M.I.T Media Laboratory Perceptual Computing Section Technical Report No. 319
S.M./E.E. Thesis, Department of Electrical Engineering and Computer Science

Kinetic Occlusion

Sourabh A. Niyogi
Department of Electrical Engineering and Computer Science
Room E15-384, The Media Laboratory
Massachusetts Institute of Technology
20 Ames St., Cambridge, MA 02139
Email: sourabh@media.mit.edu

Abstract

Visual motion boundaries provide a powerful cue for perceptual organization of scenes. Motion boundaries are present when surfaces in motion occlude one another. Conventional approaches to motion analysis have relied on assumptions of data conservation and smoothness, which has made analysis of motion boundaries difficult. We show that a common source of motion boundary, kinetic occlusion, can be detected using spatiotemporal junction analysis. Junction analysis is accomplished by modeling kinetic occlusion as the multiplication of spatiotemporally oriented patterns. Using this model, we show that kinetic occlusion appears as locally oriented power in the frequency domain. Detecting kinetic occlusion is possible by detecting and isolating this locally oriented power. We provide an analysis of kinetic occlusion in one and two dimensional spatiotemporal stimuli, and develop algorithms to detect it. Our algorithms are applied successfully on spatiotemporal stimuli containing occluding surfaces in motion.

1 Introduction

With motion boundaries as the sole cue, surfaces can easily be segmented, recognized, and perceived as having a distinct ordering in depth [19, 29]. Under natural conditions, motion boundaries are formed by occluding surfaces in motion. Relative motion of occluding surfaces generically results in accretion or deletion of surface texture of the *occluded* surface by the *occluding* surface. This basic phenomena, which we will call *kinetic occlusion*, is illustrated in Figure 1. Figure 1(a) depicts three frames of an image sequence where a tree occludes a flowerbed. As the tree moves in front of the flowerbed, portions of the flowerbed area are *accreted* and *deleted* due to relative motion between the tree and the flowerbed. Observers perceive distinct motion boundaries between the tree and the flowerbed, and perceive an *ordinal* depth relationship between the two surfaces.

Kinetic occlusion, when viewed in space-time, possesses a structure which observers may detect and analyze. Figure 1(b) illustrates the spatiotemporal structure present in the spatiotemporal stimulus of Figure 1(a). In space-time, there are spatiotemporal *junctions* indicative of the ordinal depth relationship between the two surfaces. That is, whenever spatiotemporal orientation *stops*, there is a strong cue to occlusion, i.e. *deletion* of surface texture; and, whenever spatiotemporal orientation *starts*, there is a strong cue to disocclusion, i.e. *accretion* of surface texture. The central contribution of this paper is an analysis of this spatiotemporal structure. By analyzing this spatiotemporal structure, we develop methods to detect and analyze kinetic occlusion.

2 Background

With a detected motion boundary, interpretation of kinetic occlusion is straightforward: if the motion boundary moves consistently with either of the surfaces that define it, then that surface is the *occluding* surface [34]. If motion could be estimated accurately near motion boundaries, then edges could be detected in the velocity field. By deducing which side of the moving edge possessed motion consistent with the motion boundary, the ordinal depth relationship between two surfaces could be determined.

However, motion estimation near motion boundaries is difficult! Conventional motion estimation algorithms assume that locally: (1) one motion is present; (2) the intensity is constant in space time; (3) the flow field is smooth. These assumptions are violated in regions undergoing kinetic occlusion because locally: (1) there are at least two motions present; (2) intensity is not conserved; instead, texture of the occluded surface is accreted or deleted; (3) the flow field is discontinuous. Recently, many have proposed algorithms with more relaxed assumptions, thus yielding significantly improved velocity estimates near motion boundaries:

- *Regularization/line processes.* If significant velocity gradients are detected during motion estimation, then line processes can be activated [16, 17, 24, 27, 33]; subsequent smoothing of motion is prevented across these line processes.
- *Multiple motions locally.* Flow constraints have been generalized to estimate multiple motions directly [22, 30], assuming a stimulus is a linear superposition of independent motion signals; kinetic occlusion cannot be considered as such.
- *Robust estimation.* Motion estimation methods have been modified to be less sensitive to violations of underlying assumptions of smoothness and data conservation [3, 4, 31]. By discounting deviations from underlying assumptions, occlusions are viewed as an anomalous process that should be ignored.
- *Multiple motions globally.* Parametrized estimates of local motion are typically clustered together in velocity space; these clusters can be used to describe global

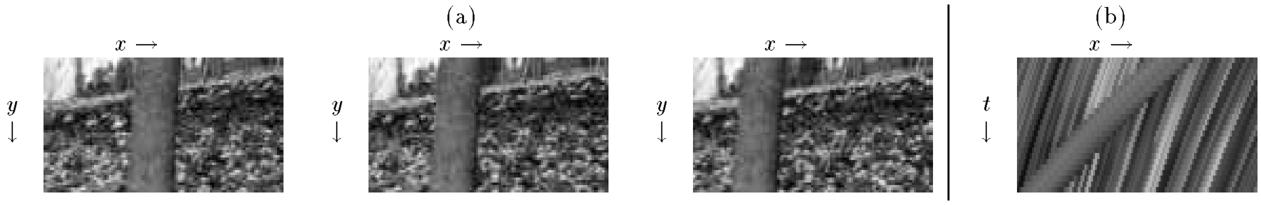

Figure 1: *Kinetic occlusion.* (a) Three frames of a spatiotemporal stimulus where two surfaces occlude one another. Texture of the posterior surface, the flowerbed, is *accreted* and *deleted* as the anterior surface, the tree, moves over it. (b) An (x, t) slice of the spatiotemporal stimulus of (a) reveals that when an orientation stops in space-time, there is a strong cue to occlusion.

surface motions effectively [18, 35]. Motion registration techniques where multiple transparent motions are estimated using global parametrized motion models have also been developed [2].

All of the above approaches pursue the goal of flow field estimation. Few have attempted to analyze kinetic occlusion directly. Exceptions are Bolles and Baker [5], who apply 3-d spatiotemporal edge detection to image sequences in order to link edges across space-time, and Zetsche et al [38], who suggest that occlusion can be detected via application of generic 3-d curvature operators directly to image sequences.

Distributed representations of motion, as used by biological visual systems, do not suffer from the problems of explicit velocity field construction [11, 13, 32]. Observing that translation in space-time appears as a plane passing through the origin in the frequency domain, these models construct distributed representations of motion via a two-stage process, as suggested by [23]. The first stage involves filtering image sequences with spatiotemporally oriented filters. The second stage pools the squared responses of these filters to construct units tuned to velocity. By constructing mechanisms which observe how distributed representations of motion change over time, we can detect kinetic occlusion.

3 Approach

We model the local structure present in occlusions directly, using the "layers" model of [35]. Layers $I_1(\mathbf{x})$ through $I_n(\mathbf{x})$ may attenuate the luminance of the layer beneath it by a factor $\alpha(\mathbf{x})$ $(0 \leq \alpha(\mathbf{x}) \leq 1)$, and may contribute its own emission $e_n(\mathbf{x})$. If layer $n-1$ contains a luminance pattern $I_{n-1}(\mathbf{x})$, then the luminance pattern at layer n is:

$$I_n(\mathbf{x}) = \alpha_n(\mathbf{x}) I_{n-1}(\mathbf{x}) + e_n(\mathbf{x}) \tag{1}$$

The variable $\mathbf{x} = (x, y)$ models spatial stimuli, $\mathbf{x} = (x, t)$ models one-dimensional spatiotemporal stimuli, and $\mathbf{x} = (x, y, t)$ models two-dimensional spatiotemporal stimuli. Occlusion and transparency can be modeled through variation of the attenuation mask $\alpha(\mathbf{x})$.

With just two layers, $I_1(\mathbf{x})$ and $I_2(\mathbf{x})$, representing an occluded and occluding surface respectively, and an attenuation mask $M(\mathbf{x})$, a simpler model can be used:

$$I(\mathbf{x}) = I_1(\mathbf{x}) M(\mathbf{x}) + I_2(\mathbf{x})(1 - M(\mathbf{x})) \tag{2}$$

Figure 2 illustrates how surfaces combine under the model. In space, where $\mathbf{x} = (x, y)$, the stimulus represents one oriented spatial pattern $I_1(\mathbf{x})$ disoccluded by another oriented spatial pattern $I_2(\mathbf{x})$; here, the spatial attenuation mask $M(\mathbf{x})$ has the same orientation as $I_2(\mathbf{x})$. In space-time, where $\mathbf{x} = (x, t)$, the stimulus represents a surface $i_1(x)$ moving leftward being occluded by a surface $i_2(x)$ moving rightward; here, the spatiotemporal attenuation mask $M(\mathbf{x})$ moves consistently with the occluder, and the texture of the occluded surface is accreted. Note that this is a *local* model for surface combination; ordinal relationships for surfaces may not be valid in a global context.

By exploiting the characteristic structure in the spatiotemporal domain and in the frequency domain, we show that the above image formation process involves *modulation* of information about the mask $M(\mathbf{x})$, and that for purposes of analysis it is natural to *demodulate* it. Recently, Fleet and Langley [9] have shown that many "second-order" motion stimuli, including kinetic occlusion, possess *locally* oriented power in the frequency domain. We specialize the analysis to kinetic occlusion in this paper, and consider both one-dimensional and two-dimensional kinetic occlusion. We illustrate that the filtering and squaring operations commonly employed in biological models of motion analysis demodulate this locally oriented power; detecting kinetic occlusion is possible with linear filtering mechanisms with simple non-linearities on this demodulated information.

As an example of extracting motion boundaries defined by kinetic occlusion, we illustrate our approach on a simple stimulus where a rightward moving square occludes a leftward moving background. Figure 3(a,b) depicts the stimulus; detected occlusion is shown in Figure 3(c). At the left side of the square, positive responses in Figure 3(c) indicate accretion of background surface texture. At the right side of the square, negative responses indicate deletion of background surface texture. Above and below the square, there is no occlusion, so little response is obtained.

By detecting kinetic occlusion, we provide a model for one of most salient interactions between the motion processing stream and the form processing stream. And, by detecting motion boundaries, other visual modules may be aided: Motion estimation can be improved by indicating where smoothing of motion estimates should not occur; and, as spatiotemporal junction analysis yields information similar to spatial junction analysis [14, 28], our module can be used for surface segregation processes [6, 7, 12, 15, 25].

In section 4, we analyze one-dimensional kinetic occlusion and develop techniques to analyze one-dimensional spatiotemporal stimuli. In section 5, we extend the analysis to two-dimensional spatiotemporal stimuli and develop algorithms to detect kinetic occlusion in real image se-

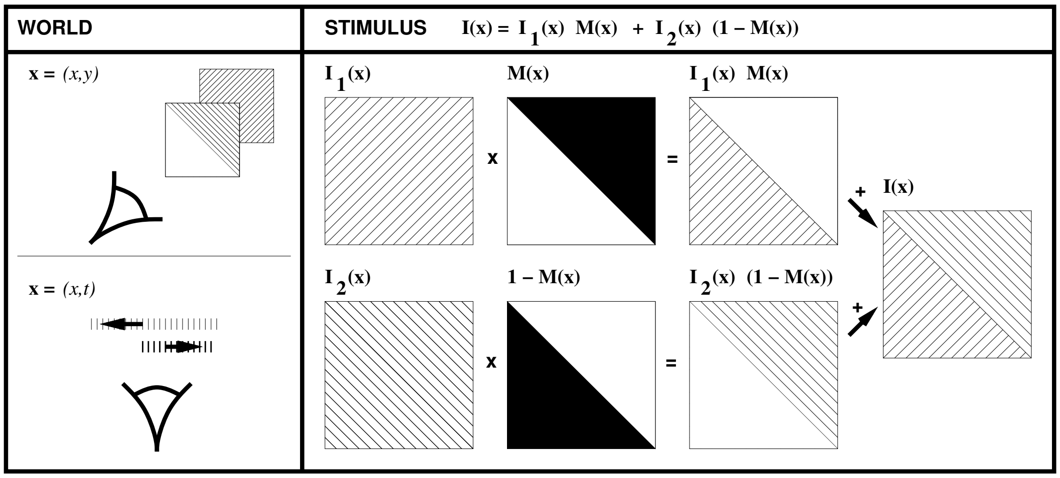

Figure 2: *Layers model of occlusion.* Locally, occlusion is modeled as the sum of multiplied oriented patterns, both in space, where $\mathbf{x} = (x, y)$, and in space-time, where $\mathbf{x} = (x, t)$ or $\mathbf{x} = (x, y, t)$.

quences. In section 6, results are presented on synthetic and natural image sequences containing occluding surfaces in motion. And, in section 7, we discuss future lines of investigation based on our work.

4 One-dimensional Kinetic Occlusion

We analyze kinetic occlusion in the frequency domain by taking the Fourier transform of our image formation equations. By understanding occlusion in the frequency domain, we can develop methods to detect it. The algorithms we will develop are *not* based on global Fourier transforms, but are instead are based on linear filtering mechanisms with simple non-linearities; we only use global Fourier analysis to illustrate the *local* structure present in occlusion so as to motivate our algorithms.

We first consider one-dimensional spatiotemporal stimuli, where $\mathbf{x} = (x, t)$ and $\mathbf{w} = (w_x, w_t)$. Using $\hat{I}(\mathbf{w})$ to denote the Fourier transform of $I(\mathbf{x})$, and using $*$ to denote convolution, our two-layer image formation model is:

$$I(\mathbf{x}) = I_1(\mathbf{x})M(\mathbf{x}) + I_2(\mathbf{x})(1 - M(\mathbf{x})) \tag{3}$$

$$\hat{I}(\mathbf{w}) = \hat{I}_1(\mathbf{w}) * \hat{M}(\mathbf{w}) + \hat{I}_2(\mathbf{w}) * (\delta(\mathbf{w}) - \hat{M}(\mathbf{w})) \tag{4}$$

By applying a translation model locally, we model layers $I_n(\mathbf{x})$ as 1-d spatial surfaces $i_n(x)$ which move with velocity v_n:

$$I_n(\mathbf{x}) = i_n(x + v_n t) \tag{5}$$

For a translating surface moving with velocity v_n, in the Fourier domain all the power lies along a line $w_x v_n + w_t = 0$:

$$\hat{I}_n(\mathbf{w}) = \hat{i}_n(w_x)\delta(v_n w_x + w_t) \tag{6}$$

where $\delta(x)$ is a Dirac delta function. For one-dimensionally moving patterns, the velocity v_n of a surface is directly related to the measured spatiotemporal orientation θ_n by $v_n = \tan(\theta_n)$. In our discussion, we use the terms motion and orientation interchangeably.

