A Two-Layer Model of Natural Stimuli Estimated with Score Matching

Urs Köster[1] **and Aapo Hyvärinen**[1, 2]
[1]Department of Computer Science and HIIT, University of Helsinki, Finland.
[2]Department of Mathematics and Statistics, University of Helsinki, Finland.

Abstract

We consider a hierarchical two-layer model of natural signals in which both layers are learned from the data. Estimation is accomplished by Score Matching, a recently proposed estimation principle for energy-based models. If the first layer outputs are squared and the second layer weights are constrained to be non-negative, the model learns responses similar to complex cells in primary visual cortex from natural images. The second layer pools a small number of features with similar orientation and frequency, but differing in spatial phase. For speech data, we obtain analogous results. The model unifies previous extensions to ICA such as subspace and topographic models and provides new evidence that localized, oriented, phase invariant features reflect the statistical properties of natural image patches.

1 Introduction

A variety of methods like Independent Component Analysis (ICA, see Comon, 1994) and Sparse Coding (Olshausen & Field, 1997) have been applied to model the statistical structure of natural signals such as images and sounds. In computational neuroscience, the goal of modelling these signals with unsupervised learning methods is to gain a better understanding of sensory processing, which is assumed to be linked to the statistics of ecologically valid stimuli (Barlow, 1961; Hyvärinen et al., 2009).

Linear ICA is limited in scope and cannot capture arbitrary dependencies, so more recent models use a nonlinear representation to better capture the structure of the data. In particular, there is a growing number of hierarchical models with two weight layers. These include direct extensions to ICA such as Independent Subspace Analysis (ISA) and topographic ICA (TICA, see Hyvärinen et al., 2001; Hyvärinen & Hoyer, 2000) which can be viewed as employing a manually selected, fixed second layer that pools over first layer features modelling dependencies which cannot be removed by a linear transform. The fixed second layer in these models has the advantage that the probability density function (pdf) can still be normalized in closed form, or approximations for the likelihood can be found. Thus a straightforward estimation of these models by maximizing likelihood is possible.

More recently, models where the second layer is also learned from the data have received attention. However, this comes the expense of a more complicated estimation, since these models can in general not be normalized in closed form, making maximum likelihood learning very difficult. Two recent models of this kind are the hierarchical Bayesian model by Karklin & Lewicki (2005, 2006) and the hierarchical Product of Experts (Osindero et al., 2006). The first is a generative model in which the components are not independent and identically distributed, but the variance is given by hidden variables. The second model is an energy-based model with an intractable partition function, similar to the one we consider here, and it is estimated using Contrastive Divergence (CD, Hinton, 2002).

We present a two-layer model of natural stimuli where both layers are estimated from the data, and analyze the resulting pooling patterns in the second layer. Following the classical energy model of complex cells (Adelson & Bergen, 1985; Spitzer & Hochstein, 1985), linear filter outputs from the first layer are squared and then pooled, where the pooling is learned from the data.

In our analysis we focus on two points in particular. We compare the results obtained by estimating both layers of the model simultaneously, with a simplified model where the second layer is estimated on top of a fixed ICA basis in the first layer, and report differences in tuning of the linear filters as well as the higher order units. Furthermore, we analyze the effect of L_1-normalization of the second layer on the resulting outputs, and show that this normalization plays a significant role in obtaining pooling patterns in line with previous complex cell models.

The model is estimated with Score Matching (Hyvärinen, 2005, 2007a), a consistent estimation method for energy-based models which cannot be normalized in closed form. Traditionally, these energy-based models would have to be estimated with Markov Chain Monte Carlo (MCMC) methods, which is computationally expensive and it is hard to evaluate convergence. While recent methods like Contrastive Divergence are computationally more efficient, it is still necessary to set up a Markov chain, the choice of which may greatly influence the convergence properties. Score Matching, in contrast, gives an objective function which can simply be optimized by gradient methods.

This paper is organized as follows. In Section 2, we discuss previous models of natural images, focussing on ICA and its extensions. In Section 3, the two-layer model is presented including details of our implementation and the Score Matching estimation. In addition, we review how the Score Matching objective function is derived. We test the model and estimation on synthetic data in Section 4. In Section 5, we apply the method to natural image data. We present models estimated with different constraints and analyze the tuning statistics of the model cells for their complex cell properties. We compare the effect of different normalization methods for the second layer of the model, and compare the results with a complete model to those with an overcomplete first layer. We focus on models with a non-negative second layer, but also consider models without the non-negativity constraint. In Section 6 we perform similar experiments with speech data, where we obtain a sparse pooling in the second layer that is very similar to the results for natural images. In Section 7 we discuss how our work compares to other recently developed two-layer models, highlighting the principled estimation with Score

Matching and the analysis of the complex cell-like properties of the outputs in our model. Finally we conclude with Section 8. Preliminary results have been published in (Köster & Hyvärinen, 2007).

2 Modelling of Natural Images

Ever since mammalian visual receptive fields were described by Hubel and Wiesel in the 1960's (see e.g. Hubel & Wiesel, 1959, 1962), efforts have been made to understand why the receptive fields have the observed properties. One successful approach is based on the idea that neural processing should be matched to statistics of ecologically valid stimuli, i.e. natural images. This lead to the development of statistical models like sparse coding (Olshausen & Field, 1996) and ICA (Jutten & Herault, 1991; Comon, 1994), which result in basis functions with a strong resemblance to the receptive fields of simple cells.

The approach we use in this paper is inspired by the classical ICA model, so we will briefly look at ICA and its application to natural images. For the ICA model we suppose that a vector of independent components, or sources $\mathbf{s}$ is mixed to generate the observed data vector $\mathbf{x}$. This can be written as

$$\mathbf{x} = \mathbf{A}\mathbf{s}. \tag{1}$$

In the simplest case, which is usually considered, the dimensionality of the source vector and the data vector is the same, so $\mathbf{A}$ is a square, invertible mixing matrix. Thus the components can be recovered from the data using the filter matrix $\mathbf{W} = \mathbf{A}^{-1}$ as

$$\mathbf{s} = \mathbf{W}\mathbf{x}. \tag{2}$$

There is a range of methods for the estimation of this model; here we focus on the likelihood-based approach. The distribution of the individual components is modelled by densities p_i, so by independence we have

$$p(\mathbf{s}) = \prod_i p(s_i) \tag{3}$$

This allows us to write the pdf of the data as

$$p(\mathbf{x}) = |\det \mathbf{W}| \prod_i p_i(\mathbf{w}_i^T \mathbf{x}) \tag{4}$$

where $\mathbf{w}_i^T$ are the rows of $\mathbf{W}$, and the determinant is a normalization factor due to the transformation of the density. Thus we obtain the log-likelihood of the parameters for a finite sample of data as

$$\log L(\mathbf{W}) = \sum_t \sum_i \log p_i(\mathbf{w}_i^T \mathbf{x}(t)) + T \log(|\det \mathbf{W}|) \tag{5}$$

where $\mathbf{x}(t)$ runs over T samples from the data. The likelihood can easily be maximized w.r.t. the filters $\mathbf{W}$ by gradient ascent. Estimating an ICA model for natural image patches results in filters that are localized, oriented and band-pass, resembling the spatial receptive fields of simple cells in the primary visual cortex (V1).

However, a large fraction of cells in V1 is not well described by a linear response. In particular, complex cells, which are insensitive to spatial phase, cannot be modeled with a linear transform. To account for these responses, the ICA model can be extended by adding a fixed second layer on top of squared linear filter responses: Methods such as Independent Subspace Analysis (ISA, see Hyvärinen & Hoyer, 2000) and topographic ICA (TICA, see Hyvärinen et al., 2001) employ a pooling of linear filter responses to model residual dependencies between linear filters. In ISA, the vector of components s is projected onto a number of subspaces. Squared norms of projections onto subspaces are then computed as

$$u_j = \sum_{i \in S_j} s_i^2 \tag{6}$$

where the index i runs over all the components that belong to the j^{th} subspace. The pdf of the model then takes the form

$$p(\mathbf{x}) = |\det \mathbf{W}| \exp\left(-\sum_j f\left(\sum_{i \in S_j} (\mathbf{w}_i^T \mathbf{x})^2\right)\right) \tag{7}$$

where the scalar nonlinearity $f(.)$ defines the overall shape of the pdf. The pooling can also be viewed as a second linear transformation, or weight layer, where a number of first-layer units converge into one higher order unit. Since independence is assumed only for the higher order units u_j, the linear features that are projected onto one subspace may have dependencies. Applied to natural images, this results in a pooling of features with similar frequency, orientation and location, but different spatial phase. Thus it can be argued that complex cells are tuned to capture dependencies, in particular correlations in the variance of linear filters (Schwartz & Simoncelli, 2001), which the above model makes explicit by computing squared norms.

