

A Wavelet Basis for Euclidean Invariant Computation of Visual Contours

By John Zweck and Lance Williams

Vision is by far the most highly developed sense in humans. Much of our knowledge of the external world is based on information presented to us in visual form, and a large portion of the brain is devoted to visual information processing. One of the basic tasks of human—and computer—vision systems is to complete the boundaries of partially occluded objects. The goal of our research is to develop a biologically plausible theory of boundary-contour completion. In this article we describe the first discrete numerical algorithm [11] for completing contours in a Euclidean invariant manner and then discuss its parallel implementation.

APPLICATIONS ON ADVANCED ARCHITECTURE COMPUTERS

Greg Astfalk, Editor

A first step in the process of identifying objects in the environment is to locate the contours that define the boundaries of the objects. Because the boundaries of objects in complex environments are often partially hidden (or occluded) by other objects, a robustly functioning visual system must be able to fill in, or complete, the missing portions of the boundaries. The boundary of an object, even when not actually hidden by another object, is often difficult to detect; typically, this happens when the reflectance of the object matches the reflectance of some part of its background.

The human visual system is particularly good at filling in missing object boundaries. The results are called “illusory contours.” To test your own visual system’s ability to form illusory contours, take a look at the well-known Ehrenstein stimulus and Kanizsa triangle (Figure 1).

Our new algorithm is based on a recently developed computational theory of contour completion [9, 10]. Like computations in the human visual cortex (and unlike all previous models of illusory contour formation), our computation is *Euclidean invariant*—an arbitrary rotation and translation of the input image produce an identical rotation and translation of the resulting illusory contours (see Figure 2). We achieve this invariance by representing the various states of the computation in a basis of “shiftable–twistable” functions. Neuroscience experiments suggest that ensembles of neurons in the visual cortex can be modeled by a shiftable–twistable basis. From a mathematical point of view, the novelty of our approach comes from the use of a shiftable–twistable basis to eliminate grid orientation effects in the numerical solution of an advection–diffusion equation.

Since our algorithm uses a basis of fine-scale three-dimensional functions to complete object boundaries in complex scenes, it is not practical to perform the computation, at the high resolutions required, on serial machines. Furthermore, in the next stage of our research, the main routine in the algorithm will be iterated to produce contours that are a priori closed curves. For these reasons, we decided on a parallel implementation of our algorithm (on an IBM–SP2 parallel computer).

The Human Visual System

The architecture of the visual cortex—the region of the brain responsible for visual information processing—provides several clues as to how illusory contour formation might occur in the brain. Two of the most important areas for visual information processing are the primary visual cortex (V1) and the secondary visual cortex (V2) (which receives its input from V1). Orientation selectivity first emerges

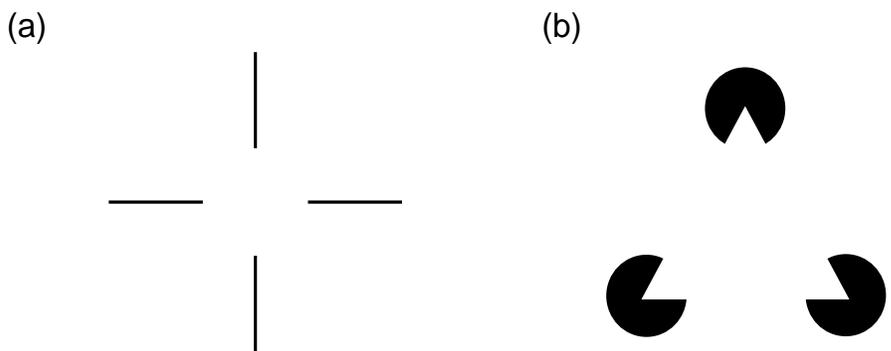


Figure 1. The Ehrenstein stimulus (a) and the Kanizsa triangle (b).

