Computational modeling of depth ordering in occlusion through accretion or deletion of texture

Harald Ruda
Computational Vision Laboratory, Department of Communication Sciences and Disorders, Northeastern University, Boston, MA, USA

Gennady Livitz
Computational Vision Laboratory, Department of Communication Sciences and Disorders, Northeastern University, Boston, MA, USA

Guillaume Riesen
Computational Vision Laboratory, Department of Communication Sciences and Disorders, Northeastern University, Boston, MA, USA

Ennio Mingolla
Computational Vision Laboratory, Department of Communication Sciences and Disorders, Northeastern University, Boston, MA, USA

Understanding the depth ordering of surfaces in the natural world is one of the most fundamental operations of the primate visual system. Surfaces that undergo accretion or deletion (AD) of texture are always perceived to be behind an adjacent surface. An updated ForMotionOcclusion model (Barnes & Mingolla, 2013) includes two streams for computing motion signals and boundary signals. The two streams generate depth percepts such that AD signals together with boundary signals generate a farther depth on the occluded side of the boundary. The model fits the classical data (Kaplan, 1969) as well as the observation that moving surfaces tend to appear closer in depth (Royden, Baker, & Allman, 1988), for both binary and grayscale stimuli. The recent "Moonwalk illusion" described by Kromrey, Bart, and Hegdé (2011) upends the classical view that the surface undergoing AD always becomes the background. Here the surface that undergoes AD appears to be in front of the surrounding surface—a result of the random flickering noise in the surround. As an additional challenge, we developed an AD display with dynamic depth ordering. A new texture version of the Michotte rabbit hole phenomenon (Michotte, Thinès, & Crabbé, 1964/1991) generates depth that changes in part of the display area. Because the ForMotionOcclusion model separates the computation of boundaries from the computation of AD signals, it is able to explain the counterintuitive Moonwalk stimulus. We show simulations that explain the workings of the model and how the model explains the Moonwalk and textured Michotte phenomena.

Introduction

Detection of a boundary defined only by the accretion or deletion of texture elements is effortless for human observers, but computationally difficult. A variety of motions can be present on either side of the boundary, and the boundary may or may not be well defined, but when accretion or deletion is involved the boundary is usually perceived as quite sharp. Merely detecting the boundary is not all that is done, because along with the boundary, the correct depth order relationship is also determined. The surface with accretion or deletion is farther back. This is the depth order from motion (DOFM) assignment, which is one of the first steps in the determination of figure and ground in time-varying displays. This depth order relationship can be used to assign border ownership to one side or the other. The DOFM is not the only way to make this assignment, but when accretion or deletion of texture occurs, it is one of the strongest perceptual cues (Ono, Rogers, Ohmi, & Ono, 1988; Royden, Baker, & Allman, 1988).
The classical accretion and deletion phenomena (shown in Figure 1) are based on ecologically valid conditions, i.e., occurring naturally outside of the lab. Accretion or deletion occurs along static (top row) or moving (bottom row) edges, but in all cases it is associated with the surface that is farther away from the observer, i.e., in a depth plane farther from the observer. While the stimuli shown in the figure are indeed impoverished compared to conditions outside the lab—lacking elements such as contrast, color, and binocular depth cues—we suggest that accretion and deletion should be studied in isolation, as well as in combination with other cues, because they are such reliable monocular cues to depth order.

While easy for human observers, detecting this moving boundary and assigning the proper depth relationships is computationally challenging. A number of models of accretion and deletion have been proposed (Black & Anandan, 1990; Chou, 1995; Niyogi, 1995; Fleet, Black, & Jepson, 2000; Tsotsos et al., 2005; Feldman & Weinshall, 2008; Sundberg, Brox, Maire, Arbeláez, & Malik, 2011; Tschechne & Neumann, 2012), with varying attempts at neurological plausibility, that solve this problem in a number of ways. Barnes and Mingolla (2013) developed a neural model based on the Barlow–Levick (1965) motion detection circuit that uses two interacting streams of processing, a MOTION stream and a FORM stream, to solve the DOFM problem. The model detects motion, motion-defined boundaries, and static and moving boundaries and assigns behaviorally correct depth relationships to the surfaces creating the boundary; from this it computes the proper depth ordering.

Some recently formulated stimuli are pushing the competencies of these models. This article modifies the model of Barnes and Mingolla (2013) that successfully explains classical accretion and deletion phenomena (Kaplan, 1969; Royden et al., 1988) in order to explain the paradoxical Moonwalk illusion and the new texture version of the Michotte rabbit hole illusion.

The Moonwalk illusion

Recently a new stimulus was introduced that appears to contradict the standard dogma of accretion and deletion. Kromrey, Bart, and Hegdé (2011) created a type of stimulus, dubbed the Moonwalk illusion, in which even though there is accretion and deletion of the moving central portion of a random dot display, the central portion appears in front of rather than behind the surrounding area. As described in Figure 2, this is achieved by scintillating the dots in the surrounding area, with sequential frames containing uncorrelated
noise in this area. (See Supplementary Movie S7 for a short demonstration of this effect.) Thus when the surrounding dots are classically static, the central portion appears behind; but when the surrounding dots are scintillating, the central portion appears in front. The actual dots of the moving central portion are identical in both cases. Because the central portion appears in front even though it undergoes accretion and deletion, this is a paradoxical stimulus. It is important to keep in mind, however, that even though it is paradoxical, it is not ecological, in the sense that it is difficult to imagine how such a display could be observed with material surfaces and environmental illumination. In any case, a model of human motion perception that assigns DOFM should explain human responses to this stimulus.

The textured Michotte rabbit hole phenomenon

Furthermore, we have a created a new stimulus that has a dynamic depth order structure. This stimulus is a texture version of the Michotte rabbit hole effect (Michotte, Thinès, & Crabbé, 1964/1991). The standard Michotte rabbit hole effect is illustrated in Figure 3. As the black circle moves back and forth, it is occluded on the right side by a rectangle the same color as the background, and it appears to slide in and out of a slit formed out of the background. The slit is always only as wide as it needs to be—just slightly longer than the chord of the circle where it is occluded—and when the circle is completely unoccluded, the slit disappears entirely.

Our texture version of the Michotte rabbit hole effect replaces the white background with a static random dot texture, and the black circle with a statistically identical random dot texture that is moving with the circle. This generates a strong depth percept where the moving circle is very clearly in front of the background. The depth in the texture version actually appears to be stronger than in the standard version, likely because of the accretion and deletion of the background caused by the moving circle; whereas in the standard version, the uniform white background can (with some difficulty) be seen as moving together with the black circle. In both the standard and the texture version, the moving circle appears to slide into the slit that appears just in time and also expands and shrinks to be only slightly wider than the chord of the disappearing or appearing circle. The area on the right side of the slit is part of the background at the farthest depth when the circle is unoccluded, but becomes the region that is closest to the observer when the circle is occluded. It is worthwhile to note that the area to the right of the slit can be seen “in front” at the same time as the area to the left of the circle is seen “in back” without the need for a visible boundary to create a depth discontinuity.

