Sparse coding with an overcomplete basis set: A strategy employed by V1?

Bruno A. Olshausen and David J. Field[†]

Dept. of Psychology and Center for Neuroscience
UC Davis
1544 Newton Ct.
Davis, CA 95616

[†]Dept. of Psychology, Uris Hall
Cornell University
Ithaca, NY 14853

January 8, 1998

Abstract

The spatial receptive fields of simple cells in mammalian striate cortex have been reasonably well described physiologically and can be characterized as being *localized*, *oriented*, and *bandpass*, comparable to the basis functions of wavelet transforms. Previously, we have shown that these receptive field properties may be accounted for in terms of a strategy for producing a sparse distribution of output activity in response to natural images (Olshausen and Field, 1996a). Here, in addition to describing this work in a more expansive fashion, we examine the neurobiological implications of sparse coding. Of particular interest is the case when the code is overcomplete—i.e., when the number of code elements is greater than the effective dimensionality of the input space. Because the basis functions are non-orthogonal and not linearly independent of each other, sparsifying the code will recruit only those basis functions necessary for representing a given input, and so the input-output function will deviate from being purely linear. These deviations from linearity provide a potential explanation for the weak forms of non-linearity observed in the response properties of cortical simple cells, and they further make predictions about the expected interactions among units in response to naturalistic stimuli.

1 Introduction

The mammalian visual cortex has evolved over millions of years to effectively cope with images of the natural environment. Given the importance of using resources efficiently in the competition for survival, it is reasonable to think that the cortex has discovered efficient coding strategies for representing natural images. In this paper, we explore to what extent theories of efficient coding can provide us with insights about cortical image representation.

The notion of efficiency we adopt is based on Barlow's principle of redundancy reduction (Barlow 1961; 1989), which states that a useful goal of sensory coding is to transform the input in such a manner that reduces the redundancy[1] due to complex statistical dependencies among elements of the input stream. The usefulness of redundancy reduction can be understood by considering the process of image formation, which occurs by light reflecting off of independent entities (i.e., objects) in the world and being focussed onto an array of photoreceptors in the retina. The activities of the photoreceptors themselves do not form a particularly useful signal to the organism because the structure present in the world is not made explicit, but rather is embedded in the form of complex statistical dependencies, or redundancies, among photoreceptor activities. A reasonable goal of the visual system, then, is to extract these statistical dependencies so that images may be explained in terms of a collection of independent events. The hope is that such a strategy will recover an explicit representation of the underlying independent entities that gave rise to the image, which would be useful to the survival of the organism.

Atick and colleagues (Atick & Redlich 1990, 1992; Atick 1992; Dong & Atick 1995; Dan et al., 1996) have achieved considerable success in showing how the principle of redundancy reduction may be applied toward understanding the response properties of retinal ganglion cells in terms of a strategy for "whitening," or decorrelating, a set of outputs in response to the $1/f$ amplitude spectrum of natural images. A limitation of their approach, however, was that it considered only the redundancy due to linear pairwise correlations among image pixels. In natural images, oriented lines and edges, especially curved, fractal-like edges, give rise to statistical dependencies that are of higher-order than linear pairwise correlations (e.g., three-point correlations) (Field, 1993; Olshausen and Field, 1996b), and so it is important to consider these forms of structure as well in developing an efficient code. Our goal here will be to find a linear coding strategy that is capable of reducing these higher-order forms of redundancy.

The strategy for reducing higher-order redundancy that we shall explore is based on using a probabilistic model to capture the image structure. In this scheme, images are described in terms of a linear superposition of basis functions, and the basis functions are adapted so as to best account for the image structure in terms of a

[1]A confusion that often arises from the term "redundancy reduction" is that it would seem to contradict the conventional wisdom that the brain contains redundant circuitry to deal with noise and physical damage. It is important however to distinguish between the form of redundancy that is present within the raw input stream (which reflects structure in the external world), and redundancy that is introduced by the nervous system through schemes such as population coding (e.g., as in the motor system). It is the former notion of redundancy that we refer to here.

collection of statistically independent events. We conjecture that the appropriate form for the probability distribution of these events is that they are *sparse*, meaning that that a given image may usually be described in terms of a small number of basis functions chosen out of a larger set (Field 1994), as illustrated in Figure 1. It was shown previously that when such a code is sought for natural images, the basis functions that emerge are qualitatively similar in form to simple cell receptive fields and also to the basis functions of certain wavelet transforms (Olshausen and Field, 1996a). Here, we shall examine more closely the consequences of utilizing an *overcomplete* code, in which the number of basis functions is greater than the effective dimensionality of the input. As we shall see, sparse coding with an overcomplete basis set leads to interesting interactions among the code elements, since sparsification weeds out those basis functions not needed to describe a given image structure. These interactions lead to deviations from a strictly linear input-output relationship, some of which have already been observed in the responses of cortical simple cells, and others of which could be tested for empirically.

We begin in Section 2 by introducing the representational framework, based on an overcomplete, linear generative model of images. Section 3 describes the probabilistic framework for modeling images in terms of sparse, statistically independent components. Section 4 then takes a deeper look at sparse coding, and why we believe sparseness to be a workable strategy for maximizing statistical independence in the representation of natural images. The issue of learning overcomplete sparse codes is taken up in Section 5, and the simulation and results obtained applying the algorithm to natural images are described in Sections 6 and 7. Finally, in Section 8 we discuss experimental predictions that arise from the model, as well as the relation between our algorithm and other efficient coding methods that have been proposed.

2 Image model

Before describing the image model, let us first revisit the standard notions of linear coding commonly adopted in the image processing community. A typical form of coding strategy is to apply a linear transform to the image by taking the inner-product of the image, $I(\vec{x})$, with a set of spatial weighting functions, ψ:

$$b_i = \sum_{\vec{x}} \psi_i(\vec{x})\, I(\vec{x})\,, \tag{1}$$

where $\vec{x}$ denotes a discrete spatial position within the two-dimensional image. The output of the transformation is represented by the values b_i. Alternatively we may write this operation in vector notation as

$$\mathbf{b} = \mathbf{W}\,\mathbf{I} \tag{2}$$

where $\mathbf{I}$ is the vector with components $I_i = I(\vec{x}_i)$ and $\mathbf{W}$ is the matrix with components $W_{ij} = \psi_i(\vec{x}_j)$. Generally, the goal in such a coding strategy is to find an invertible weight matrix $\mathbf{W}$ that transforms the input so that some criterion of optimality on the output activities is met (e.g., decorrelation, sparseness, etc.). This is the basis of coding strategies such as the Discrete Cosine Transform (used in JPEG image compression), or orthonormal wavelet transforms (Mallat, 1989). In terms of a physical implementation, such a coding scheme could be realized by a strictly feed-forward neural network, in which case the functions $\psi_i(\vec{x})$ would correspond to the spatial "receptive field" of each output b_i. This is what one usually thinks of as the standard model of a cortical simple cell, although the physiological data show that there are interesting forms of non-linearity in these cells (e.g., Tadmor and Tolhurst, 1989) that are not captured by such a straightforward, linear model.

An alternative way of coding images within a linear framework, which we explore in this paper, is in terms of a generative model, illustrated in Figure 2. Here, the image is modeled in terms of a linear superposition of basis functions, $\phi_i(\vec{x})$, mixed together with amplitudes a_i:

$$I(\vec{x}) = \sum_i a_i \, \phi_i(\vec{x}) \,. \tag{3}$$

The choice of basis functions, ϕ_i, determines the image code. The a_i are then computed for each image to satisfy the above equality, and these quantities constitute the output of the code. In the terminology adopted by the wavelet community, the ψ_i are *analysis functions* and the ϕ_i are *synthesis functions*. In some cases it may be possible to directly relate the analysis functions to the synthesis functions. For example, when the ϕ_i are linearly independent and there are as many of them as there are inputs, then the ϕ_i are equal to the rows of the inverse transpose of the weight matrix formed by the ψ_i, i.e., $\phi_i(\vec{x}_j) = (\mathbf{W}^{-\mathbf{1}})_{ji}$. If the ϕ_i form an orthonormal basis, then the code is self-inverting, i.e., $\psi_i(\vec{x}) = \phi_i(\vec{x})$. However, in general these conditions may not hold. In particular, if the code is *overcomplete*, which is when the number of basis functions exceeds the effective dimensionality of the input (i.e., number of non-zero eigenvalues in the input covariance matrix), then there will be multiple solutions for the a_i for explaining any given image.

In this paper we shall be exclusively concerned with the case where the basis set is overcomplete. One obvious reason for desiring an overcomplete code is that it possesses greater robustness in the face of noise and other forms of degradation. The reason more pertinent to our purposes, though, is that an overcomplete code will allow greater flexibility in matching the generative model to the input structure. This is especially important for images, because there is little reason to believe *a priori* that images are composed of N discrete independent causes (where N is the dimensionality of the input). Indeed, the features that compose images occur along a continuum of positions and scales, and so an overcomplete code should allow for smooth interpolation along this continuum. This point has been emphasized by Simoncelli et al. (1992), who show that overcomplete codes allow for small translations or scaling of local image features to result in a smooth and graceful change in the distribution of activity among coefficients. By contrast, in a *critically sampled* code, where the number of basis functions exactly equals the number of effective input dimensions, local image changes typically result in fairly global and drastic undulations among the coefficient values. Such instabilities would be undesirable for doing pattern analysis, and also in terms of maintaining a sparse image representation.