3 The Model and its Estimation

3.1 The two-layer model

While the ISA model described above gives important insights into the interpretation of simple and complex cells as feature detectors tuned to the statistics of natural stimuli, it is somewhat limited as an explanation *why* the specific pooling is taking place, since only a single linear transformation is learned from the data and the additional connectivity is pre-specified. This rules out certain types of connectivity that might provide a better model of the data, in favor of architectures that have been hypothesized from theoretical principles. It would be preferable to estimate a full two-layer model, to allow us to evaluate whether the kind of connectivity used in earlier models is actually valid from the point of view of statistical optimality. A conceptually simple extension to the ISA model described in the previous section would be to retain the basic structure, but learning the second layer from the data rather than fixing it. Thus we define a pdf that can be viewed as describing a two-layer network

$$\log p(\mathbf{x}) = \sum_h f\left[\mathbf{v}_h^T g\left(\mathbf{W}\mathbf{x}\right)\right] - \log Z(\mathbf{W}, \mathbf{V}) \tag{8}$$

where the first term in the log-probability is given by a sum over the outputs of individual second layer units. Here $Z(\mathbf{W}, \mathbf{V})$ is the partition function of the model, i.e. a function of the model parameters which ensures that the pdf integrates to unity. The $\mathbf{v}_h^T$ are rows of the second layer weight matrix $\mathbf{V}$, while the first layer $\mathbf{W}$ has been retained from the ICA model. The two weight matrices need not be square, so in general $\mathbf{W}$ will be of size $n \times m$ and $\mathbf{V}$ of size $m \times o$. We have two scalar nonlinearities $g(.)$ and $f(.)$, the first of which computes nonlinear features from the data, whereas the second shapes the overall pdf. Such a model cannot be normalized in closed form, since the normalization constant Z is given by an intractable integral. Therefore we use Score Matching for the estimation, which provides a straightforward method for learning in energy-based models.

For the results presented in this work, we have defined g to be a squaring operation, unless otherwise specified. In addition, the second layer was constrained non-negative. This is a natural choice for a model of complex cells, where outputs are computed by pooling over squared or rectified simple cell responses (Pollen & Ronner, 1983; Spitzer & Hochstein, 1985). The second nonlinear function is chosen to be of the form

$$f(u) = -\sqrt{|u| + 1} \tag{9}$$

which ensures that the overall distribution of the model is supergaussian. Again, this nonlinearity was used in all the simulations, unless otherwise mentioned. Using these nonlinearities, the model distribution becomes:

$$\log p(\mathbf{x}) = -\sum_h \sqrt{\mathbf{v}_h^T \left(\mathbf{W}\mathbf{x}\right)^2 + 1} - \log Z(\mathbf{W}, \mathbf{V}) \tag{10}$$

We further constrained the vectors $\mathbf{v}_h^T$ to be normalized to unit L_2 or alternatively to unit L_1-norm, which corresponds to constraining the second layer units to have unit output energy and encourages sparse connectivity.

3.2 Score Matching

Score Matching (Hyvärinen, 2005, 2007a,b) is an estimation method that allows learning of statistical models which are only specified up to a multiplicative normalization constant (partition function). Consider samples from a random vector $\mathbf{x} \in \mathbb{R}^n$ that follows a pdf $p_{\mathbf{x}}(\boldsymbol{\xi})$ and to which we would like to fit a model. We define a parametrized model density $p(\boldsymbol{\xi}|\boldsymbol{\Theta})$ which includes the true pdf and where $\boldsymbol{\Theta}$ is a parameter vector that we would like to estimate. Suppose that the normalization constant Z of the pdf cannot be computed in closed form, and we use q to denote the unnormalized distribution. In the form of a log-probability we have the model:

$$\log p(\boldsymbol{\xi}|\boldsymbol{\Theta}) = \log q(\boldsymbol{\xi}|\boldsymbol{\Theta}) - \log Z(\boldsymbol{\Theta}) \tag{11}$$

The model score function, which we define as the gradient of the log-probability with respect to the data, is obviously identical for q and p, and given by:

$$\Psi(\boldsymbol{\xi};\boldsymbol{\Theta}) = \nabla_{\boldsymbol{\xi}} \log p(\boldsymbol{\xi};\boldsymbol{\Theta}) \tag{12}$$

Likewise the score function of the observed data is denoted by

$$\Psi_{\mathbf{x}}(.) = \nabla_{\boldsymbol{\xi}} \log p_{\mathbf{x}}(.) \tag{13}$$

Working with the score function thus has the advantage that it does not depend on the normalization constant Z. The model can now be estimated by minimizing the squared distance between the *model score function* $\Psi(\xi;\boldsymbol{\Theta})$ and the *data score function* $\Psi_{\mathbf{x}}(.)$. This objective function is defined by

$$J(\boldsymbol{\Theta}) = \frac{1}{2}\int_{\boldsymbol{\xi}\in\mathbb{R}^n} p_{\mathbf{x}}(\boldsymbol{\xi})\|\Psi(\boldsymbol{\xi};\boldsymbol{\Theta}) - \Psi_{\mathbf{x}}(\boldsymbol{\xi})\|^2 d\boldsymbol{\xi} \tag{14}$$

This may not appear to be very useful at first sight, because estimating the data score function is a nonparametric problem, and would require no less effort than estimating the normalization constant. However, a much simpler form of the objective function can be obtained. The full proof can be found in (Hyvärinen, 2005). We start by expanding the squared term to

$$\begin{aligned} J(\boldsymbol{\Theta}) &= \frac{1}{2}\int_{\boldsymbol{\xi}\in\mathbb{R}^n} p_{\mathbf{x}}(\boldsymbol{\xi})\|\Psi(\boldsymbol{\xi};\boldsymbol{\Theta})\|^2 d\boldsymbol{\xi} + \frac{1}{2}\int_{\boldsymbol{\xi}\in\mathbb{R}^n} p_{\mathbf{x}}(\boldsymbol{\xi})\|\Psi_{\mathbf{x}}(\boldsymbol{\xi})\|^2 d\boldsymbol{\xi} && (15)\\ &- \int_{\boldsymbol{\xi}\in\mathbb{R}^n} p_{\mathbf{x}}(\boldsymbol{\xi})\Psi(\boldsymbol{\xi};\boldsymbol{\Theta})^T\Psi_{\mathbf{x}}(\boldsymbol{\xi}) d\boldsymbol{\xi} && (16)\end{aligned}$$

Here we note that the first term does not depend on the data score function, so rewriting it with the squared norm expanded as a sum we get

$$\frac{1}{2}\int_{\boldsymbol{\xi}\in\mathbb{R}^n} p_{\mathbf{x}}(\boldsymbol{\xi})\|\Psi(\boldsymbol{\xi};\boldsymbol{\Theta})\|^2 d\boldsymbol{\xi} = \int_{\boldsymbol{\xi}\in\mathbb{R}^n} p_{\mathbf{x}} \sum_{i=1}^{n} \frac{1}{2}\psi_i^2(\boldsymbol{\xi};\boldsymbol{\Theta}) d\boldsymbol{\xi} \tag{17}$$

where the ψ_i are elements of the score function. The second term is constant wrt. $\boldsymbol{\Theta}$, so we simply set

$$\frac{1}{2}\int_{\boldsymbol{\xi}\in\mathbb{R}^n} p_{\mathbf{x}}(\boldsymbol{\xi})\|\Psi_{\mathbf{x}}(\boldsymbol{\xi})\|^2 d\boldsymbol{\xi} = C \tag{18}$$

Thus we focus on the third term, where we start by writing out the inner product

$$\int_{\boldsymbol{\xi}\in\mathbb{R}^n} p_{\mathbf{x}}(\boldsymbol{\xi})\Psi(\boldsymbol{\xi};\boldsymbol{\Theta})^T\Psi_{\mathbf{x}}(\boldsymbol{\xi}) d\boldsymbol{\xi} = \sum_i \int_{\boldsymbol{\xi}\in\mathbb{R}^n} p_{\mathbf{x}}(\boldsymbol{\xi})\psi_i(\boldsymbol{\xi};\boldsymbol{\Theta})\psi_{\mathbf{x},i}(\boldsymbol{\xi}) d\boldsymbol{\xi} \tag{19}$$

and consider a single element of the sum. We now use the definition of the score function $\psi_{\mathbf{x},i}(\boldsymbol{\xi}) = \frac{\partial \log p_x(\boldsymbol{\xi})}{\partial \xi_i}$, so making use of the chain rule, the term becomes

$$\sum_i \int_{\boldsymbol{\xi}\in\mathbb{R}^n} p_{\mathbf{x}}(\boldsymbol{\xi})\psi_i(\boldsymbol{\xi};\boldsymbol{\Theta}) \left[\frac{\partial}{\partial \xi_i} \log p_x(\boldsymbol{\xi})\right] d\boldsymbol{\xi} = \sum_i \int_{\boldsymbol{\xi}\in\mathbb{R}^n} \frac{p_{\mathbf{x}}(\boldsymbol{\xi})}{p_{\mathbf{x}}(\boldsymbol{\xi})} \frac{\partial p_x(\boldsymbol{\xi})}{\partial \xi_i} \psi_i(\boldsymbol{\xi};\boldsymbol{\Theta}) d\boldsymbol{\xi} \tag{20}$$

We then use multivariate partial integration (Hyvärinen, 2005) to obtain the i-th term as

$$-\int_{\boldsymbol{\xi}\in\mathbb{R}^n} \frac{\partial p_x(\boldsymbol{\xi})}{\partial \xi_i}\psi_i(\boldsymbol{\xi};\boldsymbol{\Theta})d\boldsymbol{\xi} = \int_{\boldsymbol{\xi}\in\mathbb{R}^n} p_x(\boldsymbol{\xi})\frac{\partial \psi_i(\boldsymbol{\xi};\boldsymbol{\Theta})}{\partial \xi_i}d\boldsymbol{\xi} + D \tag{21}$$

where the integration constant D is zero as $\lim_{\boldsymbol{\xi}\to\infty} p_x(\boldsymbol{\xi}) = 0$. Working with a finite sample of data, we can replace the exact expectations with sample averages. Collecting the terms, we then obtain the expression

$$\tilde{J}(\boldsymbol{\Theta}) = \frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{n}\left[\frac{\partial}{\partial \xi_i}\psi_i(x(t);\boldsymbol{\Theta}) + \frac{1}{2}\psi_i^2(\mathbf{x}(t);\boldsymbol{\Theta})\right] + C \tag{22}$$

which is easy to evaluate since it only contains terms depending on the model pdf. Score matching has been shown to provide a consistent estimator (Hyvärinen, 2005), so if the data follows the model, $\tilde{J}$ is asymptotically minimized for the true parameters.