Unlike the Moonwalk illusion, the standard and textured versions of the Michotte rabbit hole effect are both ecologically valid. The rabbit hole effect is merely a generalization of the simple accretion and deletion stimuli shown in Figure 1. The occlusion of one object by another is so commonplace as to be completely unremarkable. It is only when the occluding object has exactly the same color as the background (as in the standard rabbit hole effect) that we speak of a perceptual “illusion.” Something similar occurs in the case of the textured version, where the textures of the background, the occluding object, and the occluded object are statistically identical. In most normal circumstances, the textures would be different and there would also be other differences in luminance and color, as well as relative motion between the background and...
the occluding object. The dynamic nature of the textured Michotte rabbit hole effect is a special challenge for models that assign DOFM, requiring the assignment of depth order to change over time consistent with the percept. It also presents a unique difficulty in that different regions of the same fronto-parallel surface must sometimes be simultaneously behind the circle and in front of it.

The revised model continues to exhibit all the functionality of the Barnes and Mingolla (2013) model while:

- working with grayscale textures as well as binary random dot displays
- explaining the Moonwalk illusion in a satisfactory manner
- explaining the complex depth ordering generated by the textured Michotte rabbit hole effect
- having a simpler structure than the original model

The reason for these changes is more easily appreciated after a complete model description, which follows. The reader can refer to the later subsection “Differences from Barnes and Mingolla” for a summary of the changes to the model. We next describe the intuitions underlying the model’s computations.

---

**The FormMotionOcclusion (FMO) model**

The FormMotionOcclusion (FMO) model discussed in this article is the successor to the ForMOcclusion model of Barnes and Mingolla (2013). The FMO model is constructed using the Barlow–Levick principle of motion detection (Barlow & Levick, 1965) in two separate computational steps. The Barlow–Levick model detects motion by using luminance changes to generate elementary motion signals. At the same time, the model also sends inhibitory signals that block the “wrong” or “null” direction. Thus, only motion that is consistent with a sequence of luminance changes is propagated uninhibited and detected.

The FMO model similarly detects accretion and deletion by propagating changes in elementary motion. It can thus be viewed as a second-order Barlow–Levick model, where the starting and stopping of motion is detected instead of elementary motion. The key mechanism again is the inhibition of signals that are “incorrect.” In the case of accretion signals, the existence of previous motion inhibits the propagation. And in the case of deletion signals, any continuation of motion in the same direction again inhibits the propagation of the deletion signal.

The MOTION pathway that detects elementary motion and accretion-deletion events is connected to the FORM pathway that is designed to detect and complete boundaries. Thus a moving boundary is represented in the model as a series of boundaries at successively different locations at successive moments in time. However, the boundaries are connected to the motion signals in the perceived motion representation of the model. Percepts are represented explicitly in the model; there are representations for luminance, motion direction, and depth for each location in visual space. Notice that there is currently no allowance for motion transparency (instead see Raudies, Mingolla, & Neumann, 2011).

A block-level representation of the model is shown in Figure 4. The MOTION stream is shown in green, and
the static-boundary mechanism is shown in blue. The beige boxes represent the actual percepts of the primate visual system related to the streams at this point. The diagram also shows the flow of information between the modules, as well as the various connections between the modules that gate or otherwise influence the computations.

The input stimulus to the model consists of a sequence of textured luminance images. The FORM stream immediately looks for orientation signals, and the MOTION stream looks for elementary motion signals. The next stage in both streams is spatial center–surround competition, which cleans up the boundary and motion signals detected in the form and MOTION streams, respectively. The FORM stream builds on the boundary fragments using long-range bipole cells to complete the boundaries. The MOTION stream computes accretion and deletion signals, which are used by the boundary completion in the FORM stream. Finally, the signals from the FORM and the MOTION streams are combined to form the percepts (corresponding to what is actually seen) of motion, and depth.

Figure 5 shows an idealized representation of the expected responses of the different stimuli shown in Figure 1. The layout of the model is the same as in Figure 4, with the same color coding of the layer types applied to the borders of the rounded rectangles. The inside of the rectangles shows the expected results of the processing in the layers. Arrows denote motion in specific directions, with “⃗” representing no motion. Dashed lines represent edge signals, with continuous lines representing fully formed boundaries. Only two depth layers are shown for simplicity, the closer one on top of the farther one, which is enough to clarify the model’s processing stages.

The key is that there are four different combinations of signals: accretion and deletion for either static or moving dots. These four conditions will generate different types of depth-ordering conditions, generally pulling one surface nearer to the observer and pushing another farther away from the observer.

A fifth basic configuration—the idealized “shear” stimulus, where there is no accretion or deletion but one side of the boundary contains motion that is parallel to the boundary—is shown on the left side of Figure 6. Below the boundary, all the dots are moving to the right, and above the boundary, the dots are stationary. Also shown on the right side in Figure 6 are the model’s idealized responses to this shear stimulus. Notice that the model does not produce a clear form boundary between the moving and stationary areas. Human observers tend to see the boundary in the shear condition to be just as sharp as the boundary in the various accretion and deletion conditions. The model is expected to generate a slightly less sharp boundary, as there will be no explicit boundary represented in the FORM pathway. This could be fixed in the model by including a module with explicit shear detectors, as there is physiological evidence for such detectors in area MT (Lagae, Maes, Raiguel, Xiao, & Orban, 1994). Introducing a new module was not deemed necessary in this case because the interesting part of the shear stimulus is actually the depth ordering, not the sharpness of the perceived boundary. The model generates a depth ordering where the
moving sheet is seen as nearer to the observer than the stationary sheet. This bias towards seeing moving surfaces, no matter the direction, as nearer the observer is built into the model explicitly and is based on observations by Royden et al. (1988).

**Differences from previous model**

A number of changes have been made to the previous model (Barnes & Mingolla, 2013) in order to achieve the new competencies. The changes are detailed in the following subsections.

**Simplifications**

The previous model used a number of model steps to “clean up” motion signals. Specifically, there were steps to deal with the generation of the ring of outward motion signals from the sudden appearance or disappearance of a dot. While these steps provided better results in certain unusual circumstances, they also interfered with the ability of the model to process grayscale patterns. They were thus removed, meaning that some signals such as the motion edge signal might be less well defined in noisy conditions, such as where dots appear and disappear.

Figure 5. Examples of expected responses for different layers of the model to four different stimuli. The layout and color coding are the same as in Figure 4, except that here only the frames of the boxes are colored. Arrows denote motion in a specific direction, and “∅” represents no motion. Dashed lines represent edge signals, with solid lines representing fully formed boundaries. Only two depth layers are shown, the closer depth on top of the farther layer. The top row shows the static boundary condition, and the bottom row shows the moving boundary condition. The left-column stimuli contain deletion of texture, and the right-column stimuli contain accretion of texture elements.
Local processing

The previous model was formulated in such a way as to represent a patch of the visual field. In the new formulation, a much larger area—perhaps the whole visual field—is described, but processing is always localized to smaller areas. As result, instead of mechanisms that appear global with conditions such as “everywhere where the motion is in direction \( d \),” we get “all locations in this circle centered on \( X \) in direction \( d \).”

