COMPUTER VISION, GRAPHICS, AND IMAGE PROCESSING 39, 311–327 (1987)

The Cortex Transform: Rapid Computation of Simulated Neural Images

ANDREW B. WATSON

Perception and Cognition Group, NASA Ames Research Center, Moffett Field, California 94035

Received April 10, 1986; accepted November 6, 1986

This paper describes a transform which maps an image into a set of images that vary in resolution and orientation. Each pixel in the output may be regarded as the simulated response of a neuron in human visual cortex. The transform is invertible through simple expansion and addition operations. The transform is designed as a tool for image processing, machine vision, and modeling human vision. © 1987 Academic Press, Inc.

1. BACKGROUND

This paper describes an image transform that may prove useful in image and vision research. It is an outgrowth of several developments in diverse areas of vision science, and the purpose of this first section is to review those developments.

1.1. Cellular Model of Biological Visual Processing

A view which dominates the current understanding of biological vision is that image data, once captured by the receptors, pass through numerous layers of cells, each of which, by means of its pattern of inputs and its particular input/output rules, imposes some transformation on the data. The traditional route to understanding this system has been to analyze the response properties of individual cells and arrays of such cells at various levels in the visual pathway.

1.2. Mathematical Analysis of Visual Cells

Traditionally, each visual neuron has been characterized by its *receptive field* (RF), the region of the retina or visual field within which light will cause some change in the response of the cell [1]. If the response to light is more or less linear, the cell may be further characterized by specifying its receptive field *weighting function*, which describes the weighted contribution of light at each point in the receptive field to the response of the cell [2]. The total response of the cell can be computed by cross correlation of the weighting function and the input image. A mathematical alternative is to convolve the image with an *impulse response* (equal to the weighting function reflected about each axis) and then to sample the resulting output (a *neural image*) [3] at the location of the receptive field. To determine the response of a large array of cells with identical receptive fields, the neural image is sampled at the cell locations. An important point is that, apart from the final sampling, the neural image reflects all the processing done by one type of visual cell.

The convolution that transforms the image into a neural image may also be expressed and performed in the frequency domain. The Fourier transform of the image is multiplied by the Fourier transform of the impulse response, and the result is inverse transformed to yield the neural image. Depending upon the form of the impulse response and the computer hardware available, this may be orders of magnitude more efficient than direct convolution.

0734-189X/87 $3.00

To model these events on a computer, the images, weighting functions, and neural images usually must be expressed in discrete form, as finite rectangular arrays of discrete values, obtained by sampling from the continuous theoretical functions. This presents no problem, provided that the proper sampling requirements are observed and that the consequences of cyclic discrete convolution are understood. However, it does raise the problem that the cell location may fall between samples of the neural image. In that event the sample value must be obtained by appropriate interpolation among the available samples of the neural image.

With the preceding remarks in mind, it is now possible to treat the action of linear visual neurons with the conventional technology of digital image processing.

1.3. Cortical Cells

Electrophysiological and anatomical data provide a rough sketch of the image transformations performed by the cells in the major nodes in the visual pathway: the retina, the dorsal lateral geniculate nucleus (LGN), and the visual cortex. Lennie [4] and Shapley and Lennie [5] provide excellent reviews of many of the essential features of cells at these various levels.

One brain area of particular interest is the portion of visual cortex known as V1, which evidently in the monkey (and probably in man) receives essentially all of the cortical projection from the retina (via the LGN). In this area, on the order of half of the cells (the so-called *simple* cells) appear to do an approximately linear integration over their receptive fields, though their output typically undergoes nonlinearities such as thresholding, compression, saturation, and halfwave rectification (see [6–7] for more extensive discussions of linearity). These cells are remarkably homogeneous in one respect: they have receptive fields composed of several parallel elongated excitatory and inhibitory regions, as illustrated in an idealized way in Fig. 1. Each receptive field can be characterized by a (radial) spatial frequency, corresponding to the inverse of the distance between bright bars, and by an orientation, defined by the angle of the orthogonal to the bars. These two quantities can also be expressed as a particular two-dimensional (2D) spatial frequency. The cell will respond to a range of frequencies and orientations about its center values, and this range can be characterized by spatial frequency and orientation bandwidths (or by a 2D frequency bandwidth). In general, it has been found that spatial frequency bandwidths are almost proportional to center frequency, so that they are commonly expressed in the logarithmic unit of octaves. This proportionality reflects the fact that the overall size of the RF tends to be inversely proportional to spatial frequency, as though each RF had approximately a constant number of stripes. In a population of 358 V1 cells, De Valois *et al.* found a median spatial frequency bandwidth (measured at optimal orientation) of about 1.4 octaves [8]. Other studies have generally corroborated these estimates [9, 10]. The median orientation bandwidth of V1 cells is about 40°, with a rather broad distribution about this value [11].

An important point is that, viewed in the 2D frequency domain, each cell RF occupies a patch that is much smaller than the total frequency region to which the ensemble of cells, or the intact monkey, is sensitive. This is so even when the cells are drawn from a small region of the visual field [5]. For example, the total 1D spatial frequency bandwidth of the system, expressed in terms of the range of center frequencies of individual cells within 1.5° of the fovea, is at least 5 octaves [8]. This

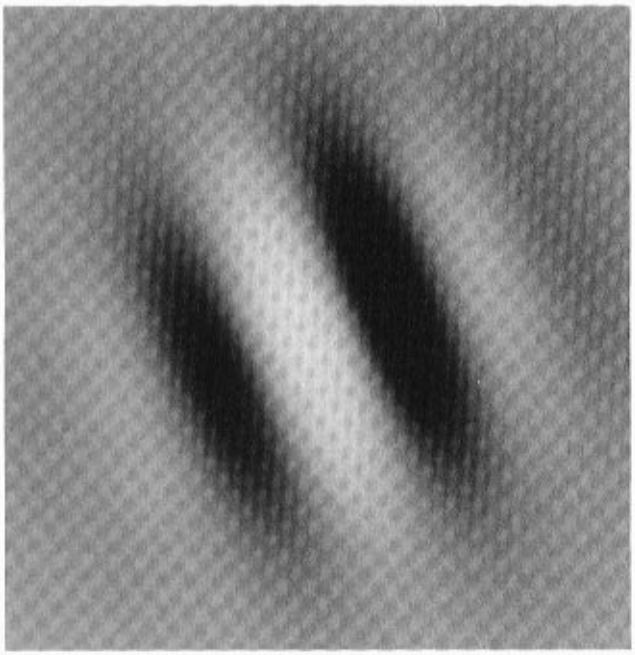

FIG. 1. An idealized cortical receptive field. The actual waveform is a 2D Gabor function.

observation has reinforced the idea, inspired originally by psychophysical data [12, 13], of multiple, parallel, band-pass neural images or "multiple channels."

The idealized RF in Fig. 1 is in fact a 2D Gabor function (the product of a 2D sinusoid and a 2D Gaussian). This function has been found to provide a good, albeit preliminary, description of V1 simple cell receptive fields [14–16, 10, 17].

1.4. Computational Models of Human Vision

There have been a number of recent attempts to model early human visual processing [18–23]. While the specific goals in these projects differ, all involve as an initial step the transformation of the image into an internal representation. Because they make up a large fraction of the cells in the primary visual pathway, because their acuity approximates that of the human observer, because they resemble the "channels" discovered psychophysically, and because they are relatively well understood, simple cortical cells have served as the basis for many of these models. One goal of the cortex transform is to provide a rapid means of generating the responses of a large array of model cells, in order to speed the execution of these models.

1.5. Image Processing

For a number of image processing and image understanding applications it has been found useful to decompose an image into a number of low-pass or band-pass subimages. The resulting subimages may be subsampled in proportion to their resolution, resulting in an image "pyramid." Pyramids differ in the algorithm used to generate each subimage "layer," but all seek to reduce the number of pixels at successive layers [24–27]. A useful survey of this subject is provided by Rosenfeld [28].

The pyramid structure is useful for a number of reasons. It permits "scale invariance" in the sense that algorithms can be executed in parallel at several scales. It also allows "coarse-to-fine" searches, in which a search is first performed on a small number of samples of low resolution data. And it may allow one to match the image quantization to the frequency-dependent contrast discrimination of the human observer.

The transform to be constructed here has this multiresolution property and may enjoy similar advantages. It has an additional feature, of as yet undetermined value, of segregating the various orientations, as well as resolutions.

1.6. Summary

In a number of diverse areas, such as simulation of visual physiology, modeling early human vision, and image processing and understanding, there is a need for a transform which maps an image into a set of images with modest bandwidths in both spatial frequency and orientation.

2. DESIGN GOALS

We have two criteria for the construction of the transform. First, it must resemble the action of primate visual cortex. Second, it must be computationally convenient. Here we enumerate some specific desirable properties that fall under one or the other criterion.

2.1. Approximate Gabor Shape

As noted above, a large fraction of cells in the primary visual cortex have receptive fields that resemble a 2D Gabor function. Accordingly, we would like the transform to generate responses of sensors with approximately 2D Gabor shape. This can be done directly by way of a 2D Gabor transform, but this is generally difficult to compute and even more difficult to reconstruct, and it lacks several of the other desirable features noted below [29].

2.2. Setable Frequency Bandwidth

The observations above on cortical cell bandwidth suggest it would be desirable for the transform to be capable of generating subimages of about one octave bandwidth, but to permit other bandwidths to be tried as well.

2.3. Setable Orientation Bandwidth

Similar considerations apply to the orientation bandwidth. We would like an orientation bandwidth of about 40°, but with the ability to adjust this bandwidth within a broad range.

2.4. Scale Invariance

By scale invariance we mean that if the input is reduced in size, then the output will simply be shifted to a new layer of the pyramid. This property is achieved if the processing at each resolution is identical to that at other resolutions except for a change of scale.

2.5. Reconstruction by Addition

A transform that can be inverted (from which one can reconstruct the original) demonstrably preserves all of the information in the image. Such transforms are of

great use in image processing because while they may rearrange the data into a more convenient form they do not lose any information. It may be argued that an invertible transform is an inappropriate structure with which to model human vision, since we certainly lose information. But the processes by which we lose information early in the visual system, notably by noise and by relative attenuation of various spatial frequencies, can be modeled as separate, simpler processes [30]. Furthermore, preservation of information in the transform allows one to experiment selectively with information losses in the various layers.

A particularly simple method of reconstruction is to simply add together the layers generated by the transform. Clearly this method will work only if the layers are generated through a process of subtraction. The process may be likened to a jigsaw puzzle: if the pieces are all to fit back together they must have all been cut from the same original. The cortex transform is designed to allow this property of reconstruction through addition.

2.6. Pyramid Structure

As noted above, pyramid image structures have a number of useful features, and we will therefore seek a pyramid structure for our transform. Pyramids typically segregate information at different resolutions into separate layers. Our transform will do this but will also segregate different orientations into different layers. Thus we are in effect creating several pyramids, one per selected orientation.

2.7. Fast Algorithm

A major goal of this exercise is to accelerate the process of developing and testing models of human vision. Part of this acceleration is provided by the use of an image transform, rather than direct computation of individual sensor responses. But that is largely a matter of more efficient notation, rather than a change in the actual algorithm. However, as we shall see, the particular transform we develop is amenable to a number of shortcuts that greatly reduce the amount of computation.

3. BUILDING THE CORTEX TRANSFORM

As noted above, visual filtering may be represented in either space or frequency domains. In what follows we will work almost exclusively in the frequency domain, because it allows a much simpler analysis. Consider an image of N by N pixels (we will deal only with square images, though the principles easily extend to rectangles). The frequency representation of this image is an array of N by N complex pixels. When the imaginary part of each pixel is zero (or when only the magnitude of each pixel is preserved) then each pixel is real and the DFT can be displayed as an image. There will be several examples of these DFT images below. In fact, for a real image, the DFT is Hermite and consequently only half its values are unique. But we will ignore this redundancy in our pictures of DFTs.

Recalling the analogy made earlier, the requirement of reconstruction-through-addition suggests that we carve up the frequency domain like a puzzle, each "piece" forming the frequency response of one layer of the pyramid. As we shall see, the puzzle analogy must be extended somewhat. The cardboard to be cut must be thought of as thick, and the cuts need not be orthogonal to the surface. Perhaps a better analogy is to cutting a pie, twisting the knife as one goes.

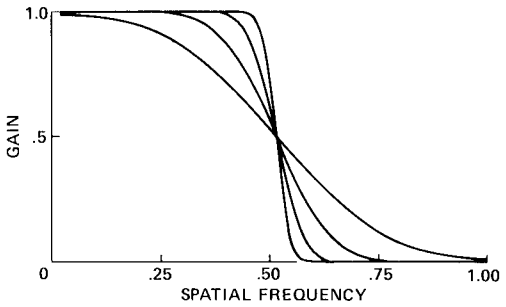

FIG. 2. Shape of the mesa filter as a function of several values of the sharpness parameter $\gamma(2, 4, 8, 16)$.

3.1. Mesa Filter

The first step is to construct a low-pass filter with unit gain within the pass-band, and Gaussian fall-off beyond some corner frequency, f, which we define as the frequency at which gain has fallen to $\frac{1}{2}$. This filter, which will look like a blurry disc in the frequency domain, is created by convolving a cylinder of radius f centered at the origin, with a Gaussian,

$$\tilde{m}_0(u, v) = (\gamma/f)^2 \exp\left[-\pi(r\gamma/f)^2\right] * \Pi(r/2f), \tag{1}$$

where $r = (u^2 + v^2)^{1/2}$, and where Π is a rectangular pulse of unit height and width centered at the origin. The Gaussian has the effect of blurring the edges of the disk by a controllable amount. This control is exerted by a *sharpness* parameter γ. The spread of the Gaussian is proportional to the corner frequency, divided by the sharpness parameter. A number of representative values of the parameter γ are illustrated in one-dimensional slices through the filter in Fig. 2. In the digital implementation of this filter we would like this filter to pass as much of the image as possible and thus to have as high a corner frequency as possible. Against this must be placed a desire for modest truncation of the Gaussian taper. To balance these concerns we introduce a parameter β, which is the ratio of the corner frequency f to the highest frequency in the image transform (the Nyquist frequency). If the image width is N pixels, the Nyquist frequency is $N/2$ cycles/width, and the corner frequency will be

$$f = \beta N/2. \tag{2}$$

In most of the examples we have used a value of $\beta = 0.9$.

Though we use it here only as a stepping stone, the mesa filter has some useful properties in its own right. It has a simple parametric form with variable control of the corner frequency and the sharpness of the cutoff. Like the Gaussian, it generates a low-pass image that exhibits less ringing than that generated by a more sharp-edged filter. It suffers relative to a simple Gaussian in not being separable in horizontal and vertical dimensions (which means that it will generally be more expensive to compute) but is superior in allowing a larger region of near-unit gain. Inverse

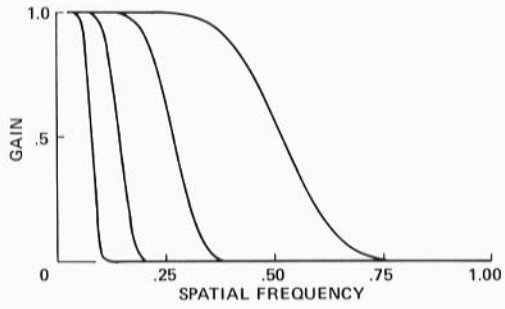

FIG. 3. 1D cross section of scaled mesa filters. Parameters, $\beta = 0.5$, $\gamma = 4$, $s = 2$.

transformation of Eq. (2) shows that the mesa filter has an impulse response that is the product of a Gaussian and a Bessel function.

3.2. Scaled Mesa Filters

Next we generate a new mesa filter, $\tilde{m}_1$, by shrinking the original by a scale factor of s. In what follows, we will generally take $s = 2$, but it may be set to some other value. The result will be another blurry disk, with a corner frequency a factor of 2 lower and with a cutoff twice as sharp. Shrinking by another factor of s will yield $\tilde{m}_2$, and so on. Thus

$$\tilde{m}_k(u, v) = \tilde{m}_0(s^k u, s^k v). \tag{3}$$

A set of scaled mesa filters generated in this way are shown in a 1D cross section in Fig. 3 and in a 2D form in Fig. 4.

3.3. Difference-of-Mesa (Dom) Filter

The difference-of-mesa (dom) filter is constructed by subtracting the smaller disk $\tilde{m}_1$ from the larger $\tilde{m}_0$. The result, pictured in the cross section in Fig. 5, has upper and lower corner frequencies of f and sf. When $s = 2$, this bandwidth is precisely one octave.

One of our goals is to generate filters with approximately Gaussian pass-bands, and if we examine the 1D cross section of the dom filter we see that the dom filter does indeed have a roughly Gaussian shape. Figure 6 shows the effect of γ upon the shape of the dom filter when $s = 2$. For values of about 3 and above, the bandwidth remains constant at 1 octave. For values above 4, the top begins to flatten and depart more severely from an approximately Gaussian shape.

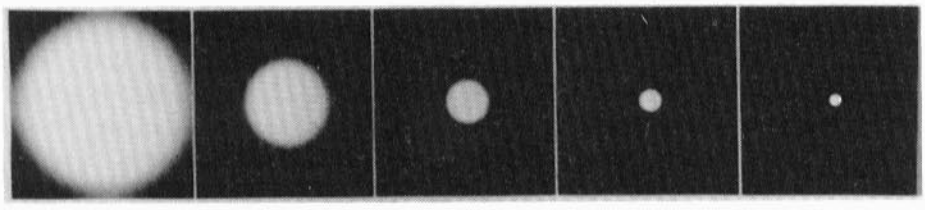

FIG. 4. Scaled mesa filters. Parameters are $\beta = 0.9$, $\gamma = 4$, $s = 2$.

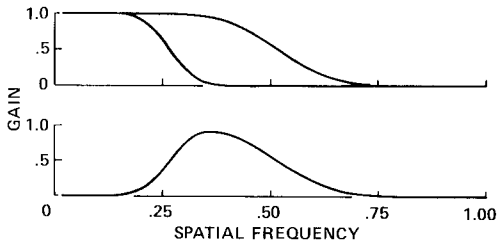

FIG. 5. Cross section of the difference-of-mesa filter: (a) mesa filters $\tilde{m}_0$ and $\tilde{m}_1$; (b) difference of the two filters in (a). Parameters, $\beta = 0.5$, $\gamma = 4$, $s = 2$.

By subtracting each adjacent pair of mesa filters we can generate a sequence of progressively smaller dom filters. The frequency spectra of these filters are shown in Fig. 7. Because the low-frequency wall of one filter forms the high-frequency wall of the next smaller filter, the dom filters fit one inside the other, like a sequence of progressively smaller donuts, to almost completely fill the frequency plane. The "almost" arises because there remains a low-frequency residue (the "hole" of the smallest donut, which is simply a very low-pass mesa filter) and a high-pass residue, which consists of the corners that remain when the largest donut is removed. The filters corresponding to these two residues are also shown in Fig. 7.

To formalize the preceding, let $\tilde{d}$ be the frequency domain representation of the dom filter. Let k index the resolution (corner frequency) of the filter. Let the largest (highest pass) filter have the index 0. Then

$$\tilde{d}_k(u, v) = \tilde{m}_k(u, v) - \tilde{m}_{k+1}(u, v). \tag{4}$$

The set of dom filters provide the desired partitioning of the frequency domain into a set of self-similar, band-limited regions, each of approximately Gaussian cross section, which when added together reconstitute the complete frequency plane. But they do not provide tuning for orientation. That is left to the next step.

3.4. Bisection Filter

Here we develop, as an intermediate tool, a filter which bisects the frequency space. We begin with a 2D step function whose edge runs along the horizontal

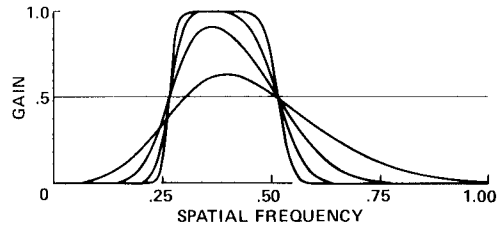

FIG. 6. 1D cross section of dom filters with various values of the sharpness parameter $\gamma(2, 4, 8, 16)$, $s = 2$.

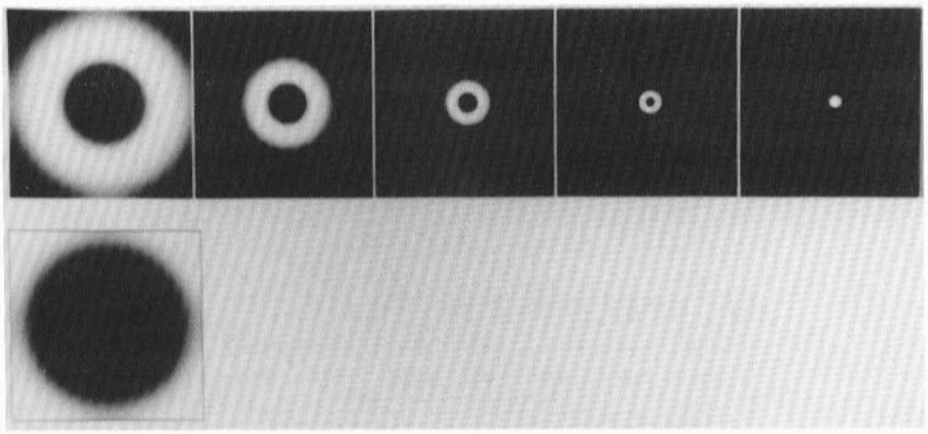

FIG. 7. Frequency spectra of scaled dom filters. Parameters, $\beta = 0.9$, $\gamma = 4$, $s = 2$. The high- and low-pass residues are also shown at the lower left and far right, respectively.

frequency axis. This can be written step(v). We convolve this step with a scaled Gaussian,

$$\tilde{b}(u, v) = \text{step}(v) * \omega \exp[-\pi\omega^2 v^2]. \tag{5}$$

This convolution with a Gaussian is the counterpart to the previous Gaussian convolution of the disk. It ensures that the edges of the filter will have Gaussian taper. Here we use a new sharpness parameter ω whose value will be discussed later. Note that we need only convolve with a 1D Gaussian in the v dimension, as the edge has no variation in the u dimension.

Convolution with an edge or step function is equivalent to integration, hence we can rewrite the expression for the blurry edge as a cumulative Gaussian,

$$\tilde{b}(u, v) = \int_{-\infty}^{v} \omega \exp(-\pi\omega^2 r^2)\, dr = \text{cum}(\omega v). \tag{6}$$

This filter bisects along the u axis, passing the upper half of the frequency plane. The corresponding expression for a blurry edge rotated by an angle α is

$$\tilde{b}_\alpha(u, v) = \text{cum}[\omega(v \cos\alpha - u \sin\alpha)], \tag{7}$$

where cum( ) is the cumulative Gaussian as written above.

3.5. Fan Filter

To discuss the selective filtering of bands of orientation we return again to the complete frequency image. Each pixel within this image specifies the complex amplitude of a sinusoid in a corresponding spatial image. The (radial) spatial frequency of the sinusoid is given by the distance of the pixel from the origin, and the orientation by the angle the pixel forms relative to the positive horizontal axis. To select a band of orientations we pass a "fan" shaped region in the frequency image, as shown in the bottom row of Fig. 8.

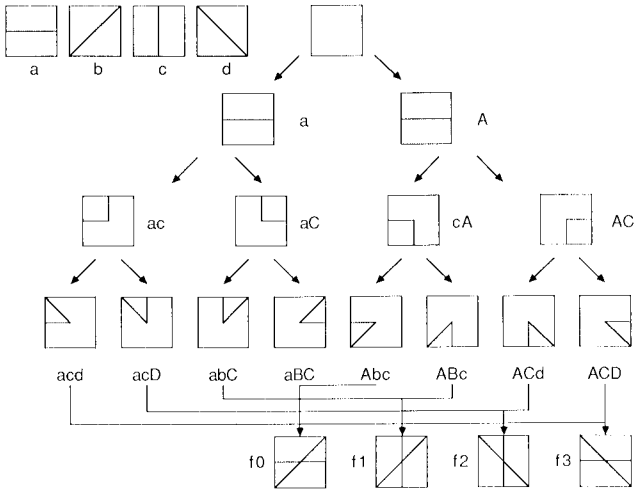

FIG. 8. Construction of a set of four fan filters from rotated bisection filters. Each box represents a frequency image with zero frequency in the center. Hatched regions have a value of one and unhatched regions, a value of zero. The border between hatched and unhatched regions represents the Gaussian transition. The images labeled a, b, c, and d are four bisection filters at angles of 0, 45, 90, and 135 degrees. Repeated bisections of the frequency image (via multiplication by bisection filters), yield eight wedges, which are combined in pairs to form the fan filters. These four fans sum to one at all points in the image. Capital letters indicate complements ($A = 1 - a$).

We create the fans by repeated bisection of the frequency space. This guarantees that the fans will sum to one at all points in the frequency space. This is illustrated in Fig. 8. The bisection filters that will be used, oriented at angles of 0, 45, 90, and 135 degrees, are indicated by the labels a, b, c, and d. Each of these also has a complement, which for brevity we represent by the corresponding capital letter ($A = 1 - a$). The frequency plane is first bisected horizontally, then vertically, then at an angle of 45° or 135°. The resulting wedges are then added together in pairs to create a set of four fan filters. It is clear from the method of construction (or from algebraic simplification), that these four filters will sum to one at all points in the frequency image.

In the implementation discussed here, we have used an orientation bandwidth of 45° ($\Omega = 4$). This agrees approximately with estimates obtained for single visual neurons in a monkey [11]. The first cut is placed arbitrarily at an angle of 0°, and successive cuts at increments of 45°.

This method is easily generalized to any number of orientations that is a power of two (in this case, 4). Other, more general methods of partitioning the space into fan filters are possible, but we have not discovered a simple one which naturally sums to one at all points. For example, Ω fan filters or orientation bandwidth $\Delta = \pi/\Omega$ are generated from the rule

$$\tilde{f}_i = \tilde{b}_{i\Delta}\left(1 - \tilde{b}_{(i+1)\Delta}\right) + \tilde{b}_{(i+\Omega)\Delta}\left(1 - \tilde{b}_{(i+\Omega+1)\Delta}\right) \qquad i = 0, \ldots, \Omega - 1. \tag{8}$$

But these sum approximately to one only where the Gaussians associated with the

FIG. 9. Construction of a Cortex filter: (a) dom filter; (b) fan filter; (c) cortex filter created by multiplying the dom and fan filters. Parameters, $\beta = 0.9$, $\gamma = 4$, $s = 2$.

bisectors do not overlap. This is not so at the origin, where there will be a prominence of amplitude $\Omega/2$. In practice this may not be a problem since the origin is removed later by multiplication by a dom filter.

3.6. A Digression

The transform of a real image is *Hermite*, which means that it has conjugate symmetry about the origin. The transforms we are dealing with are all real, so this means they must be symmetrical about the origin in order for their image to be real. The blurry edge does not have this property, since symmetric points in the bisected frequency plane, rather than being equal, have values of 0 and 1. However, the fan created from two edges does have the required symmetry, and its corresponding image is real.

3.7. Cortex Filter

The final step is to combine the spatial frequency selectivity provided by the dom filter and the orientation selectivity provided by the fan filter. This is done by multiplying the two filters, as shown in Fig. 9.

As intended, this filter passes a small band of 2D spatial frequency. The orientation and spatial frequency bandwidths are set here to values of 45° and one octave, but as noted can be set to other values. The pass-band has Gaussian taper in all directions, being sharper towards the lower frequencies. The overall sharpness of the taper can be set by the parameter γ. The orientation sharpness parameter ω is set midway between the sharpness of the inner and outer tapers of the corresponding dom filter. For example, for the level 0 cortex filter, $\omega = (\gamma + s\gamma)/2$.

The cortex filter can be inverse transformed to yield the impulse response, shown in Fig. 10. As was our goal, it resembles the 2D Gabor impulse response pictured in Fig. 1.

3.8. Filter Pyramid

For each of the dom filters created earlier, a new set of fan filters with the appropriate blur can be created and used to generate a set of cortex filters. The complete set generated in this fashion are shown in Fig. 11. The filters are arranged by frequency (rows) and by orientation (columns). Note that the lower the frequency, the smaller the portion of the frequency plane occupied by the filter, and hence the smaller the size of the frequency image required to define the filter. In fact, the 1D size is in exact proportion to the resolution of each filter; thus the sizes decrease by a factor of s from row to row. Thus the filters themselves have a (multiple) pyramid

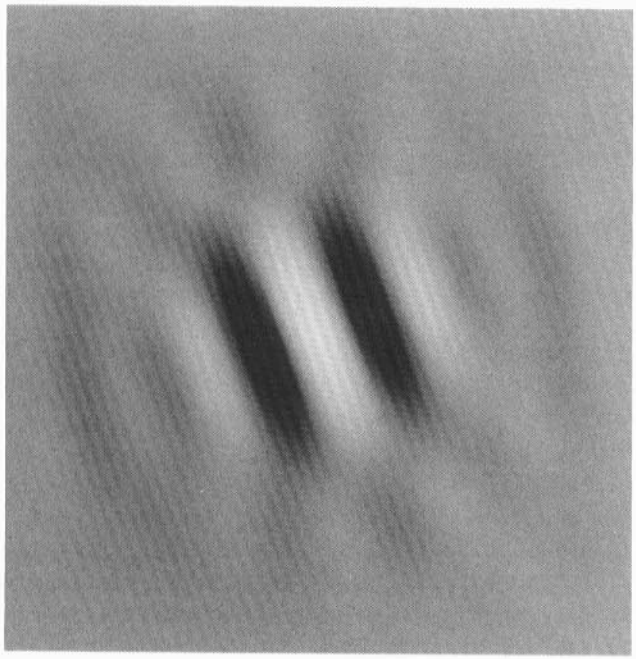

FIG. 10. Impulse response of a cortex filter.

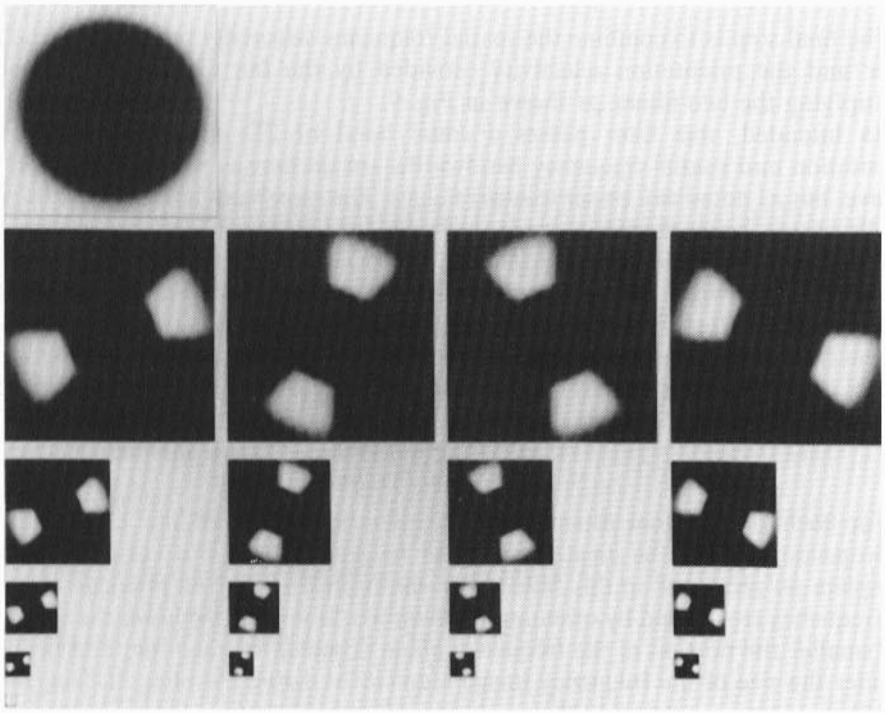

FIG. 11. Complete set of cortex filters. Included are the high and low residue filters.

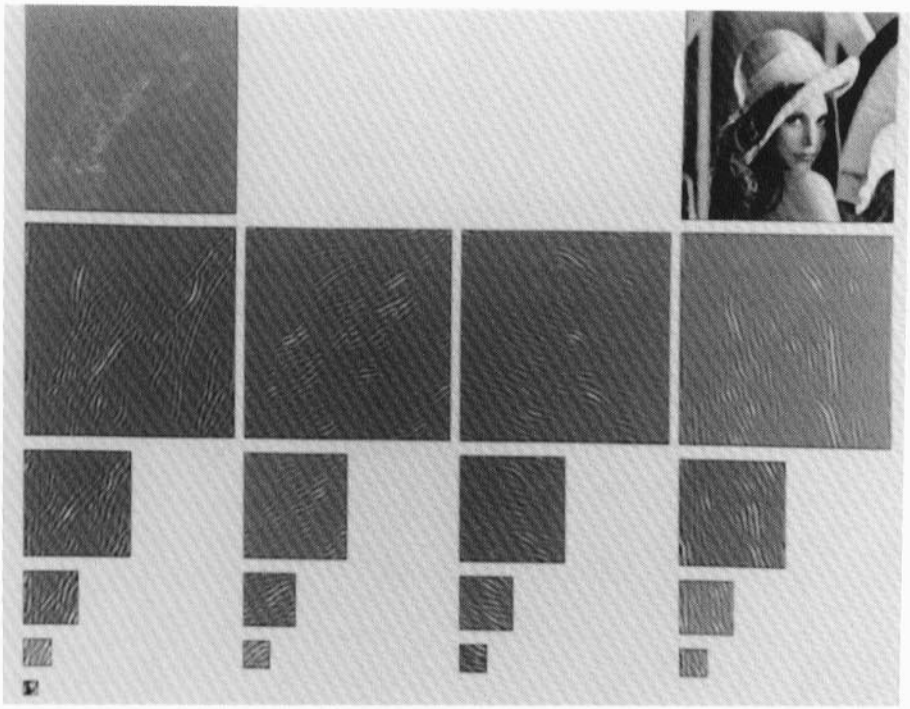

FIG. 12. Cortex transform of an image. Layers vary in resolution from row to row, and in orientation from column to column. The high and low frequency residues are also shown. All layers are scaled to full contrast to maximize visibility.

structure. This leads to an important economy in computing the responses of the various layers.

3.9. Computing Response Images

Response images are computed by multiplying the DFT of the input image by each filter in turn, and computing the inverse DFT. However, the multiplication and inverse DFT are performed only over a subimage determined by the size of the filter [27]. Thus at each successive resolution level, the size of the output image is reduced by a factor of s in each dimension. As a result, the cost of computing the cortex transform is borne almost entirely by the highest resolution level. Further consideration of the computational complexity of the transform is deferred to a later section.

Figure 12 shows the cortex transform of an image. Each layer may be regarded as the neural image corresponding to a particular cell type with a particular spatial frequency and orientation. Each pixel within the image may be considered the response of a single cell, when the cells are assumed to be arranged in a square grid. When some other cell distribution is assumed, single cell responses can be interpolated from the neural image, as noted above.

3.10. Reconstruction of the Cortex Transform

To reconstruct the image, each layer of the cortex transform is expanded to the size of the original image, and then all layers are added together. The expansion is

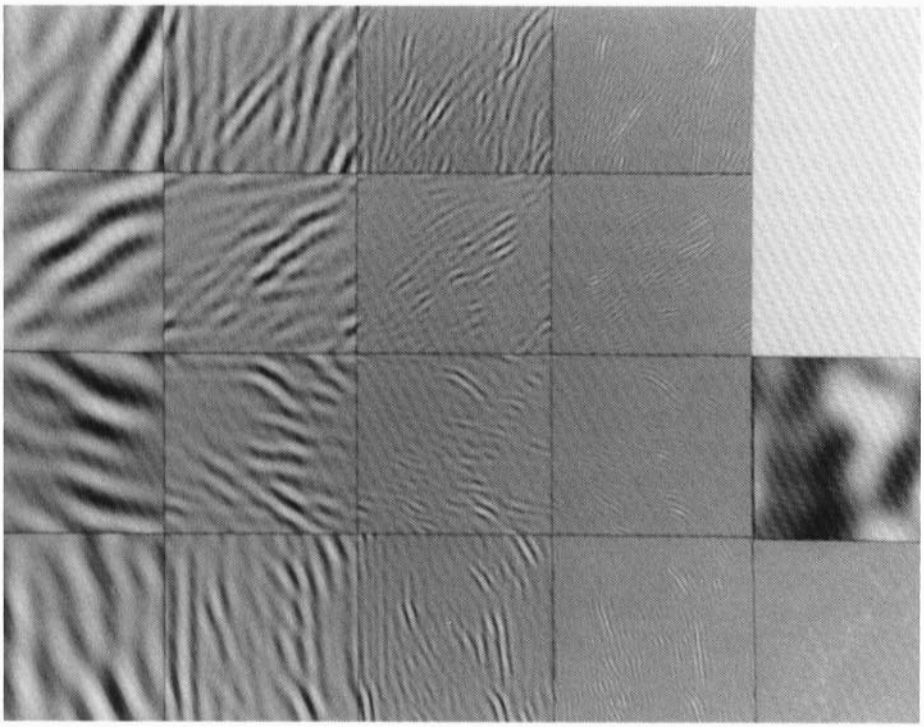

FIG. 13. The expanded cortex transform of the image shown in Fig. 12. Each layer has been expanded to the size of the original by ideal interpolation.

done by computing the DFT of the layer, embedding the DFT in a null DFT the size of the original, and computing the inverse DFT [27]. An example of the cortex transform expanded in this way is shown in Fig. 13. Note that the forward and inverse DFTs need be done only once for each resolution, since the several different orientations can be added together before expansion. This reconstruction is complete: it returns exactly the original image. That this is so is evident from the way in which each layer is constructed. Each is the output of a filter, and the set of filters are created by exactly subdividing the frequency image. If we write c_i for the filter corresponding to layer i and a for the input image, then the sum of the filtered images is

$$\sum_i c_i * a = \left(\sum_i c_i\right) * a = 1 * a = a. \tag{9}$$

4. COMPLEXITY ANALYSIS

Here we consider the number of operations required to compute the cortex transform of an image. To simplify matters, we will assume that the scale factor s is 2, and that the original image is square with width $N = 2^Q$. We will also assume that real additions and multiplications take about equally long on the target computer and can be combined in one pool called "real operations."

There are of course many ways to compute the cortex transform. Here we will discuss an algorithm that exploits several, but not necessarily all of the available shortcuts. First we compute the DFT of the input image. Straightforward calculation of the DFT of a real N by N image using the FFT algorithm requires approximately $5N^2\log_2 N$ real operations [31]. The second step is to compute the complex product of the transform of the input image and the transform of a particular filter. These transforms are of size N by N, but because they are Hermite, we need only record (and multiply) half that many values. In addition, the values of the filter transform are all real, hence each nominally complex multiplication requires only two real multiplications. The total real operations for the transform product is therefore N^2. Finally, the product must be inverse transformed. Thus the total for one filter is $N^2 + 5N^2\log_2 N$. This must now be multiplied by Ω, the number of different orientation filters at each resolution.

If all resolution layers were computed in this way, this quantity would now be multiplied by the number of resolutions, which is $\log_2 N$. But at the lower resolutions the filters extend only over a small portion of the total spectrum. Specifically, at resolution k the filter support is contained entirely within a region of size $(Ns^{-k})^2$, as can be seen in Fig. 11. Rather than computing the transform product over the whole N by N complex transform, inverse transforming, and subsampling, it is equivalent to compute the product *and inverse transform* of only this much smaller area [27]. Hence the real operations required for the transform product and inverse transform at resolution number k are $(N2^{-k})^2[1 + 5\log_2(N2^{-k})]$. If $N = 2^Q$, then this simplifies to $2^{2(Q-k)}[1 + 5(Q - k)]$. The total number of operations is therefore

$$5\,2^{2Q}Q + \Omega \sum_{k=0}^{Q} 2^{2(Q-k)}[1 + 5(Q - k)]. \tag{10}$$

The first term accounts for the forward DFT of the image, and the second term for the complex multiplication and inverse transformation of each layer. The summation from $k = 0$ to Q assumes that the lowest frequency filter is of size 1×1. In practice, we have used a smallest filter of 8×8, so the actual number of operations we required was very slightly less than this number. Typical values are shown in Table 1. The computation is dominated by the DFTs at the highest resolution, so that, for example, when $\Omega = 4$ the cost is approximately equal to the cost of computing $\Omega + 1 = 5$ DFTs. In this light it is worth noting more sophisticated algorithms which almost halve the number of operations required for multidimensional DFTs [31], as well as the wide availability of special hardware for calculation of DFTs. It is also instructive to compare these computational costs to those of

TABLE 1

Image width	Operations
8	5,856
64	763,680
512	73,778,976

other algorithms. The DOLP (difference-of-low-pass) transform takes 7.27 million operations to generate 16 band-pass images from a 256 by 256 original [26]. The cortex algorithm described above takes 16.37 million operations to compute 32 oriented band-pass images from a 256 by 256 original.