3.3 Estimating the model

We can now apply the Score Matching framework to the model defined in Equation (10). The score function of the two-layer network is given by

$$\Psi(\mathbf{x}) = \nabla_{\mathbf{x}}\sum_h f\left[\mathbf{v}_h g\left(\mathbf{W}\mathbf{x}\right)\right] \tag{23}$$

so we can write the Score Matching objective, i.e. the squared distance between model and data score function as

$$\begin{aligned}
\tilde{J}(\mathbf{V},\mathbf{W}) = \quad & \sum_{t=1}^{T}\sum_{k=1}^{n}\sum_{h=1}^{o}\sum_{\ell=1}^{m}\left[(w_\ell^k)^2 v_h^\ell g_\ell''(\mathbf{w}_i^T\mathbf{x}(t)) f'(\sum_i v_h^i g_i(\mathbf{w}_i^T\mathbf{x}(t)))\right] \\
+ \quad & \sum_{t=1}^{T}\sum_{k=1}^{n}\sum_{h=1}^{o} f_h''(\mathbf{v}_h^T g(\mathbf{W}\mathbf{x}(t)))\left[\sum_{\ell=1}^{m} w_\ell^k v_h^\ell g_\ell'(\mathbf{w}_i^T\mathbf{x}(t))\right]^2 \\
+ \quad & \sum_{t=1}^{T}\sum_{k=1}^{n}\frac{1}{2}\left[\sum_{h=1}^{o}\sum_{\ell=1}^{m} w_\ell^k v_h^\ell g_\ell'(\mathbf{w}_i^T\mathbf{x}(t)) f_h'(\mathbf{v}_h^T g(\mathbf{W}\mathbf{x}(t)))\right]^2
\end{aligned} \tag{24}$$

Optimizing this objective is straightforward by gradient descent, which requires the gradients of the above expression with respect to the elements of the weight matrices $\mathbf{W}$ and $\mathbf{V}$. These gradients are given in the Appendix. The non-negativity and norm constraint were implemented by projecting onto the constraint set after each gradient step.

4 Experiments on Simulated Data

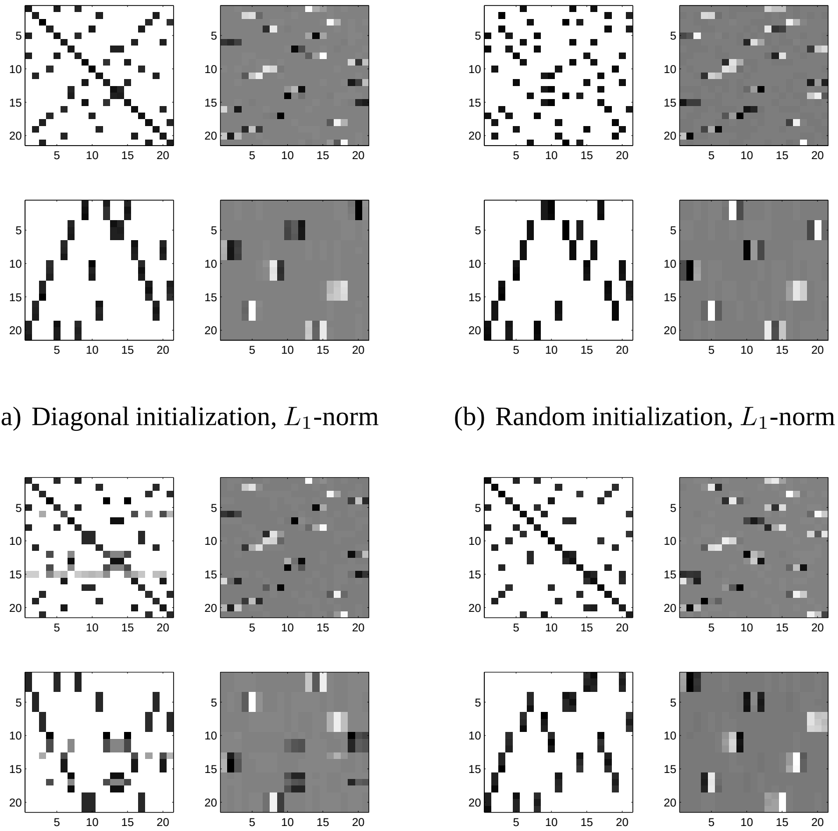
To verify the identifiability of the model we estimated it for simulated data with a known higher-order structure. We generated data following the ISA model, which is a special case of the proposed two-layer model and easy to sample from. The data generated in this way contains higher order dependencies in the form of common variances for groups of source variables, which cannot be captured by ICA. Samples from the ISA model were generated as follows: To obtain T observations of an n-dimensional vector which contains k subspaces, we first create a matrix $\mathbf{M}$ of $n \times T$ observations from an i.i.d Gaussian with unit variance, and a matrix $\mathbf{B}$ of $k \times T$ variance parameters from a uniform distribution. We introduce dependencies within groups of the Gaussians by multiplying them with a common variance from the uniform distribution: $\mathbf{U}(i,t) = \mathbf{M}(i,t)\mathbf{B}(j,t), \forall i \in S_j$. The supergaussian variables produced in this way are then multiplied with a mixing matrix $\mathbf{A}$ that is also generated randomly, so the data matrix is $\mathbf{X} = \mathbf{AU}$. Before the estimation the data is whitened. For the experiments shown below we set $T = 5000$, $n = 21$ and $k = 7$, so each subspace has three elements.

The experiments with artificial data mainly served the purpose to confirm the consistency of the estimation, but also to try out various initialization and normalization procedures for the experiments on natural stimuli. We compare L_1- and L_2-normalization of the second layer matrix $\mathbf{V}$, and we compare randomly initializing $\mathbf{V}$ and initializing it with an identity matrix, which allows pre-learning of $\mathbf{W}$ as an ICA model.

For the visualization of the results, note that in ICA, one can simply multiply the mixing matrix $\mathbf{A}$ with the estimated filter matrix $\mathbf{W}$ to obtain a permuted diagonal matrix if the components are identified correctly. Thus visual inspection of $\mathbf{Z} = \mathbf{W} \times \mathbf{A}$ can be used to to determine convergence. The ISA model is identifiable only up to subspaces due to rotational symmetry, where the second layer determines the subspace ownership of each element of $\mathbf{Z}$. By multiplying the second layer matrix $\mathbf{V}$ with $\mathbf{Z}$, a block-diagonal matrix with permuted rows should be obtained if the algorithm converges correctly.

Results are presented in Figure 1. In each of the four experiments *(a-d)* we performed, the top row shows the second layer $\mathbf{V}$ on the left and the product $\mathbf{V} \times \mathbf{Z}$, on the right. In the bottom row, on the left we show $\tilde{\mathbf{V}}$ where the rows of $\mathbf{V}$ have been permuted in such a way that identical rows are next to another. This is purely for visualization purposes and does not affect the objective function. Again, on the right we plot the product $\tilde{\mathbf{V}} \times \mathbf{Z}$. If this results in a permuted block-diagonal matrix, the second layer has correctly identified the dependency structure in the first layer.

Firstly, the comparison between *(a)* and *(b)* shows that convergence is possible both from $\mathbf{V}$ initialized with the identity matrix and from a random $\mathbf{V}$. However the number of iterations is about an order of magnitude greater starting from random. Secondly, between *(a)* and *(c)* we compare the effect of L_1 and L_2-normalization. In this case, the L_2-normalized model has converged to a local minimum and has not identified all the components correctly. In general however, there was no major difference between the L_1 and L_2 normalization for this data. Finally, in *(a)* and *(d)* we analyze the effect of estimating the layers sequentially, which means that $\mathbf{W}$ is only learned while $\mathbf{V}$ is fixed to identity, after which $\mathbf{W}$ is held fixed and learning is continued with $\mathbf{V}$ only. In this simple example, the model converges to the correct solution, but as we will see later it is preferable to estimate both layers simultaneously.



(a) Diagonal initialization, L_1-norm

(b) Random initialization, L_1-norm

(c) Diagonal initialization, L_2-norm

(d) Pre-learning of $\mathbf{W}$, L_1-norm

Figure 1: Simulations with generated data following the ISA model. For each of the four plots we show the second layer matrix $\mathbf{V}$ on the top left and the product $\mathbf{V} \times \mathbf{Z}$ on the top right. The bottom row contains the same matrices as the top row, but with the vectors permuted for visualization purposes.
(a) Both layers estimated simultaneously with $\mathbf{W}$ initialized with Gaussian white noise and $\mathbf{V}$ with an identity matrix. The rows of $\mathbf{V}$ are constrained to unit L_1-norm.
(b) Like (a), but both weight layers initialized with white noise. Convergence takes nearly an order of magnitude longer, but the the global minimum is found nevertheless.
(c) Like (a), but with rows of $\mathbf{V}$ constrained to unit L_2-norm. Note that the second layer converged to a local minimum.
(d) The estimation can be simplified by estimating only $\mathbf{W}$ first, with $\mathbf{V}$ held constant. In the second step both layers are learned. The quality of the optimum does not change, but speed of convergence is increased.

