Estimating overcomplete independent component bases for image windows

Aapo Hyvärinen and Mika Inki
Neural Networks Research Centre
Helsinki University of Technology
P.O. Box 9800, FIN-02015 HUT, Finland
aapo.hyvarinen@hut.fi

May 3, 2002

Abstract

Estimating overcomplete ICA bases for image windows is a difficult problem. Most algorithms require the estimation of values of the independent components which leads to computationally heavy procedures. Here we first review the existing methods, and then introduce two new algorithms that estimate an approximate overcomplete basis quite fast in a high-dimensional space. The first algorithm is based on the prior assumption that the basis vectors are randomly distributed in the space, and therefore close to orthogonal. The second replaces the conventional orthogonalization procedure by a transformation of the marginal density to gaussian.

1 Introduction

Recently, modeling image windows using statistical generative models has emerged as a new area of research [4, 12, 14, 15, 32]. Using statistical generative models enables principled derivation of methods for denoising, compression, and other image processing operations, and it is also useful for neurophysiological modeling of visual brain areas.

A fundamental generative model for low-level features of images is independent component analysis (ICA) [6, 18, 21]. In ICA the observed data is expressed as a linear transformation of latent variables that are nongaussian and mutually independent. We may express the model as

$$\mathbf{x} = \mathbf{A}\mathbf{s} = \sum_{i=1}^{n} \mathbf{a}_i s_i \tag{1}$$

where $\mathbf{x} = (x_1, x_2, ..., x_m)^T$ is the vector of observed random variables, $\mathbf{s} = (s_1, s_2, ..., s_n)^T$ is the vector of the latent variables called the independent components or source signals, and $\mathbf{A}$ is an unknown constant matrix, called the mixing matrix. In image processing, typically the x_i are pixel gray-scale values and the columns $\mathbf{a}_i$ are the basis vectors or features of the image windows.

In the classic case, we assume that the number of independent components equals the number of the observed variables, i.e. $n = m$. Exact conditions for the identifiability of the model were given in [6], and several methods for estimation of the classic ICA model have been proposed in the literature [1, 3, 5, 6, 11, 23, 31]; see [16] for a review, or [18] for a shorter introduction.

Recently, a non-classic modification of the model, where it is assumed that the number of independent components is larger than the number of observed variables ($n > m$), has attracted the attention of a number of researchers [29, 33, 35]. Such a model is especially interesting when ICA is used for image modeling, because it leads to decomposition of image windows that is closely related to overcomplete wavelet bases (see [33]), which seem to be in some ways superior to ordinary wavelet bases. Basically, the larger number of independent

components in the model means that we have a larger 'dictionary' from which to construct the representation. The dictionary consists of the basis vectors that are given as columns $\mathbf{a}_i$ of the mixing matrix $\mathbf{A}$. Using an overcomplete basis may also allow for some invariances (e.g. with respect to translation) in the representation [37].

Some methods have already been proposed for estimating the mixing matrix in the ICA model with $n > m$, a problem often called estimation of an overcomplete ICA basis. Some methods are reviewed below. A drawback with most proposed methods is that they are computationally very demanding. This is basically because the model then becomes a model with missing data: the computation of the likelihood is not straightforward as in the basic case. In fact, the evaluation of the likelihood contains an integral and even reasonable approximations of that integral are hard to compute [29]. On the other hand, since these methods are usually applied to data of very high dimensions, it would be very useful to have an estimation method that can cope with very large dimensions with a moderate computational load.

In this paper, we propose two methods for approximate estimation of the ICA model with overcomplete bases. The methods are computationally efficient when compared with existing methods, and appear to give quite good approximations of the optimal estimates.

2 Review of existing methods

First, we provide a short review of existing methods for estimating overcomplete bases.

2.1 Estimation of the independent components

An interesting property connected with overcomplete bases is that the values of the independent components cannot be exactly recovered even if the mixing matrix is known. This is because the mixing matrix $\mathbf{A}$ is not invertible. Therefore, even after estimating the mixing matrix, the problem of optimal estimation of the realizations of the independent components needs to be solved. This is an important problem that has already been treated in the wavelet literature. We shall not treat it in detail here, see [36] instead. However, many methods for estimating the mixing matrix use as subroutines methods that estimate the independent components for a known mixing matrix. Therefore, we shall first very briefly treat methods for reconstructing the independent components, assuming that we know the mixing matrix.

Let us denote by m the number of mixtures and by n the number of independent components. Thus, the mixing matrix has size $m \times n$ with $n > m$, and therefore it is not invertible. The simplest method of estimating the independent components would be to use the pseudoinverse of the mixing matrix. This yields

$$\hat{\mathbf{s}} = \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{x} \tag{2}$$

In some situations, such a simple pseudoinverse gives a satisfactory solution, but in many cases we need a more sophisticated estimate.

A more sophisticated estimator of $\mathbf{s}$ can be obtained by maximum a posteriori estimation [33, 29, 10]. We can write the posterior probability of $\mathbf{s}$ as follows [33]:

$$p(\mathbf{s}|\mathbf{x}, \mathbf{A}) = 1_{\mathbf{x}=\mathbf{A}\mathbf{s}} \prod_i p_i(s_i) \tag{3}$$

where $1_{\mathbf{x}=\mathbf{A}\mathbf{s}}$ is an indicator function that is 1 if $\mathbf{x} = \mathbf{A}\mathbf{s}$ and 0 otherwise. The (prior) probability densities of the independent components are given by $p_i(s_i)$. Thus, we obtain the maximum a posteriori estimator of $\mathbf{s}$ as

$$\hat{\mathbf{s}} = \arg \max_{\mathbf{x}=\mathbf{A}\mathbf{s}} \sum_i \log p_i(s_i). \tag{4}$$

Alternatively, we could assume that there is noise present as well. In this case, we get a posterior of the form

$$\log p(\mathbf{s}|\mathbf{x}, \mathbf{A}) = -\frac{1}{2\sigma^2}\|\mathbf{A}\mathbf{s} - \mathbf{x}\|^2 + \sum_{i=1}^{n} \log p_i(s_i) + C \tag{5}$$

where C is an irrelevant constant, and the covariance of the noise is assumed to be of the form $\sigma^2\mathbf{I}$.

The problem with the maximum a posteriori estimator is that it is not easy to compute. This optimization cannot be expressed as a simple function in analytic form in any interesting case. It can be obtained in closed form if the s_i have a gaussian distribution: In this case the optimum (for the no noise case) is given by the pseudoinverse in (2). However, since ICA with gaussian variables is of little interest, the pseudoinverse is not a very satisfactory solution in many cases.

In general, therefore, the estimator given by (4) can only be obtained by numerical optimization. A gradient ascent method can be easily derived. One case where the optimization is easier than usual is when the s_i have a Laplacian distribution:

$$p_i(s_i) = \frac{1}{\sqrt{2}} \exp(-\sqrt{2}|s_i|). \tag{6}$$

Ignoring uninteresting constants, we have

$$\hat{\mathbf{s}} = \arg \min_{\mathbf{x}=\mathbf{A}\mathbf{s}} \sum_i |s_i| \tag{7}$$

which can be formulated as a linear program and solved by classical methods for linear programming, see e.g. [29].

The use of the Laplacian distribution is well justified in feature extraction, where the components are supergaussian (also called leptokurtic or sparse [16]). Using the Laplacian density also leads to an interesting phenomenon: The ML estimator gives coefficients $\hat{s}_i$ of which only m are non-zero. Thus, only the minimum number of the components are activated. Thus we obtain a sparse decomposition in the sense that the components are quite often equal to zero.

Another direction of research is given by Monte Carlo methods. One such method, Gibbs sampling, has been used for estimating the $\mathbf{s}$ in [34]. The advantage of this method is that richer models of the distributions of the s_i can be used. The drawback is that the method is computationally quite demanding.

It may seem at first glance that it is useless to try to estimate the independent components by these methods because they cannot be estimated exactly in any case due to the non-invertibility of the mixing matrix. This is not so, however; due to this phenomenon of sparsity, maximum a posteriori estimation is very useful. In fact, in the case where the independent components are very supergaussian or sparse, most of them are very close to zero because of the large peak of the pdf at zero. Thus, those components that are not zero may not be very many, and the system may be invertible for those components. If we first determine which components are likely to be clearly non-zero, and then invert that part of the linear system, we may be able to get quite accurate reconstructions of the independent components. This is done implicitly in the maximum a posteriori estimation method. For example, assume that there are three speech signals mixed into two mixtures. Since speech signals are practically zero most of the time (which is reflected in their strong supergaussianity), we could assume that only two of the signals are non-zero at the same time, and succesfully reconstruct those two signals [27]. In the same way, image decompositions often assume that only a limited number of components is active at any one time, see e.g. [36].

2.2 Estimation of the mixing matrix

Now we return to our main subject, the estimation of overcomplete bases.

2.2.1 Maximizing joint likelihood

To estimate the mixing matrix, one can use maximum likelihood (ML) estimation. In the simplest case of ML estimation, we use the joint likelihood of $\mathbf{A}$ and the realizations of the s_i, and maximize it with respect to all these variables. The joint likelihood for T observations $\mathbf{s}(t), t = 1, ..., T$ can be easily derived from (5):

$$\log L(\mathbf{A}, \mathbf{s}(1), ..., \mathbf{s}(T)) = -\sum_{t=1}^{T} \left[\frac{1}{2\sigma^2} \|\mathbf{A}\mathbf{s}(t) - \mathbf{x}(t)\|^2 + \sum_{i=1}^{n} \log p_i(s_i(t)) \right] + C \tag{8}$$

Maximization of (8) with respect to $\mathbf{A}$ and s_i could be accomplished by a global gradient ascent with respect to all the variables [33]. This was probably the first method that enabled the estimation of overcomplete bases.