4.1 XT Kinetic occlusion: Model

Occlusion is modelled with a step edge $u(x)$ translating with speed v_m:

$$M(\mathbf{x}) = u(x + v_m t) \tag{7}$$

$$\hat{M}(\mathbf{w}) = \hat{u}(w_x)\delta(v_m w_x + w_t) \tag{8}$$

In kinetic occlusion, the velocity of the occluder is identical to the velocity of the attenuation map, so $v_m = v_2$. Figure 4 illustrates the image formation process for an example stimulus in both domains. In the spatiotemporal domain, a rightward moving sinusoid ($i_1(x) = \sin(w_{x1}x)$, $v_1 = 1$) is occluded by an leftward moving sinusoid ($i_2(x) = \sin(w_{x2}x)$, $v_m = v_2 = -1$). In the frequency domain, $\hat{I}_1(\mathbf{w})$ contains two impulses that lie along the line $w_x v_1 + w_t = 0$, and all the power in $\hat{M}(\mathbf{w})$ lies along the line $w_x v_m + w_t = 0$. Multiplying the patterns in the spatiotemporal domain amounts to convolving their spectra in the spatiotemporal frequency domain.[1] The *multiplication* of $M(\mathbf{x})$ by $I_1(\mathbf{x})$ *modulates* the power of $\hat{M}(\mathbf{w})$ off center to the impulses of $\hat{I}_1(\mathbf{w})$. This results in *locally oriented power* in $\hat{M}(\mathbf{w}) * \hat{I}_1(\mathbf{w})$, as shown in Figure 4(a,b,c).

Power oriented relative to the center of the frequency domain ($\vec{w} = 0$) is termed *first order* orientation. Power oriented off the center at some location $\vec{w}_0$ (where $\vec{w}_0 \neq 0$) is termed *second order* orientation. We must develop methods to analyze this second order orientation. Because $v_m = v_2$, multiplying $I_2(\mathbf{x})$ by $1 - M(\mathbf{x})$ does not introduce any observable second order orientation; all the power is present along the line $w_x v_2 + w_t = 0$, as shown in Figure 4(d,e,f). The summation of the two components shown in Figure 4(c) and (f) is shown in Figure 4(g).

[1] The two-dimensional FFTs in our figures were obtained by multiplying the image by a Gaussian in the spatial domain to reduce edge effects. We clipped the resulting magnitude of its Fourier transform for purposes of presentation.

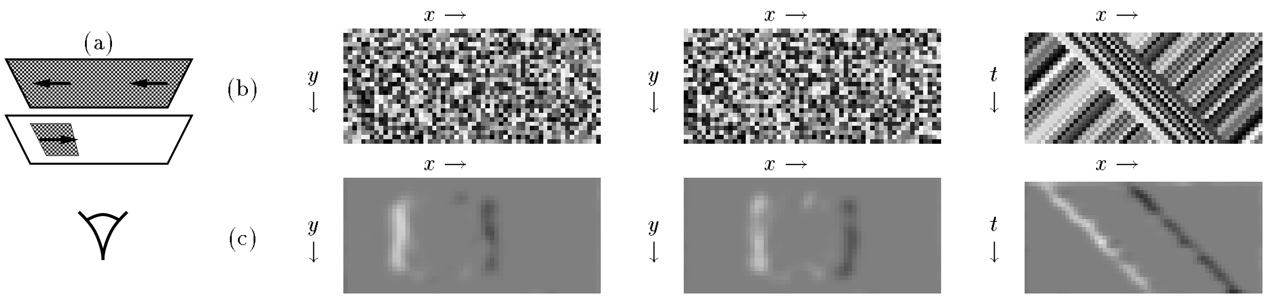

Figure 3: *Squares.* (a) Depiction of stimulus - a square moving rightward occluding a background moving leftward. (b) Two (x, y) and one (x, t) slice from a 64x32x32 representation of the stimulus. (c) Kinetic occlusion detected for the same spatiotemporal slices.

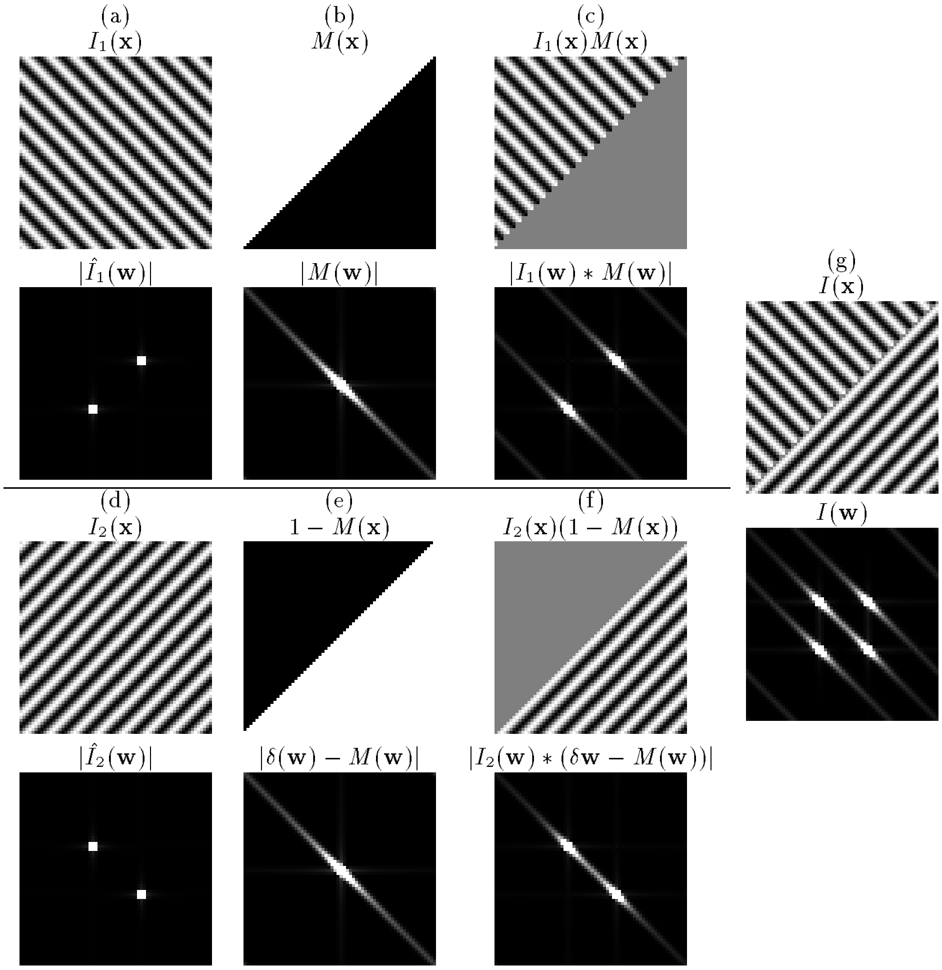

Figure 4: *XT Kinetic occlusion: Image formation.* Multiplying oriented patterns in the spatiotemporal domain results in locally oriented power in the frequency domain.

As the velocity of the mask v_m, changes, so does the second order orientation. Figure 5(a) shows a second stimulus with a different occluding velocity v_m and the resulting second order orientation.

While we assume a perfect step edge model, blurring the edge of the mask merely attenuates high frequency second order orientation. This is equivalent to replacing $M(\mathbf{x})$ with $M(\mathbf{x}) * G_0(\mathbf{x})$, where $G_0(\mathbf{x})$ represents a Gaussian blurring kernel; in the frequency domain, this corresponds to weighting the mask $\hat{M}(\mathbf{w})$ by a Gaussian. An example is shown in Figure 5(b).

An arbitrary surface $i_1(x)$ results in second order orientation at other frequencies along the line $w_x v_1 + w_t = 0$ in the frequency domain. Figure 5(c) shows an example where rightward moving uniform noise is occluded by leftward moving noise. Now, the information in $M(\mathbf{x})$ is *modulated* to *all* frequencies $\vec{w} = [w_x, w_t]^T$ along the line $w_x v_1 + w_t = 0$. Two additional examples are present in Figure 5(d,e).

A variety of other stimuli also result in second order orientation; algorithms which detect kinetic occlusion may also respond to other kinds of stimuli which also possess locally oriented power in the frequency domain. Beats, theta motion, multiplicative transparency, and drift-balanced stimuli also contain locally oriented power [9]. Because the algorithms we will develop later detect second order orientation, they will also respond to these stimuli; it is possible the visual system analyzes these second-order stimuli with the same mechanisms that analyze kinetic occlusion.

4.2 XT Kinetic occlusion: Analysis

Having made these initial observations, we can now state our goal precisely. The task of *first order* orientation analysis is to deduce as much information as possible about the layers $I_n(\mathbf{x})$; in motion analysis, this involves estimation of surface velocities $\vec{v}_n(\mathbf{x})$.

The task of *second order* orientation analysis is to deduce as much information as possible about the attenuation mask $M(\mathbf{x})$. In kinetic occlusion analysis, this involves: (1) detection of locally oriented power, thus validating the translating step edge model $M(\mathbf{x}) = u(x + v_m t)$; (2) determining the second order motion v_m.

4.2.1 First order orientation analysis

Estimation of the first order motion(s) v_n is possible by finding the line(s) $w_x v_n + w_t = 0$ containing the most energy. Spatiotemporally oriented filters allow one to selectively extract portions of the frequency domain so as to accomplish this task [1, 13, 36]. Phase-invariant energy extraction is possible through usage of a quadrature pair of filters $G_\theta(\mathbf{x})$ and $H_\theta(\mathbf{x})$ tuned to orientation θ: [1, 10]

$$E_\theta(\mathbf{x}) = (I(\mathbf{x}) * G_\theta(\mathbf{x}))^2 + (I(\mathbf{x}) * H_\theta(\mathbf{x}))^2 \qquad (9)$$

Multiple motions at each point in space-time $\mathbf{x}$ can be found by finding local peaks in $E_\theta(\mathbf{x})$ as θ is varied. Figure 6 illustrates the various stages of this calculation applied to a stimulus containing two occluding surfaces. Filtering $I(\mathbf{x})$ with $G_\theta(\mathbf{x})$ and $H_\theta(\mathbf{x})$ *selects* a portion of the frequency domain centered around $\pm\vec{w}_0$; squaring in the spatial domain *demodulates* the information present around $\pm\vec{w}_0$ to the center $\vec{w} = 0$, but creates high frequency components at $\pm 2\vec{w}_0$; however, because the filters are in quadrature, high frequency components induced by squaring are canceled when the two signals are summed.[2] Through this calculation, second order orientation centered around $\vec{w}_0$ is *demodulated* to the center of the frequency domain $\vec{w} = 0$ in $E_\theta(\mathbf{x})$.

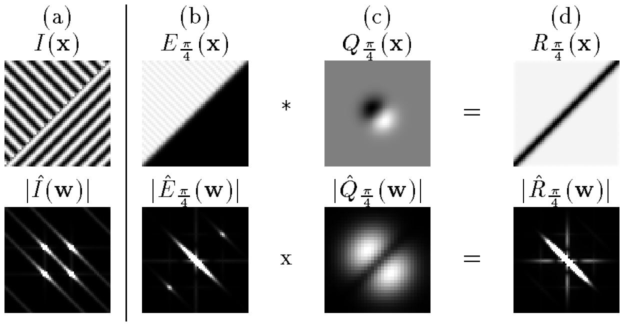

Figure 7: *XT Second order orientation extraction I.* (a) The stimulus $I(\mathbf{x})$ depicts a rightward moving sine grating occluded by a leftward moving sine grating. (b) Shown here is first order orientation $E_{\frac{\pi}{4}}(\mathbf{x})$ extracted at orientation $\frac{\pi}{4}$ and the magnitude of its FFT $|\hat{E}_{\frac{\pi}{4}}(\mathbf{w})|$. (c) Filter $Q_{\frac{\pi}{4}}(\mathbf{x})$ used to remove first order orientation; it is shown enlarged for visibility. (d) Second order orientation can be detected through removal of first order orientation. Shown here is the detected second order orientation $R_{\frac{\pi}{4}}(\mathbf{x})$ and the magnitude of its FFT $|\hat{R}_{\frac{\pi}{4}}(\mathbf{w})|$; responses are negative, indicating deletion of surface texture.

Gabor functions and derivatives of Gaussians are particularly convenient forms that are popularly used for orientation analysis. Gabor functions may be preferred because they form a natural quadrature pair and exactly "demodulate" portions of the frequency domain; we use derivatives of Gaussians in our experiments because they are steerable [10].

4.2.2 Second order orientation analysis

The key idea behind our method of *second order* orientation analysis is as follows. First order orientation analysis "demodulates" a portion of the frequency domain to compute $E_\theta(\mathbf{x})$; any second order orientation previously centered around $\pm\vec{w}_0$ is now "demodulated" to $\vec{w} = 0$ in $\hat{E}_\theta(\mathbf{w})$, effectively becoming first order orientation. So to detect *second order* motion in $I(\mathbf{x})$, we can detect *first order* motion in $E_\theta(\mathbf{x})$. And, to estimate *second order* motion v_m in $I(\mathbf{x})$, we can estimate *first order* motion in $E_\theta(\mathbf{x})$.

An example of the demodulation of second order orientation is shown in Figure 7(a,b). There is now power lying along the line $w_x v_m + w_t = 0$ *and* $w_x v_1 + w_t = 0$ in $\hat{E}_\theta(\mathbf{w})$. Given estimates of the first order orientation v_1, the first order orientation can be removed with linear filtering to *isolate* second order orientation $R_\theta(\mathbf{x})$:

$$R_\theta(\mathbf{x}) = Q_\theta(\mathbf{x}) * E_\theta(\mathbf{x}) \qquad (10)$$

where $Q_\theta(\mathbf{x})$ represents a linear filter with the property $\hat{Q}_\theta(w_x, w_t) = 0$ for all $\vec{w} = [w_x, w_t]^T$ on the line $w_x v_1 +$

[2] Perfect cancellation does not occur because $H_\theta(\mathbf{x})$ is a steerable *approximation* to the Hilbert transform of $G_\theta(\mathbf{x})$.

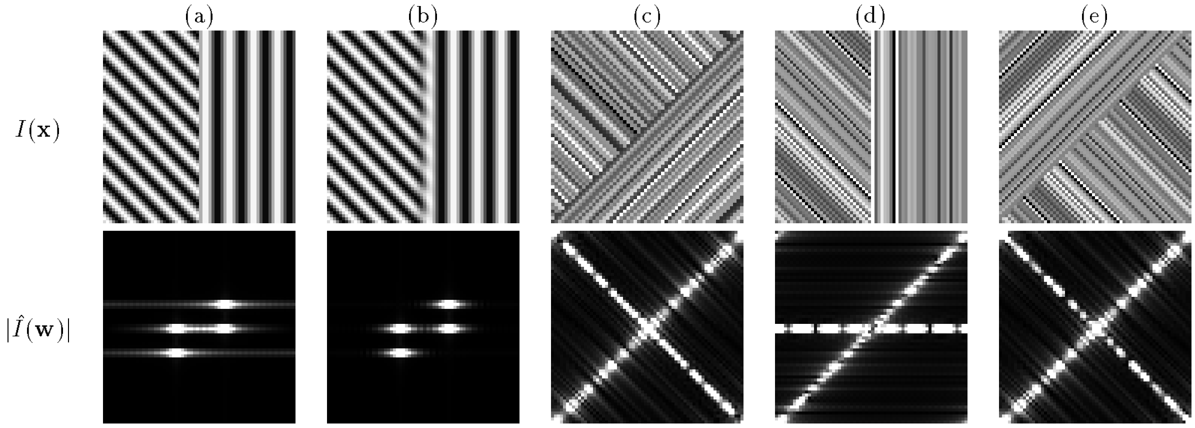

Figure 5: *XT Second order orientation: Kinetic occlusion.* (a) If the motion of the occluding surface ($v_m = v_2$) changes, the second order orientation also changes. (b) Blurring the edge $M(\mathbf{x})$ merely attenuates high frequency second order orientation. (c) With arbitrary surfaces, second order orientation is present everywhere along the line $w_x v_1 + w_t = 0$. Here a rightward moving surface ($v_1 = 1$) moves behind a leftward moving surface ($v_2 = v_m = -1$). (d) Here a rightward moving surface ($v_1 = 1$) moves behind a static surface ($v_2 = v_m = 0$). (e) Like (c), a rightward moving surface ($v_1 = 1$) moves behind a leftward moving surface ($v_2 = v_m = -1$). But unlike (c), now information is accreted at the motion boundary.

