largely an experimental research work, we should consider the ways in which we can assess our results. In view of the objectives set forth at the beginning of this chapter we have chosen to use the following methods to evaluate the outcomes of our experiments.

1. We examine the fundamental problem — the claim of transforming the data into a more independent form — by exploring pairwise independence using histograms and pairwise histograms of the source components. This will be done by computing the Kullback-Leibler distance between the product histogram and the joint histogram of the distributions, and will be discussed in detail below. Unfortunately measuring independence is itself a highly nontrivial problem.
2. The evaluation of the applicability of the algorithms is based partly on the results of the previous step and mostly on observing the behavior of the algorithms and the visual quality of the results. For example, failure to converge is a clear indication of the unsuitability of the algorithm to this problem.
3. Basic knowledge about the components is obtained mainly by visual inspection.
4. Applicability to image processing tasks is evaluated by examining some relevant properties of the sources and basis vectors. Unfortunately the results are not "strong enough" from an application point of view for us to apply them directly.
5. Connections with other disciplines are examined by checking whether our results fulfill properties that have been attributed to be central in these other fields.

CHAPTER 6

Experiments

1. Overview

Table 1 summarizes our experiments. The set of experiments is divided into two different subsets.

1–12 These experiments provide basic information of ICA of image data and the applicability of different algorithms. The first experiments are performed in a lower dimension of 49 to reduce the effect of noise. Then we examine the applicability of different algorithms when the dimension is high, i.e., there exists noise and a threat of the "curse of dimensionality."

13–16 The rest of the experiments are used to determine some interesting properties of the "independent components." We examine the effect of different initial starting points, the effect of normalization as a preprocessing step, the effect of local preprocessing and the "independent components" of textures.

In order to facilitate the inspection of the table we have marked some important attributes of each experiment with gray color.

The meanings of columns are

#	The identifying number of the experiment.
Data	We use either the 15 natural images (nat) or 15 textures (tex).
Samp	The sampling can be rectangular (rect) or radially smooth (rad).
Prep	Preprocessing may include subtraction of local mean ($\mathrm{E}\{x\}$) and normalization ($\|\|x\|\|$). Usually these steps are done for each sample. If the operations are performed separately for each pixel in a sliding neighborhood, it is indicated with (l) for local.
n	The number of sources. This is of 143 possible, because the DC component has been removed by subtracting the local mean.
Algorithm (params)	The used algorithm with values of relevant parameters. Here `FP` and `SYMFP` stand for the fixed point algorithms with Gram-Schmidt and symmetric approaches to orthogonalization, respectively.
Summary	A short description of the results.

The rest of this chapter is organized as follows. First we describe the results of the first subset of experiments. Then we evaluate the different algorithm and select one of them (`FP`) for use in the rest of the experiments, because otherwise the number of experiments would grow too large. After the algorithm has been chosen, the descriptions of the rest of the experiments follow. Some analysis of the results is presented in Section 5.

Table 1 A summary of experiments.

#	Data	Samp	Prep	n	Algorithm (params)	Summary
1	nat	rect	$\mathrm{E}\{x\}$, $\|\|x\|\|$	49	SYMFP ($\tanh(t)$)	large structures, bars, edges
2	nat	rect	$\mathrm{E}\{x\}$, $\|\|x\|\|$	49	FP ($\tanh(t)$, $s_{\mathrm{rand}} = 1$)	similar to SYMFP, waves
3	nat	rect	$\mathrm{E}\{x\}$,$\|\|x\|\|$	49	JADE	similar to SYMFP, some PCA vectors
4	nat	rect	$\mathrm{E}\{x\}$,$\|\|x\|\|$	49	SYMFP (t^3)	similar to SYMFPwith $g(t) = \tanh(t)$
5	nat	rect	$\mathrm{E}\{x\}$,$\|\|x\|\|$	49	FP (t^3)	similar to SYMFP, convergence problems
6	nat	rect	$\mathrm{E}\{x\}$,$\|\|x\|\|$	49	bigradient , $\tanh(t), \mu = 10^{-3}, N_{\mathrm{e}} = 20$	similar to SYMFP, one negatively kurtotic
7	nat	rect	$\mathrm{E}\{x\}$,$\|\|x\|\|$	49	bigradient , $\tanh(t)$, $\mu = -10^{-3}$, $N_{\mathrm{e}} = 20$	no features
8	nat	rect	$\mathrm{E}\{x\}$, $\|\|x\|\|$	143	SYMFP ($\tanh(t)$)	large structures, edges, bars and some more localized
9	nat	rect	$\mathrm{E}\{x\}$, $\|\|x\|\|$	143	FP ($\tanh(t)$)	similar to SYMFP, a bit poorer definition
10	nat	rect	$\mathrm{E}\{x\}$,$\|\|x\|\|$	143	SYMFP (t^3)	local patterns→noisy
11	nat	rect	$\mathrm{E}\{x\}$,$\|\|x\|\|$	143	FP (t^3)	convergence problems,noisy
12	nat	rect	$\mathrm{E}\{x\}$,$\|\|x\|\|$	143	bigradient , $\tanh(t)$, $\mu = 10^{-3}$, $N_{\mathrm{e}} = 20$	poor definition
13	nat	rect	$\mathrm{E}\{x\}$, $\|\|x\|\|$	49	FP ($\tanh(t)$, $s_{\mathrm{rand}} = 2$)	different results, qualitatively alike
14	nat	rect	$\mathrm{E}\{x\}$	143	FP ($\tanh(t)$)	noisy, convergence problems
15	nat	rad	$\mathrm{E}\{x\}$, $\|\|x\|\|$ (l)	49	FP ($\tanh(t)$)	localized receptive fields, convergence problems
16	tex	rect	$\mathrm{E}\{x\}$,$\|\|x\|\|$	49	FP ($\tanh(t)$)	soft forms, bars and frequencies

2. Basic properties of indendent components and algorithms

Experiment 1

Data	nat
Sampling	rect
Preprocessing	$\mathrm{E}\{x\}$, $\|\|x\|\|$
Number of sources	49
Algorithm and parameters	SYMFP ($\tanh(t)$)
Summary	large structures, bars, edges
Figure	1

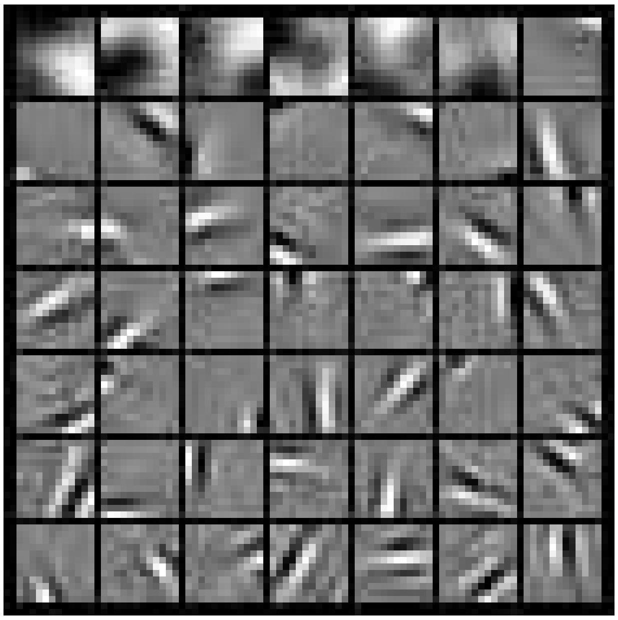


FIGURE 1. ICA basis vectors of experiment 1.

From each of the experiments we present the ICA basis vectors. The vectors have been ordered after running the algorithm. The ordering criterion is the norm of the mean frequency of the filter (see Section 4 of Chapter 3). We have used the discrete cosine transform (DCT) instead of the discrete Fourier transform (DFT) to obtain the frequency contents of each vector. DCT assumes that the signal is symmetric, while DFT assumes that it is periodic. With large scale structures the assumption of periodicity creates an edge effect and results in false high frequency contents. The same ordering will be used for all subsequent results.

The number of sources 49 was chosen by examining the distribution of the eigenvalues of the covariance matrix. In Figure 2 we can see that about 90% of the energy lies in this subspace (to be exact this number is 89.2%). The corresponding eigenvectors are shown in Figure 3. The eigenvectors have been ordered so that on the top left is the vector corresponding to the largest eigenvalue and the ordering proceeds left$\rightarrow$right top$\rightarrow$down. Note that these PCA basis vectors differ significantly from the obtained ICA basis vectors of Figure 1.

Looking at the results of Figure 1 we see that there are some low frequency basis vectors describing large structures in the image. Most of the vectors represent bars, but there are also some vectors with edge-like structure, some spots, and some vectors with other structure (Figure 4). It should be noted that most of the other structures than bars reside on the edge of the subimage, which could mean that they are just partially visible larger structures, like bars.

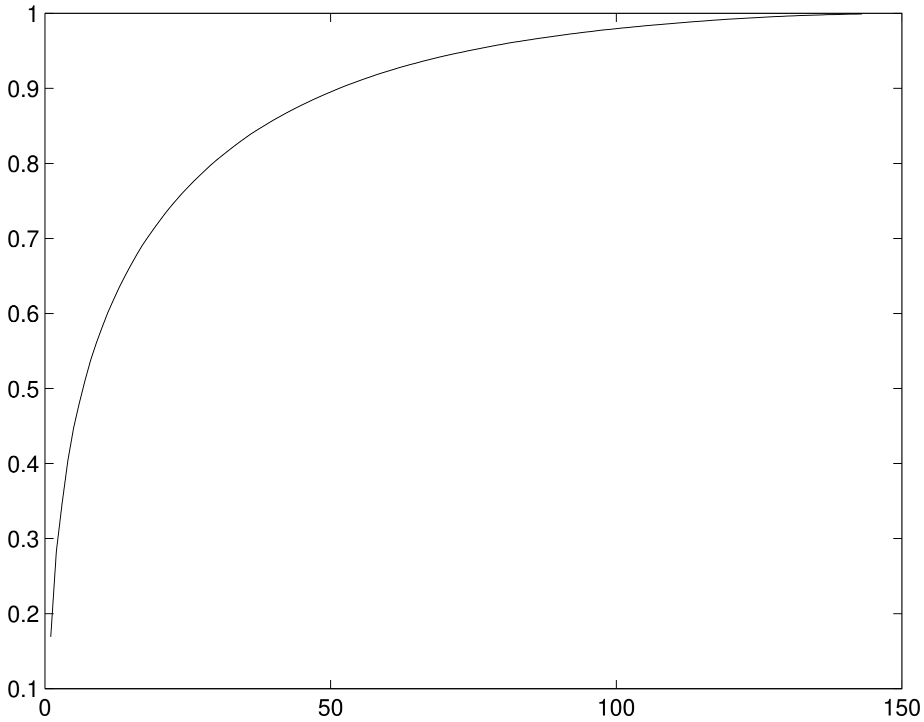


FIGURE 2. Cumulative sum of the eigenvalues of the covariance matrix.

Experiment 2

Data	nat
Sampling	rect
Preprocessing	$\mathrm{E}\{x\}$, $\|\|x\|\|$
Number of sources	49
Algorithm and parameters	`FP` ($\tanh(t)$, $s_{\text{rand}} = 1$)
Summary	similar to `SYMFP`, waves
Figure	5

In Section 3 of Chapter 4 we discussed the fixed point algorithm with symmetric and Gram-Schmidt approaches to orthogonalization. This experiment was carried out using `FP` to study this difference. The results are very similar to the ones obtained with `SYMFP`. Most notably there is a new vector type that could be described to be a wave.