Another approach to maximization of the likelihood is to use an alternating variables technique [10], in which we first compute the ML estimates of the $\mathbf{A}$ for fixed $s_i(t)$ and then, using this new $\mathbf{A}$, we compute the ML estimates of the $s_i(t)$, and so on. The ML estimate of the $s_i(t)$ for a given $\mathbf{A}$ is given by the methods of the preceding section. The ML estimate of $\mathbf{A}$ for given $s_i(t)$ can be computed as:

$$\mathbf{A} = (\sum_t \mathbf{x}(t)\mathbf{x}(t)^T)^{-1} \sum_t \mathbf{x}(t)\mathbf{s}(t)^T \tag{9}$$

This algorithm needs some extra stabilization, however. For example, normalizing the estimates of the s_i to unit norm is necessary. Further stabilization can be obtained by first whitening the data. Then we have (considering infinitely small noise)

$$E\{\mathbf{x}\mathbf{x}^T\} = \mathbf{A}\mathbf{A}^T = \mathbf{I} \tag{10}$$

which means that the rows of $\mathbf{A}$ form an orthonormal system. This orthonormality could be enforced after every step of (9) for further stabilization.

2.2.2 Maximizing likelihood approximations

Maximization of the joint likelihood is a rather crude method of estimation. From a Bayesian viewpoint, what we really want to maximize is the *marginal* posterior probability of the mixing matrix. Thus, the posterior should be marginalized with respect to $\mathbf{s}$.

The marginal posterior cannot be easily computed, however, and approximations must be used. A rather simple modification of joint likelihood estimation can thus be obtained by using a Laplace approximation of the posterior distribution of $\mathbf{A}$. This improves the stability of the algorithm, and has been succesfully used for estimation of overcomplete bases from image data [28], as well as for separation of audio signals [27]. For details on the Laplace approximation, see [29]. An alternative for the Laplace approximation is provided by ensemble learning [26], but this has not yet been applied to this particular problem.

One direction of research is to use an expectation-maximization (EM) algorithm [30, 2]. Using gaussian mixtures as models for the distributions of the independent components, the algorithm can be derived in analytical form. The problem is, however, that its complexity grows exponentially with the dimension of $\mathbf{s}$, and thus it can only be used in small dimensions. Suitable approximations of the algorithm might alleviate this limitation [2].

A very different approximation of the likelihood method was derived in [10], in which a form of competitive neural learning was used to estimate overcomplete bases with supergaussian data. This is a computationally powerful approximation that seems to work for certain data sets. The idea is that the extreme case of sparsity or supergaussianity is encountered when at most one of the ICs is non-zero at any one time. Thus we could simply assume that only one of the components is non-zero for a given data point, for example the one with the highest value in the pseudo-inverse reconstruction. This is not a realistic assumption in itself, but it may give an interesting approximation of the real situation in some cases. The validity of such a strong approximation still needs to be explored, however.

3 Approximate estimation by quasi-orthogonality

The maximum likelihood methods discussed in the preceding section give a well justified approach to ICA estimation with overcomplete bases. The problem with most of the methods in the preceding section is that they are computationally quite slow. A typical application of ICA with overcomplete bases is, however, feature extraction. In feature extraction, we usually have spaces of very high dimensions, and computational considerations may severely limit the class of methods that we can use. Therefore, we introduce in this and the following section methods that are a bit more heuristically justified, but have the advantage of being not more expensive computationally than methods for basic ICA estimation.

Our approximative approach is justified by the fact that in feature extraction for many kinds of natural data, the ICA model is only a rather coarse approximation. In particular, the number of potential "independent components" seems to be infinite: The set of such components is closer to a continuous manifold (parameterized by location, orientation, frequency, etc.) than a discrete set. One evidence for this is that in image feature extraction, basic ICA estimation methods give different basis vectors when started with different initial values, and the number of components thus produced does not seem to be limited.

Any basic ICA estimation method thus gives a rather arbitrary collection of components which are somewhat independent, and have sparse (supergaussian or leptokurtic) marginal distributions. We could argue, therefore, that it is the sparseness that is important, and the exact dependence relations between the components are secondary. In fact, recent research has revealed important dependencies between the estimated components [14, 15, 38, 39].

In the following, we propose two methods that give bases for overcomplete sparse decompositions. The method in this section is based on a Bayesian prior on the mixing matrix, and the method in the next section uses a method of gaussianization that has been proposed in projection pursuit literature. The main computational advantage of the algorithms stems from the fact that we do not compute estimates of the s_i in every step, as in most algorithms. Although here we discuss only the case of sparse (supergaussian) components, these methods can also be used on data with subgaussian sources [20].

3.1 Sparse approximately uncorrelated decompositions

Let us assume, for simplicity, that the data is prewhitened as a preprocessing step, as in most ICA methods. Then, if the basis were not overcomplete, the mixing matrix could be constrained orthogonal, and the independent components are simply given by the dot-products of the whitened data vector $\mathbf{z}$ with the basis vectors $\mathbf{a}_i$.

Due to the considerations in the preceding subsection, we assume in our approach that what is needed is a collection of basis vectors which has the following two properties.

1. The dot-products $\mathbf{a}_i^T\mathbf{z}$ of the observed data with the basis vectors have maximally sparse (supergaussian) marginal distributions.
2. The $\mathbf{a}_i^T\mathbf{z}$ should be approximately uncorrelated ("quasi-uncorrelated"). Equivalently, the vectors $\mathbf{a}_i$ should be approximately orthogonal ("quasi-orthogonal").

A decomposition with these two properties seems to capture the essential properties of the decomposition obtained by estimation of the ICA model. Such decompositions could be called sparse approximately uncorrelated decompositions.

3.2 The phenomenon of quasi-orthogonality

It is clear that it is possible to find highly overcomplete basis sets that have the first property of the two given above. Classic ICA estimation is usually based on maximizing the sparseness (or, in general, nongaussianity) of the dot-products, so the existence of several different classic ICA decompositions for a given image data set shows the existence of decompositions with the first property.

What is not obvious, however, is that it is possible to find strongly overcomplete decompositions such that the dot-products are approximately uncorrelated. The main point here is that this is possible because of the phenomenon of quasi-orthogonality.

Quasi-orthogonality [22, 24, 25] is a somewhat counterintuitive phenomenon encountered in very high-dimensional spaces. In a certain sense, there is much more room for vectors in high-dimensional spaces. The point is that in an n-dimensional space, where n is large, it is possible to have (say) $2n$ vectors that are practically orthogonal, i.e. their angles are close to 90 degrees. In fact, when n grows, the angles can be made arbitrarily close to 90 degrees. This must be contrasted with small-dimensional spaces: If, for example, $n = 2$, the even the maximally separated $2n = 4$ vectors exhibit angles of 45 degrees.

For example, in image decomposition, we are usually dealing with spaces whose dimensions are of the order of 100. Therefore, we can find decompositions of, say, 200 vectors, such that the vectors are quite orthogonal, with all the angles between basis vectors staying above 80 degrees.

3.3 Derivation of quasi-orthogonal prior

Our goal is now to formulate a Bayesian prior for quasi-orthogonality. Such a prior would give high probabilities to mixing matrices with quasi-orthogonal columns. The starting point is to assume that the elements of the basis vectors are drawn randomly, independently from each other.

We now calculate the probability density for the dot product between two randomly and independently drawn basis vectors: $\mathbf{a}_i^T \mathbf{a}_j$. Assume that the basis vectors are of unit length. In this case these basis vectors can be considered to be points on the surface of an m-dimensional unit sphere. The volume of an m-dimensional sphere of radius r is

$$V(r) = C_m r^m, \tag{11}$$

where $C_m = \frac{\pi^{\frac{m}{2}}}{\Gamma[\frac{m}{2}+1]}$ is a constant. When we take the portion of the surface of the m-dimensional unit sphere that is within an angle of α to a fixed vector, and project this onto a hyperplane orthogonal to this vector, we get an $m-1$ dimensional ball of radius $\sin(\alpha)$. When we take the derivative of this with respect to the radius we get the length of the boundary. Therefore, the infinitesimal area of a band of width $\mathrm{d}\alpha$ at an angle α on the surface of an m-dimensional sphere, which gives us the probability density function of α, can be computed as

$$p_\alpha(\alpha)\mathrm{d}\alpha = c_m \sin^{m-2}(\alpha)\mathrm{d}\alpha \tag{12}$$

Here, the surface area is scaled to one using the constant $c_m = \frac{m-1}{m} \frac{\Gamma[\frac{m}{2}+1]}{\sqrt{\pi}\Gamma[\frac{m-1}{2}+1]}$. By denoting the dot product as x, i.e. $\alpha = \arccos(x)$, we get the following probability density for the dot-product:

$$p_x(x) = c_m (1-x^2)^{\frac{m-3}{2}} \tag{13}$$

In this way we get a prior probability for the mixing matrix $\mathbf{A}$, assuming that all the dot products $\mathbf{a}_i^T \mathbf{a}_j$ are independent:

$$p(\mathbf{A}) = \prod_{i<j} c_m (1-(\mathbf{a}_i^T \mathbf{a}_j)^2)^{\frac{m-3}{2}} \tag{14}$$

Strictly speaking, the dot products are not quite independent of each other in this space. For example, if there were m orthogonal vectors in this space, the probability for any of the other $n-m$ vectors to be orthogonal to all these vectors would be zero. In most cases, however, this approximation appears to be good enough.

3.4 Posterior of mixing matrix

To use the above quasi-orthogonal prior given in (14) in the ICA likelihood, we make another approximation. Consider the likelihood for ordinary (not overcomplete) ICA:

$$\log L(\mathbf{A}) = \sum_t \sum_{i=1}^n \log p_i(\mathbf{w}_i^T \mathbf{z}(t)) + T \log |\det \mathbf{W}|. \tag{15}$$

where the $\mathbf{w}_i^T$ are the rows of the inverse of $\mathbf{A}$, and $\mathbf{W} = \mathbf{A}^{-1}$. If the mixing matrix is constrained orthogonal, we have in fact $\mathbf{W} = \mathbf{A}^T$.