It should be noted at this point that in order to recover the truly independent components of images (i.e., objects), we would need to utilize an image model that goes beyond simple linear superposition and incorporates notions of translation and scale (since the appearance of objects on the retina changes depending on viewing configuration), as well as other non-linear aspects of imaging such as occlusion. We shall revisit these concerns in Section 8.4, but for now we choose not to deal with these extra complications and we restrict ourselves to the admittedly impoverished class of overcomplete, linear image models in order to ask what set of bases, ϕ, best capture the independent structure in images. This is a useful question to ask, because simple-cells are still fairly linear, and early processing stages may be limited in the complexity of image model that can be achieved.

3 Probabilistic framework

Our problem now is to find a set of basis functions, ϕ, that can best account for the structure in images in terms of a linear superposition of sparse, statistically independent events. In the language of probability theory, we wish to match as closely as possible the distribution of images arising from our linear image model, $P(I|\phi)$, to the actual distribution of images observed in nature, $P^*(I)$. In other words, if we were to generate images stochastically by drawing each a_i in Equation 3 independently from a distribution such as depicted in Figure $1b$, what would the probability distribution of generated images look like, and how could we adapt ϕ to resemble the distribution of images generated by nature? In order to calculate the probability of images arising from the model, we need to specify the prior probability distribution over the coefficients, $P(a)$, as well as the probability of an image arising from a certain state of the coefficients in the model, $P(I|a,\phi)$. Once we have specified these two probabilistic aspects of the imaging model, then the probability of an image arising from the model is given by

$$P(I|\phi) = \int P(I|a,\phi)P(a)da\,. \tag{4}$$

We shall first specify the form of the distributions for $P(I|a,\phi)$ and $P(a)$, and then discuss the problem of assessing how closely our model distribution of images matches that observed in nature.

The probability of an image arising from a particular choice of coefficients, $P(I|a,\phi)$, essentially expresses our model of the level of noise or uncertainty in the imaging process. If we assume Gaussian, white, additive image noise, then our imaging model becomes

$$I(\vec{x}) = \sum_i a_i\phi_i(\vec{x}) + \nu(\vec{x})\,. \tag{5}$$

and the probability of an image arising from a particular choice of coefficients, a, is given by

$$P(I|a,\phi) = \frac{1}{Z_{\sigma_N}} e^{-\frac{|I-a\phi|^2}{2\sigma_N^2}} \tag{6}$$

where $|I - a\phi|^2$ denotes the sum $\sum_{\vec{x}}[I(\vec{x}) - \sum_i a_i\phi_i(\vec{x})]^2$, σ_N^2 is the variance of the noise, and Z_{σ_N} is a normalizing constant. If the basis set is overcomplete, then there will be an infinite variety of a's for explaining any given image, and so $P(I|a,\phi)$ will take the form of a Gaussian ridge along the line (or plane, etc.) where $I = a\phi$. This is illustrated in two-dimensions in Figure 3*a*.

The prior probability distribution over the coefficients, $P(a)$, is where we incorporate the notion of sparse, statistically independent components into the image model. Statistical independence is incorporated by specifying $P(a)$ to be a *factorial* distribution in the a_i:

$$P(a) = \Pi_i P(a_i). \tag{7}$$

Thus, the probability of any state, a, of the coefficients is simply given by the product of individual probabilities of each component, a_i. The notion of sparseness is incorporated by shaping the probability distribution of each a_i to be uni-modal and peaked at zero with heavy tails (i.e., implying that units are mostly inactive, as in Figure 1*b*). We choose to parameterize this distribution as

$$P(a_i) = \frac{1}{Z_\beta} e^{-\beta S(a_i)} \tag{8}$$

where the function S determines the shape of the distribution, β is a parameter that controls its steepness, and Z_β is a normalizing constant. For example, choosing $\beta = 1$ and $S(x) = \log(1 + x^2)$ corresponds to specifying a Cauchy distribution for the prior, which has the desired sparse shape (Figure 3*b*).

To assess how well the probability distribution of images generated by our model, $P(I|\phi)$, matches the actual probability distribution of images sampled from nature, $P^*(I)$, we take the Kullback-Leibler divergence between the two distributions,

$$KL = \int P^*(I) \log \frac{P^*(I)}{P(I|\phi)} dI\,. \tag{9}$$

This measures the average information gain, per image drawn from $P^*(I)$, for judging in favor of the image being drawn from $P^*(I)$ as opposed to $P(I|\phi)$ (Kullback, 1959). The greater the difference between the two distributions, the greater will be KL, with $KL = 0$ if and only if the two distributions are equal. Because $P^*(I)$ is fixed, minimizing KL amounts to maximizing $\langle \log P(I|\phi) \rangle$, since

$$\langle \log P(I|\phi) \rangle = \int P^*(I) \log P(I|\phi) dI\,. \tag{10}$$

Thus, the goal of learning will be to find a set of ϕ that maximize the average log-likelihood of the images under a sparse, statistically independent prior.

4 Why sparseness?

The reason for conjecturing that sparseness is an appropriate prior for the a_i is based on the intuition that natural images may generally be described in terms of a small number of structural primitives—for example, edges, lines, or other elementary features (Field, 1994). In addition, one can see evidence for sparse structure in images by filtering them with a set of log-Gabor filters and collecting histograms of the resulting output distributions; these distributions typically show high kurtosis (Field, 1993), which is indicative of sparse structure.

Another form of reasoning that leads one to believe sparse coding is appropriate for natural images is to consider what would be implied by seeking an alternative form of probability distribution—e.g., where the code elements are multi-modally distributed. In this case, a given event or image feature would take on two or more values frequently and spend little time in between. Indeed, it is difficult to conceive of such examples in natural images. What tends to be more typically the case is that an event occurs rarely (spends most of the time zero), and when it does occur it does so along a continuum, giving rise to a distribution such as depicted in Figure 1*b*.

Note that these reasons for desiring sparseness are separate from those that have been written about elsewhere, such as increasing capacity in associative memory (Baum et al., 1988), minimizing wiring length and ease of forming associations (Foldiak, 1995), or metabolic efficiency (Baddeley, 1996). While these are obvious advantages of a sparse code, they are independent from the criteria we are considering here. If the data were actually composed from causes with multimodal distributions with heavy peaks around non-zero values, then seeking a sparse code would constitute an inappropriate strategy. In other words, sparse coding is not a general principle for finding statistically independent components in data; it only applies if the data actually have sparse structure.

5 Learning sparse codes

We now turn to the problem of learning a set of basis functions, ϕ, for the image model that best accounts for images in terms of sparse, statistically independent components. As described in the probabilistic framework above, the goal is to find a set of bases, ϕ^*, such that

$$\phi^* = \arg\max_{\phi} \langle \log P(I|\phi) \rangle \, . \qquad (11)$$

Unfortunately, this is easier said than done because evaluation of $P(I|\phi)$ requires integrating over all possible states of a (Equation 4), which is in general intractable. However, if we assume that the function inside the integral, $P(I, a|\phi) = P(I|a, \phi)P(a)$, has a fairly tightly peaked maximum in a-space (Fig. 3*c*), then we may approximate the volume under this surface, (i.e., the integral), by evaluating $P(I, a|\phi)$ only at its maximum. Our goal then becomes to find a ϕ^* such that

$$\phi^* = \arg\max_{\phi} \left\langle \max_{a} \log P(I|a, \phi)P(a) \right\rangle \, . \qquad (12)$$

The price we pay for this approximation, though, is that there will be a trivial solution for the ϕ_i, since the greater their norm, $\sum_{\vec{x}} |\phi_i(\vec{x})|^2$, the smaller each a_i will become, thus increasing $P(I, a|\phi)$ due to the peak at zero in $P(a)$. This problem may be alleviated by adding an appropriate constraint on the length of the basis functions, as described below.

In order to see what the optimization problem of Equation 12 involves, it is helpful to first re-cast the objective in an energy function framework by defining $E(I, a|\phi) = -\log P(I|a, \phi)P(a)$, in which case Equation 12 may be restated as

$$\phi^* = \arg\min_{\phi} \left\langle \min_{a} E(I, a|\phi) \right\rangle \tag{13}$$

where

$$E(I, a|\phi) = \sum_{\vec{x}} \left[I(\vec{x}) - \sum_i a_i \phi_i(\vec{x}) \right]^2 + \lambda \sum_i S(a_i) \, . \tag{14}$$

The last step was obtained by using the expressions for $P(I|a, \phi)$ and $P(a)$ in Equations 6-8, and setting $\lambda = 2\sigma_N^2 \beta$. The function to be minimized, $E(I, a|\phi)$, is the sum of two terms: the first term computes the reconstruction error, which forces the ϕ to span the input space, and the second term incurs a penalty on the coefficient activities, which encourages sparse representations. E is minimized in two separate phases, one nested inside the other. In the inner phase, E is minimized with respect to the a_i for each image, holding the ϕ_i fixed. In the outer phase (i.e., on a long timescale, over many image presentations), E is minimized with respect to the ϕ_i.

The inner loop minimization over the a_i may be performed by iterating, by some appropriate procedure, until the derivative of $E(I, a|\phi)$ with respect to each a_i is zero. For each image, then, the a_i are determined from the equilibrium solution to the differential equation

$$\dot{a}_i = \sum_{\vec{x}} \phi_i(\vec{x}) r(\vec{x}) - \lambda \, S'(a_i) \, , \tag{15}$$

where $r(\vec{x})$ is the residual image

$$r(\vec{x}) = \left[I(\vec{x}) - \sum_i a_i \phi_i(\vec{x}) \right] \, . \tag{16}$$

According to Eq. 15, the a_i are driven by a sum of two terms. The first term takes a spatially weighted sum of the current residual image using the function $\phi_i(\vec{x})$ as the weights. The second term applies a non-linear self-inhibition on the a_i, according to the derivative of S, that differentially pushes activity towards zero as shown in Figure 4*b*. A neural network implementation of this computation is shown in Figure 5.