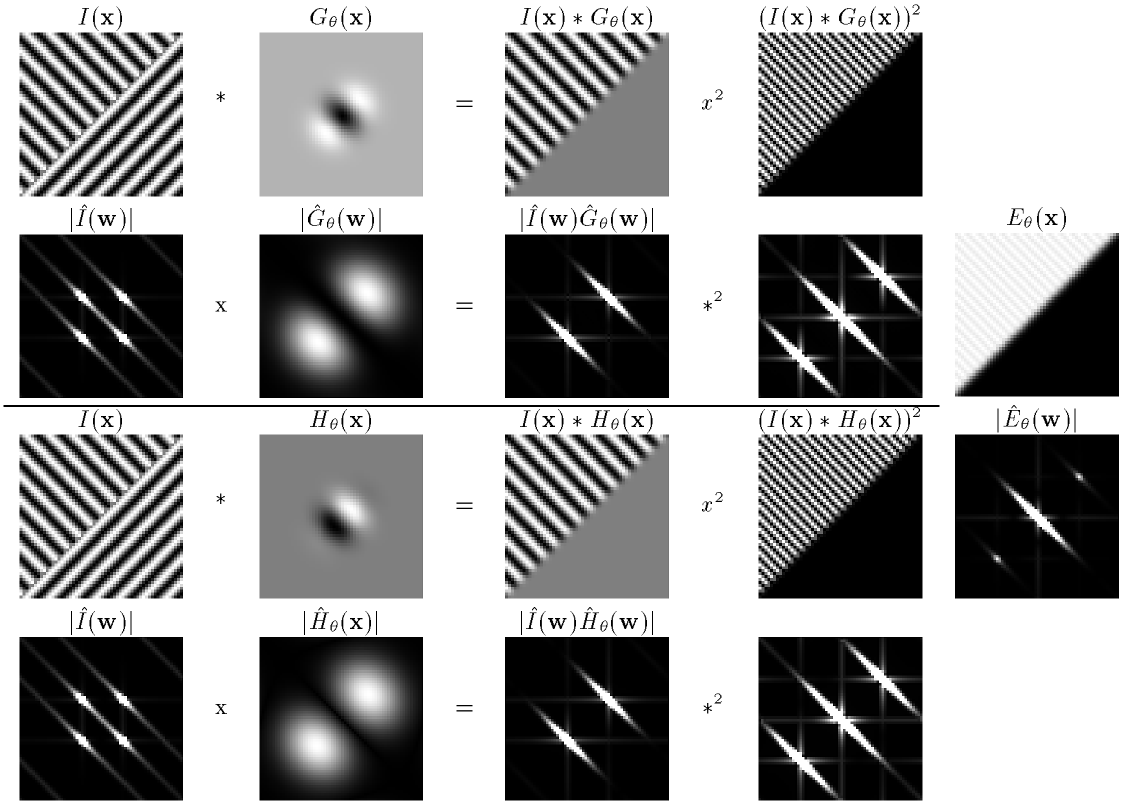

Figure 6: *XT First order orientation energy extraction.* Here we depict the demodulation of information present at $\pm\vec{w}_0$ to the center of the frequency domain, $\vec{w} = 0$, via computation of $E_\theta(\mathbf{x}) = (I(\mathbf{x}) * G_\theta(\mathbf{x}))^2 + (I(\mathbf{x}) * H_\theta(\mathbf{x}))^2$. Here, $\theta = \frac{\pi}{4}$, and rightward motion is extracted.

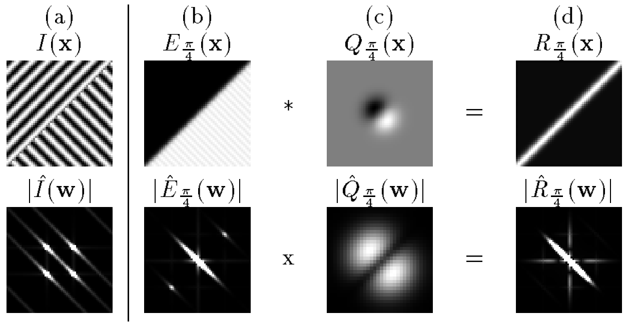

Figure 8: *XT Disocclusion vs. occlusion.* (a) "Dual" of stimulus $I(\mathbf{x})$ in the previous figure; (b) First order orientation extracted $E_{\frac{\pi}{4}}(\mathbf{x})$; (c) Filter $Q_{\frac{\pi}{4}}(\mathbf{x})$ used to remove first order orientation, enlarged for visibility; (d) Second order orientation extracted $R_{\frac{\pi}{4}}(\mathbf{x})$; responses are positive, indicating accretion of surface texture.

$w_t = 0$. There are a variety of filters that can be used; the most convenient is a first derivative of a Gaussian:

$$Q_\theta(\mathbf{x}) = \sin(\theta) G_x(\mathbf{x}) + \cos(\theta) G_t(\mathbf{x}) \tag{11}$$

where $G_x(\mathbf{x})$ and $G_t(\mathbf{x})$ are derivatives of a Gaussian in x and t respectively. Here we used the steerability property of derivatives of Gaussians [10] to filter *orthogonally* to the direction of spatiotemporal orientation.[3] With first order orientation removed, only second order information remains in $R_\theta(\mathbf{x})$. In principle, methods to estimate first order motion can now be applied to $R_\theta(\mathbf{x})$ to estimate v_m. In our example, we can detect the second order orientation present in the stimulus of Figure 7(a) by filtering $E_{\frac{\pi}{4}}(\mathbf{x})$ with $Q_{\frac{\pi}{4}}(\mathbf{x})$ to yield $R_{\frac{\pi}{4}}(\mathbf{x})$, shown in Figure 7(b), (c), and (d) respectively. First order orientation is removed; consequently $R_{\frac{\pi}{4}}(\mathbf{x})$ has significant responses near the moving occlusion boundary, indicating the presence of second order orientation.

For our particular choice of filter $Q_\theta(\mathbf{x})$, by observing the *sign* of the response of $R_\theta(\mathbf{x})$, we can differentiate occlusion from disocclusion. Occlusion, i.e. deletion of surface texture, is reflected by negative responses, and disocclusion, i.e. accretion of surface texture, is reflected by positive responses. If we change $M(\mathbf{x}) = u(x + v_m t)$ to $M(\mathbf{x}) = u(-x - v_m t)$, occlusions become disocclusions and vice versa. A "dual" for the occlusion stimulus of Figure 7(a) is shown in Figure 8(a). The first and second order orientation extracted via the above procedure is shown in Figure 8(b,c,d). The detected second order orientation $R_\theta(\mathbf{x})$ changes sign accordingly, reflecting the interposition.

For a more complicated stimulus, there is second order orientation modulated *everywhere* along the line $w_x v_1 + w_t = 0$; a multiscale analysis is appropriate. Figure 9(a) shows an example of one such stimulus, Figure 9(b) shows second order orientation demodulated from one particular orientation and scale around $\vec{w} = \vec{w}_0$ in the frequency domain.

Successful kinetic occlusion detection depends crucially on whether second order orientation is demodulated successfully. This in turn depends on the accuracy of first order orientation extraction. If we misestimate first order orientation v_1, then the second order orientation is not demodulated to the center of the frequency domain $\vec{w} = 0$; false responses will be detected. Figure 10(c) illustrates the effect of superimposing a misestimated first orientation onto our stimulus of Figure 10(a). Because of the misestimation of the velocity v_1, power along the line $w_x v_1 + w_t = 0$ is not successfully demodulated and first order orientation is not successfully removed.

Furthermore, *only* the information present along the line $w_x v_1 + w_t = 0$ should be demodulated. If the lines $w_x v_1 + w_t = 0$ and $w_x v_2 + w_t = 0$ are sufficiently close that $E_{\theta_1}(\mathbf{x})$ or $E_{\theta_2}(\mathbf{x})$ extracts *both* lines, then filtering with $Q_{\theta_1}(\mathbf{x})$ or $Q_{\theta_2}(\mathbf{x})$ will result in false responses. Figure 10(a,b,c) shows a stimulus with dominant orientations ($\theta_1 = \frac{\pi}{4}, \theta_2 = 0$) sufficiently close to one another that energy extracted at either orientation yields significant response at the other orientation. Removal of first order orientation yields significant responses. While this problem may be solved in part by extracting energy with filters more sharply tuned to orientation, this would require more oriented filters; in practice, some overlap is always to be expected. Pursuit eye movements may also aid in separating the two components.

For successful removal of first order orientation, it is necessary to have a suppression mechanism which attenuates energy extracted from non-dominant motions. Only motion which is locally dominant can be made to respond with a simple non-linear transformation, although in doing so we can no longer obtain a strict frequency domain interpretation. Rather than filtering $E_\theta(\mathbf{x})$ to compute $R_\theta(\mathbf{x})$, we can filter a transformed version of it, $\tilde{E}_\theta(\mathbf{x})$:

$$R_\theta(\mathbf{x}) = Q_\theta(\mathbf{x}) * \tilde{E}_\theta(\mathbf{x}) \tag{12}$$

where $\tilde{E}_\theta(\mathbf{x})$ represents a signal which only responds if it is a dominant orientation. Many mechanisms may be devised. One possible scheme, which we use, is:

$$\tilde{E}_\theta(\mathbf{x}) = f\left(\frac{E_\theta(\mathbf{x})}{\sigma + \sum_{\delta\theta} E_{\theta+\delta\theta}(\mathbf{x})}\right) \tag{13}$$

where $f(x)$ is a sigmoid function:

$$f(x) = \frac{1}{1 + e^{-c(x - \tilde{x})}} \tag{14}$$

which is near 1 if $x \gg \tilde{x}$ and near 0 if $x \ll \tilde{x}$. The above operations implement a local soft winner-take-all mechanism, i.e. $\tilde{E}_\theta(\mathbf{x})$ only responds if $E_\theta(\mathbf{x})$ is greater than the energy extracted at neighboring orientations $E_{\theta+\delta\theta}(\mathbf{x})$. A neighborhood of $\delta\theta = \{0, \pm\theta_0\}$ is sufficient to yield the desired responses, where $\tilde{x} = \frac{1}{3} + \epsilon$, and c is adjusted to have desirable cutoff response properties. At high contrasts, division by neighboring orientations results in contrast invariance. At low contrasts, σ dominates, so the response will be near zero.

The above operation can be performed at every orientation θ. While *all* orientations θ can be extracted with the

[3] With a first derivative of a Gaussian, two "basis" filters $G_x(\mathbf{x})$ and $G_t(\mathbf{x})$ are used, with interpolation functions $\sin(\theta)$ and $\cos(\theta)$ steering to any orientation θ. See [10] for details.

above operations, simulations may be simplified by only extracting second order orientation at k dominant orientations. We can utilize mechanisms which find the dominant orientations for one-dimensional stimuli [26] or which find the dominant motions in two-dimensional stimuli [18, 35]. We should be aware that significant deviations from the dominant motion will result in false responses. The systematic nature of these false responses can be taken advantage of, as some spatial junction analysis schemes have done [14, 28], but we do not pursue this possibility here. Our overall block diagram for processing one-dimensional stimuli is shown in Figure 11. At each orientation θ_n, where $n = 1 \ldots k$, we extract energy signals $\tilde{E}_{\theta_n}(\mathbf{x})$, which when filtered with $Q_{\theta_n}(\mathbf{x})$ yield occlusion detection signals $R_{\theta_n}(\mathbf{x})$. The representation of detected occlusion $R_{\theta_n}(\mathbf{x})$ is adequate, but can be pooled over all k dominant orientations to summarize detected occlusion boundaries with simple summation:

$$R(\mathbf{x}) = \sum_{n=1}^{k} R_{\theta_n}(\mathbf{x}) \tag{15}$$

In pooling these responses, we lose information about which moving surface is occluded, so the distributed representation $R_{\theta_n}(\mathbf{x})$, where $n = 1 \ldots k$, may be preferred for subsequent processing.

We illustrate on our example of Figure 12(a), where there are two dominant orientations, $\theta_1 = \frac{\pi}{4}$ and $\theta_2 = 0$, corresponding to the occluded and occluding surface respectively. The energy $E_{\theta_n}(\mathbf{x})$ for both orientations is shown in Figure 12(b); responses to locally non-dominant orientations are significant. The transformed signals for both orientations are shown in Figure 12(c). Now, responses in $\tilde{E}_{\theta_n}(\mathbf{x})$ are present only in regions where the orientation is dominant. Filtering these signals with $Q_{\theta_n}(\mathbf{x})$ results in significant response in regions where there is occlusion. The signal $R_{\theta_1}(\mathbf{x})$ indicates occlusion at the motion boundary for rightward motion, while the signal $R_{\theta_2}(\mathbf{x})$ only generates spurious responses because static motion is not occluded. Figure 12(d) shows the summarized response $R(\mathbf{x})$ pooled across both dominant orientations. Second order orientation extracted from the occluded surface, $R_{\frac{\pi}{4}}(\mathbf{x})$, dominates the response in $R(\mathbf{x})$.

The basic concepts of modulation of information in our image formation model and demodulation in our kinetic occlusion analysis method can easily be extended to two-dimensional motion analysis. We consider kinetic occlusion of two-dimensional moving surfaces in the next section.

5 Two-dimensional Kinetic Occlusion

Thus far we have understood kinetic occlusion in one-dimensional spatiotemporal stimuli, and have motivated the basic operations which allow for the local detection of kinetic occlusion. We now extend the analysis to two dimensional moving patterns. The occlusion model in two dimensions is exactly as before, but now $\mathbf{x} = (x, y, t)$ and $\mathbf{w} = (w_x, w_y, w_t)$:

$$I(\mathbf{x}) = I_1(\mathbf{x})M(\mathbf{x}) + I_2(\mathbf{x})(1 - M(\mathbf{x})) \tag{16}$$

$$\hat{I}(\mathbf{w}) = \hat{I}_1(\mathbf{w}) * \hat{M}(\mathbf{w}) + \hat{I}_2(\mathbf{w}) * (\delta(\mathbf{w}) - \hat{M}(\mathbf{w})) \tag{17}$$

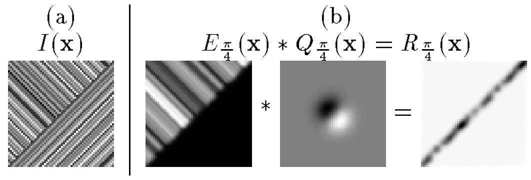

Figure 9: *XT Second order orientation analysis I.* (a) Generic occlusion stimulus $I(\mathbf{x})$, representing a noise pattern moving rightward being occluded by a noise pattern moving leftward. (b) Second order orientation extracted for the generic occlusion stimulus $I(\mathbf{x})$. Second order orientation is detected at the motion boundary; the response is negative, indicating deletion of texture of the rightward moving surface.