The last term $\log |\det \mathbf{W}|$ gives the scaling of the probability mass when the linear transformation given by $\mathbf{W}$ is performed. When the previously mentioned assumptions about $\mathbf{A}$ hold, i.e. the basis vectors $\mathbf{a}_i$ are quasi-orthogonal (randomly distributed) and of unit length, the scaling of the probability made by $\mathbf{A}$ or its inverse can be considered roughly constant. In practice, the purpose of the last term $\log |\det \mathbf{W}|$ is basically to make to $\mathbf{w}_i$ more or less orthogonal, which is equivalent to making $\mathbf{A}$ orthogonal. In fact, this term disappears if the $\mathbf{a}_i$ are constrained orthogonal. Therefore, assuming the vectors $\mathbf{a}_i$ quasi-orthogonal, we discard the last term in (15).

In this form, the method can be extended to the overcomplete case. The determinant can be computed for square matrices only, whereas the quasi-orthogonality measure in (14) can be calculated for any $\mathbf{A}$. [1]

The probability for $\mathbf{z}$ given $\mathbf{A}$ can be approximated as follows:

$$p(\mathbf{z}(t)|\mathbf{A}) \approx C \prod_{i=1} p_{y_i}(\mathbf{a}_i^T \mathbf{z}(t)) \tag{16}$$

where C is a constant and the variable y_i is the dot product between $\mathbf{a}_i$ and $\mathbf{z}$. As in ordinary ICA estimation [17], it seems that the exact form of p_{y_i} is not that important, as long as it is supergaussian when y_i is supergaussian, and subgaussian when y_i is subgaussian.

The posterior probability for the problem can be written as

$$p(\mathbf{A}|\mathbf{z}) = \frac{p(\mathbf{z}|\mathbf{A})p(\mathbf{A})}{p(\mathbf{z})} \tag{17}$$

Here $p(\mathbf{z})$ is constant with respect to $\mathbf{A}$. Note that $p(\mathbf{A})$ now assigns a higher probability to quasi-orthogonal matrices (when the dimension of the space is large), so that the assumption of quasi-orthogonality of the basis vectors holds and the approximation (16) for $p(\mathbf{z}|\mathbf{A})$ can be used.

Finally, we thus have the following *posterior probability for* $\mathbf{A}$:

$$\log p(\mathbf{A}|\mathbf{z}(t), t = 1, ..., T) \approx \sum_t \sum_{i=1}^{n} \log p_{y_i}(\mathbf{a}_i^T \mathbf{z}(t)) + \alpha T \sum_{i<j} \log(1 - (\mathbf{a}_i^T \mathbf{a}_j)^2) + \text{const.} \tag{18}$$

Here α is a constant that is affected not only by m, but also by the approximations we have made. In practice, we do not attempt to find a formula for computing α, but instead adjust it empirically. This allows us to give different strengths for the prior.

In the following, we maximize the posterior in (18) to estimate $\mathbf{A}$, and we denote the maximizing argument by $\hat{\mathbf{A}}$. The difference to maximum likelihood estimation of the classic ICA model (i.e. when $\mathbf{A}$ is square) simply is that $|\det(\mathbf{A})|$ is replaced by $p(\mathbf{A})$.

Previously one of the authors proposed a modification of FastICA to perform a similar estimation by quasi-orthogonality [13], but the quasi-orthogonality measure in the present method has been derived from first principles and it seems that the present method gives better estimates. In [19] we approximated $\log p(\mathbf{A})$ more heuristically with a power function of the dot products, which also seemed to work, although it produced no low-frequency basis vectors for image data.

3.5 Simulations

First, we tried our method on simulated data. We mixed 40 independent components with Laplacian distributions into a 20 dimensional data space, i.e. $\mathbf{A}$ was a matrix of size 20×40. The sample size was 50000.

A general problem in estimating overcomplete bases is that components whose contributions to the data are very small (as measured by the norm of the corresponding column of $\mathbf{A}$) are very difficult to estimate. To avoid this problem, the standard deviations of the sources were uniformly distributed between 0.75 and 1.5. The basis vectors were uniformly distributed on the surface of the 20-dimensional unit sphere.

As a preprocessing step, the data was whitened. We then maximized the posterior in (18) by gradient ascent. The parameter α was set to the value of 0.34 and $\log p_{y_i}(y) = -\log\cosh y$. (A rescaling of this density function was implicitly included in α.)

To investigate the quasi-orthogonality of the obtained basis vectors (in the whitened space), we can look at the minimum angle between one basis vector from the rest. This minimum angle can be calculated from the maximum of the absolute values of the dot products between the basis vector in question and the rest. These angles are depicted in Fig. 1. Note that all of these angles are above 60 degrees, which shows good quasi-orthogonality. The

[1] One should note, however, that generally $p(\mathbf{A})$ and $\det(\mathbf{A})$ do not quite behave similarly, even when $\mathbf{A}$ is a square matrix. The determinant goes to zero if there exists a dimension not spanned by the basis vectors, whereas $p(\mathbf{A})$ goes to zero if any two basis vectors point in the same direction. So, for a square $\mathbf{A}$, $p(\mathbf{A}) = 0 \Rightarrow \det(\mathbf{A}) = 0$, but not vice versa.

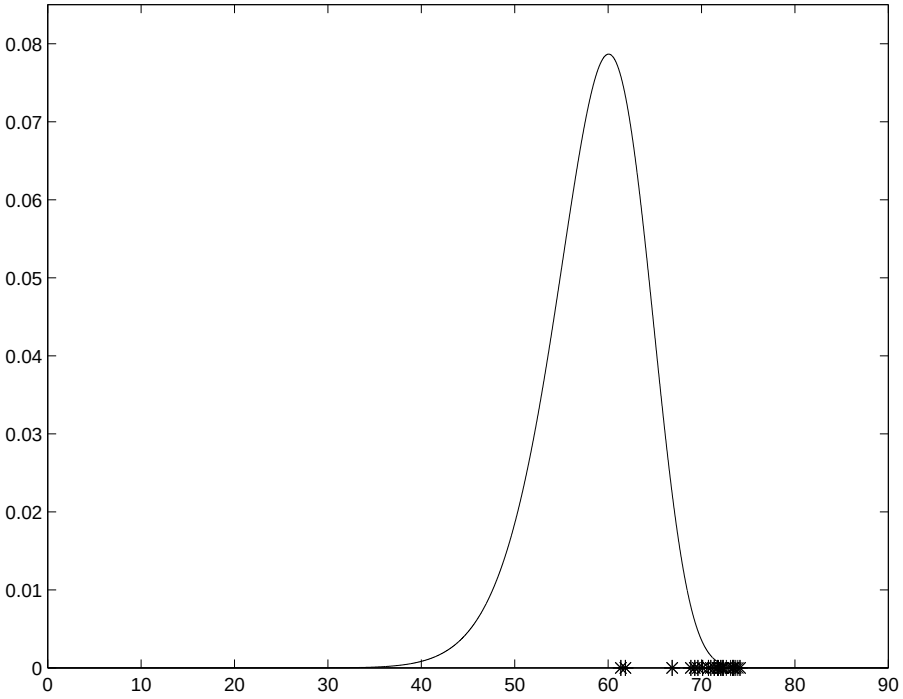

Figure 1: The quasi-orthogonality of the estimated basis vectors when 40 independent components are mixed into a 20 dimensional space, using the quasi-orthogonalizing prior. Asterisks: The minimum angles between the estimated basis vectors with Laplacian distributed sources. Solid line: Probability density that the minimum angle would have if the vectors were really generated randomly.

probability density shown by the solid line in Fig. 1 for comparison gives the distribution that one would expect for these angles if the estimated basis vectors were distributed randomly in the space. One can see that in fact, the obtained vectors are even more orthogonal than corresponding random vectors. Note that even though we generated the mixing matrix randomly, we then whitened the data, which quasi-orthogonalizes the basis vectors by a small amount.

The other thing of interest is, of course, how close the estimated basis vectors are to the original basis vectors. This can determined by looking at the absolute value of the elements of $\mathbf{A}^T\hat{\mathbf{A}}$. Some care must be taken in evaluating this matrix. If we took the maximum dot products from each column of the matrix, results where several estimated vectors are close to the same original vector might look good. Or, if we took the maximum dot products in each row, we might get results that look satisfactory in cases where an estimated basis vector is in the middle of two original basis vectors. To avoid these problems, we use a matching approach. We find the best matches between estimated basis vectors and the original ones: First we find largest dot product, remove both the real and the estimated basis vectors corresponding to it, and repeat this until we have a "match" for each basis vector.

The angles (in degrees) between the estimated basis vectors and the matched original basis vectors are shown in Fig. 2. We can see that nearly all the components were quite correctly estimated. The results are far superior to those obtained by ordinary FastICA [11]. Note that no subset of the basis vectors was in an orthogonal configuration in the whitened space, which partly explains why the results with ordinary FastICA are so poor.

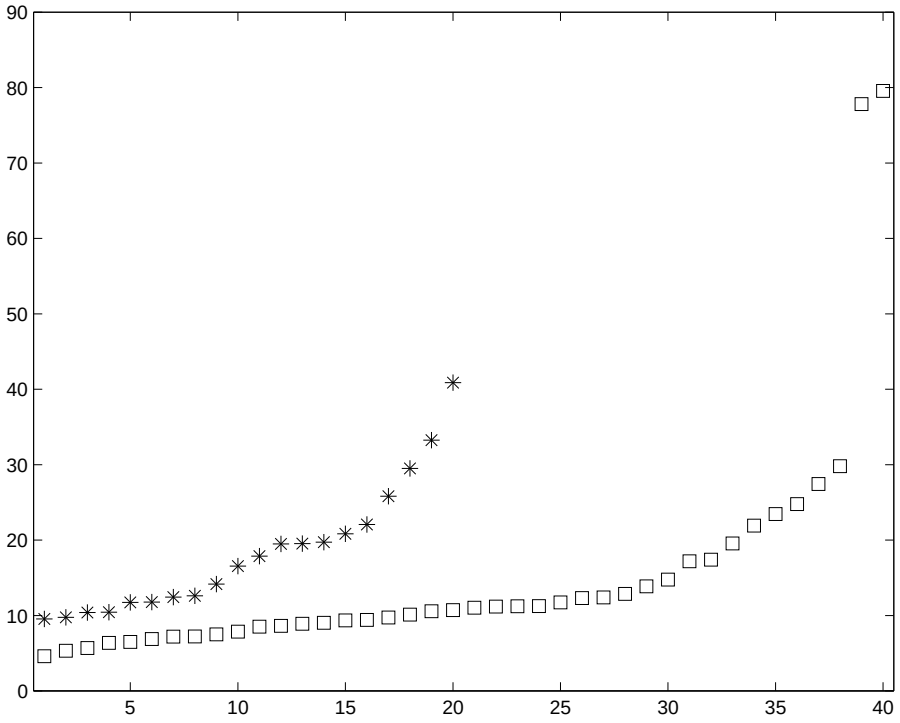

Figure 2: The angles between the real and matched components, using the quasi-orthogonalization approach. Squares: Laplacian distributed sources estimated with the present algorithm. Asterisks: For comparison, estimation of complete basis using FastICA, in symmetric mode.

