LETTERS

Emergence of complex cell properties by learning to generalize in natural scenes

Yan Karklin[1]† & Michael S. Lewicki[1]†

A fundamental function of the visual system is to encode the building blocks of natural scenes—edges, textures and shapes—that subserve visual tasks such as object recognition and scene understanding. Essential to this process is the formation of abstract representations that generalize from specific instances of visual input. A common view holds that neurons in the early visual system signal conjunctions of image features[1,2], but how these produce invariant representations is poorly understood. Here we propose that to generalize over similar images, higher-level visual neurons encode statistical variations that characterize local image regions. We present a model in which neural activity encodes the probability distribution most consistent with a given image. Trained on natural images, the model generalizes by learning a compact set of dictionary elements for image distributions typically encountered in natural scenes. Model neurons show a diverse range of properties observed in cortical cells. These results provide a new functional explanation for nonlinear effects in complex cells[3–6] and offer insight into coding strategies in primary visual cortex (V1) and higher visual areas.

As we scan across a complex natural scene, fixations at multiple locations (for example, on the trunk of a tree or along its edge) produce a coherent percept of the underlying structure (the bark texture or the contour of the edge), even though individual images collected at the retina are inherently highly variable. Figure 1 illustrates the problem our brain solves so effortlessly: perceptually distinct image regions produce response patterns that are highly overlapping and cannot be easily distinguished using low-level, linear representations. What sort of computations are required to achieve generalization across natural stimuli?

Early visual neurons are typically described as linear feature detectors[1,2]. Models developed around this idea can accurately capture the behaviour of neurons from the retina[7] to simple cells in the cortex[8] but, as the examples in Fig. 1 illustrate, neither individual features nor linear transformations can reliably discriminate images of one structure from another. More abstract features are presumably computed in later stages of the visual system, but our knowledge of processing by these neurons is limited. In V1, complex cells respond to an edge over a range of positions[1], but classical models of these cells[9,10] fail to explain a number of nonlinear effects, such as surround suppression and cross-orientation inhibition[3–5]. More importantly, there is no functional explanation for the role of these behaviours in the perception of natural scenes. In higher visual areas such as V2 and V4, neurons are more invariant to image properties such as position and scale[11–13] and might be encoding shape or texture[12,14,15]. For these neurons to generalize effectively, the neural circuitry must generate a representation that is similar across the wide distribution of images of a given type (for example, a texture or contour) yet distinct across the much larger distribution of all other images.

Previous theoretical work has shown that neurons in the primary visual cortex form an efficient code adapted to the statistics of natural images[16,17], but this says nothing about how neurons generalize across

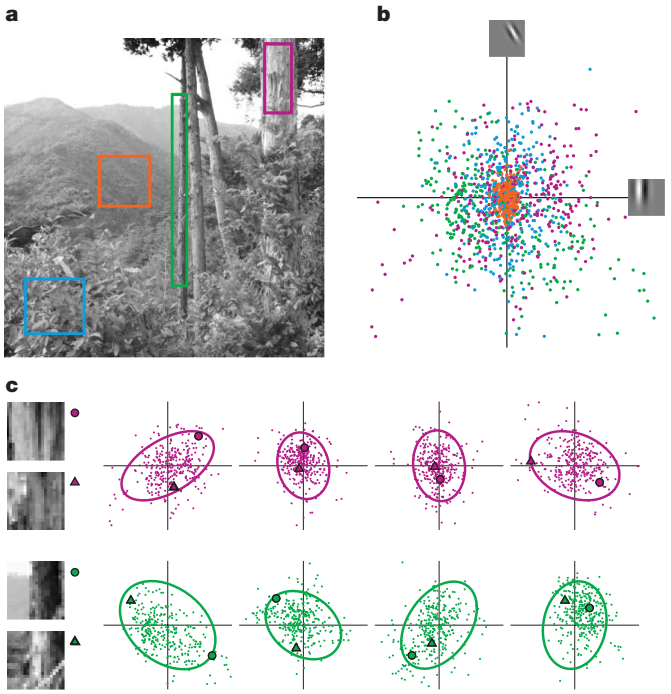


Figure 1 | Statistical patterns distinguish local regions of natural scenes. **a**, A natural scene with four distinct regions outlined (image courtesy of E. Doi). **b**, The scatter plot shows the joint output of a pair of linear feature detectors (oriented Gabor filters) for 20 × 20-image patches sampled from the four regions. The outputs from different regions are highly overlapping, indicating that linear features provide no means to distinguish between the regions. **c**, Each column shows the joint output of a different pair of linear feature detectors sampled from the regions containing the tree bark or the tree edge (the first column corresponds to features in **b**). The correlations in each panel can be described by a Gaussian distribution and its covariance (ellipses). The differences in the distributions between the rows reveal characteristic patterns in correlations, which become even more prominent as projections onto more features are considered. These patterns can be used to generalize within regions while still distinguishing among them. As an example, we highlight two patches in each region, shown by the circle and triangle in each panel. Although the pairs of images are visibly quite different, each image is consistent with the distribution of the local image region. By contrasting the distributions across multiple dimensions, it is possible to infer image type for single patches, even if the patches have similar projections along some image features.

[1]Computer Science Department & Center for the Neural Basis of Cognition, Carnegie Mellon University, Pittsburgh, PA 15213, USA. †Present address: Center for Neural Science, New York University, New York, New York, USA (Y.K.); Electrical Engineering and Computer Science Department, Case Western University, Cleveland, Ohio, USA and Wissenschaftskolleg (Institute for Advanced Study) zu Berlin, Germany (M.S.L.).

the intrinsic variability of scene elements. Here we extend the efficient coding approach and propose that an important aspect of visual computation is to represent the myriad statistical distributions that characterize local image regions. Rather than coding the pixel intensities of a patch of texture or edge, neurons in later stages encode the image distribution (that is, the range and pattern of variability of pixel intensities or image features) that is most consistent with the input image. This allows the neural representation to generalize across individual fixations and convey more abstract properties of the image. We demonstrate that a model designed around this computational goal and optimized for natural scenes explains nonlinear properties of complex cells and neurons in higher visual areas, thereby providing a new functional interpretation for these effects.

Fundamentally, generalization is the identification of common characteristics of a class from specific instances. The goal of the proposed model is to learn the statistical distributions that characterize local image regions, such as those in Fig. 1, and identify them from individual image patches. What statistical regularities are relevant for this task? As the examples in Fig. 1 suggest, the distributions of perceptually similar images show consistent patterns in the degree of variation along some dimensions, as well as in the strength of correlations (and anti-correlations) among different feature dimensions. Although these patterns appear subtle when projected onto two dimensions, as in the examples, the full multivariate distribution, consisting of hundreds of dimensions, produces prominent statistical signatures that can be exploited by the visual system.

To determine how the model generalizes, we must specify how it represents distributions of local image regions. A simple way to summarize the patterns of correlations for a given type of image is the covariance matrix of the data. A neural code for this structure could be defined by enumerating the set of observed covariances and assigning one neuron to each pattern, but this approach presents two problems. First, local image classes are not known a priori. Second, given the limited number of neurons in the visual system, it is not feasible to represent all possible image types, let alone the combinatorial number of possible image boundaries. Instead, we propose a distributed code in which the graded activity of the neural population acts to describe a continuum of potential covariance patterns.

This distribution coding model is illustrated schematically in Fig. 2. The model represents the correlations present in local image regions with a multivariate Gaussian distribution that has a fixed mean of zero and a covariance that is a function of the neural activity (see Methods). This simple statistical description affords both the flexibility to capture a continuum of natural image distributions and mathematical simplicity for tractable parameter estimation. The model uses two sets of parameters to describe correlations in image distributions. First, the vectors $\mathbf{b}_k$ (arrows within circles) specify image features along which the encoded distribution can be expanded or contracted relative to the canonical distribution (black circle). These vectors are shared by all neurons in the model (represented by the four grey circles, each of which contains the same set of arrows). Because these vectors do not necessarily line up with the axes of the input dimensions, changes in variation along a vector can correspond to changes in the correlational pattern in many dimensions at once. Neurons in the model (y_j) describe changes along these directions using weights w_{jk}: each has a different set of weights, corresponding to an expansion or contraction along feature $\mathbf{b}_k$. A positive weight (red) means that the neuron responds to a wider range of stimuli along that direction, a negative weight (blue) means it responds to a narrower range, and a weight close to zero (grey) indicates that the neuron is neutral to this direction. The combined activation of all neurons specifies the final shape of the encoded distribution (ellipses). Given a single fixation—one input image—the model computes the neural representation (that is, the image distribution) that provides the most probable explanation of the input. The model is able to generalize over different image regions if the inferred representation is similar across a region (for example, for the pairs of patches in Fig. 2).

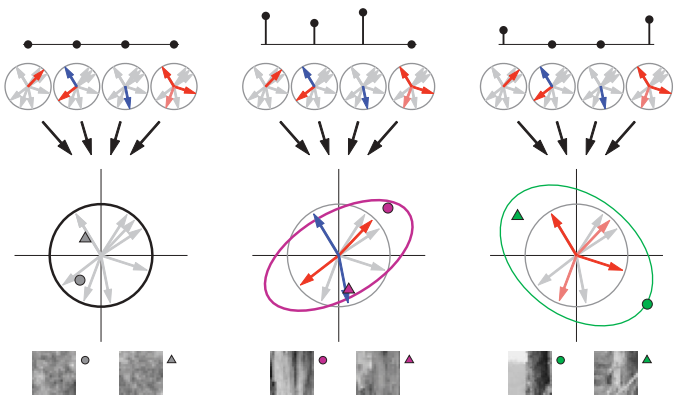


Figure 2 | Distribution coding model. Rather than encoding the precise pixel values of an input image (bottom), the proposed model infers for each image the most likely distribution (ellipses) containing it. Activation patterns for model neurons are shown at the top of each column. Absence of activity corresponds to the lack of image structure (left panel)—that is, a canonical distribution that reflects the statistics over all natural images (black circle). Increased neural activity represents deviations from this canonical distribution and captures statistical patterns in local image regions (middle and right panels, patches and symbols as in Fig. 1). In each panel, the activation pattern is the same for both inputs. See text for further details.

By adapting model parameters $\mathbf{b}_k$ and w_{jk} to the data, we are able to find the most efficient way to use a limited number of neurons to describe the wide range of distributions observed in natural images. It should be noted that, although our goal is to derive a stable representation of all patches within a local region, no assumptions about locality are made (encoding is done independently for each image patch). It is the task of the model to learn a compact representation of all patches and to automatically discover which share the statistical properties of a particular type.

If, as hypothesized, neurons in the visual cortex encode patterns in correlations in local regions and are adapted specifically to the statistics of natural scenes, we expect the representations learned by the model to reflect properties of visual neurons. To this end, we trained the model on patches sampled from a large set of natural images and examined the resulting parameters as well as the response properties of model neurons to natural images.

The vectors $\mathbf{b}_k$ encode the directions of common expansion or contraction in the shape of the image distribution. Drawn as image patches, each is an oriented and localized edge-like feature. The full set tiles the spatial extent of the image patch (Fig. 3a) and spans the range of orientations and spatial frequencies of natural images (not shown). These oriented, band-pass image features are consistent with the optimal images for exciting simple cells in the primary visual cortex[18,19]. Similar representations have been derived previously using linear statistical models that maximize the efficiency of the image codes[16,17]. In the model proposed here, however, these features are not used explicitly to reconstruct the original image, but instead function to modify the encoded distributions (arrows in Fig. 2). Thus, whereas the traditional interpretation of early sensory codes is that they are adapted for faithful reconstruction of the stimulus, our results suggest an additional interpretation: they convey variations in image distributions and allow downstream visual areas to form more abstract representations.

The second set of parameters, the weights w_{jk}, describes the role of each neuron in shaping the encoded image distribution. A set of learned weights for a typical model neuron is shown in Fig. 3b. This neuron exerts the strongest effect on features in the top left of the image patch, increasing the variability (that is, activation) of those oriented at its 'preferred' orientation of 45°, decreasing the variability of those at the orthogonal orientation, as well as those at the preferred orientation but at an offset location. Rather than

responding to a few excitatory or suppressive image features, the neuron integrates a large number to describe a pattern of variability underlying a particular image distribution. Although the functional significance of these subunits is to modify the statistical structure of the encoded distribution, they also reflect stimulus features to which this model neuron is most sensitive. It should be noted that a model neuron is activated by all images from this distribution, rather than signalling the presence of one best stimulus. Conversely, stimuli that lie in parts of image space assigned low probability by the neuron inhibit its activity.

To compare the behaviour of the model neuron to that of cells in the visual cortex, we tested its response to stimuli used in classical physiological experiments (sinusoidal gratings). Model parameters were fixed after training on natural images, and neural response computed on a set of patterns centred in the visual area that evoked maximal response. This particular model neuron showed a variety of properties observed in complex cells in V1 and cells in V2, including phase invariance, orientation tuning and complex suppressive effects (Fig. 3c). A large subset of the population exhibits similar properties, whereas others encode more complex patterns that have been observed in higher visual areas V2 and V4 (a detailed analysis of the population and similarities to other experimental data are provided in the Supplementary Information). We emphasize that these results, as well as image features described in Fig. 3a, were obtained with no assumptions about the image structure encoded by visual neurons and without fitting a model to data from physiological experiments. Specifically, we did not restrict the encoded image features to be localized and oriented, nor did we prescribe in advance how the subunits are to be combined in the pattern represented by each neuron.

Finally, we looked at the way in which the model uses the population of neurons to represent images. If the model is able to generalize across the wide variability present in natural images, then image patches that are widely scattered in the original space of linear features should be tightly clustered in the space of the model's representation. This can be illustrated by projecting into two dimensions (as was done with image space in Fig. 1) the model representation of a collection of images (Fig. 4). As hypothesized, by encoding image distributions rather than the precise feature content of each image, model neurons are able to encode perceptually similar images with similar representations and to separate distinct image types.

One limitation of the statistical framework used here is that it does not furnish an explicit feed-forward algorithm for neural encoding. Nevertheless, it is possible to approximate inference in the model by a sequential feed-forward computation: a neuron integrates the squared responses of a large number of image features $\mathbf{b}_k$ and correlates the pattern against its weights w_{jk} (see Supplementary Information for details). This computation can be viewed as a generalization of the standard model of complex cells, in which each complex cell takes as input the squared output of two simple cells[9,10,20,21]. In contrast, model neurons can receive many inputs, and the linear features themselves are learned. We find that the optimal number of input features varies greatly, and the features are integrated in a variety of ways. These predictions are consistent with recent analyses of functional subfields in V1 complex cells[6,22]. In addition, some model neurons integrate more complex spatial patterns (see Supplementary Information), which predicts a neural response to a richer variety of images than has been tested

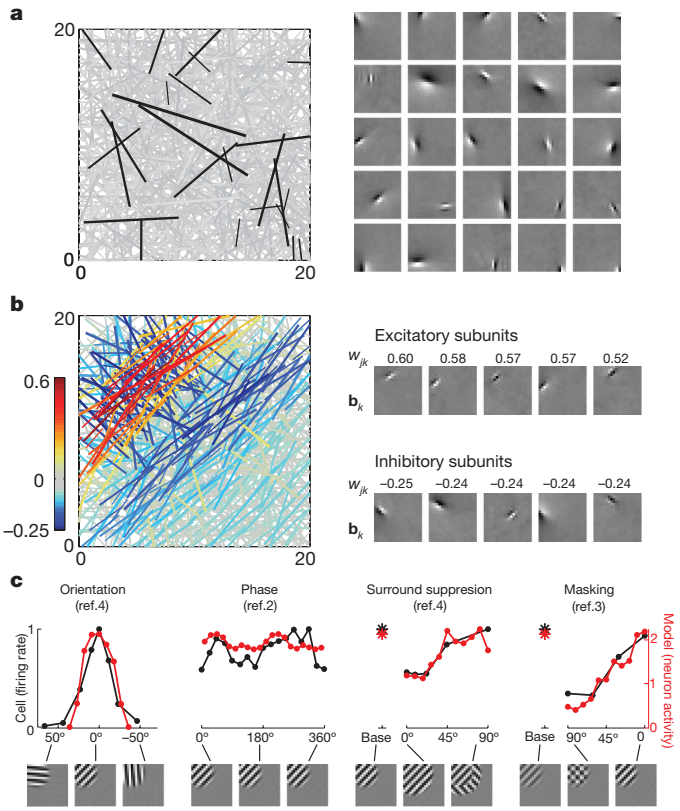


Figure 3 | Model neurons exhibit properties of cortical visual neurons. **a**, When adapted to natural images, the vectors $\mathbf{b}_k$ are oriented, localized in space, and span the spatial extent of the 20 × 20-pixel image patch. Each line reflects the orientation, spatial position within the image patch, and scale (length of line) of one of the image features. Twenty-five representative features (from a total of 1,000) are drawn in black, and shown in image form on the right. **b**, Weights of one typical model neuron to the features $\mathbf{b}_k$. As in **a**, each feature is represented by a line, and the colour of the line indicates the sign and magnitude of the weight to the feature (see colour bar). Positive weights indicate increased variability in the corresponding feature; negative weights indicate decreased variability; features to which the neuron is insensitive are shaded grey. Image features ($\mathbf{b}_k$) corresponding to the five most positive and the five most negative weights are shown in the right panel; the corresponding weights are above each feature. These act as excitatory and inhibitory subunits for this neuron. **c**, When presented with sinusoidal gratings, this model neuron replicates common aspects of the neural response in complex cells in cortical area V1. It is highly tuned to the grating's orientation, but insensitive to its phase. Adding a grating into the surrounding region suppresses the response (third plot, 0°) relative to baseline response to a single grating (asterisk), but this suppression is tuned to the orientation of the surround and is weakest when the surround is orthogonal to the preferred orientation (90°). Masking with a superimposed orthogonal grating suppresses the response (fourth plot, 90°), but this suppression is also orientation-dependent. All model neuron responses are plotted on the same scale (red axis); cell firing rates in each plot were normalized to a maximum value of 1; preferred orientation was shifted to 0° for the model neuron and the cell in all plots.