The outer loop minimization over the ϕ_i may be accomplished by simple gradient descent. This yields the learning rule:

$$\Delta\phi_i(\vec{x}) = \eta \, \langle a_i \, r(\vec{x}) \rangle \, , \tag{17}$$

where η is the learning rate. In terms of the network implementation shown in Figure 5, the ϕ_i are updated by simple Hebbian learning between the outputs computed for each image, a_i, and the resulting residual image, $r(\vec{x})$. As mentioned above, though, doing this alone will result in the ϕ_i growing without bound, and so to prevent this from happening the L2 norm of each basis function, $l_i = \sum_{\vec{x}} |\phi_i(\vec{x})|^2$, is then adapted so that the output variance of each a_i is held at an appropriate level:

$$l_i^{new} = l_i^{old} \left[\frac{\langle a_i^2 \rangle}{\sigma_{goal}^2} \right]^\alpha , \tag{18}$$

where σ_{goal}^2 is the desired variance of the coefficients.

An intuitive interpretation of the algorithm is that on each image presentation, the gradient of S "sparsifies" the distribution of activity on the a_i by differentially reducing the value of low-activity coefficients more than high-activity coefficients. The ϕ_i then learn on the error induced by this sparsification process, resulting in a set of bases that can tolerate sparsification with minimum mean square reconstruction error. When the basis set is overcomplete and non-orthogonal, the effect of sparsification will be to choose, in the case of overlaps, which bases are most effective for describing a given image structure. This interaction between bases will cause the outputs to be a somewhat non-linear function of the inputs. Note also that there is no closed-form solution for the a_i in terms of the input, $I(\vec{x})$. Rather, the a_i are determined as the result of a recurrent computation. The form of this computation is very similar to an *analysis/synthesis loop*, which has been proposed by Mumford (1994) as a way that cortical feedback could be used to perform inference on images. In this case, the system is trying to infer which bases are most appropriate for explaining a given image.

6 Simulation methods

In order to confirm that the algorithm is capable of recovering sparse, independent structure, we tested it on a number of artificial data sets containing known forms of sparse structure. The method and results of these tests are described elsewhere (Olshausen and Field, 1996a). Here, we focus on applying the algorithm to natural images.

The data for training were taken from ten 512×512 pixel images of natural surroundings (trees, rocks, mountain scenes, etc.). These data in their raw form pose potential problems, however, because of vast inequities in variance along different directions of the input space, and also because of corrupted and artifactual data at the highest image spatial-frequencies. The large inequities in variance are due to the $1/f^2$ power spectrum of natural images. (Because the image statistics are roughly stationary, the eigenvectors of the covariance matrix will essentially be equivalent to the Fourier bases. Thus, the variance along the low-frequency eigenvectors will be much larger than the variance along the high-frequency eigenvectors). This produces huge differences in the variance along different directions, which will be troublesome for gradient descent techniques searching for structure in this space. A standard technique to ameliorate these effects is to "sphere" the data by equalizing the variance in all directions (Friedman, 1987), as schematically illustrated in Figure 6*a*. Since the amplitude spectrum falls as roughly $1/f$ at all orientations in the 2D frequency plane, sphering may be accomplished by filtering with a circularly symmetric "whitening filter" with frequency response, $W(f) = f$, thereby attenuating the low frequencies and boosting the high frequencies so as to yield a roughly flat amplitude spectrum across all spatial-frequencies. However, it is not wise to boost all high-frequencies indiscriminately for several reasons: One is that the highest spatial frequencies in most digitized images will typically be corrupted by noise and effects of aliasing[2]. Second, the energy present in the corners of the 2D frequency domain is an artifact of working on a rectangular sampling lattice, since there is an effectively higher sampling density along the diagonals (by a factor of $\sqrt{2}$) than along the vertical or horizontal directions (Fig. 6*b*). For these reasons, it is appropriate to cut out the energy at the highest spatial frequencies and also in the corners of the 2D Fourier plane by filtering with a circularly symmetric low-pass filter. We chose for this purpose an exponential filter with frequency response, $L(f) = e^{-(f/f_0)^n}$, with a cutoff frequency, f_0, of 200 cycles/picture, and a "steepness parameter," n, of four. The latter was chosen to produce a fairly sharp cutoff (to avoid eliminating too much of the data) but without being so sharp as to introduce substantial ringing in the space domain. The combined whitening/lowpass filter used to preprocess the data thus had a frequency response of

$$\begin{aligned} R(f) &= W(f)L(f) && (19) \\ &= f\, e^{-(f/f_0)^4}\,. && (20) \end{aligned}$$

The phase of the filter was set to zero across all frequencies. The profile of the resulting filter in both the frequency and space domain is shown in Figure 6*c*. Such a filter roughly resembles the spatial-frequency response characteristic of retinal ganglion cells.

Training data were obtained by extracting 12x12 image patches at random from the preprocessed images, skipping over any patch within four pixels of the border of the image. Also, to speed up training, any image patch with less than 10% of the average image variance was discarded, as these patches have such low variance that they contribute little to establishing a gradient for the ϕ's, yet they consume an equal amount of computation time.

The a_i were computed by first initializing to

$$a_i^0 = \sum_{\vec{x}} \phi_i(\vec{x}) I(\vec{x}) \qquad (21)$$

and then iterating Eq. 15 using the conjugate gradient method, halting after 10 iterations, or when the change in E was less than 1% (whichever came first). The stopping point was chosen by observing that after this many iterations only very slight changes occurred on the a_i.

[2]In order to avoid aliasing, an image should be sufficiently blurred before sampling so that the power spectrum is reduced to nearly zero by the Nyquist frequency corresponding to the largest sample spacing in the grid. In order to do this, though, the resulting sampled image will end up looking blurred, and so more often than not the integrity of data at the highest spatial-frequencies is sacrificed in order to make the image look "sharp."

A set of 144 basis functions was initialized to random values and was updated according to Equations 17 and 18 based on averages computed over every 100 image presentations. The learning rate parameter η was gradually lowered during learning, with an initial setting of 5.0 for the first 600 iterations, then 2.5 for the second 600 iterations, and finally 1.0 for the remainder. The rate parameter for the gain adjustment, α, was set to .01 and the target level for the output variance, σ^2_{goal}, was set to the variance of the image pixels, σ^2_I.

The value of the parameter λ was set relative to σ_I so that $\lambda/\sigma_I = 0.1$. The form of the sparseness cost function was $S(x) = \log(1 + x^2)$.

7 Results

A stable solution was usually arrived at after approximately 2000 updates ($\sim 200,000$ image presentations). The result is shown in Figure 7. The vast majority of basis functions have become well localized within each array (with the exception of the low frequency functions which, as expected, occupy a larger spatial extent). The functions are also oriented and broken into different spatial-frequency bands. This result should not come as a surprise, because it simply reflects the fact that natural images contain localized, oriented structures with limited phase alignment across spatial-frequency (Field, 1993). Indeed, the result makes intuitive sense, because common image structures such as lines and edges may be captured using only a handful of oriented basis functions, rather than having a separate descriptor for each pixel along the line or edge. This can be observed from Figure 9: the learned bases code for the structures in natural images more sparsely than pixels or a set of bases chosen at random.

The general form of the solution (i.e., localized, oriented, bandpass functions) is very robust, and has been observed for values of λ ranging from 0.05 to 0.15, as well as for different forms for the prior (e.g., Laplacian). It should also be noted that the preprocessing steps mentioned previously do not affect the overall, qualitative appearance of the basis functions (i.e., localized, oriented, bandpass functions). The main effect of whitening is that it vastly decreases the time required for learning, because better gradients (i.e., those pointing toward the true minimum) are obtained for minimizations with respect to both the a_i and ϕ_i, and so fewer iterations are required for both of these variables. Harpur (1997) has devised a modification to the algorithm that speeds up the learning without requiring whitening, and the results look very similar to those shown here. The main effect of low-pass filtering is that it removes artifacts of rectangular sampling. Without lowpass filtering, there is a visible anisotropy in orientation tuning, with diagonally oriented functions becoming somewhat more elongated than horizontal or vertically oriented functions. In addition, some functions appear like localized checkerboards, which would be expected in order to tile the far corners of the 2D Fourier plane.

The entire set of basis functions forms a complete image code that spans the joint space of spatial position, orientation, and scale in a manner similar to wavelet codes, which have previously been shown to form sparse representations of natural images

(Field, 1987; Field, 1994; Daugman, 1989). Shown in Figure 8 is the distribution of the basis functions in spatial frequency and orientation. The vast majority lie within the high spatial frequency bands, as expected of a wavelet code in order to form a complete tiling of space and spatial-frequency. Note however that the basis functions deviate somewhat from strict self-similarity, in that the high spatial-frequency functions have more wobbles (are more narrowly tuned in log-frequency) than the low spatial-frequency functions. Characterizing the bandwidth vs. spatial-frequency relationship more adequately will require simulations over larger window sizes in order to span a larger range of spatial-frequencies.