3.6 Experiments on image data

Next, we tested our method on image feature extraction. We sampled 12×12 image windows from 13 natural images. We removed the mean (DC component) from the windows and whitened the data vectors thus obtained. From this 143 dimensional space we estimated 288 components, i.e. a basis that is twice overcomplete. We used the same parameter $\alpha = 0.34$ that we used in the simulations with the Laplacian distributed sources. A supergaussian density was assumed for the independent components by again taking $\log p_{y_i}(y_i) = -\log\cosh y_i$.

In Fig. 3, the basis vectors are shown. They are quite similar to what one obtains with ordinary ICA using a supergaussian prior for the independent components. In Fig. 4, we show the distances between the estimated basis vectors in the whitened space; these show that the basis vectors are really quasi-orthogonal. [2]

To further analyze the basis vectors, Gabor functions were fitted to each vector by a least-squares fit. Thus, every basis vector was described using a limited number of parameters, including spatial position, orientation and frequency. Before fitting the Gabor function, we upsampled the basis functions by a factor of three in both dimensions, and then applied a 3×3 averaging filter on them. This eliminated certain spurious minima. A problem with these Gabor fits was that often the optimization converged to a solution with a narrow envelope and a very low frequency (with a zero crossing at the center). In such cases, the frequency tells little about the actual function. To avoid this problem, we calculated numerically the mean frequency of the Fourier power spectrum from these parameters.

First, we plotted the joint distribution of orientation and frequency [28] in Fig. 5. We can see that the two parameters are quite independent from each other. Orientation has strong concentrations along the multiples of 45 degrees. Likewise, we plotted the spatial positions (of the centers of the basis vectors) in Fig. 6. We can see that the distribution of the centers is quite uniform inside the sampling window.

Next, we tested the limits of the method by estimating highly overcomplete bases: 4 and 8 times. The parameter α was simply scaled by dividing it with 2 and 4 to give the values 0.17 and 0.085. Some results are shown in Figs 7 and 8. For reasons of space, only half of the vectors are shown in the latter case. We can see that the 4 times overcomplete basis is quite well estimated. Even in the 8 times overcomplete basis, only a few basis vectors are a bit messy. The orientation-frequency plot as is Fig. 5 is shown in Fig. 9 for the 4 times overcomplete basis. The 8 times overcomplete basis yielded similar results (not shown). The minimum angles as in Figs 1 and 4 are shown for the 8 times overcomplete set in Fig. 10, to demonstrate that even here, the vectors were rather quasi-orthogonal; in particular, no two vectors were too similar. This was of course also the case in the 4 times overcomplete basis (not shown).

4 Approximate estimation by gaussianization

4.1 Gaussianization vs. orthogonalization

The second method that we propose for approximate estimation of overcomplete ICA bases is based on gaussianization. This idea comes from projection pursuit literature [8]. The point is to replace orthogonalization or quasi-orthogonalization by a nonlinear transform that makes the projections onto already estimated basis vectors gaussian.

We use a deflationary estimation of the independent components [7, 11], which means that we first estimate one independent component (typically by maximizing a measure of nongaussianity), then estimate a second component somehow discarding the direction of the first one, and so on, repeating the procedure n times.

The question is then, how to discard the already estimated components. Typically this is done by constraining the search for new independent components to the space that is orthogonal to the already found components; this is more or less equivalent to removing the estimated independent components from the data by linear regression, assuming that the data is prewhitened.

In the gaussianization procedure, we do not remove the components from the data, but we attempt to remove the nongaussianity associated with the component. First considering the non-overcomplete case, assume that we

[2] Note that this method can also be used to find a complete or undercomplete basis. The complete basis we obtained with image data had all the angles between the basis vectors above 85 degrees, compared with the orthogonality constraint (90 degrees) of standard ICA methods. The basis vectors seemed to be very similar to those we obtained when searching for overcomplete bases.

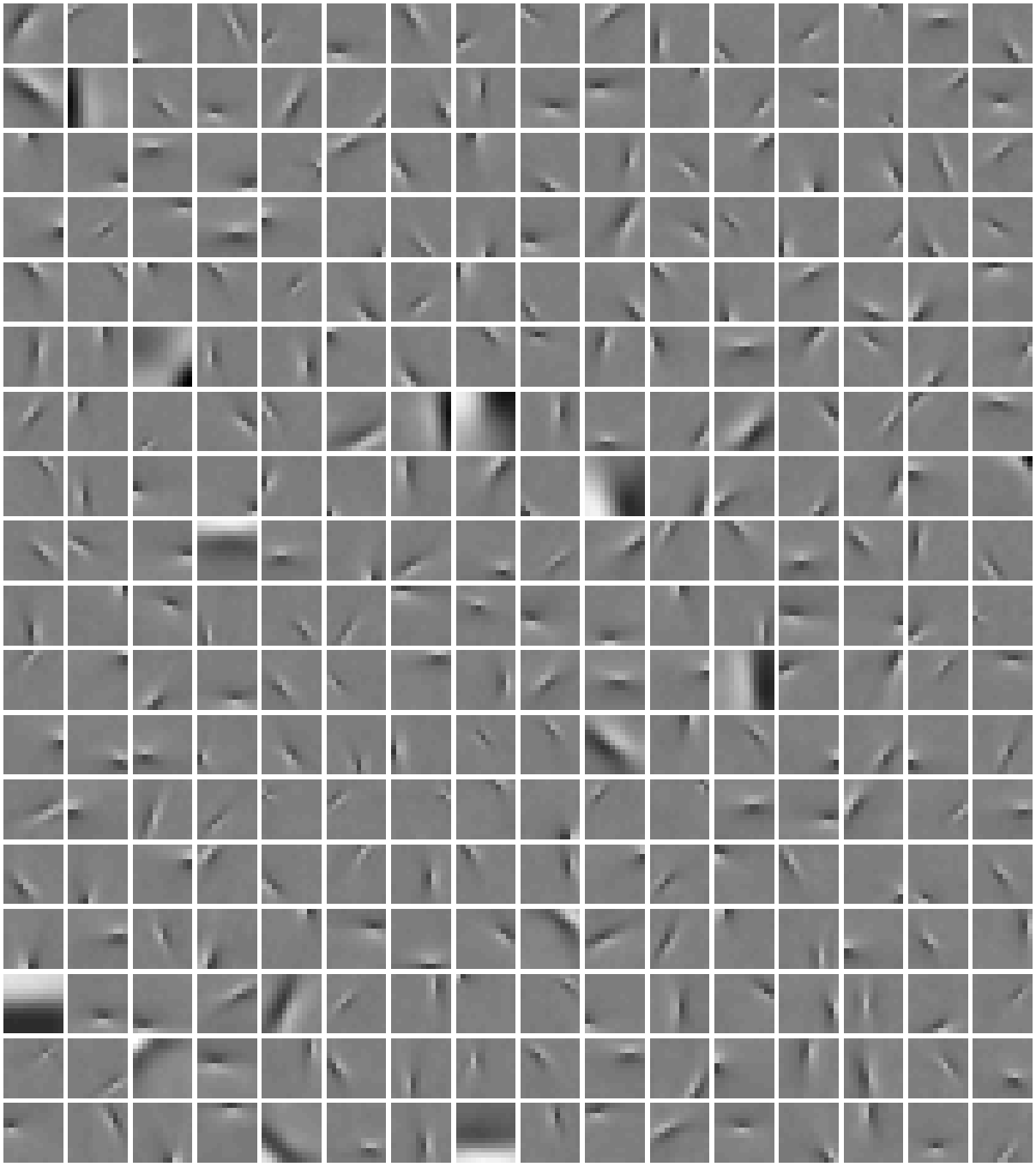

Figure 3: The basis vectors obtained with the quasi-orthogonalizing prior. The basis vectors are quite similar to those obtained by ordinary ICA, but the basis is 2 times overcomplete.

