*Behavior Research Methods, Instruments, & Computers*
*1997, 29 (1), 37–46*

# Inverse 3-D graphics: A metaphor for visual perception

DANIEL KERSTEN
*University of Minnesota, Minneapolis, Minnesota*

There are two key elements in defining the problem of visual perception. The first is that useful information about the world, such as the shape, material, illumination, and spatial relationships of objects, is encrypted in the image. Second, the encryption process, of going from a description of the world to an image, is not in general reversible. Any single source of image information is usually ambiguous about its causes in the scene. Seeing is the process of decoding the image information. 3-D computer graphics simulates the process of encrypting scene information into the image. By creating images from synthetic scenes, we can gain insights into the constraints used by the visual system to decode image information, and we can begin to bridge the gap between the simple images of the laboratory and complex natural scenes. Computer graphics modeling and animation tools provide the means to generate stills and animations that produce strong perceptual interpretations, yet are theoretically indeterminate. I will describe several illusions involving computer renderings and animations that illustrate the constraints human perception uses to solve ambiguity about material, shape, and depth.

Computer graphics technology provides the means for an unprecedented control of visual information. While our understanding of early visual processing has informed the science of display design, the technology has also handed back an extraordinarily flexible set of tools for the study of human visual perception. In this paper, I will first introduce 2-D and 3-D computer graphics tools and their role in the study of visual perception. In the following main sections, I will ask two questions: (1) What is 3-D graphics? And (2) What is visual perception? The reader should not, of course, expect either a thorough treatise on 3-D computer graphics, or a deep answer to the nature of perception. However, appropriate answers to both questions lead naturally to a metaphor of visual perception as *inverse 3-D graphics*. 3-D graphics specifies how to make an image from a 3-D scene, whereas inverse 3-D graphics specifies how to construct a scene from an image. This metaphor of inverse 3-D graphics, in turn, leads to experimental questions about how perception resolves ambiguities in going from image data to descriptions of the 3-D scene. Finally, in the third main section of this paper, I will illustrate how we have used 3-D graphics in two domains: lightness perception, and movement in depth. I will begin here by distinguishing 2-D and 3-D computer graphics.

## 2-D Graphics and the Proximal Stimulus

2-D graphics software and hardware technology provides environments to simulate the processes of drawing and painting, with the added advantage of direct and precise control of the image color and intensity at each pixel. Historically, both vector and raster-based 2-D graphics have been used to measure the limits of human spatial, temporal, and color visual discriminations. In the traditional terms of perceptual psychology, 2-D graphics specifies the *proximal stimulus* to vision. Control over the proximal stimulus provides for the study of intermediate-level organization processes in vision, such as illusory contours and their relation to surfaces (Nakayama, Shimojo, & Ramachandran, 1990). 2-D graphics allows one to control the image information used to specify contours at a level appropriate for understanding early and intermediate-level vision (Cavanagh, 1987). However, in everyday visual functioning, scene factors such as object shape, depth, lighting, and material together with the eye's optics interact in rich and complex ways to determine the proximal stimulus. The scene parameters specify what perceptual psychologists have traditionally called the *distal stimuli* to vision. A distal stimulus, such as object size (say, 10 cm), gives rise to a pattern of intensities on the retina. This retinal pattern contains information about the proximal stimulus—a corresponding retinal size (say, a 1-mm patch). For the study of vision, information about distal stimuli can be specified either implicitly, in terms of image intensities and 2-D geometry (2-D graphics), or explicitly, in terms of scene parameters and camera specification (3-D graphics). 2-D graphics has the advantage of giving flexible and direct control of the image geometry and intensities—the input and proximal stimulus to vision—but at the expense of an exact description of possible scene causes of the image.[1]

## 3-D Graphics and the Distal Stimulus

Visual perception is more than the receiving of images; it entails the understanding of images in terms of the

I thank Cindee Madison for useful comments and suggestions. This research was supported by NSF Grant SBR-9631682. Correspondence should be sent to D. Kersten, N218 Elliott Hall, Psychology Department, 75 East River Road, University of Minnesota, Minneapolis, MN 55455 (e-mail: kersten@eye.psych.umn.edu).

 

causes, in the scene, of image intensities. 3-D graphics provides the experimenter with the technology to simulate how images are naturally caused. In contrast to 2-D graphics, 3-D graphics has the advantage of direct control or simulation of the scene or distal stimulus parameters, but with the drawback of confounding these causes in the image. However, as elaborated in the next section, this inherent "drawback" simulates the natural encryption process of image formation, and it can be exploited in the attempt to understand how human perception decodes the image to extract useful information about the distal scene. Insofar as vision is a process that extracts useful scene parameters, 3-D graphics provides technology for manipulating scenes so that we may study perception's determination of scene parameters from image data. In the remainder of this paper, I will focus on the technological and scientific possibilities of 3-D graphics for perception. An understanding of 3-D graphics is important for an appreciation of its potential. However, because there is a large gap between real and virtual 3-D scenes, it is also important to understand its limitations. The fundamentals are reviewed in the next section.

## WHAT IS 3-D GRAPHICS?

3-D graphics simulation is a large, complex, rapidly changing field.[2] This section provides a broad overview of 3-D graphics at the level of the application user, rather than the programmer. Let us consider the 3-D computer graphics programming environment as a simulation of some of the jobs one would expect to find among a film production crew.[3]

### Carpenter

A basic component is the modeling software required to build and design objects—a "carpenter." Most modelers characterize objects in terms of polygonal surfaces joined together. This representation takes advantage of built-in hardware for manipulating polygons in 3-D, which works fine for polygonal objects; but smooth objects require many polygons for a good approximation. Spline-based models offer a more flexible alternative for smoothly varying shapes. A variety of tools for manipulating shapes go beyond mere carpentry; they include machine-shop metaphors of twisting, lathing, and extrusion, as well as the fuzzier concepts of sculpting with clay. Flexible modeling tools provide the opportunity for generating novel stimulus classes for studies of shape-based recognition (Bülthoff & Edelman, 1993; Gauthier & Tarr, in press) as well as shape perception (Bülthoff & Mallot, 1988; Mamassian, Kersten, & Knill, 1996; Todd & Mingolla, 1983).

Many objects are not rigid, and advanced modeling techniques compute the change in form of articulated objects, such as the linked segments of the human body, using inverse kinematics. For example, in order to create a scene with a walking human figure, the user specifies the start and end points of, say, a hand, and the computer calculates the joint angles required to achieve that trajectory. One step beyond kinematics is to model the dynamics, masses, and moments of inertia of objects as well as the forces between them. To date, studies of visual perception have been primarily limited to the simpler modeling tools. In the examples below, I will focus on the rendering and kinematics of simple nonarticulated objects that can be easily modeled as polygonal approximations.

### Painter

Real objects have more properties than geometric ones; they are made of "stuff." Stuff is characterized by various kinds of surface materials: textures with varying colors and bumpiness, colored paints and pigments, degrees of shininess, translucency, and transparency. Modeling materials is a challenging problem involving an understanding of the physics of how real surfaces reflect light (Cook & Torrance, 1982; Nayar & Oren, 1995), as well as reasonable approximations that seem to work well visually (Bui-Tuong, 1975). Materials can refract light, generate colored interference patterns, and reflect light at wavelengths different from any of those absorbed. Textures depend on how material and shape varies over a wide range of spatial scales. Some material appearances are not solely dependent on the surface material itself. Whether a material appears like chrome, for example, requires a scene environment to reflect off of the material. A brass door knob will not look mirror-like unless there is an appropriate pattern of reflections from the surrounding surfaces.

Most of the basic material properties can be modeled and generated independently in a "material property editor" and then stuck on to the objects as needed. A fundamental property is the reflectance of a surface—the fraction of incident light reflected. But the light reaching the camera also depends on the direction of the light source, and the viewpoint. A classic material model is the "Lambertian" surface, in which the intensity at the image is independent of the viewpoint, and proportional to the cosine of the angle, $\theta$, between the surface normal and the vector pointing toward the light source:

$$\text{intensity} = \text{reflectance} \times \cos\theta$$

One way to model textures is to vary the degree of reflectance as a function of distance along a surface. Computers thus make it easy to apply the texture of a banana onto a telephone.

### Gaffer

Lights too are part of the scene description and must be synthesized. Lights can be single points or extended area lights, as with a fluorescent panel. Realistic illumination is a nontrivial computational problem. The main challenge is that the effects of illumination are not local; they depend on the global geometry of the scene as well as on the material properties of objects. The image color of a surface depends on direct light coming from luminous sources as well as indirect light bouncing off other surfaces (Christou & Parker, 1995; Greenberg, 1989; Hurlbert, 1995; Langer & Zucker, 1994). Cast shadows can

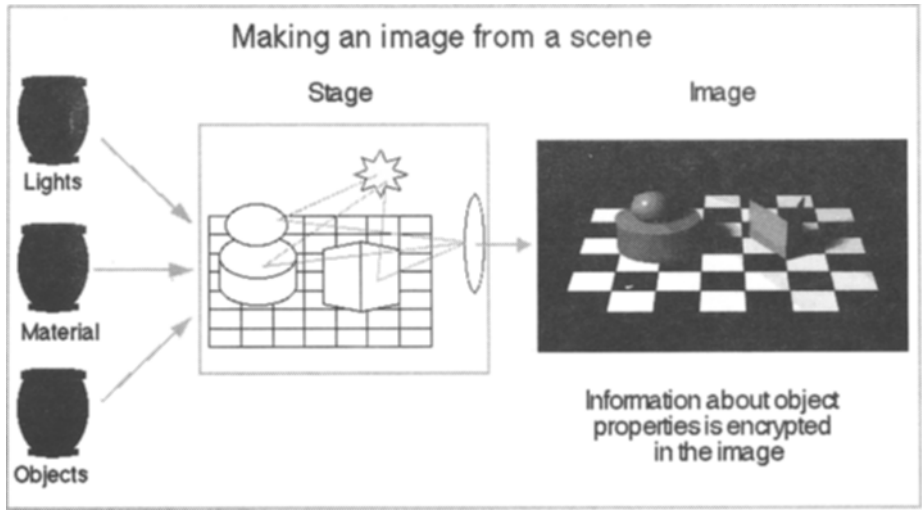


**Figure 1. Illustration of the process of synthesizing images from models of lights, material, and objects. These scene properties can, in principle, be thought of as being drawn from an "urn" which specifies a probabilistic model that constrains object shapes, properties, and lighting. For real objects, object shapes and materials are not necessarily independent.**

be modeled by ray tracing, in which the image intensity at a point is determined only by those rays from a light source which are not blocked by other surfaces. Shortcuts to rendering often avoid ray tracing and approximate or ignore cast shadows because of the computation time required for accurate physical modeling. Below, we will consider an example of ray-traced shadows for a study of depth perception.

### Camera Operator

A number of camera parameters determine the viewpoint and the geometry of projection. A synthetic camera can be set to perspective or orthographic projection. Perspective projection scales an object's image scale size inversely proportional to distance. Orthographic projection maintains a fixed image size proportional to object size and independent of distance, as with telephoto long shots. Some software packages model the effect of depth-of-field on focus, in which large apertures have a narrow depth range of sharp focus, and small apertures have a large range of sharpness.

### Director and Action

The prepared collection of objects, lights, and cameras are assembled onto a stage. The positions, orientations, and properties of the objects—actors and props—can be specified at each point in time with the use of an "animation editor" or "Director." The final stage is "action." Using a program called a *renderer*, the computer simulates rolling the film, and the images are rendered to make an animation. I have left out a description of the final stage of film production, corresponding to film editing. An increasing number of software and hardware packages can be used to arrange the final composition (e.g., Adobe Premiere).

A highly simplified schematic of the 3-D graphics process is shown in Figure 1. The decisions about lights, material, and objects are represented by "urns," because later I will talk about the idea of a stochastic prior model for object, material, and lighting parameters. The fact that 3-D graphics programs treat these processes as independent is related to how human perception understands images.

So what does 3-D computer graphics technology offer to the study of human visual perception?

## WHAT IS VISUAL PERCEPTION?

At this point, it should be clear that a major advantage of the use of computer graphics technology to gain an understanding of visual perception is that it provides the means for an unprecedented control of the stimulus variables in an experiment. I will illustrate this in a sample study later. But let us consider perception itself. The first thing to note is that visual perception's input is the output of processes that make images from scenes. This is true regardless of whether the image is synthetic or real. Visual processing starts off with the image, which is an encryption of the object properties used to make it. The function of human vision is to determine the identities, properties, and relations between objects and between the viewer. But the description of what vision needs is no longer explicitly represented in the image and is more closely related to the elements that a 3-D graphics programmer might have used to make the image than to the image itself. The information about objects, however, is severely encrypted, because variability over illumination and camera (or eye) position can produce an infinite variety of images for a given collection of objects in the world. How does vision decode an image to arrive at ob-

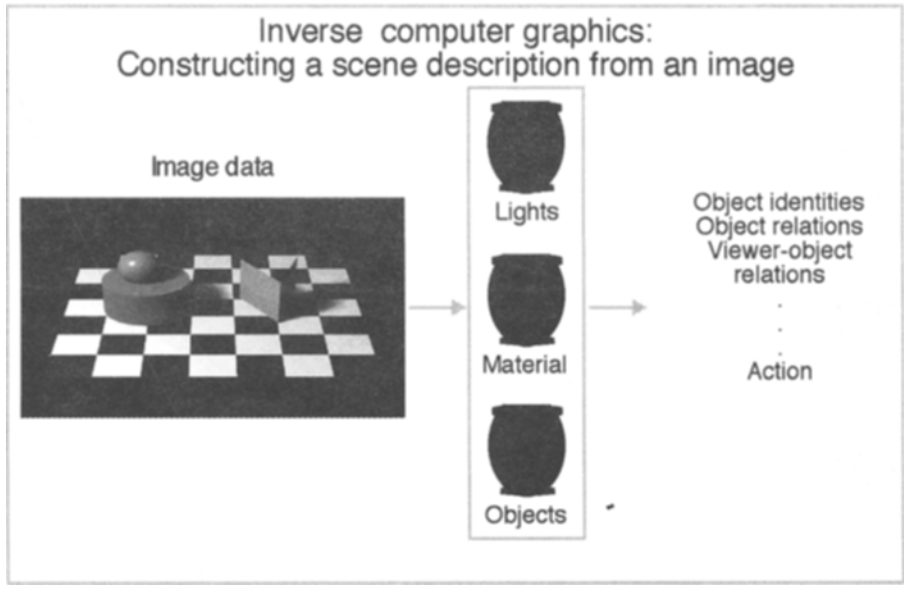


**Figure 2. Schematic of the process of arriving at scene descriptions from image data. The process of inverse 3-D graphics is a metaphor for perception.**

ject properties from images that are theoretically ambiguous?

We can gain significant insight into the process of vision by thinking about visual perception as *inverse 3-D graphics* (Figure 2). This basic idea, of course, is not new. Much research in computer vision over the past couple of decades has been characterized as "inverse optics." And there is the much longer history of characterizing the problem of perception as how one goes from information obtained from the proximal stimulus (the image) to inferences about the distal stimulus (e.g., the object or scene). Clearly, too, there is more to visual perception than just an attempt to infer object shape, material, and lighting properties from images. Decisions about object identity, relations, actions, and categories all involve perception in some form or another. The advantage of thinking about the inverse computer graphics metaphor is that it makes clearer how one can use computer graphics technology to ask questions that would have been very difficult to ask and answer only a decade ago. But first, let us look at some of the problems faced by any system doing inverse 3-D graphics.

## Inverse Computer Graphics: A Metaphor for Perception

The central problem is that information about scene geometry and object properties is confounded in the image data that an eye or robot camera might receive. There are two sorts of ambiguity: geometrical and photometrical. Consider, first, the geometry. Given the 2-D image of what to us looks like a wire-frame cube, there is an infinite set of distinct 3-D objects that project to the same 2-D image. For example, the polygonal representation of a 3-D object is a list of coordinates corresponding to the vertices of each polygon: $\{(x_1,y_1,z_1), (x_2,y_2,z_2), (x_3,y_3,z_3), \ldots\}$. The geometrical component of the image data is also a list of vertices, but under orthographic projection, minus any information about the $z$-coordinates: $\{(x_1,y_1), (x_2,y_2), (x_3,y_3), \ldots\}$. A classic example of this ambiguity is the Necker cube illusion, in which human vision sees only two object descriptions of a line-drawing of a transparent cube. The real puzzle is not why there are two percepts instead of one, but rather: Why does human vision settle on just two out of an infinite number? (For one answer to this question, see Sinha & Adelson, 1993.)

Consider, now, the photometry, and especially changes in image intensity. Here the profound nature of the image ambiguities have only become appreciated with the advent of computer vision research. Intuitively, it seems that intensity edges should be really important for determining the boundaries of objects. But two decades of computer vision research in "edge detection" have not produced an algorithm that can generate a cartoon from a natural image with the sophistication of a cartoonist's copy. What's the problem? A basic problem is that edges can have various degrees of fuzziness and be quite noisy. Intensity changes can occur over a wide range of spatial scales, and the scale of importance cannot be determined from local measurements. For example, the intensity changes in fine wood grain are largely independent of the object's bounding contours. Spatial scale and noise pose well-known challenges to edge detection algorithms (Canny, 1986). But there is a second fundamental problem of establishing edge identity, illustrated in Figure 3.

Given a local patch of intensity change (e.g., as measured by an edge detector), there are many possible causes of that change (Figure 3). The conclusion is that *there is no local source of image intensity information that can*

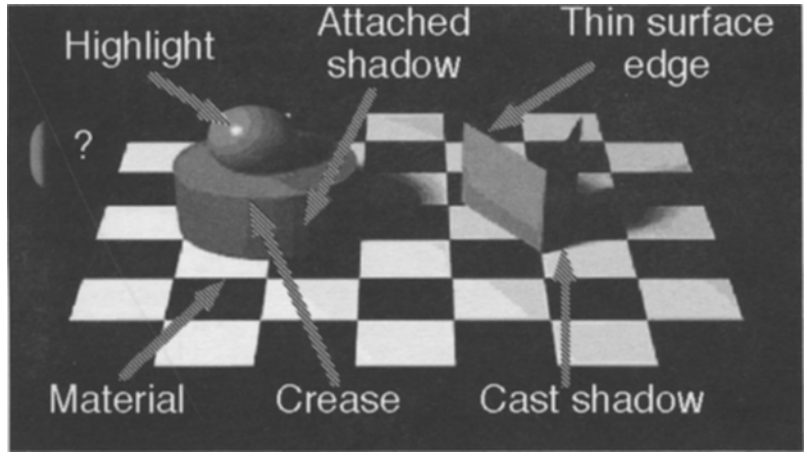


**Figure 3. A measurement of a local change of image intensity, illustrated by the elliptical patch in the upper left, is highly ambiguous as to what in the scene caused it. Changes in material, depth, and surface orientation can create a local intensity change. Cast shadows and specularities also produce similar local intensity changes.**

*completely disambiguate one kind of edge from another* (e.g., a shadow edge from a material change). Edge signatures can be classified as providing various degrees of support for inferences about different scene causes, but the support is weak at best.

For example, the elliptical sample in Figure 3 could be from defocus (not shown), a cast shadow penumbra, or even a shape orientation change or crease at some spatial scale. A solution to the dilemma of local ambiguity is to somehow incorporate global information, in terms of intermediate-level grouping or top-down knowledge about the class of objects that vision typically deals with. I will touch on these points later. So how should we think about the resolution of ambiguity?

## Inverse Computer Graphics as Statistical Inference

A natural framework for analyzing the information available for making reliable decisions in the face of uncertainty is provided by a Bayesian formalism for statistical inference (Adelson & Pentland, 1991; Bülthoff & Yuille, 1991; Kersten, 1990; Knill & Richards, 1996). A brief sketch of this framework should make clear the formal relationship between inverse and forward 3-D graphics. Suppose we have a scene description, represented by a vector of parameter values, *scene*, and image measurements, *image*. (These could, for example, correspond to the two lists [3-D and 2-D] of polygon vertices in the example above.) The inverse graphics problem is "given *image*, find *scene*." One approach is to characterize our knowledge of what constrains the solution as a posterior probability of a scene description given the image data, $p(scene|image)$:

$$p(scene|image) \propto p(image|scene)\, p(scene)$$

Inverse 3-D graphics $\Leftrightarrow$ Forward 3-D graphics & Scene synthesis model

Finding the scene description that maximizes the left-hand side of the equation is the problem of sorting through possible scene descriptions to find ones most likely to have caused the image data. This is a hard theoretical problem because of the ambiguities in the mapping. The key idea from Bayes's theorem is that the probability of a scene description can be rewritten in terms of a forward 3-D graphics model of image formation (the "likelihood" model) and a model of scene synthesis (the "prior" model). These models can be thought of as embodying constraints to resolve the ambiguity of the image data. The first constraint is the probability of the image data, given a 3-D computer graphics scene. If there is no added uncertainty in going from the 3-D scene to the image, then $p(image|scene) = \delta(image - \varphi(scene))$, where $\varphi()$ is the 3-D graphics operation specifying how the lighting, camera, and surfaces all interact to produce the image. $\delta()$ is called a "delta" function and is a filter which is zero wherever the model does not predict the image data, and infinitely high where it does. The main point is that the forward 3-D graphics model completely characterizes the likelihood constraints. However, as we have noted with the Necker cube example, there can remain unresolved ambiguities. The second constraint is the a priori model of the world. Think of a prior model of the world as a statistical model for draws from the "urns" in Figure 1. The urns are filled with paper slips specifying possible scene parameters, and the proportion of identical slips matches the probability of that description. For example, cubes may be a priori preferred over other wireframe objects because of familiarity or more generic constraints such as compactness (Sinha & Adelson, 1993).

Writing down Bayes's rule is only a first step toward solving the problem of inverse optics. A full scene reconstruction is not feasible, because of the huge dimensionality inherent in a scene. Human vision does not compute a full reconstruction at each moment either, and the hard part of the inverse problem is to discover short, yet

functional descriptions of the scene parameters that perception is actually interested in. This is where experimental studies of the constraints and representations used in human vision are important, and where the technology of computer graphics is particularly useful. The Bayesian framework provides a common language and intuition for computer vision, 3-D graphics, and human visual inference. For behavioral scientists, the promising prospect of a Bayesian perspective is to be found in the psychophysical study of constraints that human vision adopts to arrive at unique conclusions about the state of the world from an ambiguous image. Let us consider two examples: empirical studies of human lightness perception and of apparent motion from cast shadows.

## APPLICATIONS OF 3-D COMPUTER GRAPHICS TO PERCEPTION

### Lightness Perception

The left-hand panel of Figure 4 shows a version of a lightness illusion due to Edwin Land and John McCann (1971). As Land and McCann describe, it is closely related to the Craik–O'Brien–Cornsweet illusions that go back to the 1940s and 50s. The observation is that the left side of the two-tone slab appears darker than the right side of the slab, even though the left and right sides have identical intensity patterns—the patterns are both identical luminance gradients.

A number of models have been proposed to "explain" this illusion. Most have the following elements, which are shared by many lightness algorithms (Hurlbert, 1986). The image intensity pattern is differentiated with respect to distance (e.g., via lateral inhibitory processes in the retina). This amplifies rapid intensity changes. Then small values are thresholded out (set to zero), and the signal is integrated. Information about slow spatial changes is thus lost. The big intensity change in the middle of the slab signals an edge between regions of constant lightness, which is interpreted as "darker to the left, and lighter to the right." There are a number of problems with this kind of spatial filter explanation of the Land–McCann and related effects, but rather than focus on the details here, let us consider the problem from a functional rather than a mechanistic point of view—a point of view closely related to inverse 3-D graphics.

How could the carpenter, painter, and gaffer have combined their skills to make the horizontal gradient in Land and McCann's illusion? Suppose that the assignment was given to two different construction teams. It turns out that they could end up making two quite different scene sets to produce the intensity gradients (see Figure 5).

Consider the solution represented by Scene 1 in Figure 5. Here, the carpenter gives a flat panel to the painter who brushes the left and right sides of a flat panel with dark and light gray paint, respectively. Then the gaffer lines up the light sources so that an illumination gradient falls from right to left. (This is what Land & McCann did originally to make this illusion, before the days of convenient computer graphics.) But there is second way to construct a scene to generate two gradients (Knill & Kersten, 1991). This is illustrated on the right side of Figure 5 (Scene 2). The carpenter supplies two cylinders. The painter paints them both with the same medium gray paint. The gaffer arranges the lights so that illumination is strongest on the left.[4]

Knill and Kersten (1991) pointed out that the perceived lightness difference for the cylinder version on the right side of Figure 4 was different than the perceived lightnesses for the flat panel on the left. The lightnesses of the two cylinders on the right appear similar, and to many observers the same. And why not? After all, they were made with the same paint! This perceptual solution depends on information in the curved contours which supports the hypothesis that the two gradients are caused

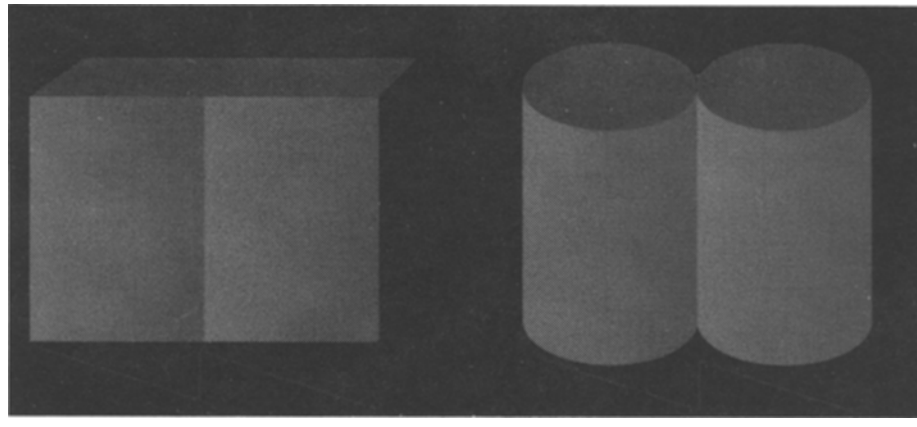

**Figure 4. The intensity pattern in the horizontal direction for the "slab" (left) and "two cylinders" (right) is identical. However, the apparent relative lightness of the left and right sides of the slab is greater than that for the left and right sides of the two cylinders. The lines at the bottom illustrate the light intensity that a photometer might measure going from left to right across the screen. From "Apparent Surface Curvature Affects Lightness Perception," by D. C. Knill and D. Kersten, 1991, *Nature, 351*, p. 228. Copyright 1991 by Macmillan Magazines, Ltd. Adapted with permission.**

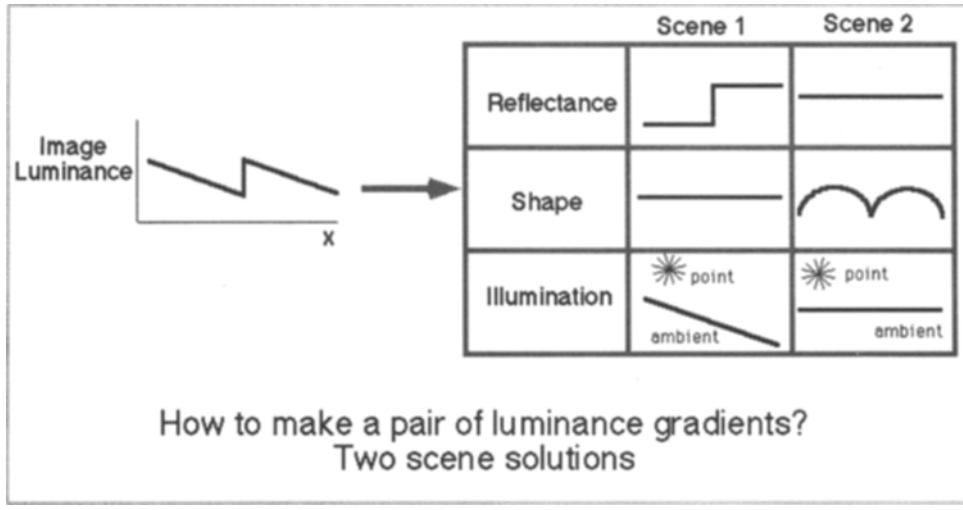


**Figure 5. Imagine two set-design crews assigned the task of producing a scene that generates a pair of horizontal luminance gradients as shown on the left. One easy solution, of course, would be to use a 2-D computer graphics program, or an airbrush. But what if the crews were constrained to make a 3-D scene and could make appropriate surface shapes, arrange the illumination, and manipulate the reflectances with paint? Two possible solutions are shown on the right.**

by a change in surface shape, rather than a change in illumination. Evaluating which of several scene constructions, each of which is consistent with some of the image data, can be constrained by additional image information (e.g., the contour curvature), or prior models. Stereovision can also provide information about curvature that will affect lightness perception (Buckley, Frisby, & Freeman,1994). The lesson from this example is that the visual perception of lightness better resembles a process of inverse graphics, in which perceived lightness is associated with reflectance of the paint, than it does the output of a spatial filter.

### The Perception of Motion in Depth From Shadows

Let us turn to a second example, this time involving the perception of depth from shadows. Artists have known at least since the time of Leonardo Da Vinci that a shadow cast by an object is useful to portray relative depth between the object and the surface receiving the shadow (Yonas, Goldsmith, & Hallstrom, 1978). One might also expect that when an object moves away from a background surface, motion of the shadow will provide information about a change in depth.[5] But there is a problem—there are many cues to object depth which somehow must be integrated (see, e.g., Cutting & Vishton, 1995) . When an object approaches an observer, the image size usually grows measurably. In fact, a very strong cue to change in depth is a change in an object's image size. For example, the rate of change of an object's image size is a very powerful source of information for collision time (see, e.g., Lee & Reddish, 1981). Also, when an object changes depth, the object's image almost always moves relative to the background. The lack of change in either size or position is, conversely, a strong cue to object stationarity.

We wanted to know whether shadows are strong enough to override image information (zero motion) that would normally provide an overwhelming signal supporting the hypothesis, "no motion in depth." Although shadow motion typically accompanies a change in size and position of an object, is shadow motion information strong enough to override these cues to motion in depth? It is not unreasonable to suppose that shadow cues alone would be too weak—human vision may have specific sensitivities to both dilation and translational motion (Regan, 1986). Furthermore, from a theoretical point of view, computing motion using global rather than local information is a difficult, and still unsolved problem.

Computer graphics gives one the ability to control cues that normally covary. We used Wavefront's "Advanced Visualizer" on a Silicon Graphics computer to simulate the motion of a square surface under specific conditions. First, we aligned the camera, the central square, and the background center to be constrained to remain along a line perpendicular to the background. Together with orthographic projection, which maintains constant image size of the objects, this ensured that there was no movement (in the image) of the central square relative to the checkerboard background (Figure 6). This is called an "accidental alignment" of the central square with the background and the line of sight—it almost never happens under typical viewing arrangements. We were interested in whether a moving cast shadow is sufficient by itself to induce an apparent change in depth of the central square.

A physics-based simulation of illumination is computationally intensive for two main reasons: (1) Illumination sources can be blocked, creating shadows. Calculating

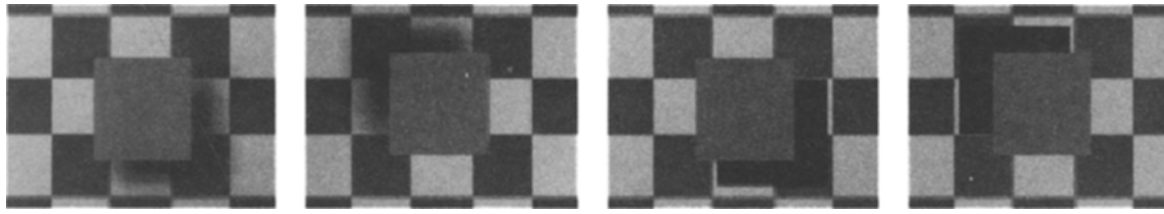

**Figure 6. A computer animation was made in which a central square was moved away from, and then back toward, a checkerboard background. The motion was directly along the line of sight, and the camera was set to orthographic projection, resulting in no size change for the central square. Animation frames for the greatest separation between the central square and background are shown for four lighting conditions. From left to right, they are: extended light source from above, extended light source from below, point light source from above, and point light source from below. Despite the lack of objective image motion of the central square, it nevertheless appears to move in depth for the extended light from above condition (left-most case). A QuickTime movie demonstrating illusory motion from shadows, using an extended light source above the square, can be viewed and downloaded from: http://vision.psych.umn.edu/www/kersten-lab/shadows.html From "Illusory Motion From Shadows," by D. Kersten, D. Knill, P. Mamassian, and I. Bülthoff, 1996. *Nature, 379*, p. 31. Copyright 1996 by Macmillan Magazines, Ltd. Adapted with permission.**

shadows requires knowledge of what surfaces lie between the light source and the surface receiving the shadow (e.g., the central square is between the light and the background checkerboard of Figure 6). (2) Surfaces receive light not only from direct sources, but also from reflections off of other surfaces.

A standard computer solution to the first problem is to calculate the image by using a ray-tracing algorithm. Shadow penumbrae are a result of an extended light source, which can be modeled in terms of density (an infinite collection of sources spanning a finite area), or approximated in terms of a finite set of point sources arrayed on a panel. We opted for the later approximation, with 20 light sources arranged on a rectangular panel, simulating a fluorescent light fixture. Unlike some penumbra approximations, ray-tracing from multiple light sources accurately simulates the increase in blurriness of the penumbra as an object moves further from the background. (It is also possible, with a simple stimulus as in Figure 6, to calculate the penumbra in the image domain directly, and to use 2-D graphics to generate the animations.)

The second problem is particularly challenging, because it means that the intensity of the light at one point of a surface depends on all the other surfaces it "sees" (which also receive light from it!), as well as on the direct light sources. A common and simple solution to the second problem is to approximate all of the nondirect source contributions to illumination as one term, called the "ambient light," which has no direction. Some graphics packages model the ambient light as a component of the reflectance, which is a bit misleading from the point of view of an approximation to the physics of lighting. What it means is that a given surface will have a nonzero intensity even when the direct light sources are turned off. Unlike with directional light sources, the image of a surface rendered with only ambient light shows no dependence on surface orientation. A more sophisticated solution to the problem of indirect illumination is to use a radiosity model (Greenberg, 1989). The surfaces in our simulation were parallel and flat and could not reflect light onto each other. However, more often than not there are other nonvisible surfaces that contribute ambient light to a surface. Given uncertainty regarding these nonvisible surfaces, we chose a simple ambient light model as a reasonable approximation. If there were no ambient term, the shadow umbra would be black.

We measured the proportion of observers who reported seeing apparent motion in depth for four conditions: light from above versus from below; and a point light source versus extended light source illumination (Figure 6). Four groups of 16 observers each were asked to say whether the central square appeared to move in depth or not. We found that when the central square was illuminated with an extended light source from above, all 16 observers reported seeing the central square apparently moving in depth. The strength and reliability of the effect was less for the other conditions (Kersten, Knill, Mamassian, & Bülthoff, 1996).

The percept is compelling, yet an analysis of how the animations could have been produced in other ways using 3-D graphics shows that numerous ambiguities must be resolved if one is to arrive at even a few interpretations. Figure 7 illustrates some of those ambiguities. Again, imagine that we have asked several 3-D graphics programmers (or set design crews) to make a scene scenario to produce this animation. They could have done it in quite different ways.

Consider just the kinematics. A shadow displacement can be achieved by moving the light source or the object. The background rather than the central square could have moved. Consider the material properties. The shadow could be a transparent film surface that moves. The checkerboard background could be transparent, and the shadow could be a surface that is out of focus behind the checkerboard background. How does human vision resolve these ambiguities?

Earlier, I presented a view of human image understanding as Bayesian statistical inference. Recall that ambiguity can be resolved by using two basic types of constraints: ones that depend on how the image is formed,

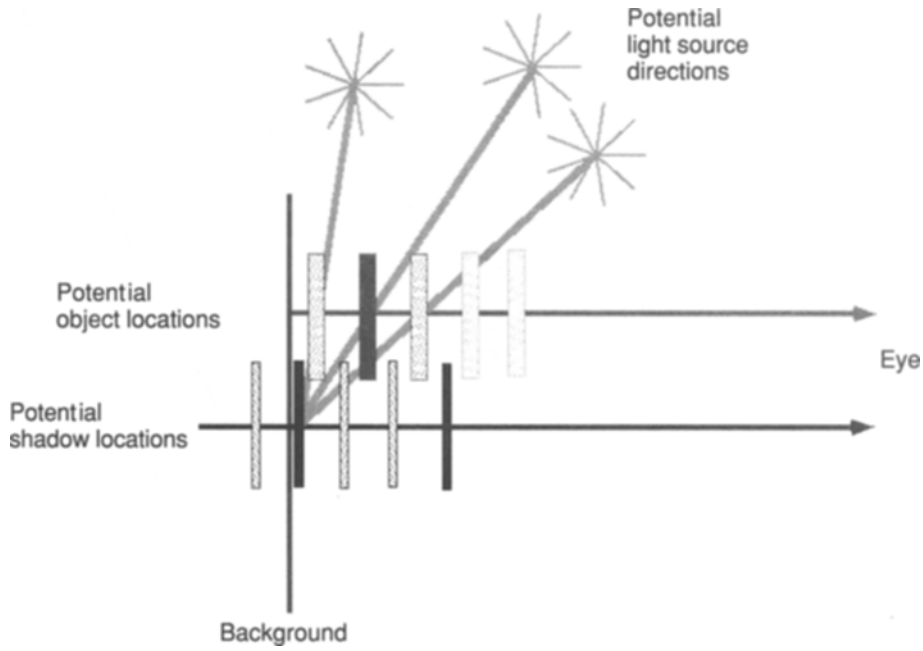


**Figure 7. Illustration of some of the ambiguities in scene interpretation of a moving shadow animation. Both the central square and the dark shadow regions could be anywhere along the line of sight. The background could be transparent and in front of the shadow, rather than the reverse. The light source could be moving, rather than the central square.**

and the a priori constraints that are independent of image informativeness. Image formation constraints include the following: (1) A fuzzy edge is more likely to be a shadow than a material change; (2) when a shadow crosses a material change, the contrast across the material is unchanged. The prior constraints reflect the statistical structure of world properties. For example, on the basis of the analysis above, we can make a plausible partial list of a priori constraints that help resolve the ambiguities of Figure 7: (1) Opacity is more likely than transparency; (2) backgrounds do not move; (3) backgrounds are opaque; (4) light sources do not move. Note that any of these could be violated. A major challenge for vision research is to discover how multiple weak prior constraints and the image data are integrated to arrive at confident decisions of not only what, but where, objects are.

## GENERAL DISCUSSION

It is clear that 3-D computer graphics, through scene creation and rendering, provides the means for the experimental study of perception in ways that would have been extremely difficult just a decade ago. However, there are clear limits to the application of computer graphics technology to understanding human perception. These limits are to be found in both the hardware technology and our theoretical models of scenes and images. To understand perception, we will continue to need research into statistical models of specific domains at both the object and the image level (e.g., human faces; see Hallinan, 1995; Vetter & Troje, 1995). We will continue to require a better understanding of surface properties and illumination. For studies of vision and action, we both need and can expect significant progress in virtual reality technology (VR) in the near future. VR promises unprecedented control of whole environments; however, there are major technical challenges to be faced, such as the achievement of visual and cross-modality cue consistency, as well as the need for higher temporal and spatial bandwidth.[6] For the foreseeable future, experimental design using both computer graphics and VR will require careful consideration of where and how the virtual world of computer graphics departs from the real one. Despite the limitations of computer graphics, we can expect that vision research will become increasingly more limited by our scientific imagination than by our experimental tools.

## REFERENCES

ADELSON, E. H. (1993). Perceptual organization and the judgment of brightness. *Science*, **262**, 2042-2044.

ADELSON, E. H., & PENTLAND, A. P. (1991). The perception of shading and reflectance. In B. Blum (Ed.), *Channels in the visual nervous system* (pp. 195-207). London: Freud Publishing.

BUCKLEY, D., FRISBY, J. P., & FREEMAN, J. (1994). Lightness perception can be affected by surface curvature from stereopsis. *Perception*, **23**, 869-881.

BUI-TUONG, P. (1975). Illumination for computer generated pictures. *Communications of the ACM*, **18**, 311-317.

BÜLTHOFF, H. H., & EDELMAN, S. (1993). Evaluating object recognition theories by computer graphics psychophysics. In T. A. Poggio & D. A. Glaser (Eds.), *Exploring brain functions: Models in neuroscience* (pp. 139-164). New York: Wiley.

BÜLTHOFF, H. H., & MALLOT, H. A. (1988). Integration of depth modules: Stereo and shading. *Journal of the Optical Society of America A*, **5**, 1749-1758.

BÜLTHOFF, H. H., & YUILLE, A. L. (1991). Bayesian models for seeing shapes and depth. *Comments on Theoretical Biology*, **2**, 283-314.

CANNY, J. F. (1986). A computational approach to edge detection. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, **8**, 679-698.

CAVANAGH, P. (1987). Reconstructing the third dimension: Interactions between color, texture, motion, binocular disparity and shape. *Computer Vision, Graphics, & Image Processing*, **37**, 171-195.

CHRISTOU, C., & PARKER, A. (1995). Visual realism and virtual reality: A psychological perspective. In K. Carr & R. England (Eds.), *Simulated and virtual realities: Elements of perception* (pp. 53-84). Bristol, U.K.: Taylor & Francis.

COOK, R., & TORRANCE, K. (1982). A reflectance model for computer graphics. *ACM Transactions on Graphics*, **1**, 7-24.

CUTTING, J., & VISHTON, P. (1995). Perceiving layout and knowing distances: The integration, relative potency, and contextual use of different information about depth. In W. Epstein & S. J. Rogers (Eds.), *Perception of space and motion* (pp. 69-117). San Diego: Academic Press.

DISTLER, H. (1996). Psychophysical experiments in virtual environments. *Virtual Reality World '96*, pp. 1-11. Available: www.mpik-tueb.mpg.de/projects/bicycle/vrworld/poster.html

FOLEY, J. D., VAN DAM, A., FEINER, S. K., & HUGHES, J. F. (1990). *Computer graphics: Principles and practice* (2nd ed.). Reading, MA: Addison-Wesley.

GAUTHIER, I., & TARR, M. J. (in press). Becoming a "Greeble" expert: Exploring mechanisms for face recognition. *Vision Research*.

GREENBERG, D. P. (1989). Light reflection models for computer graphics. *Science*, **244**, 166-173.

HALLINAN, P. W. (1995). *A deformable model for the recognition of human faces under abitrary illumination*. Unpublished doctoral dissertation, Harvard University.

HURLBERT, A. (1986). Formal connections between lightness algorithms. *Journal of the Optical Society of America A*, **3**, 1684-1693.

HURLBERT, A. C. (1995). *Computational models of colour constancy* (Rep. No. NCSS TR 95-04). University of Newcastle upon Tyne, Neural Computation and Sensory Systems Research Group.

KERSTEN, D. (1990). Statistical limits to image understanding. In C. Blakemore (Ed.), *Vision: Coding and efficiency* (pp. 32-44). Cambridge: Cambridge University Press.

KERSTEN, D., KNILL, D., MAMASSIAN, P., & BÜLTHOFF, I. (1996). Illusory motion from shadows. *Nature*, **379**, 31.

KNILL, D. C., & KERSTEN, D. (1991). Apparent surface curvature affects lightness perception. *Nature*, **351**, 228-230.

KNILL, D. C., & RICHARDS, W. (EDS.) (1996). *Perception as Bayesian inference*. Cambridge: Cambridge University Press.

LAND, E. H., & MCCANN, J. (1971). Lightness and retinex theory. *Journal of the Optical Society of America A*, **61**, 1-11.

LANGER, M. S., & ZUCKER, S. W. (1994). Shape-from-shading on a cloudy day. *Journal of the Optical Society of America A*, **11**, 467-478.

LEE, D. N., & REDDISH, P. E. (1981). Plummeting gannets: A paradigm of ecological optics. *Nature*, **293**, 293-294.

MAMASSIAN, P., KERSTEN, D., & KNILL, D. C. (1996). Categorical local shape perception. *Perception*, **25**, 95-107.

NAKAYAMA, K., SHIMOJO, S., & RAMACHANDRAN, V. S. (1990). Transparency: Relation to depth, subjective contours, luminance, and neon color spreading. *Perception*, **19**, 497-513.

NAYAR, S. K., & OREN, M. (1995). Visual appearance of matte surfaces. *Science*, **267**, 1153-1156.

REGAN, D. (1986). Visual processing of four kinds of relative motion. *Vision Research*, **26**, 127-145.

SINHA, P., & ADELSON, E. (1993). Recovering reflectance and illumination in a world of painted polyhedra. *Proceedings of the Fourth International Conference on Computer Vision* (pp. 156-163). (Conference held in Berlin, May 1993).

TODD, J. T., & MINGOLLA, E. (1983). Perception of surface curvature and direction of illumination from patterns of shading. *Journal of Experimental Psychology: Human Perception & Performance*, **9**, 583-595.

VETTER, T., & TROJE, N. F. (1995). *A separated linear shape and texture space for modeling two-dimensional images of human faces* (MPI-Memo No. 15). Tübingen: Max Planck Institute for Biological Cybernetics. Available: ftp://ftp.mpik-tueb.mpg.de/pub/mpi-memos/TR-015.ps.Z

YONAS, A., GOLDSMITH, L. T., & HALLSTROM, J. L. (1978). Development of sensitivity to information provided by cast shadows in pictures. *Perception*, 7, 333-341.

## NOTES

1. Some computer graphics applications specialize in precise draftsman-like control of the geometry, and others provide painter-like control of image intensities and colors. Adobe Photoshop provides painter-style control over images. Canvas from Deneba is an example of an application that provides a blend of geometry and paint control for 2-D graphics. Canvas has been used in recent published studies of brightness perception (Adelson, 1993). 2-D animation packages, such as Macromedia's Director, provide control of 2-D objects over time as well.

2. An excellent source of current information is the SIGGRAPH home page (URL: http://www.siggraph.org). SIGGRAPH is the ACM's special interest group for computer graphics. For a detailed introduction to computer graphics, see Foley, van Dam, Feiner, and Hughes, 1990.

3. Examples of such applications are Macromedia's Extreme 3D for the personal computer, and the Alias/Wavefront package "The Advanced Visualizer" for a high-end graphics workstations with specialized 3-D hardware, such as those made by Silicon Graphics, Inc.

4. The astute reader will also have thought of a third scene construction. The painter could have used an "air-brush" to make reflectance gradients. And in fact, that is what can be done when this illusion is generated using a 2-D, rather than 3-D, graphics program. This third option is the veridical scene description for the printed page—but perception does not seem to "know" this.

5. Cast shadow motion is a standard technique used in video games, such as "Kings of the Beach" for Nintendo, to convey information about where an object is relative to the ground plane.

6. An example of a virtual reality laboratory being constructed for basic research into visual perception and action is at the Max Planck Institute for Biological Cybernetics in Tübingen, Germany (Distler, 1996; http://www.mpik-tueb.mpg.de/projects/bicycle/bicycle.html).