Faster diffusion and direct competition between depth pools

The actual filling-in process has been modified in two ways. The diffusion process is now assumed to spread faster, which means that the filling-in part of the simulation must execute at a finer timescale than the rest of the model. It also means that the filling-in process can better keep up with moving edges. In addition, the three layers that represent depth pools (near, middle, far) now compete directly between each other instead of just the inputs to the pools competing with each other. This new arrangement is more biologically plausible. The precise implementation of this filling-in process is actually not important for the model. The equations and simulations in this article use a diffusion filling-in process; this is a computational simplification of what may be biologically expressed by rapid communication through myelinated fibers connecting cortical areas rather than by diffusion within laminae (Layton, Mingolla, & Yazdanbakhsh, 2012).

Push–pull signals

The most significant departure from the previous version of the model is the manner in which depth signals are generated. Previously, if accretion or deletion was detected for a given direction of motion, then all motion in that direction was strongly biased towards the far plane. While this works just fine if one is considering only local effects, as in the four classical stimuli, for more complex stimuli such as the textured Michotte rabbit hole effect it will not work. As shown in Figure 7, accretion and deletion signals generate corresponding push–pull signals on either side of where the boundary ought to be if there is one. Thus the push–pull signals are generated even if it turns out that there is no boundary. The boundary is generated by the FORM system, and at the point where accretion and deletion is detected it is not yet possible to know whether a boundary exists, because the very accretion and deletion signals may be needed to generate the boundary.

Model specification

We now describe the model using mathematical expressions. A new notation has been developed that is suitable to modeling using a certain class of ordinary differential equations, sometimes referred to as “shunting equations,” applied to layers of computational nodes (analog neurons or neural populations). After the notation is explained, it makes the description
of the model more straightforward than is the case for previous notations.

**The general form of the equations**

The equations that describe each layer of the model are based on the standard shunting equations of the form

$$\frac{dw}{dt} = -A w + (B - w) X - (C + w) K I,$$

where $A$, $B$, $C$, and $K$ are constants and $X$ and $I$ are, respectively, the excitatory and inhibitory inputs. The values of $X$ and $I$ are generally unique to each equation. Equation 1 can also be solved at equilibrium, yielding

$$w = \frac{B X - C K I}{A + X + K I},$$

with the same meanings for the symbols. In most of the equations of the model, the constants $A$, $B$, $C$, and $K$ are the same, which means that only the name of the layer $w$ and the excitatory and inhibitory inputs are actually interesting and need to be specified. The layer (in this case $w$) is a spatially distributed collection of nodes whose values change over time. Thus all nodes can be indexed by spatial position $ij$ as well as time $t$; as these are always present, they will not be used unless necessary, and thus $w(k, \bar{d})$ always implies $w(t, ij; k, \bar{d})$. Therefore we will rewrite the description of the computation as

$$w(k, \bar{d}) \leftarrow \text{shunt}(X = \text{excitation}, I = \text{inhibition}).$$

This both simplifies the description and allows straightforward implementation using either differential equations or equilibrium computations. Here $k$ refers to orientations (of edges) and $\bar{d}$ to directions (of motion). Also, when superscripted on the layer, “$+$” and “$-$” refer to the on and off systems (“$\pm$” refers to both together). Depending on the specific layer of the model, a subset of these dimensions will apply. The orientation $k$ and direction $\bar{d}$ dimensions are not seen together in the model described here.

In addition, it is important to remember that the output is always transformed before it is transmitted to the next layer. The most common transformation is the simple half-wave rectification described by

$$\text{output of } w \text{ for input to other layers} = [w]^+ = \max(0, w).$$

Furthermore, variables $w$ represent fields of nodes but should be thought of as multidimensional arrays, and are therefore denoted by a bold symbol $w$. Subscripts denoting a specific position within the array $ij$ are not needed (because all the nodes within a layer...
perform the same computation) and therefore are not used. If the spatial position or orientation is required to clarify a specific computation, then the only the necessary variables are specified; for example,

\[ w(t - 1, i + 1) \]

refers to the value of \( w \) at the previous time unit in the past, but also one spatial unit to the right. Subscripts, when used, mostly denote identification of a constant or kernel with a specific layer. In this way, the description remains valid whether simulations are implemented using a rectilinear, hexagonal, or spatially variant sampling regime.

By rewriting the equations in this particular manner, we anticipate the development of simulation systems where the computations are specified in this way and the choice of ordinary differential equation or equilibrium computation (see equations) is made at a later point. This choice can be made manually or perhaps even automatically decided. In some applications, such as real-time robotic applications, it may be necessary to use the faster equilibrium computations whenever possible.

**Convolutions and common kernels**

The most common operation on a layer of neurons in the model is the convolution, which will here be denoted by an asterisk. It involves a kernel \( \mathbf{M} \) of finite size that is applied to the layer \( \mathbf{w} \) at every location, yielding an output value at each location applied. The simplified notation is here expanded into the more conventional notation

\[
\mathbf{w} \ast \mathbf{M} = (\mathbf{w} \ast \mathbf{M})(i, j, k) = \sum_{i,j,k \in \mathbf{M}} w(i - \bar{r}, j - \bar{j}, k - \bar{k}) \mathbf{M}(\bar{i}, \bar{j}, \bar{k}).
\]

Most kernels are isotropic, the same in all directions, and are denoted by \( \mathbf{M} \); common forms of these kernels are multidimensional Gaussian kernels, such as this two-dimensional Gaussian kernel:

\[
\mathbf{M}(\sigma) = e^{-((i^2 + j^2)/(2 \sigma^2))}.
\]

Other kernels are anisotropic, such as edge detectors, and are denoted by \( \mathbf{N} \). Gabor-type kernels may also be used, and are denoted by \( \mathbf{G} \).

**Equation variations**

Not all equations are simple shunting equations. There are other similar types of equations that can be written using the same shorthand: nonhyperpolarizing shunting, diffusion shunting, and direct equations.