Although the number of basis functions equals the number of input pixels, the representation is effectively about 1.5 times overcomplete (this one can discern by observing that the eigenvalues of the input covariance matrix, as well as the singular values of the ϕ matrix, begin to drop off sharply at about 100 dimensions). The effect of sparsification with an overcomplete representation is demonstrated in Figure 9. Here we compare the distribution of activity obtained with a purely feedforward computation,

$$b_i = \sum_{\vec{x}} \phi_i(\vec{x}) I(\vec{x}) \tag{22}$$

to the sparsified coefficient values, a_i. One can readily see that in the latter case, the sparseness cost function shifts the responsibility for coding the structure onto only those units that best match the structure, silencing the other units. Thus, the input-output relationship for any given unit will be somewhat non-linear, with units becoming more choosy in what aspects of the image they respond to. Because of this non-linear response property, and because there is no closed-form solution for the response of each a_i to any given image, the "receptive field" of each unit may only be discerned by mapping it out with various spatial functions, similar to methods employed in physiological experiments. Previously, we ascertained the receptive field for each unit by spot-mapping and showed that they are basically similar in form to the basis functions, with somewhat tighter spatial localization (Olshausen and Field, 1996, Figure 4*b*). Here, in addition to mapping out with spots, we also mapped out the response to gratings at every spatial frequency and orientation. The result of these assays for one unit (#120, 11th row, 1st column), are shown in Figure 10. In both the space and frequency domains, the unit becomes more selective to stimulus properties, because if there is another unit that does better it will take over. The effect of this can be seen by taking the inverse Fourier transform of the spatial frequency response, which shows more undulations than obtained with spot mapping, due to the sharper cutoff in spatial-frequency. A similar effect has been observed in cortical cells (Tadmor and Tolhurst, 1989).

8 Discussion

8.1 Model predictions

The most important prediction that arises from the overcomplete sparse coding model is that one would expect to observe interesting forms of interaction in the response of simple-cells while coding images. An example scenario is illustrated in Figure 11. Given two units with overlapping basis functions, then a strictly feedforward computation that took the inner product of each basis function with the image would result in both units responding, the one most aligned with the edge having somewhat higher activity than the other. If the code is sparsified, then the unit most aligned with the edge will take responsibility for coding it, and the other unit will be suppressed since it is not needed. A potential advantage of such a coding scheme is that forming associations will be made easier by not having to consider relationships among more units than are truly necessary for representing a given structure. A possible disadvantage is that the loss of a population-style code would be more susceptible to noise, and small changes in the input (e.g., a small translation of an image feature) will result in distinctly different patterns of neural activity (since a new basis function will code for the translated structure). In any case, the question of which of these coding schemes are employed could be resolved using multi-unit recording methods. By isolating two overlapping simple-cells, as ascertained by spot mapping or other methods, and observing their joint activity in response to more naturalistic stimuli containing edges, contours and the like, one could see if there is a trade off of responses as depicted in Figure 11.

An outcome of the sparse coding learning algorithm that pertains to cortical image representation is that there are many more basis functions at the high-spatial frequency bands, with substantially fewer in the lower spatial-frequency bands. Such a tiling of space and spatial-frequency would be expected of a wavelet code (the exact proportions depend upon bandwidth spacing: a factor of four decrease in number would be expected for an octave decrease in spatial frequency). However, the currently available physiological assays on the relative numbers of cells in different spatial-frequency bands are in disagreement with this general picture. Two studies in macaque V1 put the vast majority of simple-cells in the mid to low spatial-frequency range—i.e., around 4-8 cy/deg in the parafoveal region, when the highest spatial frequency band should be in the range of 16-32 cy/deg. (DeValois et al., 1982; Parker and Hawken, 1988). However, there is good reason to believe that the number of high-frequency cells may have been substantially underestimated since these units will generally have smaller receptive fields and so will be much more difficult to isolate than a low-frequency unit that exhibits a more prolonged response to bars and the like (Olshausen and Anderson, 1994). This will be an important issue to resolve in future experiments if wavelet-like codes are to be taken seriously as models of a complete early visual representation.

8.2 Sparse coding vs. coarse coding

The notion of sparse coding, in which a relatively small number of units are recruited to represent a given image, would seem to be at odds with the notion of coarse coding or population codes, in which large numbers of units participate in coding a single parameter or attribute such as color or stimulus velocity. However, it should be noted that the code being utilized here is a sparse, *distributed* code, which actually occupies a middle ground between dense population codes at one end and local representations (i.e., grandmother cells) at the other (Foldiak 1995; Hinton & Ghahramani 1997). Note for example that the learned basis functions are broadly tuned to some stimulus dimensions (e.g., spatial-frequency), as would be expected of a coarse code, while narrowly tuned to others (e.g., position), as in a local code. In a sparse distributed code, units both share in the representation of different images and also minimize the total number active per image.

Dense population codes are appropriate in situations where only a single or few attributes need to be encoded, such as the intended position of an actuator (i.e., in the motor system). When it is important to represent many attributes simultaneously, such as various spatial features in an image, introducing population codes would effectively blur over spatial position, and so two nearby features would be indistinguishable from a single feature positioned at the mean of the two. Thus, sparse coding and coarse coding schemes are appropriate under different circumstances. Resolving where and how in the nervous system these different coding schemes are played out will be an important goal of future experiments.

8.3 Relation to other work

Harpur and Prager (1996) have developed an algorithm, concurrently and independently to us, that is virtually identical to ours. They have applied their algorithm to a number of test problems, showing that it can learn sparse structure in data. The have also tested it on natural images, obtaining similar results to ours (without prewhitening).

There are several algorithms quite similar to ours based on the idea of finding independent components in data, or so-called "Independent Components Analysis" (ICA). Among these are the algorithms of Comon (1994), Amari (1995), and Bell and Sejnowski (1995). The one most closely related to ours is that of Bell and Sejnowski (see also their article in this issue). The formal relationship described in Appendix A (see also Olshausen, 1996), shows that both algorithms are solving the same maximum-likelihood problem, described in Section 3, but by making different simplifying assumptions. Bell and Sejnowski assume the weight matrix to be square and of full rank so that a unique solution exists for the a_i in terms of a feedforward model (Eq. 1). The advantage of this approach is that the algorithm runs considerably faster. The disadvantage is that the code cannot be made overcomplete and so would be limited to a critically sampled representation (see discussion of this issue in Section 2). When trained on natural images, Bell and Sejnowski's algorithm develops both receptive fields and basis functions qualitatively similar to those described here, the major difference being that more of the units are grouped into high-frequency, broadband functions rather than spanning a range of spatial-frequencies. In addition, their algorithm develops "checkerboard" receptive fields which arise as an artifact of working on a rectangular sampling lattice (their training images were not lowpass filtered like ours to remove the energy in the corners of the 2D Fourier domain). When their algorithm is trained on our images (which have been lowpass filtered and thus have reduced dimensionality), many of the basis functions dimply drop out (i.e., take on zero norm) because the algorithm is not able to utilize the extra dimensions.

Other methods for learning sparse codes have been described by Foldiak (1990) and Zemel (1993). The principal difference here is that unit values were considered to be binary, although the models could conceivably be extended to the analog domain. Both of these algorithms formed the inspiration for the development of our algorithm.

Another class of efficient coding methods is based on projection pursuit methods (Friedman 1987; Intrator, 1992; Law and Cooper, 1994; Fyfe and Baddeley, 1995; Press and Lee, 1996; Lu et al., 1996). Some of these were trained on natural images, but with the exception of Press and Lee (1996) and Lu et al. (1996), they did not show a full family of receptive fields for forming a complete image code.

Finally, in the realm of generative models, Dayan et. al. (1995) and Rao and Ballard (1997) have described methods for learning the causal structure in data in a hierarchical fashion. Rao and Ballard's network, when reduced to a single-layer system such as ours, is very similar but uses a quadratic penalty term (corresponding to a Gaussian prior). When trained on natural images, it does not develop localized receptive fields (they are artificially localized by using a Gaussian spatial window), presumably because of the prior being Gaussian rather than sparse.

8.4 Future challenges

A major limitation of the work we have presented here is that it relies entirely on a linear image model, and so it will necessarily be limited in the forms of independent structure that it can extract from images. The real causes of images (e.g., objects) do not mix linearly but rather occlude one another and also undergo shifts in position, changes in size, rotations, etc. These types of interaction would need to be included in the generative image model in order to have any hope of recovering the real causes of images using the independence principle. For example, to deal with translation one may modify Eq. 3 to be of the form

$$I(\vec{x}) = \sum_i a_i \, \phi_i(\vec{x} + \Delta\vec{x}_i) \,, \tag{23}$$

in which case one would need to determine the shift parameters, $\Delta\vec{x}_i$, in addition to the a_i, for each image. For the case of occlusion, Saund (1995) has described a "soft-or" function for dealing with feature overlaps in a binary image domain. In the analog domain, it would appear necessary to introduce another dynamic variable to represent depth or ordering in the scene in order to properly render overlapping objects.

Another shortcoming of the current image model is that it utilizes only a single stage. Surely, there will be statistical dependencies among the elements of the single-stage model, and it would be desirable to have these modeled by a second and third stage in a hierarchical fashion (Dayan et al., 1995; Lewicki and Sejnowski, 1996). In order to do this though, non-linearities such as those mentioned above will also need to be dealt with. Simply adding another linear image model on top of the current one, using the same number of units, results merely in the identity transform being discovered (unpublished observations). Indeed, it would be surprising if something other than this happened, as it would beg the question of why the combined transformation was not discovered by the first linear stage to begin with. Adding more units in the second stage may enable the discovery of further complex structure, but this would be an unacceptable solution because it would simply result in combinatorial explosion as complex features are replicated at each and every position and scale.

9 Conclusions

When considered purely from an empirical point of view, the response properties of cortical neurons present one with a bewildering array of data that can make very little sense without a theory for interpretation. The form of theory we have attempted to offer here is based on the notion that the visual cortex is trying to produce an efficient representation, in terms of extracting the statistically independent (and hopefully, meaningful) structure in images. We drew upon our prior notions of the structure of natural images in order to propose sparse coding as a viable option for reducing statistical dependencies among elements of the representation. The receptive fields that emerge from this algorithm strongly resemble those found in the primary visual cortex, and also those that have been previously deduced by engineers to form efficient image representations. The solution is very robust, as long as some notion of sparseness is enforced, and so provides a compelling functional account of the response properties of cortical simple cells in terms of a sparse code for natural images. When the code is overcomplete, interesting forms of non-linearity arise in the input-output relationship, and these forms of interaction may be tested for experimentally. While the current theory merely sheds light on the response properties of cortical simple-cells, it is our hope that when this general approach is extended in a hierarchical fashion it may lend insights into other aspects of cortical processing, such as the response properties of neurons at higher stages of cortical processing, as well as the role of feedback.