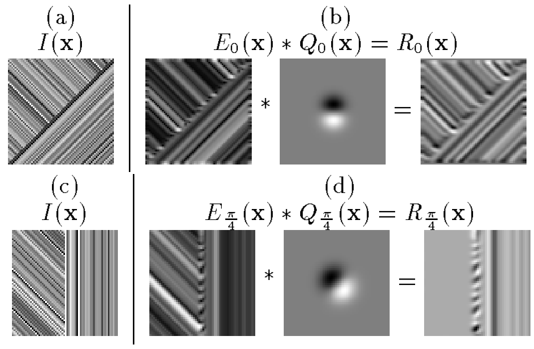

Figure 10: *XT Second order orientation analysis II.* (a) Generic occlusion stimulus $I(\mathbf{x})$, representing a rightward moving noise pattern being occluded by a leftward moving noise pattern. (b) Filtering extracted first order orientation $E_0(\mathbf{x})$ with $Q_0(\mathbf{x})$ detects occlusion of static surfaces, but only if there is static orientation present; $Q_0(\mathbf{x})$ is shown enlarged for visibility. Anomalous responses are everywhere in $R_0(\mathbf{x})$, due to imposed misestimate $v_1 = 0$. (c) Generic occlusion stimulus $I(\mathbf{x})$, representing a noise pattern moving rightward behind a static noise pattern. (d) Second order orientation extracted for this signal are spuriously positive and negative because of interference from neighboring orientations.

Layers $I_n(\mathbf{x})$ are two-dimensional spatial surfaces $i_n(x, y)$ which move with velocity $\vec{v}_n = [v_{xn}, v_{yn}]^T$:

$$I_n(\mathbf{x}) = i_n(x + v_{xn}t, y + v_{yn}t) \tag{18}$$

Unambiguously translating surfaces have power constrained to lie along planes in the frequency domain [13]:

$$\hat{I}_n(\mathbf{w}) = \hat{i}_n(w_x, w_y)\delta(w_x v_{xn} + w_y v_{yn} + w_t) \tag{19}$$

5.1 XYT Kinetic occlusion: Model

Kinetic occlusion is modeled with a spatially oriented step edge $u_{\theta_m}(x, y)$ translating with speed v_m and oriented at angle θ_m:

$$M(\mathbf{x}) = u_{\theta_m}(x + v_m \cos(\theta_m)t, y + v_m \sin(\theta_m)t) \tag{20}$$

$$\hat{M}(\mathbf{w}) = \hat{u}_{\theta_m}(w_x, w_y)\delta(w_x v_m \cos(\theta_m) + w_y v_m \sin(\theta_m) + w_t) \tag{21}$$

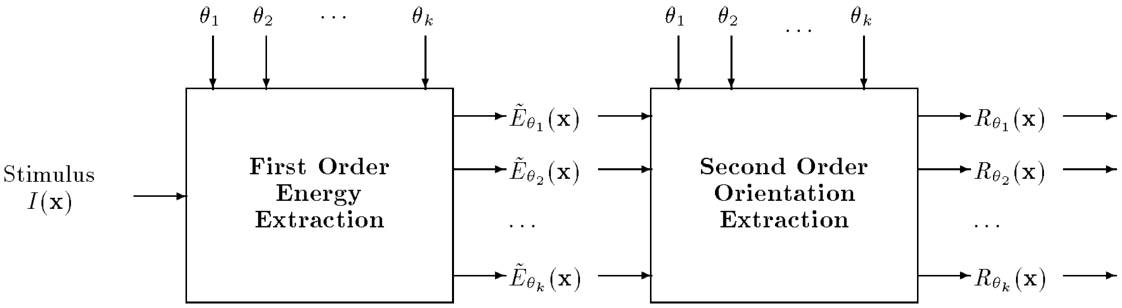

Figure 11: *XT Kinetic occlusion detection: Block diagram.* One dimensional kinetic occlusion stimuli are analyzed via a two stage process. The first stage extracts inhibited energy signals $\tilde{E}_{\theta_n}(\mathbf{x})$ for k orientations θ_n, where $n = 1 \ldots k$. The second stage removes first order orientation by filtering $\tilde{E}_{\theta_n}(\mathbf{x})$ with derivative of Gaussians $Q_{\theta_n}(\mathbf{x})$ to result in occlusion detection signals $R_{\theta_n}(\mathbf{x})$.

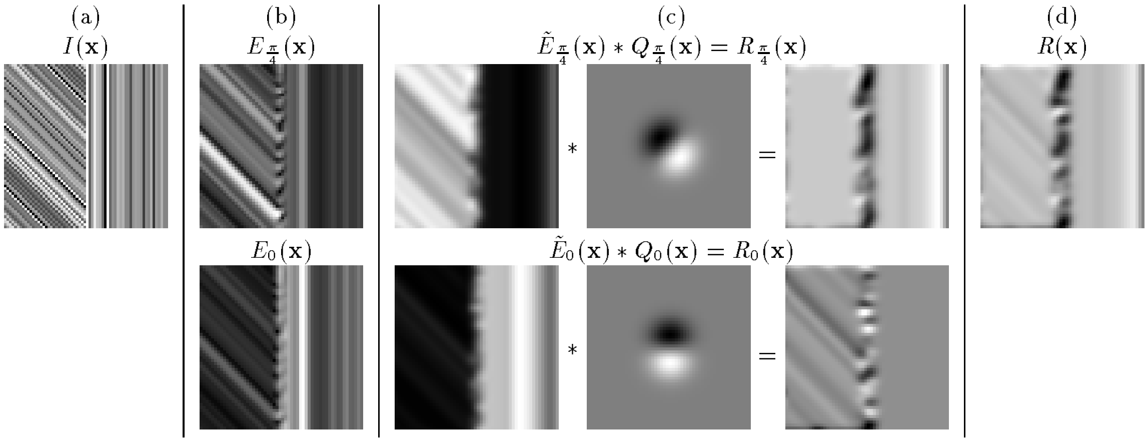

Figure 12: *XT Kinetic occlusion detection: Example.* (a) Generic occlusion stimulus $I(\mathbf{x})$, representing a noise pattern moving rightward behind a static noise occluder. (b) First order orientation extracted for both orientations. (c) Extraction of second order orientation $R_{\theta_n}(\mathbf{x})$ from signals $\tilde{E}_{\theta_n}(\mathbf{x})$ which respond only where motion is locally dominant. (d) Pooling across all orientations can be used to summarize detected occlusion; the response is negative at the motion boundary, indicating deletion of surface texture.

It is well known that the motion of an isolated moving edge is ambiguous; only the normal velocity $\vec{v}_m^n = v_m \vec{n}_{\theta_m}$ can be measured, where $\vec{n}_{\theta_m} = [\cos(\theta_m), \sin(\theta_m)]^T$. An unobservable tangent component $\vec{v}_m^t$ may also be present, where $\vec{v}_m = \vec{v}_m^n + \vec{v}_m^t$. A geometric decomposition of $\vec{v}_1$, $\vec{v}_m$ and $\vec{v}_2$ is depicted in Figure 13(a), where $\vec{v}_1^n = \vec{v}_1 \cdot \vec{n}_{\theta_m}$ and $\vec{v}_1 = \vec{v}_1^n + \vec{v}_1^t$. In our model, the mask in general will have a motion consistent with the occluder's motion, where $\vec{v}_m = \vec{v}_2$. However, generically, $\vec{v}_m^n \neq \vec{v}_2$.

Since $\hat{u}_{\theta_m}(w_x, w_y) = \hat{u}(r)\delta(-w_x \sin(\theta_m) + w_y \cos(\theta_m))$, the power in $\hat{M}(\mathbf{w})$ lies along a line defined by the intersection of two planes:

$$[v_m \cos(\theta_m), v_m \sin(\theta_m), 1]^T \cdot \vec{w} = 0$$

$$[-\sin(\theta_m), \cos(\theta_m), 0]^T \cdot \vec{w} = 0$$

Thus $\hat{M}(\mathbf{w})$ lies along the line $\vec{w}_m(k)$, parametrized by k:

$$\vec{w}_m(k) = k[\cos(\theta_m), \sin(\theta_m), -v_m]^T \qquad (22)$$

Thus as the *orientation* θ_m of the mask $M(\mathbf{x})$ changes, the power in $\hat{M}(\mathbf{w})$ rotates about the w_t axis; this is depicted in Figure 13(b). As the *speed* v_m of the mask $M(\mathbf{x})$ changes, the power is skewed about the (w_x, w_y) plane; this is depicted in Figure 13(c).

Using the above mask model, we can understand the image formation process in both domains. Figure 14 depicts an example occluding stimulus pattern. In the spatial domain, a rightward moving plaid ($i_1(x,y) = \sin(w_{x1a}x + w_{y1a}y + w_{t1a}) + \sin(w_{x1b}x + w_{y1b}y + w_{t1b})$, $\vec{v}_1 = [1, 0]^T$) is occluded by another plaid moving leftward ($i_2(x,y) = \sin(w_{x2a}x + w_{y2a}y + w_{t2a}) + \sin(w_{x2b}x + w_{y2b}y + w_{t2b})$, $\vec{v}_m^n = \vec{v}_2 = [-1, 0]^T$). Each plaid is composed of two sinusoids whose normal motions are consistent with rightward and leftward motion respectively. Thus $\hat{I}_1(\mathbf{w})$ contains four impulses that lie along the plane $w_x v_{x1} + w_y v_{y1} + w_t = 0$, as shown in Figure 14(a). The mask $\hat{M}(\mathbf{w})$ appears as a line, as shown in Figure 14(b). Multiplying the two pat-

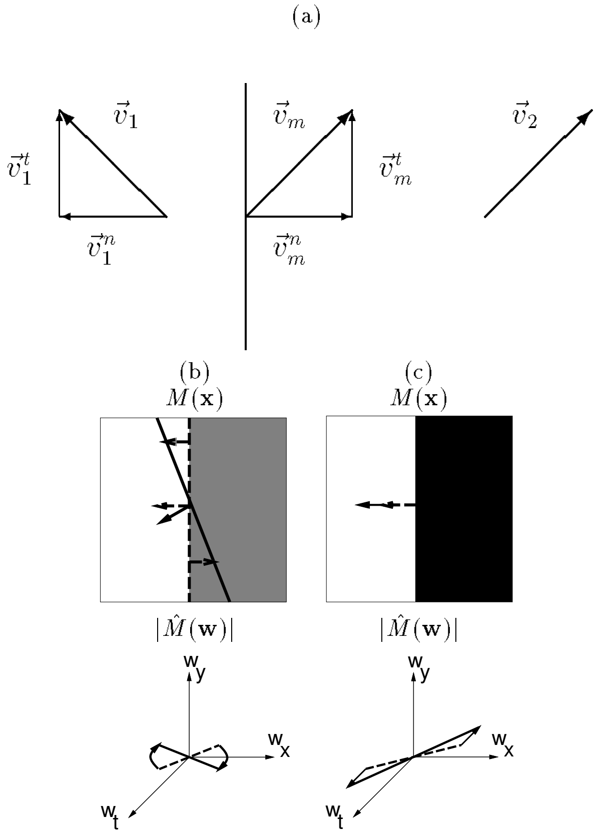

Figure 13: *XYT Mask model $M(\mathbf{x})$.* (a) A geometric decomposition of the velocity vectors of the occluded and occluding surface $I_1(\mathbf{x})$ and $I_2(\mathbf{x})$ and mask $M(\mathbf{x})$; (b) As the mask changes orientation θ_m, the oriented power rotates about the w_t axis; (c) As the mask changes speed v_m, the oriented power does not rotate, but instead skews to and from the (w_x, w_y) plane.

terns $I(\mathbf{x})$ and $M(\mathbf{x})$ in space-time is the same as convolving their spectra in the frequency domain. This results in *locally oriented power* around the coordinates of the impulses of $\hat{I}_1(\mathbf{w})$, as shown in Figure 14(c). The orientation of the power in $\hat{M}(\mathbf{w})$ is *modulated* off the center to the coordinates of the impulses in $\hat{I}_1(\mathbf{w})$.

As before, *first order* motion is manifested as power passing through the origin $\vec{w} = 0$; *second order* motion is manifested as oriented power centered off the origin around $\vec{w} = \pm\vec{w}_0$. Since the mask $M(\mathbf{x})$ moves consistently with the occluder $I_2(\mathbf{x})$, the second component $I_2(\mathbf{x})(1 - M(\mathbf{x}))$ does not introduce any observable second order motion, as shown in Figure 14(d,e,f); all the power in $\hat{I}_2(\mathbf{w}) * (\delta(\mathbf{w}) - \hat{M}(\mathbf{w}))$ lies inside the plane $w_x v_{x2} + w_y v_{y2} + w_t = 0$. The stimulus $I(\mathbf{x})$ is just the sum of the two components in Figure 14(c) and (f), and is shown in Figure 14(g).

If the mask $M(\mathbf{x})$ is moving consistently with the occluded surface $I_1(\mathbf{x})$, then the occlusion is hidden. Refering to Figure 13(a), clearly the mask is moving consistently with the occluded surface if and only if $\vec{v}_1 = \vec{v}_m$. Under such conditions, the power in $\hat{M}(\mathbf{w})$ is a line contained *within* the plane $w_x v_{x1} + w_y v_{y1} + w_t = 0$. There are the two possible extremes. At one extreme, $\vec{v}_m^n = \vec{v}_1^n = \vec{v}_1$ and $\vec{v}_1^t = 0$, and there is no perceived motion boundary; it is illustrated in Figure 15(a). The second extreme, illustrated in Figure 15(b), corresponds to motion *shear*. Under such conditions, $\vec{v}_m^n = \vec{v}_1^n = 0$ and $\vec{v}_1^t = \vec{v}_1$. There is a perceived motion boundary, but no cue to occlusion! A case midway between these two extremes is illustrated in Figure 15(c), where $\vec{v}_1 = \vec{v}_m^n + \vec{v}_m^t$ and $\vec{v}_m^n \neq 0$ and $\vec{v}_m^t \neq 0$. In all three cases, nothing is accreted or deleted, there is no ordinal cue to depth, and there is no second order orientation to extract. However, with considerable deviation from the plane $w_x v_{x1} + w_y v_{y1} + w_t = 0$, the second order orientation is visible, and there is an ordinal cue to depth that can be detected.

With a more complicated spatial surface $i_1(x, y)$, information in $\hat{M}(\mathbf{w})$ is modulated to frequencies *everywhere* along the plane $w_x v_{x1} + w_y v_{y1} + w_t = 0$.

5.2 XYT Kinetic occlusion: Analysis

Having made these observations, we can now define the goal of kinetic occlusion analysis in image sequences.

The task of *first order* motion analysis is to deduce as much information as possible about $I_n(\mathbf{x})$. In motion analysis, this involves estimation of surface velocities $\vec{v}_n(x)$. This is a well understood problem, and there are now numerous motion estimation algorithms available. First order motion analysis in two dimensions involves determining the plane(s) which contain the most energy [8, 11, 13].