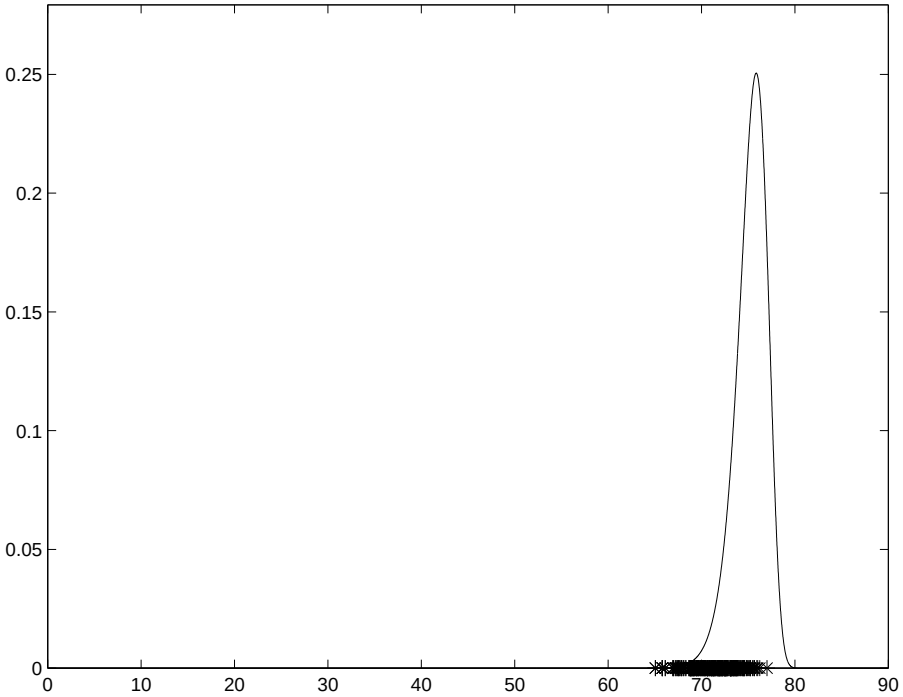

Figure 4: The minimum angles between the estimated components in the whitened space, using the quasi-orthogonalization approach on image data. Asterisks: The minimum angles between the estimated image basis vectors. Solid line: The distribution this quantity would have, if the vectors were drawn randomly.

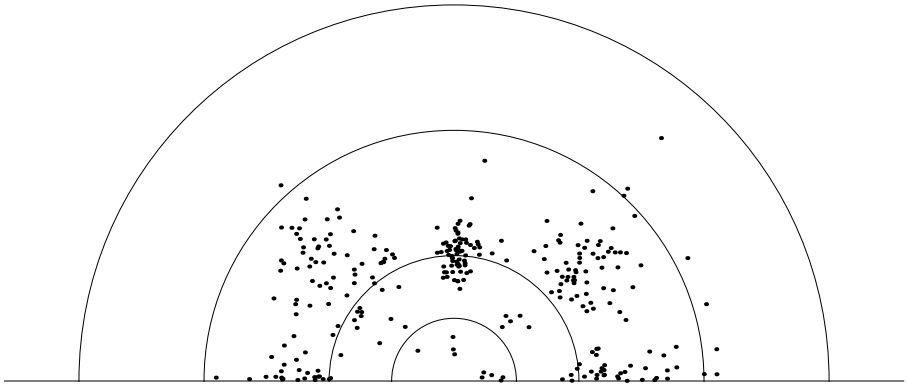

Figure 5: The distribution of the orientation and frequency in the 2 times overcomplete basis estimated by quasi-orthogonality. Each point in this polar plot is one basis vector. The distance from the origin is proportional to the frequency, and the angle gives the orientation. The innermost semicircle represents frequencies with a wavelength at 12 pixels (i.e. window size). The other semicircles represent wavelengths of 6, 3, and 2 pixels, respectively.

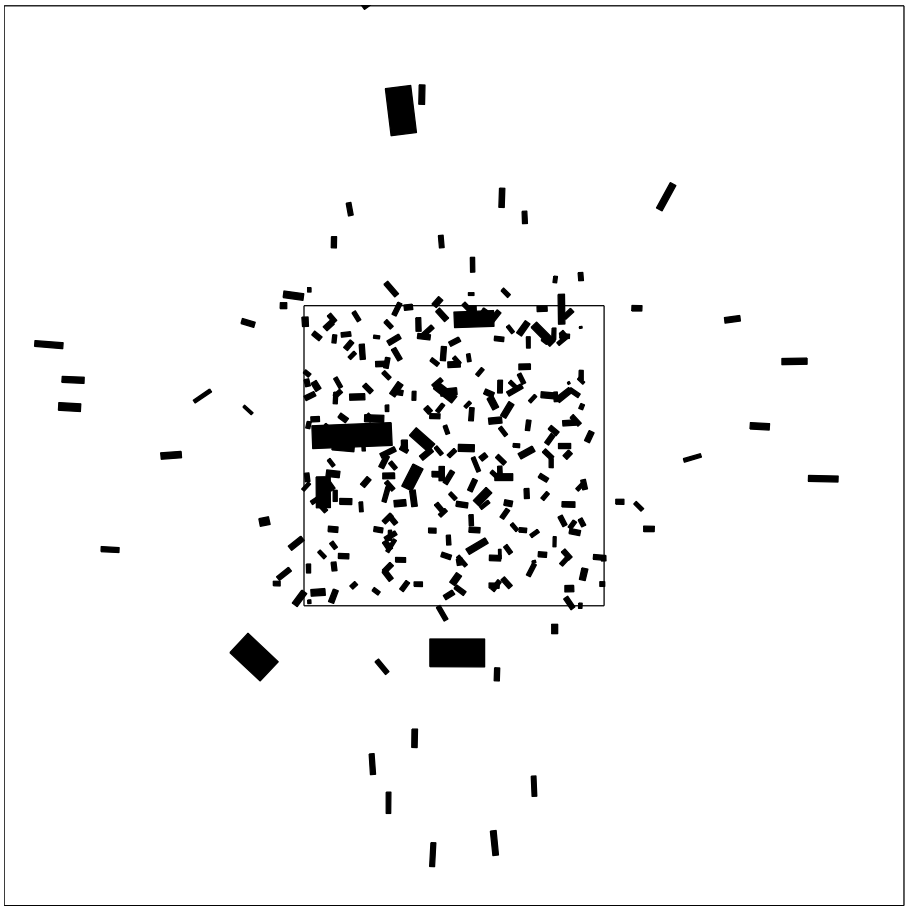

Figure 6: The spatial distribution of basis vectors inside the sampling window. The inner square is the sampling window. Each bar gives the position and orientation of a basis vector, and the size of the envelope of the basis vector is proportional to the bar width.

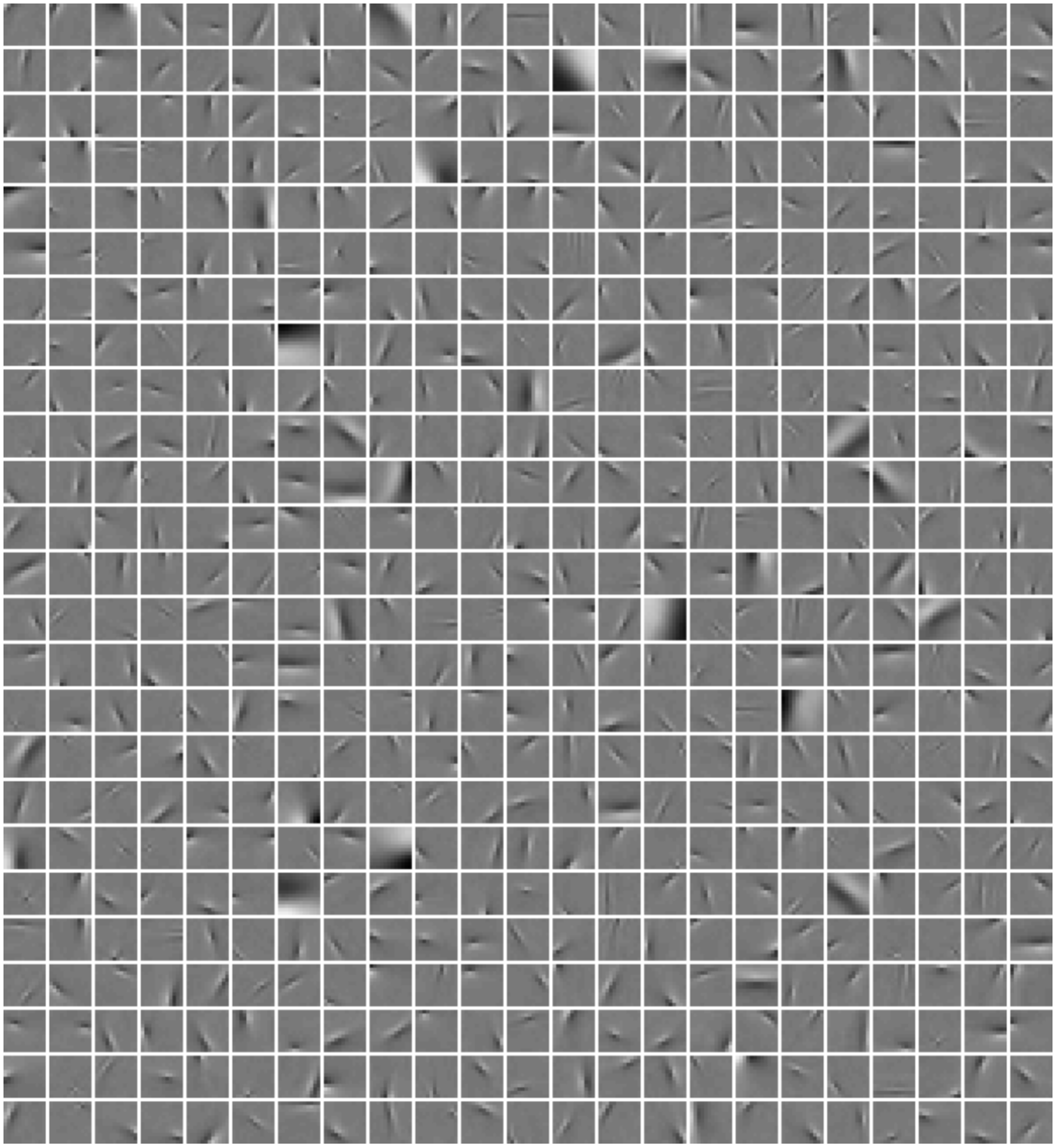

Figure 7: The basis vectors obtained with the quasi-orthogonalizing prior, this time 4 times overcomplete.