5 Experiments on Natural Images

5.1 Methods

All experiments were performed on images taken from P. O. Hoyer's ImageICA package[1], using $20,000$ image patches of size 16×16. The whole images were preprocessed by approximate whitening assuming a $\frac{1}{f^2}$ power spectrum and contrast gain control with a Gaussian neighborhood of 16 pixels diameter. Details of this preprocessing can be found in (Hyvärinen & Köster, 2007). The preprocessing can be given a physiological justification in terms of the processing in the retina and lateral geniculate nucleus, or it can be viewed more pragmatically as simplifying the statistical structure of the images slightly. We then randomly sampled patches from the images and removed the DC component from the patches. We also discarded any image patches with low variance, since they contribute little to the gradient and slow down learning. Finally we whitened the patches and simultaneously reduced the dimensionality from 256 to 120 using principal component analysis. The dimensionality reduction corresponds to low-pass filtering and eliminates aliasing artifacts due to the rectangular sampling grid from the image patches. Both weight matrices $\mathbf{W}$ and $\mathbf{V}$ were chosen to be square, of size 120×120, unless otherwise noted.

The matrix $\mathbf{W}$ was initialized with Gaussian white noise, $\mathbf{V}$ was initialized as an identity matrix for the experiments in Sec. 5.2 and with noise from a uniform distribution for the experiments in Sec. 5.3. The models were optimized using gradient descent with a constant stepsize. To increase the speed of convergence of the experiments in Sec. 5.2, we initialized by estimating the first layer only, keeping the second layer fixed. After the convergence of the first layer to an ICA basis, we performed two different experiments: In the first type of experiment, both layers were estimated simultaneously. In the second type $\mathbf{W}$ was held fixed after initial convergence to an ICA basis, and only $\mathbf{V}$ was estimated. In all experiments, the outputs units (rows of $\mathbf{V}$) were normalized to unit L_1 or L_2 norm after every step. Convergence was determined by visual inspection and took about 300 hours on a Pentium IV workstation.

To analyze the tuning properties of the filters in the first layer, we fit Gabor functions to the basis functions obtained by inverting the filter matrix $\mathbf{W}$. For each of the first layer responses, we used a least squares fit (adapted from Hyvärinen et al., 2001) to determine location, orientation, size, phase and frequency of the optimal Gabor. To compute tuning curves of the second layer outputs, we followed the model by taking squares of the first layer filter responses and summing them, weighted by rows of $\mathbf{V}$. The second layer nonlinearity is required in the estimation to define a supergaussian pdf, but it is not considered part of our complex cell model, so we analyzed the second layer outputs without any further nonlinearity. To obtain tuning curves, we designed the test stimulus for each higher order unit as a Gabor function constructed from a weighted average of the constituent first layer Gabor parameters. One of the parameters (location, orientation, phase, frequency) was then varied while the others were held at the optimum value.

[1] www.cs.helsinki.fi/patrik.hoyer

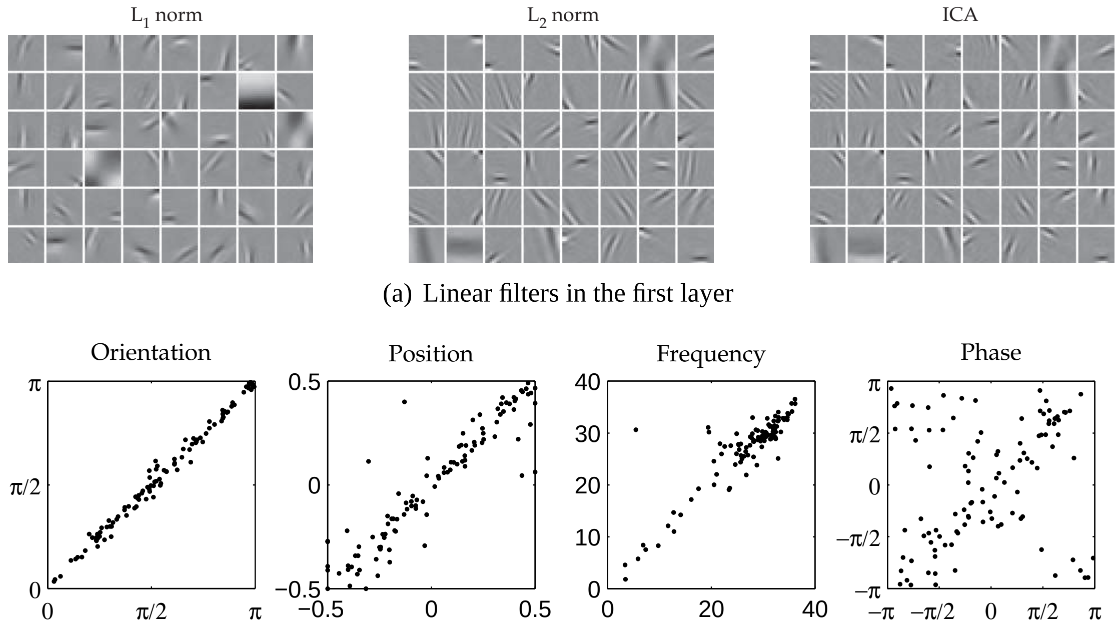
5.2 Results

In Figure 2 we analyze the first layer features and how they differ from those of an ICA model. Figure 2 *(a)* shows a random selection of 48 of the 120 basis functions for models estimated with L_1 and L_2-normalization, as well as the ICA basis functions obtained with $\mathbf{V}$ fixed to identity, for comparison. It can be seen that all the filters are Gabor-like and tuned in orientation, position, frequency and phase, resembling the responses of simple cells. The basis functions from the L_1 model appear slightly more localized, but less frequency and orientation selective than the filters from the L_2 model, with ICA falling between the two extremes. Comparing the Score Matching objective function for the three models, we observe that the L_2 model has the best fit to the data with $\tilde{J} = -79.1$, followed by the L_1 model with $\tilde{J} = -70.5$, while the ICA model only achieves a $\tilde{J} = -58.1$ which improves very little to $\tilde{J} = -59.6$ if the second layer is learned on top of the ICA basis without simultaneously allowing the basis functions in $\mathbf{W}$ to adapt to the new pooling patterns.

In *(b)* we investigate how the ICA basis functions in $\mathbf{W}$ change when the first layer adapts to the pooling patterns in the second layer. For this comparison, the L_2-norm model was estimated from the same random seed as the ICA model, where $\mathbf{V}$ was fixed to identity. As it can already be seen in *(a)*, the features change only little, but from a scatter plot showing the changes in the Gabor parameters for each linear filter, some systematic changes become visible. The orientation tuning is least affected by the estimation of both layers, so the parameter does not change significantly. Positions and frequencies change slightly for most of the features, but strong changes in the tuning are rare. The phase tuning is very different however, with many linear filters completely changing the phase tuning to better adapt to the pooling in the second layer. This gives some intuition why the improvement in model fit is so small if $\mathbf{V}$ is estimated on top of a fixed ICA basis, without allowing the features in $\mathbf{W}$ to adapt.

In Figure 3 we show the second layer of the model and the emerging pooling patterns in more detail. In *(a)* and *(b)* we show a subset of the pooling patterns in a representation adapted from (Hyvärinen et al., 2005), which also allows easy comparison with (Karklin & Lewicki, 2005). For each higher order unit, the linear filters that contribute to the output are represented by ellipses. The location and the orientation of each ellipse correspond to the location and orientation of the underlying first-layer basis function. Frequency is represented by the size of the ellipse, where larger corresponds to lower frequencies. The shading of the ellipse represents the connection strength, light gray being close to zero and black corresponding to maximal contribution. For the L_1 model, most of the outputs pool over a small number of linear filters, which share similar orientations and positions, while the pooling is more heterogeneous for the L_2 model. While the sparseness of the pooling is more pronounced with L_1-regularization, the average number of significantly active linear filters is still well below 10% for the L_2 norm model.

In *(c)* and *(d)* this is further analyzed for the two models: the plot on the left shows the most active features for a random selection of second order units. The units can be seen to share similar frequency, orientation and location, but differ in spatial phase. On the right hand side, the second layer matrix $\mathbf{V}$ is shown directly. Again, the connectivity can be seen to be sparse, with only a few first layer features contributing to each row



(a) Linear filters in the first layer

(b) Change in tuning of the first layer

Figure 2: *(a)* A subset of 48 randomly selected linear basis functions in the first weight layer. On the left and in the middle, features for models with an L_1-normalized and L_2-normalized second layer are shown. On the right we show a model with the second layer fixed to identity which corresponds to an ICA model. The L_2-norm model was initialized with this ICA basis, so the filters are similar. The L_1-norm model shows somewhat more location selectivity, whereas the L_2-norm model has more precise frequency and orientation tuning.
(b) Change in tuning properties of the linear filters in $\mathbf{W}$ as the first layer adapts to the pooling patterns in the second layer. The scatter plots show how the tuning of the individual Gabors changes as we go from the ICA model to the L_2-normalized model. The horizontal axis shows the value of the parameter in the ICA model, the vertical axis the value after learning both layers. While orientation tuning changes very little, and position as well as frequency also remain relatively stable, the spatial phase changes more dramatically.

of $\mathbf{V}$. Here the linear filter inputs were sorted by frequency and output rows by sparseness. The pooling is quite homogeneous for the L_1 case, but for L_2-normalization, large groups of high-frequency filters are pooled into one output. There are several near-identical copies of these outputs, indicating a comparably large contribution to the model pdf.