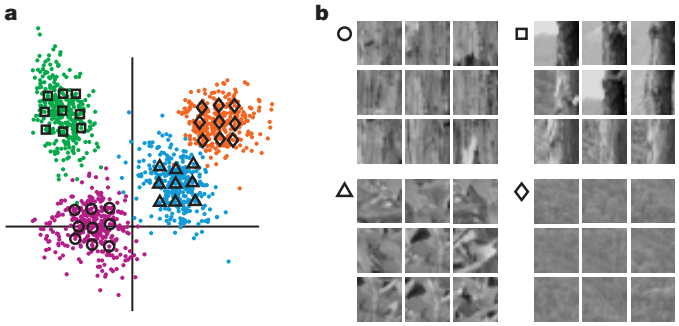


Figure 4 | Generalization across natural variability. **a**, In contrast to linear projections (compare to Fig. 1b), a two-dimensional projection of the model's representation (the activity of 150 model neurons) reveals well-separated clusters. **b**, Each 3 × 3-image group corresponds to the array of symbols in **a**. Despite the variability in the appearance of edges and textures, the model's representation of natural images generalizes within each region while still distinguishing among them.

physiologically. Experiments that incorporate such stimuli will provide an important validation of the proposed model.

The nonlinear effects shown by neurons in the model (Fig. 3c) have been previously incorporated into models of complex cells[5,8,20,21]. Much of this work has focused on fitting mathematical models to neural data[5,8,20,23] and does not provide a functional explanation of the observed neural properties. Other models have been motivated by specific computational goals, such as statistical independence[24,25], stability of representation over time[26,27], or position or scale invariance[28]. However, these models do not explicitly address the problem of generalization, which here is performed by inferring the statistical distribution that is most likely to explain the input image. An important advantage of our approach is that, rather than assuming invariance (or sensitivity) to limited stimulus parameters such as position or orientation, the model learns a much more general set of features that are determined by the statistical structures in natural images. If higher-level visual neurons are generalizing according to these statistics, they should have invariance along specific stimulus dimensions, and their responses to natural images should reflect common statistical structure in local image regions. Thus, the model provides a quantitative way to explore neural responses to complex stimuli characterized by their statistical regularities.

METHODS SUMMARY

The model describes individual image patches $\mathbf{x}$ with multivariate Gaussian probability distributions:

$$P(\mathbf{x}|\mathbf{y}) = \mathcal{N}(\boldsymbol{\mu}, \mathbf{C}) \quad (1)$$

with mean $\boldsymbol{\mu} = 0$ and with covariance a function of the neural encoding vector $\mathbf{C} = f(\mathbf{y})$. The logarithm of the covariance matrix is given by the combination of outer products of feature vectors $\mathbf{b}_k$, weighted by neural activities y_j through weights w_{jk}:

$$\log \mathbf{C} = \sum_{jk} y_j w_{jk} \mathbf{b}_k \mathbf{b}_k^T \quad (2)$$

Because a different covariance can be inferred for each image, the distribution over the entire ensemble of images is highly non-Gaussian. (This model is a generalized version of the hierarchical model described previously[29], which captured patterns among the variances, but not the correlations, of linear features.)

We trained the model on a large set of 20×20 image patches, sampled randomly from greyscale photographs of outdoor scenes[19]. The number of neurons was set to 150 and the number of linear features $\mathbf{b}_k$ to 1,000. The 'response' of model neurons was computed as the most probable neural representation given the input image by maximizing the posterior probability $P(\mathbf{y}|\mathbf{x}, \{\mathbf{b}_k, w_{jk}\})$. Model parameters were initialized to small random values and optimized by maximizing the likelihood of the data under the model $P(\mathbf{x}|\{\mathbf{b}_k, w_{jk}\})$ using standard gradient ascent.

For the 'physiological' analysis of Fig. 3c, we first identified the location, orientation, and spatial extent and frequency of a windowed sinusoidal grating that best activated the model neuron (one that yielded the most positive value of y_j). We then varied each tested parameter and computed the model's representation of the stimulus (the vector of responses of model neurons).

Full Methods and any associated references are available in the online version of the paper at www.nature.com/nature.

Received 4 May; accepted 26 September 2008.
Published online 19 November 2008.

1. Hubel, D. H. & Wiesel, T. N. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *J. Physiol. (Lond.)* **160,** 106–154 (1962).
2. Movshon, J. A., Thompson, I. D. & Tolhurst, D. J. Spatial summation in the receptive fields of simple cells in the cat's striate cortex. *J. Physiol. (Lond.)* **283,** 53–77 (1978).
3. Bonds, A. B. Role of inhibition in the specification of orientation selectivity of cells in the cat striate cortex. *Vis. Neurosci.* **2,** 41–55 (1989).
4. Jones, H. E., Wang, W. & Sillito, A. M. Spatial organization and magnitude of orientation contrast interactions in primate V1. *J. Neurophysiol.* **88,** 2796–2808 (2002).
5. Cavanaugh, J. R., Bair, W. & Movshon, J. A. Nature and interaction of signals from the receptive field center and surround in macaque V1 neurons. *J. Neurophysiol.* **88,** 2530–2546 (2002).
6. Chen, X., Han, F., Poo, M. & Dan, Y. Excitatory and suppressive receptive field subunits in awake monkey primary visual cortex (V1). *Proc. Natl Acad. Sci. USA* **104,** 19120–19125 (2007).
7. Chichilnisky, E. J. A simple white noise analysis of neuronal light responses. *Network: Comp. Neural Syst.* **12,** 199–213 (2001).
8. Carandini, M., Heeger, D. J. & Movshon, J. A. Linearity and normalization in simple cells of the macaque primary visual cortex. *J. Neurosci.* **17,** 8621–8644 (1997).
9. Movshon, J. A., Thompson, I. D. & Tolhurst, D. J. Receptive field organization of complex cells in the cat's striate cortex. *J. Physiol. (Lond.)* **283,** 79–99 (1978).
10. Adelson, E. H. & Bergen, J. R. Spatiotemporal energy models for the perception of motion. *J. Opt. Soc. Am. A* **2,** 284–299 (1985).
11. Kobatake, E. & Tanaka, K. Neuronal selectivities to complex object features in the ventral visual pathway of the macaque cerebral cortex. *J. Neurophysiol.* **71,** 856–867 (1994).
12. Gallant, J. L., Connor, C. E., Rakshit, S., Lewis, J. W. & Van Essen, D. C. Neural responses to polar, hyperbolic, and Cartesian gratings in area V4 of the macaque monkey. *J. Neurophysiol.* **76,** 2718–2739 (1996).
13. Connor, C. E., Brincat, S. L. & Pasupathy, A. Transformation of shape information in the ventral pathway. *Curr. Opin. Neurobiol.* **17,** 140–147 (2007).
14. Hegdé, J. & Van Essen, D. C. Selectivity for complex shapes in primate visual area V2. *J. Neurosci.* **20,** RC61:1–6 (2000).
15. Pasupathy, A. & Connor, C. E. Shape representation in area V4: position-specific tuning for boundary conformation. *J. Neurophysiol.* **86,** 2505–2519 (2001).
16. Olshausen, B. A. & Field, D. J. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature* **381,** 607–609 (1996).
17. Bell, A. J. & Sejnowski, T. J. The ''independent components'' of natural scenes are edge filters. *Vision Res.* **37,** 3327–3338 (1997).
18. Jones, J. P. & Palmer, L. A. The two-dimensional spatial structure of simple receptive fields in cat striate cortex. *J. Neurophysiol.* **58,** 1187–1211 (1987).
19. van Hateren, J. H. & van der Schaaf, A. Independent component filters of natural images compared with simple cells in primary visual cortex. *Proc. R. Soc. Lond. B* **265,** 359–366 (1998).
20. Heeger, D. J. Normalization of cell responses in cat striate cortex. *Vis. Neurosci.* **9,** 181–197 (1992).
21. Heeger, D. J., Simoncelli, E. P. & Movshon, J. A. Computational models of cortical visual processing. *Proc. Natl Acad. Sci. USA* **93,** 623–627 (1996).
22. Rust, N. C., Schwartz, O., Movshon, J. A. & Simoncelli, E. P. Spatiotemporal elements of macaque V1 receptive fields. *Neuron* **46,** 945–956 (2005).
23. Cadieu, C. *et al.* A model of V4 shape selectivity and invariance. *J. Neurophysiol.* **98,** 1733–1750 (2007).
24. Schwartz, O. & Simoncelli, E. P. Natural signal statistics and sensory gain control. *Nature Neurosci.* **4,** 819–825 (2001).
25. Hyvärinen, A. & Hoyer, P. A two-layer sparse coding model learns simple and complex cell receptive fields and topography from natural images. *Vision Res.* **41,** 2413–2423 (2001).
26. Berkes, P. & Wiskott, L. Slow feature analysis yields a rich repertoire of complex cell properties. *J. Vis.* **5,** 579–602 (2005).
27. Hurri, J. & Hyvärinen, A. Simple-cell-like receptive fields maximize temporal coherence in natural video. *Neural Comput.* **15,** 663–691 (2003).
28. Riesenhuber, M. & Poggio, T. Hierarchical models of object recognition in cortex. *Nature Neurosci.* **2,** 1019–1025 (1999).
29. Karklin, Y. & Lewicki, M. S. A hierarchical Bayesian model for learning non-linear statistical regularities in non-stationary natural signals. *Neural Comput.* **17,** 397–423 (2005).

Supplementary Information is linked to the online version of the paper at www.nature.com/nature.

Acknowledgements This work was supported by the Department of Energy through the Computational Science Graduate Fellowship (to Y.K.), the National Science Foundation Grant under grant numbers 0413152 and 0705677 (to M.S.L.) and the Office of Naval Research under the Multidisciplinary University Research Initiative N000140710747.

Author Contributions Y.K. and M.S.L. developed the model, analysed the results and wrote the paper; Y.K. ran the simulations.

Author Information Reprints and permissions information is available at www.nature.com/reprints. Correspondence and requests for materials should be addressed to Y.K. (yan.karklin@nyu.edu) or M.S.L. (michael.lewicki@case.edu).

METHODS

Data. We used 110 greyscale images of outdoor scenes as training data[19]. Pixel intensities were log-transformed (corresponding roughly to the transformation at the retinal cone cells[30]), and the images were low-pass filtered to remove corner frequency sampling artefacts. We randomly extracted overlapping 20×20-image patches from the entire data set. The mean luminance value was subtracted from each patch (which sped up model training but had no significant influence on the results). We 'whitened' all image patches to remove global correlations and to normalize the variance; this allowed the model to encode only the deviations of each image distribution from the global statistics (the canonical distribution). For visualization of image features, the results were projected back into the original image space. All stimuli in the physiological analysis of Fig. 3c were preprocessed in the same way as the natural images used in training.

Model parameter estimation. We estimated the optimal model parameters $\theta = \{\mathbf{b}_k, w_{jk}\}$ by maximizing the likelihood of the data under the model

$$P(\mathbf{x}|\theta) = \int P(\mathbf{x}|\mathbf{y}, \theta)P(\mathbf{y})d\mathbf{y} \quad (3)$$

The conditional distribution $P(\mathbf{x}|\mathbf{y}, \theta)$ is a multivariate Gaussian that captures correlations in local image regions (equation (1)). Neural activities were assumed to be sparse[31] and independent, and were modelled with a Laplacian (symmetric exponential) distribution, $P(\mathbf{y}) = \Pi P(y_j) \propto \Pi e^{-|y_j|}$. The integral over all possible neural states in equation (3) is intractable and was replaced by a single evaluation at the maximum a posteriori value $\hat{\mathbf{y}} = \arg\max_{\mathbf{y}} P(\mathbf{y}|\mathbf{x}, \theta)$. Although this approximation ignores the volume around the maximum, it is one standard approach to tackling this problem.

We assumed the training patches were sampled independently and that the likelihood for the data ensemble was a product of terms for individual images (equation (3)). In practice, we maximized the log-likelihood using gradient ascent on batches of 100 image patches. Repeated training runs produced convergence to similar parameter values.

Model responses to grating stimuli. The orientation tuning of model neurons in Fig. 3c was measured using 20×20 patches of sinusoidal gratings at different positions, orientations, spatial frequencies and phases. We first eliminated neurons that were 'unresponsive' to gratings, that is, those whose maximal response did not reach 2 standard deviations of the population response to gratings. This was necessary to discount small random activation of neurons tuned for other types of image structures. For each neuron we found the grating with the maximal response and measured modulation in response to varying orientation, phase, or the addition of masks in the receptive field or the surround. Because neural activity could be positive or negative, the full amplitude of modulation was considered as twice the maximum absolute value of the response.

A neuron was considered to be orientation-tuned if its response was modulated by more than 50% over the range of stimulus orientations, and to be phase invariant if the response varied less than 50% over phase-shifted gratings. Cross-orientation inhibition and surround suppression corresponded to greater than 25% decrease in neural response. Bandwidth of orientation tuning was computed as the width at $1/\sqrt{2}$ of the full amplitude of the response modulation.

The projection of neural activity in Fig. 4 was computed using linear discriminant analysis, a technique that finds the linear projections that best separate different classes of data. Applied to the raw pixel data or to the outputs of linear features (data shown in Fig. 1), this method failed to separate the clusters.

30. van Hateren, J. H. Processing of natural time series of intensities by the visual system of the blowfly. *Vision Res.* **37,** 3407–3416 (1997).
31. Olshausen, B. A. & Field, D. J. Sparse coding of sensory inputs. *Curr. Opin. Neurobiol.* **14,** 481–487 (2004).

A Hierarchical Bayesian Model for Learning Nonlinear Statistical Regularities in Nonstationary Natural Signals

Yan Karklin
yan+@cs.cmu.edu
Michael S. Lewicki
lewicki@cnbc.cmu.edu
Computer Science Department and Center for the Neural Basis of Cognition, Carnegie Mellon University, Pittsburgh, PA 15213, U.S.A.

Capturing statistical regularities in complex, high-dimensional data is an important problem in machine learning and signal processing. Models such as principal component analysis (PCA) and independent component analysis (ICA) make few assumptions about the structure in the data and have good scaling properties, but they are limited to representing linear statistical regularities and assume that the distribution of the data is stationary. For many natural, complex signals, the latent variables often exhibit residual dependencies as well as nonstationary statistics. Here we present a hierarchical Bayesian model that is able to capture higher-order nonlinear structure and represent nonstationary data distributions. The model is a generalization of ICA in which the basis function coefficients are no longer assumed to be independent; instead, the dependencies in their magnitudes are captured by a set of density components. Each density component describes a common pattern of deviation from the marginal density of the pattern ensemble; in different combinations, they can describe nonstationary distributions. Adapting the model to image or audio data yields a nonlinear, distributed code for higher-order statistical regularities that reflect more abstract, invariant properties of the signal.

1 Introduction

The goal of many algorithms in machine learning, signal processing, and computational perception is to discover and process intrinsic structures in the data. Extracting these from real signals is a difficult problem, because often the relationships among the observable variables are complex, and there is little a priori knowledge about the types of structures that exist. When some a priori knowledge is available, specialized algorithms can be designed, but this approach is generally less desirable, as it places restrictions on the type of structure that can be learned. Another difficulty is that the dimensionality of the data is often very high, and properties of inter-

est lie in a relatively low-dimensional subspace. Because of the inherent variability of most real-world signals, intrinsic regularities are statistical in nature, which makes them that much more difficult to learn.

One approach to learning statistical regularities is to formulate a probabilistic model of how the data are generated and adapt its parameters to fit the observed distribution. The adapted parameters reflect the statistics of the data ensemble, while internal representations encode individual data patterns. These models make minimal assumptions about the data and can result in more general representations than those in algorithms tailored for specific tasks or types of data.

There are several ways in which data patterns are represented in probabilistic generative models. Distributed representations of linear componential models, such as those for principal component analysis (PCA) and independent component analysis (ICA), are particularly useful for modeling complex high-dimensional data because they can capture independent regularities with independent internal parameters (Bell & Sejnowski, 1995). This makes it possible to model a continuum of different statistical relationships and allows scaling of the algorithms to large numbers of dimensions. Current models, however, are limited in the type of structure they can represent; in order to understand these limitations, it is helpful to look at their mathematical formulation.

Linear componential models achieve a distributed representation by describing the data as a combination of linear basis functions (for a review, see Hyvärinen, Karhunen, & Oja, 2001; Cichocki & Amari, 2002). This yields a probabilistic generative model in which the data ($\mathbf{x}$) are generated as a linear combination of basis functions ($\mathbf{A}$) weighted by coefficients ($\mathbf{u}$),

$$\mathbf{x} = \mathbf{A}\mathbf{u}. \tag{1.1}$$

The likelihood of the observed data under this model is

$$p(\mathbf{x}) = p(\mathbf{u})/|\det(\mathbf{A})| \tag{1.2}$$

(Pearlmutter & Parra, 1996; Cardoso, 1997), and the basis function matrix $\mathbf{A}$ is adapted to maximize the data likelihood. The coefficients $\mathbf{u}$ are the unknown (latent) variables. They are assumed to be independent and identically distributed (i.i.d.),

$$p(\mathbf{u}) = \prod_i p(u_i). \tag{1.3}$$

The priors $p(u_i)$ are typically chosen to be fixed sparse distributions (although parameters of the prior may be adjusted to maximize data likelihood). Because basis function coefficients are assumed to be i.i.d., the dependence among the data is represented solely by the learned matrix of basis functions.