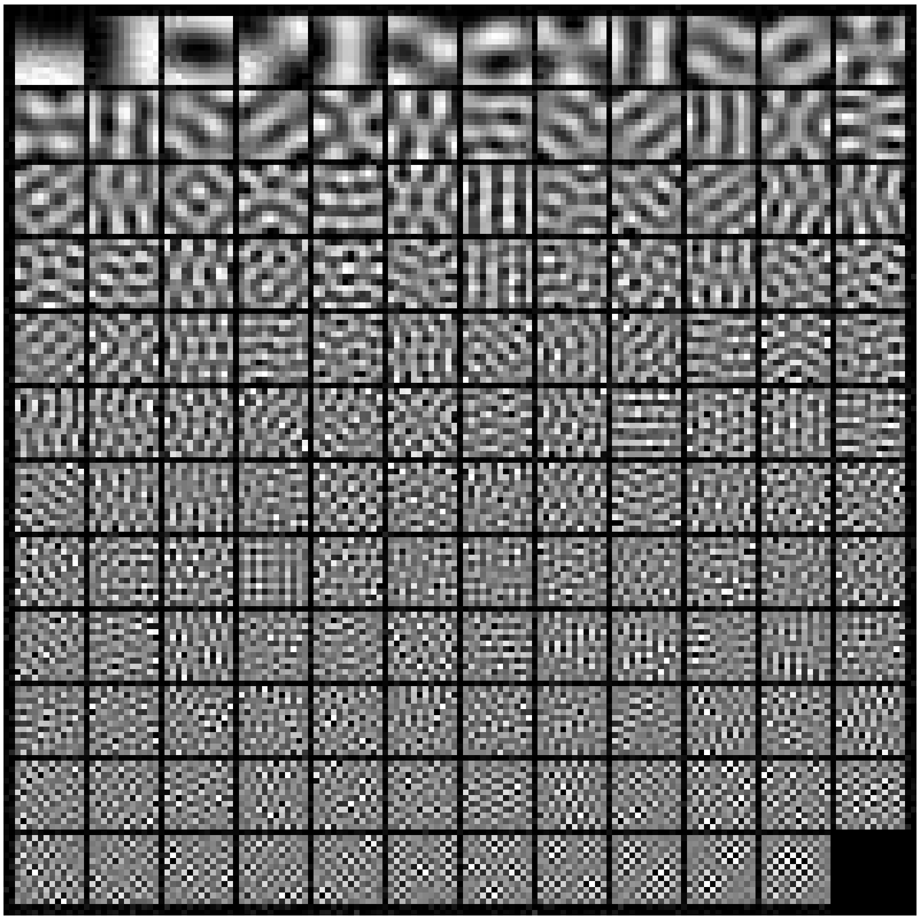


FIGURE 3. Eigenvectors of the covariance matrix.

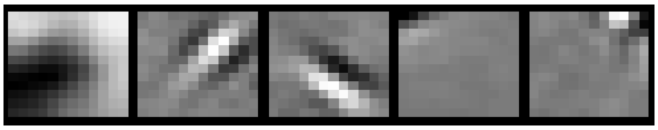


FIGURE 4. A large structure, a bar, an edge, a spot and an unclassified structure.

Experiment 3

Data	nat
Sampling	rect
Preprocessing	$\mathrm{E}\{x\}$,$\|\|x\|\|$
Number of sources	49
Algorithm and parameters	`JADE`
Summary	similar to `SYMFP`, some PCA vectors
Figure	6

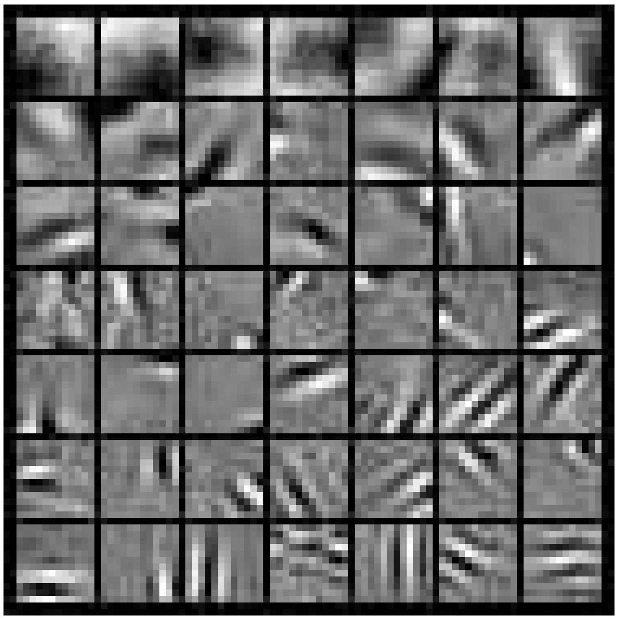


FIGURE 5. ICA basis vectors of experiment 2.

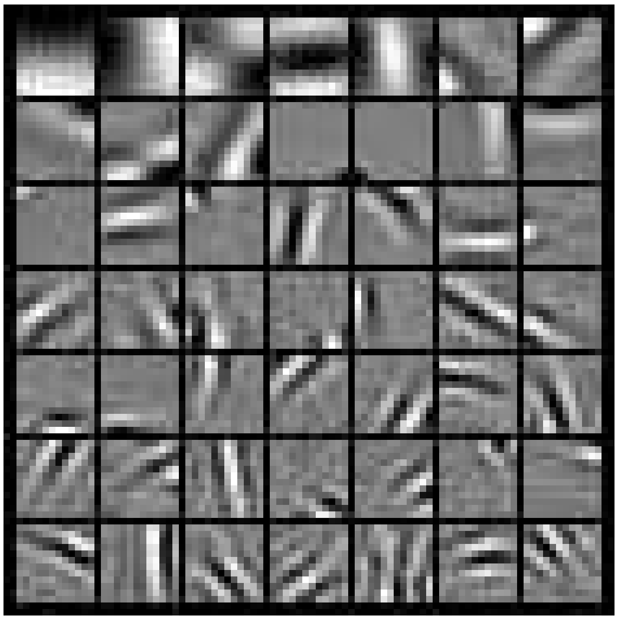


FIGURE 6. ICA basis vectors of experiment 3.

The results obtained with `JADE` are very much like those obtained with the fixed point algorithms. There is one significant new property: the large structures found are almost exactly the same as the eigenvectors of the covariance matrix corresponding to the largest eigenvalues. The reasons for this are still unclear.

It is a promising sign that the results obtained using this algebraic diagonalization method are so similar to those using measures of deviation from normality.

Unfortunately `JADE` needs the estimates for all fourth order cumulants. The number of them is n^4, where n is the number of sources. This causes both the running time and the amount of needed memory to grow too large for practical purposes when higher dimensions are considered (even this experiment took about 11 hours to complete).

Experiment 4

Data	nat
Sampling	rect
Preprocessing	$\mathrm{E}\{x\}$,$\|\|x\|\|$
Number of sources	49
Algorithm and parameters	`SYMFP` (t^3)
Summary	similar to `SYMFP`with $g(t) = \tanh(t)$
Figure	7

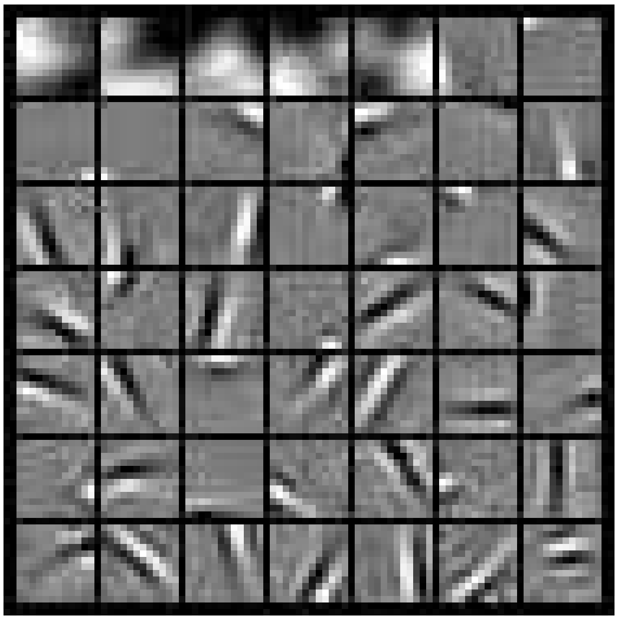


FIGURE 7. ICA basis vectors of experiment 4.

With the reduced dimension 49 the change of the nonlinearity of the fixed point algorithm from $g(t) = \tanh(t)$ to $g(t) = t^3$ does not change the quality of the results (see Figure 1). The reduction of the dimension has removed noise from the system, so the fact that the cubic nonlinearity works well is not a surprise.

Experiment 5

Data	nat
Sampling	rect
Preprocessing	$\mathrm{E}\{x\}, \|\|x\|\|$
Number of sources	49
Algorithm and parameters	`FP` (t^3)
Summary	similar to `SYMFP`, convergence problems
Figure	8

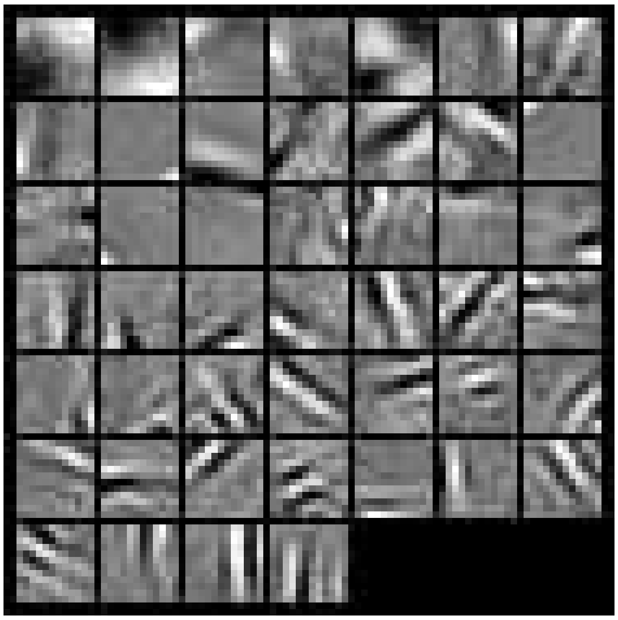


FIGURE 8. ICA basis vectors of experiment 5.

With the deflation approach the use of the cubic nonlinearity $g(t) = t^3$ creates a convergence problem. The algorithm is unable to converge for the last two basis vectors.

Experiment 6

Data	nat
Sampling	rect
Preprocessing	$\mathrm{E}\{x\}, \|\|x\|\|$
Number of sources	49
Algorithm and parameters	bigradient, $\tanh(t), \mu = 10^{-3}, N_{\mathrm{e}} = 20$
Summary	similar to `SYMFP`, one negatively kurtotic
Figure	9

The results from the first run using the bigradient algorithm include local structures common from previous experiments and one larger structure. This larger structure,

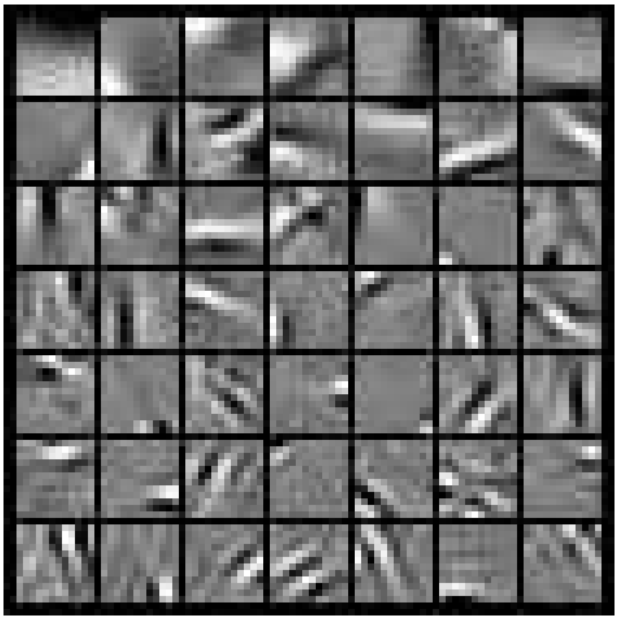


FIGURE 9. ICA basis vectors of experiment 6.

the source of which has negative kurtosis (-0.43), has presumably appeared as a result of the orthogonality constraint. The emerged structures are not as clear as in many other experiments. This may be due to problems with selecting the parameters for the algorithm.