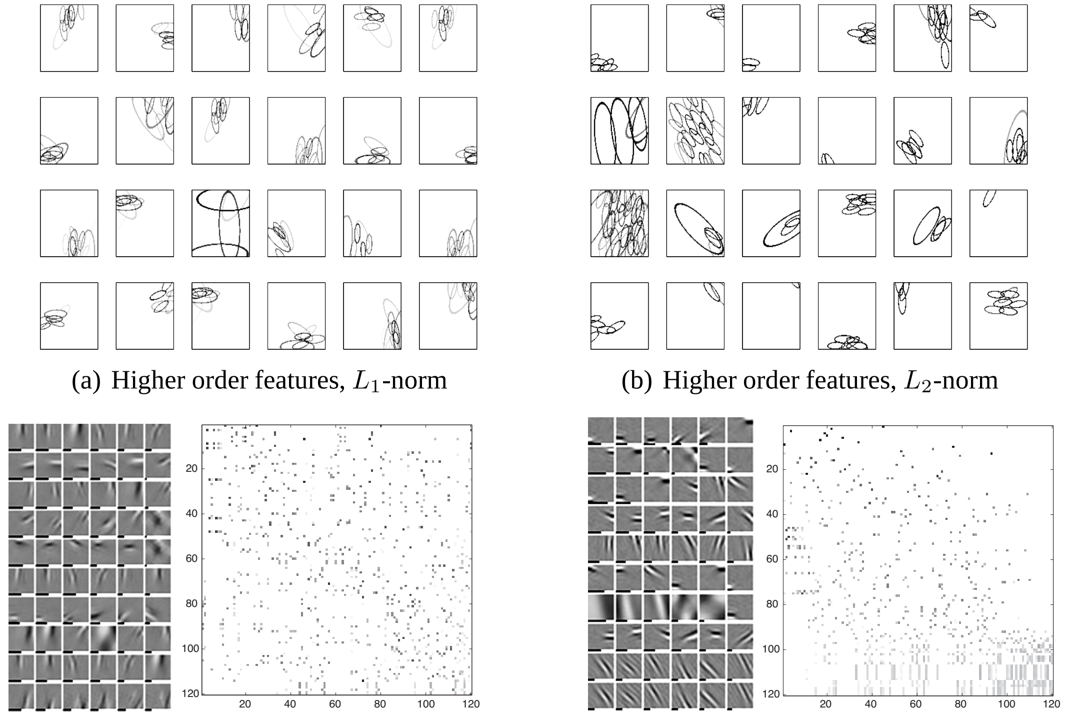
In Figure 4 we analyze the complex cell properties of the higher order outputs for the different models. We further investigate the effect of simultaneous vs. sequential estimation of the weight layers and the difference between L_1 and L_2-normalization. The tuning curves are computed by taking the optimal Gabor stimulus for each higher order unit and changing one of the parameters (phase, position, orientation and frequency) at a time. In *(a)*, only the first layer was learned and the second fixed to the identity matrix, so the model corresponds to ICA. This results in simple cell behavior with strong phase selectivity. Sequential estimation of the two weight layers with L_1 normalization is shown in *(b)*, so the first layer filters are not adapted to the pooling patterns. There is a decrease in selectivity to spatial phase, indicating complex cell properties. In *(c)* both layers have been estimated simultaneously with L_1 normalization. The adaptation of the phase of the linear filters to the pooling patterns leads to a striking decrease in phase selectivity, i.e. the second layer outputs become more complex cell-like. In particular the upper 10% quantile of the outputs becomes essentially completely phase-invariant, whereas in the sequential estimation, there is still a 40% modulation in this quantile. At the same time the selectivity for position, orientation and frequency are not affected considerably, with only a slight broadening. In *(d)* we show the responses for a model with simultaneous estimation of both layers and L_2 normalization. Due to the heterogeneous pooling patterns, much of the selectivity, in particular for position, is lost. At the same time the large number of simple cell-like outputs with only a single strongly active linear filter leads to a loss of phase invariance. The regularization with an L_1 norm seems to be an important requirement to obtain complex-cell like responses.

5.3 Estimation of an overcomplete model

To generalize our experiments to an overcomplete model, we propose a model in which the number of linear filters is higher than the data dimensionality, but the dimensionality is reduced again for the higher order units. This is motivated by the observation that with no normalization on the second layer, many of the outputs go to zero (experiments not shown). Since we do not need to take the normalizability of the model into account, it is straightforward to make the set of filters in $\mathbf{W}$ overcomplete. We consider such a model which is overcomplete by a factor of four, with 240 filters in $\mathbf{W}$ estimated on data with the dimensionality reduced to 60, but otherwise identical to the image data used in the previous section.

In order to make the model overcomplete, we need to drop two simplifications used to far. Firstly, we cannot use an ICA initialization since this would require as many output units as linear filters. Instead of the initialization with an identity matrix, we initialze $\mathbf{V}$ randomly with uniform noise. Secondly, the matrix $\mathbf{W}$ can no longer be constrained to be an orthogonal rotation, so it is estimated with rows constrained to unit L_2 norm.

To analyze the effect of the random initialization separately from that of overcom-



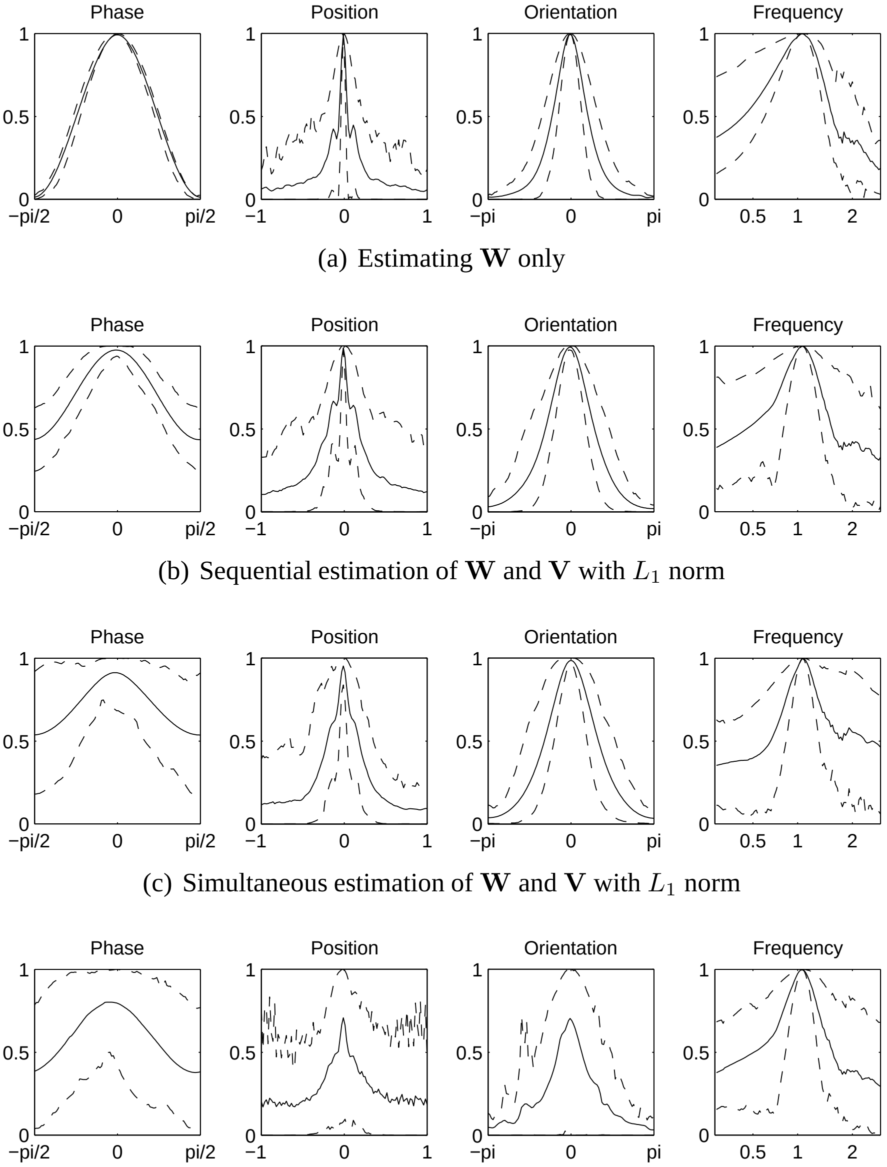
(a) Higher order features, L_1-norm

(b) Higher order features, L_2-norm

(c) Pooling patterns and second layer, L_1-norm

(d) Pooling patterns and second layer, L_2-norm

Figure 3: *(a-b)* A random selection of 24 higher order features, corresponding to individual rows of $\mathbf{V}$. Each feature is represented by a number of ellipses corresponding to individual first layer basis functions with the same orientation and position as the ellipse. Spatial phase is not shown in this representation. Each unit can be seen to pool over a small number of basis functions that tend to be iso-oriented and co-localized. This is typical behavior for complex cell receptive fields. While the L_1-norm penalty in *(a)* leads to a relatively uniform population of outputs, the features with an L_2-norm constraint in *(b)* show a distinct splitting into two sub-populations: Some features pool over a larger number of inputs and lose much of the location selectivity, while the rest of the features pool over fewer features than with the L_1-norm.
(c-d) Left hand side: Pooling patterns visualized in more detail by plotting the most active linear filters contributing to some randomly selected higher order units. Each row corresponds to one output and the black bars represent the relative strength of the linear filter inputs. Right hand side: Plot of the second layer matrix $\mathbf{V}$.



(a) Estimating $\mathbf{W}$ only

(b) Sequential estimation of $\mathbf{W}$ and $\mathbf{V}$ with L_1 norm

(c) Simultaneous estimation of $\mathbf{W}$ and $\mathbf{V}$ with L_1 norm

(d) Simultaneous estimation of $\mathbf{W}$ and $\mathbf{V}$ with L_2 norm

Figure 4: Analysis of complex cells properties of the second layer outputs, following (Hyvärinen & Hoyer, 2001). One parameter of the fitted Gabor was changed at a time, and the normalized response was plotted as a function of the tuning parameter. The solid line shows the mean response of 120 tested cells, the dashed lines give 10% and 90% quantiles.
(a) Only the first layer $\mathbf{W}$ was estimated and $\mathbf{V}$ was fixed to identity.
(b) After $\mathbf{W}$ had converged it was held constant and $\mathbf{V}$ was estimated using this constant first layer.
(c) $\mathbf{W}$ was initialized as above, but then both layers were estimated simultaneously. This shows significantly less phase sensitivity in the tuning curves, indicating that $\mathbf{W}$ has adapted to the pooling imposed by $\mathbf{V}$.
(d) Responses obtained with L_2-normalization under simultaneous optimization. Not only is some of the phase invariance lost, but position and orientation tuning are significantly worse than for the L_1 case.

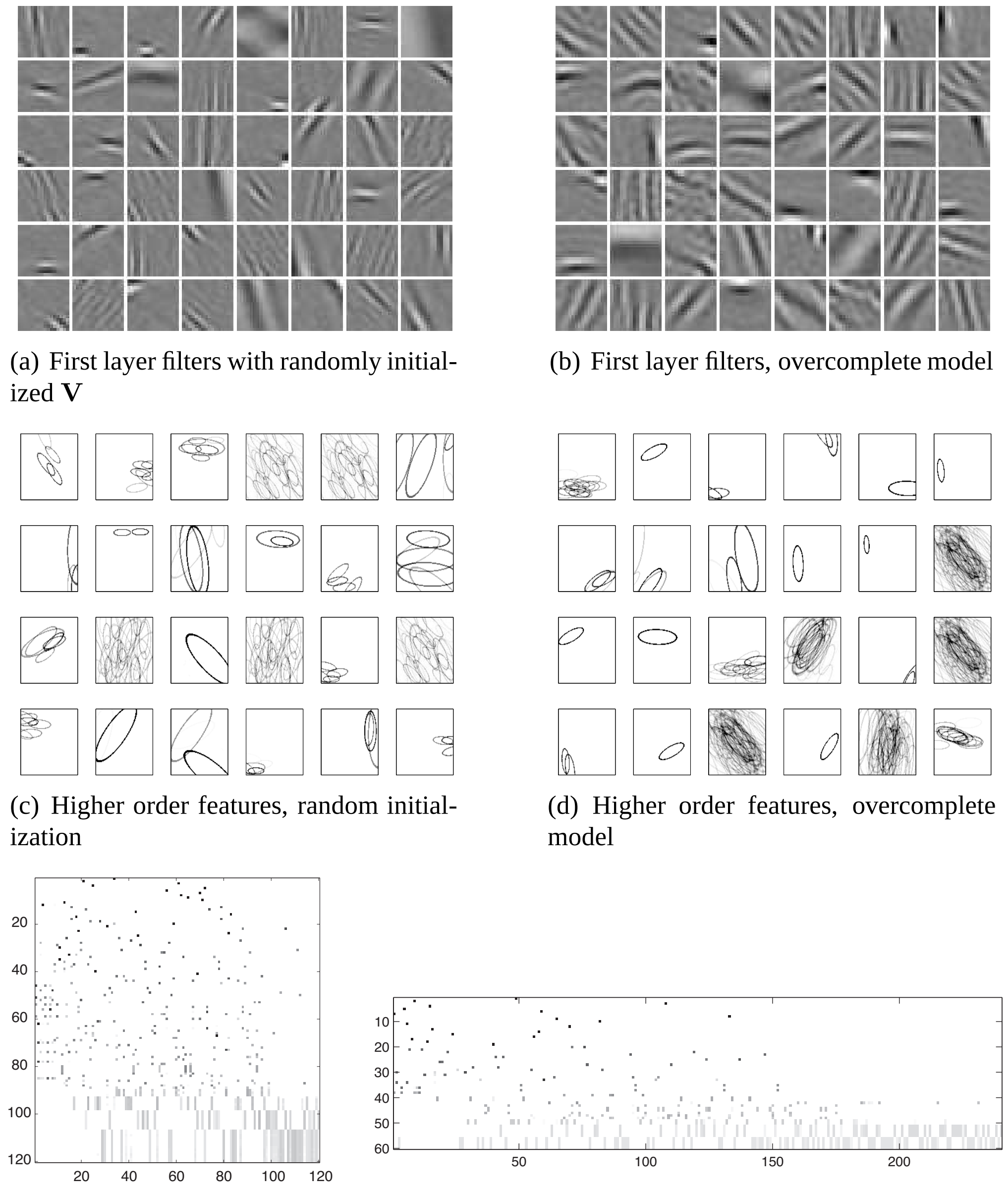
pleteness, we compare the results from the overcomplete model with those from a complete model with a randomly initialized second layer. Both models were estimated with L_2 normalization on the second layer. Figure 5 shows the results in the same way as the previous plots, i.e. the first and second layer features and pooling patterns for these models. For the overcomplete model, it can be seen that some of the features, in particular at higher frequencies, are less localized than in the models with ICA initialization. The reason for this is evident when considering the pooling patters: there are many higher order units pooling over a large number of linear filters, so selectivity in these features is reduced and more global pooling patterns emerge. It is also worth pointing out that some of the basis functions in the overcomplete case contain multiple Gabor functions, indicating convergence to a local minimum. This problem is not due to the random initialization as can be seen from the complete model, which in fact has a $\tilde{J} = -79.1$ and is thus not significantly different in quality from the diagonally initialized model we considered earlier. Rather, we suggest that the orthogonality of $\mathbf{W}$ is an important requirement to avoid convergence to local minima.

5.4 Estimation without non-negativity constraints

In addition to the experiments with a non-negative second layer, we also estimated the model with two different nonlinearities that did not require a non-negativity constraint, allowing us to model negative energy correlations in addition to positive ones. First, we analyzed the case of a symmetric nonlinearity $f(u) = \log\cosh(u)$, which leads to an output distribution that is sparse for both negative and positive outputs. Additionally we report on the results with an asymmetric nonlinearity, where the negative half of the nonlinearity corresponds to a low variance Gaussian distribution, and the positive half follows a logistic distribution. In both cases an L_2-norm constraint was imposed on rows of $\mathbf{V}$, which was initialized randomly with Gaussian white noise. Moreover, the first-layer nonlinearity was set as $g(u) = \log\cosh(u)$ instead of the squaring, in order to be able to better model a heavy-tailed distribution in conjunction with the second nonlinearity.

In the symmetric model (results not shown), we found sparse connectivity of the second layer, but with higher order features very different from the complex cell-like responses of the non-negative model: For some of the outputs, the first layer forms pairs of features which both contain two Gabors in their receptive field, one that is identical in both features, and one that is identical but of opposite sign, as in (Lindgren & Hyvärinen, 2006). Individual higher order outputs pool one such pair of features with one strong negative and one positive weight. Using the identity $a^2 - b^2 = (a-b)(a+b)$, this can be interpreted as taking the product of the sum and difference of the two linear filters, which turns out to be the product of two individual Gabor filters. Thus the model is effectively taking products of linear filter outputs, with results very similar to those observed previously in quadratic ICA (Lindgren & Hyvärinen, 2006). In the model with sparse positive and Gaussian negative outputs (results not shown), we report higher order features with one or a small number of highly active positive inputs, and a larger number of small negative inputs. This could possibly be interpreted as surround-inhibition, or a gain control phenomenon.

Thus we see that the results depend strongly on the choice of the nonlinearity f, and



(a) First layer filters with randomly initialized $\mathbf{V}$

(b) First layer filters, overcomplete model

(c) Higher order features, random initialization

(d) Higher order features, overcomplete model

(e) Second layer V, random initialization

(f) Second layer V, overcomplete model

Figure 5: Comparison of a complete and a four times overcomplete model, both with randomly initialized $\mathbf{V}$, and L_2-normalization on the second layer. *(a-b)* shows a subset of the filters in $\mathbf{W}$, for the two models. Random initialization does not lead to qualitative differences, but the overcomplete model learns features with are less localized and more frequency selective. *(c-d)* shows a representation of the higher order features (see Fig. 3 for details). The overcomplete model can be seen to give rise to a population of highly selective outputs which include only a single linear filter as well as a population of largely invariant higher order units pooling over a large fraction of inputs and retaining only orientation selectivity. *(e-f)* shows the second layer weight matrix $\mathbf{V}$ with inputs arranged by frequency and outputs ordered by sparseness of the pooling.

very different results can be obtained by changing the nonlinearity. It seems difficult to make a principled choice of f, because the usual measures of sparseness do not easily generalize to two-layer models. In future research, f could possibly be estimated from the data. In the current work, we choose to analyze only the results from the non-negative model, because it seems to be most in line with the visual processing in mammalian visual cortex, i.e. complex cell responses. Furthermore, we view our model as a direct extension to models with non-negative second layers such as ISA and topographic ICA. These have previously been motivated by the observation of strong positive energy correlations between linear filter outputs, so non-negativity seems to be a reasonable constraint.

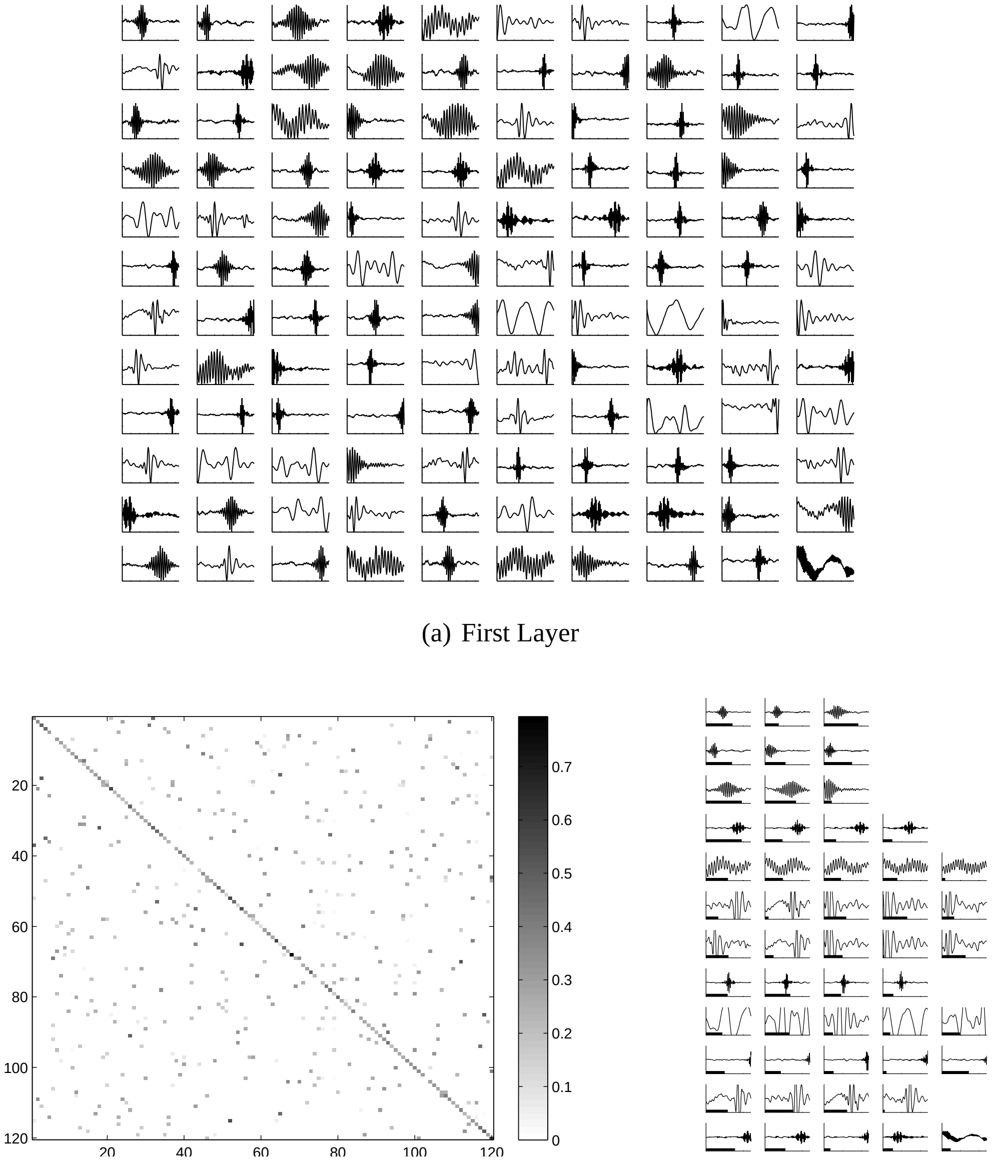
6 Experiments on Audio data

In order to demonstrate the general applicability of our model to a variety of data sets, we also tested it on speech data from the TIMIT database. The model was estimated in exactly the same way as for image data, and the data was preprocessed as follows: We took 100 short utterances from the database, and resampled them to 8kHz. The data was high-pass filtered with a cutoff at 100Hz and then normalized to unit variance. We sampled 10,000 random sound windows of 16ms length, which corresponds to 128-dimensional data and removed the DC component. We also applied our standard preprocessing consisting of whitening and contrast gain control, as described above for image data. Simultaneously we reduced the dimensionality from 128 to 120, which amounts to low-pass filtering and serves to eliminate artifacts due to the windowing procedure. The rows of the second layer were constrained to unit L_1-norm.

The results we obtained from the speech data are remarkably similar to those from image data and are presented in Figure 6. In *(a)* we show the first layer features in the time domain, which are tuned to specific frequency bands as well as onset time and duration. The second layer in *(b)* shows that a sparse connectivity has been learned between groups of first layer features. This is analyzed further in *(c)*, where we we show which first layer features are pooled in the second layer. We obtain higher order features where similar frequencies with slightly different temporal onset are pooled. Interestingly, the pooling size is considerably smaller than for image data, some of the outputs have as few as three contributing first order features. This indicates that individual linear filters outputs are closer to independent for audio data than for images, and that residual dependencies after the first layer are smaller.

7 Discussion

Hierarchical models have a long history in computational neuroscience and machine learning, and they are a promising approach to modelling the complicated structure of natural signals. However, it has not been feasible until recently to estimate multiple layers of these models from the data. The estimation can be difficult in different ways, such as the need to integrate over latent variables in generative models, or the computation of the intractable partition function in energy-based models. In this work we have



(a) First Layer

(b) Second Layer

(c) Some pooling patterns

Figure 6: Experiments for speech data from the TIMIT database.
(a) The first layer gives outputs localized in both frequency and time.
(b) The second layer shows connections between features with dependencies of squares.
(c) A random selection of output units. Each row shows the active first layer filters in one row of $\mathbf{V}$.

used Score Matching for estimating an energy-based model while avoiding to compute partition function, and we have shown how both layers of a two-layer model can be estimated from the data.

Two recent models have a similar hierarchical structure but are estimated in a different way from the model considered here. In particular the hierarchical Product of Experts (PoE, Osindero et al., 2006) is closely related to our work. Instead of using the "independent component" point of view, the model is defined as an overcomplete "product of experts" model with experts following Student-t distributions. The estimation is performed using Contrastive Divergence, which was recently shown (Hyvärinen, 2007a) to be related to Score Matching. The results obtained are similar to those reported here: Estimating the model on natural images also leads to pooling of linear filters with similar position and orientation, but different spatial phase. However, the authors do not show the pooling patterns, so it is not clear whether they also observed sparse connectivity in the second layer. A surprising difference to our model is that the authors observe it makes little difference whether the second layer is estimated on top of a fixed, predetermined first layer. However, model comparison is not straightforward with CD, which does not provide an objective function, so it is not clear if there is no further change in the linear filters in the PoE model, or if the authors simply did not investigate this further. In any case, the change in first layer units, which leads a significantly more complex cell-like behavior of output units, shows the importance of estimating the layers simultaneously in our model. The first layer features adapt to the structure of the second layer by changing the spatial phase tuning, which results in a better model fit as gauged by the Score Matching objective function, as well as increasing the phase invariance of the output units.

The hierarchical Bayesian model by Karklin & Lewicki (2005, 2006) is also related to our work in that the authors estimate two layers of a hierarchical model of natural images, but it is different in that the model is generative rather than energy-based. It is related to the generative topographic ICA model (Hyvärinen et al., 2001), where sources are not generated independently, but have dependencies introduced by multiplication with hidden variance variables, which are themselves given by a linear mixing of higher order sources. An exact estimation of such a generative model is not tractable because it requires integration over the possible states of the higher order variance variables. The authors thus resort to using a maximum a posteriori (MAP) approximation for the higher order sources. Similar to our results, the authors obtain Gabor-like filters in the first layer, but the second layer, which describes patterns of variance dependencies between linear filters, is quite different from the results of our energy-based models. The authors report broadly tuned features, encoding global properties of the data, and do not obtain simple pooling patters with e.g. common orientations and spatial frequencies. In agreement with our results, the linear filters in the first layer are reported to change depending on the pooling patterns in the second layer.

While the sparseness of the connectivity in the second layer is an important factor in obtaining complex cell-like responses, it is important to note that sparseness was not explicitly enforced in our model. Experiments with L_2-normalization and random initialization of $\mathbf{V}$ still result in sparse connectivity, even though the output units lose much of their complex cell properties. The non-negativity and L_1-norm penalty appear to be important factors in obtaining a uniform population of phase invariant higher order cells which pool over a small number of first order units. The L_1-norm can be related to energy efficient coding and wiring length constraints, which has been proposed to play a role in shaping the receptive fields in retinal cells (Vincent et al., 2005) and may be of similar importance for early visual cortex. With the L_2-norm we observed a fragmentation into two populations of output cells with different pooling patterns. In addition to one very sparse population, a second population pools over a larger number of inputs, and loses much selectivity.

8 Conclusion

We have presented an energy-based model of natural images and sounds with two layers of weights estimated from the data. The two layer model was estimated using Score Matching, which allows estimation without knowledge of the partition function. On natural images, the estimation of both layers with an L_1-norm penalty on the second layer leads to the emergence of complex cell properties for the higher order units. We analyze how the model parameters differ if the second layer is estimated on top of a fixed ICA basis and report that the phase tuning of the linear filters changes when both layers are estimated simultaneously, which results in an increase in phase invariance of the outputs. We performed experiments with a randomly initialized second layer, L_2-normalization and without non-negativity constraints, which all lead to sparse pooling but less complex cell-like outputs.

Acknowledgments

Both authors were supported by the Academy of Finland, project #111786 and the Centre-of-Excellence in Algorithmic Data Analysis (Algodan). Urs Köster was supported by a scholarship from the *Alfried Krupp von Bohlen und Halbach-Stiftung*.