The obvious limitation of this model is that its inherent linearity restricts the type of structure it can capture. Even simple, low-dimensional data often exhibit statistical dependencies that cannot be captured by linear transformations. In many applications, data are complex and rich with statistical structure, and latent variables of linear models adapted to these data exhibit significant residual mutual dependence (Hyvärinen & Hoyer, 2000; Schwartz & Simoncelli, 2001; Karklin & Lewicki, 2003).

Another shortcoming of these models is that they assume that the statistical regularities in the data do not change; they describe stationary probability distributions. For example, once model parameters are adapted in ICA, both the prior and the basis functions are fixed, leading to a stationary distribution over the data. This does not depend on the form of the prior and also applies to models with adaptive or entirely nonparametric priors. In many domains, however, the statistics of the data are known to change, as the physical properties of the environment or conditions for data acquisition vary. While the stationary prior assumption gives a valid approximation of true density over a large enough corpus of training data, it does not reflect the variation across contexts that is observed in many signals.

Figure 1 illustrates nonstationary statistics observed in images of natural scenes. ICA basis functions were adapted to 20×20 patches taken from an ensemble of natural images. Over the full ensemble of the training data, the basis function coefficients have marginal distributions that are consistent with the prior assumed by the model (not shown). However, computing coefficient histograms over particular image regions reveals systematic deviations from the (globally valid) stationary distribution. Patterns in the histograms suggest that basis functions of certain orientations are more active in some parts of the image (e.g., textured, oriented surface of the log), while in other regions, different subsets tend to be activated. This is observed in other types of data as well: temporal basis functions adapted to speech also yield coefficients whose statistics vary greatly across local regions of the signal (see Figure 2).

Figures 1 and 2 give just a few examples of patterns in latent variable distributions that depend on the local context. In fact, there is a wide range of statistical regularities in complex data, a continuum of contexts that is as multidimensional as the physical properties of the environment that give rise to it. Local representations, as employed by clustering or mixture model techniques, assume that the contexts are discrete and thus cannot describe regularities that arise from a combination of different contexts. A model that captures this variation must form flexible, distributed representations of higher-order structure. Moreover, because the dimensions of the contexts are not known a priori, the model must be able to automatically discover this underlying structure. Finally, many previous models of nonstationary distributions have relied on the assumption that data statistics vary smoothly from sample to sample (Everson & Roberts, 1999; Pham & Cardoso, 2001) and computed local estimates of context-dependent variation.

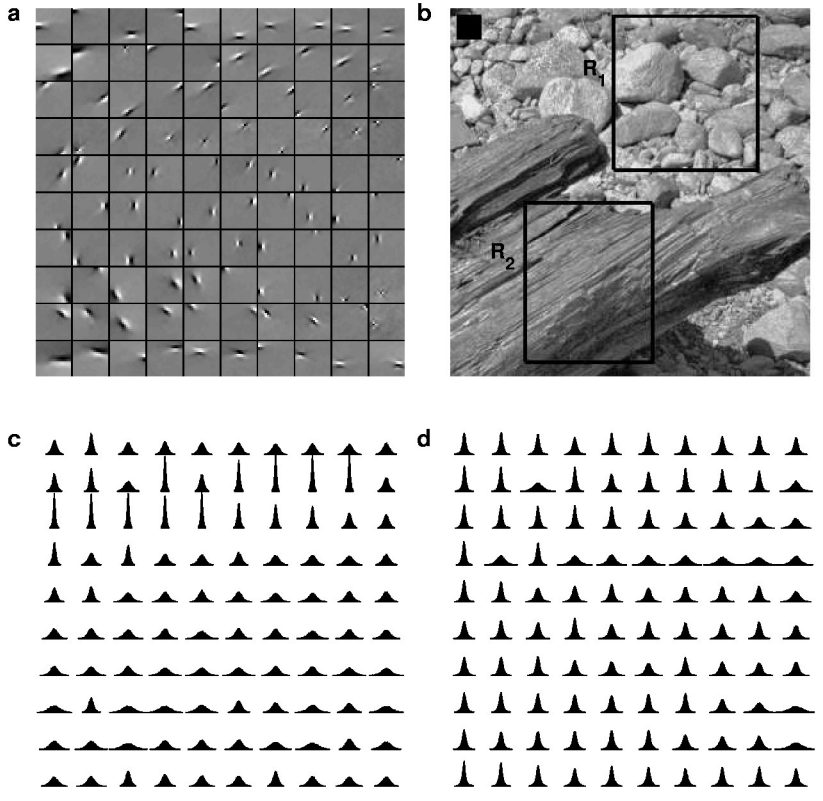


Figure 1: The distribution of ICA basis function coefficients exhibits nonstationary statistics that reflect local image structure. (a) A subset of image basis functions learned from an ensemble of natural images, ordered by orientation. The small black square on the image indicates the size, relative to the image, of the learned basis functions. (b) Coefficients of independent components were computed over two regions of an image. (c, d) Histograms of the coefficients for the two regions reveal patterns in the joint distributions. Each histogram in the 10×10 grid in *c* and *d* corresponds to a basis function at the same grid position in *a* and is normalized so that the filled area sums to 1. Different types of local image structure produce different patterns in the joint activities. For example, the image region containing the log yields higher coefficient variation for basis functions oriented along the grain and matching the approximate spatial frequency of the wood texture.

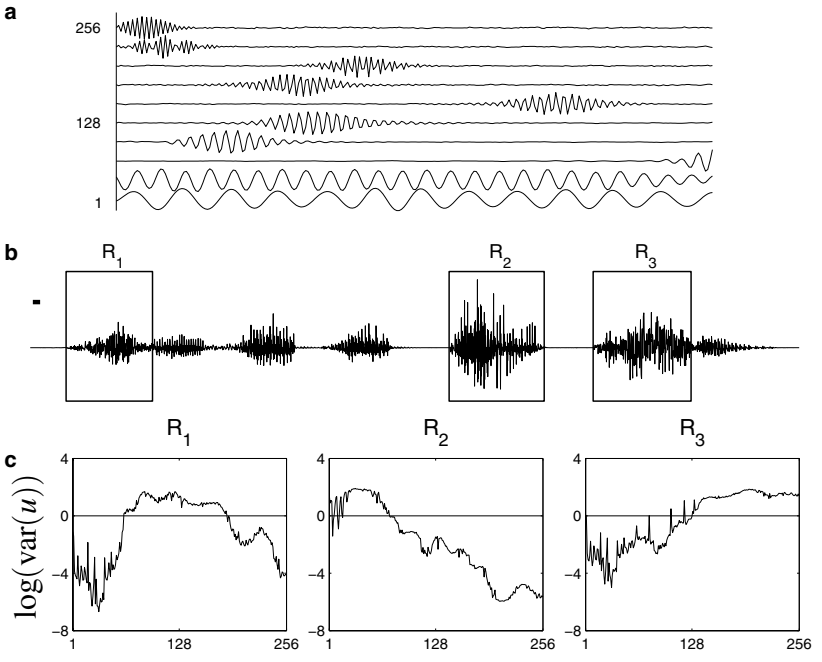


Figure 2: ICA basis functions adapted to speech data also exhibit nonstationary statistical dependencies. (a) A subset of 256 ICA-derived basis functions ordered by dominant frequency. (b) Each basis function was convolved with three different regions of a speech signal. The length of the basis function is indicated by the short bar above the start of the speech signal. (c) The variances of coefficients sampled over the three regions, with the 256 coefficients ordered by frequency as in *a*. Although all basis function coefficients have unit variance when sampled over the whole data ensemble, local regions show characteristic variance patterns that reflect local signal structure.

This assumption does not always hold; even spatially and temporally coherent data exhibit abrupt changes that cannot be modeled as slowly evolving processes.

Here we address the limitations of previous models with a hierarchical Bayesian model that forms a distributed code of higher-order statistical regularities and captures nonstationarities in the data distribution. The model is a generalization of ICA; thus, we begin with a standard linear componential model in which the data are generated as a combination of linear basis functions. However, instead of assuming that the basis function coefficients are independent (and their joint prior distribution is factorable; see equation 1.3), we explicitly model the dependence among hyperparameters of their priors. In order to capture variable, context-dependent activation of

basis functions, the dependence is specified through the scale parameters governing the width of the prior (and hence the variance of the coefficients). This dependence is modeled with a set of density components, a distributed code that describes the shape of the joint density of the linear coefficients and captures patterns in the variances of the coefficients as observed in the motivating examples.

Each density component describes a common underlying deviation from the standard assumption of independence (the i.i.d. joint prior) associated with a frequently encountered context. Using a weighted combination of density components, the model is able to represent a continuum of context-dependent changes in probability distributions. Adapting the set of density components and modeling their activation with a sparse prior yields a compact description of higher-order statistical regularities of the data ensemble. Unlike other recent methods, the model makes no assumptions of temporal or spatial coherence; it is able to infer, independently for each data sample, the higher-order code that describes the generating distribution.

Below we present the probabilistic framework for the model and describe the associated learning algorithms. Previously, we have used this model to discover higher-order structure in natural images (Karklin & Lewicki, 2003). Here, we describe the algorithm in more detail and frame it as a general method of statistical density estimation for high-dimensional nonstationary data. We verify the recovery of correct model parameters using a toy data set, apply the learning algorithm to a wider range of data types, and show how the learned higher-order code accounts for observed dependencies. We provide results and analysis for photographs of natural scenes, scanned images of newspapers, and speech waveforms. However, the model is not tailored specifically to images or audio data, and can be used to automatically learn the nonlinear statistical dependencies in any data set with sufficiently rich structure.

2 A Hierarchical Model for Nonstationary Distributions

Our model is a generalization of previous linear models. Hence, we begin by assuming that each data vector is generated as a combination of linear basis functions, $\mathbf{x} = \mathbf{Au}$. As in standard ICA models (e.g., Cichocki & Amari, 2002), basis function coefficients are assumed to be sparsely distributed. Here we use a generalized gaussian distribution with zero mean:

$$p(u_i) = \mathcal{N}(0, \lambda_i, q_i) \tag{2.1}$$

$$= z_i \exp\left(-\left|\frac{u_i}{\lambda_i}\right|^{q_i}\right), \tag{2.2}$$

where $z_i = q_i/(2\lambda_i\Gamma[1/q_i])$ is a normalizing constant. The parameter q_i determines the weight of the distribution's tails and can be estimated from the data; in many ICA applications, the coefficients tend to be sparse, making

their distributions supergaussian ($q_i < 2$). Typically, the scale parameter λ_i is fixed to a constant, since the basis functions in $\mathbf{A}$ can themselves scale to fit the data.

In order to capture residual dependence among coefficients $\mathbf{u}$, we must abandon the assumption of fixed, independent priors. The motivating examples suggested that intrinsic structures in the data give rise to patterns in the scales of the coefficients (similar dependencies have been observed previously in wavelet coefficients; Simoncelli, 1997). A natural way to model this is through the scale parameters of the prior, which we model as a nonlinear transformation of latent higher-order variables. Specifically, we use a matrix of density components $\mathbf{B}$ and density component coefficients $\mathbf{v}$ to describe the logarithm of the scale parameter,

$$\log(\boldsymbol{\lambda}/c) = \mathbf{B}\mathbf{v}. \tag{2.3}$$

If we define the constant $c = \sqrt{\Gamma(1/q)/\Gamma(3/q)}$, the variance of the coefficients becomes 1 when the right side of the equation is 0 (this becomes convenient when a zero-centered prior is selected for the distribution of $\mathbf{v}$; see below).

The joint prior distribution of coefficients $\mathbf{u}$ can now be expressed as

$$-\log p(\mathbf{u}|\mathbf{B}, \mathbf{v}) \propto \sum_i [\mathbf{B}\mathbf{v}]_i + \left| \frac{u_i}{c \exp([\mathbf{B}\mathbf{v}]_i)} \right|^{q_i}, \tag{2.4}$$

where $[\mathbf{B}\mathbf{v}]_i$ represents the ith element of the vector $\mathbf{B}\mathbf{v}$ (see the appendix for the derivation).

Basis function coefficients are assumed to be independent conditional on the higher-order variables, $p(\mathbf{u}|\mathbf{v}) = \prod p(u_i|\mathbf{v})$. This accounts for the dependence in the magnitudes of basis function coefficients. The new form of the prior (2.4) implies that if $\mathbf{v}$ is 0, the model reduces to standard ICA in which the linear coefficients are independent and identically distributed with variance equal to 1. Nonzero values of $\mathbf{v}$ scale and combine density components (columns of $\mathbf{B}$) that define patterns in the distributions of $\mathbf{u}$. Because each v_i can be positive or negative, each density component represents contrast in the magnitudes of coefficients $\mathbf{u}$ (see Figure 3).

We place a nongaussian, sparse prior on the latent variables $\mathbf{v}$ and infer their values for each data sample.[1] This means that a priori, we assume that the activity of density component coefficients is sparse, and relatively few components are needed to describe how the generating distribution associated with each data sample differs from the i.i.d. ICA model. Using this parameterization, we adapt the density components to the entire data ensemble, which produces a compact description of higher-order statistical regularities.

[1] A Laplacian prior was used in the simulations, but other distributions may be more appropriate.

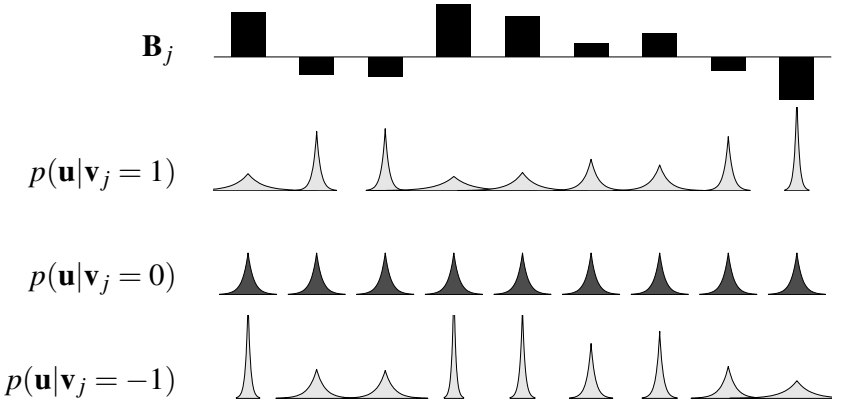


Figure 3: Each density component defines a pattern in the joint distribution $p(\mathbf{u})$. The plot at the top shows an example nine-dimensional density component $\mathbf{B}_j$. The distributions of coefficients $u_{1,\ldots,9}$ are shown for different values of v_j. Here we show only a single density component $\mathbf{B}_j$, whereas the model adapts a set of them $\mathbf{B} = \{\mathbf{B}_1, \mathbf{B}_2, \ldots, \mathbf{B}_M\}$ to obtain a compact description for common scale patterns in the data.

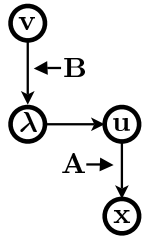
The full generative model is shown in graphical form in Figure 4. There are two sets of random variables that give rise to the data, $\mathbf{v}$ and $\mathbf{u}$, and two sets of parameters adapted to the data: the linear basis functions $\mathbf{A}$ and the density components $\mathbf{B}$. A crucial difference between this generative form and several other models that account for higher-order dependence is that here, the density components specify a distribution over the coefficients, as opposed to exact values or pooled magnitudes, which have been used in other models (Hoyer & Hyvärinen, 2002; Welling, Hinton, & Osindero, 2003). Thus, the model forms a hierarchical representation in which the lower-level codes data values precisely and the higher level represents more abstract properties associated with the shape of the data distribution.

3 Inference of Density Component Coefficients

For each data sample, it is necessary to compute the higher-order representation $\mathbf{v}$ that best describes the pattern in the scale of coefficients $\mathbf{u}$. This transformation is nonlinear and cannot be expressed in closed form. Here, we compute the best value of $\mathbf{v}$ by maximizing the posterior distribution,

$$\hat{\mathbf{v}} = \arg\max_{\mathbf{v}} p(\mathbf{v}|\mathbf{u}, \mathbf{B}), \tag{3.1}$$

$$= \arg\max_{\mathbf{v}} p(\mathbf{u}|\mathbf{B}, \mathbf{v})p(\mathbf{v}). \tag{3.2}$$



$$
\begin{aligned}
v_i &\sim \mathcal{N}(0, 1, q_i) \\
\lambda_j &= c \exp^{[\mathbf{Bv}]_j} \\
u_j|\lambda_j &\sim \mathcal{N}(0, \lambda_j, q_j) \\
x_k &= \sum_j \mathbf{A}_{kj} u_j
\end{aligned}
$$

Figure 4: Schematic of the hierarchical generative model. Sparsely distributed random variables $\mathbf{v}$ specify (through a nonlinear transformation) the scale hyperparameters $\boldsymbol{\lambda}$ for the distribution of coefficients $\mathbf{u}$. The data $\mathbf{x}$ are a linear combination of coefficients $\mathbf{u}$. Matrices $\mathbf{A}$ and $\mathbf{B}$ are parameters that are adapted to the statistical distribution of the data.

We assume that v_i's are independent ($p(\mathbf{v}) = \prod_i p(v_i)$) and sparsely distributed ($\log p(v_i) \propto -|v_i|$). For the simulations below, $\hat{\mathbf{v}}$ was derived by gradient ascent. We used second-order methods (LeCun, Bottou, Orr, & Müller, 1998) to stabilize and speed up convergence to optimal estimates.