APPENDIX: NOTATION

β	ratio of corner and Nyquist frequencies
γ	frequency sharpness parameter
ω	orientation sharpness parameter
f	corner frequency
s	scale factor
u, v	2D spatial frequency
$\tilde{m}_k(u, v)$	mesa filter at resolution level k
$\tilde{d}_k(u, v)$	dom filter at resolution level k
$\tilde{b}_\alpha(u, v)$	bisection filter at orientation α
$\tilde{f}(u, v)$	fan filter
N	image width and height in pixels
Ω	number of orientation bands
$*$	convolution operator

REFERENCES

1. S. W. Kuffler, Discharge patterns and functional organization of mammalian retina., *J. Neurophysiol.* **16**, 1953, 37–68.
2. J. G. Robson, Receptive fields: Neural representation of the spatial and intensitive attributes of the visual image, in *Handbook of Perception, Vol. 5, Seeing* (C. Carterrette and M. P. Friedman, Eds.), Academic Press, New York, 1975.
3. J. G. Robson, Neural images: The physiological basis of spatial vision, in *Visual Coding and Adaptability* (C. S. Harris, Ed.), pp. 177–214, Erlbaum, Hillsdale, NJ, 1980.
4. P. Lennie, Parallel visual pathways: A review, *Vision Res.* **20**, 1980, 561–594.
5. R. M. Shapley and P. Lennie, Spatial frequency analysis in the visual system, in *Annual Review of Neuroscience*, Vol. 8 (W. M. Cowan, Ed.), pp. 547–583, *Annual Reviews*, Palo Alto, CA, 1985.
6. D. G. Albrecht and D. B. Hamilton, Striate cortex of cat and monkey: Contrast response function, *J. Neurophysiol.* **48**, 1982, 217–237.
7. A. F. Dean, The relationship between response amplitude and contrast for cat striate cortical neurones, *J. Physiol.* **318**, 1981, 413–427.
8. R. L. De Valois, D. G. Albrecht, and L. G. Thorell, Spatial frequency selectivity of cells in macaque visual cortex, *Vision Res.* **22**, 1982, 545–559.
9. J. A. Movshon, I. D. Thompson, and D. J. Tolhurst, Spatial and temporal contrast sensitivity of neurones in areas 17 and 18 of the cat's visual cortex, *J. Physiol.* **283**, 1978, 101–120.
10. M. A. Webster and R. L. De Valois, Relationship between spatial-frequency and orientation tuning of striate-cortex cells, *J. Opt. Soc. Amer. A* **2**, 1985, 1124–1132.
11. R. L. De Valois, E. W. Yund, and H. Hepler, The orientation and direction selectivity of cells in macaque visual cortex, *Vision Res.* **22**, 1982, 531–544.
12. N. Graham and J. Nachmias, Detection of grating patterns containing two spatial frequencies: A comparison of single-channel and multiple-channel models, *Vision Res.* **11**, 1971, 251–259.
13. C. Blakemore and F. W. Campbell, On the existence of neurones in the human visual system selectively sensitive to the orientation and size of retinal images, *J. Physiology London* **203**, 1969, 237–260.
14. S. Marcelja, Mathematical description of the responses of simple cortical cells, *J. Opt. Soc. Amer.* **70**, 1980, 1297–1300.

15. J. J. Kulikowski, S. Marcelja, and P. O. Bishop, Theory of spatial position and spatial frequency relations in the receptive fields of simple cells in the visual cortex, *Biol. Cybernet.* **43**, 1982, 187–198.
16. J. D. Daugman, Two-dimensional spectral analysis of cortical receptive field profiles, *Vision Res.* **20**, 1980, 847–856.
17. D. J. Field, Even- and odd-symmetric receptive fields and equivalent stimuli, *Perception* **14**, 1985, A14.
18. A. B. Watson, Detection and recognition of simple spatial forms, in *Physical and Biological Processing of Images* (A. C. Slade, Ed.), Springer-Verlag, Berlin, 1983.
19. K. R. K. Nielsen, A. B. Watson, and A. J. Ahumada, Jr., Application of a computable model of human spatial vision to phase discrimination, *J. Opt. Soc. Amer. A* **2**, 1985, 1600–1606.
20. B. Sakitt and H. B. Barlow, A model for the economical encoding of the visual image in cerebral cortex, *Biol. Cybernet.* **43**, 1982, 97–108.
21. H. R. Wilson and J. R. Bergen, A four mechanisms model for threshold spatial vision, *Vision Res.* **19**, 1979, 19–33.
22. S. A. Klein and D. M. Levi, Hyperacuity thresholds of 1 sec: Theoretical predictions and empirical validation, *J. Opt. Soc. Amer. A* **2**, 1985, 1170–1190.
23. R. J. Watt and M. J. Morgan, A theory of the primitive spatial code in human vision, *Vision Res.* **25**, 1985, 1661–1674.
24. S. Tanimoto and T. Pavlidis, A hierarchical data structure for picture processing, *Comput. Graphics Image Process.* **4**, 1975, 104–119.
25. P. J. Burt and E. H. Adelson, The laplacian pyramid as a compact image code, *IEEE Trans. Commun.* **COM-31**, 1983, 532–540.
26. J. L. Crowley and R. M. Stern, Fast computation of the difference of low-pass transform, *IEEE Trans. Pattern Anal. Mach. Intell.* **PAMI-6**, 1984, 212–222.
27. A. B. Watson, *Ideal Shrinking and Expansion of Discrete Sequences*, NASA Technical Memorandum 88202, 1986, 1–11, National Technical Information Service, Springfield, VA.
28. A. Rosenfeld, *Multiresolution Image Processing and Analysis*, Springer-Verlag, New York, 1984.
29. M. Porat and Y. Y. Zeevi, *The Generalized Gabor Scheme of Image Representation in Vision*, EE Pub. 519, Faculty of Electrical Engineering, Technion Israel Institute of Technology, Haifa, Israel., 1985, 1–22.
30. A. J. Ahumada, Jr. and A. B. Watson, Equivalent noise model for contrast detection and discrimination, *J. Opt. Soc. Amer. A* **2**, 1985, 1133–1139.
31. H. J. Nussbaumer, *Fast Fourier Transformation and Convolution Algorithms*, Springer-Verlag, New York, 1982.