**Nonhyperpolarizing**

When a version of the shunting equation that does not hyperpolarize is needed, it is denoted by “shunt\(^+\),” which just means that \( C = 0 \) and \( K = 1 \) for this layer. For example,

\[
y(d) \leq \text{shunt}^+(X = w \ast \mathbf{M}(\sigma_i), I = w \ast \mathbf{M}(\sigma_i)).
\]

**Diffusion**

Another specific variation is the case of gated diffusion equations. These are also nonhyperpolarizing shunting equations, but they have the specific form

\[
\frac{d\mathbf{w}}{dt} = -A\mathbf{w} + (B - \mathbf{w}) \{X + [F(\mathbf{w}, Z)]^+\} - w \{I + [-F(\mathbf{w}, Z)]^+\},
\]

 coupled with the gating signal

\[
F(\mathbf{w}, Z) = F(i, j; \mathbf{w}, Z) = \sum_{(i,j) \in H} \Phi \left( \frac{\delta}{1 + \epsilon[Z]} \right) (w(i,j) - w(i,j))
\]

where \( H \) is the neighborhood of location \( ij \) (implicit in the definition of \( F \)), which in a rectangular simulation would correspond to \( H = \{(i + 1, j), (i - 1, j), (i, j + 1), (i, j - 1)\}; (\delta \) and \( \epsilon \) are constants). These equations are rewritten as

\[
w(i, j) \leq \text{diffusion}(X = \text{excitation}, I = \text{inhibition}, Z = \text{gating signal}).
\]

If the diffusion equation cannot be solved at equilibrium, iterative algorithms may be used instead to approximately solve these types of equations.

**Direct**

Sometimes the computation is simple and best stated directly, as in

\[
\ell^+(t) \leq L(t) - L(t - 1).
\]

This includes the case where the computation is simply a sum of previous layers:

\[
w(k) \leq i^+ \mathbf{G}(k, \sigma_i, \text{phase} = \pi/2) + i^+ \mathbf{G}(k, \sigma_i, \text{phase} = 0).
\]

Here \( \mathbf{w} \) is the sum of two Gabor components convolved with an input layer \( i \); \( \mathbf{G} \) is a Gabor kernel of
orientation $k$ with spread $\sigma_k$; and both the even (phase $= \pi/2$) and odd (phase $= 0$) parts are used in the computation.

FMO model equations

Parameters, variables, and constants have the same names as in the previous version of the model (Barnes & Mingolla, 2013). The rest of the notation is reworked and simpler.

Transient (MOTION) system

The input image is a movie with luminance $L$, varying as a function of space and time—i.e., $L(t, i, j)$. Luminance increases (on channel) are defined by

$$\ell^+(t) \triangleq L(t) - L(t-1),$$

and luminance decreases (off channel) are defined by

$$\ell^-(t) \triangleq L(t-1) - L(t).$$

Remember that all outputs are half-wave rectified ($[x]^+$) unless otherwise specified. Inhibitory interneurons, implementing Barlow–Levick inhibition, are governed by

$$a^\pm(\vec{d}) \triangleq \text{shunt}(X = \ell^\pm, I = a^\pm((i, j) \pm \vec{d}); \vec{d}/2),$$

where $(i, j) \pm \vec{d}$ refers to the node next to $ij$ in the direction of $\vec{d}$ and $\vec{d}$ is the direction of motion. Note that although the predecessor model restricted the motion signals to eight directions of motion (Barnes & Mingolla, 2013, equation 11 and after), we have chosen to allow any reasonable fixed number of directions (say from four through 16, depending on the geometry and connectivity of nodes). The direction opposite of $\vec{d}$ is denoted by $-\vec{d}$. This means that the direction-specific inhibitory interneurons inhibit each other if they are at neighboring locations representing motion towards each other. Directional transients (which always computed at equilibrium, because they decay too quickly) are described by

$$b^\pm(\vec{d}) \triangleq \text{shunt}(X = \ell^\pm, I = a^\pm((i, j) \pm \vec{d}); -\vec{d}),$$

which is almost identical to Equation 16. The equations for $a()$ and $b()$ are very similar, but remember that $a()$ represents known interneurons that inhibit each other, as well as $b()$ neurons, whereas $b()$ represents excitatory neurons that represent motion in a specific direction. Figure 8 explains the geometrical arrangement of these motion neurons. The signals are combined by adding the rectified on and off components to yield elementary motion signals:

$$c(\vec{d}) \triangleq b^+(\vec{d}) + b^-(\vec{d}).$$

Short-range filtering is performed according to:

$$d(\vec{d}) \triangleq \text{shunt}^+(X = c(\vec{d})*M(\sigma_1),$$

$$I = F_1 \sum_{\vec{D} \neq \vec{d}} c(\vec{D})*M(\sigma_1) + F_2 c(\vec{d})*N(\sigma_2, \sigma_3),$$

where $\sigma_1$, $\sigma_2$, and $\sigma_3$ are the sizes of Gaussian kernels and $F_1$ and $F_2$ are constants. The kernel defined by $\sigma_1$ is the small isotropic on-center for the same direction and also off-center for other directions; whereas $\sigma_2$ and $\sigma_3$ are the anisotropic off-surround for the same direction (as shown in Figure 8C). This signal is then smoothed using a direction-specific uniform-density circular kernel with radius $\sigma_c$ according to

$$e(\vec{d}) \triangleq \left[ U(\sigma_c)*d(\vec{d}) - \theta_e \right]^+.$$

The filtered motion signals are derived by choosing a maximum activity at each location using

$$f(\vec{d}) \triangleq \begin{cases} e(\vec{d}) & e(\vec{d}) \geq e(\vec{d}) \forall \vec{d}, \\ 0 & \text{otherwise} \end{cases},$$

and thus only one direction at each spatial location is active. In the previous version of the model (Barnes & Mingolla, 2013), the concept of stationary motion needed to be introduced—i.e., in addition to the different directions of motion a no-motion “direction” was used. This concept has been eliminated entirely from the version of the model presented here. Nothing is lost with this simplification; indeed, the equations are both simpler and more general as a result.

Next, the filtered and maximized motion signals are used to compute the various accretion and deletion signals. The key notion is to go back in time slightly (2 units) and see if the motion observed then corresponds to the motion observed now. As shown in Figure 9, if there is a discontinuity such as a static edge, then there will be an unexpected change ($g_A$ and $g_B$). If the edge is moving, then the change will occur at the same location ($g_C$ and $g_D$). There are four different conditions, corresponding to deletion at a static edge (A),

$$g_A(\vec{d}) \triangleq \text{shunt}(X = f(t-2, i, j - \vec{d}; \vec{d}),$$

$$I = N(\sigma_4, \sigma_5)*f(t, i, j + \vec{d}; \vec{d}),$$

accretion at a static edge (B),

$$g_B(\vec{d}) \triangleq \text{shunt}(X = f(t, i, j + \vec{d}; \vec{d}),$$

$$I = N(\sigma_4, \sigma_5)*f(t-2, i, j - \vec{d}; \vec{d}),$$

deletion by a moving edge (C),

$$g_C(\vec{d}) \triangleq \text{shunt}(X = f(t - \vec{d}; \vec{d}),$$

$$I = N(\sigma_4, \sigma_5)*f(t, i, j + \vec{d}; \vec{d}),$$

and addition by a moving edge (D),

$$g_D(\vec{d}) \triangleq \text{shunt}(X = f(t, i, j - \vec{d}; \vec{d}),$$

$$I = N(\sigma_4, \sigma_5)*f(t-2, i, j + \vec{d}; \vec{d}).$$
Figure 8. The generation and filtering of motion signals. (A) A one-dimensional version of the specified Barlow–Levick-type circuit, where luminance changes that generate motion are either inhibited or permitted. The inhibitory (red) nodes are slower to respond, providing the time delay of inhibition that is characteristic of the Barlow–Levick model. (B) While the model allows any number of motion directions, the simulations shown use these eight directions. The circuit shown in (A) is present all around each node. (C) The generated motion signals are noisy and need to be filtered using a shunting equation with a small excitatory kernel and an inhibitory kernel with both an anisotropic spatial extent and inhibition from other directions.

\[
g_C(\vec{d}) \leftarrow \text{shunt}(X = f(t, ij - \vec{d}; \vec{d})),
I = N(\sigma_4, \sigma_5) \ast f(t - 2, ij - \vec{d}; \vec{d})),
\]

and accretion by a moving edge (D),

\[
g_D(\vec{d}) \leftarrow \text{shunt}(X = f(t - 2, ij - \vec{d}; \vec{d})),
I = N(\sigma_4, \sigma_5) \ast f(t, ij - \vec{d}; \vec{d})),
\]

In all cases, the inhibitory part of the shunting equations is a convolution with a “short but wide” kernel with length \(\sigma_4\) and width \(\sigma_5\) in order to normalize sensitivity based on lateral neighboring locations. These accretion and deletion signals pair up to generate the push–pull signals. First, the two signals that produce a push ahead of the signal are combined into one,

\[
h_a(\vec{d}) \leftarrow g_B(\vec{d}) + g_C(\vec{d}),
\]

then the two signals that produce a push behind the signal are combined,

\[
h_b(\vec{d}) \leftarrow g_A(\vec{d}) + g_D(\vec{d}).
\]

These signals are used to produce the separate push–pull signals; the push signal is defined by

\[
j_{\text{push}}() \leftarrow \sum_d \left(h_a(ij + \vec{d}; \vec{d}) + h_b(ij - \vec{d}; \vec{d})\right),
\]

and the pull signal by

\[
j_{\text{pull}}() \leftarrow \sum_d \left(h_a(ij - \vec{d}; \vec{d}) + h_b(ij + \vec{d}; \vec{d})\right).
\]

The motion edge signals are generated by combining all four accretion and deletion signals,

\[
m() \leftarrow \sum_d \left(g_A(\vec{d}) + g_B(\vec{d}) + g_C(\vec{d}) + g_D(\vec{d})\right),
\]

and then passed to the boundary completion of the FORM stream.

**Static (FORM) system**

The static FORM system is used to generate boundaries that exist in the input. The structure is identical to the FORM system presented by Barnes and Mingolla (2013), and thus to avoid replication we refer the reader to the equations presented there. The input is first processed by a center–surround structure much like lateral geniculate nucleus p-cells (LGNe). From this simple cell, responses are computed, and then complex cells are formed by the combination of even and odd symmetric simple cells. The complex cells are thus sensitive to orientation but not to contrast polarity or phase. This is followed by spatial competition, which adds in any suggestions of a motion edge. Specifically, the \(k()\) and \(l()\) motion onset and offset signals of Barnes and Mingolla’s (2013) equation 40 are replaced by the motion edge signal \(m()\) defined in our Equation 30. Bipole cells are used to connect edge fragments into longer-range boundaries, which are then used to compute both the motion and depth perceptual systems.
Perception system

While the MOTION and FORM systems are used to recognize those parts of the input that are defined in those terms, the model representation of what is actually seen is represented in the perception system. This set of model layers contains representations of the motion and depth at each location in visual space. The depth representation is actually coded by activations in the three separate depth layers (near, middle, and far).

The perceptual representations all use the diffusion form of the shunting equation. This form is used when there is a need to fill in the representation and the boundary is clearly defined. Recall that the diffusion shunting equation is defined by excitatory (\(X\)), inhibitory (\(I\)), and gating (\(Z\)) terms.

The motion perceptual representation is also defined by excitation and boundaries:

\[
\psi(\vec{d}) \Leftarrow \text{diffusion}(X = e(\vec{d}), \ I = \emptyset, \ Z = \sum_k p(\vec{k})).
\]  

(31)

where the spread of motion signals is stopped by boundary signals \(p\). The depth-order perceptual representation is defined by the depth pools

\[
\xi(s) \Leftarrow \text{diffusion}(X = V(s), \ I = \sum_{s \neq s} \xi(\vec{s}), \ Z = \sum_k p(\vec{k})),
\]  

(32)

where the excitatory input \(V(s)\) is defined according to

---

**Figure 9.** An explanation of the accretion and deletion detection using time-delayed signals. (A) Motion signals disappear at a boundary (left side) and move together with a boundary (right side). Motion is expected to continue; if it does not, the lack of inhibition indicates a discontinuity that can be detected by the red and green receptive fields. (B) The specific detector arrangements needed for the four basic conditions.
This means that if there is no motion, the bias is towards the middle pool. If there is accretion or deletion, there will be a push signal toward the far pool as well as a pull signal toward the near pool, but not at the exact same location. If there is motion, then the corresponding area will be pushed to the near pool. If there is no boundary, the push and pull signals will generally cancel each other out, whereas the presence of a boundary allows the differences to be perceptually realized.

Lists of parameters

The various shunting equations are generally computed at equilibrium, except for a few that are computed dynamically using ordinary differential equation integration. The dynamically computed layers are $a()$ (Equation 16), $d()$ (Equation 19), $\psi(\tilde{d})$ (Equation 31), and $\xi()$ (Equation 32). Table 1 lists the relevant parameters used in the simulation (the parameters for the FORM system are not included). Most are the same as those used by Barnes and Mingolla (2013). While we have not done a formal dimensional analysis, we mention in the table when the ratios of some parameters seem to matter.

Differences from Barnes and Mingolla

The following change was needed to make the model work with grayscale imagery:

1) Eliminating the smoothing of motion discontinuities (i.e., Barnes & Mingolla, 2013, equations 20–23). The elimination caused noisier signals, but they were no longer completely smoothed away, which is important because the motion signals from grayscale imagery are weaker.

A further change was needed to make the model work with the Moonwalk illusion:

2) Modifying all local processing so that instead of computing the maximum or minimum of the whole simulated field, it applies to a local area only. This change was essential to be able to handle the effects of a flickering field of dots, which generate spurious accretion and deletion events simultaneously.

The additional changes needed to make the model work for the textured version of the Michotte rabbit hole phenomenon are as follows:

3) Modifying the filtering of elementary motion signals so that the surround inhibition filters are elongated in the direction of motion. This was necessary in order to properly detect the smaller and finer features of the moving circle.

4) Localizing depth-order processing by creating push–pull signals that get filled in where appropriate.

Table 1. Parameters used in the simulations.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Explanation</th>
<th>Value used by Barnes and Mingolla (2013)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A$</td>
<td>0.1</td>
<td>Should be a lot smaller than $B$</td>
<td>0.1</td>
</tr>
<tr>
<td>$B$</td>
<td>1.0</td>
<td>Arbitrary but simple choice</td>
<td>2.0</td>
</tr>
<tr>
<td>$C$</td>
<td>0.3</td>
<td>Works best if smaller than $B$, but not too small</td>
<td>0.4</td>
</tr>
<tr>
<td>$K$</td>
<td>1.0</td>
<td>Not manipulated in this study</td>
<td>10</td>
</tr>
<tr>
<td>$\theta$</td>
<td>0.0</td>
<td>Baseline threshold used for most layers</td>
<td>0.0</td>
</tr>
<tr>
<td>$\sigma_1$</td>
<td>1.0</td>
<td>On-center for same direction, off-surround for other directions; small is better—do not want to spread out too much</td>
<td>1.0</td>
</tr>
<tr>
<td>$\sigma_2$</td>
<td>4.0</td>
<td>Length of off-surround for same direction, should be larger than width ($\sigma_3$)</td>
<td>8.0</td>
</tr>
<tr>
<td>$\sigma_3$</td>
<td>2.0</td>
<td>Width of off-surround for same direction</td>
<td>8.0</td>
</tr>
<tr>
<td>$F_1$</td>
<td>0.1</td>
<td>Factor for other directions off-surround</td>
<td>0.1</td>
</tr>
<tr>
<td>$F_2$</td>
<td>0.01</td>
<td>Factor for same direction off-surround</td>
<td>0.01</td>
</tr>
<tr>
<td>$\sigma_a$</td>
<td>6</td>
<td>Radius of uniform filter for max direction</td>
<td>—</td>
</tr>
<tr>
<td>$\theta_a$</td>
<td>0.2</td>
<td>Threshold for max direction, ignoring residual noise</td>
<td>—</td>
</tr>
<tr>
<td>$\sigma_4$</td>
<td>0.5</td>
<td>Length of off-surround for motion discontinuity</td>
<td>—</td>
</tr>
<tr>
<td>$\sigma_6$</td>
<td>2.0</td>
<td>Width of off-surround for motion discontinuity</td>
<td>—</td>
</tr>
<tr>
<td>$K_v$</td>
<td>1,000</td>
<td>Amplifying the motion for near depth pool</td>
<td>—</td>
</tr>
<tr>
<td>$\delta$</td>
<td>1,000</td>
<td>Delta used in diffusion numerator, needs to be high for diffusion to spread far enough</td>
<td>10</td>
</tr>
<tr>
<td>$\epsilon$</td>
<td>1,000</td>
<td>Epsilon used in diffusion denominator, needs to be similar to $\delta$</td>
<td>1,000</td>
</tr>
</tbody>
</table>

\[
V(s) = \begin{cases} 
1 + 8f_{\text{pull}}() + 2 \sum_d \min(1, K_v \psi(\tilde{d})) & s = 1 \text{ (near)} \\
2 & s = 2 \text{ (middle)} \\
1 + 8f_{\text{push}}() & s = 3 \text{ (far)} 
\end{cases} 
\]
5) Removing the need for “static” motion signals. This created better localization of moving edges and also simplified the computations needed.

6) Including a short-but-wide off-surround for the motion discontinuity detectors.

The parameters modified for the simulation are shown in Table 1.

### Implementation details

The simulations were performed in MATLAB using Simulink together with several MATLAB and Simulink toolboxes. Each stimulus was represented by 64 × 64 grayscale images, with 21 frames. The simulations were run for 20 time units, one frame per time unit; a new frame is presented at the beginning of each time unit without any intervening “blank” space. The last frame of the stimulus (Frame 21) is not really used, as it is presented just as the simulation ends. Four orientations are used, and eight directions. All simulations used a fixed-step Runge–Kutta (ode4) solver; the simulations generating the push–pull and boundary signals used a step size of 0.02, and the filling-in simulations used a step size of 0.0005.

The simulation figures that follow show the push–pull signals, depth maps, and other signals at the end of the simulation. The depth maps are generated from the three depth pools by combining the information at each location according to the simple formula

\[
\text{depth}[-1, +1] = \frac{(\xi(1) - \xi(3))}{(0.1 + \xi(1) + \xi(2) + \xi(3))}
\]

The Simulink models and other code necessary to run the simulations are available at the lab Web site (http://www.neu.edu/cvl).

### Results

Results are shown for the classical conditions, shearing, the Moonwalk illusion and related stimuli, and finally for the textured version of the Michotte rabbit hole effect.
Model results on classical stimuli

In order to demonstrate that the model works on the classical accretion/deletion phenomena, we show in Figure 10 the push–pull signals and in Figure 11 the colorized depth maps for the stimuli described in Figure 1. The model contains three depth pools: near, middle, and far. In these figures, the three pools have been combined into a single dimension from near to far. In the maps, near is represented by the red pixel value and far by the blue pixel value. Because the three depth pools compete with each other for representation at each location in the visual field, only one pool is active at each location. Therefore this simple color scheme is sufficient.

Figure 12 shows the state of the various layers of the model while processing the “deletion-static” stimulus shown earlier. The left side is moving to the right, deleted at the vertical boundary; the right side is stationary. The figure shows that the FORM stream does not initiate boundary formation. Instead, the MOTION stream detects and filters the motion, finding the discontinuity and passing it to the FORM stream for inclusion in boundary formation.

Model results for the shearing stimulus

It has been shown that a simple shearing stimulus without any accretion or deletion can still generate a depth-ordering that is consistent across observers. The area that is moving faster is seen as nearer the observer.

Figure 13 shows the result of the model on a simple shearing stimulus. The bottom half is moving, and the output of the model shows that the bottom half is also seen as nearer (red).

The boundary between the two areas looks sharp as a result of the difference in the stationary top part and
the moving bottom part. The border of the depth ordering is somewhat fuzzy, but that is actually consistent with the percept. The actual percept has a very slight depth difference, which should therefore be represented as a very small difference, which is consistent with the model output.

The Moonwalk illusion

Our model also explains why the Moonwalk stimulus is perceived the way it is. Because the Moonwalk illusion is a nontrivial stimulus, and because it engages many parts of the model, it is useful to explore it in some detail.

The Moonwalk illusion results when the aperture that contains the moving surface undergoing accretion and deletion is surrounded by a randomly flickering surface. There are two keys to understanding this effect: The first is to consider what happens at the boundary between the moving surface and the flickering surface; the second is to consider just the random flicker and types of motion signals that are generated. Therefore we started by simulating a simple stimulus with a moving surface that is deleted at the boundary on the left side, and a randomly flickering surface on the right. This situation is shown in Figure 14A.

As before, the FORM system will not detect any boundaries (other than the boundaries of individual dots), so the only boundaries that will be represented are those generated by the MOTION system. Normally accretion and deletion signals would combine to form an input into the FORM system and create a boundary. However, because the flickering surround interacts with the actual accretion and deletion signals, causing them to be misplaced, boundaries cannot be completed. This means that there are no sharp depth boundaries.

The randomly flickering dots that form the surround in the Moonwalk stimulus create random motion signals. A single dot appearing and disappearing will create motion signals moving away from this dot. Consequently, small groups of random dots that appear and disappear in random ways will generate motion signals in many different directions. The
directed center–surround filtering actually enhances the persistence of these motion signals and makes it more likely that spurious accretion and deletion signals are generated. This is in fact what occurs, with both push and pull signals being generated for much of the flickering surround. However, because both push and pull signals are generated, there is no net effect on the perceived depth of the surround.

The model contains four separate depth-biasing signals (tonic-neutral, push-to-back, pull-to-front, and motion-bias-pull). The explanation of the Moonwalk stimulus involves all of these signals. The flickering surround gets the tonic-neutral, wide coverage of the push-to-back and pull-to-front signals, and only a small amount of motion-bias-pull; the net effect is neutral, resulting in a middle depth. The moving central aperture, on the other hand, gets only the tonic-neutral and a consistent motion-bias-pull, yielding a slight net pull towards the observer and resulting in a somewhat near depth for this part of the stimulus. As mentioned before, the boundary between the regions is fuzzy, and thus the depth differences are both small and indistinct.

In Figures 14 and 15 the flickering edge stimulus is shown in panel A. This is a simplified version of the Moonwalk illusion with only a single edge. Panels A and B form an interesting comparison because the left side is moving to the right in the same way in both panels, but the results of the depth ordering are completely different. The output of the actual Moonwalk stimulus is shown in panel C, and a relevant comparison—a surface moving behind an aperture—is shown in panel D.

The texture version of the Michotte rabbit hole illusion

The textured version of the Michotte rabbit hole effect consists of a circle of random dot texture moving across a field of the same texture. If the motion stops at any point, the circle melts into the background. As the circle moves across the background, eventually it reaches a part of the background that seems to be in front of the moving circle. Thus the circle seems to disappear behind part of the background. It is as if a slit appears and the circle slides into it.

The depth ordering that one might expect from this stimulus is shown in Figure 16A. As in the colorized depth maps from before, red is used to denote depths closer the observer and blue is used to denote depths farther from the observer. In panel A, the circle is shown in gray, as it is reasonable to think of it as all at one depth. However, the background must be behind the left side of the circle and in front of the right side of the circle. Thus the background is not all at the same depth. Alternatively, one could consider that the background is all at one depth, but then the circle would have to be slanted in depth. Figure 16B shows what we think is a compromise, where both surfaces are slanted where it is necessary. Here the background has some slant and the circle is also slanted; in particular, where it appears to slide into the “slit” it also bends in depth.

The rest of Figure 16 shows the result of the simulation using the model. Panel C shows the push–pull signals, push-to-back in blue and pull-to-front in red. Panels D–F show the filled-in depth ordering for
different positions of the circle. At first, the front edge of the circle causes the background to the right of it to be pushed to the back. As the circle slips into the slit, the background on the other side of the slit is lifted nearer the observer. The trailing edge of the circle is still pushing the background to the back.

The model’s treatment of the Michotte rabbit hole effect is perhaps more straightforward than that of the Moonwalk illusion. There are several parts of the Michotte phenomenon that are interesting and addressed by the model: (a) the perceived depth is dynamic—i.e., it varies over time—especially around the slit, (b) the circle appears in front, (c) except where the circle goes into the slit, and (d) the slit appears and disappears and has structure in depth.

### Discussion

The FMO model produces a DOFM map for each moment of a stimulus. The DOFM is different from the motion percept that is also computed in the model and also different from luminance and color percepts that are computed as well. Therefore, it is possible for the motion percept to have sharp discontinuities while the DOFM percept might not be so sharply defined in depth.

Surface motion where one surface is in front of another generally produces two types of cues that can be used to reconstruct the depth ordering: accretion/deletion (AD) and boundary flow (BF). The AD cue is the actual accretion or deletion of texture elements at the border. The FMO model actually computes the accretion and deletion of motion signals, not texture elements; the implications of this are discussed more thoroughly later. The BF cue is the correlation between the front surface and its edge. Thus, if the texture of a surface is moving in a manner correlated with the boundary of the surface, then this is a strong BF cue that this surface is in front.

The FMO model ignores the BF cues in favor of the AD cues. This is not to say that the BF cues are not
important; rather, they are complementary to the AD cues. The primate visual system most likely uses several redundant and complementary mechanisms for all manner of computations, including DOFM. Thus models that use BF cues, such as that of Layton and Yazdanbakhsh (2015), could possibly be combined with the FMO model in order to take full advantage of all the available cues. For example, stimuli that are not random-dot based, where there are BF cues but no AD cues—such as the demos of Brooks and Palmer (2003)—cannot be explained by the FMO model. Some additional mechanism that uses the BF cues is necessary at some point to explain the full range of kinetic depth phenomena.

Similarly, stereopsis has been ignored in this model, since it does not depend on motion. If stereopsis were available for both the figure and background surfaces, then this information would be very useful in determining the relative depth. We envision binocular disparity cues providing extremely strong push–pull signals that would likely dominate the depth-order generation within Panum’s fusional area. Outside of this region, AD cues provide the most reliable cues to depth order. Of course, in natural scenarios the textures will not usually be as dense as in our simulations, but there will be other cues (contrast, color) that help delineate the boundaries that the AD cues can interact with to generate depth order.

The FMO model uses the AD cues to predict where kinetically defined boundaries might be located. The FORM stream contains luminance-based boundary-fragment detectors; these can be simple Gabor-type filters (as earlier) or some other edge-detecting filter. Wherever AD cues are present, signals are passed to the FORM stream so that these locations can be added to these boundary fragments and the boundary formation process can complete the fragments into boundaries. As a result, the boundaries can be completed using fragments defined by luminance, motion, or a combination of the two. It is also worth noting that although AD cues influence the boundary formation process, kinetic edges are not explicitly computed in the model.

Froyen, Veldman, and Singh (2013) explored concavity and convexity of boundaries, yet another set of
cues that contribute to the perception of figure or background. They suggest that even models that include a FORM stream still need to incorporate processing such that the shape of the border can influence the perceived depth ordering. The FMO model currently lacks any mechanism to determine concavity or convexity, or to have the shape influence the outcome of the depth ordering.

**Accretion/deletion and the Moonwalk stimulus**

In their article discussing the Moonwalk illusion, Kromrey et al. claim that the fact that “the visual system cannot use the AD cue by itself to determine DOFM in a ‘bottom-up’ fashion, suggests that the extraction of AD information is closely associated with segmentation processes” (2011, online publication). Indeed, the key to fully understanding the FMO model is understanding the interaction between the MOTION and the FORM streams. In the model, the AD cues are computed in the MOTION stream and communicated to the FORM stream, perhaps through the recently rediscovered vertical occipital fasciculus (Yeatman et al., 2014). Thus the extraction of AD information and segmentation processes are inextricably linked. The interstream connection is not purely bottom up, but it is also not a top down, and it shows how interdependent some of the computations of the visual system must be.

If the AD cues do not form a clearly defined boundary, then the boundary cannot be completed. In the Moonwalk illusion, the AD cues do not form such a boundary because the flickering surround cases spurious AD signals. As a result, no boundary is formed, and the central area is perceived in front. Thus, even though the AD cues are present, the depth ordering computed by the model is disrupted by the lack of a clearly defined boundary.

The AD cues generate push–pull signals that, when combined with boundary signals, create an appropriate depth ordering. The two basic conditions are shown in Figure 7: Push–pull signals together with a boundary signal generate a strong depth difference; and push–pull

![Figure 16](https://jov.arvojournals.org/pdfaccess.ashx?url=/data/journals/jov/934201/)
Accretion/deletion and the textured Michotte stimulus

The textured Michotte rabbit hole effect provides a simple yet effective stimulus that shows how seemingly contradictory depth perceptions must be combined into a coherent percept. The apparent contradictions are that both the moving circle and the background appear to be flat, yet the circle can still slip into a slit formed in the background. Any model of DOFM must be able to provide an account of this effect. The FMO model account previously described suggests that neither the circle nor the background is entirely planar, but must bend in depth, and that is how the circle can slip into the slit. This slight variation in apparent depth across surfaces is actually a surprising finding that was unexpected at the outset. However, as a result, one prediction of the model is that the perceived depth of these stimulus elements should vary as shown in Figure 16B. We do not yet know how such small differences in perceived depth can be measured.

One of the new features of this version of the model is the introduction of the push–pull signals. A possible alternative would be to have only push signals without the pull on the other side of the edge. This is basically equivalent to an implicit pull wherever there is no push, and thus would not be all that different. Furthermore, a pull signal is needed to accomplish the bias where motion on its own causes a surface to be seen as nearer. Additionally, in a stimulus such as the textured Michotte rabbit hole effect, one perceptually feels as if the background is pushed back around the circle as it moves; one also feels that the slit is opened by pulling the edge closer to the observer. These are highly subjective speculations, and it is therefore still possible that push signals alone could work.

As it is formulated, the FMO model uses explicit filling in (Pessoa, Thompson, & Noé, 1998) of the layers representing different depth pools. Whether or not explicit filling in is necessary, we believe that the keys are the push–pull signals that determine the course of the filling-in and of course the boundaries, which cause depth differences to be enhanced or minimized. If an alternative mechanism such as that of Keil, Cristóbal, Hansen, and Neumann (2005) is used, then again, both boundaries and push–pull signals must be used. The results described here suggest that the filling-in process cannot be ignored, as the model depends on the interaction between the signals.

The standard Michotte rabbit hole effect with a black circle moving on a white background is just about as effective as the textured version. In the textured version, the circle is always clearly moving above the background, whereas in the standard black-and-white version it is possible to see the circle moving with the background. It is not possible to determine with certainty whether the background moves or not in the standard version; but when the texture is present, there is certainty. However, because there is no texture, the FMO model cannot explain the standard version. In fact, any nontextured motion display is problematic. Some possible ways to deal with this issue would probably need to recognize the whole circle as an object moving coherently, the general conservation of object area, and possibly tracking of objects even when occluded (Marshall, Alley, & Hubbard, 1996).

Accretion/deletion of motion signals and the Great Roe stimulus

The model described here looks for the accretion or deletion of motion signals, not the accretion or deletion of texture elements. In the case of realistic imagery, this would seem to be a rather pedantic distinction of no real practical concern, but it does provide a possible way to experimentally test this aspect of the model. It seems very unusual, probably impossible ecologically, to find a situation where surfaces generate the same motion signals while otherwise being indistinguishable. However, if it is possible to create such a situation, then it might be informative about what types of representations are used by the visual system to represent motion and to detect differences in motion.

For example, if only motion signals are represented and nothing else about specific texture elements, then there should be no boundary between two such surfaces. On the other hand, if there is a boundary that is just as distinct as when there is classic accretion or deletion, then the motion signals are probably irrelevant and some other information about specific texture elements is being represented. If observers can perceive the boundary, it is not because of different motion signals on the left and right side of the boundary; it must be because of some other reason.
In order to test the model, we created a stimulus where the motion signals on each side of the (invisible) boundary are identical and the texture looks identical (has the same statistics) but where the actual texture elements are different. We created such a stimulus using one sheet of moving random dots on the left and a sheet of similarly moving random dots on the right, but not the same sheet. In other words, the statistics of the random dots and the motion are both identical. We call this the “Great Roe stimulus,” a reference to the Borgesian chimeric creature described by Allen: “The Great Roe. A mythological beast with the head of a lion and the body of a lion, though not the same lion” (1974, p. 20). While it is not clear whether the mythical beast has a clearly visible boundary between the head and the body, observers still see in our stimulus a clear boundary between the moving sheets of random dots. However, the boundary does feel “diminished” somehow compared to the classical accretion and deletion stimuli with at least one static surface. Our initial conclusions are that the motion signals are indeed important, but that they are not the only cue to seeing the boundary between the moving sheets. We are continuing to investigate this phenomenon.

Keywords: computational modeling, accretion/deletion, depth order from motion, Moonwalk illusion, Michotte rabbit hole phenomenon

Acknowledgments

This work was supported by a grant from the Air Force Office for Scientific Research, AFOSR (FA9550-12-1-0436), and also by a grant from the Office of Naval Research, ONR (N00014-13-1-0092). We thank them for their generous support.

Commercial relationships: none.
Corresponding author: Harald Ruda.
Email: harald.ruda@post.harvard.edu.
Address: Computational Vision Laboratory, Department of Communication Sciences and Disorders, Northeastern University, Boston, MA, USA.

References


Supplementary materials

These supplementary movies are essential for understanding the stimuli that are discussed in this article.

Movie 1. Deletion with static boundary (Figure 1A, simulated in Figures 10A, 11A, 12, 14B, and 15B).
Movie 2. Accretion with static boundary (Figure 1B, simulated in Figures 10B and 11B).
Movie 3. Deletion with moving boundary (Figure 1C, simulated in Figures 10C and 11C).
Movie 4. Accretion with moving boundary (Figure 1D, simulated in Figures 10D and 11D).
Movie 5. Shear (Figure 6A, simulated in Figure 13).
Movie 6. Flickering edge (simplified Moonwalk, simulated in Figures 14A and 15A).
Movie 7. Moonwalk with circular aperture (Figure 2, simulated in Figures 14C and 15C).
Movie 8. Actual circular aperture (simulated in Figures 14D and 15D).
Movie 9. Standard Michotte rabbit hole effect (Figure 3).
Movie 10. Textured Michotte rabbit hole effect (simulated in Figure 16).
Movie 11. The Great Roe stimulus.