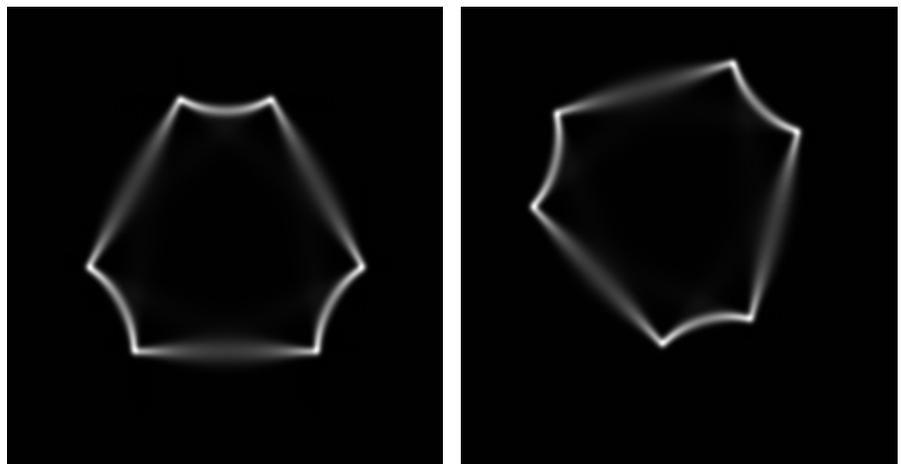


Figure 2. Stochastic completion fields, for the Kanizsa triangle (left) and after rotation and translation of the initial conditions (right).

in V1, and there is strong evidence to suggest that illusory contour formation occurs in V2.

Any computational model of human visual information processing must reconcile two apparently contradictory observations: On the one hand, computations in the primary visual cortex are largely Euclidean invariant; on the other hand, the discrete spatial sampling of the visual field by V1 is exceedingly sparse. Put succinctly: Why doesn't our perception of the world change dramatically when we tilt our head by 5 degrees? A similar issue was raised in a 1999 *Nature* paper [2]:

“On average, a region of just 1 mm² on the surface of the cortex will contain all possible orientation preferences, and, accordingly, can analyze orientation for one small area of the visual field. This topographical arrangement allows closely spaced objects with different orientations to interact. But it also means that a continuous line across the whole visual field would be cortically depicted in a patchy, discontinuous fashion. How can the spatially separated elements be bound together functionally?”

These issues are precisely the ones we address in this article. One of our main goals is to show how the sparse sampling of the visual field can be reconciled with the Euclidean invariance of visual computations. To realize this goal, we introduce the notion of a shiftable–twistable basis of functions on the space $\mathbf{R}^2 \times S^1$ of positions and directions. This notion is a generalization of the notion of a shiftable–steerable basis of functions on the plane \mathbf{R}^2 , introduced by Simoncelli et al. [8] to perform Euclidean invariant computations on \mathbf{R}^2 . Since many computations in V1 and V2 likely operate on functions on $\mathbf{R}^2 \times S^1$, rather than on \mathbf{R}^2 (e.g., [3, 5, 7, 9]), we propose that shiftable–twistability (in addition to shiftable–steerability) is the property that binds spatially separated elements together functionally to perform Euclidean invariant computations in the visual cortex.

Neurons in the visual cortex are characterized by their *receptive fields*. Classically, the receptive field of such a neuron is defined as the two-dimensional function that indicates the response (i.e., the firing rate) of the neuron to a point of light at every location on the retina. Typically, receptive fields are localized in both space and frequency.

There are neurons in V1 called *simple cells* whose receptive fields exhibit a marked preference for narrow stimuli of a specific orientation. Traditionally, these cells have been described as edge (or bar) detectors. Accurate models of the receptive fields of simple cells can be produced with two-dimensional Gabor functions [1], which are the product of a Gaussian and a complex plane wave. Gabor functions, unique in being maximally localized in both space and frequency, are well suited to the encoding of visual information. Based on experimental observations, Daugmann [1] (and others) suggest that an ensemble of receptive fields of simple cells can be regarded as performing a wavelet basis expansion of the image, in which the responses of the neurons correspond to the coefficients in the expansion and the receptive fields correspond to the basis functions.

Results of a recent experiment by von der Heydt et al. [4] suggest that illusory contour formation occurs in V2. They observed that the firing rate of certain neurons in V2 increases when their receptive fields are crossed by illusory contours (of specific orientations) induced by pairs of bars flanking the receptive field. Significantly, when the same bars are presented singly, the neurons do not respond; they respond only to pairs.

Although our new contour-completion algorithm does not provide a model for illusory contour formation in the visual cortex that is realistic in every respect, it does have several biologically plausible features, none of which are found in previous algorithms, e.g., [3, 5, 7, 10]. Specifically, in our algorithm: (1) all states of the computation are represented in a wavelet-like basis of functions that are localized in both space and frequency (with the spatial localization making it possible to perform the computation in parallel), (2) the computation operates on the coefficients in the wavelet-like expansion, (3) the computation is Euclidean invariant, and (4) the basis functions used in the computation have centers lying on a relatively sparse grid in the image plane.

Stochastic Completion Fields

Like many problems in vision, illusory contour formation is ill-posed: The visual system cannot know the precise shape of the portions of an object's boundary that are hidden from view. In [9] it is argued that the visual system computes the local image plane statistics for the distribution of all possible completions, rather than simply the most probable one. As in [6], the assumption is that any prior probability distribution of completion shapes encoded in the human visual system would favor smoother, shorter shapes; moreover, the distribution of completion shapes that can extend a contour at a point depends only on the position and direction of the contour at that point, i.e., the distribution can be modeled by a Markov process. These assumptions led to the use of the device of a particle undergoing a stochastic motion in modeling the distribution of completion shapes. Particles, which possess a position, $\vec{x} = (x, y) \in \mathbf{R}^2$, and a direction of motion, $\theta \in S^1$, move with constant speed in directions governed by Brownian motions. In [6], it was observed that the probability $P(\vec{x}, \theta, t)$ that a particle is at (\vec{x}, θ) at time t evolves according to the Fokker–Planck advection–diffusion–decay equation:

$$\begin{aligned} \frac{\partial P}{\partial t} = & -\cos\theta \frac{\partial P}{\partial x} \\ & -\sin\theta \frac{\partial P}{\partial y} + \frac{\sigma^2}{2} \frac{\partial^2 P}{\partial \theta^2} - \frac{1}{\tau} P \end{aligned} \quad (1)$$

where σ and τ are the θ -diffusion and decay parameters.

Given a set of position and direction constraints, called sources and sinks, that represent the beginning and ending points of a set of contour fragments, the stochastic completion field [9] at (\vec{x}, θ) is defined as the probability that a contour from the prior distribution of completion shapes will pass through (\vec{x}, θ) on a path from a source to a sink. Because the stochastic process is Markov,

the completion field is the product of a source field and a sink field. At (\vec{x}, θ) the source field measures the probability that a contour beginning at a source will pass through (\vec{x}, θ) , and the sink field measures the probability that a contour beginning at (\vec{x}, θ) will reach a sink. The source and sink fields are obtained by integrating solutions of the Fokker–Planck initial-value problem (IVP) over time. Figure 2 shows the completion field for the Kanizsa triangle stimulus.

The Algorithm

In the computation of stochastic completion fields, the input and output are functions on $\mathbf{R}^2 \times S^1$, and the appropriate notion of invariance is determined by the symmetries $T_{\vec{x}_0, \theta_0}$ of $\mathbf{R}^2 \times S^1$ that perform a shift in \mathbf{R}^2 by \vec{x}, θ , followed by a *twist* in $\mathbf{R}^2 \times S^1$ through an angle θ_0 (i.e., a rotation of \mathbf{R}^2 and a corresponding translation in S^1 by θ_0). A visual computation on $\mathbf{R}^2 \times S^1$ is called shift–twist invariant if it is invariant under all the transformations $T_{\vec{x}_0, \theta_0}$ for $(\vec{x}_0, \theta_0) \in \mathbf{R}^2 \times S^1$. Shift–twist invariance is the appropriate notion of invariance for completion field computations because, in the continuum, the transformations $T_{\vec{x}_0, \theta_0}$ are symmetries of the Fokker–Planck equation.

A shiftable–twistable basis of functions on $\mathbf{R}^2 \times S^1$ is a set of functions with the property that whenever a function $P(\vec{x}, \theta)$ is in their span, then so is $P(T_{\vec{x}_0, \theta_0}(\vec{x}, \theta))$, for every choice of (\vec{x}_0, θ_0) in $\mathbf{R}^2 \times S^1$. (We call such a set of functions a basis, even though the functions need not be linearly independent.) As such, the notion of a shiftable–twistable basis on $\mathbf{R}^2 \times S^1$ generalizes that of a shiftable–steerable basis on \mathbf{R}^2 . The discrete numerical algorithm described in this article computes completion fields in a shift–twist invariant manner by evolving the Fokker–Planck equation in a shiftable–twistable basis.

A discrete Dirac basis consisting of functions $\Psi_{\vec{k}, m}(\vec{x}, \theta) = \delta(\vec{x} - \vec{k}\Delta) \delta(\theta - m\Delta_\theta)$, where (\vec{k}, m) is a triple of integers, is not shiftable–twistable. (This is true because a Dirac function located off the grid of Dirac basis functions is not in their span.) A major shortcoming of all previous contour-completion algorithms (e.g., [3, 5, 7, 10]) is that the computations are performed in such a basis. As a consequence, initial conditions that do not lie directly on the grid cannot be accurately represented in the basis and the completed contour will contain visible artifacts. Furthermore, completion fields computed by the finite differencing scheme of [10] exhibit a second visible artifact—namely, a marked anisotropic spatial smoothing because of the manner in which two-dimensional advection is performed on a grid (see Figure 3). Although probability mass (i.e., particles evolving according to the Fokker–Planck equation) advects perfectly in either of the coordinate directions, mass moving at an angle to the grid will gradually disperse, since bilinear interpolation is used to place the mass on the grid.

For our new algorithm, we first define the concept of a shiftable–twistable function on $\mathbf{R}^2 \times S^1$ and construct a shiftable–twistable basis by taking a discrete number of transformations $T_{\vec{k}\Delta, m\Delta_\theta}$ of a given shiftable–twistable function. We then design a shiftable–twistable basis in which the initial conditions (the sources and sinks) can be accurately represented. To solve the Fokker–Planck equation, we express its solution in terms of the basis functions $\Psi_i(\vec{x}, \theta)$ as

$$P(\vec{x}, \theta; t) = \sum_i c_i(t) \Psi_i(\vec{x}, \theta) \quad (2)$$

where the coefficients $c_i(t)$ depend on time. Finally, we derive a linear transformation, $c(t + \Delta t) = (\mathbf{A} \circ \mathbf{D})c(t)$, to evolve the coefficient vector. This transformation is the composition of an advection transformation, \mathbf{A} , which transports probability mass in directions θ , and a diffusion-decay transformation, \mathbf{D} , which diffuses mass in θ and decays mass over time.

Shiftability–twistability is used in two ways to obtain shift–twist invariant completion fields. First, it enables all pairs of initial conditions related by an arbitrary transformation $T_{\vec{x}_0, \theta_0}$ to be represented equally well in the basis. Second, it is used to derive a shift–twist invariant advection transformation.

Shiftable–Twistable Bases

The concept of the shiftability [8] of a function is closely related to the Shannon sampling theorem. A periodic function $\psi(x)$ of period X is shiftable if there is an integer K such that the shift of ψ by an arbitrary amount, x_0 , can be expressed as a linear combination of K basic shifts of ψ ,

$$\psi(x - x_0) = \sum_{k=0}^{K-1} b_k(x_0) \psi(x - k\Delta) \quad (3)$$

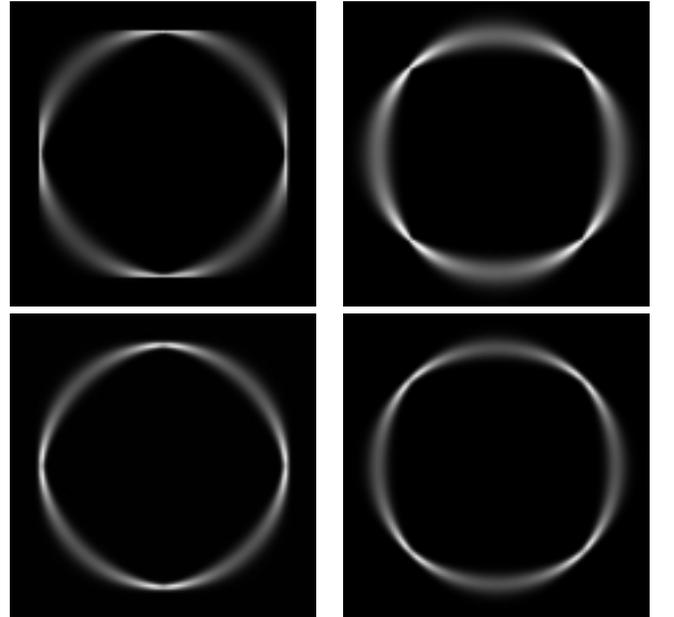


Figure 3. Completion fields for rotations of the Ehrenstein stimulus shown in Figure 1(a), obtained with the finite differencing scheme of [10] (top row) and with the new algorithm (bottom row).

for some interpolation functions $b_{\vec{k}}(x_0)$, where $\Delta = X/K$. The simplest shiftable function is a pure harmonic. More generally, we can shift any band-limited function by choosing K to be the number of frequencies in the band and setting $b_{\vec{k}}(x_0)$ equal to $b(x_0 - k\Delta)$, where $b(x)$ is the complex conjugate of the corresponding band-pass filter. Although they are linearly dependent, an appropriate collection of discrete dilations and translations of a shiftable function of mean zero forms an over-complete self-inverting wavelet basis (i.e., a tight frame).

A periodic function $\Psi(\vec{x}, \theta)$ on $\mathbf{R}^2 \times S^1$ is called *shiftable–twistable* if, for each (\vec{x}, θ) , there are interpolation functions $b_{\vec{k}, m}(\vec{x}_0, \theta_0)$ such that

$$\Psi\left(T_{\vec{x}_0, \theta_0}(\vec{x}, \theta)\right) = \sum_{\vec{k}, m} b_{\vec{k}, m}(\vec{x}_0, \theta_0) \Psi\left(T_{\vec{k}\Delta, m\Delta_\theta}(\vec{x}, \theta)\right) \quad (4)$$

for some choice of Δ and Δ_θ . Examples of shiftable–twistable functions include functions of the form $\Psi(\vec{x}, \theta) = \psi(\vec{x})f(\theta)$, where ψ is a Gaussian, Gabor, or directional derivative of Gaussian and f is a pure harmonic or Gaussian. In particular, receptive fields in V1 can be modeled by shiftable–twistable functions.

Solution of the Fokker–Planck Equation

The initial conditions for the Fokker–Planck IVP are modeled by fine-scale, three-dimensional Gaussians whose centers are determined by the locations and directions of the edge fragments to be completed. For simplicity, we solve the Fokker–Planck equation in a Gaussian–Fourier basis consisting of functions $\Psi_{\vec{k}, \omega}(\vec{x}, \theta) = \psi(\vec{x} - \vec{k}\Delta)e^{i\omega\theta}$, where ψ is a radial Gaussian of the same fineness as the Gaussian initial conditions. These choices enable us to use shiftable to accurately represent the initial conditions in the basis. The computation of completion fields in the Gaussian–Fourier basis, as explained in [11], can be interpreted as a computation in a directional derivative of the Gaussian–Fourier basis, whose elements look more like receptive fields in V1.

The solution of the Fokker–Planck equation in the basis is given by a linear transformation, $c(t + \Delta t) = (\mathbf{A} \circ \mathbf{D}) c(t)$, of the coefficient vector $c(t) = \{c_{\vec{k}, \omega}(t)\}$, which evolves the probability density function,

$$P(\vec{x}, \theta; t) = \sum_{\vec{k}, \omega} c_{\vec{k}, \omega}(t) \Psi_{\vec{k}, \omega}(\vec{x}, \theta) \quad (5)$$

according to the Fokker–Planck equation.

Let $b(\vec{x}_0)$ denote the interpolation functions used to shift $\psi(\vec{x})$ by \vec{x}_0 . The advection transformation \mathbf{A} , which has the effect of translating $P(\vec{x}, \theta; t)$ in direction θ at unit speed for time Δt , is given by

$$c_{\vec{l}, \eta}(t + \Delta t) = \sum_{\vec{k}, \omega} \hat{b}_{\vec{l} - \vec{k}, \eta - \omega}(\Delta t) c_{\vec{k}, \omega}(t) \quad (6)$$

where $\hat{b}_{\vec{k}, \omega}(\Delta t)$ denotes the ω th Fourier series coefficient, with respect to θ , of $b_{\vec{k}}(\Delta t(\cos \theta, \sin \theta))$. In particular, \mathbf{A} is a convolution operator on the vector space of coefficients $c_{\vec{k}, \omega}$.

To derive the expression for the advection transformation \mathbf{A} , we exploit the fact that shiftable can be used to perform perfect spatial advection in direction θ , with shiftable basis functions $\psi_{\vec{k}}(\vec{x})$ in \mathbf{R}^2 and the continuous variable $\theta \in S^1$. The similarity transformation given by the standard analysis and synthesis formulae for Fourier series in θ is used to obtain the required expression for \mathbf{A} in the shiftable–twistable basis.

Parallel Implementation

For a spatially parallel implementation of our algorithm on an IBM–SP2, we used the Fastest Fourier Transform in the West (FFTW) software (see *SIAM News*, November 1999, page 1). The output of our algorithm is a picture of the completion field obtained for an input configuration of sources and sinks. The main computational task is to compute the coefficient vectors $c_{\vec{k}, \omega} = c_{k_x, k_y, \omega}$ of the source and sink fields. FFTW dictated that we use a slab decomposition of the coefficient vector, which, to minimize communication between processors, was chosen with respect to k_x . If the number of processors divides the number K of basic translates of $\psi(\vec{x})$ in the x variable, then the computation is load-balanced.

The iterative computation of the source (or sink) field coefficient vectors is performed in the three-dimensional Fourier domain of the coefficient vector $c_{\vec{k}, \omega}$, where the advection transformation is a multiplication operator and the diffusion–decay transformation is given by an explicit, stable, three-point stencil finite differencing scheme in one variable. For the source and sink fields, the processors need to communicate only when the three-dimensional FFT and its inverse are computed. We avoid aliasing effects in the advection transformation by making the number of frequencies in θ proportional to the number K of basic translations. Consequently, doubling the resolution of the Gaussian initial conditions in all three variables requires multiplying the number of basis functions by 16.

Pictures of the three-dimensional completion fields are computed by analytically integrating out the θ variable in the product of the source field, $\sum_{\vec{k}, \omega} c_{\vec{k}, \omega} \psi_{\vec{k}}(\vec{x})e^{i\omega\theta}$, and the sink field, $\sum_{\vec{l}, \eta} c_{\vec{l}, \eta} \psi_{\vec{l}}(\vec{x})e^{i\eta\theta}$. Because $\psi(\vec{x})$ decays exponentially, for each \vec{k} , we needed to sum only over nearby \vec{l} 's. Hence, each processor needs to communicate only with its left and right neighbors. Each of the completion fields in Figure 2 and Figure 3 took about 12 minutes to compute on 16 processors.

Experimental Results

The results of two experiments demonstrate the Euclidean invariance of our algorithm. In each experiment, the shiftable–twistable basis consisted of $K = 160$ translates, in x and y , of a Gaussian (of period $X = 40.0$) and 92 frequencies in θ , for a total of 2.355×10^6 basis functions. Pictures of completion fields were obtained by analytic integration over θ and rendering of the field on a 256×256 grid. For both experiments, the decay constant was $\tau = 4.5$ and the time increment was $\Delta t = 0.1$. The diffusion parameter was $\sigma = 0.14$ in the first experiment and $\sigma = 0.08$ in the second.

In the first experiment we computed completion fields of Kanizsa triangles. The left-hand side of Figure 2 shows the completion field for the Kanizsa triangle stimulus from Figure 1(b); on the right-hand side of Figure 2, the initial stimulus has been rotated and translated. The completion field on the right is itself a rotation and translation of the completion field on the left—a demonstration of the Euclidean invariance of our algorithm.