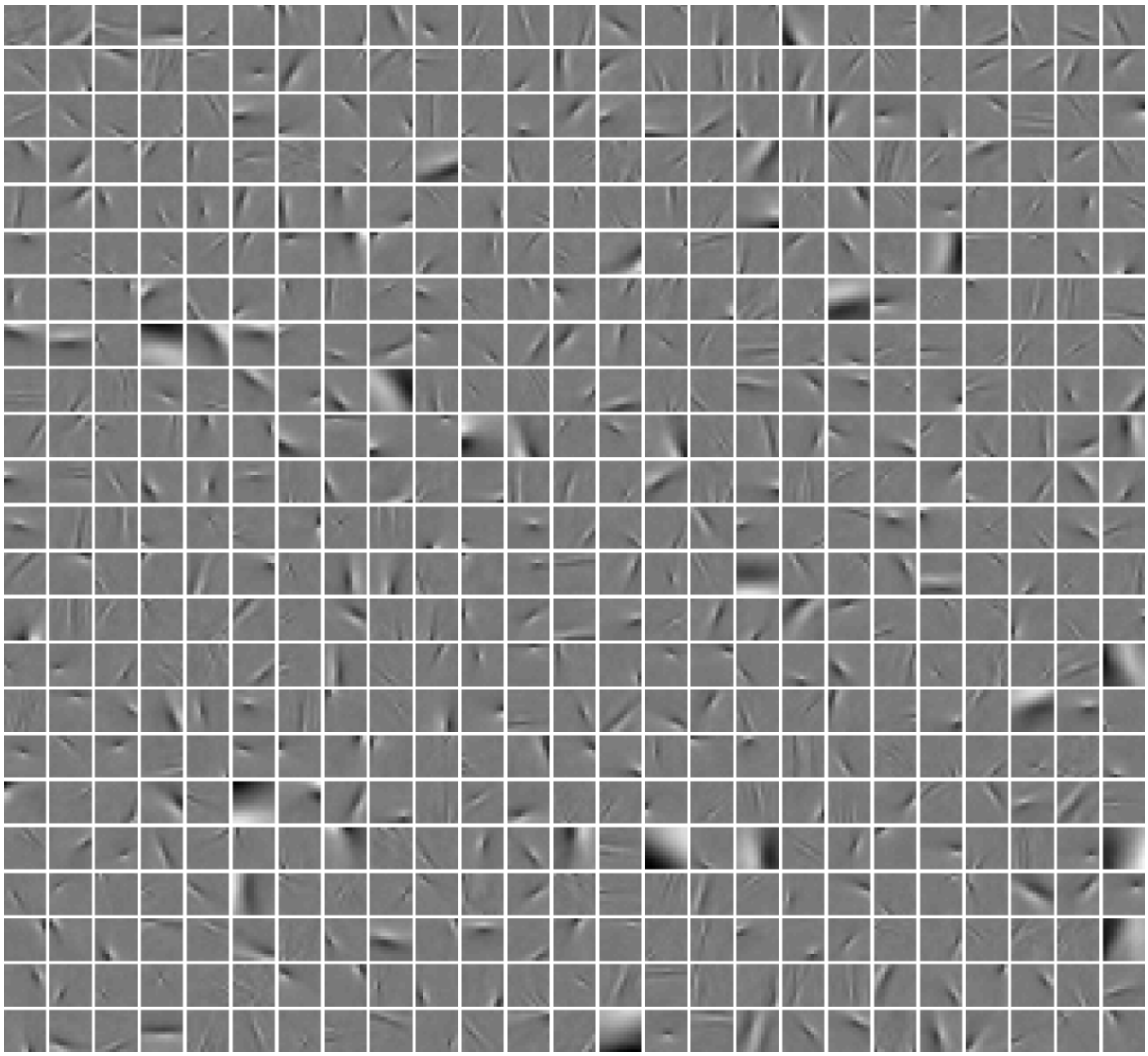

Figure 8: Basis vectors obtained with the quasi-orthogonalizing prior, this time 8 times overcomplete. For reasons of space, only a randomly selected half of the vectors are shown.

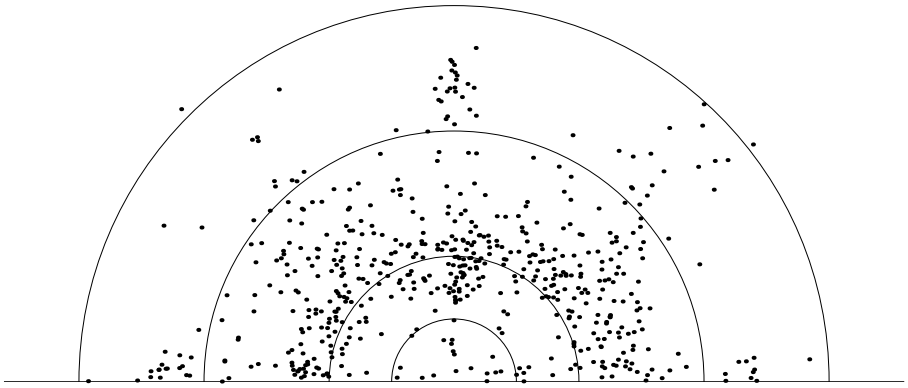

Figure 9: The distribution of the orientation and frequency in the 4 times overcomplete basis estimated by quasi-orthogonality. Each point in this polar plot is one basis vector. See caption of Fig. 5.

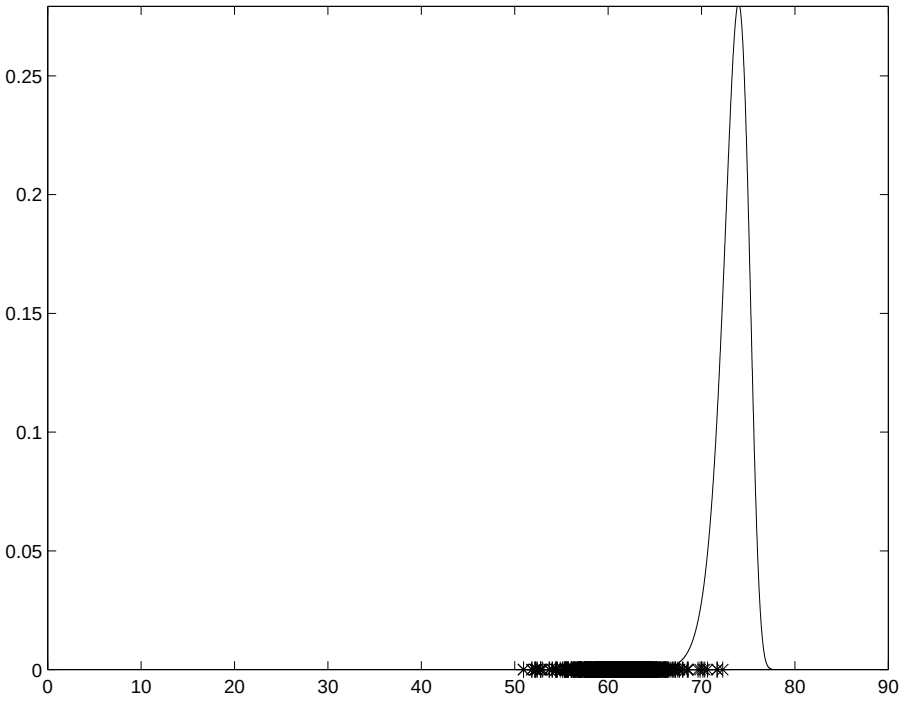

Figure 10: The minimum angles between the estimated components in the whitened space, using the quasi-orthogonalization approach on image data, this time with an 8 times overcomplete basis. Asterisks: The minimum angles between the estimated image basis vectors. Solid line: The distribution this quantity would have if the vectors were drawn randomly.

have estimated the i-th component as the linear combination $y_i = \mathbf{a}_i^T \mathbf{z}$ (we assume that the data is whitened). To gaussianize this direction, we compute the cumulative distribution function, say F of y_i. Then we compute for every observation $y_i(t) = \mathbf{a}_i^T \mathbf{z}(t)$ the transform $h(t) = \Phi^{-1}(F(y_i(t)))$, where Φ is the cumulative distribution function of the standardized gaussian distribution. This variable h has a gaussian distribution [8]. To reconstruct the observed $\mathbf{z}(t)$ after this gaussianization, we transform the data back as

$$\mathbf{z}(t) \leftarrow \mathbf{a}_i h(t) + (\mathbf{I} - \mathbf{a}_i \mathbf{a}_i^T)\mathbf{z}(t) \tag{19}$$

The above justification for gaussianization is not exactly valid for overcomplete bases. The dot product between the i-th basis vector and the whitened data vector can be written as:

$$\mathbf{a}_i^T \mathbf{z} = \mathbf{a}_i^T \mathbf{A}\mathbf{s} = s_i + \sum_{j \neq i} \mathbf{a}_i^T \mathbf{a}_j s_j \tag{20}$$

The first term is the i-th independent component. The second part is not zero in general, and includes contributions from other independent components. Therefore, it is impossible to remove only the nongaussianity related to one component, leaving others intact. However, the gaussianization transformation presented above is the only monotonous (and growing) transformation that produces a gaussian distribution for the dot-product. Other transformations would either produce a nongaussian y_i or add noise.

Note that even after m marginal gaussianizations (where m is the dimension of the data) the data is still not distributed according to a joint gaussian distribution: Forcing m marginal distributions to be gaussian does not, in general, make the joint distribution gaussian. In fact, the marginal gaussianizations may interact because the directions are not necessarily orthogonal, so that even the m components that were gaussianized need not have gaussian distributions after the whole process is finished. Compare this with the case of orthogonalization: In orthogonalizing deflation, it is completely impossible to estimate more than m components since one cannot have more than m orthogonal vectors in an m-dimensional space. This is exactly why we had to use quasi-orthogonalization instead of exact orthogonalization in the method of the preceding section.

With gaussianization, we do not need to modify the estimation method to use it for overcomplete bases. Note, however, that gaussianization is only applicable in deflationary mode, in which we estimate the components one by one; it cannot easily be used in the symmetric mode where all the components are estimated in parallel.