The task of *second order* motion analysis is to deduce as much information as possible about $M(\mathbf{x})$. For kinetic occlusion analysis of two-dimensional motion, this involves: (1) detection of second order motion; (2) estimating the parameters of the mask $M(\mathbf{x})$, i.e. the orientation θ_m and speed v_m. Analogously to methods developed for (x, t) analysis, we can detect kinetic occlusion and estimate second order motion in $I(\mathbf{x})$ by: (1) extracting first order motion (i.e. demodulating second order orientation to $\vec{w} = 0$) to compute $E_{\vec{v}}(\mathbf{x})$; In one dimension, where $\mathbf{x} = (x, t)$, we used oriented energy measures to demodulate energy contained along the line $w_x v + w_t = 0$ to $\vec{w} = 0$. In two dimensions, where $\mathbf{x} = (x, y, t)$, we must demodulate energy contained in the *plane* $v_x w_x + v_y w_y + w_t = 0$ to $\vec{w} = 0$ in (x, y, t). (2) applying *first order* motion analysis algorithms to $E_{\vec{v}}(\mathbf{x})$ to analyze *second order* motion of $I(\mathbf{x})$.

5.2.1 First order motion analysis

For a given velocity $\vec{v} = [v_x, v_y]^T$ that contains power along the plane $v_x w_x + v_y w_y + w_t = 0$, extracting first order motion $E_{\vec{v}}(\mathbf{x})$ is possible through linear filtering of the stimulus $I(\mathbf{x})$ with banks of spatiotemporal filters centered on the plane [11, 13]. Simoncelli [32] has recently developed methods of doing so using tools of steerable filtering [10, 21]. We summarize the method below.

Convolution of an image sequence $I(\mathbf{x})$ with a set of basis filters $G_n(\mathbf{x})$ allows for "steerable" filtering to any portion of the frequency domain $\vec{w} = \pm[w_1, w_2, w_3]^T$, where $|\vec{w}| = 1$, with interpolation functions $f_n(\vec{w})$:

$$F_{\vec{w}}(\mathbf{x}) = \sum_n f_n(\vec{w})(G_n(\mathbf{x}) * I(\mathbf{x})) \tag{23}$$

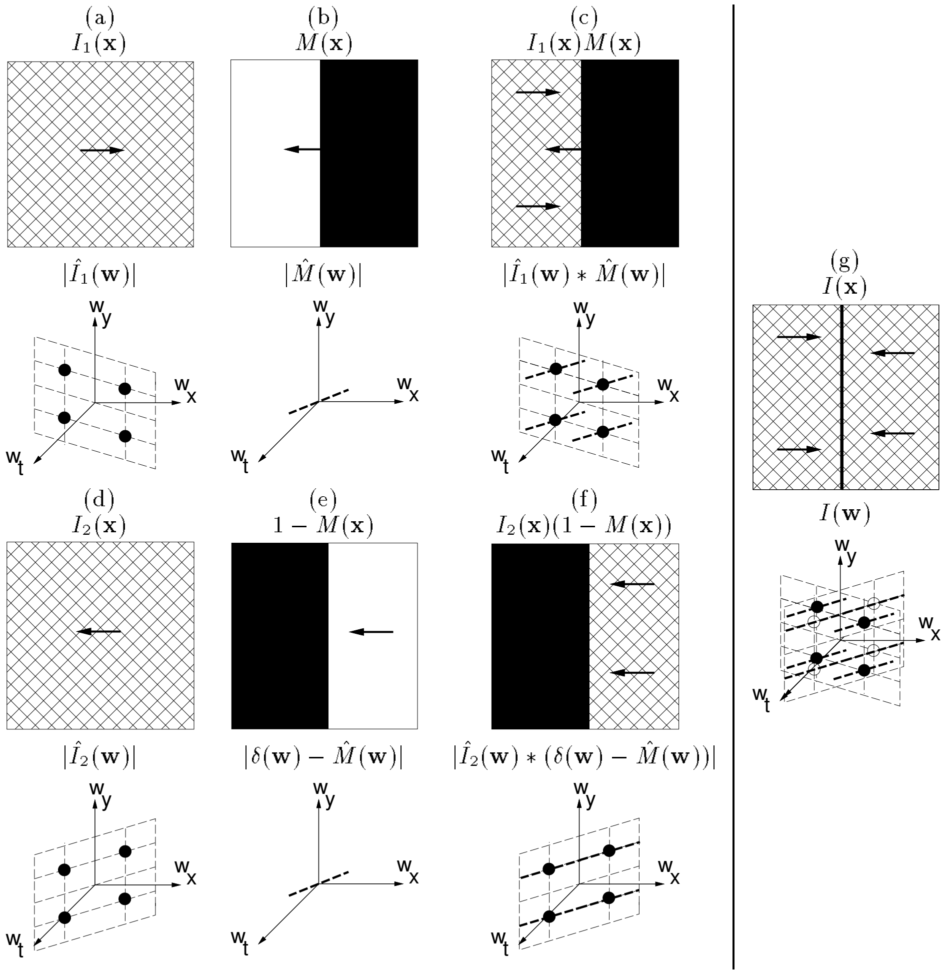

Figure 14: *XYT Kinetic occlusion: Image formation.* Multiplying oriented patterns in the spatiotemporal domain results in locally oriented power in the frequency domain.

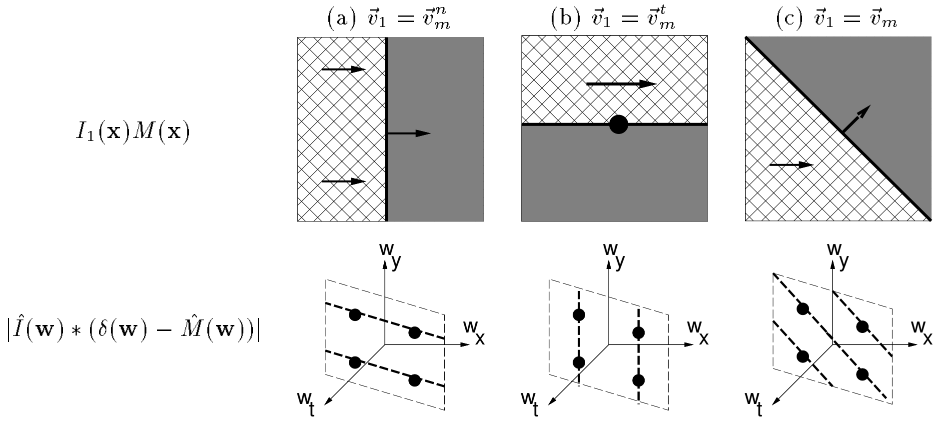

Figure 15: *XYT Degenerate cases.* If the mask $M(\mathbf{x})$ moves consistently with the occluded surface, then $\vec{v}_1 = \vec{v}_m$, and the occlusion is hidden. Under such conditions, the power in $\hat{M}(\mathbf{w})$ is modulated within the plane $w_x v_{x1} + w_y v_{y1} + w_t = 0$ and there is no second order orientation to extract. (a) Extreme 1: Here, the normal motion of $\vec{v}_m$ is identical to $\vec{v}_1$. (b) Extreme 2: Here, the normal motion of $\vec{v}_m$ is 0, and the tangential component is identical to $\vec{v}_1$. This corresponds to motion shear. (c) In between the two extremes, there exists a continuum of degenerate motions where occlusion cannot be detected.

To tile the plane $w_x v_x + w_y v_y + w_t = 0$, energy is extracted in several portions of the frequency domain centered at $\vec{w} = [w_1, w_2, w_3]^T$ by steering along several different directions and subsequently applying squaring operations:

$$E_{\vec{v}}(\mathbf{x}) = G_0(\mathbf{x}) * \sum_{i=1}^{N+1} (F_{\vec{w}_i}(\mathbf{x}))^2 \tag{24}$$

High frequency components introduced by squaring are diminished through subsequent convolution with a 3-d Gaussian $G_0(\mathbf{x})$. Figure 16 depicts how energy is extracted from a plane in the frequency domain with a first derivative of a 3-d Gaussian; two separate weightings are required to tile the plane. For Nth order derivative of Gaussians, $N+1$ separate weightings of basis filters are necessary. Figure 17(a,b) summarizes steering functions for $N = \{1, 2, 3\}$ and summarizes the filter coefficients used in all of our experiments.

The above procedure retains the key demodulation property considered in (x, t), except now second order information is not demodulated from just one frequency around $\vec{w} = \pm\vec{w}_0$, but instead a *ring* centered on the plane $w_x v_x + w_y v_y + w_t = 0$; choosing a different scale of Gaussian demodulates a different sized ring centered around that plane. An example image sequence with two occluding surfaces is shown as a sliced XYT volume in Figure 18(a); the occluded surface is moving to the left ($\vec{v}_1 = [-1, 0]^T$) and the occluding surface is moving to the right ($\vec{v}_2 = [1, 0]^T$). The first order motion energy extracted for the two motions present, $E_{\vec{v}_1}(\mathbf{x})$ and $E_{\vec{v}_2}(\mathbf{x})$, are shown in Figure 18(b,c). The second order motion is visible in $E_{\vec{v}_1}(\mathbf{x})$ but not in $E_{\vec{v}_2}(\mathbf{x})$, as was illustrated in Figure 14(c) and Figure 14(f).

5.2.2 Second order motion analysis

The previous calculation *demodulates* energy along the plane $v_x w_x + v_y w_y + w_t = 0$ to the center of the frequency domain $\vec{w} = 0$. Second order orientation in $I(\mathbf{x})$ effectively becomes first order orientation in $E_{\vec{v}}(\mathbf{x})$. If there is no second order orientation to extract, then *all* the energy in $\hat{E}_{\vec{v}}(\mathbf{w})$ will pass through the plane $v_x w_x + v_y w_y + w_t = 0$. If there is second order orientation, there will be energy along a second line in $\hat{E}_{\vec{v}}(\mathbf{w})$ corresponding to $\hat{M}(\mathbf{w})$. To isolate second order orientation, we must remove the first order orientation also demodulated in $\hat{E}_{\vec{v}}(\mathbf{w})$, i.e. the energy along the plane $v_x w_x + v_y w_y + w_t = 0$. This is easily done by filtering local measures of first order orientation $E_{\vec{v}}(\mathbf{x})$ with a linear filter $Q_{\vec{v}}(\mathbf{x})$ via:

$$R_{\vec{v}}(\mathbf{x}) = E_{\vec{v}}(\mathbf{x}) * Q_{\vec{v}}(\mathbf{x}) \tag{25}$$

where $\hat{Q}_{\vec{v}}(\mathbf{w}) = 0$ for every $\vec{w}$ such that $\vec{w} \cdot [v_x, v_y, 1]^T = 0$, i.e. zero frequency response along the plane $v_x w_x + v_y w_y + w_t = 0$. A first derivative of a Gaussian is convenient because of its steerability and separability:

$$Q_{\vec{v}}(\mathbf{x}) = \frac{1}{\sqrt{v_x^2 + v_y^2 + 1}} (v_x G_x(\mathbf{x}) + v_y G_y(\mathbf{x}) + G_t(\mathbf{x})) \tag{26}$$

where $G_x(\mathbf{x})$, $G_y(\mathbf{x})$, and $G_t(\mathbf{x})$ are derivatives of 3-d Gaussians in the x, y, and t directions. The second order orientation detected via the above operation in our example of Figure 18(a) is shown in Figure 18(d). For our occluded leftward moving surface, this second operation successfully detects occlusion near the motion boundary. For our particular choice of filter $Q_{\vec{v}}(\mathbf{x})$, the *sign* of the response indicates the nature of the occlusion: negative responses indicate deletion of surface texture; positive responses indicate accretion of surface texture. In our example, the response $R_{\vec{v}_1}(\mathbf{x})$ near the boundary is positive, signaling accretion of surface texture $i_1(x, y)$.

Just as in one-dimensional kinetic occlusion analysis, successful detection of locally oriented power relies on how well energy from *one* motion can be demodulated to the origin $\vec{w} = 0$. *Only* the information present along the plane $w_x v_{x1} + w_y v_{y1} + w_t = 0$ should be demodulated for subsequent filtering with $Q_{\vec{v}_1}(\mathbf{x})$. If information from a second plane $w_x v_{x2} + w_y v_{y2} + w_t = 0$ is sufficiently close to the plane $w_x v_{x1} + w_y v_{y1} + w_t = 0$ of interest, then filtering $E_{\vec{v}_1}(\mathbf{x})$ with $Q_{\vec{v}_1}(\mathbf{x})$ will result in false responses. For successful removal of first order orientation, a suppression mechanism which attenuates energy extracted from non-dominant motions must be designed.

5.3 XYT Kinetic occlusion detection

To suppress non-dominant motions, we construct a signal $\tilde{E}_{\vec{v}}(\mathbf{x})$ which responds only if $\vec{v}$ is present at point $\mathbf{x}$. Given this signal, we can detect occlusion for that motion via:

$$R_{\vec{v}}(\mathbf{x}) = Q_{\vec{v}}(\mathbf{x}) * \tilde{E}_{\vec{v}}(\mathbf{x}) \tag{27}$$

where $Q_{\vec{v}}(\mathbf{x})$ is as before. Only motion which is locally dominant can be made to respond in $\tilde{E}_{\vec{v}}(\mathbf{x})$ with simple non-linear operations on local energy extracted at velocity $\vec{v}$ and neighboring velocities $\vec{v} + \delta\vec{v}$. This can be accomplished with:

$$\tilde{E}_{\vec{v}}(\mathbf{x}) = f\left(\frac{E_{\vec{v}}(\mathbf{x})}{\sigma + \sum_{\delta\vec{v}} E_{\vec{v}+d\vec{v}}(\mathbf{x})}\right) \tag{28}$$

where $f(x)$ is a sigmoid:

$$f(x) = \frac{1}{1 + e^{-c(x-\bar{x})}} \tag{29}$$

The above operation implements a soft local winner-take-all mechanism, i.e. $\tilde{E}_{\vec{v}}(\mathbf{x})$ only responds if the energy extracted at a velocity $\vec{v}$, $E_{\vec{v}}(\mathbf{x})$, is significantly greater than the energy extracted at neighboring motions $\vec{v} + \delta\vec{v}$, $E_{\vec{v}+\delta\vec{v}}(\mathbf{x})$. At low contrasts, σ dominates, and the response will be near zero. At high contrasts, divisive inhibition by energy extracted at neighboring motions results in contrast invariance. Choosing the neighborhood of $\vec{v}+\delta\vec{v}$ is straightforward. The unit normal $\vec{w}_{\vec{v}} = \frac{1}{\sqrt{v_x^2+v_y^2+1}}[v_x, v_y, 1]^T$ to the plane $w_x v_x + w_y v_y + w_t = 0$ is used to find four *neighboring* planes passing through the origin with normals $\vec{w}_{\vec{v}+\delta\vec{v}}$ at fixed angles to $\vec{w}_{\vec{v}}$. Each normal vector $\vec{w}_{\vec{v}+\delta\vec{v}}$ defines a "neighboring" plane whose energy may extracted in the same manner as before.

The above operation can be performed at *every* velocity $\vec{v}$. While it is possible to extract energy for all possible motions $\vec{v}$ with the above operations, simulations may be simplified tremendously by only extracting second order motion for k dominant motions. Our overall architecture is summarized in Figure 19. For each of the k dominant motions $\vec{v}_n(\mathbf{x})$ present in the stimulus, where $n = 1 \ldots k$, first order motion energy signals $\tilde{E}_{\vec{v}_n}(\mathbf{x})$ are extracted. Each of these signals $\tilde{E}_{\vec{v}_n}(\mathbf{x})$ are filtered with $Q_{\vec{v}_n}(\mathbf{x})$ to construct

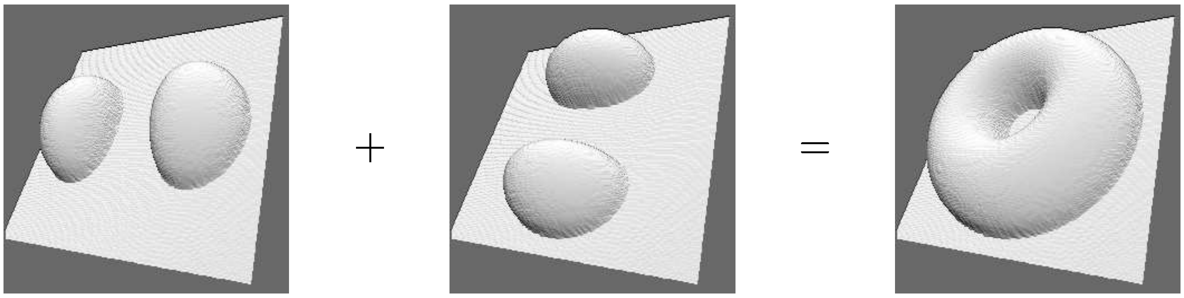

Figure 16: *XYT First order motion energy extraction I.* The energy centered on the plane $w_x v_x + w_y v_y + w_t = 0$ in the frequency domain is extracted by summing the power responses of directional derivatives lying in this plane corresponding to a given velocity. Idealized level surfaces of the power spectra of two such derivative filters are shown here. The level surfaces of the sum of the two will form a smooth ring bisecting the plane. From [32], with permission. Second order orientation that may be contained in this ring is *demodulated* to the origin $\vec{w} = 0$.

occlusion detection signals $R_{\vec{v}_n}(\mathbf{x})$. If desired, we can pool responses over all k dominant motions with simple summation:

$$R(\mathbf{x}) = \sum_{n=1}^{k} R_{\vec{v}_n(\mathbf{x})}(\mathbf{x}) \tag{30}$$

to summarize the occlusion detected in the spatiotemporal stimulus.

In our experiments, dominant motions $\vec{v}_n(\mathbf{x})$ are extracted via the clustering technique used in [35]; the technique uses a k-means clustering algorithm on local affine parameter fits of optical flow to yield k dominant motions described by:

$$\vec{v}_n(x, y, t) = \left[\begin{array}{c} a_x(t) + a_{xx}(t)x + a_{xy}(t)y \\ a_y(t) + a_{yx}(t)x + a_{yy}(t)y \end{array} \right] \tag{31}$$

where the vector $\vec{a}_n(t) = [a_x(t), a_{xx}(t), a_{xy}(t), a_y(t), a_{yx}(t), a_{yy}(t)]^T$ represents affine parameter fits for a cluster at time t. Other recently developed global multiple motion estimation algorithms, e.g. [2, 18], yield similar representations.

We illustrate our approach on a simple example containing two moving surfaces. Figure 20(a) depicts a random-dot stimulus where a rightward moving square occludes a leftward moving background. Figure 20(b) shows three representative spatiotemporal slices of the stimulus, and Figure 20(c) shows first order motion energy $E_{\vec{v}_1}(\mathbf{x})$ extracted for the occluded surface's motion $\vec{v}_1$. Spurious responses are present both inside and outside the square; applying the above linear filtering operations to this signal would result in false detection. Figure 20(d) shows first order motion $\hat{E}_{\vec{v}_1}(\mathbf{x})$ extracted with our local winner-take-all mechanism. Now, responses are present only outside the square. Applying the above filtering operation detects occlusion for leftward motion on both sides of the square in $R_{\vec{v}_1}(\mathbf{x})$, shown in Figure 20(e). On the left side of the square, positive responses indicate that the texture of the background is being accreted. On the right side of the square, negative responses indicate the texture of the posterior surface is being deleted. In the region directly above and below the square, there is a motion boundary but no occlusion, so comparatively little second order orientation is extracted. Identical operations applied to the motion of the occluding surface, $\vec{v}_2$, successfully extracts the square, and is shown in Figure 20(f,g). But occlusion is not detected; signals in Figure 20(h) are comparatively smaller than the signals in Figure 20(e). Summing the two signals allows us to summarize the detected occlusion for the stimulus, and is shown in Figure 20(i).

6 Results

We have applied our kinetic occlusion detection algorithm to a battery of synthetic spatiotemporal stimuli as well as a few real stimuli. For our synthetic stimuli, we impose perfect estimated first order motions on the stimulus for controlled experimentation. For real stimuli, we use the dominant motion extraction scheme of [35].

Vertical Bars. A three-motion stimulus with non-ideal spatiotemporal junctions is shown in Figure 21(a,b). Here, two bars, moving rightward and leftward, respectively, occlude a static background. Figure 21(c,d,e) show kinetic occlusion extraction steps for static motion; kinetic occlusion is reliably detected at the edges of the bars. Figure 21(f,g,h) and Figure 21(i,j,k) show the same steps applied to rightward and leftward motion respectively. Occlusion is detected for the rightward moving bar but not for the leftward moving bar. The detected occlusion can be summed to form the signal shown in Figure 21(l).

Square-Circle I. A similar three motion stimulus is shown in Figure 22(a,b). Now, a rightward moving square and a leftward moving circle occlude a static background. Figure 22(c) shows four frames of the stimulus, and Figure 22(d) shows occlusion detected in the stimulus. In the region above and below the square, there is a motion boundary but no occlusion, so comparatively little second order orientation is extracted. At the top and bottom of the circle, a sign reversal is present. A spatiotemporal slice of the stimulus and the occlusion detected is shown in Figure 22(e,f).

Square-Circle II. The previous stimuli contained surface motions only in the horizontal direction. Figure 23(a,b,c) shows a similar stimulus, but now the background is moving upward. Now, accretion and deletion are robustly extracted at the top and bottom sides of the circle and square as well as the left and right sides, as shown in Figure 23(d). A spatiotemporal slice of the stimulus and

(a)

Order	Steering function $F_{\vec{w}}(\mathbf{x})$
N=1	$F_{\vec{w}}(\mathbf{x}) = I_x(\mathbf{x})w_1 + I_y(\mathbf{x})w_2 + I_t(\mathbf{x})w_3$ $\vec{w}_1 = \vec{w}_a, \qquad \vec{w}_2 = \vec{w}_b$
N=2	$F_{\vec{w}}(\mathbf{x}) = I_{xx}(\mathbf{x})w_1^2 + 2I_{xy}(\mathbf{x})w_1w_2 + 2I_{xt}(\mathbf{x})w_1w_3 + I_{yy}(\mathbf{x})w_2^2 + 2I_{yt}(\mathbf{x})w_2w_3 + I_{tt}(\mathbf{x})w_3^2$ $\vec{w}_1 = \vec{w}_a, \qquad \vec{w}_2 = \frac{1}{2}\vec{w}_a + \frac{\sqrt{3}}{2}\vec{w}_b, \qquad \vec{w}_3 = \frac{\sqrt{3}}{2}\vec{w}_a + \frac{1}{2}\vec{w}_b$
N=3	$F_{\vec{w}}(\mathbf{x}) = I_{xxx}(\mathbf{x})w_1^3 + 3I_{xxy}(\mathbf{x})w_1^2w_2 + 3I_{xxt}(\mathbf{x})w_1^2w_3 + 3I_{xyy}(\mathbf{x})w_1w_2^2 + 6I_{xyt}(\mathbf{x})w_1w_2w_3 + 3I_{xtt}(\mathbf{x})w_1w_3^2 + I_{yyy}(\mathbf{x})w_2^3 + 3I_{yyt}(\mathbf{x})w_2^2w_3 + 3I_{ytt}(\mathbf{x})w_2w_3^2 + I_{ttt}(\mathbf{x})w_3^3$ $\vec{w}_1 = \vec{w}_a, \qquad \vec{w}_2 = \vec{w}_b, \qquad \vec{w}_3 = \frac{\sqrt{2}}{2}\vec{w}_a + \frac{\sqrt{2}}{2}\vec{w}_b, \qquad \vec{w}_4 = \frac{\sqrt{2}}{2}\vec{w}_a - \frac{\sqrt{2}}{2}\vec{w}_b$

$$\vec{w}_a = \frac{1}{\sqrt{v_x^2+v_y^2+(v_x^2+v_y^2)^2}}[-v_x, -v_y, v_x^2+v_y^2]^T$$

$$\vec{w}_b = \frac{1}{\sqrt{v_x^2+v_y^2}}[-v_y, v_x, 0]^T$$

(b)

Filter	$n=-2$	$n=-1$	$n=0$	$n=1$	$n=2$
$g_0(n)$	.03342604	.24112500	0.450898	.24112500	.03342604
$g_1(n)$	−.09452048	−.30649700	0.000000	.30649700	.09452048
$g_2(n)$	.20619200	.06440215	−0.541189	.06440215	.20619200
$g_3(n)$	−.29033300	.70326000	0.000000	−.70326000	.29033300

Figure 17: *XYT First order motion energy extraction II.* (a) *Steering functions.* Summary of 3-d steering functions for Nth derivative of Gaussians ($N = \{1,2,3\}$). Steerable filters allow for energy extraction at any portion of the frequency domain centered at $\vec{w} = [w_1, w_2, w_3]^T$, where $|\vec{w}| = 1$, given spatiotemporal derivative approximations of a stimulus $I(\mathbf{x})$. We denote $I_{xxx}(\mathbf{x})$ as the signal obtained by 3-d separable convolution of $I(\mathbf{x})$ with a third derivative of Gaussian in the x direction, and a Gaussian in the y and t direction; $I_{xyy}(\mathbf{x})$ is obtained by convolution of $I(\mathbf{x})$ with a first derivative of a Gaussian in the x direction, a second derivative of a Gaussian in the y direction, and a Gaussian in the t direction, etc. (b) *Filter taps.* Orientation analysis is performed through multidimensional derivatives of Gaussians, where rth order derivatives are approximated by convolution of the signal with separable 5-tap kernels $g_r(n)$, presented here.

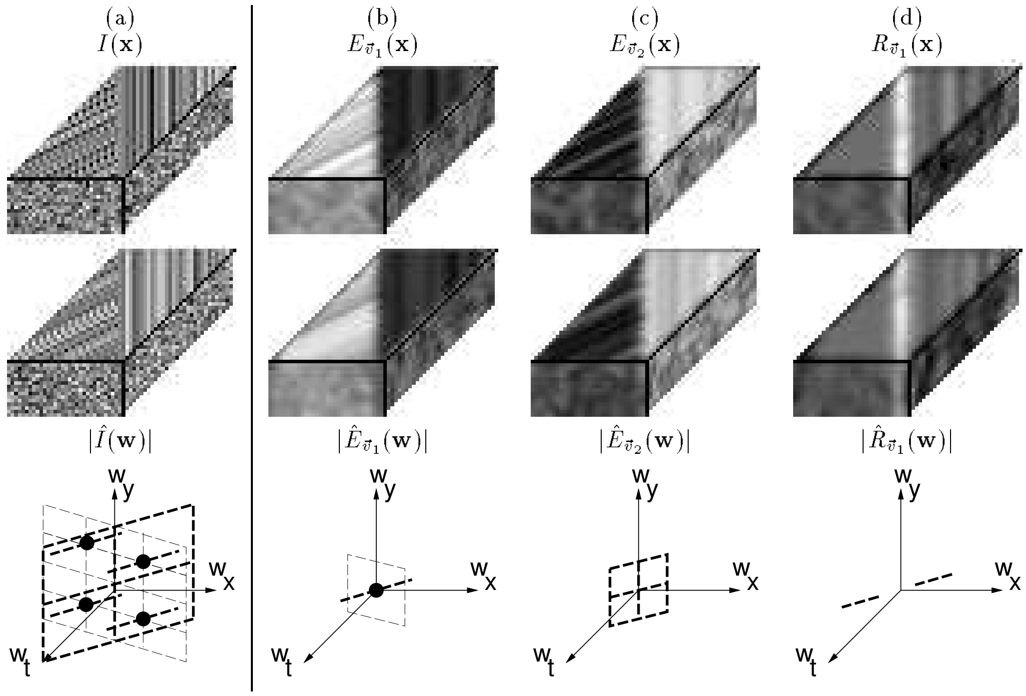

Figure 18: *XYT Second order orientation extraction.* We depict (x, y, t) stimuli with image cubes, sliced at a particular y to reveal a representative XT slice; the t direction points out of the page and to the left. (a) Stimulus $I(\mathbf{x})$, representing a rightward moving noise pattern discoccluding a leftward moving noise pattern. (b) Energy $E_{\vec{v}_1}(\mathbf{x})$ along plane $w_xv_{x1} + w_yv_{y1} + w_t = 0$ where $\vec{v}_1 = [v_{x1}, v_{y1}]^T$. (c) Energy $R_{\vec{v}_1}(\mathbf{x})$ along plane $w_xv_{x2} + w_yv_{y2} + w_t = 0$ where $\vec{v}_2 = [v_{x2}, v_{y2}]^T$. (d) Detected second order motion $R_{\vec{v}_1}(\mathbf{x})$ for the leftward moving surface; responses are positive at the motion boundary, indicating disocclusion of surface texture.

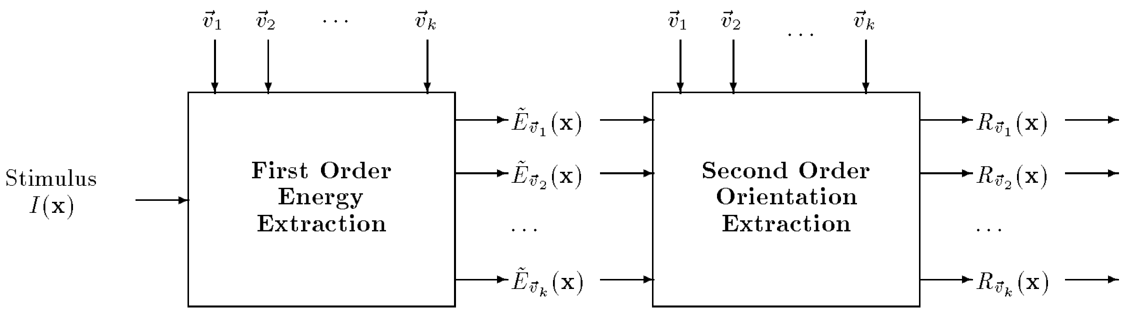

Figure 19: *XYT Kinetic occlusion detection: Block diagram.* Two dimensional kinetic occlusion stimuli are analyzed via a two stage process. The first stage extracts inhibited energy signals $\tilde{E}_{\vec{v}_n}(\mathbf{x})$ for k motions $\vec{v}_n(\mathbf{x})$, where $n = 1 \dots k$. The second stage removes first order orientation by filtering $\tilde{E}_{\vec{v}_n}(\mathbf{x})$ to result in signals $R_{\vec{v}_n}(\mathbf{x})$.

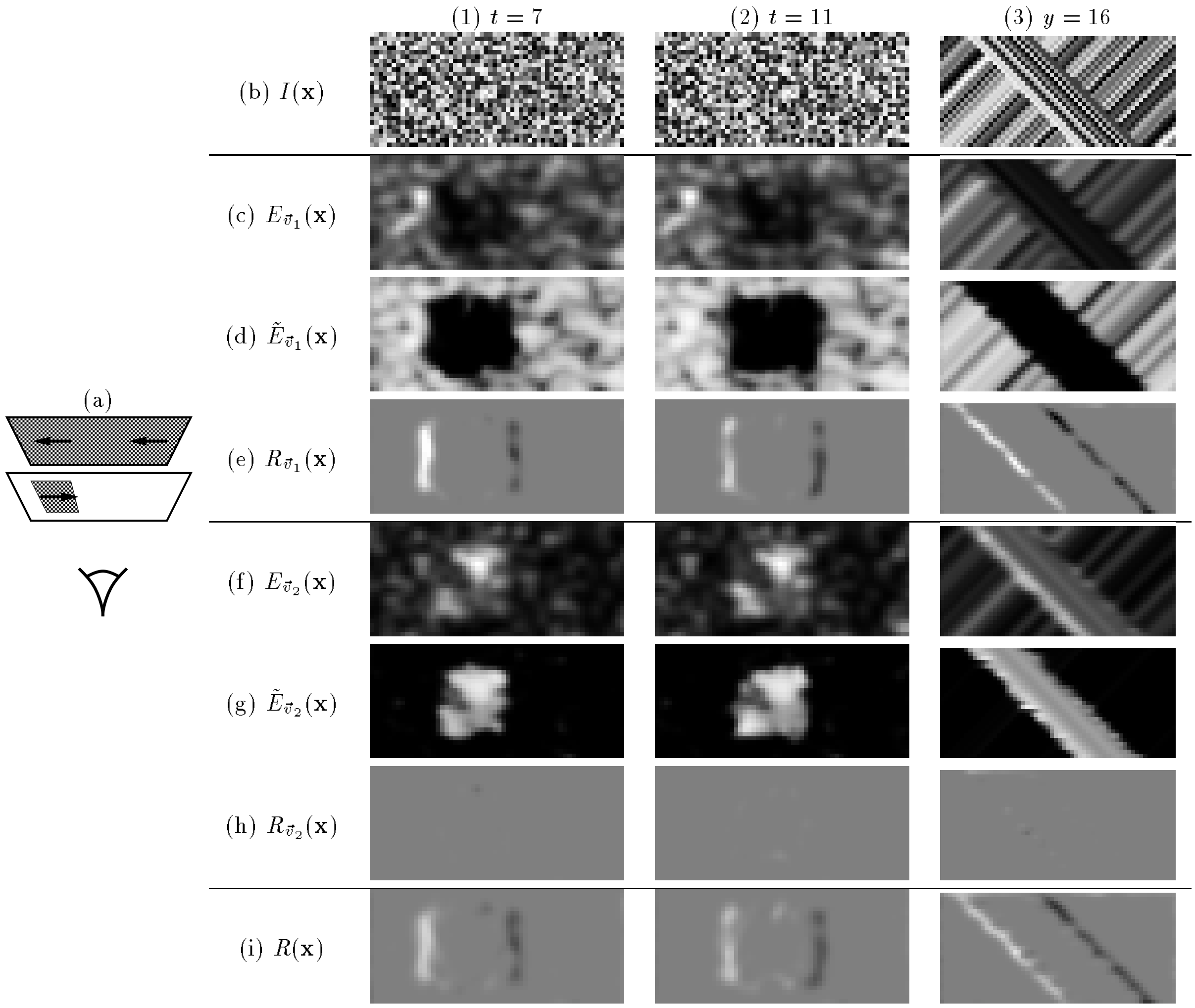

Figure 20: *XYT Kinetic occlusion detection: Example.* (a) Depiction of stimulus - a square moving rightward occluding a background moving leftward. Here, there are two dominant motions, $\vec{v}_1 = [-1, 0]^T$ and $\vec{v}_2 = [1, 0]^T$. (b) Two frames and an (x, t) slice of the stimulus. (c,d,e) First and second order orientation extracted for the occluded surface's velocity, $\vec{v}_1$. (f,g,h) First and second order orientation extracted for the occluding surface's velocity $\vec{v}_2$. (i) Response $R(\mathbf{x})$ pooled across all dominant motions can be used to summarize detected occlusion.

the occlusion detected is shown in Figure 23(e,f).

Rotations/Expansions. Rotations and expansions of moving surfaces can be handled by the affine model we adopt. Figure 24(a) depicts a rotating circle in front of a background translating upward. Three frames from the stimulus are shown in Figure 24(b). Detected occlusion is shown in Figure 24(c); disocclusion is detected at the top half of the circle, while occlusion is detected on the bottom half.

With expansions and contractions of moving stimuli, we must be cautious in applying our algorithm. Information shifts in scale as surfaces expand and contract; this may cause spurious responses. Figure 25(a) depicts an expanding circle in front of a background translating rightward. Three frames from the stimulus are shown in Figure 25(b), and detected occlusion is shown in Figure 25(c). Disocclusion is detected at the left side of the circle, while occlusion is detected on the right side of the circle. Note, however, that as the circle expands, the rightward moving edge moves similarly to the background. Thus kinetic occlusion signal detected in the third frame is significantly weaker.

Garden sequence. Figure 26 shows the same steps applied to a real image sequence, where a tree occludes a flowerbed. The motion of the tree and flowerbed is represented accurately by the affine model we have adopted. Affine motion estimates were obtained from the k-means clustering algorithm used in [35]. Second order motion is detected in regions where the tree occludes the flowerbed; as before, the sign of $R(\mathbf{x})$ differentiates accretion from deletion. At the right side of the tree, the tree disoccludes the flowerbed, so $R(\mathbf{x})$ responds positively at the motion boundary. At the left side of the tree, the tree occludes the flowerbed, so $R(\mathbf{x})$ responds negatively. A spatiotemporal (x, t) slice reveals the robustness of the response over time.

7 Discussion

We have demonstrated feasibility of detecting motion boundaries defined by kinetic occlusion, and offer a new tool which can be used in a variety of computer vision tasks. First, surfaces can be segregated on the basis of kinetic occlusion. In contrast to the large number of vision algorithms using luminance-based edge detection as a base representation, there are very few that utilize motion boundaries. Detecting edges in velocity fields [34] is difficult due to erroneous velocity estimates at motion boundaries; we offer an alternate possibility. Tools such as deformable contours [20] may also use our detected motion boundaries as an image potential.

With a detected ordinal cue to depth, we can determine the depth ordering of surfaces locally. Our detected occlusion can be used to activate direction of figure representations, as is done in the models of [6, 7, 15]. A global depth ordering of surfaces can also be attempted from these local cues (c.f. [35]).

Furthermore, by discontinuing use of optical flow upon occlusion detection, and continuing use of optical flow upon disocclusion detection, we can potentially track surfaces *through* occlusion. This is commonly attempted in feature tracking approaches in a brittle fashion; we offer an alternate approach.

Finally, the detection of motion boundaries allows for better motion estimates. In addition to activating line processes (c.f. [27]) due to velocity gradients, we can activate them through the detection of kinetic occlusion. Preventing smoothing across detected motion boundaries can yield better estimates of optical flow.

There are a variety of additional directions that can be taken based on this work. First, considerable improvements can be made on our detection methods alone. The systematic nature of "false responses" has been taken advantage of in two-dimensional spatial occlusion analysis [14, 28]; it should be possible to extend the basic idea to spatiotemporal occlusion analysis. We can also combine spatial junction analysis and our spatiotemporal junction analysis in unified framework.

Second, we are currently using non-causal operations. An asymmetry in detecting appearance and disappearance is intuitively expected, but currently they are treated identically. Considering the temporal dynamics of winner-take all mechanisms could handle this asymmetry in part. Producing a causal analysis of kinetic occlusion would require a causal model for the analysis of first order motion, which has not been done yet either.

Third, while we have explored occlusion exclusively here, a number of other naturally occurring phenomena also result in second order orientation. Multiplicative transparency, most commonly appearing as shadows, naturally fits into the model we adopt, and could be analyzed by similar or identical mechanisms.

Most importantly, our current analysis is purely local; no spatiotemporal propagation has been attempted. Solutions to spatial contour completion [6, 12, 15, 25, 7, 37] are likely to be extendable to spatiotemporal surface completion.

8 Conclusion

Motion estimation algorithms, relying on "constant intensity" and "single velocity" assumptions, have always had difficulty near motion boundaries. Providing a better model of motion boundaries is of paramount importance to two-dimensional motion analysis. Unlike conventional approaches to kinetic occlusion analysis, our approach recognizes the spatiotemporal structure present in occlusion. Robust estimation, regularization with line processes, and higher order motion models all pursue the goal of vector flow field computation; this alone cannot represent the ordinal cue to depth that kinetic occlusion provides. Our approach is complementary to these conventional approaches; by detecting kinetic occlusion, we can potentially prevent smoothing across motion boundaries. Conversely, detection of kinetic occlusion requires accurate estimates of first order motion.

Rather than ignoring the structure present in occlusions, we have provided a model for its analysis. Occlusion is a multiplicative phenomenon that results in second order orientation. The methods we developed extract this second order orientation, and yield representations which differentiate occlusion from disocclusion. We showed that it is possible to use standard concepts of signal processing to handle occlusion, which remains one of the outstanding problems in computer vision.

We have cast kinetic occlusion in a framework that can allow for the formulation of other occlusion detection algo-

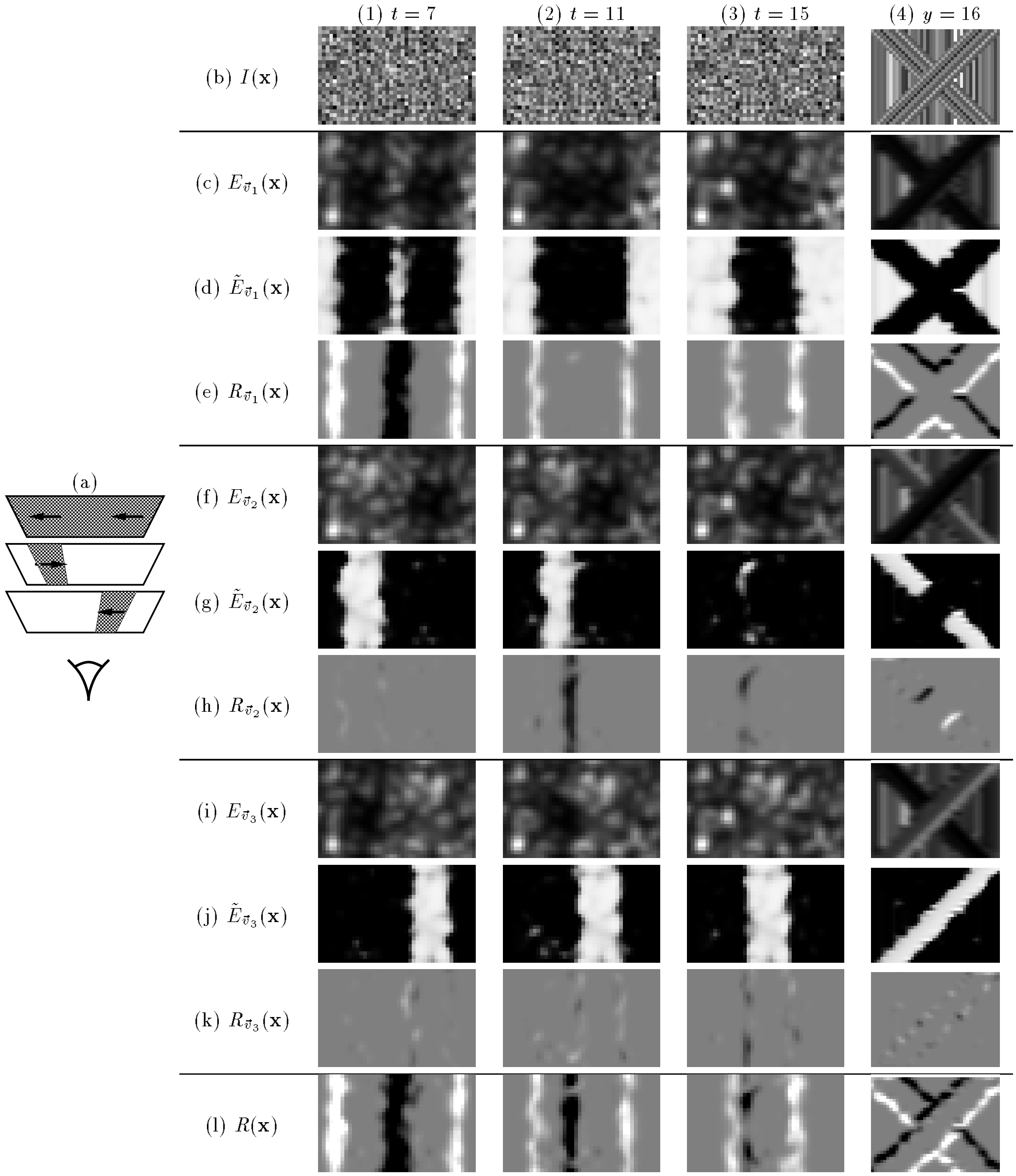

Figure 21: *Occluding bars.*

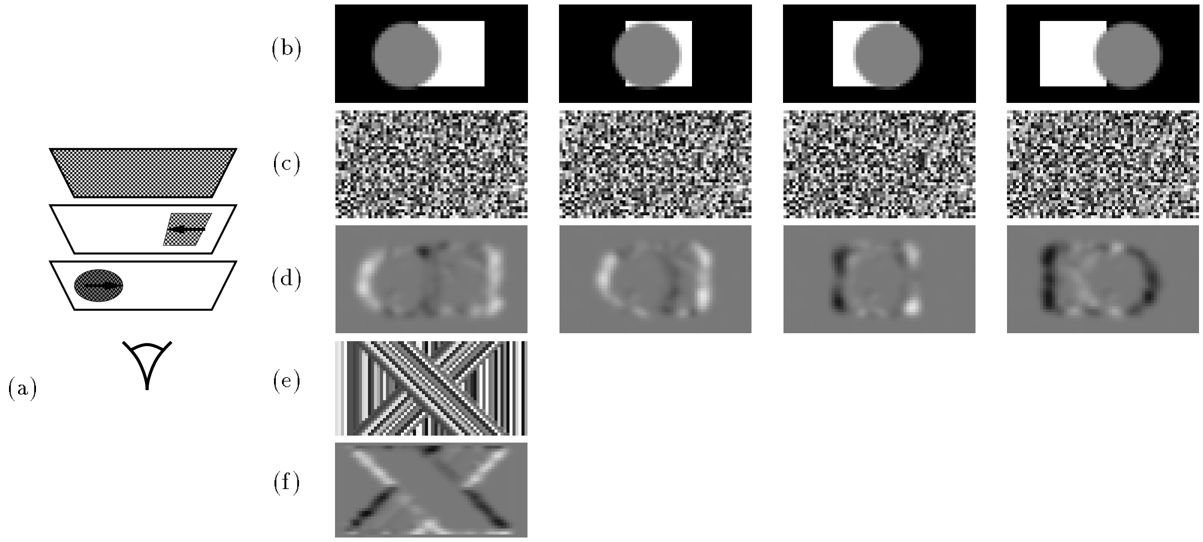

Figure 22: *Circle-Square I.* (a) Layered decomposition of stimulus - a rightward moving circle moving in front of a leftward moving square, both of which are in front of a static background. (b) Idealized segmentation of stimulus, where black, white, and grey correspond to the background, square, and circle respectively. (c) Four sample frames of the stimulus $I(\mathbf{x})$. (d) Detected kinetic occlusion $R(\mathbf{x})$ for the same four frames. (e) An (x,t) slice of the stimulus $I(\mathbf{x})$. (f) An (x,t) slice of the detected kinetic occlusion $R(\mathbf{x})$.

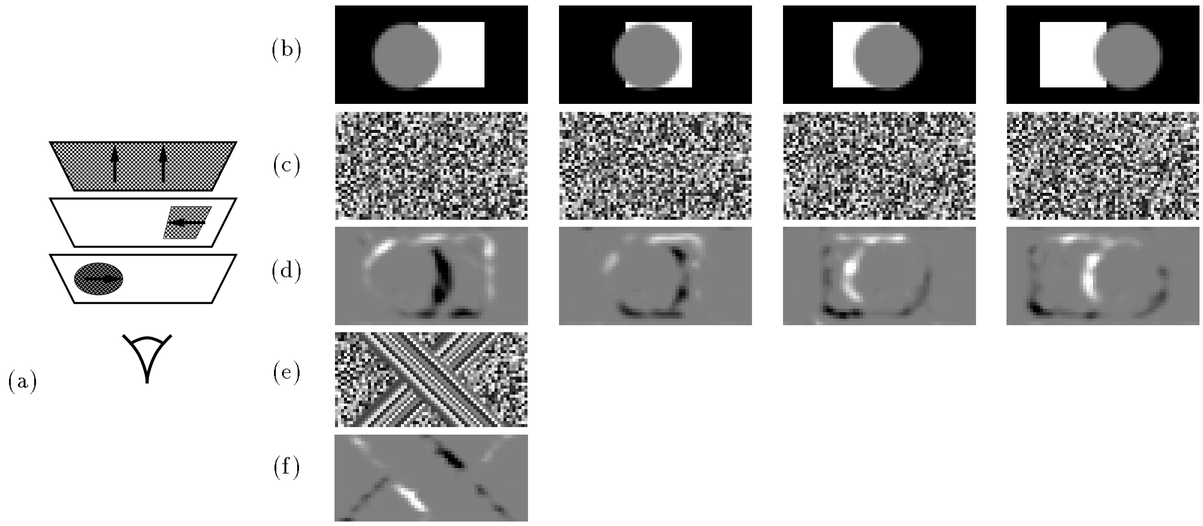

Figure 23: *Circle-Square II.* (a) Layered decomposition of stimulus - a rightward moving circle moving in front of a leftward moving square, both of which are in front of a background moving upward. (b) Idealized segmentation of stimulus, where black, white, and grey correspond to the background, square, and circle respectively. (c) Four sample frames of the stimulus $I(\mathbf{x})$. (d) Detected kinetic occlusion $R(\mathbf{x})$ for the same four frames. (e) An (x,t) slice of the stimulus $I(\mathbf{x})$. (f) An (x,t) slice of the detected kinetic occlusion $R(\mathbf{x})$.

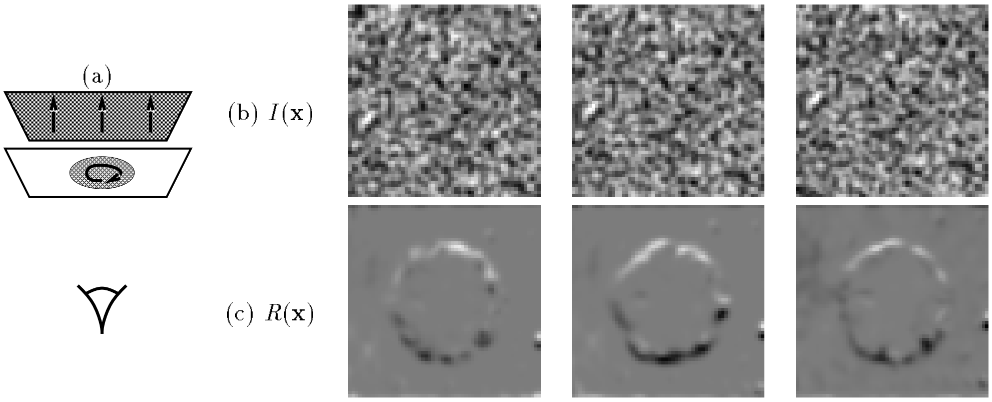

Figure 24: *Rotation/Translation.* (a,b) Shown here is a layered decomposition and three sample frames of a stimulus with a random dot circle rotating in front of a translating background moving upward. (c) Detected occlusion $R(\mathbf{x})$ for the three sample frames.

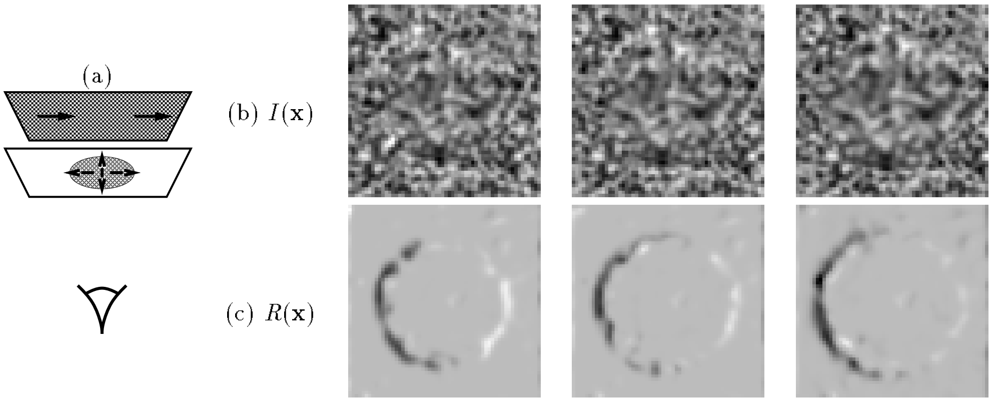

Figure 25: *Expansion/Translation.* (a,b) Stimulus - Shown here is a layered decomposition and three sample frames from a stimulus with a circle expanding in front of a translating background. (c) Detected occlusion $R(\mathbf{x})$ for the three sample frames.

rithms. We have presented a representative method here, and demonstrated success on spatiotemporal imagery containing occluding surfaces in motion.

References

[1] E. H. Adelson and J. R. Bergen. Spatiotemporal energy models for the perception of motion. *Journal of the Optical Society of America A*, 2:284–299, 1985.

[2] J. R. Bergen, P. J. Burt, K. Hanna, R. Hingorani, and S. Peleg. A three-frame algorithm for estimating two-component image motion. *IEEE Pattern Analysis and Machine Intelligence*, 14:886–896, 1992.

[3] M. J. Black and P. Anandan. Constraints for the early detection of discontinuity from motion. In *Proc. National Conf. on Artificial Intelligence, AAAI-90*, pages 1060–1066, Boston, MA, 1990.

[4] M. J. Black and P. Anandan. A framework for the robust estimation of optical flow. In *Proceedings of the International Conference on Computer Vision*, pages 231–236, Berlin, Germany, 1993.

[5] R. C. Bolles and H. H. Baker. Epipolar-plane image analysis: An approach to determining structure from motion. *International Journal of Computer Vision*, 1(1):7–56, 1987.

[6] L. H. Finkel and P. Sajda. Object discrimination based on depth-from-occlusion. *Neural Computation*, 4:901–921, 1992.

[7] L. H. Finkel and P. Sajda. Constructing visual perception artificial vision systems reveal the rules of human visual processing. *American Scientist*, 82(3):224–237, 1994.

[8] D. Fleet and A. Jepson. Measurement of image velocity by phase information. *International Journal of Computer Vision*, 1990.

[9] D. Fleet and K. Langley. Computational analysis of non-Fourier motion. Technical report, Queens University, 1994.

[10] W. T. Freeman and E. H. Adelson. The design and use of steerable filters. *IEEE Pattern Analysis and Machine Intelligence*, 13(9):891–906, 1991.

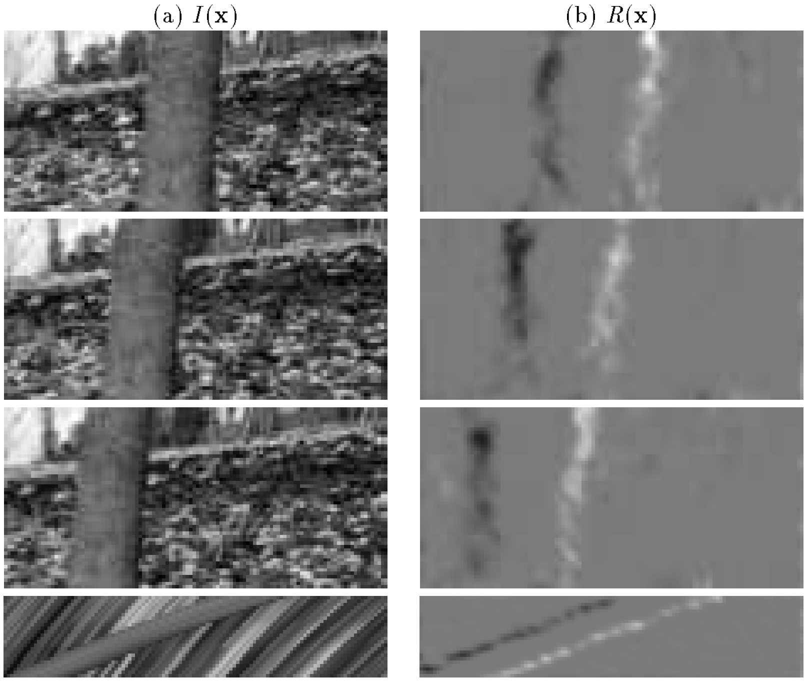

Figure 26: *Flower garden.* (a) Stimulus $I(\mathbf{x})$ - a tree moving quickly leftward occluding a flowerbed moving slowly leftward. Shown here are three sample frames from the sequence, and an (x,t)-slice. (b) Detected occlusion $R(\mathbf{x})$ extracted for the three sample frames shown, and the same (x,t)-slice.

[11] N. M. Grzywacz and A. L. Yuille. A model for the estimate of local image velocity by cells in the visual cortex. *Proceedings Royal Society London B*, 239:129–61, 1990.

[12] G. Guy and G. Medioni. Inferring global perceptual contours from local features. Technical report, University of Southern California, 1994.

[13] D. Heeger. Model for the extraction of image flow. *Journal of the Optical Society of America A*, 4:1455–1471, 1987.

[14] F. Heitger, L. Rosenthaler, R. von der Heydt, E. Peterhans, and O. Kubler. Simulation of neural contour mechanisms: From simple to end-stopped cells. *Vision Research*, 32(5):963–981, 1992.

[15] F. Heitger and R. von der Heydt. A computational model of neural contour processing: Figure-ground segregation and illusory contours. In *Proceedings of the International Conference on Computer Vision*, pages 32–40, Berlin, Germany, 1993.

[16] F. Heitz and P. Bouthemy. Multimodal motion estimation and segmentation using Markov random fields. In *Proceedings IEEE International Conference on Pattern Recognition*, pages 378–383, 1990.

[17] J. Hutchinson, C. Koch, J. Luo, and C. Mead. Computing motion using analog and binary resistive networks. *IEEE Computer*, pages 52–63, 1988.

[18] A. Jepson and M. J. Black. Mixture models for optical flow computation. In *Proceedings of Computer Vision and Pattern Recognition*, pages 760–761, New York, NY, 1993.

[19] G. A. Kaplan. Kinetic disruption of optical texture: The perception of depth at an edge. *Perception and Psychophysics*, 6(4):193–198, 1969.

[20] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1(1):1, 1987.

[21] J. Koenderink. Generic neighborhood operators. *IEEE Pattern Analysis and Machine Intelligence*, 14(6):597–605, 1992.

[22] K. Langley, D. Fleet, and T. Atherton. Multiple motions from instantaneous frequency. In *Proceedings of Computer Vision and Pattern Recognition*, pages 846–849, 1992.

[23] A. J. Movshon, E. H. Adelson, M. S. Gizzi, and W. T. Newsome. The analysis of moving visual patterns. *Experimental Brain Research*, 11:117–152, 1986.

[24] D. W. Murray and B. F. Buxton. Scene segmentation from visual motion using global optimization. *IEEE Pattern Analysis and Machine Intelligence*, 9(2):220–228, 1987.

[25] M. Nitzberg, D. Mumford, and S. Shiota. *Lecture Notes in Computer Science: Filtering, Segmentation and Depth*, volume 622. Springer-Verlag, 1993.

[26] R. Picard and M. Gorkani. Finding perceptually dominant orientations in natural textures. *Spatial Vision*, 8(2):221–253, 1993.

[27] T. Poggio, V. Torre, and C. Koch. Computational vision and regularization theory. *Nature*, 317(26):314–319, 1985.

[28] L. Rosenthaler, F. Heitger, O. Kubler, and R. von der Heydt. Detection of general edges and keypoints. In *Proceedings of the European Conference on Computer Vision*, pages 78–86, 1992.

[29] C. S. Royden, J. F. Baker, and J. Allman. Perceptions of depth elicited by occluded and shearing motions of random dots. *Perception*, 17:289–296, 1988.

[30] M. Shizawa and K. Mase. A unified computational theory for motion transparency and motion boundaries based on eigenenergy analysis. In *IEEE Computer Vision and Pattern Recognition*, 1991.

[31] D. Shulman and J. Hervé. Regularization of discontinuous flow fields. In *Proceedings of Workshop on Visual Motion*, pages 81–85, Irvine, CA, 1989. IEEE Computer Society Press.

[32] E. Simoncelli. *Distributed Representations of Motion.* PhD thesis, MIT Dept. Electrical Engineering and Computer Science, January 1993.

[33] D. Terzopoulos. Regularization of inverse problems involving discontinuities. *IEEE Pattern Analysis and Machine Intelligence*, 8:413–424, 1986.

[34] W. B. Thompson, K. M. Mutch, and V. A. Berzins. Dynamic occlusion analysis in optical flow fields. *IEEE Pattern Analysis and Machine Intelligence*, 7(4):374–383, 1985.

[35] J. Y. A. Wang and E. H. Adelson. Representing moving images with layers. *IEEE Transactions on Image Processing*, 3(5):625–638, 1994.

[36] A. B. Watson and A. J. Ahumada. Model of human visual motion sensing. *Journal of the Optical Society of America A*, 2:322–341, 1985.

[37] L. R. Williams and D. W. Jacobs. Stochastic completion fields: A neural model of illusory contour shape and salience. In *Fifth International Conference on Computer Vision*, 1995.

[38] C. Zetzsche, E. Barth, and J. Berkmann. Spatiotemporal curvature measures for flow field analysis. In *SPIE Geometric Methods in Computer Vision*, volume 1570, pages 337–350, 1991.