Because the prior is zero centered and sparse, only a few nonzero values will contribute to the representation of each data sample. The inference of optimal density component coefficients is analogous to estimating sample variance based on a single observation, but the problem is further constrained by the structure of the learned density components. Because the model is constrained to describe the pattern of variance with a sparse combination of density components, the value of $\mathbf{v}$ for a typical pattern is usually well determined. In addition, the high dimensionality of the input facilitates the inference process, as it provides more directions of variation that make up the variance pattern.

4 Adapting Model Parameters to the Data

The linear basis functions and the density components are adapted to the data ensemble by maximizing the posterior $p(\mathbf{A}, \mathbf{B}|\mathbf{X})$. We assume that samples in the data ensemble $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ are independent, so that

$$p(\mathbf{X}|\mathbf{A}, \mathbf{B}) = \prod_{n=1}^{N} p(\mathbf{x}_n|\mathbf{A}, \mathbf{B}). \tag{4.1}$$

For each data sample $\mathbf{x}$, the posterior distribution is

$$
\begin{aligned}
p(\mathbf{A}, \mathbf{B}|\mathbf{x}) &\propto p(\mathbf{x}|\mathbf{A}, \mathbf{B}) p(\mathbf{A}, \mathbf{B}) && (4.2) \\
&= p(\mathbf{u}|\mathbf{B}) p(\mathbf{B}) / |\det(\mathbf{A})|. && (4.3)
\end{aligned}
$$

Ideally, the marginal distribution $p(\mathbf{u}|\mathbf{B})$ would be computed by integrating over $\mathbf{v}$, but evaluating this integral for equation 2.4 is intractable. Here we approximate it using the maximum a posteriori estimate $\hat{\mathbf{v}}$:

$$p(\mathbf{u}|\mathbf{B}) = \int p(\mathbf{u}|\mathbf{B}, \mathbf{v})p(\mathbf{v})d\mathbf{v}, \tag{4.4}$$

$$\approx p(\mathbf{u}|\mathbf{B}, \hat{\mathbf{v}})p(\hat{\mathbf{v}}). \tag{4.5}$$

Substituting this approximation into the posterior gives

$$p(\mathbf{A}, \mathbf{B}|\mathbf{x}) \propto p(\mathbf{u}|\mathbf{B}, \hat{\mathbf{v}})p(\hat{\mathbf{v}})p(\mathbf{B})/|\det \mathbf{A}|. \tag{4.6}$$

The prior on $\mathbf{B}$ places a small a priori bias for small values of $B_{i,j}$ and eliminates the problem of a degenerate case in which $\mathbf{B}$ grows without bounds while $\mathbf{v}$'s rescale to be smaller. For the results here, we assumed $B_{i,j}$ followed a gaussian distribution. The matrices $\mathbf{A}$ and $\mathbf{B}$ can be optimized iteratively by maximizing $p(\mathbf{A}|\mathbf{X}, \mathbf{B})$ and then maximizing $p(\mathbf{B}|\mathbf{X}, \mathbf{A})$. In this case, the first step amounts to performing ICA in which the priors incorporate the scale estimates $\hat{\mathbf{v}}$. Alternatively, we can assume that optimal linear basis functions are largely independent of the set of density components, and optimize $\mathbf{B}$ using a fixed $\mathbf{A}$. For computational efficiency, $\mathbf{A}$ and $\mathbf{B}$ were assumed to be independent and were adapted separately in the simulations described below. We confirmed the validity of this approach by training a model on data of reduced dimensionality and with fewer density components; results were qualitatively similar to optimizing the parameters independently.

In order to verify that the learning algorithm produces a valid solution, we adapted model parameters to an artificial data set for which the optimal solution was known. The data were generated by constructing a set of density components and then sampling basis function coefficients according to $p(\mathbf{u}|\mathbf{B})$. An illustration of the process and the obtained results is shown in Figure 5. Optimizing density components from random initial values produced a matrix that was identical (up to a permutation of its columns) to the true model parameters (see Figures 5a and 5b). The patterns in the learned density components specify nonlinear dependencies among coefficient magnitudes; in fact, there are no linear correlations among basis function coefficients sampled from the model (even when the same $\mathbf{v}$ is used to generate the coefficients). Linear models like ICA are unable to recover these statistical regularities.

As a control, we adapted the density component model to a pure noise data set in which coefficients $\mathbf{u}$ were random samples from independent sparse distributions. In this case, no regularities in the magnitudes of coefficients existed, and the resulting density components consisted of small, random values.

5 Discovering Structure in Complex Data

5.1 Learned Density Components. We optimized model parameters on several data sets and analyzed the learned density components. For computational simplicity, the model was optimized in two stages in all the simulations. First, a complete linear basis **A** was adapted to the data using standard methods; next, the density component matrix **B** was optimized on the coefficients of the fixed **A**. Since the linear basis functions were learned using standard ICA methods, our analysis and discussion here is limited to the recovered matrix of density components. The density components were initialized to small random values, and gradient ascent was performed on stochastically sampled batches of data. The maximum a posteriori estimate $\hat{\mathbf{v}}$ was obtained using 20 steps of gradient ascent. Convergence of the gradient procedures for the optimization of **B** and estimation of $\hat{\mathbf{v}}$ was tested in a number of ways, including varying the step size, the number of iterations, and the initial conditions. The given optimization parameters yielded reasonable speed and accuracy, as well as consistent solutions for different random initial conditions.

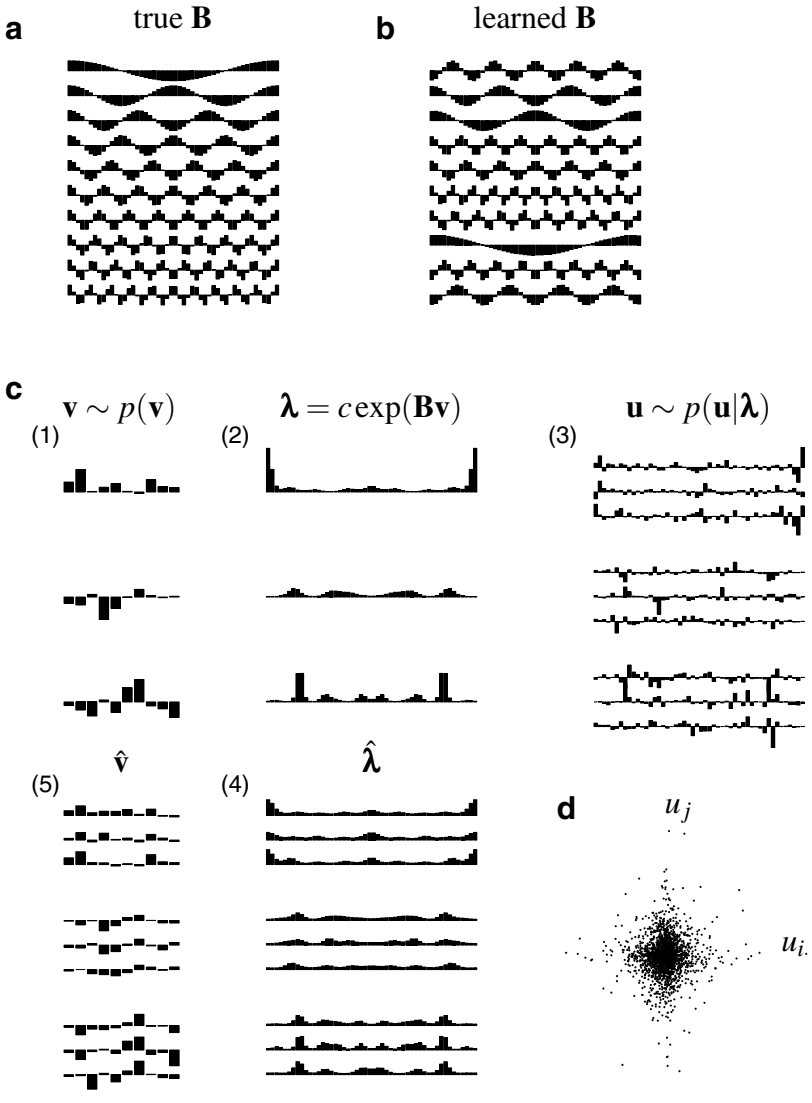
We first applied the learning algorithm to small (20×20) image patches sampled from a standard set of 10 gray-scale images of natural scenes (Olshausen & Field, 1996; Karklin & Lewicki, 2003). We used a complete set of 400 linear basis functions. The number of density components was set to 100 (although the algorithm is able to recover any number that yield a sparse distribution for coefficients **v**). We used batches of 1000 samples for 35,000 iterations of gradient ascent with a fixed step size of 0.3.

Statistical regularities of the data ensemble are captured in the matrix of density components. In order to analyze the structure described by this matrix, we need to examine its weights as they relate to the basis functions whose distributions they affect. (Recall that each weight in a density component vector specifies how a particular $p(u_i)$ is rescaled). The initial ordering of basis functions in the learned matrix **A** is arbitrary; hence, weights in **B** also appear random in their original ordering. However, we can rearrange the weights in **B** according to some property of the linear basis functions and examine whether the learned density components capture structure related to the chosen property. For example, ICA basis functions adapted to natural images are spatially localized; arranging density component weights according to the location of corresponding basis functions within the image patch reveals patterns in their organization (see Figure 6). Thus, density components that appear structured in this arrangement specify dependence among spatially related linear basis functions. As parameters in the generative model, they describe common data distributions that reflect localized image structure. Some density components also appear random when arranged spatially, but these often show organization along other dimensions of the lower-order representation, such as orientation or spatial frequency (Karklin & Lewicki, 2003). Changing the number of density components

does not affect the type of structure captured by the hierarchical model. A larger number of density components allows the model to represent finer-scale spatial regularities, as well as other statistical structure that is not as obvious to interpret.

We also applied the model to speech data from the TIMIT database. Linear basis functions were adapted to bandpass filtered speech segments of 256 samples (16 msec of 16 kHz sound). The number of density components was set to 100, and the parameters were optimized using stochastic learning on data batches of 1000 for 10,000 iterations. A representative set of the learned density components is shown in Figure 7. In order to display the weights in the density components as they relate to the linear code, we first computed the Wigner distributions (WD) of the linear basis functions using the DiscreteTFDs Matlab package (O'Neill, 1999). The Wigner distribution of a basis function is a surface in the time-frequency space; we took a contour at 95% peak value for each basis function and drew all these contours on a single time-frequency plot (time on the horizontal axis, 0 to 16 msec, and frequency on the vertical axis, 0 to 8 kHz). Because the linear basis functions adapted to speech tile most of the the time-frequency space, the contours also exhibit relatively even tiling of the plots. In Figure 7, nine WD plots show the weights in nine density components to the same set of linear basis functions. Here, as in image density components, the shading of each patch corresponds to the value of the weight. Some density components

Figure 5: *Facing page*. The model correctly recovers the density components used to generate synthetic data. We constructed a 50×10 matrix $\mathbf{B}$ composed of 10 cosine-shaped density components (a). After 3000 iterations, the model recovers (up to a permutation) the correct density components (b). (c) The generative and inference steps of the algorithm. (1) Three 10-dimensional density component coefficients are drawn from a sparse distribution; (2) each $\mathbf{v}^{(i)}$ specifies a vector of scaling variables $\boldsymbol{\lambda}^{(i)}$ through the nonlinear transformation $\boldsymbol{\lambda}^{(i)} = c \exp[\mathbf{Bv}]^{(i)}$. (3) The scaling variables are hyperparameters for nonstationary distributions $p(\mathbf{u})$, from which data samples $\mathbf{u}$ are drawn. In order to emphasize that each vector of scaling variables $\boldsymbol{\lambda}^{(i)}$ specifies a distribution, not fixed values of $\mathbf{u}$, we plotted several $\mathbf{u}$'s drawn from the distribution $p(\mathbf{u}|\boldsymbol{\lambda}^{(i)})$. In actual simulation, each data point was generated independently. Using the learned density components, estimates of (4) $\hat{\mathbf{v}}$ and (5) $\hat{\boldsymbol{\lambda}}$ were obtained for each data sample. Because the inference problem involves the estimation of density parameters from single data points, $\hat{\mathbf{v}}$ and $\hat{\boldsymbol{\lambda}}$ only approximately match true parameters. Although the complete hierarchical model includes another transformation $\mathbf{x} = \mathbf{Au}$, the projection to data space $\mathbf{x}$ is linear and is not necessary for inference of $\hat{\mathbf{v}}$ when coefficients $\mathbf{u}$ are known. The scatter plot of 1000 samples of u_1 and u_2 drawn from the model (d) shows that there is no linear dependence among basis function coefficients.


a
true B
b
learned B
c
$\mathbf{v} \sim p(\mathbf{v})$
(1)
$\boldsymbol{\lambda} = c\exp(\mathbf{B}\mathbf{v})$
(2)
$\mathbf{u} \sim p(\mathbf{u}|\boldsymbol{\lambda})$
(3)
$\hat{\mathbf{v}}$
(5)
$\hat{\boldsymbol{\lambda}}$
(4)
d
u_j
u_i

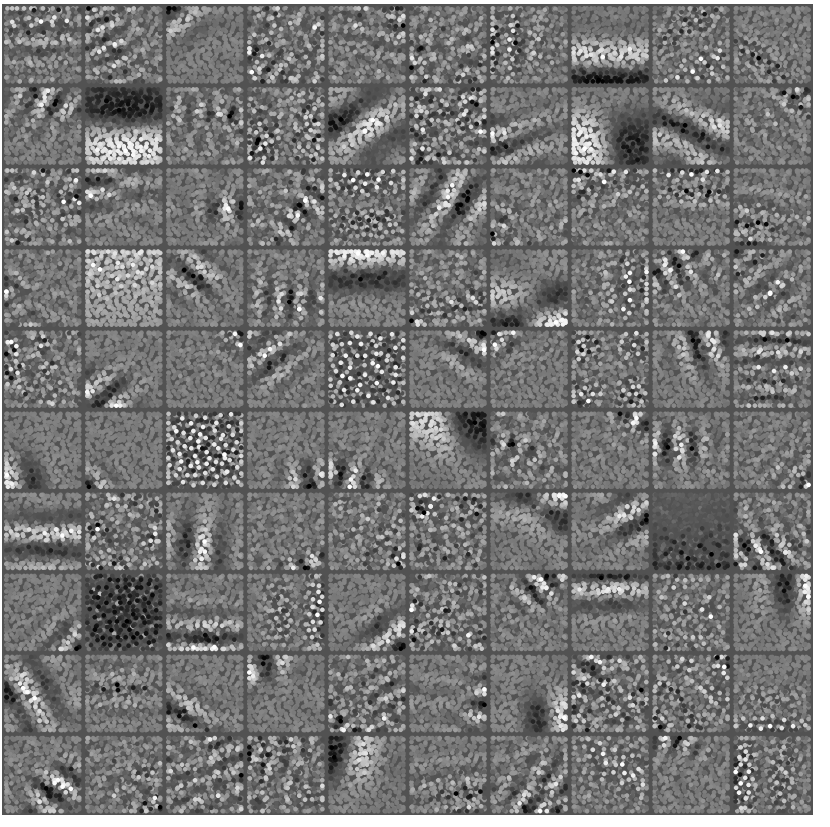


Figure 6: Density components optimized on an ensemble of 20×20 image patches drawn from natural scenes. Each column of $\mathbf{B}$ is represented here as a square; its weights to 400 image basis functions are plotted as dots, placed in locations corresponding to the center of each image basis function in the image patch. Each dot is colored according to the value of the weight, with white indicating positive weights, black negative weights, and gray weights that are close to zero. Most density components describe spatial relationships and capture coactivation of linear basis functions localized to a particular area of the image patch. For example, the density component in the second row, second column indicates whether contrast in the image patch is localized to the top or the bottom half. While most density components represent location, orientation, or spatial frequency regularities, the organization of some is not obvious.

describe coactivation of linear basis functions of adjacent frequency bands, while others are localized in time within the sample window. Most density components capture periodic higher-order structure and regularities across

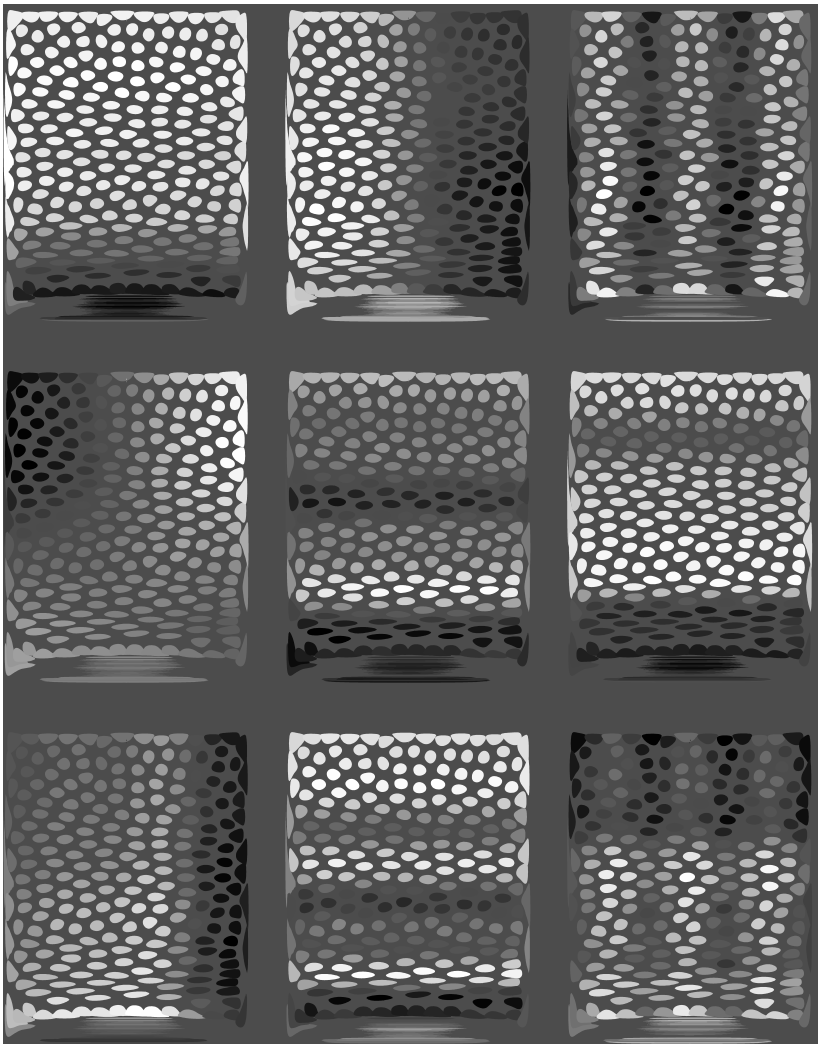


Figure 7: A subset of density components of speech. The weights in a column of **B** are plotted as shaded patches in one of the nine panels. Each patch is placed according to the temporal and frequency distribution of the associated linear basis function and shaded according to the value of the weight, with white indicating positive weights, black negative weights, and gray weights that are close to zero. The axes represent time, 0 to 16 msec, horizontally, and frequency, 0 to 8 kHz, vertically. The density components form a distributed representation of the frequency of the signal and the location of energy within the sample window. Density components coding for multiple frequencies might capture harmonic regularities in the speech signal (see the text for details).

multiple frequencies or time intervals, and a few are tuned specifically to subtle shifts in dominant frequency over the sample window.

5.2 Higher-Order Code. In order to better understand the type of structure captured by the model, it is informative to look at the higher-order code—the coefficients of density components—and the statistical regularities it represents. Individual density component coefficients indicate the presence, in each data sample, of the type of structure represented in Figure 6. As a distributed code, their joint activity describes the data density whose shape reflects underlying structure in the data.

Figure 6 shows that among other statistical regularities, the higher-order code captures spatial relationships in the data. How does this representation compare to the lower-level, linear code for image structure? The activity of density component coefficients over contiguous regions of the data suggests that the higher-order representation captures more abstract properties of the data (see Figure 8). When a sliding window is applied to a natural scene image, the resulting lower-level representation changes rapidly from sample to sample, as would be expected from what are essentially outputs of linear filters. The higher-order representation varies more slowly over the image and captures more invariant properties of the data, such as overall image contrast or the dominance of certain spatial frequencies. Also shown in Figure 8 are the values of the linear and the density component coefficients for a model trained on images of newspaper text. Here too the density component coefficients describe more abstract properties: several combine to form a distributed representation of text line position in the image patch (the activity of one such coefficient is shown in the first panel of Figure 8f), while others represent commonly observed structures in the data, such as recurring shapes of letters or blank spaces between words.

Applied to audio data, the model also captures more abstract properties of the stimulus. In Figure 9a, we plot an example audio signal, along with the activities of three linear coefficients in Figure 9b and three density component coefficients in Figure 9c. We emphasize that, as for the images, the model is trained on segments drawn randomly from the data set, and the values of the coefficients for each sample position in the signal shown in the figure are determined independently. The higher-order representation varies more slowly than responses of the linear filters and captures structural elements that extend well beyond the small sampling window. This may reflect a general property of natural signals—fast fluctuations in their exact values are caused by interactions of underlying physical properties, which themselves change more slowly.

5.3 Modeling Residual Dependencies. The motivating examples (see Figures 1 and 2) showed specific types of residual dependencies among the "independent" linear coefficients, such as the dependence among the scale of coefficients, which formed patterns that changed from context to context.

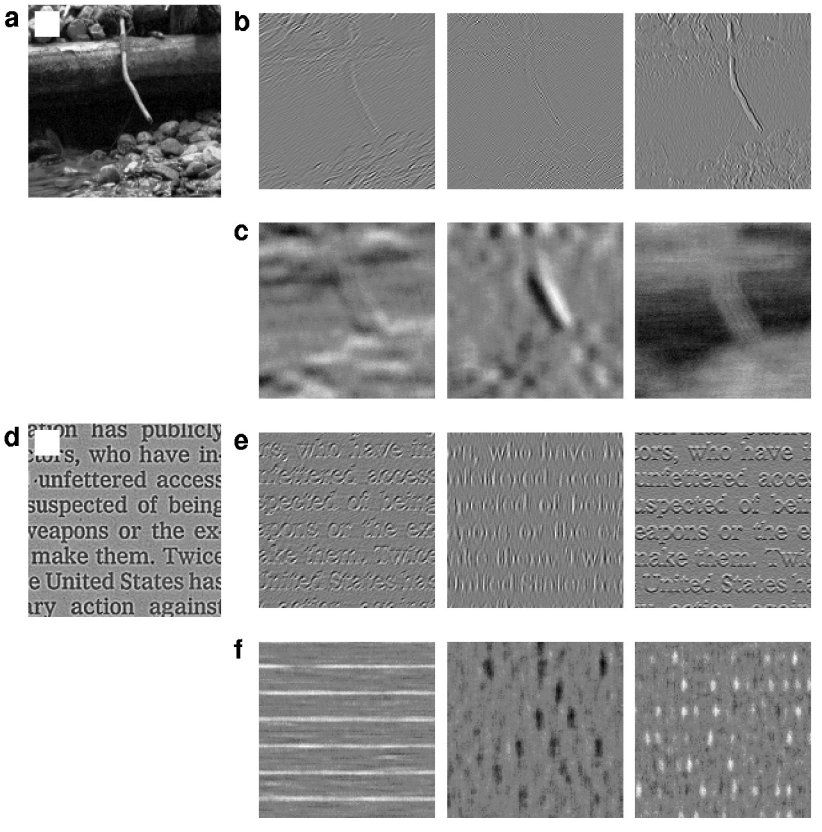


Figure 8: The higher-order code captures more abstract properties of image data and therefore forms a more invariant representation than the coefficients of linear basis functions. We trained the model on natural images (a–c) and scanned newspaper clippings (d–f) and analyzed the representation formed by the model as it varied over the images. A sliding window (represented as white squares in the images) was applied over contiguous sections of the training data (a,d), and values of three linear coefficients u_i (b,e) and three higher-order coefficients v_j (c,f) were plotted as they varied over the signal. White represents large pos values, black large negative values, and gray zeros. Although the model is trained on image patches selected randomly from the data set, the higher-order code forms a representation that changes more slowly over space and captures properties of the data that extend beyond the sampling window, such as the overall contrast in natural images or the position of the text-line in newspaper images.

The adapted hierarchical density component model is able to capture these dependencies. First, drawing from the model generates data with similar

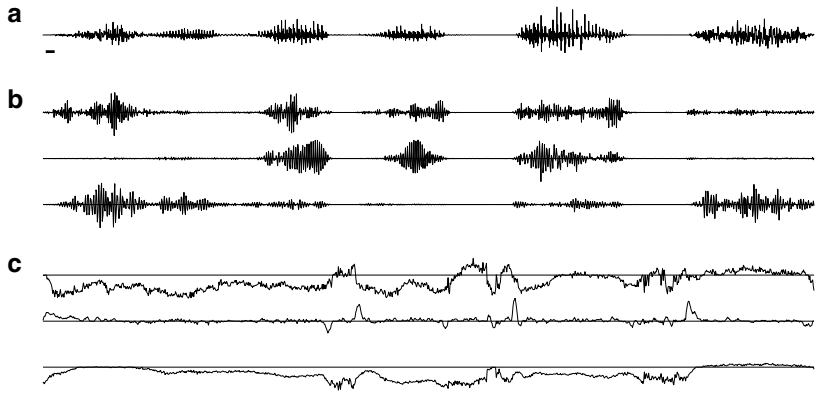


Figure 9: The higher-order representation formed by the hierarchical model trained on speech data is more invariant than simple outputs of linear filters. A sliding window was applied to a speech signal (a; size of window indicated by a short bar). At each point, the linear basis function coefficients **u** were computed (b) and the higher-order coefficients **v** were inferred (c). Values of **v** change slowly and represent more abstract properties, such as the presence of silence or the onset of vocalization. Only three examples for **u** and **v** are shown.

statistical regularities. Furthermore, the higher-order representation in the model defines an implicit normalization of the linear code, and the residual dependencies are no longer observed in the normalized code.

Figure 10a shows the empirical joint distributions (top row) of two linear coefficients when sampled from the image regions R_1 or R_2 of Figure 1. In the two contexts, the shape of the distribution is different: the coefficients have high variance in one context but not in the other. The statistical properties in the two contexts are captured by the inferred density component coefficients. Fixing the density component coefficient to the empirical distributions and sampling the linear coefficients reveals the same type of statistical structure (middle row). At the same time, it is possible to use the estimated parameters of the generating distribution to normalize the data. Dividing the linear coefficients by the estimated scale parameters $\hat{\boldsymbol{\lambda}}$ results in joint distributions that are symmetric with uniform variance across different contexts and image regions (bottom row).

Another way to observe dependence among coefficient magnitudes is to draw a conditional histogram that plots distributions of one coefficient conditional on different values of another (Simoncelli, 1997; Schwartz & Simoncelli, 2001). While the joint histograms show that coefficient magnitudes are dependent on the sampling context, conditional histograms reveal pair-wise dependencies between coefficients across all contexts. For natural images, most linear coefficients show a positive magnitude dependence; the

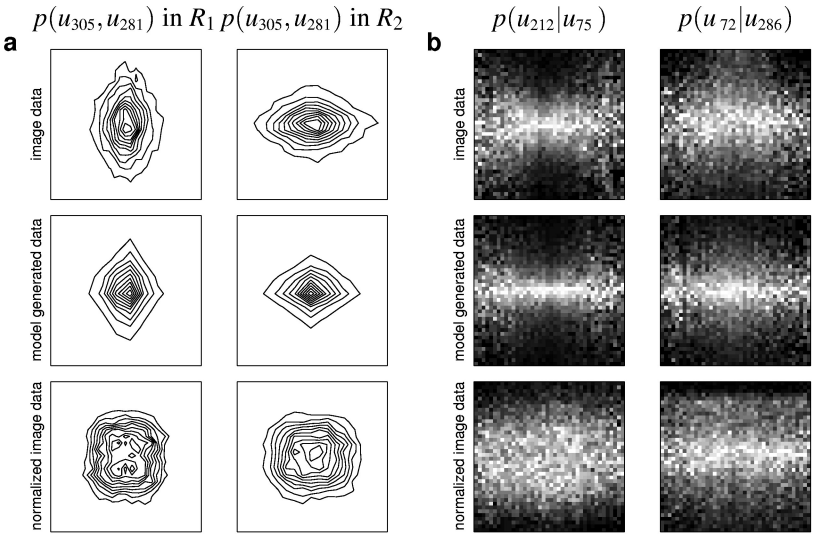


Figure 10: Dependence in the magnitudes of linear basis function coefficients is captured by the density component model. (a) The joint distributions of linear coefficients are different in the two image regions from Figure 1, that is, the data distribution is not stationary. Sampling from the model under the estimated higher-order representation of each context results in similar distributions. Normalizing the image data by the estimated scale parameters, $\bar{u}_i = u_i/\lambda_i$, eliminates the non-stationarity. (b) Over the full data ensemble, empirical conditional histograms for pairs of coefficients show statistical dependencies in the magnitude. Sampling from the model adapted to this data ensemble produces similar dependencies, and normalizing by the estimated scale parameters removes the magnitude correlations. See the text for more details.

magnitude of one coefficient is positively correlated with the magnitude of another, (e.g., the left pair in Figure 10b), but some exhibit the reverse pattern. Sampling from the model produces data with the same statistical dependencies (see Figure 10b, middle row), while normalized linear coefficients show no conditional magnitude dependence (see Figure 10b, bottom row).

Joint and conditional histograms illustrate pair-wise structure in the linear coefficients; global patterns in coefficients, such as those observed in Figures 1 and 2, are also captured by the model. In the top row of Figure 11, we replot the statistics from Figure 2 that show variance patterns in different regions of the speech signal. In the bottom row, we plot the same statistics for the coefficients normalized by the estimated scale parameters; after nor-

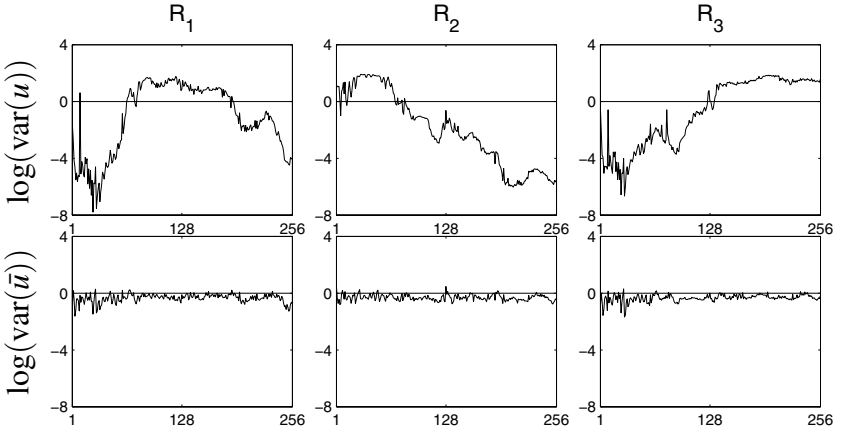


Figure 11: The model accounts for nonstationary statistics of coefficients. (Top row) Log variance of **u** for the three regions in the speech signal from Figure 2b. Each plot shows the log variance of 256 basis functions, sorted by dominant frequency (replotted from Figure 2c). (Bottom row) Log variance of the normalized basis function coefficients $\bar{u}_i = u_i/\hat{\lambda}_i$.

malization, the statistics are stationary, and the coefficients are identically distributed. The same global normalization effect is observed for natural images (plots not shown).

6 Discussion

Some previous work has focused on extending linear probabilistic models. Mixtures of linear ICA models have been used to describe high-dimensional, nongaussian data drawn from distinct classes (Lee, Lewicki, & Sejnowski, 2000; Lee & Lewicki, 2002). In this approach, the number of classes is specified in advance, and an optimal linear basis is learned for each class. This nonlinear generative model describes different data distributions for different classes, but its higher-order representation is fundamentally local and does not scale well in domains where the variation in higher-order structure is continuous and high-dimensional. A key problem addressed by the model presented here is the presence and interaction of multiple instrinsic structures, and this is achieved by a continuous, distributed higher-order code.

Other models have extended ICA to handle nonstationary data distributions. Everson and Roberts (1999) proposed a model in which ICA basis functions evolve with time as a first-order Markov diffusion process. Similarly, Pham and Cardoso (2001) developed and Choi, Cichocki, and Belouchrani (2002) extended algorithms for non-stationary models in which

the variances of the sources modulate slowly in time. These are also related to models of time-varying mean and variance in economics (Bollerslev, Engle, & Nelson, 1994), and typically model data whose statistics change slowly over time or space. Alternatively, one can describe the variance with a sparse but temporally coherent latent variable (Hyvärinen, Hurri, & Väyrynen, 2003). However, in many cases, real-world data are subject to both smooth and abrupt changes that do not follow diffusion dynamics or smooth amplitude modulation. In contrast to these approaches, the density component model makes no assumptions of temporal or spatial smoothness. It infers an optimal generating distribution for each data sample based on only the values of that sample, though the inference process is constrained by parameters adapted to the statistical regularities of the entire data ensemble. Thus, it is able to capture both smooth and abrupt changes in the underlying structure.

Another approach to capturing intrinsic structures in the data has been to incorporate a specific nonlinearity, such as the sum of squares (Krüger, 1998; Hoyer & Hyvärinen, 2002) or sigmoid functions (Lee, Koehler, & Orglmeister, 1997). The drawback to these models is that the type of structure learned is limited by the specific choice of the nonlinearity. Most of these methods also assume a fixed linear representation (e.g., a set of oriented, localized 2D basis functions for image models), and those that adapt the linear representation assume a more constrained form of the nonlinear dependence (see below). In the model presented here, the linear basis is adapted to the data and maximizes the statistical independence of the linear representation. This ensures that the statistical regularities captured by the higher-order code represent fundamentally nonlinear dependencies rather than residual dependence resulting from the choice of a suboptimal linear basis. Furthermore, in some applications, there is no clear choice of linear representation (such as Gabor filters or wavelets in image processing); in such cases, it is sensible to derive the linear code from the statistics of the data.

Several earlier models have explicitly represented the dependence among coefficients of linear basis functions. In the subspace ICA model (Hyvärinen & Hoyer, 2000), the linear basis functions are grouped into neighborhoods and adapted to maximize the independence of the vector norms of the neighborhoods. Basis functions within a neighborhood are no longer assumed to be independent; in fact, the energies of their coefficients are correlated. In the more generalized form of the model, called topographic ICA, the disjoint sets of dependent basis functions are replaced by a topographic arrangement that defines magnitude dependencies among basis functions (Hyvärinen, Hoyer, & Inki, 2001). The generative forms of subspace ICA and topographic ICA can be interpreted as more constrained versions of the density component model presented here. Neighborhood or topographic dependencies can be equivalently represented by density components whose weights are specified in advance to reflect tree-dependent or topographic relationships. The density component model, however, places no such constraints on the

higher-order representation; thus, density components adapted to the data can capture nontopographic dependencies as well.

A related set of work has attempted to model the dependence among coefficients of a fixed linear transform, such as a multiscale wavelet decomposition. Romberg, Choi, and Baraniuk (2001) used a set of discrete latent variables, propagated along a multiscale wavelet tree, to describe the distribution of each wavelet coefficient. The transition probabilities of the latent states were adapted to match the scale dependencies between adjacent nodes in the tree. Buccigrossi and Simoncelli (1999) computed a linear predictor of scale for each coefficient as a function of the magnitudes of its neighbors. Wainwright, Simoncelli, and Willsky (2001) extended this approach by modeling the wavelet coefficients as observed variables in a gaussian scale mixture, in which random gaussian variables are multiplied by latent scaling variables. Dependence among coefficients adjacent on the wavelet tree is captured through the structure of a gaussian process defined on the scaling variables. In addition to its reliance on a fixed linear representation (the drawbacks of this are outlined above), this model is limited in that it can only describe pairwise dependencies between variables adjacent on the wavelet tree. Adapting a model to learn global statistical regularities, as opposed to local representations of class structure or pairwise dependence, allows it to capture a wider range of intrinsic structures. Also, learning an efficient basis to describe these dependencies facilitates their interpretability and provides a better fit to the underlying structure.

7 Conclusion

We have introduced a hierarchical, generative Bayesian model that can be considered a nonlinear extension to ICA. It uses parametric density estimation to learn statistical regularities from the data and makes no assumptions about the type of structure it expects to find. The model is general, it is not specific to any domain and can be applied to any data set with rich statistical structure. Because the model forms distributed representations at all levels of its hierarchy, it scales well to large-dimensional data.

Adapted to patches from natural images or samples from speech data, the density component model was able to learn nonlinear statistical regularities. It yielded a distributed representation of context, which included higher-order spatial relationships for image data and frequency and harmonic structure for audio data. Sampling from the model produced data with the same statistical regularities observed in the training data sets and the model's implicit normalization of the lower-order code accounted for the residual dependencies observed in various data sets.

Recently, it has been argued that higher-order properties of natural signals change slowly across time or space and that this spatial and temporal coherence can be used to extract higher-order structure from the data (Foldiak, 1991; Kayser, Einhäuser, Dümmer, König, & Körding, 2001; Wiskott & Se-

jnowski, 2002; Hurri & Hyvärinen, 2003). We show that in some cases, simply learning higher-order statistical regularities in the data leads the model to recover more abstract properties that tend to vary slowly with time or space. This raises the possibility that the explicit computational goal of extracting coherent (slowly changing) parameters is helpful, but not necessary to learning intrinsic structures that underlie the variation in the data.

One result of learning global statistical regularities is that the learned structure is not necessarily obvious; for example, density components adapted to natural images describe a variety of statistical regularities, some of which are not easily interpreted. This is true for many unsupervised learning models that do not specify in advance the structure to be learned. For example, ICA applied to natural images yields a matrix of basis functions whose functional interpretation has ranged from edge detectors (Bell & Sejnowski, 1997) to models of biological sensory systems (van Hateren & van der Schaaf, 1998). The work presented here suggests that as more powerful unsupervised learning models are developed, the analysis of learned parameters and data representations will gain in importance.

The approach taken in this work is to attack a difficult problem—capturing intrinsic regularities in complex high-dimensional data—incrementally. Although the model is able to capture some nonlinear statistical regularities, the structure it learns is still quite low level. This step-wise approach stands in contrast to other computational schemes that solve specific problems, such as perceptual invariance or scene segmentation. This may prove more tractable and robust because it does not rely on preconceived notions of intrinsic structures but learns them from the data. This approach might also give more insight into the organization of biological perceptual systems, where each processing unit performs a relatively simple computational task, and many computational goals might be achieved incrementally and in parallel.

Appendix

The value of $\hat{\mathbf{v}}$ for a given $\mathbf{u}$ was obtained by maximizing the log posterior distribution

$$L = \log p(\mathbf{v}|\mathbf{u}, \mathbf{B}) \propto \log p(\mathbf{u}|\mathbf{B}, \mathbf{v}) p(\mathbf{v}). \tag{A.1}$$

We use the Laplace distribution for the prior on $\mathbf{v}$ and a generalized gaussian distribution with the scale parameters $\boldsymbol{\lambda}$ for the likelihood $p(\mathbf{u}|\mathbf{B}, \mathbf{v})$, so that

$$L \propto \log \prod_{i=1}^{N} z_i \exp\left(-\left|\frac{u_i}{\lambda_i}\right|^{q_i}\right) \prod_{j=1}^{M} z_j \exp\left(-\left|\frac{v_j}{c}\right|^{q_j}\right) \tag{A.2}$$

$$\propto \sum_{i=1}^{M}\left[\log \frac{q_i}{2\lambda_i \Gamma(1/q_i)} - \left|\frac{u_i}{\lambda_i}\right|^{q_i}\right] + \sum_{j=1}^{M}\left[\log \frac{q_j}{2\Gamma(1/q_j)} - \left|\frac{v_j}{c}\right|^{q_j}\right] \tag{A.3}$$

$$\propto \sum_{i=1}^{N}\left[-\log\lambda_i - \left|\frac{u_i}{\lambda_i}\right|^{q_i}\right] - \sum_{j=1}^{M}\left|\frac{v_j}{c}\right|^{q_j}, \tag{A.4}$$

where $z = q/(2\lambda\Gamma(1/q))$ is the normalization term, $\lambda_i = ce^{[\mathbf{Bv}]_i}$, and $c = \sqrt{\Gamma(1/q)/\Gamma(3/q)}$. For a given data sample, $\mathbf{u}$ is the $N \times 1$ vector of linear basis function coefficients and $\mathbf{v}$ the $M \times 1$ vector of density component coefficients. $\mathbf{A}$ is the $N \times N$ matrix of linear basis functions, and $\mathbf{B}$ is the $N \times M$ matrix of density components. We use $[\mathbf{Bv}]_i$ to denote the ith element of the vector $\mathbf{Bv}$, and $\mathbf{B}_i$ to denote the ith row of the matrix $\mathbf{B}$.

The MAP estimate $\hat{\mathbf{v}}$ was obtained by gradient ascent,

$$\frac{\partial L}{\partial v_j} = \frac{\partial}{\partial v_j}\left[\sum_{i=1}^{N}\left[-\log\lambda_i - \left|\frac{u_i}{\lambda_i}\right|^{q_i}\right] - \sum_{j=1}^{M}\left|\frac{v_j}{c}\right|^{q_j}\right] \tag{A.5}$$

$$= \sum_{i=1}^{N}\left[-B_{ij} + q_i B_{ij}\left|\frac{u_i}{ce^{[\mathbf{Bv}]_i}}\right|^{q_i}\right] - sign(v_j)q_j\frac{|v_j|^{q_j-1}}{c^{q_j}}. \tag{A.6}$$

The gradient ascent procedure was sensitive to initial conditions and in some cases did not converge to a solution. We tried several alternatives, including a closed-form approximation to the MAP estimate. Ultimately, the most effective learning method was to adjust the step size ϵ by the stochastic estimate of the Hessian over each batch of data (LeCun et al., 1998):

$$\eta_j = \frac{\epsilon}{\langle\frac{\partial^2 L}{\partial v_j^2}\rangle + \mu}, \tag{A.7}$$

where μ is a small constant that improves stability when the second derivative is very small. The second derivative for a data sample is given by

$$\frac{\partial^2 L}{\partial v_j^2} = -\sum_{i=1}^{N} q_i^2 B_{ij}^2\left|\frac{u_i}{ce^{[\mathbf{Bv}]_i}}\right|^{q_i} - q_j(q_j-1)\frac{|v_j|^{q_j-2}}{c^{q_j}}. \tag{A.8}$$

The density component matrix $\mathbf{B}$ was estimated by maximizing the posterior over the data ensemble,

$$\log p(\mathbf{B}|\mathbf{x}_1,\ldots,\mathbf{x}_N,\mathbf{A}) \propto \log p(\mathbf{x}_1,\ldots,\mathbf{x}_N|\mathbf{A},\mathbf{B})p(\mathbf{B}) \tag{A.9}$$

$$\propto \sum_n \log p(\mathbf{x}_n|\mathbf{A},\mathbf{B})p(\mathbf{B}) \tag{A.10}$$

$$\propto \sum_n \log p(\mathbf{u}_n|\mathbf{B},\hat{\mathbf{v}}_n)p(\hat{\mathbf{v}}_n)p(\mathbf{B})/|\det\mathbf{A}|. \tag{A.11}$$

Let $\hat{L}_n = \log p(\mathbf{u}_n|\mathbf{B},\hat{\mathbf{v}}_n)p(\hat{\mathbf{v}}_n)p(\mathbf{B})$. We place a gaussian prior on $\mathbf{B}$ and implement gradient ascent $\Delta B = \frac{1}{N}\sum_n \partial\hat{L}_n/\partial B_{ij}$, where the posterior for each

data sample $\mathbf{x}_n$ is

$$\frac{\partial \hat{L}}{\partial B_{ij}} = \frac{\partial}{\partial B_{ij}} \left[\log p(\mathbf{u}_n | \mathbf{B}, \hat{\mathbf{v}}_n) + \log p(\hat{\mathbf{v}}_n) + \log p(\mathbf{B}) \right] \tag{A.12}$$

$$= \frac{\partial}{\partial B_{ij}} \left[\sum_{i=1}^{N} \left[-\log \lambda_i - \left| \frac{u_i}{\lambda_i} \right|^{q_i} \right] - \sum_{j=1}^{M} \left| \frac{v_j}{c} \right|^{q_j} - \sum_{i=1,j=1}^{N,M} \frac{B_{ij}^2}{2} \right] \tag{A.13}$$

$$= -v_j + v_j q_i \left| \frac{u_i}{c e^{[\mathbf{B}\mathbf{v}]_i}} \right|^{q_i} - B_{ij} \,. \tag{A.14}$$

Acknowledgments

We thank Bruno Olshausen and Eero Simoncelli for helpful discussions. This work was supported by a Department of Energy Computational Science Graduate Fellowship to Y.K. and National Science Foundation grant no. 0238351 to M.S.L.

References

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

Bell, A. J., & Sejnowski, T. J. (1997). The "independent components" of natural scenes are edge filters. *Vision Res., 37*(23), 3327–3338.

Bollerslev, T., Engle, R. F., & Nelson, D. B. (1994). ARCH models. In R. F. Engle & D. L. McFadden (Eds.), *Handbook of Econometrics*. Amsterdam: Elsevier.

Buccigrossi, R. W., & Simoncelli, E. P. (1999). Image compression via joint statistical characterization in the wavelet domain. *IEEE Transactions on Image Processing, 8*(12), 1688–1701.

Cardoso, J.-F. (1997). Infomax and maximum likelihood for blind source separation. *IEEE Signal Processing Letters, 4*, 109–111.

Choi, S., Cichocki, A., & Belouchrani, A. (2002). Second order nonstationary source separation. *Journal of VLSI Signal Processing, 32*(1–2), 93–104.

Cichocki, A., & Amari, S.-I. (2002). *Adaptive blind signal and image processing: Learning algorithms and applications*. New York, Wiley.

Everson, R., & Roberts, S. (1999). Non-stationary independent component analysis. In *Proceedings of the 8th International Conference on Artificial Neural Networks* (pp. 503–508). Berlin: Springer-Verlag.

Foldiak, P. (1991). Learning invariance from transformation sequences. *Neural Computation, 3*(2), 194–200.

Hoyer, P. O., & Hyvärinen, A. (2002). A multi-layer sparse coding network learns contour coding from natural images. *Vision Res., 42*, 1593–1605.

Hurri, J., & Hyvärinen, A. (2003). Simple-cell-like receptive fields maximize temporal coherence in natural video. *Neural Computation, 15*(3), 663–691.

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

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

Hyvärinen, A., Hurri, J., & Väyrynen, J. (2003). Bubbles: A unifying framework for low-level statistical properties of natural image sequences. *Journal of the Optical Society of America A, 20*(7), 1237–1252.

Hyvärinen, A., Karhunen, J., & Oja, E. (2001). *Independent component analysis*. New York: Wiley.

Karklin, Y., & Lewicki, M. (2003). Learning higher-order structures in natural images. *Network: Computation in Neural Systems, 14*, 483–499.

Kayser, C., Einhäuser, W., Dümmer, O., König, P., & Körding, K. (2001). Extracting slow subspaces from natural videos leads to complex cells. *Artificial Neural Networks, 2130*, 1075–1080.

Krüger, N. (1998). Collinearity and parallelism are statistically significant second order relations of complex cell responses. *Neural Processing Letters, 8*, 117–129.

LeCun, Y., Bottou, L., Orr, G., & Müller, K. (1998). Efficient backprop. In G. Orr & K. Müller (Eds.), *Neural networks: tricks of the trade*. New York: Springer.

Lee, T.-W., Koehler, B., & Orglmeister, R. (1997). Blind source separation of nonlinear mixing models. In *Proceedings of IEEE International Workshop on Neural Networks for Signal Processing*.

Lee, T.-W., & Lewicki, M. S. (2002). Unsupervised classification, segmentation and de-noising of images using ICA mixture models. *IEEE Trans. Image Proc., 11*(3), 270–279.

Lee, T.-W., Lewicki, M. S., & Sejnowski, T. J. (2000). ICA mixture models for unsupervised classification of non-Gaussian sources and automatic context switching in blind signal separation. *IEEE Trans. on Pattern Analysis and Machine Intelligence, 22*(10), 1078–1089.

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

O'Neill, J. C. (1999). DiscreteTFDs time-frequency analysis software. Available online at: http://tfd.sourceforge.net/.

Pearlmutter, B. A., & Parra, L. C. (1996). A context-sensitive generalization of ICA. In *International Conference on Neural Information Processing* (pp. 151–157). New York: Springer.

Pham, D.-T., & Cardoso, J.-F. (2001). Blind separation of instantaneous mixtures of non stationary sources. *IEEE Trans. Signal Processing, 49*(9), 1837–1848.

Romberg, J., Choi, H., & Baraniuk, R. (2001). Bayesian tree-structured image modeling using wavelet domain Hidden Markov models. *IEEE Transactions on Image Processing, 10*(7), 1056–1068.

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

Simoncelli, E. P. (1997). Statistical models for images: Compression, restoration and synthesis. In *Proc. 31st Asilomar Conference on Signals, Systems and Computers*. Pacific Grove, CA.

van Hateren, J. H., & van der Schaaf, A. (1998). Independent component filters of natural images compared with simple cells in primary visual cortex. *Proc. Royal Soc. Lond. B*, *265*, 359–366.

Wainwright, M. J., Simoncelli, E. P., & Willsky, A. S. (2001). Random cascades on wavelet trees and their use in analyzing and modeling natural images. *Applied Computational and Harmonic Analysis*, *11*, 89–123.

Welling, M., Hinton, G. E., & Osindero, S. (2003). Learning sparse topographic representations with products of student-t distributions. In S. Becker, S. Thrun, & K. Obermayer (Eds.), *Advances in neural information processing systems*, *15*, Cambridge, MA: MIT Press.

Wiskott, L., & Sejnowski, T. J. (2002). Slow feature analysis: Unsupervised learning of invariances. *Neural Computation*, *14*(4), 715–770.

Received December 23, 2003; accepted July 2, 2004.

To appear in: Neural Information Processing Systems (NIPS), http://nips.cc/
Granada, Spain. December 12-15, 2011.

Efficient coding of natural images with a population of noisy Linear-Nonlinear neurons

Yan Karklin and Eero P. Simoncelli
Howard Hughes Medical Institute and
Center for Neural Science
New York University
New York, NY 10003
{yan.karklin, eero.simoncelli}@nyu.edu

Abstract

Efficient coding provides a powerful principle for explaining early sensory coding. Most attempts to test this principle have been limited to linear, noiseless models, and when applied to natural images, have yielded oriented filters consistent with responses in primary visual cortex. Here we show that an efficient coding model that incorporates biologically realistic ingredients – input and output noise, nonlinear response functions, and a metabolic cost on the firing rate – predicts receptive fields and response nonlinearities similar to those observed in the retina. Specifically, we develop numerical methods for simultaneously learning the linear filters and response nonlinearities of a population of model neurons, so as to maximize information transmission subject to metabolic costs. When applied to an ensemble of natural images, the method yields filters that are center-surround and nonlinearities that are rectifying. The filters are organized into two populations, with On- and Off-centers, which independently tile the visual space. As observed in the primate retina, the Off-center neurons are more numerous and have filters with smaller spatial extent. In the absence of noise, our method reduces to a generalized version of independent components analysis, with an adapted nonlinear "contrast" function; in this case, the optimal filters are localized and oriented.

1 Introduction

Coding efficiency is a well-known objective for the evaluation and design of signal processing systems, and provides a theoretical framework for understanding biological sensory systems. Attneave [1] and Barlow [2] proposed that early sensory systems are optimized, subject to the limitations of their available resources, for representing information contained in naturally occurring stimuli. Although these proposals originated more than 50 years ago, they have proven difficult to test. The optimality of a given sensory representation depends on the family of possible neural transformations to which it is compared, the costs of building, maintaining, and operating the system, the distribution of input signals over which the system is evaluated, and the levels of noise in the input and output.

A substantial body of work has examined coding efficiency of early visual representations. For example, the receptive fields of retinal neurons have been shown to be consistent with efficient coding principles [3, 4, 5, 6]. However, these formulations rely on unrealistic assumptions of linear response and Gaussian noise, and their predictions are not uniquely constrained. For example, the observation that band-pass filtering is optimal [4] is insufficient to explain rotationally symmetric (center-surround) structure of receptive fields in the retina.