4.2 Simulations

We applied our method first on simulated data. The data we used with this approach was identical to that used with the quasi-orthogonalizing prior. The procedure for the estimation was as follows: first we whitened the observed data. Then we estimated one component by using FastICA [11] with the tanh nonlinearity, and then we gaussianized (using the cumulative distribution functions) the component in the direction that FastICA found. Then we estimated another component by FastICA, and so on.

We evaluated the angles between estimated basis vectors in the same manner as with the quasi-orthogonalizing prior. The minimum angles are shown in Fig. 11. All of these angles are above 52 degrees, which shows that we again obtained quite quasi-orthogonal basis vectors. The distances between the original basis vectors and their matched estimates are shown in Fig. 12. Almost all the components were properly estimated. For comparison, the Figure shows the results of estimating the components with ordinary (not overcomplete) ICA, which is able to estimate only a small part of the components, and even them with large errors.

4.3 Experiments with image data

Finally, we applied our algorithm for image feature extraction. The image data was similar to that used with the quasi-orthogonalizing prior. In Fig. 13 we have the obtained basis vectors of a 4 times overcomplete basis. Note that due to the deflationary (one-by-one) estimation approach, the estimate of a 2 times overcomplete basis is simply given by the upper half of this plot. These basis vectors are again similar to those obtained by basic ICA estimation, or the method of the previous section.

After the twice overcomplete basis, however, basis vectors which no longer resemble ICA basis vectors start to appear gradually (in the lower half of the plot). One should note that if gaussianization is continued, the data

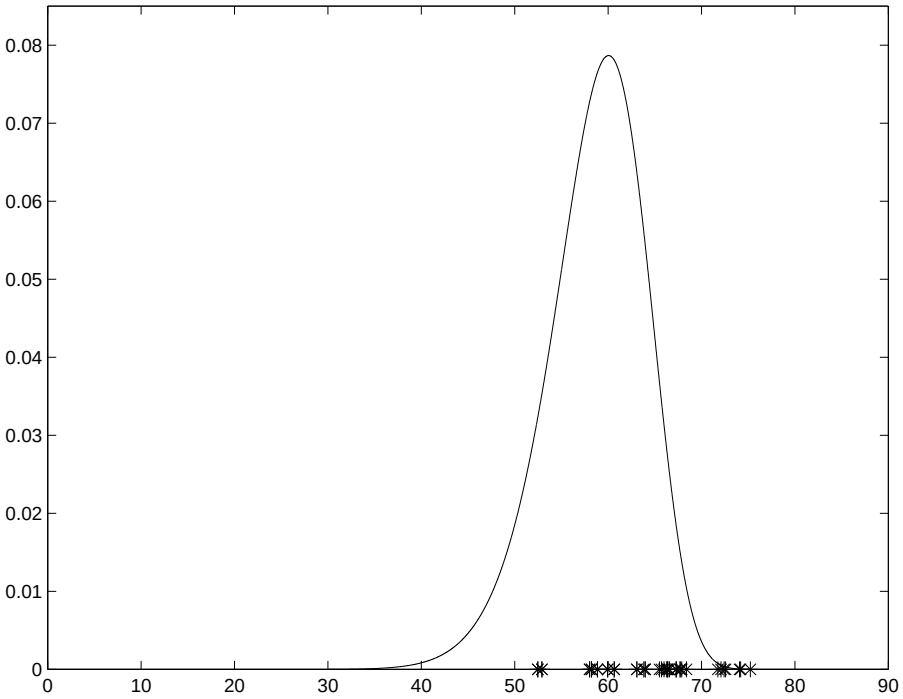

Figure 11: The quasi-orthogonality of the estimated basis vectors when 40 independent components are mixed into a 20 dimensional space, in the case of the gaussianization method. Asterisks: The minimum angles between the estimated basis vectors with Laplacian distributed sources. Solid line: Probability density that the minimum angle would have if the vectors were really generated randomly.

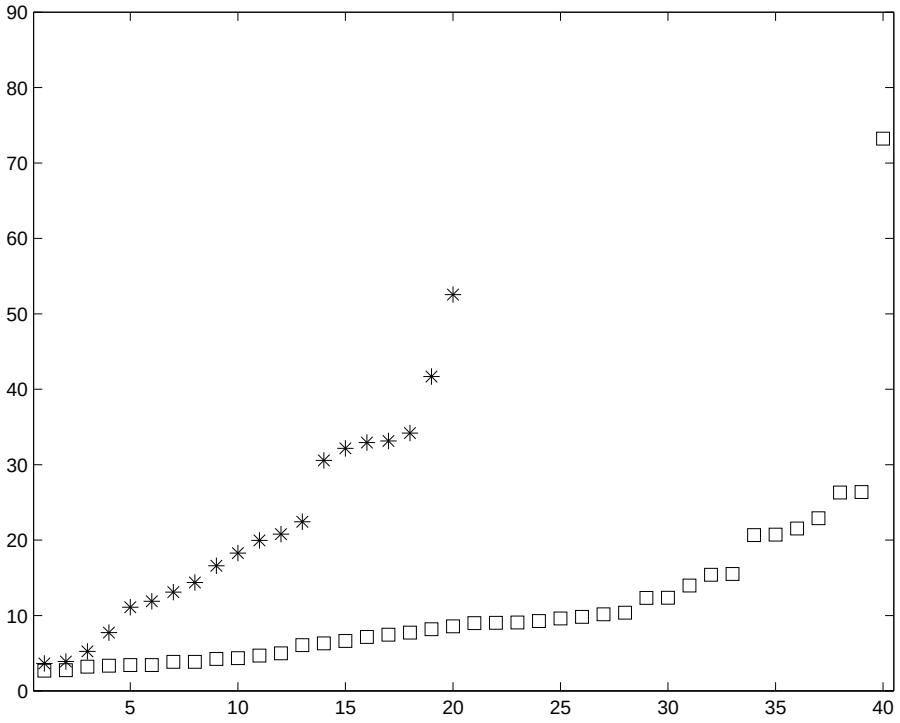

Figure 12: The angles between the real basis vectors and the matched estimates, for simulated data using the gaussianization procedure. Squares: Laplacian distributed sources estimated with the present algorithm. Asterisks: For comparison, estimation of complete basis using ordinary FastICA, in deflation mode.

distribution will converge weakly to a gaussian distribution [9]. Therefore, it is natural that after some point, the basis vectors found by the algorithm will no longer resemble normal ICA basis vectors, and are more influenced by nonlinear structures, possibly due to previous gaussianizations.

In Fig. 14, we have the distances between the estimated directions in the whitened space (for the whole 4 times overcomplete basis), showing that the basis vectors are quite different from each other.

5 Conclusion

We introduced two new methods for estimating overcomplete ICA bases from images. They were based on simply extending the estimation principles of basic ICA to the overcomplete case. The first method was based on using a Bayesian prior on the basis vectors, and the second on gaussianization. Simulations and experiments on image data show that the methods work surprisingly well, thus offering computationally efficient alternatives for more statistically principled methods.

Acknowledgements

We would like to thank Patrik Hoyer for sharing his expertise and code for Gabor fitting. A.H. was funded by the Academy of Finland, research fellow position. M.I. was funded by the Helsinki Graduate School of Computer Science and Engineering, and The Finnish Foundation of Technology.

References

[1] S.-I. Amari, A. Cichocki, and H.H. Yang. A new learning algorithm for blind source separation. In *Advances in Neural Information Processing Systems 8*, pages 757–763. MIT Press, 1996.

[2] H. Attias. Independent factor analysis. *Neural Computation*, 11(4):803–851, 1999.

[3] A.J. Bell and T.J. Sejnowski. An information-maximization approach to blind separation and blind deconvolution. *Neural Computation*, 7:1129–1159, 1995.

[4] A.J. Bell and T.J. Sejnowski. The 'independent components' of natural scenes are edge filters. *Vision Research*, 37:3327–3338, 1997.

[5] J.-F. Cardoso and B. Hvam Laheld. Equivariant adaptive source separation. *IEEE Trans. on Signal Processing*, 44(12):3017–3030, 1996.

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

[7] N. Delfosse and P. Loubaton. Adaptive blind separation of independent sources: a deflation approach. *Signal Processing*, 45:59–83, 1995.

[8] J.H. Friedman. Exploratory projection pursuit. *J. of the American Statistical Association*, 82(397):249–266, 1987.

[9] P.J. Huber. Projection pursuit. *The Annals of Statistics*, 13(2):435–475, 1985.

[10] A. Hyvärinen. Independent component analysis in the presence of gaussian noise by maximizing joint likelihood. *Neurocomputing*, 22:49–67, 1998.

[11] A. Hyvärinen. Fast and robust fixed-point algorithms for independent component analysis. *IEEE Trans. on Neural Networks*, 10(3):626–634, 1999.

[12] A. Hyvärinen. Sparse code shrinkage: Denoising of nongaussian data by maximum likelihood estimation. *Neural Computation*, 11(7):1739–1768, 1999.