Appendix

Derivatives of the objective function

We need to evaluate the gradients in the objective function (equation 24) w.r.t. the elements of $\mathbf{W}$ and $\mathbf{V}$. Since the expression can readily separated into a sum of three terms, which we call A, B and C, we treat these separately. We get six terms $\frac{\partial A}{\partial \mathbf{W}}$, $\frac{\partial B}{\partial \mathbf{W}}$, etc. Writing these out, we get for the first term

$$\begin{aligned}
\frac{\partial A}{\partial w_c^d} &= \sum_{k=1}^{n}\sum_{h=1}^{o}\sum_{\ell=1}^{m} v_h^\ell \frac{\partial}{\partial w_c^d}\left[(w_\ell^k)^2 g_\ell''(.) f'(\sum_i v_h^i g_i(.))\right] \tag{25}\\
&= \sum_{h}^{o}\Bigg\{ v_h^c 2 w_c^d g_c''(.) f_h'(.) \tag{26}\\
&+ \sum_{k}^{n} v_h^c (w_c^k)^2 g_c'''(.) x_d f_h'(.) \tag{27}\\
&+ \sum_{k}^{n}\sum_{\ell}^{m} v_h^\ell (w_\ell^k)^2 g_\ell''(.) f_h''(.) v_h^c g_c'(.) x_d \Bigg\} \tag{28}
\end{aligned}$$

For the second term we get

$$\begin{aligned}
\frac{\partial B}{\partial w_c^d} &= \sum_{k=1}^{n}\sum_{h=1}^{o} \frac{\partial}{\partial w_c^d} f_h''(.) \left[\sum_{\ell=1}^{m} w_\ell^k v_h^\ell g_\ell'(.)\right]^2 \tag{29}\\
&= \sum_{h}^{o}\Bigg\{ \sum_{k}^{n} f_h'''(.) v_h^c g_c'(.) x_d \left[\sum_{\ell=1}^{m} w_\ell^k v_h^\ell g_\ell'(.)\right]^2 \tag{30}\\
&+ f_h''(.) 2 \left[\sum_{\ell=1}^{m} w_\ell^d v_h^\ell g_\ell'(.)\right] \left[v_h^c g'(.)\right] \tag{31}\\
&+ \sum_{k}^{n} f_h''(.) 2 \left[\sum_{\ell=1}^{m} w_\ell^k v_h^\ell g_\ell'(.)\right] \left[w_c^k v_h^c g_c''(.) x_d\right] \Bigg\} \tag{32}
\end{aligned}$$

and for the third term, the derivative is

$$\frac{\partial C}{\partial w_c^d} = \sum_{k=1}^{n}\left[\sum_{h=1}^{o}\sum_{\ell=1}^{m} w_\ell^k v_h^\ell g_\ell'(.) f_h'(.)\right] \sum_{h=1}^{o}\sum_{\ell=1}^{m}\left[\frac{\partial}{\partial w_c^d} w_\ell^k v_h^\ell g_\ell'(.) f_h'(.)\right] \tag{33}$$

For better readability we substitute $\left[\sum_{h=1}^{o}\sum_{\ell=1}^{m} w_\ell^k v_h^\ell g_\ell'(.) f_h'(.)\right] = A_k$ in all subsequent equations.

$$\begin{aligned}
\frac{\partial C}{\partial w_c^d} &= A_d \left[\sum_{h} v_h^c g_c'(.) f_h'(.)\right] \tag{34}\\
&+ \sum_{k=1}^{n} A_k \left[\sum_{h} v_h^c w_c^k g''(w_c^d) x_d f_h'(.)\right] \tag{35}\\
&+ \sum_{k=1}^{n} A_k \left[\sum_{h,\ell} v_h^l w_\ell^k g_\ell'(.) f_h''(.) v_h^c g_c'(.) x_d\right] \tag{36}
\end{aligned}$$

Next we evaluate the derivatives for V, for the first term:

$$\begin{aligned} \frac{\partial A_k}{\partial v_a^b} &= \sum_{k=1}^{n}\sum_{h=1}^{o}\sum_{\ell=1}^{m}\frac{\partial}{\partial v_a^b}\left[(w_\ell^k)^2 v_h^\ell g_\ell''(.) f_h'(.)\right] && (37) \\ &= \sum_{k}^{n}\left\{\sum_{\ell}^{m}(w_\ell^k)^2 v_a^\ell g_\ell''(.) f''(v_a^b g_b) g_b + (w_b^k)^2 g_b''(.) f_h'(.)\right\} && (38) \end{aligned}$$

the second term:

$$\begin{aligned} \frac{\partial B_k}{\partial v_a^b} &= \sum_{k=1}^{n}\sum_{h=1}^{o}\frac{\partial}{\partial v_a^b} f_h''(.)\left[\sum_{\ell=1}^{m} w_\ell^k v_h^\ell g_\ell'(.)\right]^2 && (39) \\ &= \sum_{k}^{n}\left\{ f_a'''(.) g_b(.)\left[\sum_{\ell=1}^{m} w_\ell^k v_h^\ell g_\ell'(.)\right]^2\right. && (40) \\ &+ \left. f_a''(.) 2\left[\sum_{\ell=1}^{m} w_\ell^k v_h^\ell g_\ell'(.)\right] w_b^k g_b'(.)\right\} && (41) \end{aligned}$$

and similar for the third term:

$$\begin{aligned} \frac{\partial C}{\partial v_a^b} &= \sum_{k=1}^{n}\left[\sum_{h=1}^{o}\sum_{\ell=1}^{m} w_\ell^k v_h^\ell g_\ell'(.) f_h'(.)\right]\sum_{h=1}^{o}\sum_{\ell=1}^{m}\left[\frac{\partial}{\partial v_a^b} w_\ell^k v_h^\ell g_\ell'(.) f_h'(.)\right] && (42) \\ &= \sum_{k=1}^{n} A_k \sum_{h=1}^{o}\sum_{\ell=1}^{m}\left[\frac{\partial}{\partial v_a^b} w_\ell^k v_h^\ell g_\ell'(.) f_h'(.)\right] && (43) \\ &= \sum_{k=1}^{n}\left\{A_k w_b^k g_b'(.) f_a'(.) + A\sum_{\ell} w_l^k g_l'(.) v_a^l f_a''(.) g_b(.)\right\} && (44) \\ & && (45) \end{aligned}$$

References

Adelson, E., & Bergen, J. (1985). Spatiotemporal energy models for the perception of motion. *J. Opt. Soc. Am. A 2,* (pp. 284 – 299).

Barlow, H. (1961). *Possible principles underlying the transformation of sensory messages*. Cambridge, MA: MIT Press. W. Rosenblith (Ed.) Sensory Communication.

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

Hinton, G. E. (2002). Training products of experts by minimizing contrastive divergence. *Neural Computation, 14*(8), 1771–1800.

Hubel, D. H., & Wiesel, T. N. (1959). Receptive fields of single neurones in the cat's striate cortex. *J Physiol., 148,* 574 – 591.

Hubel, D. H., & Wiesel, T. N. (1962). Receptive fields, binocular interaction and functional architecture in cat's visual cortex. *J Physiol.*, *160,* 106 – 154.

Hyvärinen, A. (2005). Estimation of non-normalized statistical models using score matching. *Journal of Machine Learning Research, 6:695–709*.

Hyvärinen, A. (2007a). Connections between score matching, contrastive divergence, and pseudolikelihood for continuous-valued variables. *IEEE Transactions on Neural Networks*, *18*(5), 1529–1531.

Hyvärinen, A. (2007b). Some extensions of score matching. *Computational Statistics and Data Analysis*, *51*, 2499–2512.

Hyvärinen, A., Gutmann, M., & Hoyer, P. (2005). Statistical model of natural stimuli predicts edge-like pooling of spatial frequency channels in V2. *BMC Neuroscience*, *6*(12).

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

Hyvärinen, A., Hoyer, P., & Inki, M. (2001). Topographic independent component analysis. *Neural Computation.*, *13*(7), 1527–1558.

Hyvärinen, A., & Hoyer, P. O. (2001). A two-layer sparse coding model learns simple and complex cell receptive fields and topography from natural images. *Vision Research*, *41*, 2413 – 2423.

Hyvärinen, A., Hurri, J., & Hoyer, P. O. (2009). *Natural Image Statistics*. Springer-Verlag.

Hyvärinen, A., & Köster, U. (2007). Complex cell pooling and the statistics of natural images. *Network: Computation in Neural Systems*, *18*, 81–100.

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

Karklin, Y., & Lewicki, M. S. (2005). A hierarchical bayesian model for learning nonlinear statistical regularities in non-stationary natural signals. *Neural Computation*, *17*(2), 397–423.

Karklin, Y., & Lewicki, M. S. (2006). Is early vision optimized for extracting higher-order dependencies? *Advances in Neural Information Processing Systems*, *18*, 625–642.

Köster, U., & Hyvärinen, A. (2007). A two-layer ICA-like model estimated by score matching. In *Artificial Neural Networks - ICANN 2007, Lecture Notes in Computer Science*, (pp. 798–807). Springer Berlin / Heidelberg.

Lindgren, J. T., & Hyvärinen, A. (2006). Emergence of conjunctive visual features by quadratic independent component analysis. *Advances in Neural Information Processing Systems*.

Olshausen, B., & Field, D. (1996). Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, *381*, 607–609.

Olshausen, B. A., & Field, D. J. (1997). Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision Research*, *37*(23), 3311–3325.

Osindero, S., Welling, M., & Hinton, G. E. (2006). Topographic product models applied to natural scene statistics. *Neural Computation*, *18*, 381–414.

Pollen, D., & Ronner, S. (1983). Visual cortical neurons as localized spatial frequency filters. *IEEE Trans. on Systems, Man, and Cybernetics*, *13*, 907–916.

Schwartz, O., & Simoncelli, E. P. (2001). Natural signal statistics and sensory gain control. *Nature Neuroscience*, *4*(8), 819–825.

Spitzer, H., & Hochstein, S. (1985). A complex-cell receptive-field model. *J. Neurophysiol. 53*, (pp. 1266 – 1286).

Vincent, B., Baddeley, R., Troscianko, T., & Gilchrist, I. (2005). Is the early visual system optimised to be energy efficient? *Network*, *16*(2-3), 175–190.