In the second experiment we compared our algorithm with the finite differencing scheme of [10]. For the finite difference scheme, the $40.0 \times 40.0 \times 2\pi$ space was discretized using a 256×256 spatial grid with 36 discrete orientations, for a total of 2.359×10^6 Dirac basis functions. The intent was to use approximately the same number of basis functions for both algorithms. Trilinear interpolation was used to represent the initial conditions on the grid. Pictures of the completion fields are shown in Figure 3; those in the top row were computed by the method of [10] and those in the bottom row by the new method. The completion fields in the left-hand column are for the Ehrenstein stimulus from Figure 1(a); for the pictures in the right-hand column, the initial conditions have been rotated by 45° . The completion fields in the bottom row demonstrate the Euclidean invariance of the new method, as compared with the obvious lack of Euclidean invariance in the completion fields in the top row. The visible straight line artifacts in these completion fields, oriented along the coordinate axes, result from the anisotropic nature of the advection process in the algorithm of [10].

Conclusion

An important initial stage in the analysis of a scene is the completion of the boundaries of partially occluded objects. In [9] the idea of a stochastic completion field was introduced to measure the probability distribution of completed boundary shapes in a given scene. As required of any computational model of human visual information processing, the method we have described for computing completion fields attempts to reconcile the apparent contradiction between the obvious Euclidean invariance of human visual computations and the observed sparseness of the discrete spatial sampling of the visual field by the primary visual cortex. Our method reconciles this contradiction by performing computations in a shiftable–twistable wavelet-like basis consisting of functions that are similar to the receptive fields of visual neurons.

Acknowledgments

John Zweck was supported (in part) by the Albuquerque High Performance Computing Center at the University of New Mexico. The authors thank Sue Minkoff for her very helpful comments on preliminary versions of this article.

References

- [1] J. Daugman, *Complete discrete 2-D Gabor transforms by neural networks for image analysis and compression*, IEEE Trans. Acoustics, Speech, and Signal Proc., 36:7 (1988), 1169–1179.
- [2] U. Eysel, *Turning a corner in vision research*, Nature, 399 (1999), 641–644.
- [3] S. Grossberg and E. Mingolla, *Neural dynamics of form perception: Boundary completion, illusory figures, and neon color spreading*, Psych. Rev., 92 (1985), 173–211.
- [4] R. von der Heydt, E. Peterhans, and G. Baumgartner, *Illusory contours and cortical neuron responses*, Science, 224 (1984), 1260–1262.
- [5] Z. Li, *A neural model of contour integration in primary visual cortex*, Neur. Comput., 10:4 (1998), 903–940.
- [6] D. Mumford, *Elastica and computer vision*, in *Algebraic Geometry and Its Applications*, Chandrajit Bajaj (ed.), Springer-Verlag, New York, 1994.
- [7] A. Sashua and S. Ullman, *Structural saliency: The detection of globally salient structures using a locally connected network*, Second International Conference on Computer Vision, Clearwater, FL, 1988, 321–327.
- [8] E. Simoncelli, W. Freeman, E. Adelson, and D. Heeger, *Shiftable multiscale transforms*, IEEE Trans. Inform. Theory, 38:2 (1992), 587–607.
- [9] L.R. Williams and D.W. Jacobs, *Stochastic completion fields: A neural model of illusory contour shape and salience*, Neur. Comput., 9:4 (1997), 837–858.
- [10] L.R. Williams and D.W. Jacobs, *Local parallel computation of stochastic completion fields*, Neur. Comput., 9:4 (1997), 859–881.
- [11] J. Zweck and L.R. Williams, *Euclidean group invariant computation of stochastic completion fields using shiftable-steerable wavelets*, Albuquerque High Performance Computer Center, University of New Mexico, Technical Report 99–007, October 1999.

John Zweck (zweck@cs.unm.edu) is a researcher and Lance Williams (williams@cs.unm.edu) a faculty member in the Computer Science Department at the University of New Mexico in Albuquerque.