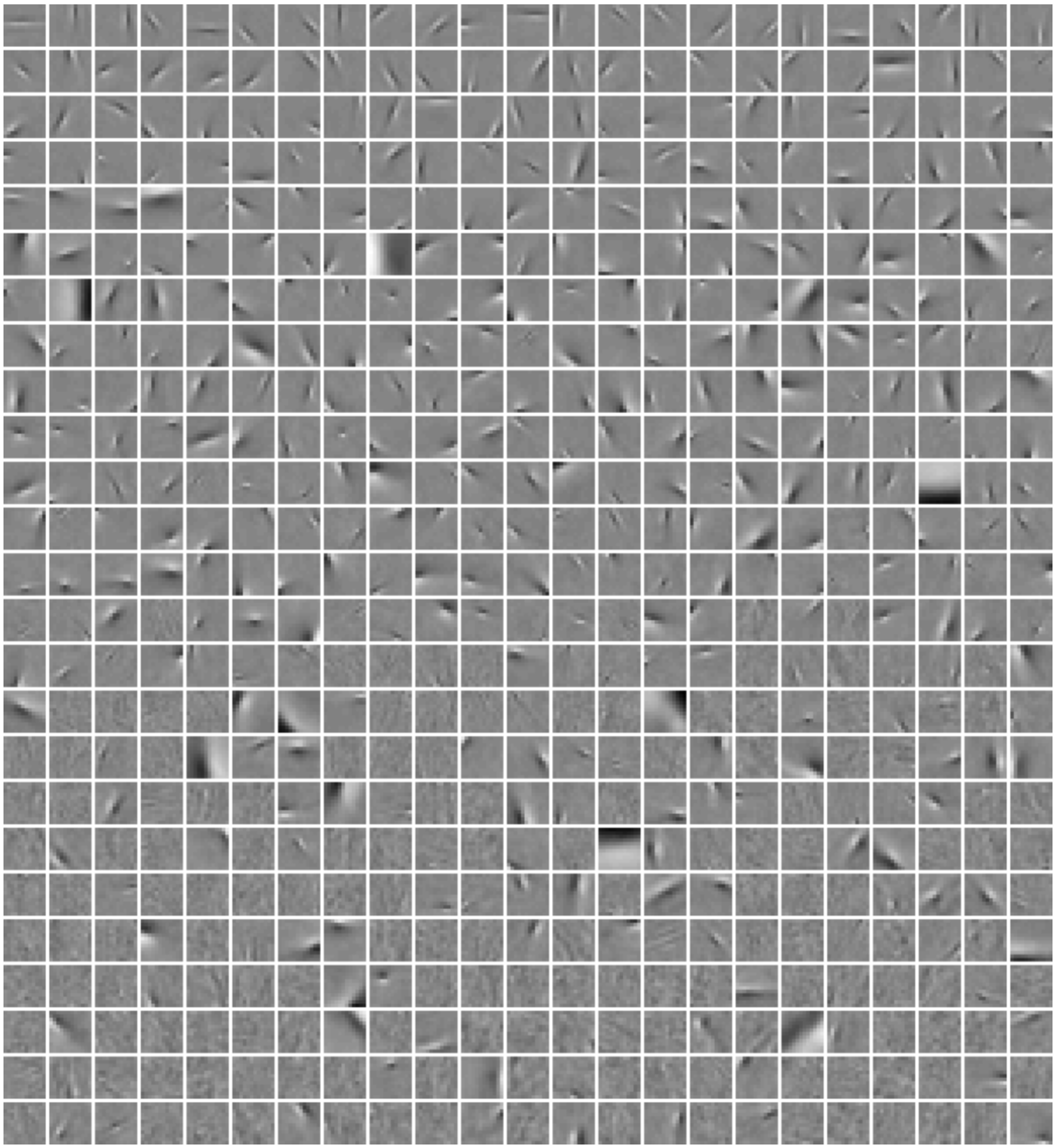

Figure 13: The image basis vectors obtained with the method using gaussianization. The order in which the vectors were obtained was left upper-hand corner to right lower-hand corner, scanning row by row. The upper half of the plot thus shows what one would obtain if one only estimated a 2 time overcomplete basis. The whole set of vectors gives a 4 times overcomplete basis.

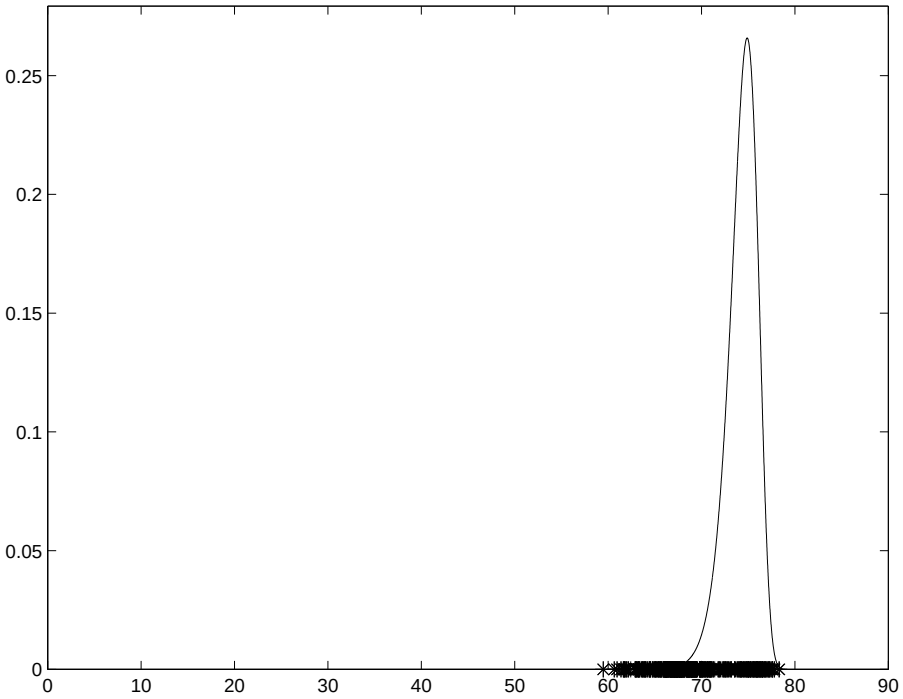

Figure 14: The angles between the estimated components in the whitened space, for image data and the gaussianization approach, considering the 2 times overcomplete basis. Asterisks: The minimum angles between the estimated basis vectors with Laplacian distributed sources. Solid line: Probability density that the minimum angle would have if the vectors were really generated randomly.

[13] A. Hyvärinen, R. Cristescu, and E. Oja. A fast algorithm for estimating overcomplete ICA bases for image windows. In *Proc. Int. Joint Conf. on Neural Networks*, pages 894–899, Washington, D.C., 1999.

[14] A. Hyvärinen and P. O. Hoyer. Emergence of phase and shift invariant features by decomposition of natural images into independent feature subspaces. *Neural Computation*, 12(7):1705–1720, 2000.

[15] A. Hyvärinen, P. O. Hoyer, and M. Inki. Topographic independent component analysis. *Neural Computation*, 13(7):1527–1558, 2001.

[16] A. Hyvärinen, J. Karhunen, and E. Oja. *Independent Component Analysis*. Wiley Interscience, 2001.

[17] A. Hyvärinen and E. Oja. Independent component analysis by general nonlinear Hebbian-like learning rules. *Signal Processing*, 64(3):301–313, 1998.

[18] A. Hyvärinen and E. Oja. Independent component analysis: Algorithms and applications. *Neural Networks*, 13(4-5):411–430, 2000.

[19] M. Inki and A. Hyvärinen. Two methods for estimating overcomplete independent component bases. In *Proc. Int. Workshop on Independent Component Analysis and Blind Signal Separation (ICA2001)*, San Diego, California, 2001.

[20] M. Inki and A. Hyvärinen. Two approaches to estimation of overcomplete independent component bases. In *Proc. Int. Joint Conference on Neural Networks (IJCNN 2002)*, Honolulu, Hawaii, 2002.

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

[22] P. Kanerva. *Sparse Distributed Memory*. MIT Press, 1998.

[23] J. Karhunen, E. Oja, L. Wang, R. Vigário, and J. Joutsensalo. A class of neural networks for independent component analysis. *IEEE Trans. on Neural Networks*, 8(3):486–504, 1997.

[24] S. Kaski. Dimensionality reduction by random mapping: fast similarity computation for clustering. In *Proc. Int. Joint Conf. on Neural Networks (IJCNN'98)*, pages 413–418, Anchorage, Alaska, 1998.

[25] T. Kohonen. *Self-Organizing Maps*. Springer, 1995.

[26] H. Lappalainen and A. Honkela. Bayesian nonlinear independent component analysis by multi-layer perceptrons. In M. Girolami, editor, *Advances in Independent Component Analysis*, pages 93–121. Springer-Verlag, 2000.

[27] T.-W. Lee, M.S. Lewicki, M. Girolami, and T.J. Sejnowski. Blind source separation of more sources than mixtures using overcomplete representations. *IEEE Signal Processing Letters*, 4(5), 1999.

[28] M. Lewicki and B. Olshausen. A probabilistic framework for the adaptation and comparison of image codes. *J. Opt. Soc. Am. A: Optics, Image Science, and Vision*, 16(7):1587–1601, 1998.

[29] M. Lewicki and T.J. Sejnowski. Learning overcomplete representations. *Neural Computation*, 12:337–365, 2000.

[30] E. Moulines, J.-F. Cardoso, and E. Gassiat. Maximum likelihood for blind separation and deconvolution of noisy signals using mixture models. In *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP'97)*, pages 3617–3620, Munich, Germany, 1997.

[31] E. Oja. The nonlinear PCA learning rule in independent component analysis. *Neurocomputing*, 17(1):25–46, 1997.

[32] B. A. Olshausen and D. J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381:607–609, 1996.

[33] B. A. Olshausen and D. J. Field. Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision Research*, 37:3311–3325, 1997.

[34] B. A. Olshausen and K. J. Millman. Learning sparse codes with a mixture-of-gaussians prior. In *Advances in Neural Information Processing Systems*, volume 12, pages 841–847. MIT Press, 2000.

[35] P. Pajunen. Blind separation of binary sources with less sensors than sources. In *Proc. Int. Conf. on Neural Networks*, Houston, Texas, 1997.

[36] A. Pece. The problem of sparse image coding. *Journal of Mathematical Imaging and Vision*, 2002. In this issue.

[37] E. P. Simoncelli, W. T. Freeman, E. H. Adelson, and D. J Heeger. Shiftable multiscale transforms. *IEEE Transactions on Information Theory*, 38:587–607, 1992.

[38] E. P. Simoncelli and O. Schwartz. Modeling surround suppression in V1 neurons with a statistically-derived normalization model. In *Advances in Neural Information Processing Systems 11*, pages 153–159. MIT Press, 1999.

[39] C. Zetzsche and G. Krieger. Nonlinear neurons and high-order statistics: New approaches to human vision and electronic image processing. In B. Rogowitz and T.V. Pappas, editors, *Human Vision and Electronic Imaging IV (Proc. SPIE vol. 3644)*, pages 2–33. SPIE, 1999.