Acknowledgement

We thank Mike Lewicki, Chris Lee, Tony Bell, George Harpur, Peter Dayan, and Federico Girosi for very useful conversations during the development of this work. Both authors were supported by grants from NIMH: F32-MH11062 (BAO) and R29-MH50588 (DJF). Part of this work was carried out at the Center for Biological and Computational Learning at MIT.

A Relation to the ICA algorithm of Bell and Sejnowski

Bell and Sejnowski (1995) describe an algorithm for "independent components analysis" (ICA) based on maximizing the mutual information between the inputs and outputs of a neural network (see also the article in this issue). Here, we show that this algorithm may be understood as solving the same maximum-likelihood problem described in Section 3, except by making a different simplifying assumption. This connection has also been shown recently by Pearlmutter & Parra (1996) and Mackay (1996).

Bell and Sejnowski examine the case where the number of basis functions is equal to the number of inputs, and where the ϕ_i are linearly independent. In this case, there is a unique set of a_i for which $|I - a\phi|^2$ equals zero for any given image, I. In terms of the previous discussion, $P(I|a,\phi)$ is now a gaussian hump with a single maximum at $a = I\phi^{-1}$, rather than a gaussian ridge as in Figure 1b. If we let σ_N go to zero in Equation 6, then $P(I|a,\phi)$ becomes like a delta function and the integral of Equation 4 becomes

$$\begin{aligned} P(I|\phi) &= \int \delta(I - a\phi) P(a) da && (24) \\ &= P(I\phi^{-1}) \times |\det \phi^{-1}| && (25) \end{aligned}$$

and so

$$\begin{aligned} \phi^* &= \arg\max_{\phi} \left[\langle \log P(I\phi^{-1}) \rangle + \log |\det \phi^{-1}| \right] && (26) \\ &= \arg\min_{\phi} \left[\langle \lambda \sum_i S((\phi^{-1})_i \cdot I) \rangle - \log |\det \phi^{-1}| \right] . && (27) \end{aligned}$$

By making the following definitions according to the convention of Bell and Sejnowski (1995),

$$\begin{aligned} \mathbf{W} &= \phi^{-1} && (28) \\ u_i &= \mathbf{W}_i \cdot I && (29) \end{aligned}$$

then, the gradient descent learning rule for $\mathbf{W}$ becomes

$$\Delta W_{ij} \propto -\lambda S'(u_i) I_j + \frac{\operatorname{cof} W_{ij}}{\det \mathbf{W}} . \quad (30)$$

This is precisely Bell and Sejnowski's learning rule when the output non-linearity of their network, $g(x)$, is equal to the cdf (cumulative density function) of the prior on the a_i, i.e.,

$$\begin{aligned} y_i &= g(u_i) && (31) \\ g(u_i) &= \int_{-\infty}^{u_i} \frac{1}{Z_\beta} e^{-\beta S(x)} dx . && (32) \end{aligned}$$

Thus, the independent component analysis algorithm of Bell and Sejnowski (1995) is formally equivalent to maximum likelihood in the case of no noise and a square system (dimensionality of output = dimensionality of input). It is easy to generalize this to the case when the number of outputs is less than the number of inputs, but not the other way around. When the number of outputs is greater than the effective dimensionality of the input (# of non-zero eigenvalues of the input covariance matrix), then the extra dimensions of the output will simply drop out. While this does not pose a problem for blind separation problems where the number of independent sources (dimensionality of a) is less than or equal to the number of mixed signals (dimensionality of I), it will become a concern in the representation of images, where overcompleteness is a desirable feature (Simoncelli et al., 1992).

References

Amari S, Cichocki A, Yang HH (1996) A new learning algorithm for blind signal separation. In: *Advances in Neural Information Processing Systems, 8*, MIT Press.

Atick JJ (1992) "Could information theory provide an ecological theory of sensory processing?" *Network, 3*: 213-251.

Atick JJ, Redlich AN (1990) "Towards a theory of early visual processing," *Neural Computation, 2*: 308-320.

Atick JJ, Redlich AN (1992) "What does the retina know about natural scenes?" *Neural Computation, 4*: 196-210.

Baddeley R (1996) "An efficient code in V1?" *Nature, 381*: 560-561.

Barlow HB (1961) "Possible principles underlying the transformations of sensory messages." In: *Sensory Communication*, W.A. Rosenblith, ed., MIT Press, pp. 217-234.

Barlow HB (1989) "Unsupervised learning," *Neural Computation, 1*: 295-311.

Baum EB, Moody J, Wilczek F (1988) "Internal representations for associative memory," *Biological Cybernetics, 59*: 217-228.

Bell AJ, Sejnowski TJ (1995) "An information-maximization approach to blind separation and blind deconvolution," *Neural Computation, 7*: 1129-1159.

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

Dan Y, Atick JJ, Reid RC (1996) "Efficient coding of natural scenes in the lateral geniculate nucleus: experimental test of a computational theory," *Journal of Neuroscience, 16*: 3351-62.

Daugman JG (1989) "Entropy reduction and decorrelation in visual coding by oriented neural receptive fields," *IEEE Transactions on Biomedical Engineering, 36*: 107-114.

Dayan P, Hinton GE, Neal RM, Zemel RS (1995) "The Helmholtz machine," *Neural Computation, 7*: 889-904.

De Valois RL, Albrecht DG, Thorell LG (1982) "Spatial frequency selectivity of cells in macaque visual cortex," *Vision Res, 22*: 545-559.

Dong DW, Atick JJ (1995) "Temporal decorrelation: a theory of lagged and non-lagged responses in the lateral geniculate nucleus," *Network: Computation in Neural Systems, 6*: 159-178.

Field DJ (1987) "Relations between the statistics of natural images and the response properties of cortical cells," *J Opt Soc Am, A, 4*: 2379-2394.

Field DJ (1993) "Scale-invariance and self-similar 'wavelet' transforms: an analysis of natural scenes and mammalian visual systems." In: *Wavelets, Fractals, and Fourier Transforms*, Farge M, Hunt J, Vascillicos C, eds, Oxford UP, pp. 151-193.

Field DJ (1994) "What is the goal of sensory coding?" *Neural Computation, 6*: 559-601.

Friedman JH (1987) "Exploratory projection pursuit," *Journal of the American Statistical Association, 82*, 249-266.

Foldiak P (1990) "Forming sparse representations by local anti-Hebbian learning," *Biol. Cybernetics, 64*: 165-170.

Foldiak P (1995) "Sparse coding in the primate cortex," In: *The Handbook of Brain Theory and Neural Networks*, Arbib MA, ed, MIT Press, pp. 895-989.

Fyfe C, Baddeley R (1995) "Finding compact and sparse- distributed representations of visual images," *Network, 6*: 333-344.

Harpur GF (1997) "Low entropy coding with unsupervised neural networks," Ph.D. Thesis, Dept. of Electrical Engineering, Cambridge University.

Harpur GF, Prager RW (1996) "Development of low entropy coding in a recurrent network," *Network, 7*: 277-284.

Hinton GE, Ghahramani Z (1997) "Generative models for discovering sparse distributed representations," *Philosophical Transactions of the Royal Society B*, (in press).

Intrator N (1992) "Feature extraction using an unsupervised neural network," *Neural Computation, 4*: 98-107.

Kullback S (1959) *Information theory and statistics*, John Wiley & Sons, New York.

Law CC, Cooper LN (1994) "Formation of receptive fields in realistic visual environments according to the Bienenstock, Cooper, and Munro (BCM) theory," *Proc Natl Acad Sci, USA, 91*: 7797-7801.

Lewicki M, Sejnowski TJ (1996) "Bayesian unsupervised learning of higher order structure," In: *Advances in Neural Information Processing Systems, 9*, MIT Press.

Lu ZL, Chubb C, Sperling G (1996) "Independence rejection: An unsupervised learning algorithm for extracting latent source structures from arbitrary image populations," *Technical Report MBS 96-15*, Institute for Mathematical Behavioral Sciences, UC Irvine.

Mallat SG (1989) "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence, 11(7)*: 674-693.

Mumford D (1994) "Neuronal architectures for pattern-theoretic problems," In: *Large Scale Neuronal Theories of the Brain*, Koch C, Davis, JL, eds., MIT Press, pp. 125-152.

Olshausen BA (1996) "Learning linear, sparse, factorial codes," AI Memo 1580, Massachusetts Institute of Technology.

Olshausen BA, Anderson CH (1994) "A model of the spatial-frequency organization in primate striate cortex," In: *The Neurobiology of Computation*, J.M. Bower, ed., Kluwer, pp. 275-280.

Olshausen BA, Field DJ (1996a) "Emergence of simple-cell receptive field properties by learning a sparse code for natural images." *Nature, 381*, 607-9.

Olshausen BA, Field DJ (1996b) "Natural image statistics and efficient coding," *Network, 7*: 333-339.

Parker AJ, Hawken MJ (1988) "Two-dimensional spatial structure of receptive fields in monkey striate cortex," *Journal of the Optical Society of America A, 5*: 598-605.

Pearlmutter BA, Parra LC (1996) A context-sensitive generalization of ICA. International Conference on Neural Information Processing, September 1996, Hong Kong.