The simplest models that attempt to capture both the receptive field properties and the response nonlinearities are linear-nonlinear (LN) cascades, in which the incoming sensory stimulus is projected onto a linear kernel, and this linear response is then passed through a memoryless scalar nonlinear function whose output is used to generate the spiking response of the neuron. Such approaches have been used to make predictions about neural coding in general [7, 8], and, when combined with a constraint on the mean response level, to derive oriented receptive fields similar to those found in primary visual cortex [9, 10]. These models do not generally incorporate realistic levels of noise. And while the predictions are intuitively appealing, it is also somewhat of a mystery that they bypass the earlier (e.g., retinal) stages of visual processing, in which receptive fields are center-surround.

A number of authors have studied coding efficiency of scalar nonlinear functions in the presence of noise and compared them to neural responses to variables such as contrast [11, 12, 13, 14, 15]. Others have verified that the *distributions* of neural responses are in accordance with predictions of coding efficiency [16, 17, 18, 19]. To our knowledge, however, no previous result has attempted to jointly optimize the linear receptive field and the nonlinear response properties in the presence of realistic levels of input and output noise, and realistic constraints on response levels.

Here, we develop methods to optimize a full population of linear-nonlinear (LN) model neurons for transmitting information in natural images. We include a term in the objective function that captures metabolic costs associated with firing spikes [20, 21, 22]. We also include two sources of noise, in both input and output stages. We implement an algorithm for jointly optimizing the population of linear receptive fields and their associated nonlinearities. We find that, in the regime of significant noise, the optimal filters have a center-surround form, and the optimal nonlinearities are rectifying, consistent with response properties of retinal ganglion cells. We also observe asymmetries between the On- and the Off-center types similar to those measured in retinal populations. When both the input and the output noise are sufficiently small, our learning algorithm reduces to a generalized form of independent component analysis (ICA), yielding optimal filters that are localized and oriented, with corresponding smooth nonlinearities.

2 A model for noisy nonlinear efficient coding

We assume a neural model in the form of an LN cascade (Fig. 1a), which has been successfully fit to neural responses in retina, lateral geniculate nucleus, and primary visual cortex of primate visual systems [e.g., 23, 24, 25]. We develop a numerical method to optimize both the linear receptive fields and the corresponding point nonlinearities so as to maximize the information transmitted about natural images in the presence of input and output noise, as well as metabolic constraints on neural processing.

Consider a vector of inputs $\mathbf{x}$ of dimensionality D (e.g. an image with D pixels), and output vector $\mathbf{r}$ of dimensionality J (the underlying firing rate of J neurons). The response of a neuron r_j is computed by taking an inner product of the (noise-corrupted) input with a linear filter $\mathbf{w}_j$ to obtain a generator signal y_j (e.g. membrane voltage), which is then passed through neural nonlinearity f_j (corresponding to the spike-generating process) and corrupted with additional neural noise,

$$r_j = f_j\left(y_j\right) + n_r \tag{1}$$

$$y_j = \mathbf{w}_j^T\left(\mathbf{x} + \mathbf{n}_x\right) , \tag{2}$$

(Fig. 1a). Note that we did not constrain the model to be "complete" (the number of neurons can be smaller or larger than the input dimensionality) and that each neuron can have a different nonlinearity.

We aim to optimize an objective function that includes the mutual information between the input signal and the population responses, denoted $I(X;R)$, as well as an approximate measure of the metabolic operating cost of the system. It has been estimated that most of the energy expended by spiking neurons is associated with the cost of generating (and recovering from) spikes and that this cost is roughly proportional to the neural firing rate [22]. Thus we incorporate a penalty on the expected output, which gives the following objective function:

$$I(X;R) - \sum_j \lambda_j \left\langle r_j \right\rangle . \tag{3}$$

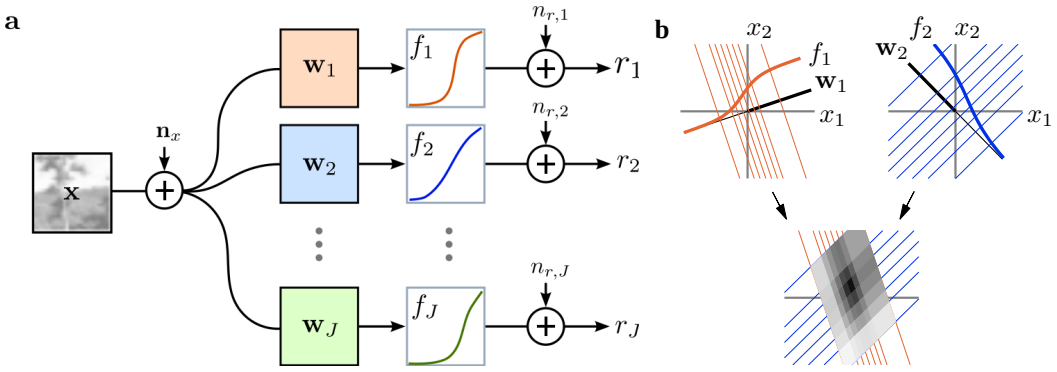


Figure 1: **a.** Schematic of the model (see text for description). The goal is to maximize information transfer between images $\mathbf{x}$ and the neural response $\mathbf{r}$, subject to metabolic cost of firing spikes. **b.** Information about the stimulus is conveyed both by the arrangement of the filters and the steepness of the neural nonlinearities. *Top:* two neurons encode two stimulus components (e.g. two pixels of an image, x_1 and x_2) with linear filters (black lines) whose output is passed through scalar nonlinear functions (thick color lines; thin color lines show isoresponse contours at evenly spaced output levels). The steepness of the nonlinearities specifies the precision with which each projection is represented: regions of steep slope correspond to finer partitioning of the input space, reducing the uncertainty about the input. *Bottom:* joint encoding leads to binning of the input space according to the isoresponse lines above. Grayscale shading indicates the level of uncertainty (entropy) in regions of the input (lighter shades correspond to higher uncertainty). Efficient codes optimize this binning, subject to input distribution, noise levels, and metabolic costs on the outputs.

Parameter λ_j specifies the trade-off between information gained by firing more spikes, and the cost of generating them. It is difficult to obtain a biologically valid estimate for this parameter, and ultimately, the value of sensory information gained depends on the behavioral task and its context [26]. Alternatively, we can use λ_j as a Lagrange multiplier to enforce the constraint on the mean output of each neuron.

Our goal is to adjust both the filters and the nonlinearities of the neural population so as to maximize the expectation of (3) under the joint distribution of inputs and outputs, $p(\mathbf{x}, \mathbf{r})$. We assume the filters are unit norm ($\|\mathbf{w}_j\| = 1$) to avoid an underdetermined model in which the nonlinearity scales along its input dimension to compensate for filter amplification. The nonlinearities f_j are assumed to be monotonically increasing. We parameterized the *slope* of the nonlinearity $g_j = df_j/dy_j$ using a weighted sum of Gaussian kernels,

$$g_j(y_j|c_{jk}, \mu_{jk}, \sigma_j) = \sum_{k=1}^{K} c_{jk} \exp\left(-\frac{(y_j - \mu_{jk})^2}{2\sigma_j^2}\right), \tag{4}$$

with coefficients $c_{jk} \geq 0$. The number of kernels K was chosen for sufficiently flexible nonlinearity (in our experiments $K = 500$). We spaced μ_{jk} evenly over the range of y_j and chose σ_j for smooth overlap of adjacent kernels (kernel centers $2\sigma_j$ apart).

2.1 Computing mutual information

How can we compute the information transmitted by the nonlinear network of neurons? Mutual information can be expressed as the difference between two entropies, $I(X;R) = H(X) - H(X|R)$. The first term is the entropy of the data, which is constant (i.e. it does not depend on the model) and can therefore be dropped from the objective function. The second term is the conditional differential entropy and represents the uncertainty in the input after observing the neural response. It is computed by taking the expectation over output values $H(X|R) = E_r\left[-\int p(\mathbf{x}|\mathbf{r}) \ln p(\mathbf{x}|\mathbf{r}) d\mathbf{x}\right]$. In general, computing the entropy of an arbitrary high dimensional distribution is not tractable. We make several assumptions that allow us to approximate the posterior, compute its entropy, and maximize mutual information. The posterior is proportional to the product of the likelihood and the prior, $p(\mathbf{x}|\mathbf{r}) \propto p(\mathbf{r}|\mathbf{x})p(\mathbf{x})$; below we describe these two functions in detail.

The likelihood. First, we assume the nonlinearity is smooth enough that, at the level of the noise (both input and output), f_j can be linearized using first-order Taylor series expansion. This means that locally, for each input $\mathbf{x}^i$ and instance of noise,

$$\mathbf{r}^i \approx \mathbf{G}^i\mathbf{W}^T(\mathbf{x}^i + \mathbf{n}_x^i) + \mathbf{n}_r^i + \mathbf{f}_0^i\,, \tag{5}$$

where $\mathbf{W}$ is a matrix collecting the neural filters, $\mathbf{f}_0^i$ is a vector of constants, and $\mathbf{G}^i$ is a diagonal matrix containing the local derivatives of the response functions $g_j(y_j)$ at $y_j(\mathbf{x}^i)$. Here we have used i to index parameters and random variables that change with each input. (Similar approximations have been used to minimize reconstruction error in neural nonlinearities [27] and maximize information in networks of interacting genes [28].)

If input and output noises are assumed to be constant and Gaussian, with covariances $\mathbf{C}_{n_x}$ and $\mathbf{C}_{n_r}$, respectively, we obtain a Gaussian likelihood $p(\mathbf{r}|\mathbf{x})$, with covariance

$$\mathbf{C}_{r|x}^i = \mathbf{G}^i\mathbf{W}^T\mathbf{C}_{n_x}\mathbf{W}\mathbf{G}^i + \mathbf{C}_{n_r}\,. \tag{6}$$

We emphasize that although the likelihood *locally* takes the form of a Gaussian distribution, its covariance is not fixed but depends on the input, leading to different values for the entropy of the posterior across the input space. Fig. 1b illustrates schematically how the organization of the filters and the nonlinearities affects the entropy and thus determines the precision with which neurons encode the inputs.

The prior. We would like to make as few assumptions as possible about the prior distribution of natural images. As described below, we rely on sampling image patches to approximate this density when computing $H(X|R)$. Nevertheless, to compute local estimates of the entropy we need to combine the prior with the likelihood. For smooth densities, the entropy depends on the curvature of the prior in the region where likelihood has significant mass. When an analytic form for the prior is available, we can use a second-order expansion of the prior around the maximum of the posterior (known as the "Laplace approximation" to the posterior). Unfortunately, this is difficult to compute reliably in high dimensions when only samples are available. Instead, we use the *global* curvature estimate in the form of the covariance matrix of the data, $\mathbf{C}_x$.

Putting these ingredients together, we compute the posterior as a product of two Gaussian distributions. This gives a Gaussian with covariance

$$\mathbf{C}_{x|r}^i = \left(\mathbf{C}_x^{-1} + \mathbf{W}\mathbf{G}^i(\mathbf{G}^i\mathbf{W}^T\mathbf{C}_{n_x}\mathbf{W}\mathbf{G}^i + \mathbf{C}_{n_r})^{-1}\mathbf{G}^i\mathbf{W}^T\right)^i \tag{7}$$

This provides a measure of uncertainty about each input and allows us to express information conveyed about the input ensemble by taking the expectation over the input and output distributions,

$$-H(X|R) = -E\left[\frac{1}{2}\ln 2\pi e\det(\mathbf{C}_{x|r}^i)\right]\,. \tag{8}$$

We obtain Monte Carlo estimates of this conditional entropy by averaging the term in the brackets over a large ensemble of patches drawn from natural images and input/output noise sampled from assumed noise distributions.

2.2 Numerical optimization

We made updates to model parameters using online gradient ascent on the objective function computed on small batches of data. We omit the gradients here, as they are obtained using standard methods but do not yield easily interpretable update rules. One important special case is derived when the number of inputs equals the number of outputs, and both noise levels approach zero. In this setting, the update rule for the filters reduces to the ICA learning rule [8], with the gradient updates maximizing the entropy of the output distributions. Because our response constraint effectively limits the mean firing rate and not the maximum, the anti-Hebbian term is different from that found in standard ICA, and the optimal (maximum entropy) response distributions are exponential, rather than uniform. Note also that our method is more general than standard ICA: it adaptively adjusts the nonlinearities to match the input distribution, whereas standard ICA relies on a fixed nonlinear "contrast" function.

To ensure all nonlinearities were monotonically increasing, the coefficients c_{jk} were adapted in log-space. After each step of gradient ascent, we normalized filters so that $\|\mathbf{w}_j\| = 1$. It was also

necessary to adjust the sampling of the nonlinearities (location of μ_{jk}'s) because, as the fixed-norm filters rotated through input space, the variance of the projections can change drastically. Thus, whenever data fell outside the range, the range was doubled, and when all data fell inside the central 25%, it was halved.

3 Training the model on natural images

3.1 Methods

Natural image data were obtained by sampling 16×16 patches randomly from a collection of grayscale photographs of outdoor scenes [29], whose pixel intensities were linear w.r.t. light luminance levels. Importantly, we did not whiten images. The only preprocessing steps were to subtract the mean of each large image and rescale the image to attain a variance of 1 for the pixels.

We assumed that the input and output noises were i.i.d., so $\mathbf{C}_{n_x}=\sigma_{n_x}^2\mathbf{I}_D$ and $\mathbf{C}_{n_r}=\sigma_{n_r}^2\mathbf{I}_J$. We chose 8dB for the input ($\sigma_{n_x}\approx 0.4$). Although this is large relative to the variance of a pixel, as a result of strong spatial correlations in the input, some projections of the data (low frequency components) had SNR over 40dB. Output noise levels were set to -6dB (computed as $20\log_{10}(\langle r_j\rangle/\sigma_{n_r})$; $\sigma_{n_r}=2$) in order to match the high variability observed in retinal ganglion cells (see below). Parameter λ_j was adjusted to attain an average rate of one spike per neuron per input image, $\langle r_j\rangle=1$.

The model consisted of 100 neurons. We found this number to be sufficient to produce homogeneous sets of receptive fields that spatially tiled the image patch. In the retina, the ratio of inputs (cones) to outputs (retinal ganglion cells) varies greatly, from almost 1:3 in central fovea to more than 10:1 in the periphery [30]. Our ratio of 256:100 is within the physiological range, but other factors, such as eccentricity-dependent sampling, optical blur, and multiple ganglion cell subtypes make exact comparisons impossible.

We initialized filter weights and nonlinearity coefficients to random Gaussian values. Batch size was 100 patches, resampled after each update of the parameters. We trained the model for 100,000 iterations of gradient ascent with fixed step size. Initial conditions did not affect the learned parameters, with multiple runs yielding similar results. Unlike algorithms for training generative models, such as PCA or ICA, it is not possible to synthesize data from the LN model to verify convergence to the generating parameters.

3.2 Optimal filters and nonlinearities

We found that, in the presence of significant input and output noise, the optimal filters have center-surround structure, rather than the previously reported oriented shapes (Fig. 2a). Neurons organize into two populations with On-center and Off-center filters, each independently tiling the visual space. The population contains fewer On-center neurons (41 of 100) and their filters are spatially larger (Fig. 2b). These results are consistent with measurements of receptive field structure in retinal ganglion cells [31] (Fig. 3).

The optimal nonlinear functions show hard rectification, with thresholds near the mode of the input distribution (Fig. 2c). Measured neural nonlinearities are typically softer, but when rectified noise is taken into account, a hard-rectified model has been shown to be a good description of neural variability [32]. The combination of hard-rectifying nonlinearities and On/Off filter organization means that the subspace encoded by model neurons is approximately half the dimensionality of the output. For substantial levels of noise, we find that even a "complete" network (in which the number of outputs equals the number of inputs) does not span the input space and instead encodes the subspace with highest signal power.

The metabolic cost parameters λ_j that yielded the target output rate were close to 0.2. This means that increasing the firing rate of each neuron by one spike per image leads to an information gain of 20 bits for the entire population. This value is consistent with previous estimates of 40-70 bits per second for the optic nerve [33], and an assumption of 2-5 fixations (and thus unique images seen) per second.

To examine the effect of noise on optimal representations, we trained the model under different regimes of noise (Fig. 4). We found that decreasing input noise leads to smaller filters and a reduction

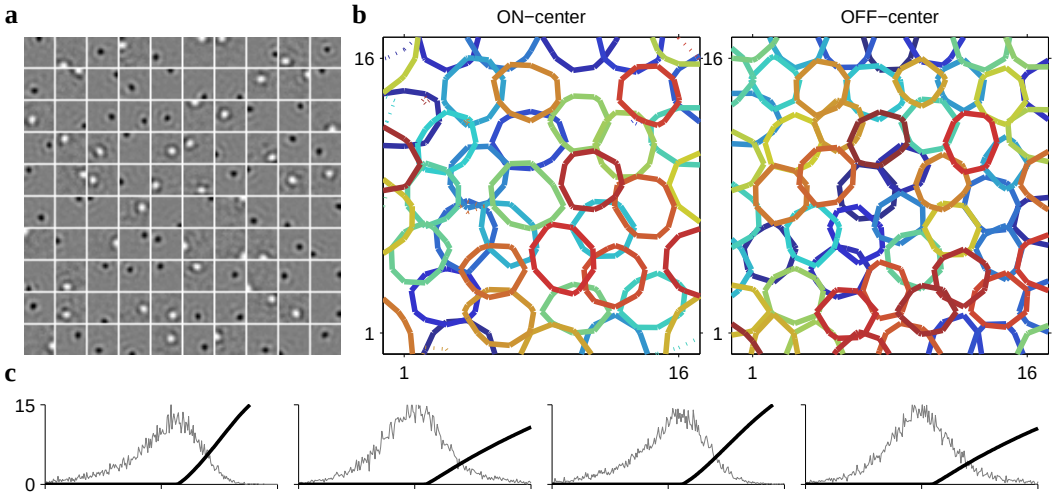


Figure 2: In the presence of biologically realistic level of noise, the optimal filters are center-surround and contain both On-center and Off-center profiles; the optimal nonlinearities are hard-rectifying functions. **a**. The set of learned filters for 100 model neurons. **b**. In pixel coordinates, contours of On-center (Off-center) filters at 50% maximum (minimum) levels. **c.** The learned non-linearities for the first four model neurons, superimposed on distributions of filter outputs.

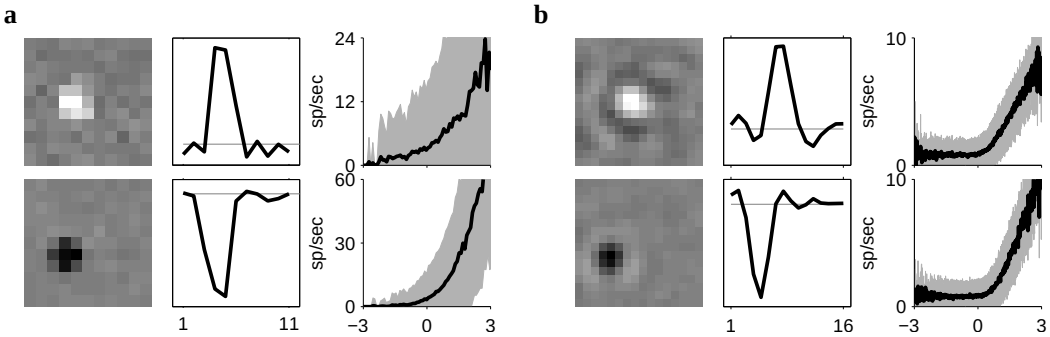


Figure 3: **a**. A characterization of two retinal ganglion cells obtained with white noise stimulus [31]. We plot the estimated linear filters, horizontal slices through the filters, and mean output as a function of input (black line, shaded area shows one standard deviation of response). **b**. For comparison, we performed the same analysis on two model neurons. Note that the spatial scales of model and data filters are different.

in the number of On-center neurons (bottom left panel). In this case, increasing the number of neurons restored the balance of On- and Off-center filters (not shown). In the case of vanishing input and output noise, we obtain localized oriented filters (top left panel), and the nonlinearities are smoothly accelerating functions that map inputs to an exponential output distribution (not shown). These results are consistent with previous theoretical work showing that optimal nonlinearity in the low noise regime maximizes the entropy of the output subject to response constraints [11, 7, 17].

How important is the choice of linear filters for efficient information transmission? We compared the performance of different filtersets across a range of firing rates (Fig. 5). For each simulation, we re-optimized the nonlinearities, adjusting λ_j's for desired mean rate, while holding the filters fixed. As a rough estimate of input entropy $H(X)$, we used an upper bound – a Gaussian distribution with the covariance of natural images. Our results show that when filters are mismatched to the noise levels, performance is significantly degraded. At equivalent output rate, the "wrong" filters transmit approximately 10 fewer bits; conversely, it takes about 50% more spikes to encode the same amount of information.

We also compared the coding efficiency of networks with variable number of neurons. First, we fixed the allotted population spike budget to 100 (per input), fixed the absolute output noise, and

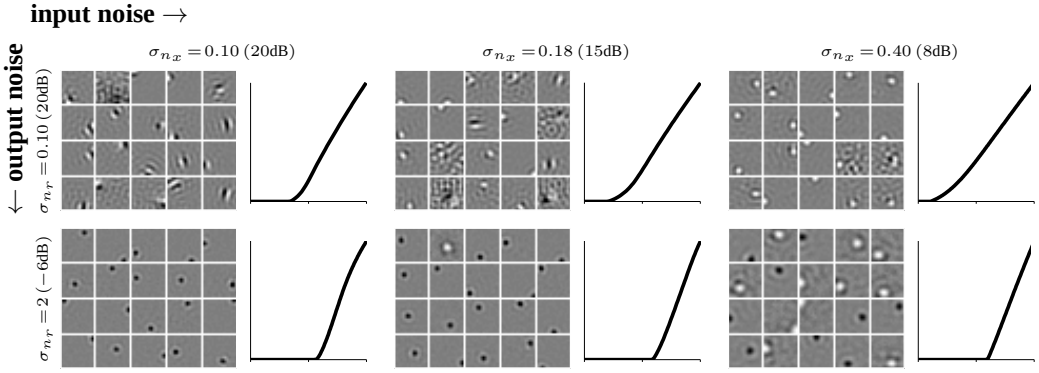


Figure 4: Each panel shows a subset of filters (20 of 100) obtained under different levels of input and output noise, as well as the nonlinearity for a typical neuron in each model.

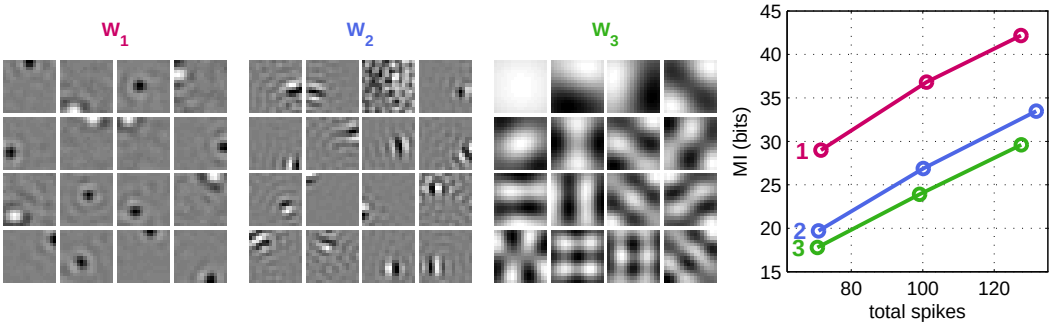


Figure 5: Information transmitted as a function of spike rate, under noisy conditions (8dB SNR$_{in}$, -6dB SNR$_{out}$). We compare the performance of optimal filters ($\mathbf{W}_1$) to filters obtained under low noise conditions ($\mathbf{W}_2$, 20dB SNR$_{in}$, 20dB SNR$_{out}$) and PCA filters, i.e. the first 100 eigenvectors of the data covariance matrix ($\mathbf{W}_3$).

varied the number of neurons from 1 (very precise) neuron to 150 (fairly noisy) neurons (Fig. 6a). We estimated the transmitted information as described above. In this regime of noise and spiking budget, the optimal population size was around 100 neurons. Next, we repeated the analysis but used neurons with fixed precision, i.e., the spike budget was scaled with the population to give 1 noisy neuron or 150 equally noisy neurons (Fig. 6b). As the population grows, more information is transmitted, but the rate of increase slows. This suggests that incorporating an additional penalty, such as a fixed metabolic cost per neuron, would allow us to predict the optimal number of canonical noisy neurons.

4 Discussion

We have described an efficient coding model that incorporates ingredients essential for computation in sensory systems: non-Gaussian signal distributions, realistic levels of input and output noise, metabolic costs, nonlinear responses, and a large population of neurons. The resulting optimal solution mimics neural behaviors observed in the retina: a combination of On and Off center-surround receptive fields, halfwave-rectified nonlinear responses, and pronounced asymmetries between the On- and the Off- populations. In the noiseless case, our method provides a generalization of ICA and produces localized, oriented filters.

In order to make the computation of entropy tractable, we made several assumptions. First, we assumed a smooth response nonlinearity, to allow local linearization when computing entropy. Although some of our results produce non-smooth nonlinearities, we think it unlikely that this systematically affected our findings; nevertheless, it might be possible to obtain better estimates by considering higher order terms of local Taylor expansion. Second, we used the global curvature of the prior density to estimate the local posterior in Eqn. 7. A better approximation would be obtained

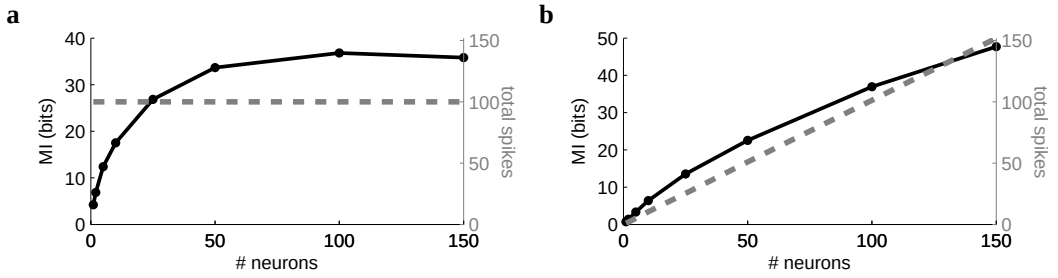


Figure 6: Transmitted information (solid line) and total spike rate (dashed line) as a function of the number of neurons, assuming (**a**) fixed *total* spike budget and (**b**) fixed spike budget *per neuron*.

from an adaptive second-order expansion of the prior density around the maximum of the posterior. This requires the estimation of local density (or rather, its curvature) from samples, which is a non-trivial problem in a high-dimensional space.

Our results bear some resemblance to previous attempts to derive retinal properties as optimal solutions. Most notably, optimal linear transforms that optimize information transmission under a constraint on total response power have been shown to be consistent with center-surround [4] and more detailed [34] shapes of retinal receptive fields. But such linear models do not provide a unique solution, nor can they make predictions about nonlinear behaviors. An alternative formulation, using linear basis functions to *reconstruct* the input signal, has also been shown to exhibit center-surround shapes [35, 6]. However, this approach makes additional assumptions about the sparsity of weights in linear filters, nor does it explicitly maximize the efficiency of the code.

Our results suggest several directions for future efforts. First, noise in our model is a known constant value. In contrast, neural systems must deal with changing levels of noise and signal, and must estimate them based only on their inputs. An interesting question, unaddressed in current work, is how to adapt representations (e.g., synaptic weights and nonlinearities) to dynamically regulate coding efficiency. Second, we are interested in extending this model to make predictions about higher visual areas. We do not interpret our results in the noiseless case (oriented, localized filters) as predictions for optimal cortical representations. Instead, we intend to extend this framework to cortical representations that must deal with accumulated nonlinearity and noise arising from previous stages of the processing hierarchy.

References

[1] F. Attneave, "Some informational aspects of visual perception.," *Psychological Review*, vol. 61, no. 3, pp. 183–193, 1954.

[2] H. Barlow, "Possible principles underlying the transformations of sensory messages," in *Sensory Communication*, pp. 217–234, MIT Press, 1961.

[3] M. V. Srinivasan, S. B. Laughlin, and A. Dubs, "Predictive coding: A fresh view of inhibition in the retina," *Proceedings of the Royal Society of London. Series B. Biological Sciences*, vol. 216, pp. 427 –459, Nov. 1982.

[4] J. J. Atick and A. N. Redlich, "Towards a theory of early visual processing," *Neural Computation*, vol. 2, no. 3, pp. 308–320, 1990.

[5] J. J. Atick, "Could information theory provide an ecological theory of sensory processing?," *Network Computation in Neural Systems*, vol. 3, no. 2, pp. 213–251, 1992.

[6] E. Doi and M. S. Lewicki, "A theory of retinal population coding," in *Advances in Neural Information Processing Systems 19* (B. Schölkopf, J. Platt, and T. Hoffman, eds.), pp. 353–360, Cambridge, MA: MIT Press, 2007.

[7] J. Nadal and N. Parga, "Nonlinear neurons in the low-noise limit: a factorial code maximizes information transfer," *Network: Computation in Neural Systems*, vol. 5, no. 4, pp. 565–581, 1994.

[8] A. J. Bell and T. J. Sejnowski, "An Information-Maximization approach to blind separation and blind deconvolution," *Neural Computation*, vol. 7, no. 6, pp. 1129–1159, 1995.

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

[10] A. J. Bell and T. J. Sejnowski, "The "independent components" of natural scenes are edge filters," *Vision Research*, vol. 37, no. 23, pp. 3327–3338, 1997.

[11] S. Laughlin, "A simple coding procedure enhances a neuron's information capacity," *Z Naturforsch*, no. Sep-Oct, 1981.

[12] A. Treves, S. Panzeri, E. T. Rolls, M. Booth, and E. A. Wakeman, "Firing rate distributions and efficiency of information transmission of inferior temporal cortex neurons to natural visual stimuli," *Neural Computation*, vol. 11, no. 3, p. 601–631, 1999.

[13] N. Brenner, W. Bialek, and R. de Ruyter van Steveninck, "Adaptive rescaling maximizes information transmission," *Neuron*, vol. 26, no. 3, pp. 695–702, 2000. PMID: 10896164.

[14] A. L. Fairhall, G. D. Lewen, W. Bialek, and R. R. de Ruyter van Steveninck, "Efficiency and ambiguity in an adaptive neural code," *Nature*, vol. 412, no. 6849, p. 787–792, 2001.

[15] M. D. McDonnell and N. G. Stocks, "Maximally informative stimuli and tuning curves for sigmoidal Rate-Coding neurons and populations," *Physical Review Letters*, vol. 101, no. 5, p. 058103, 2008.

[16] W. B. Levy and R. A. Baxter, "Energy efficient neural codes," *Neural Computation*, vol. 8, no. 3, pp. 531–543, 1996.

[17] R. Baddeley, L. F. Abbott, M. C. Booth, F. Sengpiel, T. Freeman, E. A. Wakeman, and E. T. Rolls, "Responses of neurons in primary and inferior temporal visual cortices to natural scenes.," *Proceedings of the Royal Society B: Biological Sciences*, vol. 264, no. 1389, pp. 1775–1783, 1997.

[18] V. Balasubramanian and M. J. Berry, "A test of metabolically efficient coding in the retina," *Network: Computation in Neural Systems*, vol. 13, no. 4, p. 531–552, 2002.

[19] L. Franco, E. T. Rolls, N. C. Aggelopoulos, and J. M. Jerez, "Neuronal selectivity, population sparseness, and ergodicity in the inferior temporal visual cortex," *Biol. Cybernetics*, vol. 96, no. 6, pp. 547–560, 2007.

[20] S. B. Laughlin, R. R. V. Steveninck, and J. C. Anderson, "The metabolic cost of neural information," *Nat. Neurosci*, vol. 1, no. 1, p. 36–41, 1998.

[21] P. Lennie, "The cost of cortical computation," *Current Biology*, vol. 13, pp. 493–497, Mar. 2003.

[22] D. Attwell and S. B. Laughlin, "An energy budget for signaling in the grey matter of the brain," *Journal of Cerebral Blood Flow and Metabolism*, vol. 21, no. 10, pp. 1133–1145, 2001.

[23] D. K. Warland, P. Reinagel, and M. Meister, "Decoding visual information from a population of retinal ganglion cells," *Journal of Neurophysiology*, vol. 78, no. 5, pp. 2336 –2350, 1997.

[24] J. W. Pillow, L. Paninski, V. J. Uzzell, E. P. Simoncelli, and E. J. Chichilnisky, "Prediction and decoding of retinal ganglion cell responses with a probabilistic spiking model," *The Journal of Neuroscience*, vol. 25, no. 47, pp. 11003 –11013, 2005.

[25] D. J. Heeger, "Half-Squaring in responses of cat striate cells," *Visual Neuroscience*, vol. 9, no. 05, pp. 427–443, 1992.

[26] W. Bialek, R. van Steveninck, and N. Tishby, "Efficient representation as a design principle for neural coding and computation," in *IEEE International Symposium on Information Theory*, pp. 659–663, 2006.

[27] T. von der Twer and D. I. A. MacLeod, "Optimal nonlinear codes for the perception of natural colours," *Network: Computation in Neural Systems*, vol. 12, no. 3, pp. 395–407, 2001.

[28] A. M. Walczak, G. Tkačik, and W. Bialek, "Optimizing information flow in small genetic networks. II. feed-forward interactions," *Physical Review E*, vol. 81, no. 4, p. 041905, 2010.

[29] E. Doi, T. Inui, T.-W. Lee, T. Wachtler, and T. J. Sejnowski, "Spatiochromatic receptive field properties derived from information-theoretic analyses of cone mosaic responses to natural scenes," *Neural Computation*, vol. 15, pp. 397–417, 2003.

[30] H. Wässle, U. Grünert, J. Röhrenbeck, and B. B. Boycott, "Retinal ganglion cell density and cortical magnification factor in the primate," *Vision Research*, vol. 30, no. 11, pp. 1897–1911, 1990.

[31] E. J. Chichilnisky and R. S. Kalmar, "Functional asymmetries in ON and OFF ganglion cells of primate retina," *The Journal of Neuroscience*, vol. 22, no. 7, pp. 2737 –2747, 2002.

[32] M. Carandini, "Amplification of Trial-to-Trial response variability by neurons in visual cortex," *PLoS Biol*, vol. 2, no. 9, p. e264, 2004.

[33] C. L. Passaglia and J. B. Troy, "Information transmission rates of cat retinal ganglion cells," *Journal of Neurophysiology*, vol. 91, no. 3, pp. 1217 –1229, 2004.

[34] E. Doi, J. L. Gauthier, G. D. Field, J. Shlens, A. Sher, M. Greschner, T. Machado, K. Mathieson, D. Gunning, A. M. Litke, L. Paninski, E. J. Chichilnisky, and E. P. Simoncelli, "Redundant representations in macaque retinal populations are consistent with efficient coding," in *Computational and Systems Neuroscience (CoSyNe)*, February 2011.

[35] B. T. Vincent and R. J. Baddeley, "Synaptic energy efficiency in retinal processing," *Vision Research*, vol. 43, no. 11, pp. 1285–1292, 2003.