Here too the most notable property is the uniformity of the results with those of other algorithms, which confirms our notion about the basic nature of the "independent components" of natural images in the lower dimensional case.

Experiment 7

Data	nat
Sampling	rect
Preprocessing	$\mathrm{E}\{x\}, \|\|x\|\|$
Number of sources	49
Algorithm and parameters	bigradient, $\tanh(t)$, $\mu = -10^{-3}$, $N_e = 20$
Summary	no features
Figure	10

When the bigradient algorithm is used to extract the negatively kurtotic sources, the results have no clearly visible structure. The previous experiments with other algorithms indicate that there may not be a full basis set of such vectors. This may be the cause of these uninteresting results. Because of these observations the algorithm was not applied in the higher dimensional case.

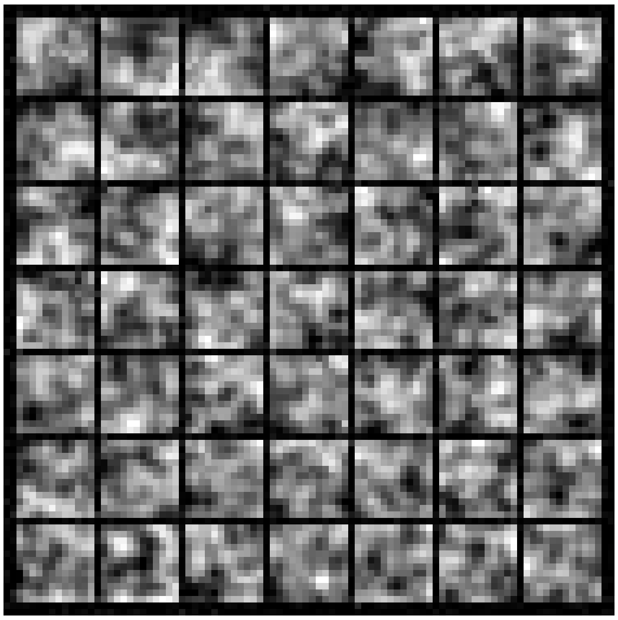


FIGURE 10. ICA basis vectors of experiment 7.

Experiment 8

Data	nat
Sampling	rect
Preprocessing	$\mathrm{E}\{x\}$, $\|\|x\|\|$
Number of sources	143
Algorithm and parameters	SYMFP ($\tanh(t)$)
Summary	large structures, edges, bars and some more localized
Figure	11

When the whole set of 143 components is used (remember that one component has been removed while subtracting the local mean), we note that the biggest change has happened in the range of the sizes of different structures (see Figure 1). This is understandable, since previously restricting to a PCA subspace removed some of the details of the image data.

It seems that some of the previous types, like the large structures, edges and some bars have remained in the set. This is something of a good sign, since it suggests that the large scale sources are sources also in the data with smaller details. In addition to the previously known feature types more local features have appeared.

If the data contains a noise subspace which is not removed by dimensionality reduction, then during whitening the variance of the noise may be enlarged significantly (because all variances are equalized). Thus the noise level in the whitened data may be quite large. In this case 51 smallest eigenvalues of the covariance matrix are smaller than 10^{-3} when the largest eigenvalue is 0.17, so the smallest ones correspond to very little detail or plain noise. The algorithm seems to cope very well with the situation

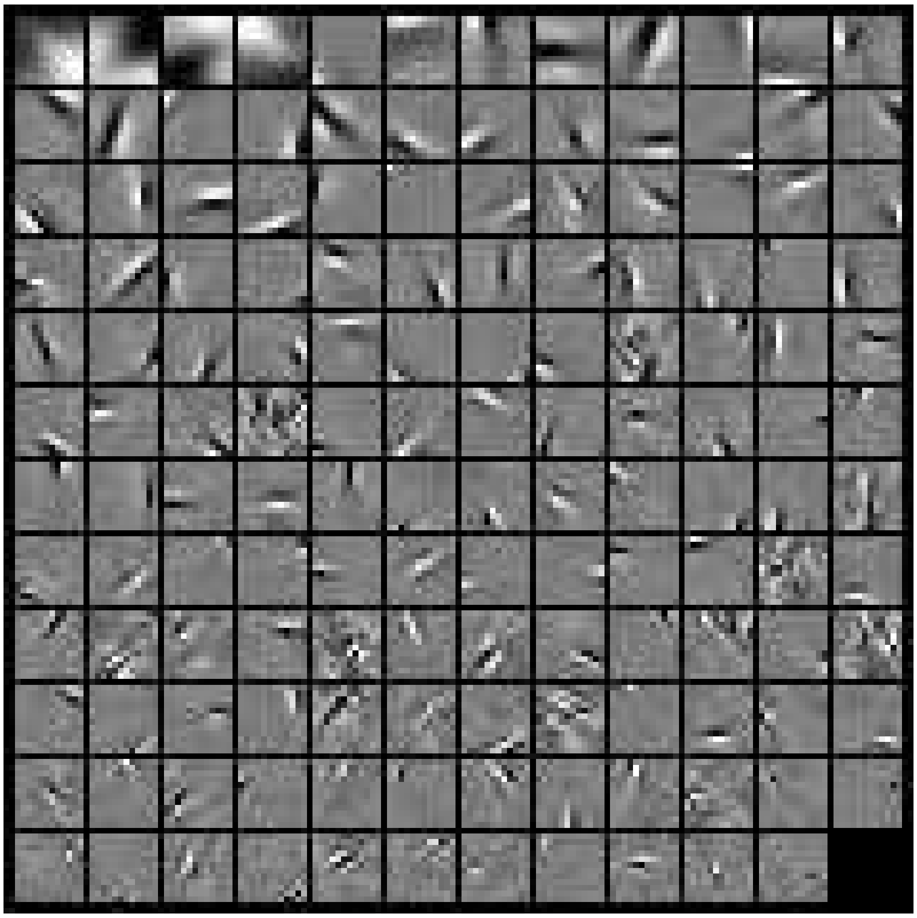


FIGURE 11. ICA basis vectors of experiment 8.

in the sense that most of the basis vectors have a meaningful structure, although there are also a few with poorer definition.

Experiment 9

Data	nat
Sampling	rect
Preprocessing	E $\{x\}$, $\|\|x\|\|$
Number of sources	143
Algorithm and parameters	FP ($\tanh(t)$)
Summary	similar to SYMFP, a bit poorer definition
Figure	12

With the deflation approach the results in the higher dimension are very much like those obtained with the symmetric one. However, the structures are a bit "underdeveloped", and a few basis vectors have no visible structure.

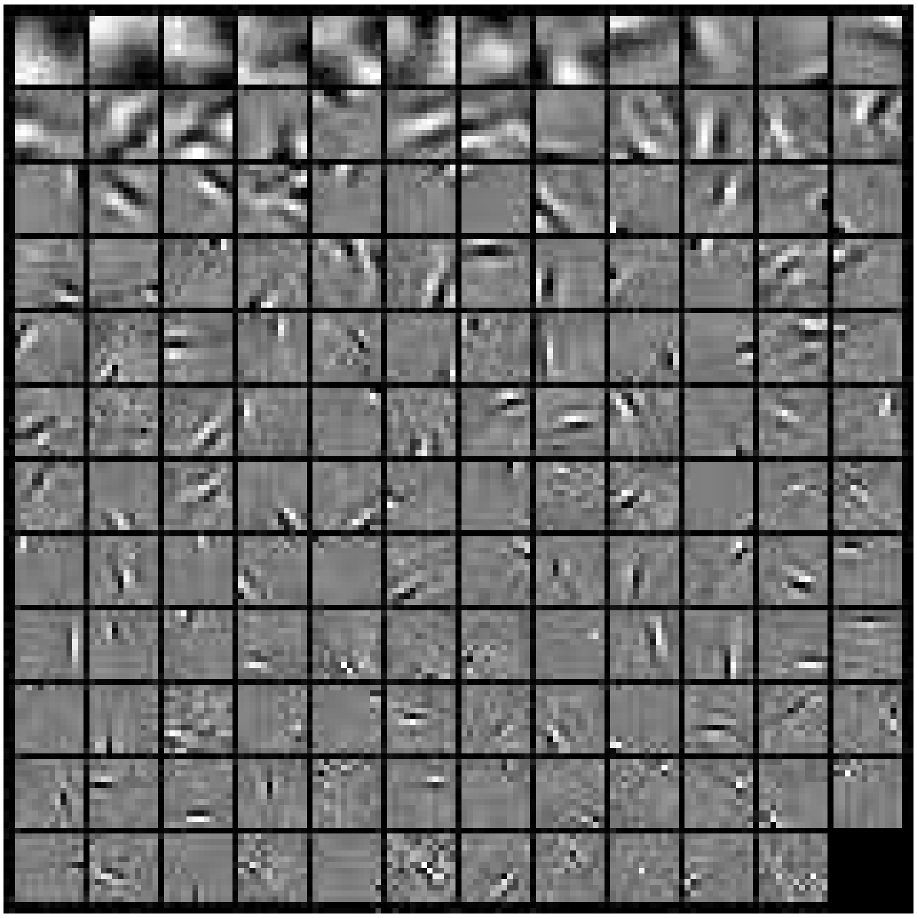


FIGURE 12. ICA basis vectors of experiment 9.

Experiment 10

Data	nat
Sampling	rect
Preprocessing	$\mathrm{E}\{x\}$,$\|\|x\|\|$
Number of sources	143
Algorithm and parameters	`SYMFP` (t^3)
Summary	local patterns$\rightarrow$noisy
Figure	13

With the fast growing nonlinearity $g(t) = t^3$ the results can again be divided into two parts. The large structures appear similar as before, but this time the other part consists of noisy patterns. This can be explained by thinking of the effect the noise has on kurtosis.

The sources corresponding to the large structures have negative kurtosis while the sources of the local structures have positive kurtosis. Now thinking of the effect noise has on the estimation of kurtosis, it is intuitive that a peaked distribution is easily

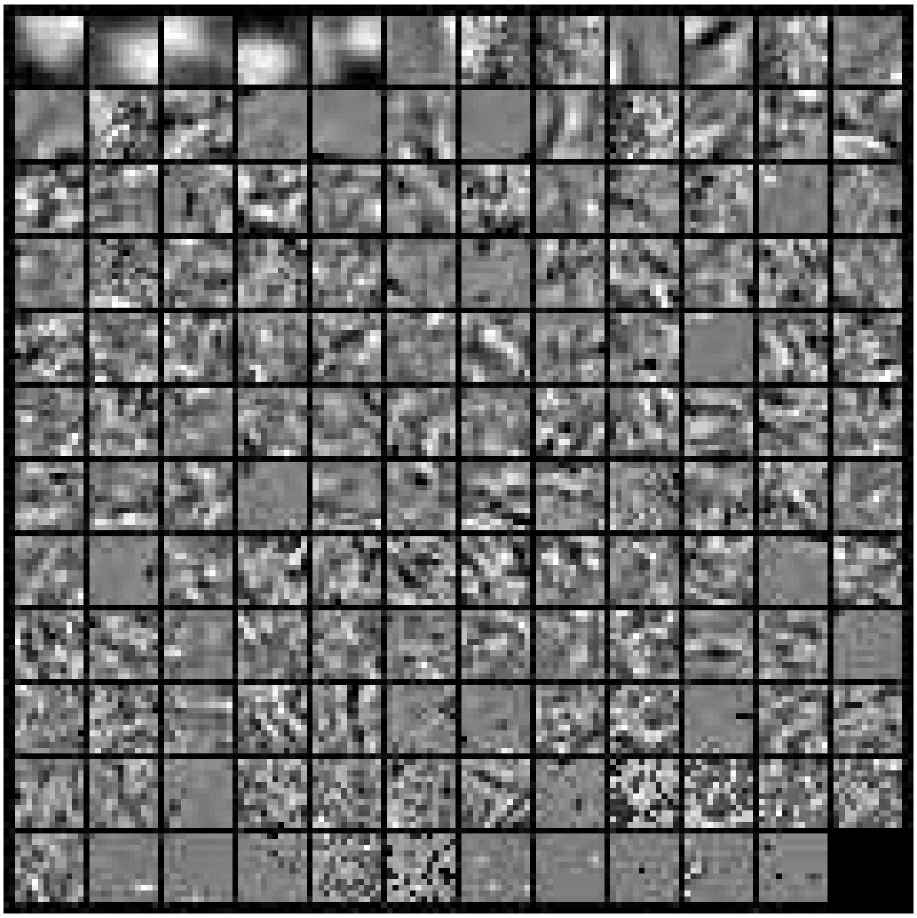


FIGURE 13. ICA basis vectors of experiment 10.

rendered flatter under the influence of errors in the data. The projections will be spread and even a small number of erroneus large values will change the estimate quite significantly because of the high powers involved. This is not true for unconcentrated flat distributions, which remain flat if the noise has no particular structure.

Experiment 11

Data	nat
Sampling	rect
Preprocessing	$\mathrm{E}\{x\}$,$\|\|x\|\|$
Number of sources	143
Algorithm and parameters	`FP` (t^3)
Summary	convergence problems,noisy
Figure	14

Again with cubic nonlinearity the deflation approach suffers from convergence problems. Otherwise the results resemble those of the previous experiment, so the preceding discussion applies here as well.

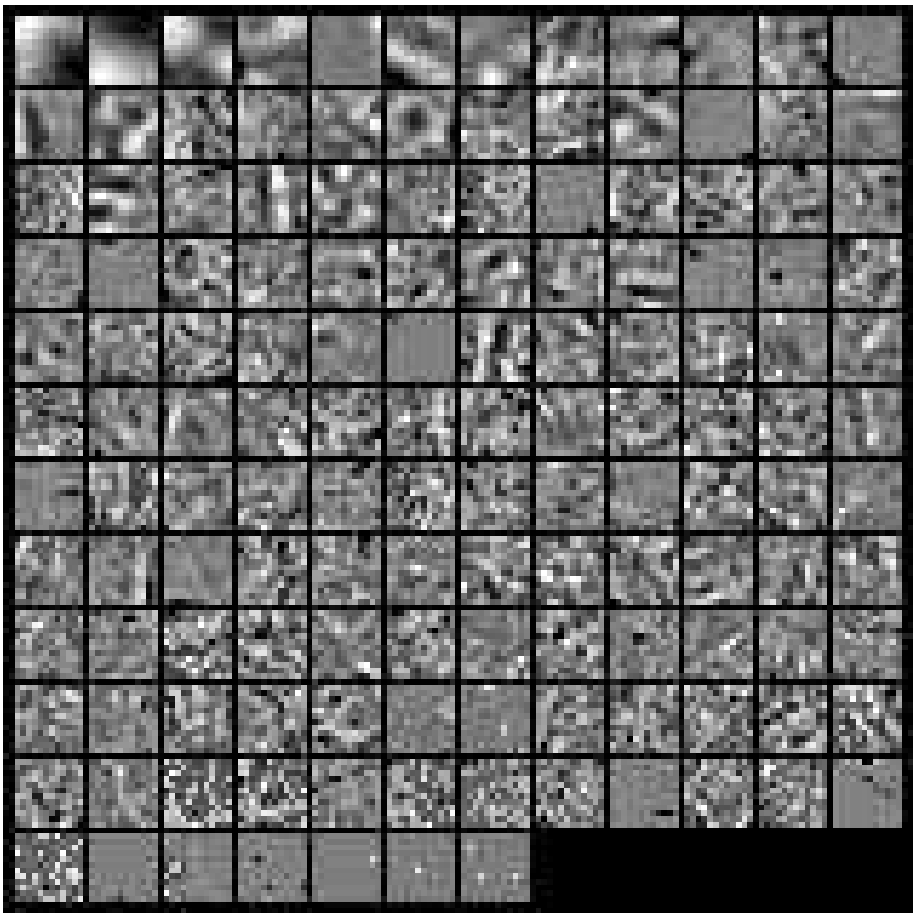


FIGURE 14. ICA basis vectors of experiment 11.

Experiment 12

Data	nat
Sampling	rect
Preprocessing	$\mathrm{E}\{x\}$,$\|\|x\|\|$
Number of sources	143
Algorithm and parameters	bigradient, $\tanh(t)$, $\mu = 10^{-3}$, $N_{\mathrm{e}} = 20$
Summary	poor definition
Figure	15

The selection of the parameters of the bigradient algorithm makes it difficult to use. However, this in itself is an important property when we consider the fitness of different algorithms. In the experiment the results may be inferior because the wrong values have been chosen. Our results show again some larger structures and localized formations, but mostly noisy basis vectors.

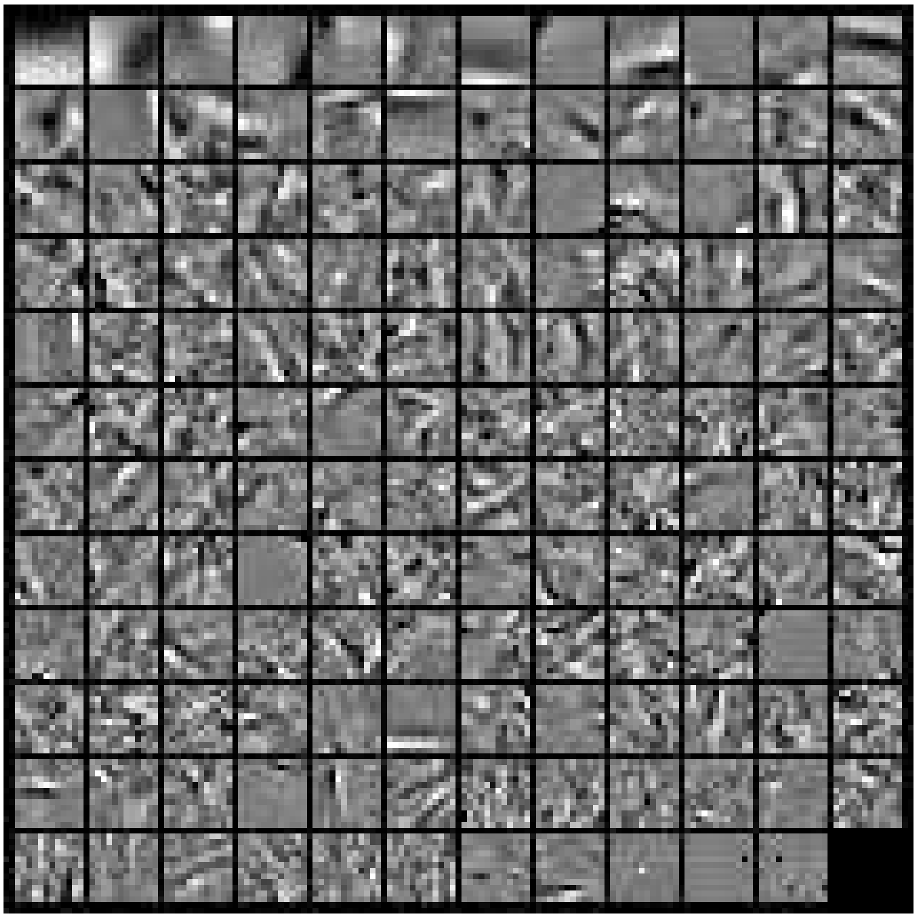


FIGURE 15. ICA basis vectors of experiment 12.

3. A short comparison of different algorithms

When starting to select an algorithm for the rest of the experiments we immediately remove the fixed point algorithm `FP` with $g(t) = t^3$ from the set of candidates because of its convergence problems. We proceed with the examination by visual inspection of different results both in the lower and higher dimensional cases. In this comparison one algorithm stands up from the others. The fixed point algorithm with symmetric orthogonalization (`SYMFP`) and nonlinearity $g(t) = \tanh(t)$ produces visually best results in both cases. Close to it in this comparison is `FP` with $g(t) = \tanh(t)$. Unfortunately we are unable to compare the results of `JADE` on this scale, because it can not be used in the higher dimensional case due to its high computational cost.

The selection of the algorithm should not be done purely on such heuristic basis. Fortunately we are able to use an independence measure based on the Kullback-Leibler distance to evaluate the results numerically. In the test we only measure pairwise independence for practical reasons. For each component pair we compute the two dimensional histogram with 10×10 bins and corresponding one dimensional histograms with 10 bins. Divided by the number of samples these give coarse approximations of the joint density function and marginal densities. If the components are pairwise independent, then the product of the estimates of marginal densities should be roughly

Table 2 Results of measuring pairwise independence.

Algorithm	$I(\hat{s})$
PCA whitening	18.8
`SYMFP`, $g(t) = \tanh(t)$	19.4
`FP`, $g(t) = \tanh(t)$	18.2
`JADE`	23.4
bigradient	20.7
`SYMFP`, $g(t) = t^3$	23.1

equal to the estimate of the joint density. The disparity between them is measured with the discrete version of the Kullback-Leibler measure (see (16)). Let $\hat{s}$ be the estimated n dimensional source vector, and let $\hat{p}_{\hat{s}_i\hat{s}_j}(k,l)$ denote the estimate of the joint density at bin (k,l) and $\hat{p}_{\hat{s}_i}\hat{p}_{\hat{s}_j}(k,l)$ the corresponding estimate of the product density. Then an overall measure $I(\hat{s})$ of the independence of the vector is computed with

$$I(\hat{s}) = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \underbrace{\sum_{k=1}^{10} \sum_{l=1}^{10} \hat{p}_{\hat{s}_i\hat{s}_j}(k,l) \log \frac{\hat{p}_{\hat{s}_i\hat{s}_j}(k,l)}{\hat{p}_{\hat{s}_i}\hat{p}_{\hat{s}_j}(k,l)}}_{\text{KL distance for one pair.}} . \tag{23}$$

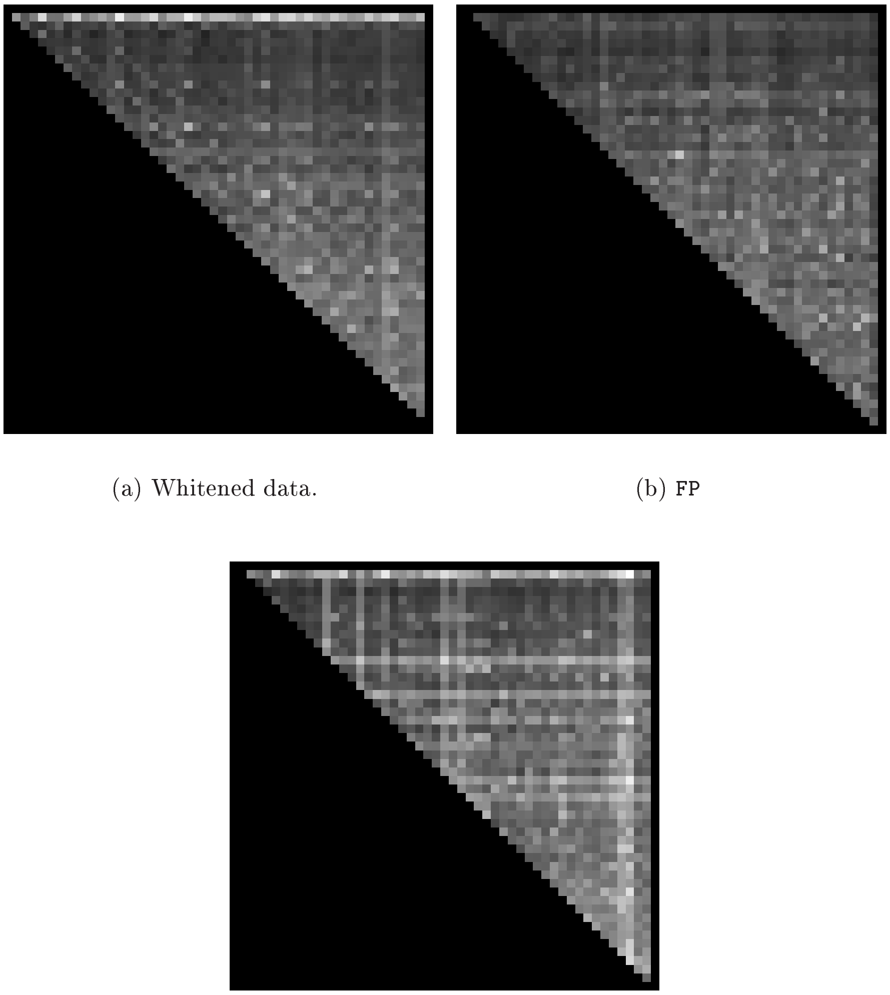
The problem with this distance is that it is not defined if any of the bins receives zero measurements. This is fixed here by adding one measurement to every slot. Overall this can not change the estimate of the density significantly, since we have 10000 samples and the number of inserted points in a 10×10 histogram is 100.

The distances will be calculated in the lower dimensional case ($n = 49$), because here we can also measure the outcomes of the experiments done with `JADE`. The results are given in Table 2.

The results can only be used as a guideline and should not be taken too seriously. Many factors affect the results: the number of bins, the placement of the bins, the amount of samples used etc. The effect of such parameters should be examined in future work. In addition the comparison is based on just one run of the algorithm, the results measure only pairwise independence and no statistical analysis of the significance of the differences has been performed.

As a guideline the measures are in favor of `FP`, with $g(t) = \tanh(t)$. The same algorithm produced "sufficiently good results" in visual inspection for both low and high dimensional data. Based on these considerations we select this algorithm to be used in consequent experiments. It is also an excellent property that we can link the fixed points of the algorithm directly to optimality conditions, as was shown in Section 3 of Chapter 2.

The computed distances do not really back up the independence of the results. In fact, most of the algorithms *increased* interdependencies between the components according to this measure. It is interesting to look at the magnitudes of the different terms in (23) for whitened data and the "best" and the "worst" algorithms, `FP` and `JADE` (Figure 16). In the plots i proceeds top→down and j left→right. Here we observe that first source corresponding to a PCA-like basis vectors has a significant dependence with the other components in both whitened data and sources produced with `JADE`. `FP` in turn has distributed the interdependencies evenly between different pairs of components.

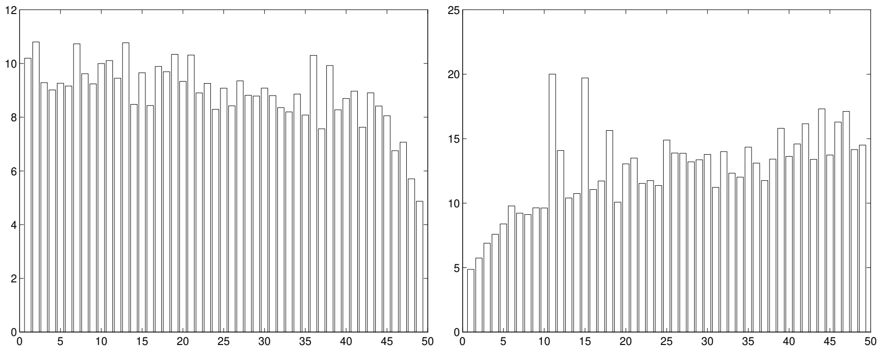


(a) Whitened data. (b) FP

(c) JADE

FIGURE 16. The magnitudes of the KL distances between different components.

However, in the previous example the placement of the bins may be an important factor. In Figure 17 we have plotted the difference between the maximum and minimum value of each component for both whitened data and the sources produced by JADE. Because the bins are positioned at even distances between the minimum and maximum values, the 11th and the 15th components of the sources found by JADE have larger binwidths. From Figure 16 we note that using our measure the same components have a relatively high dependence with all of the other components.



(a) Whitened data. (b) Sources extracted by `JADE`.

FIGURE 17. The differences between the maximum and minimum values of components.

4. Miscellaneous experiments with natural image data

Experiment 13

Data	nat
Sampling	rect
Preprocessing	$\mathrm{E}\{x\}$, $\|\|x\|\|$
Number of sources	49
Algorithm and parameters	`FP` ($\tanh(t)$, $s_{\mathrm{rand}} = 2$)
Summary	different results, qualitatively alike
Figure	18

In this experiment we study the effect of initializing the algorithm with different starting points. We notice that the results are quantitatively different, but qualitatively alike. From this we suspect that the number of different "independent" sources may be larger than the dimension of the problem.

If the results of the previous experiments somehow do correspond to most independent components, this new result should not be a big surprise. Because of the great variability of the data, rotations and translations of the localized structures should be roughly statistically equal. That is, none of the rotations of a bar should be preferred to another.

The preliminary consequences of these observations are twofold. On one hand, like stated above, these results back up our intuition about the possible sources. On the other hand, this may hinder the applicability of the results, since the results from one run seem to be somewhat arbitrary and partial.

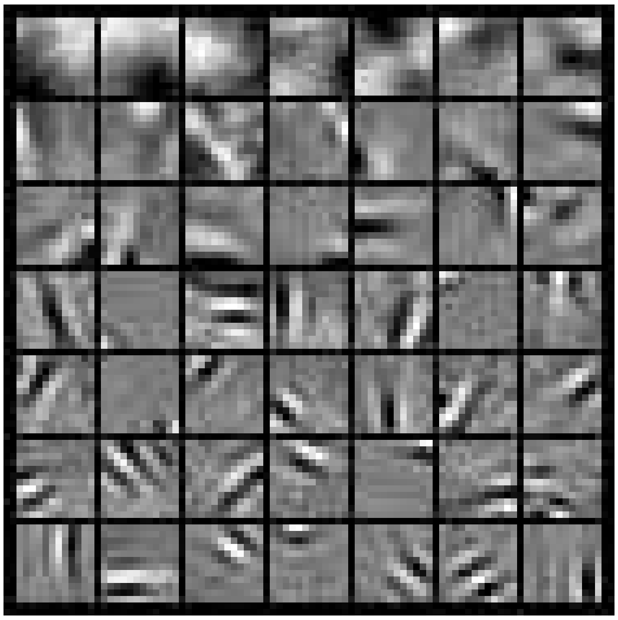


FIGURE 18. ICA basis vectors of experiment 13.

Experiment 14

Data	nat
Sampling	rect
Preprocessing	$\mathrm{E}\{x\}$
Number of sources	143
Algorithm and parameters	`FP` ($\tanh(t)$)
Summary	noisy, convergence problems
Figure	19

In previous experiments we have subtracted the mean from each sample and normalized them to unit length. The subtraction of mean is necessary to have zero mean samples. The normalization, however, is not necessary for ICA, but was introduced as a means to equalize the local variance at different areas of the image.

To test for the effect of normalization, the symmetric fixed point algorithm was run on nonnormalized data. As a result the fixed point algorithm had convergence problems and the resulting basis vectors are much less structured. So the normalization seems to be a necessary preprocessing step in order to produce meaningful features.

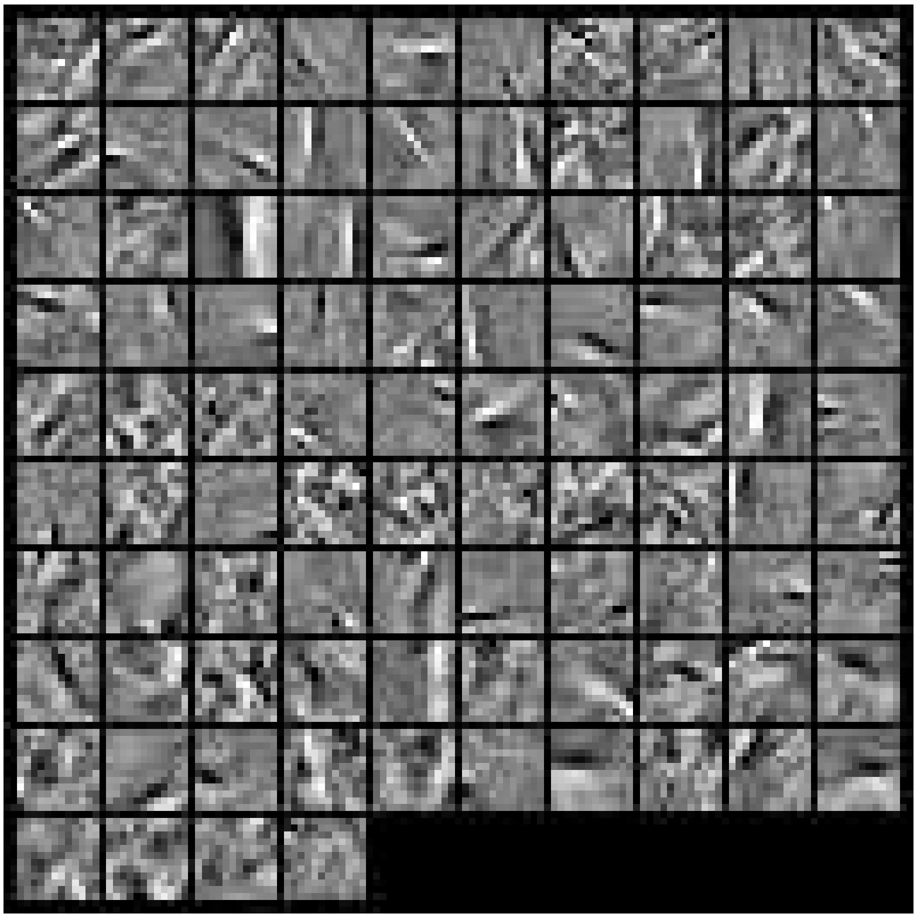


FIGURE 19. ICA basis vectors of experiment 14.

Experiment 15

Data	nat
Sampling	rad
Preprocessing	$\mathrm{E}\{x\}, \|\|x\|\|$ (l)
Number of sources	49
Algorithm and parameters	`FP` ($\tanh(t)$)
Summary	localized receptive fields, convergence problems
Figure	20

In order to examine the effect of a more biologically plausible preprocessing and sampling method we have run the algorithm with locally preprocessed and radially sampled data. A cross-section of the Gaussian filter with which each sample has been multiplied can be seen in Figure 21.

There were some convergence problems with the approach. This was probably because the efficient dimension of the problem was very small after Gaussian windowing. The emerged filters represent localized receptive fields of different orientation.

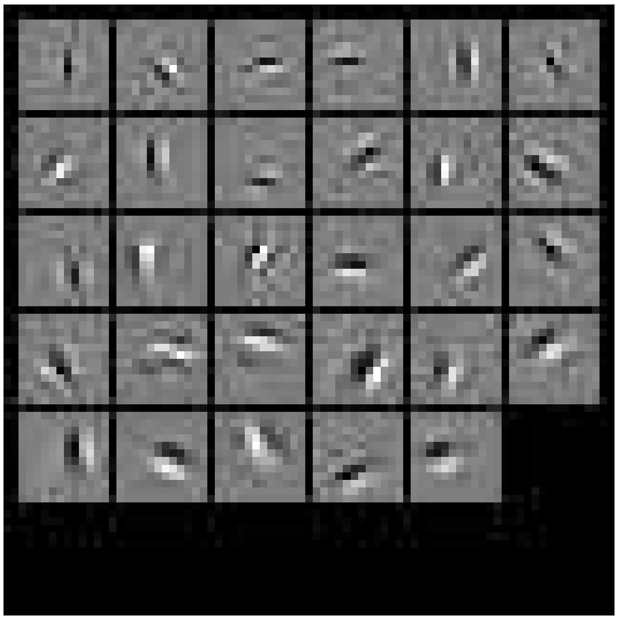


FIGURE 20. ICA basis vectors of experiment 15.

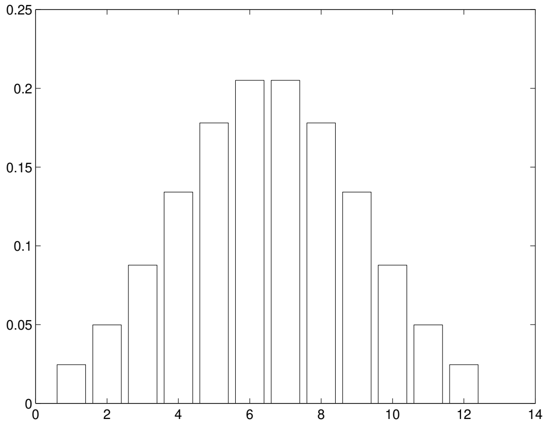


FIGURE 21. Cross section of the Gaussian windowing function.

Experiment 16

Data	tex
Sampling	rect
Preprocessing	$\mathrm{E}\{x\}, \|\|x\|\|$
Number of sources	49
Algorithm and parameters	`FP` ($\tanh(t)$)
Summary	soft forms, bars and frequencies
Figure	22

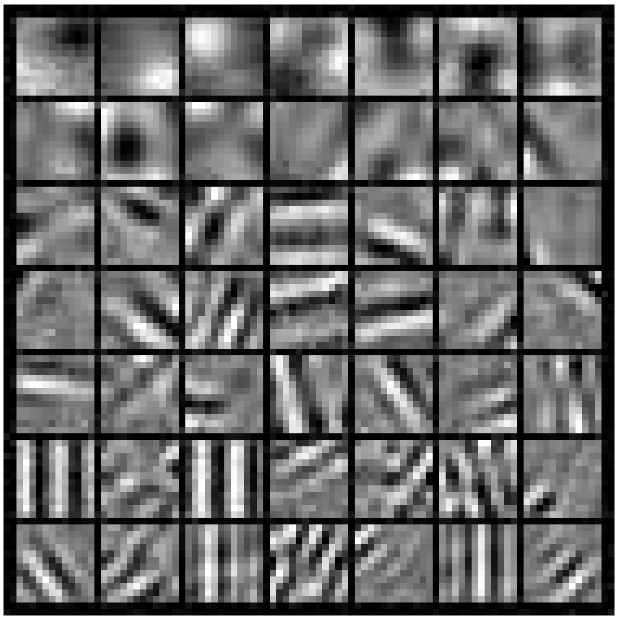


FIGURE 22. ICA basis vectors of experiment 16.

The application of the method to the set of 15 textures (see page 40) gave a somewhat different ICA basis than the natural image data. Some of the basis vectors seem to correspond to characteristics of a particular texture. It is extremely difficult to draw any conclusions of the independent components of textures, since each texture determines its own statistics and describes a related environment.

5. Some assessments of the results

5.1. Applicability of ICA model. Do the results of our experiments back up the idea that image data would follow the linear ICA model? Those facts that support this assumption are the similarity of the results obtained with different algorithms in the lower dimensional case. The results are similar among themselves and also with results obtained with sparse coding [**45**]. This would suggest that the found features actually do reduce the statistical dependencies present in image data.

Against this assumption we have the following observations.

- Some algorithms that rely on the linear model failed to converge.
- The independence test did not indicate any significant reduction of statistical dependencies.

However, the failure to converge may be caused be relatively small variations in the data, especially in the case of kurtoses. A clear majority of the algorithms did converge. As was noted above, the results of the independence test were only rough estimates possibly prone to many types of errors.

So all in all there is no convincing evidence for or against the model. This will still be a major topic in future research.

But if the linear ICA model does not hold, how could we still get such interesting results? To give a plausible explanation, we have to think of the principles behind the algorithms. Except for `JADE`, we know that the ICA algorithms used here try to maximize the deviation of the “sources” from a normal distribution, because in the linear case this is equivalent to ICA. Even if the model does not hold, this is still the basic idea of the algorithms. In fact, in the case of the fixed point algorithm `FP`, we have shown above that the fixed point fulfills the first order necessary conditions of an optimization problem. The problem is to optimize deviation from a normal distribution with the constraint that the found basis is orthogonal in the whitened coordinates. This is equivalent to a constraint that the sources are uncorrelated. So the found results represent *at least locally maximal deviations from a normal distribution under the constraint that the sources are uncorrelated.* This does not sound like a bad idea for an information processing approach. Uncorrelatedness is relatively easy to achieve and guarantees first order independence, and the nonnormality criterion tries to select the "most interesting" sources in the space of uncorrelated measurements. Indeed, methods like this have been presented in the projection pursuit literature [**19**]. What remains unknown is the connection between these criteria and independence in the case of image data.

5.2. Applicability of the algorithms. Based on the experiences gotten from the experiments we note the following observations about the ICA algorithms used.

`FP`: The generalized version of the fixed point algorithm works surprisingly well even when the original one with cubic nonlinearity fails to converge or produce any interesting results. The algorithm was rated second in the visual inspection category, and produced the most independent results when measured with our rough test. `FP` also seemed to be able to tolerate noise and high dimensional data. A very nice property of the algorithm is its direct relation to an optimization problem. The algorithm contains no parameters and converges fast, which are extremely important qualities for any algorithm. Overall, given this set of algorithms, `FP` with $g(t) = \tanh(t)$ would be my personal choice to apply first when faced with an ICA problem.

`SYMFP`: The symmetric version of the fixed point algorithm is very similar to `FP`, but the symmetric orthogonalization procedure is something of a "black box" which should be analyzed more closely. The algorithm produced visually impressive results, and its tolerance of noise and high dimensional data seemed to be good. The algorithm was rated second in our test for independence.

`JADE`: Cardosos `JADE` is a problematic algorithm to be used for high dimensional data because its need for memory and time consumption grow as the fourth power of the dimension. The basic idea of the algorithm relies heavily on the ICA model. In the low dimensional case the algorithm produced visually interesting results that did not do well in the independence test.

bigradient: The bigradient algorithm suffers from its dependence on the stepsize and penalty coefficient parameters. It is also able to extract only sources with the same kurtosis in one run. Selecting the parameters for a given problem may be quite a task. With our selection of the parameters the algorithm produced interesting but not so independent results in the lower dimensional case. In the high dimensional case the results were not that interesting, which may be caused by wrong selection of parameters, high dimension or noise.

5.3. Properties of "independent components". Generally speaking the "independent components" extracted by the ICA algorithms present meaningful features of image data. These features are

- large structures,

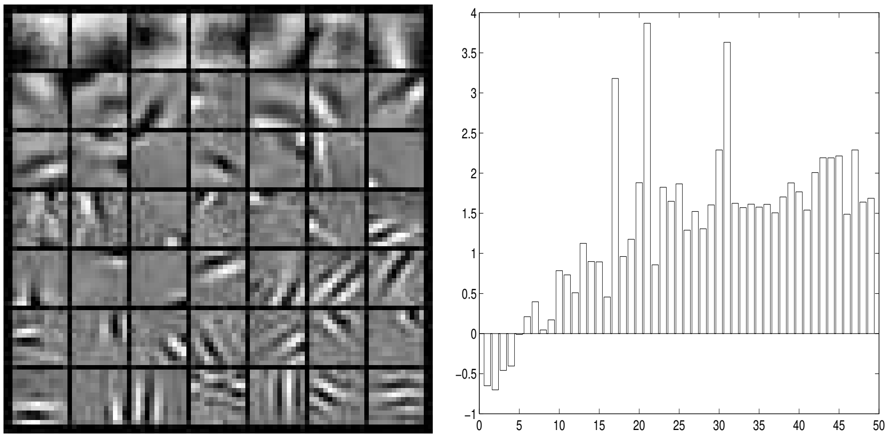


FIGURE 23. Basis vectors and kurtoses of corresponding sources.

- bars,
- lines,
- edges,
- waves,
- spots, or
- points.

This makes the results interesting from an image processing point of view, since these features emerge in a totally unsupervised manner from image data.

Viewing at the source signals of different components we notice a significant change in the kurtoses of the sources as we pass from the large structures to the localized basis vectors. The basis vectors of experiment 2 and their kurtoses are shown in Figure 23. This may be an interesting property from the point of view of image compression, as will be discussed below.

5.4. Connections with other areas of research. The ICA basis vectors presented above have similarities with the results of two major areas of research — sparse coding and wavelets.

In [**45**] the basis vectors obtained by sparse coding represent mostly bars and waves, with some vectors representing edges or spots. These are very similar to the basis vectors we have found in the low dimensional case. In sparse coding the compression of data is present in the algorithm, whereas in our case we only look for statistically independent or nonnormal sources. But in the lower dimensional case we have also compressed the data, which might explain why the results are so much alike.

Like presented above, the localized basis vectors do indeed correspond to sparse or positively kurtotic sources even if we are not using kurtosis as a contrast function. This suggests that in our measure of nonnormality sources with high values of kurtosis are still considered to be "far" from a normal distribution. The results in [**45**] also contain a constant vector or DC component. This component can not be present in our results since we have extracted the local mean from each subimage. But the larger structures, which have no direct counterparts in the sparse coding set, could be seen as generalizations of the DC component with respect to their "sparsifying" effect. This will be discussed in the next section.

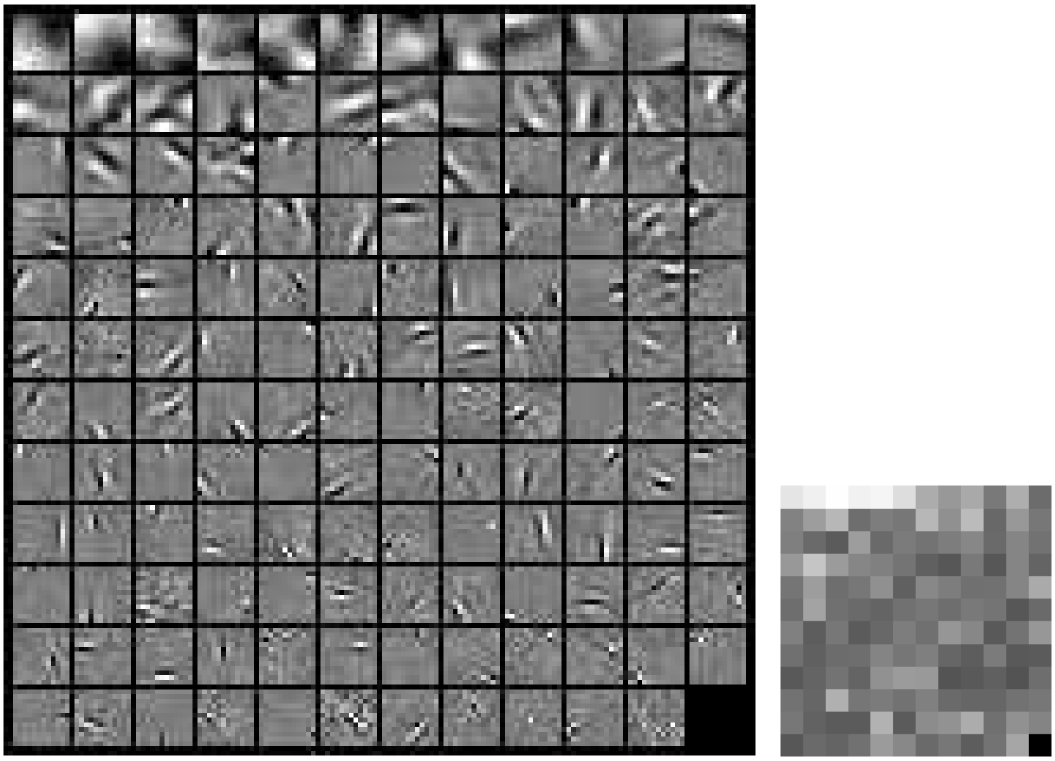


FIGURE 24. Basis vectors and their spreads. The image has been scaled for visual display so that the mean grayscale value corresponds to value zero in the original vectors. A lighter color indicates a larger value.

As was explained above in Section 4 of Chapter 3, wavelets are families of basis functions that are localized both in time and frequency [**52**]. The localization of wavelets is specialized so that wavelets that respond to low frequencies are more frequency selective, i.e., localized in the frequency domain, but also more spread in time than wavelets which respond to high frequencies.

Like in all of the presentations of different results, the ICA basis vectors in Figure 24 have been ordered by the magnitudes of their mean frequencies. In the smaller subimage we can see their spreads. In Figure 25 we can see the two-dimensional discrete cosine transformations of the basis vectors of Figure 24 along with their bandwidths.

As can be seen in Figures 24 and 25, most basis vectors obtained using the fixed point algorithm are localized in both space and frequency, that is, the energies of the vectors are localized to a subset of the space. We can also see that those basis functions representing lower frequencies are spatially more spread, but also more localized in the frequency domain than those representing high frequencies. These relations are not exact, but the correlations between the properties are clear. These are central properties of wavelets [**52, 51**], which makes this an interesting result for the analysis of similarities and connections between ICA and wavelets.

6. Applicability to image processing problems

The fact that the fixed point algorithms produce similar but quantitatively different features complicates the usage of the algorithms as automatic extractors of filters.

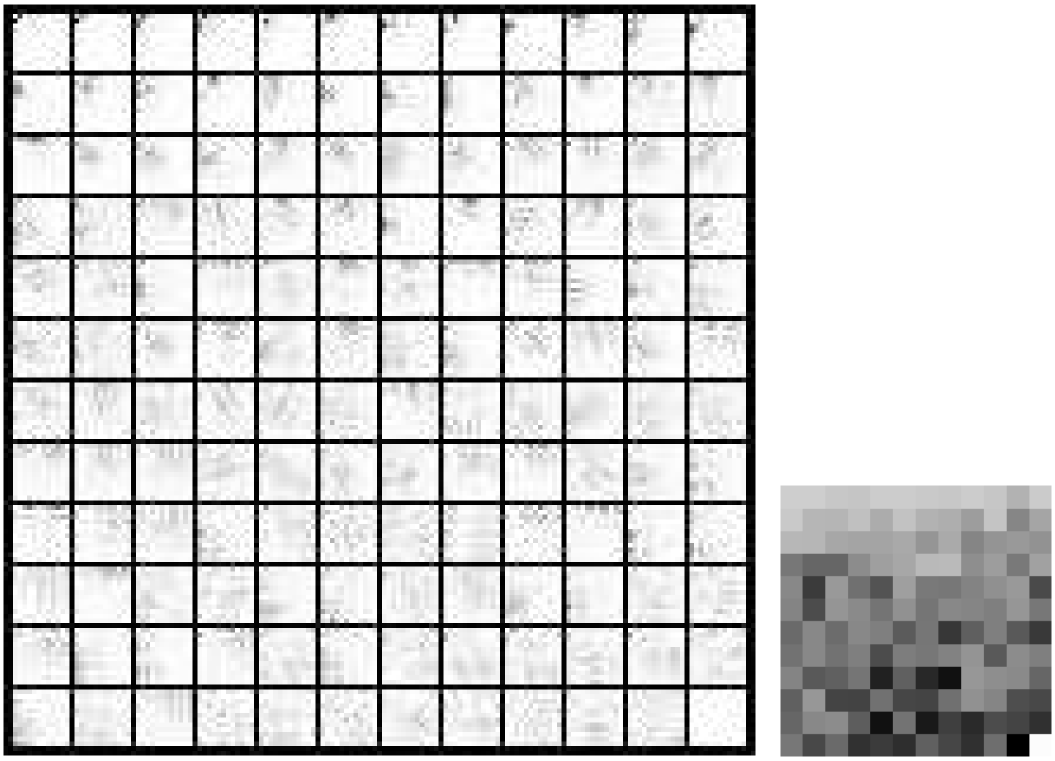


FIGURE 25. Magnitudes of the 2-dimensional discrete cosine transformations of basis vectors of Figure 24 and their bandwidths. The energies of the vectors have been equalized. Here a darker color indicates a larger value.

Looking at the basic types of features this behavior is understandable since the number of such features is much bigger than the dimension of the problem. However, obtaining a full set of features would require an indefinite number of executions of the algorithm. In addition, the obtained vector set would then be overcomplete (linearly dependent) and would not form a basis any longer. For such tasks it would be highly beneficial if some additional, possibly application dependent constraint could be added to restrict the set of solutions.

Otherwise it is a very interesting result that the method is able to extract localized feature detectors directly from image data. Very few unsupervised methods are able to do this [**17**]. A tool that would extract features automatically from images would be of great help in automating visual processes. Of course here we speak loosely about the subject, since what constitutes a useful feature is highly dependent on the application. On the other hand, features like lines and edges have been used in a great number of different problems.

There also exist interesting connections between the observed ICA basis vectors and image compression. One such connection is with wavelets and sparse coding as was discussed above. If we look at an obtained ICA basis, one of the most striking qualities is the division between large structures and localized fields. The local structures resemble those obtained with sparse coding, but the larger structures are typical of just our results. What significance could they have for image compression?

Think about sparse coding of an image subblock, i.e., coding a subimage so that few of the coefficients representing the block have large values. Then if the majority of the basis vectors with which the coding is performed consists of localized filters, some filters must represent the overall grayscale level in the image (the results in [**45**] contain the DC component). So removing the DC component sparsifies the subimage for the localized filter.

The large structures obtained in our ICA experiments might take this idea further. If we subtract for example the difference between the right and the left parts of an image in addition to the mean grayscale level, it is intuitive that the image block is sparsified further, that is, the information in the image is concentrated into more local areas. But it must be said that this is just an idea of the role that the larger structures might have in compression. Future work must be done to establish some quantifiable properties in this respect.

CHAPTER 7

Summary and conclusions

In this thesis the problem of reducing statistical dependencies in image data has been approached experimentally by applying independent component analysis to the data. While applying the algorithms we have constantly been aware that the methods have been designed for such cases where the measurements are linear mixtures of the original independent sources.

After presenting some general theory in Part 1 we have applied a set of ICA algorithms to natural image data in Part 2. For this purpose we have selected three different types of algorithms. One is based on diagonalization of cumulant matrices and the second one is a gradient descent/ascent algorithm for optimizing a contrast function. The third algorithm type is a class of fixed point algorithms for optimizing a contrast function.

The image data consisted of 15 natural images, from which a large set of 12×12 subimages was extracted and used as mixed data. We first applied the algorithms to the data so that its dimension was first reduced from 144 to 49. This number was selected by preserving 90% of the energy of the original vectors. The purpose of the reduction of dimensionality was to suppress noise and avoid some of the problems associated with high dimensional data. As a result the algorithms were able to extract meaningful features from the data. These features included lines, bars, edges, spots, wavelet-like undulating structures and larger low frequency structures.

After this the algorithms were applied to data of dimension 143 which was suspected to contain significant amounts of amplified noise after the whitening procedure. The fixed point algorithms with hyperbolic tangent nonlinearity were able to sustain their ability to find meaningful features from the data. The results of the other algorithms consisted at least partially of unclear shapes.

A simple test of independence was performed for the extracted sources to select one algorithm for some additional experiments. The measure was based on the marginal and pairwise histograms of the components and a discrete version of the Kullback-Leibler measure. The inadequateness of this test was acknowledged, and it was only used as a guideline. From observations of previous experiments and the results of the test we concluded that the fixed point algorithm based on Gram-Schmidt type of orthogonalization was the most suitable algorithm to work with.

After this some more experiments were done with different initial points, preprocessing, sampling and image data. Different initial points produced quantitatively different results with similar qualitative structures. The method was observed to be dependent on the normalization preprocessing step, or constant local variance. It was also seen to produce localized receptive fields when a more biologically plausible local form of preprocessing was used. The texture experiment was performed to find out what the components look like when the data is more restricted.

The most difficult problem, the assessment of the linear ICA model for image data was still left open in the absence of convincing evidence. Emphasis was placed on the fact that the ICA algorithms work with another objective of nonnormality which in the case of linear mixtures can often also solve the independence problem. The interesting features are optima of nonnormality in the set of uncorrelated components.

The results obtained in the lower dimensional case were very similar to those reported from the sparse coding community, although a significant difference was caused by the large structures present in the ICA results. The high positive kurtoses of the sources corresponding to localized basis vectors also indicated a connection with sparse coding. ICA basis vectors were shown to have a weak connection with wavelets with respect to localization in the spatial and frequency domains.

The applicability of the results to feature extraction was considered. A significant problem in this is that the algorithm produces only partial results from each feature set. However, the ability to extract meaningful features was considered to be a very interesting property.

All in all the work has shown that several ICA algorithms produce similar results when applied to image data. These results represent commonly used features of images. A link can be established between these results and measures of nonnormality under the constraint of uncorrelatedness. Unfortunately the connection to independence has not been solved or quantified. The next topic of research could be the development of reliable and computationally feasible measures of independence for assessing the results.

APPENDIX A

Optimization of kurtosis

The problem is to optimize

$$f(a) = \sum_{i=1}^{n} a_i^4 k_i,$$

subject to

$$h(a) = ||a||^2 - 1 = 0.$$

The derivatives of objective function and constraint are

$$\begin{aligned} f'(a) &= 4\left[a_1^3 k_1 \ \cdots a_n^3 k_n\right] \\ h'(a) &= 2a^T. \end{aligned}$$

By the method of Lagrange multipliers we know that at the optimum there exists $\lambda \in R$ so that

$$a_i^3 k_i + \lambda a_i = 0, \ i = 1, \ldots, n.$$

So either $a_i = 0$ or

$$\lambda = -a_i^2 k_i. \tag{24}$$

Denote the set of nonzero indices in a by $I(a) = \{i \mid a_i \neq 0\}$.

From equation (24) we can see that if the optimum point is a sum of more than one column of B, kurtosis along these columns have the same sign, that is, the projections of the data onto these vectors are either *positively* or *negatively kurtotic.* From now on we shall also talk about positively or negatively kurtotic vectors according to the kurtosis of projection of the data.

To study further properties of optima, we will use second order optimality conditions, in which the definiteness of the second derivative of the Lagrangian is examined in the tangent set of the constraint functions [**4**]. The second derivative or Hessian of the Lagrangian is given by

$$L''(a) = 12\,\mathrm{diag}\left(a_1^2 k_1, \ldots, a_n^2 k_n\right) + 2\lambda I = \mathrm{diag}\left(12a_1^2 k_1 + 2\lambda, \ldots, 12a_n^2 k_n + 2\lambda\right).$$

The quadratic for checking optimality is

$$H(d) = d^T L''(a) d = \sum_{i=1}^{n} \left(12a_i^2 k_i + 2\lambda\right) d_i^2 = 10 \sum_{i \in I(a)} a_i^2 d_i^2 k_i + 2\lambda \sum_{i \notin I(a)} d_i^2,$$

where we have used (24). To determine the quality of optima we have to check the definiteness of H in the set $C(a) = \{d \mid d \neq \theta \ \wedge \ h'(a)d = 0\} = \left\{d \mid d \neq \theta \ \wedge \ a^T d = 0\right\}$.

Consider first that only one coefficient differs from 0, that is, $a_j = \pm 1$, where $j \in \{1, \ldots, n\}$. Then clearly a vector $d \in C(a)$ (orthogonal to a) has this component set to 0, and

$$H(d) = 2\lambda \sum_{\substack{i=1\\ i\neq j}}^{n} d_i^2.$$

Equation (24) shows that λ has always a sign opposite to kurtosis, so for a positively kurtotic source $H(d) < 0$ and negatively kurtotic source $H(d) > 0$. (Here you can also see why it was necessary to constrain sources to have nonzero kurtoses.) These are exactly the KKT second order sufficient conditions, so positively kurtotic and negatively kurtotic column vectors of B are strict local maxima and strict local minima of f, respectively,

Now assume that more than one component of a differs from zero. Because in this case the dimension of the subspace spanned by vectors with nonzero components at $I(a)$ indices is at least two, it is possible to choose a vector $d_1 \in C(a)$ (orthogonal to a) that has nonzero components at $I(a)$ indices and zero components at other indices. Then

$$H(d_1) = 10 \sum_{i\in I(a)} a_i^2 d_i^2 k_i, \tag{25}$$

which is negative for negatively kurtotic directions and positive for positively kurtotic ones. But if $I(a) \subsetneq \{1, \dots, n\}$, that is, if some components of a are 0, then for a vector $d_2 \in C(a)$ with nonzero entries at indices not in $I(a)$

$$H(d_2) = 2\lambda \sum_{i\notin I(a)} d_i^2,$$

which is positive for negatively kurtotic directions and negative for positively kurtotic ones. Hence at such a point H is indefinite, and the point is a saddle point. The condition $I(a) \subsetneq \{1, \dots, n\}$ is fulfilled if there are both positively and negatively kurtotic sources, since by first order condition an optimum was a sum of similarly kurtotic directions.

If $I(a) = \{1, \dots, n\}$ at an optimum, then we know that all sources are of positive or negative kurtosis. Then by (25) there may be maxima or minima not corresponding to independent directions in the negatively or positively kurtotic cases, respectively.

All in all we have proved that for objective function f with constraint h columns of B up to sign are optima and they are the only optima, if there are both positively and negatively kurtotic sources. If the kurtoses of the sources are all of the same sign, there may be spurious optima.

Bibliography

[1] Amari, S.-i., Cichocki, A., and Yang, H. A new learning algorithm for blind signal separation. In *Advances in Neural Information Processing Systems (NIPS) 8*. The MIT Press, 1996, pp. 757–763.

[2] Atick, J., and Redlich, N. What does the retina know about natural scenes? *Neural Computation 4*, 2 (1992), 196–210.

[3] Barlow, H. The coding of sensory messages. *Current problems in animal behavior* (1961).

[4] Bazaraa, M. S., Hanif, D. S., and Shetty, C. M. *Nonlinear Programming – Theory and Algorithms*, 2nd ed. Series in discrete mathematics and optimization. John Wiley and Sons, New York, 1993.

[5] Bell, A., and Sejnowski, T. An information maximization approach to blind separation and blind deconvolution. *Neural Computation 7*, 6 (1995), 1129–1159.

[6] Bell, A., and Sejnowski, T. The 'independent components' of natural images are edge filters. Submitted to *Vision Research.*, 1996.

[7] Cao, X.-R., and Liu, R.-w. General approach to blind source separation. *IEEE Transactions on Signal Processing 44*, 3 (1996), 562–571.

[8] Cardoso, J.-F., and Laheld, B. Equivarian adaptive source separation. *IEEE Transactions on Signal Processing 44*, 12 (Dec. 1996), 3017–3030.

[9] Cardoso, J.-F., and Souloumiac, A. Blind beamforming for non gaussian signals. *IEE Proceedings-F 140*, 6 (Dec. 1993), 362–370.

[10] Cichocki, A., Kasprzak, W., and Amari, S.-i. Multi-layer neural networks with a local adaptive learning rule for blind separation of sources. In *Proceedings of the Research Society of Nonlinear Theory and its Applications (NOLTA) '95* (Las Vegas, USA, 1995), pp. 61–65.

[11] Cichocki, A., and Unbehauen, R. Robust neural networks with on-line learning for blind identification and blind separation of sources. *IEEE Transactions on Circuits and Systems — I : Fundamental Theory and Applications 43*, 11 (Nov. 1996), 894–906.

[12] Cohen, L. *Time-Frequency Analysis*. Prentice Hall Signal Processing Series. Prentice Hall, Englewood Cliffs, New Jersey, 1995.

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

[14] Daubechies, I. Where do wavelets come from? — A personal point of view. *Proceedings of the IEEE 84*, 4 (Apr. 1996), 510–513. Special Issue on Wavelets.

[15] Deco, G., and Brauer, W. Nonlinear higher-order statistical decorrelation by volume-conserving neural architectures. *Neural Networks 8*, 4 (1995), 525–535.

[16] Delfosse, N., and Loubaton, P. Adaptive blind source separation of independent sources: A deflation approach. *Signal Processing 45* (1995), 59–83.

[17] Field, D. What is the goal of sensory coding? *Neural Computation 6*, 4 (1994), 559–601.

[18] Field, D. Visual coding, redundancy and "feature detection". In *The Handbook of Brain Theory and Neural Networks*, M. A. Arbib, Ed. The MIT Press, Cambridge, Massachusetts, 1995, pp. 1012–1016.

[19] Friedman, J. Exploratory projection pursuit. *Journal of the American Statistical Association 82*, 397 (Mar. 1987), 249–266.

[20] Girolami, M., and Fyfe, C. Blind separation of sources using exploratory projection pursuit networks. In *Proceedings of the International Conference on the Engineering Applications of Neural Networks (EANN) '96* (1996), pp. 249–252.

[21] Girolami, M., and Fyfe, C. Higher order cumulant maximisation using nonlinear hebbian and anti-hebbian learning for adaptive blind separation of source signals. In *Proceedings of the*

IEEE/IEE International Workshop on Signal and Image Processing '96 (1996), Elsevier, pp. 141–144.

[22] Girolami, M., and Fyfe, C. Kurtosis extrema and identification of independent components : A neural network approach. In *Proceedings of the International Conference on Acoustics, Speech & Signal Processing (ICASSP) '97* (1997), pp. 3329–3333.

[23] Girolami, M., and Fyfe, C. Multivariate density factorisation for independent component analysis : An unsupervised artificial neural network approach. In *Proceedings of the International Workshop on Artificial Intelligence and Statistics (AISTAT) '97* (Fort Lauderdale, Florida, 1997), pp. 249–252.

[24] Haykin, S. *Neural Networks, A Comprehensive Foundation*. Macmillan College Publishing Company, New York, 1994.

[25] Hyvärinen, A. A family of fixed-point algorithms for independent component analysis. Tech. Rep. A 40, Helsinki University of Technology, Faculty of Information Technology, Laboratory of Computer and Information Science, 1996.

[26] Hyvärinen, A. The maximum negentropy neuron for independent component analysis, sparse coding, and projection pursuit. Submitted to a journal, 1996.

[27] Hyvärinen, A. A family of fixed-point algorithms for independent component analysis. In *Proceedings of the International Conference on Acoustics, Speech & Signal Processing (ICASSP) '97* (Munich, Germany, 1997).

[28] Hyvärinen, A. One unit contrast functions for independent component analysis: A statistical analysis, 1997. Submitted to a journal.

[29] Hyvärinen, A., and Oja, E. A fast fixed-point algorithm for independent component analysis. Tech. Rep. A 35, Helsinki University of Technology, Faculty of Information Technology, Laboratory of Computer and Information Science, 1996.

[30] Jones, M., and Sibson, R. What is projection pursuit? *Journal of the Royal Statistical Society, series A 150* (1987), 1–36.

[31] Jutten, C., and Herault, J. Blind separation of sources, part I: An adaptive algorithm based on neuromimetic architecture. *Signal Processing 24* (1991), 1–10.

[32] Kandel, E., Schwartz, J., and Jessell, T. *Essentials of Neural Science and Behavior*. Appleton & Lange, Norwalk, Connecticut, 1995.

[33] Karhunen, J., and Joutsensalo, J. Representation and separation of signals using nonlinear PCA type learning. *Neural Networks 7*, 1 (1994), 113–127.

[34] Karhunen, J., Oja, E., Wang, L., Vigário, R., and Joutsensalo, J. A class of neural networks for independent component analysis. *IEEE Transactions on Neural Networks*. To appear in 1997.

[35] Karhunen, J., and Pajunen, P. Hierarchic nonlinear PCA algorithms for neural blind source separation. In *Proceedings of the IEEE Nordic Signal Processing Symposium (NORSIG) '96* (Espoo, Finland, 1996), pp. 71–74.

[36] Karhunen, J., and Pajunen, P. Blind source separation and tracking using nonlinear PCA criterion: a least-squares approach. In *Proceedings of the IEEE International Conference on Neural Networks (ICNN) '97* (1997).

[37] Luenberger, D. G. *Optimization by Vector Space Methods*. Series in decision and control. John Wiley and Sons, New York, 1969.

[38] McEliece, R. J. *The Theory of Information and Coding*. Encyclopedia of Mathematics. Addison-Wesley, Reading, Massachusetts, 1977.

[39] Mendel, J. Tutorial on higher-order statistics (spectra) in signal processing and system theory: Theoretical results and some applications. *Proceedings of the IEEE 79*, 3 (Mar. 1991), 278–305.

[40] Moreau, E., and Macchi, O. High-order constrasts for self-adaptive source separation. *International Journal of Adaptive Control and Signal Processing* (1996), 19–46.

[41] Nadal, J.-P., and Parga, N. Nonlinear neurons in the low noise limit: a factorial code maximizes information transfer. To appear in *Network*, 1997.

[42] Nikias, C., and Mendel, J. Signal processing with higher-order spectra. *IEEE Signal Processing Magazine* (July 1993), 10–37.

[43] Oja, E. *Subspace Methods of Pattern Recognition*. Research Studies Press, Letchworth, England, 1983.

[44] Oja, E., Ogawa, H., and Wangviwattana, J. Learning in nonlinear constrained hebbian networks. In *Artificial Neural Networks*, T. Kohonen, K. Mäkisara, J. Kangas, and O. Simula, Eds., vol. 1. North-Holland, Amsterdam, 1991, pp. 385–390.

[45] Olshausen, B., and Field, D. Wavelet-like receptive fields emerge from a network that learns sparse codes for natural images. *Nature* (1996).

[46] Oppenheim, A., and Schafer, R. *Digital signal processing.* Prentice-Hall, 1975.

[47] Pajunen, P., Hyvärinen, A., and Karhunen, J. Nonlinear blind source separation by self-organizing maps. In *Proceedings of the International Conference on Neural Information Processing (ICONIP) '96* (Hong Kong, 1996), vol. 2, pp. 1207–1210.

[48] Papoulis, A. *Probability, Random Variables, and Stochastic Processes*, 3rd ed. Electrical & Electronic Engineering Series. McGraw-Hill, Inc., Singapore, 1991.

[49] Pearlmutter, B., and Parra, L. Maximum likelihood blind source separation: A context-sensitive generalization of ica. In *Advances in Neural Information Processing Systems (NIPS) 9.* The MIT Press, 1997.

[50] *Proceedings of the IEEE* (Apr. 1996), vol. 84. Special Issue on Wavelets.

[51] Strang, G., and Nguyen, T. *Wavelets and Filter Banks.* Wellesley-Cambridge Press, Wellesley, MA, 1996.

[52] Vetterli, M., and Kovačević, J. *Wavelets and Subband Coding.* Prentice Hall Signal Processing Series. Prentice Hall, Englewood Cliffs, New Jersey, 1995.

[53] Wang, L., Karhunen, J., and Oja, E. A bigradient optimization approach for robust PCA, MCA, and source separation. In *Proceedings of the IEEE International Conference on Neural Networks (ICNN) '95* (Perth, Australia, 1996), pp. 1684–1689.