Press WA, Lee CW (1996) "Projection pursuit analysis of the statistical structure in natural scenes," paper presented at CNS96, Cambridge, MA.

Rao RPN, Ballard DH (1997) "Dynamic model of visual recognition predicts neural response properties in the visual cortex," *Neural Computation, 9*: 721-763.

Saund E (1995) "A multiple cause mixture model for unsupervised learning," *Neural Computation, 7*: 51-71.

Simoncelli EP, Freeman WT, Adelson EH, Heeger DJ (1992) "Shiftable multiscale transforms," *IEEE Transactions on Information Theory, 38(2)*: 587-607.

Tadmor Y, Tolhurst DJ (1989) "The effect of threshold on the relationship between the receptive field profile and the spatial-frequency tuning curve in simple cells of the cat's striate cortex," *Visual Neuroscience, 3*: 445-454.

Zemel RS (1993) "A minimum description length framework for unsupervised learning," Ph.D. Thesis, University of Toronto, Dept. of Computer Science.

Figures

Figure 1: Sparse coding. *a*, An image is represented by a small number of "active" code elements, a_i, out of a large set. Which elements are active varies from one image to the next. *b*, Since a given element in a sparse code will most of the time be inactive, the probability distribution of its activity will be highly peaked around zero with heavy tails. This is in contrast to a code where the probability distribution of activity is spread more evenly among a range of values (such as a Gaussian).

Figure 2: In a generative image model, one attempts to capture the underlying causes of images. In this case, images are assumed to be composed of a linear superposition of basis functions, $\phi_i(\vec{x})$, mixed together with amplitudes a_i. The goal of efficient coding is to learn the basis functions that can best account for the structure in images in terms of statistically independent events.

Figure 3: Two-dimensional iso-probability plots of *a*, Gaussian likelihood, *b*, Cauchy prior, and *c*, their product. The axes on each plot are a_1, a_2.

Figure 5: A network implementation for computing the a_i. Each output unit represents the value of a single coefficient, a_i. The output activities are fed back through the basis functions, ϕ_i, to form a reconstruction of the image. The reconstructed image is then subtracted from the input image, and the residual image is fed forward through the ϕ_i to drive each output, a_i, which is also being self-inhibited by S'. This process is analogous to the analysis-synthesis loop proposed by Mumford (1994) for performing inference on images. Learning is accomplished by doing a Hebbian update of the ϕ_i based on the average joint activity between the outputs (a_i) and the residual image computed via the negative feedback connections.

Figure 4: The effect of the sparseness cost function on the input-output relationship of each unit. *a*, The sparseness cost function, $S(x) = log(1 + x^2)$. *b*, The derivative of the sparseness cost function utilized in gradient descent, S'. The effect of S' will be to differentially suppress values near zero.

Figure 6: Illustration of the steps taken in preprocessing. *a*, A hypothetical distribution being "sphered" so that the variance in all directions is equal. *b*, The energy present in the corners of the 2D Fourier plane is an artifact of rectangular sampling. Lowpass filtering within a circle in frequency space eliminates these artifacts. *c*, The profile of the combined lowpass/whitening filter, $R(f) = f\, e^{-(f/f_0)^4}$, in spatial frequency and space (the inverse Fourier transform (IFT) assumes zero phase). Atick and Redlich (1992) have shown that such a whitening filter corresponds well to the response properties of retinal ganglion cells.

Figure 7: The set of 144 basis functions learned by the sparse coding algorithm. The basis functions are totally overlapping (i.e., the entire set codes for the same

image patch). All have been normalized to fill the grey scale, but with zero always represented by the same grey level.

Figure 8: The distribution of the 144 basis functions in spatial-frequency. Each basis function was fit with a Gabor function, and the spatial-frequency underlying the Gabor function was plotted in the upper-half of the f_x, f_y plane.

Figure 9: Outputs computed in response to a given input image, according to 1) the feedforward activation only (b_i), 2) the sparsified a_i, 3) sparsified a_i for randomly chosen bases, and 4) sparsified a_i for pixel bases. At right are shown the corresponding reconstructions (not applicable for the b_i).

Figure 10: The results of mapping the response profile of a given unit with bars (*top*) and gratings (*bottom*). At the right is shown a Hilbert transform pair of spatial profiles recovered from the inverse Fourier transform of the frequency response, assuming zero phase. Note that these show more ringing, due to the sharper cutoff in frequency response incurred by sparsification.

Figure 11: An example scenario of two basis function that overlap. Both respond in a strictly linear feedforward network, but in the sparse coding network only the function that best describes the stimulus responds.

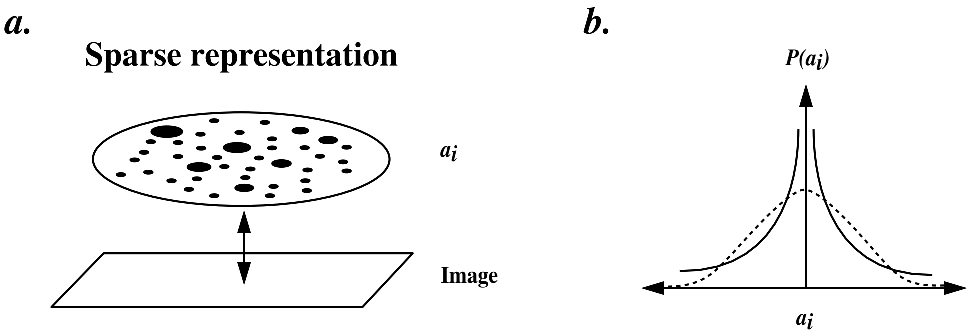

Figure 1:

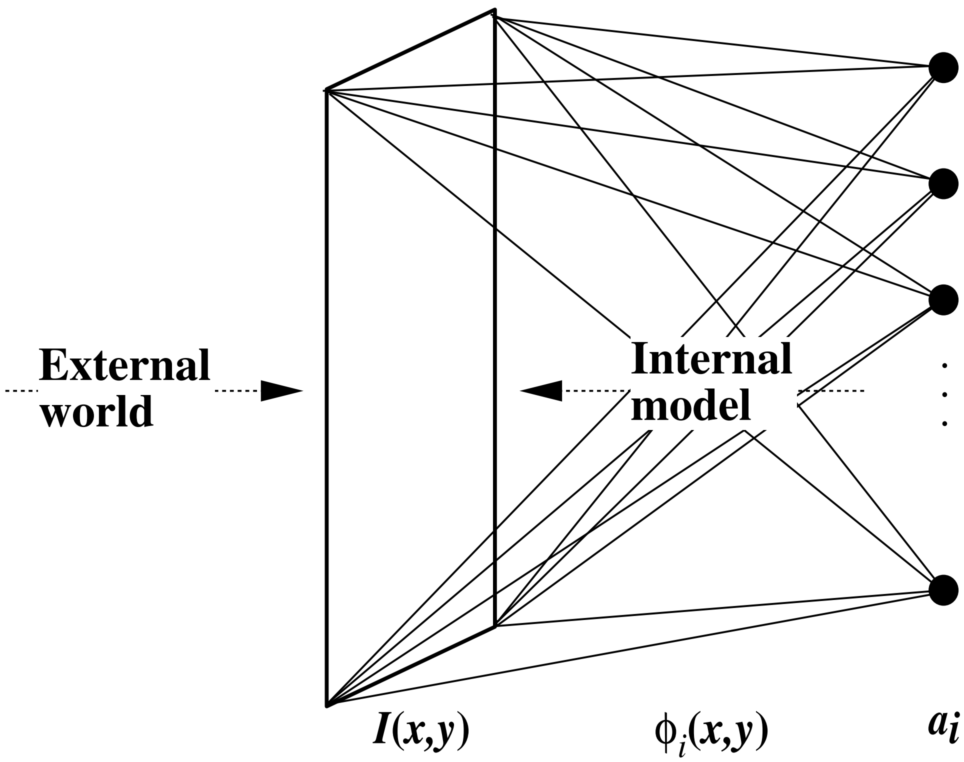

Figure 2:

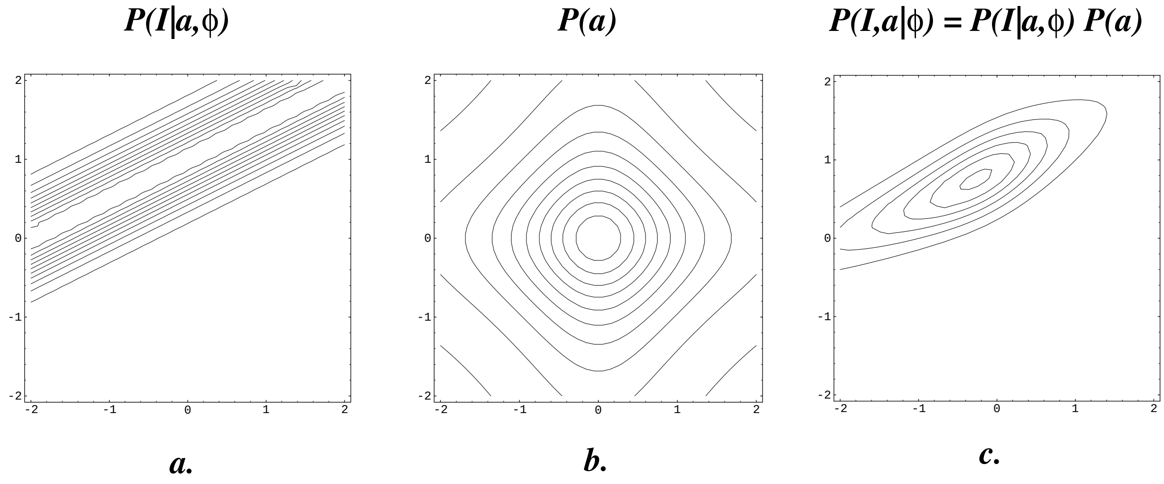

Figure 3:

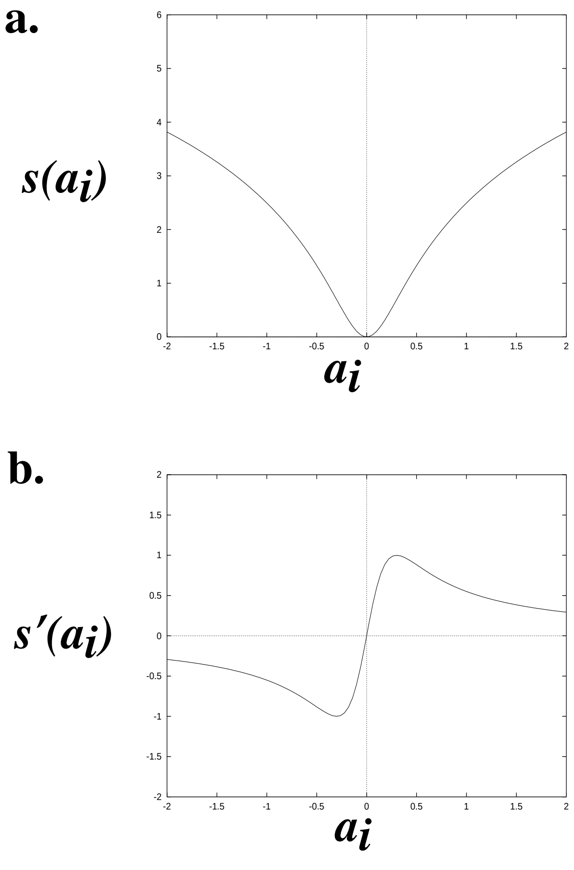

Figure 4:

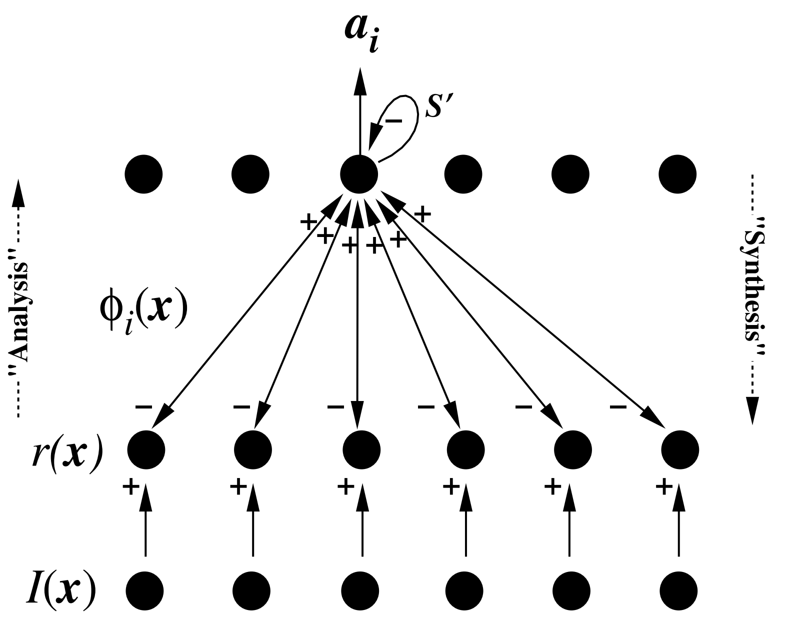

Figure 5:

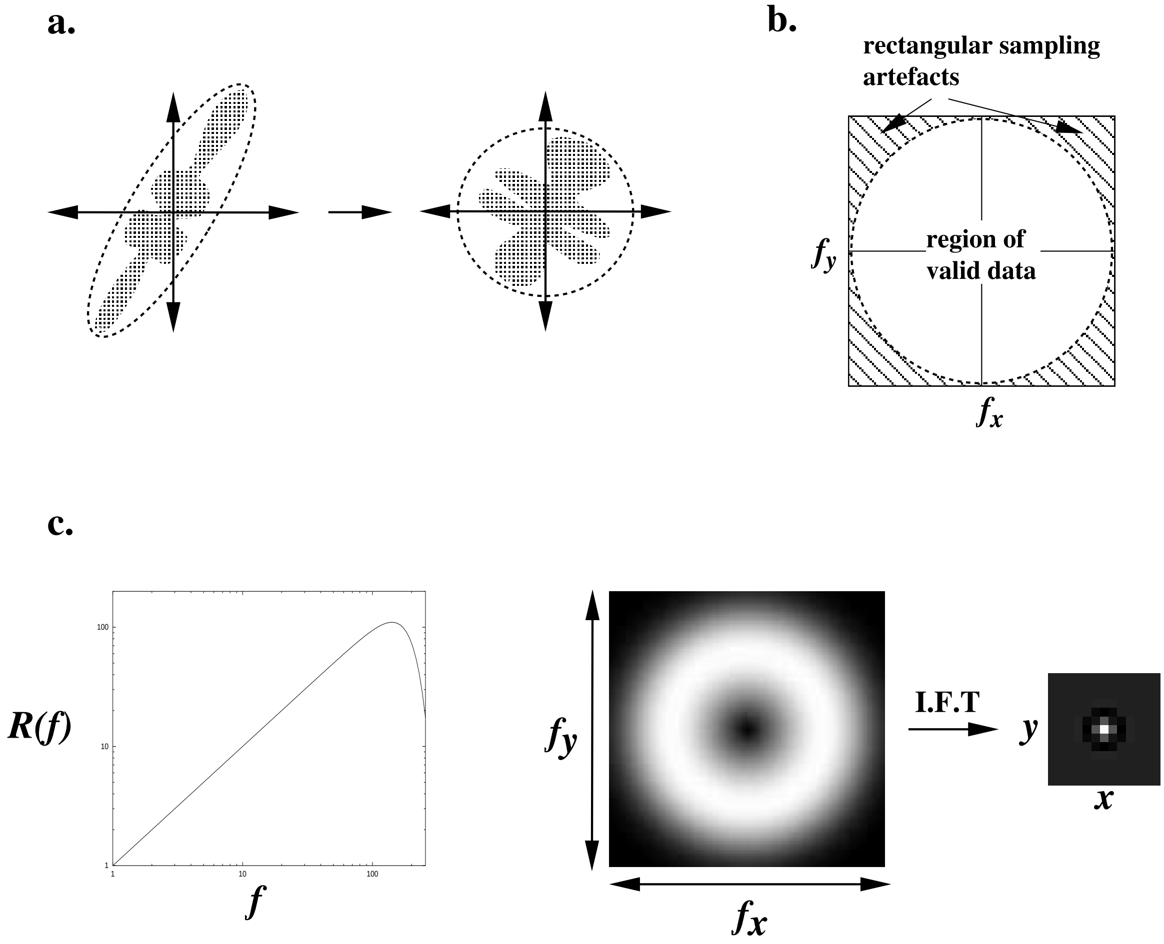
a.
b.
rectangular sampling artefacts
region of valid data
f_y
f_x
c.
$R(f)$
f
100
10
1
1
10
100
f_y
f_x
I.F.T
y
x

Figure 6:

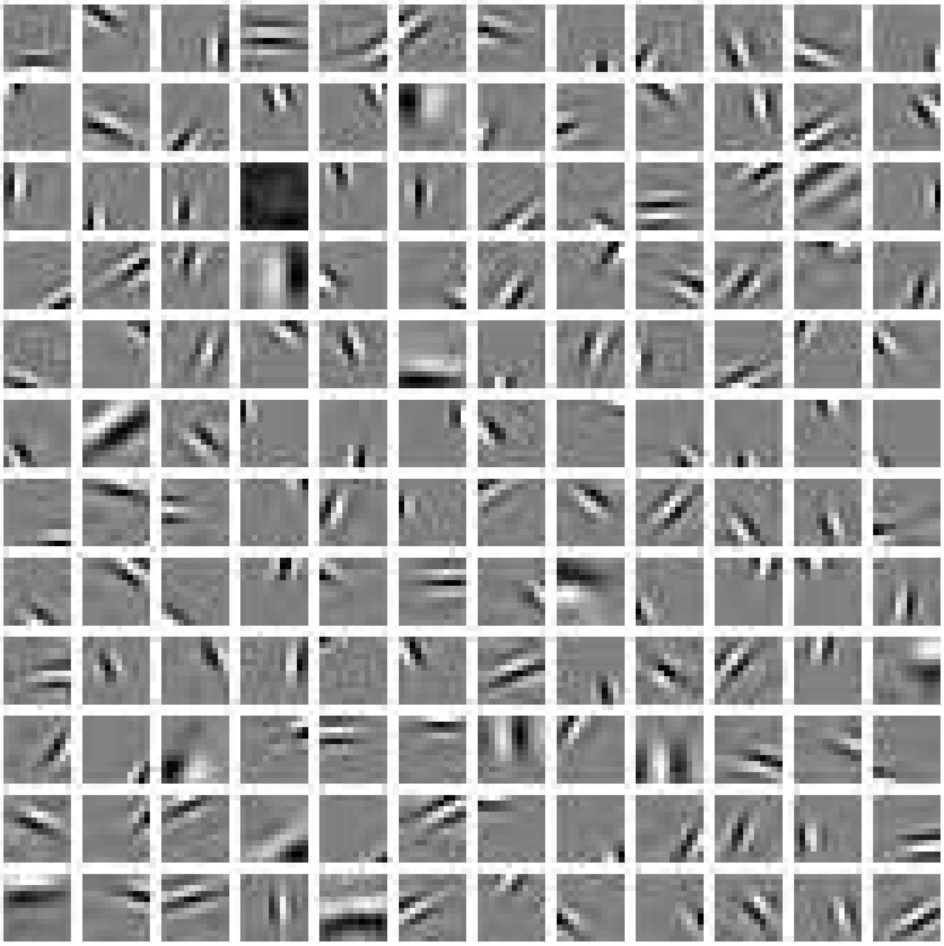

Figure 7:

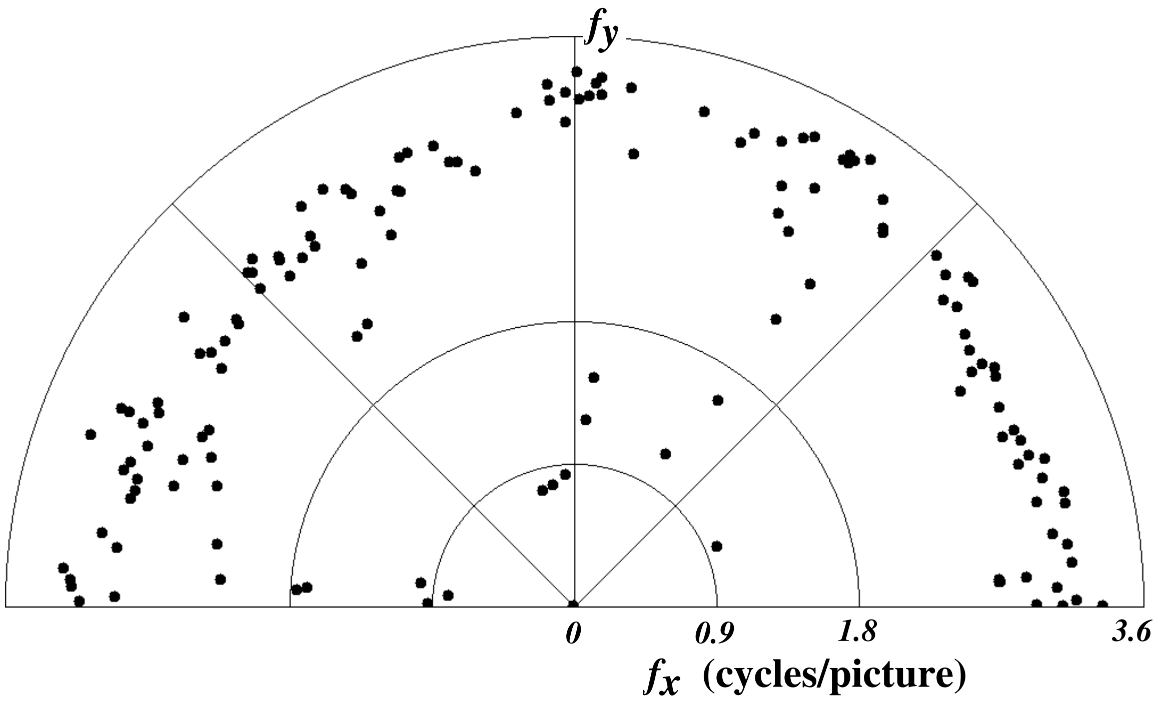

Figure 8:

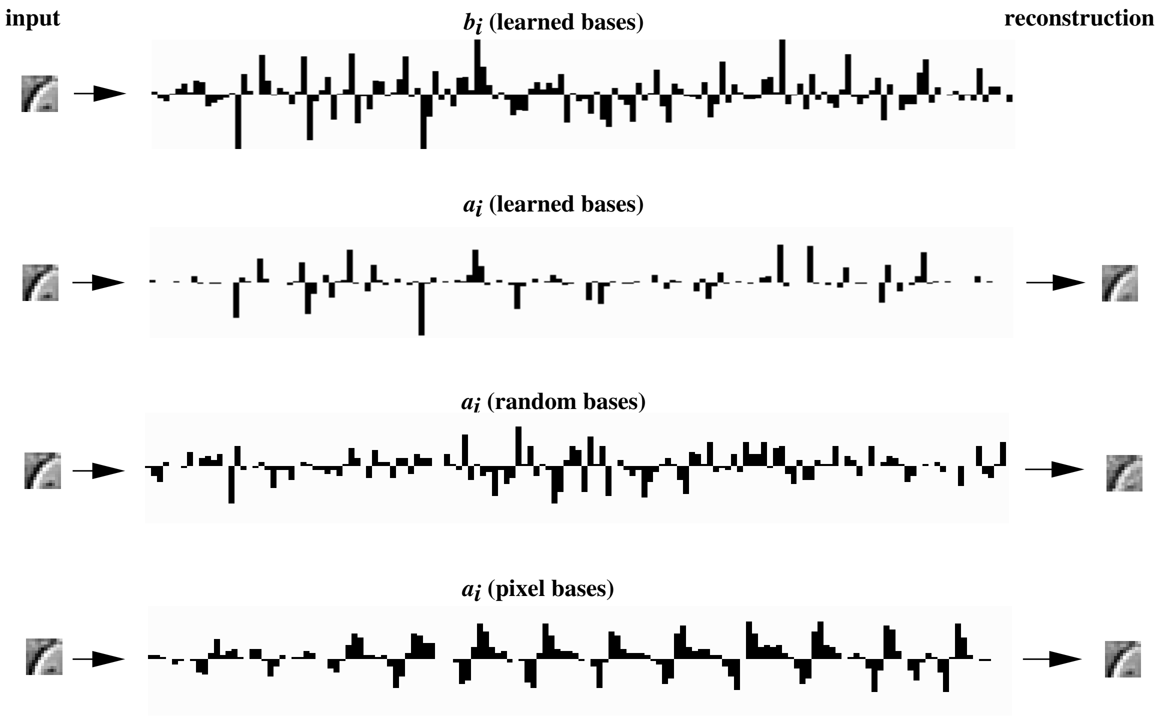

Figure 9:

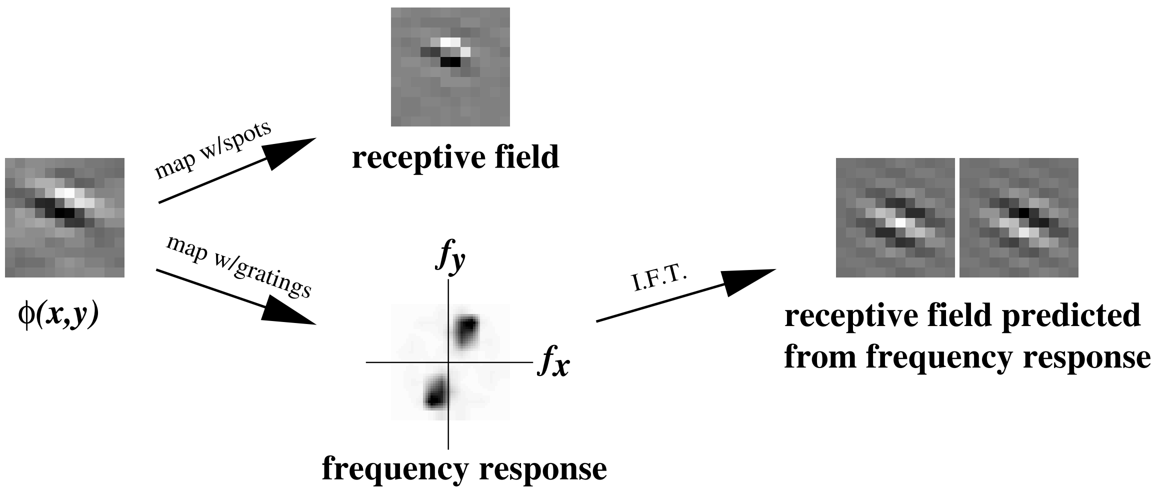

Figure 10:

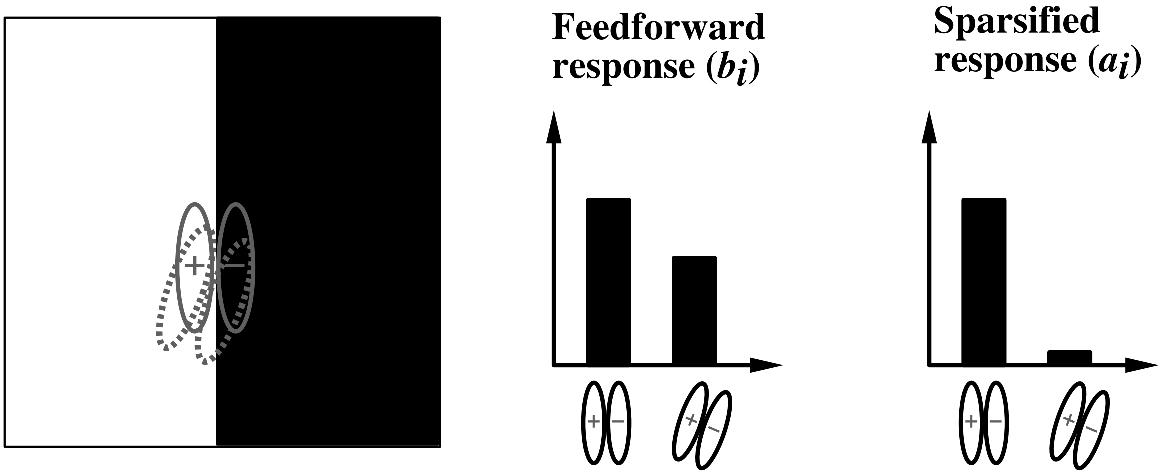

Figure 11: