Shading in Two Dimensions

Lance Williams
Advanced Technology Group
Apple Computer, Inc.
MS:60-W, 20705 Valley Green Dr.
Cupertino, CA 95014 USA

Abstract

Shading is a fundamental technique for conveying three dimensional form in two dimensions. Shading can also be used, however, to emphasize the relative importance of different elements in a scene, to establish the visible priority of overlapping regions, and to indicate three dimensionality symbolically by gradations of tone. In animation, shading adds a graphic dimension to two-dimensional objects and characters which is very expensive to achieve by conventional means, and may be of stylistic importance when merging animated figures with live-action scenes and situations. This paper reviews some of the typical computer techniques employed for two dimensional shading functions, and attempts to apply them to generate three-dimensional objects which can be shaded as such. This addresses a problem of persistent interest in the machine vision community: the estimate of three-dimensional shape from silhouettes.

Résumé

La "mise en ombre" est une technique de base pour donner une impression tri-dimensionelle à une forme en deux dimensions. On peut également utiliser la "mise en ombre" pour souligner l'importance subjective de certains objets d'une scène, pour préciser le positionnement de differents éléments selon leur distance relative au spectateur, ou pour indiquer un volume d'une manière symbolique, par un dégradé. En animation, "l'ombre" rajoutte aux objets et aux personnages en 2D une dimension graphique de qualité qui est trés difficile à rendre d'une manière traditionnelle, et qui peut créer un style lorsqu'on veut intégrer des éléments de dessin animés dans une scène en prise de vue réelle. Cet article montre quelques unes des techniques courrament employées sur ordinateur pour faire des ombres en deux dimensions et essaye d'appliquer ces fonctions pour générer des objets en trois dimensions auquels on peut appliquer les techniques de mise en volume appropriées. Ceci renvoit au problème d'un interêt constant, bordé dans le domaine de la vision par ordinateur: l'estimation d'une forme en 3D à partir d'une silhouette.

Keywords: Shading, Texture, Two-Dimensional Graphics, Animation, Extrusion, Silhouette.

Introduction

One of the critical simplifications which makes conventional cel animation economically practical is that objects and characters are represented as outline drawings filled with areas of solid color. Details, texture, and chiaroscuro must be dispensed with in order to generate 8 to 24 images per second of film. For many purposes, this is not a disadvantage; the clear, easily readable visual cues provided by a cartoonist's unshaded line drawings may offer the best explication for many subjects [1]. Graphic designers, however, much prefer to use gradations of tone and color in their palette, and such designs are frequently animated (for short films or commercial spots) at great expense. Rare indeed is an attempt to utilize graduated shading as Disney Studios did in feature-length animation (soft blushes of rouge, for example, were painstakingly applied to Snow White's cheeks in every frame). Not only is shading each frame of animation an extremely labor-intensive undertaking, but the irregularity of shading performed by hand (with an airbrush, for example) often causes the shaded surface to "boil" when displayed in motion.

Our interest in surveying this field is to discover improved methods for computer animation of smooth shading, shading that is flexibly specified, readily computed, smooth within each frame and continuous from frame to frame. Since this is a solved problem for three-dimensional animation, one possible solution is to create a three-dimensional surface from two-dimensional animated art. This converges with efforts of the machine vision community to derive 3D shapes from 2D silhouettes.

Skeletons and Sweeps

Most commercial computer "paint" systems offer region or seed fills to a constant color or specified pattern. Some can fill regions with a linear or conical ramp of intensity, a simple but powerful variation which is a mainstay of computer-aided comic book production [2]. Such shading is not based on the specific shape being shaded; the ramp or cone of intensity is simply matted into the filled region. [3] describes a generalization of this 2D shading scheme, which separates the shading pattern (designated the "support"), an arbitrary colormap it indexes, and the two dimensional shapes to be filled. Two shading techniques which are shape-dependent are in relatively common use: "skeleton" fills and 2D sweeps.

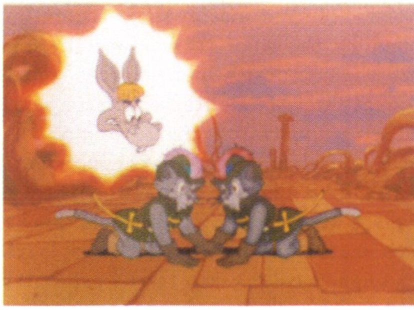
The term, "skeleton" fills, is proposed for algorithms that fill a region with graduated intensities along the trace of a region thinning algorithm of the sort used to compute Blum's "medial axis transform" [4] and related skeletons. An example would be the "shapeburst" fill offered by PixelPaint Professional [5]. The intensity profile of the filled region is defined by the shape of the region, but (like all region thinning algorithms) is critically sensitive to small changes in the boundaries of thick regions [4], and the resulting shaded regions are not smooth, showing derivative discontinuities along the axes of the skeleton.

Intra-frame shape interpolation is a method of defining smoothly shaded objects in 2D, offered initially by computer slidemaking packages like Dicomedia, and now commonly available in interactive illustration programs like Adobe Illustrator [6]. It seems particularly attractive for the user interface to a 2D animation system, since it is invoked much like shape interpolation in time. The user specifies two lines or curves of two different colors, and a smoothly interpolated region is swept out. A red apple-shaped silhouette, interpolated to a small round white disk in the upper corner of the apple, makes a smoothly shaded apple with a small round highlight. This is a general and powerful specification for smooth shading, but it is potentially expensive to evaluate (like making little movies in every frame), and is subject to sampling problems (or explosion of the database when making high-resolution pictures). One approach to these rendering problems is to use the two curves to define opposite boundaries of a 2D Coons patch. Then standard tessellation or subdivision methods can be used to render the patch. Coons patches will be discussed as shading primitives in a different context; suffice it to say here that these patches are not unrestricted, and, like brush extrusions, they require the designer to specify some geometry.

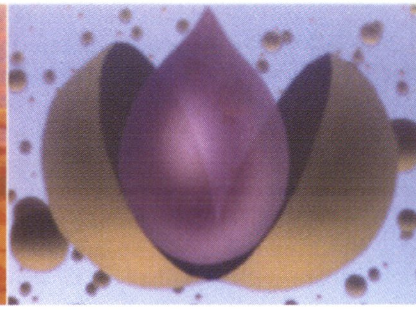
2D Shading at New York Institute of Technology

In the summer of 1977 the author was assigned the task of developing interesting shading effects to enhance 2D animation generated at the New York Institute of Technology's Computer Graphics Lab. The animation was processed by Garland Stern's SoftCel system [7], which permitted computer-aided inking, painting, and composition of penciled artwork on conventional peg-registered animation sheets. Many of the shading effects developed over that summer involved cycling colors, neon glows, and animated textures matted through specified colors of the original painted animation. Color plate 1 represents an initial experiment in producing a "special effect" specifically for character animation. The kneeling Puss 'n Boots character is displayed twice: on the left, as conventionally rendered, and on the right, with gradations of shade and color produced by an "automatic airbrushing" algorithm. In this example, the shading is created by low-pass filtering a matte based on the outline drawing of the character; the matte is used to blend the original cel with a copy of the cel which is painted in "shaded" versions of the original colors. The shading may vary between completely different colors, as it does in the plume on the cat's hat, or between colors which are the same in the "shaded" and "unshaded" cel, in which case no gradation in color appears. No shading appears in the whites of the eyes, the light grey face, or in smaller details of the costume such as the sword and belt buckle. One reason for avoiding shading in the smaller regions is that, with a blur kernel of fixed size, small regions will be obliterated by the filter and appear strictly in the "shaded" color. This difficulty can be avoided, as we shall see, by using variable filtering. Nevertheless, being able to specify on a region-by-region basis whether or not there will be a change in shading induced by the process provides valuable artistic control.

Color plates 2 and 3 represent refined examples of 2D shading developed at NYIT. Color plate 2 ("Lasso," ©1984 NYIT/CGL-Feigenbaum Productions; client: Monsanto; December, 1984) was shaded by animation effects specialist Norma Jean Sundy, from an animation by Francis Glebas using Ed Catmull's TWEEN 2D animation system. When shown this animation, and told that it is strictly two-dimensional, computer graphics engineers typically react with disbelief! One contributing subliminal cue suggesting 3D graphics may be the polygonal outline of the droplet's enclosing sphere. Although the outline of the sphere is animated as a circle, it is rather coarsely sampled to permit hundreds of spheres to be instanced and previewed in real time. The shading along the limb of the droplet, and the central highlight of the droplet, are produced with separate filtering passes. Color plate 3 is from an experimental short, "Pot O' Luck," by Cam MacMillan and Alex Hawley, with effects support by Norma Jean Sundy (©1985 NYIT). This is a good example of 2D shading applied to character animation. The soft drop shadow on the wall, as well as the shading on the leprechaun and his derby, is "automatic airbrushing," computed from the cels of the original unshaded artwork plus some filtering and matting instructions tailored to the desired effect.

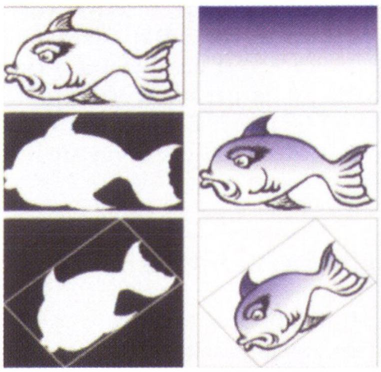


Color Plate 1. "Automatic airbrushing": cat, left, with shading algorithm, right.

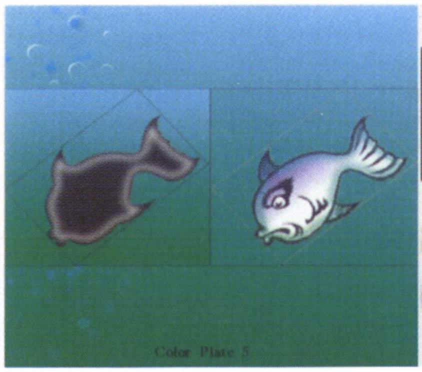


Color Plate 2. "Automatic airbrushing": the droplet is shaded in two passes.

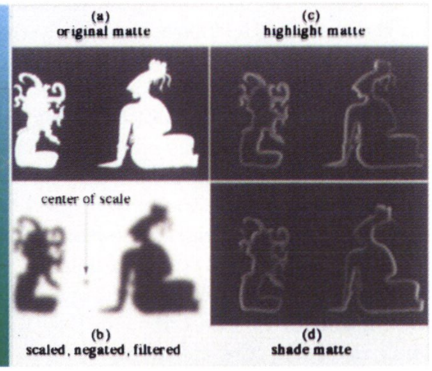
Color Plate 3. Drop shadow on wall, highlights and shading on leprechaun.



Color Plate 4. Graduated color, defined on the figure's bounding box



A phase function of the filtered matte applies "color bleed" from a screen aligned bounding box.

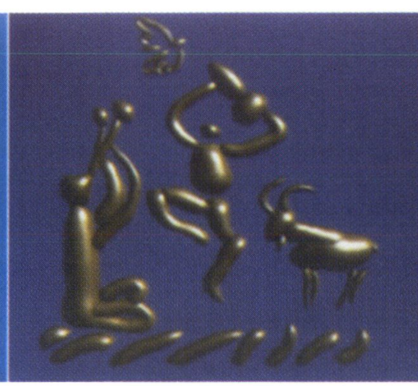


Color Plate 6. Shading and highlight mattes for "local" light source.

Color Plate 7a. Shaded background, flat characters.

Color Plate 7b. Shading and highlights, lightly applied.

Color Plate 8. Picasso's "Rite of Spring": automatic segmentation into superquadrics



Color Plate 9. Manual segmentation into "symmetry seeking" generalized cylinders.

Color Plate 10. Automatic inflation by masked pyramidal convolutions

MAGI and Disney

In 1983, Walt Disney Productions and MAGI (Mathematical Applications Group, Inc., Elmsford, NY) undertook a test combining three-dimensional computer animated backgrounds with conventionally-animated foreground characters. To match the camera moves and perspective on the modeled set, the animation artists were supplied 3D animation of the backgrounds with simple geometric stand-ins for the characters, which moved about the scene as the characters were supposed to but lacked detail or articulation. Using the "stand-ins" as guides, the participating Disney animators produced some lively and convincing three-dimensional animated movements for the characters. To harmonize these characters with the shaded backgrounds, smooth shading was applied to the cel animation. In the Disney/MAGI test, additional artwork elements defining regions of highlight and shadow were outlined. The highlight/shadow regions were filled with solid color on the computer, and digital filtering techniques were used to blend the highlight and shadow colors smoothly into the final image. This form of "automatic airbrushing" motivated the development of the filtering techniques described in [8]; the animation itself is described in [9][10]. (Copyright problems involving the characters in the test, "Where the Wild Things Are," unfortunately preclude the inclusion of an example image in this paper.)

An important observation concerning this pioneer "automatic airbrushing" is that the filtering techniques were required because the original artwork was on paper, and was scanned into the computer. An interactive system which permitted original drawings (or final paintings) to be entered directly would have permitted the shading and highlight regions to be specified in vector form, as airbrush strokes. Brush extrusion [11] (which may be considered a different form of "sweep") more efficiently computes the requisite soft shading, since the prestored filter kernels are applied only along the trace of the stroke, rather than to every pixel in the image.

Simple 2D Shading

The drawing in color plate 4 has an associated matte, so we can color the interior with a gradation of shading, here a simple colormap indirection on a ramp of pixel values running from top to bottom of the drawing's bounding box. This gradation moves, rotates and scales with the drawing. Color plate 5 show a rotated instance of the drawing, with its new bounding box in screen space traced around it. Colors are defined at the corners of the screen-space bounding box, which defines a second bilinear color surface; since these colors remain aligned with the screen, we have used them to represent colors in the background, and essay a little "2D radiosity" by defining a color bleed from the background to the drawing. The most straightforward way of doing this is simply to blur the drawing's matte, and interpolate the character's color ramp over the screen-aligned gradation according to this blurred matte; then, the outline drawing is multiplied by the color to produce the final image. To visualize the composite more clearly, and to simplify the subsequent discussion of refraction mapping, a "color bleed matte" is illustrated on the left of color plate 5. This matte is derived from the original matte for the drawing, after blurring and indirection through a colormap which gives us a smooth "phase function" of the boundary. This matte has been used to apply the colors of the screen-aligned color ramp to the fish on the right.

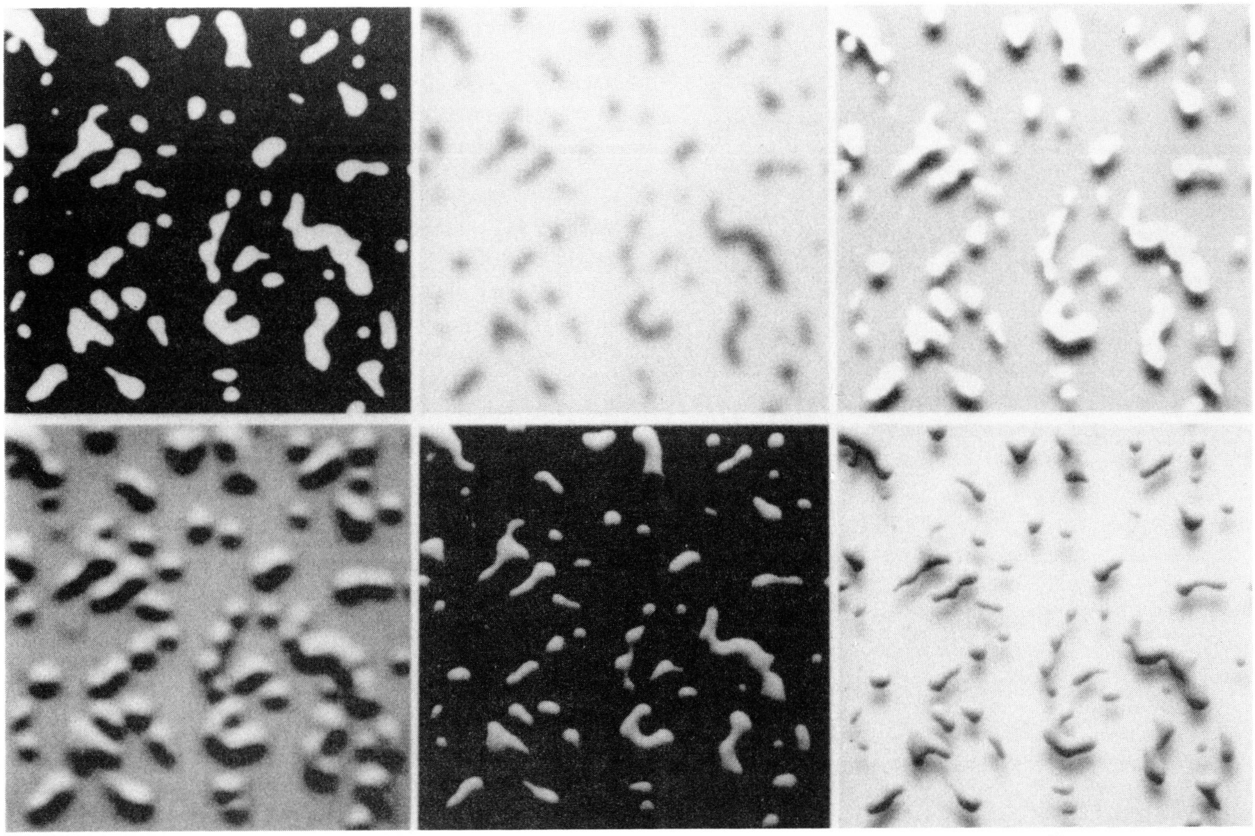
Representing the shading on curved surfaces as simple linear gradations is visually quite effective; contour cues dominate shading cues in vision [12]. This permits us to use shading to indicate that an object is a curved surface, without fidelity to its particular shape. When accompanied by the high frequency detail of the outlines, simple shading functions effectively communicate a stylized light and space.

Shading with Filters

The blurred matte of the previous example is the basic shading primitive of the two dimensional "automatic airbrushing" discussed. Color plate 6 exemplifies the process. In this case, the illumination of a "local" light source is simulated by scaling the blurred matte up from the desired center of illumination, negating it, and multiplying by the original matte to create a "highlight" matte. A "shading" matte is created similarly, to modify regions facing away from the light. (This is not the same procedure that was used to create the shading in the images from NYIT, as the reader can verify by inspection). Both the "highlight" and "shading" that are matted over the original artwork in this process can be spatially-varying gradations of color. Color plate 7 features an animation-in-progress with characters designed by David Em (©1991 Apple Computer, Inc.). The original renditions are shown on the left; 7b shows simply white and black applied through the highlight and shading mattes respectively.

Pyramidal Airbrushing

Smooth shading of two-dimensional images was advanced considerably by crossover work from the image processing community, when pyramidal filter-banks were first applied to the problem in 1985 [13]. Figure 1, reprinted from "Pyramid-based Computer Graphics," illustrates the process. The original 2D shapes in (a) are offset, low-pass filtered, and inverted in intensity, to create the soft "drop shadows" in (b). This operation alone is quite useful, and the graphicist who suspects these effects to be decorative but useless would be advised to study the improvement in interface performance when "drop shadows" are applied to the display of windows



(a) original matte; random two dimensional shapes

(b) offset, negated, low pass filtered shapes

(c) shapes in (a), superimposed on the "drop shadows" of (b)

(d) multigrid "gradient shading" on a Gaussian pyramid

(e) shaded shapes of (d), multiplied by original matte (a)

(f) shaded matted shapes, superimposed on drop shadows

Figure 1. Pyramid-based two-dimensional shading.

[14]. The original shapes are superimposed on the shadows in (c). Gradient filtering of a low-pass filtered image of (a) is illustrated in (d) (operators which combine the low-pass and gradient-filter functions at arbitrary angles are described in [15]); (a) and (d) are multiplied together to produce smooth shading on sharp-edged blobs in (e). Image (f) is (e), matted by (a) over (b), which yields shaded blobs casting shadows on a plane.

The key advantage of the pyramidal filtering is, that the shading can be adapted to the local shape of the silhouettes. The same low-pass filter kernel, applied globally, would shade large regions satisfactorily, but flatten small ones. The Burt-Adelson pyramid permits us to apply filtering at a range of scales, and preserve shading on both fat blobs and skinny ones. This style of "automatic airbrushing" is probably preferable to the application of fixed kernels for most applications.

Refraction Mapping

The resemblance of the Burt-Adelson blobs to droplets suggests that we might want to apply 2D shading techniques to suggest refraction and reflection as well as illumination by a light source. Here we follow the example of [16], which outlined a method of "refraction mapping" (Kay tracing?) appropriate to situations where we have a translucent 3D surface in front of a background image. In such a case, we cannot hope to compute true refraction, and must settle for a symbolic refraction, in which the background is distorted by the translucent surface.

How can such a distortion be computed without depth or normals for our foreground surface? A reasonable convention that is consistent with the appearance of many translucent objects is to increase distortion at the edges. In hollow glass vessels, the material is much thicker at the edge, and lines of sight strike the surface at grazing angles. In the interior of the object, lines of sight are more nearly normal to the surfaces. Figure 2a shows a reflectance-mapped drinking glass modeled from boolean combinations of quadric surfaces. Figure 2b shows a background image of a dishwasher full of dishes "refracted" through the glass, with the image of the glass faintly dissolved over the distorted background. (Refraction mapping test images by the author, ©1985 CGL, Inc.) This image was computed in about four minutes on a VAX 11-780, using Z-buffer techniques. The missing

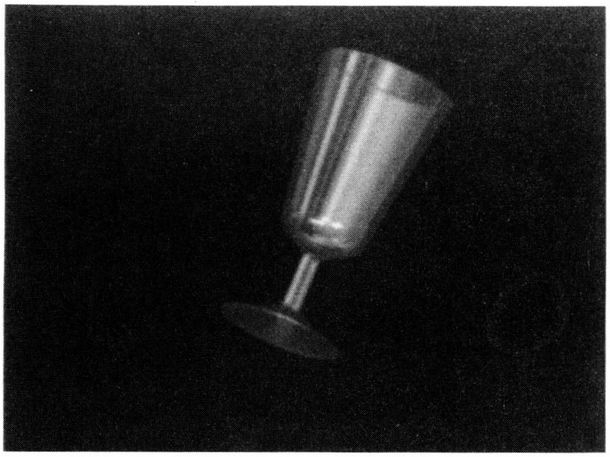


Figure 2a. A drinking glass, modeled from CSG quadrics.

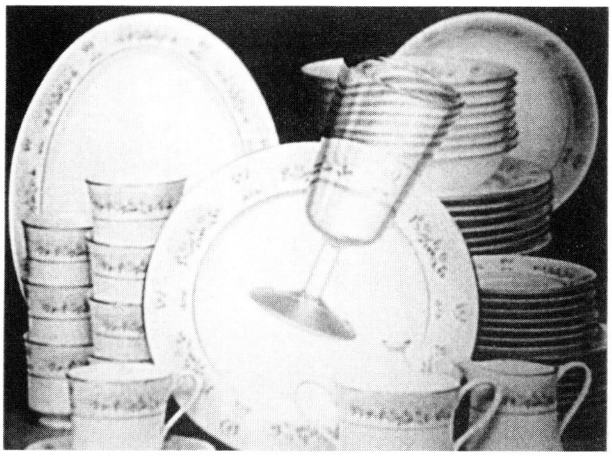


Figure 2b. Refraction mapping based on a phase function of the limb of the glass.

element, an image no longer available, is the "refraction" map which controlled the distortion. It was simply an opaque version of the same glass, illuminated from the eye point with a single light source, shaded as a diffuse surface. This picture was used to create a matte like that in color plate 5. The image was composed with a colormap which ramped rapidly up to white and down again to black in the first 32 pixel values. The result was a smooth but strongly "edge lit" image of the glass. This image was used to modulate an X-Y offset in the background image. Offset vectors for pixel values of 0 and 255 in the refraction map image were specified; intermediate pixel values exerted interpolated displacements. After the refraction mapping was performed, the glass was interpolated slightly into the picture to complete the effect.

Inflated Silhouettes

The attempt to create interesting ways of shading arbitrary shapes from their outlines and silhouettes has its counterpart in the machine vision community. Here the effort is to derive three-dimensional shapes from outlines and silhouettes. Clearly, if 3D shape can be inferred from a drawing, we can use the methods of 3D graphics to shade and composite our animation. (Needless to say, such a problem is "AI complete," and the prudent researcher takes along a lunch. But a partial solution may be very useful.)

The degree to which people are able to interpret silhouettes as three-dimensional objects is striking, and may in many cases be based on recognition. Even unfamiliar objects are generally perceived as three-dimensional; the likelihood that they are cast by flat cutouts is usually small. Figure 3, Picasso's "Rite of Spring," (©SPADEM, Paris/VAGA, New York 1981) was offered as an example by David Marr [17], and has since been the subject of various algorithms to infer 3D shape. Figure 4 is a 3D surface computed by "shape from shading" methods [18]. The superquadric segments of color plate 8 were automatically assigned, after many 1-bit correlations of the image with a family of superquadric silhouettes [19]. Color plate 9 is perhaps the most satisfying rendition, but the number of segments, and starting values for their lengths and orientations, were entered manually; "symmetry-seeking" generalized cylinders [20] relaxed toward the outlines of the image segments from those initial conditions.

A cascade of filters has been applied to the negative of the "Rite of Spring" image in figure 5, much in the spirit of pyramidal airbrushing. Gaussian kernels of radius 32, 16, 8, 4, 2, and 1, were applied to individual connected regions of the "Rite of Spring," each region being filtered separately, as if surrounded by an infinite plane of blackness. After each blurring pass is complete, the blurred image is multiplied by the original (negative) image, so as to eliminate all of the surface outside the silhouettes. After the first pass, the image resembles a hard extrusion with some soft undulations in shape, much in the character of figure 4. As the kernels become smaller and smaller, they pull the edge of the white extrusions more steeply toward black, rounding off the edges of the surface, forcing the silhouette extrusion toward tangebcy to the line of sight at the (enforced) planar limb. Figure 6 is figure 5 viewed as a relief, photographed from a realtime analog display [21]. Finally, color plate 10 shows the filtered silhouette as a shaded 3D surface. The smoothness of the forms, and planarity of their outline, may be thought of as conditions evoked by generalizing objects and viewpoint.

This process can easily be viewed as multigrid smoothing with constraints, the action of the convolutions being subject to the zeroes enforced by the matte. It is thus straightforward to loacally constrain the outlines of the silhouetted regions to an Z values whatever, a useful tool

Figure 3. "Rite of Spring," by Pablo Picasso.

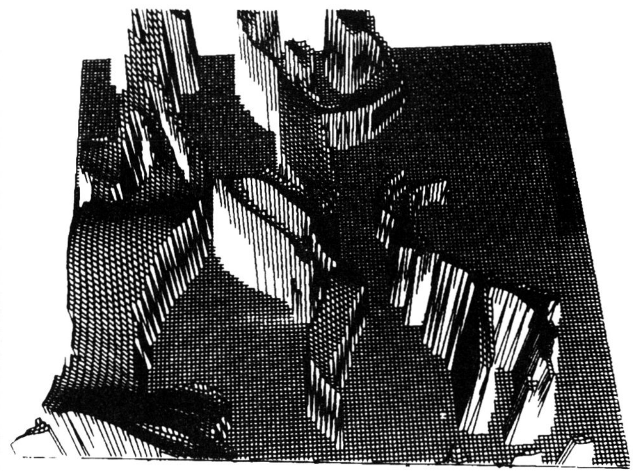


Figure 4. A "shape from shading" inflation of figure 3.

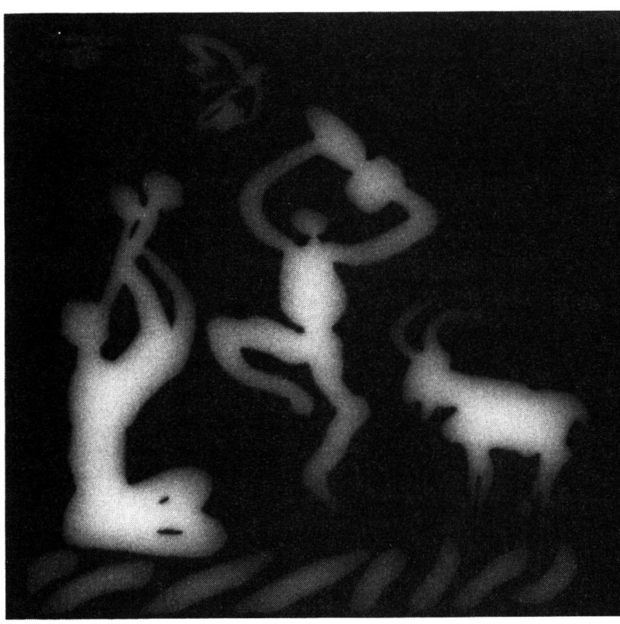


Figure 5. Masked convolution with a pyramid of Gaussian kernels.

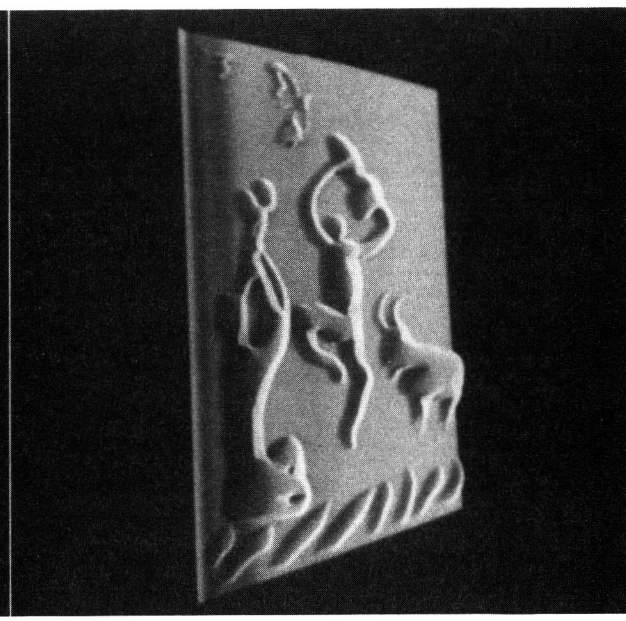


Figure 6. Relief display of figure 5.

for modeling as well as a promising means of adapting our "inflated silhouettes" to confabulate the backfacing surfaces in a segmented range image.

At this point, our essay has come full circle. Expedients for creating shading where insufficient information exists to compute it, have now been used to estimate the missing information. Our way is now clear to use the normals of the estimated surface to "bump map" our 2D objects, applying reflectance mapping or other standard display techniques which require surface normals, or to use the complete 3D estimate as a displacement map for a surface mesh.

Further Research

Countless computational methods of enriching the appearance or enhancing the readability of illustrations remain to be discovered and exploited. Illustration is the symbolic use of visual cues to convey information in pictures; as we increasingly come to understand human vision, we can increase the effectiveness with which these cues are employed.

The effects briefly outlined here can be applied in many combinations which have never been tried. The cross-section lines of relief displays could be used to shade animation in the style of engravings. Shape-from-shading algorithms could be applied to generate a different range of objects from "airbrushed" images [22]. Gradations of hue based on symmetrical filtering of internal lines can be complemented by directional shading based on the silhouette. Brush extrusions for specific highlights and shadings, specified by the animator, could be keyframed over 2D shading algorithms.

The relatively satisfying construction from the silhouette of a not-terribly-energetic three-dimensional surface, using filtering techniques, suggests several further lines of inquiry:

(1) The constructed surface could be a contributor to a minimization based on trading off "shape" and "pigmentation" from shading, serving as an explicit estimate of local orientation from the silhouette;

(2) A pyramid or filter bank of oriented or steerable kernels [15] could be used to generate a segmentation such as that of color plate 8.

(3) Modeling tools based on the "inflation" of silhouettes could be explored as a complement to ordinary sweeps and extrusions in computer-aided design.

Acknowledgments

I would like to thank Pete Litwinowicz for many insights, an animation system to go with them, and assistance with color plate 10; Libby Patterson, for many hours of discussion on pyramidal surfaces; Michael Kass, for a critical ear, a tubular "Rite of Spring," and some key references; to Robin Myers, for photographic support; and to Laurence Arcadias, for constant animation and occasional translation.

Picture Credits

Specific thanks are owed Dr. Alexander Schure, founder of the New York Institute of Technology Computer Graphics Lab., and Louis Schure, its director, for kindly providing figure 2 and color plates 1-3, and permission to feature them in this paper.

Thanks to Ted Adelson, for providing figure 1 and permission to reprint it; I hope the text makes clear the importance of "Pyramid-based Computer Graphics" [13] in advancing the ideas outlined here.

Finally, my thanks to Alexander Pentland, for graciously supplying the photograph of an unpublished experiment (figure 4), as well as color plate 8, and the inspiration of his work.

References

[1] Hochberg, Julian, "The Representation of Things and People," in Gombrich, E.H., Hochberg, Julian, and Black, Max, *Art Perception and Reality*, pp. 74-75, Johns Hopkins Press, London and Baltimore, 1970.

[2] Saenz, Mike, and Bates, William, *Crash*, pp. 68-72, Epic Comics, Inc., New York 1988.

[3] Bourdin, J.J., and Braquelaire, J.P., "Color Shading in 2D Synthesis," *Proceedings of Eurographics '90*, Elsevier Science Publishers B.V., North-Holland, 1990.

[4] Pavlidis, Theo, *Algorithms for Graphics and Image Processing*, Chapter 9, "Thinning Algorithms," pp. 195-214, Computer Science Press, Maryland, 1982.

[5] Harris, Jerry, McGregor, Keith, Wolcott, Jim, and Samborn-Kaliczak, Anne, Pixel Paint Professional User's Manual, pp. 130-136, SuperMac Technology, California 1989.

[6] Adobe Illustrator 88 User Guide, pp. 138-144, Adobe Systems, Inc., California, 1988.

[7] Stern, Garland, "SoftCel - An Application of Raster Scan Graphics to Conventional Cel Animation," Computer Graphics 13, #2, *Proceedings of SIGGRAPH '79*, pp. 284-288, Chicago, Illinois, Aug. 8-10, 1979.

[8] Heckbert, Paul S., "Filtering by Repeated Integration," Computer Graphics 20, #4, *Proceedings of SIGGRAPH '86*, pp. 315-321, Dallas, Texas, August 18-22, 1986.

[9] Perlin, Ken, "State of the Art in Image Synthesis - '85 Course Notes," *SIGGRAPH '85 tutorial: "State of the Art in Image Synthesis,"* San Francisco, California, July 22, 1985.

[10] "Where the Wild Things Are," videotape, ©1983 Walt Disney Productions / Mathematical Applications Group, Inc., New York, 1983.

[11] Whitted, Turner, "Anti-Aliased Line Drawing Using Brush Extrusion," Computer Graphics 17, #3, *Proceedings of SIGGRAPH '83*, pp. 151-156, Detroit, Michigan, July 25-29, 1983.

[12] Ramachandran, Vilayanur S., "Perceiving Shape from Shading," *Scientific American 259*, #2, pp.76-83, August, 1988.

[13] Ogden, J.M., Adelson, E.H., Bergen, J.R., and Burt, P.J., "Pyramid-Based Computer Graphics," *RCA Engineer 30*, #5, pp. 4-15, Princeton, New Jersey, Sept.-Oct. 1985.

[14] Cowan, William, and Loop, Sandra, "Perceiving Window Geometry: An Experimental Study," *Proceedings of Graphics Interface'91.*

[15] Freeman, William T., and Adelson, Edward H., "The Design and Use of Steerable Filters," M.I.T. Media Laboratory - Vision Science Technical Report #118, Cambridge, Massachusetts, June 27, 1989.

[16] Kay, Douglas, Transparency, Refraction and Ray Tracing for Computer Synthesized Images, M.S. thesis, Cornell University, Dept. of Architecture, pp. 30-41, Ithaca, New York, January 1979.

[17] Marr, David, Vision, pp. 216-233, W.H. Freeman and Company, San Francisco, 1982.

[18] Pentland, Alexander, "A Neural Mechanism for Computing Shape from Shading," M.I.T. Media Lab. Vision Science Technical Report 116, Cambridge, Massachusetts, January 1989.

[19] Pentland, Alexander, "Automatic Extraction of Deformable Part Models," International Journal of Computer Vision, 4, pp.107-126, Kluwer Academic Publishers, The Netherlands, 1990.

[20] Kass, Michael, and Witkin, Andrew, "Constraints on Deformable Models: Recovering 3D Shape and Nonrigid Motion," *Artificial Intelligence 36*, pp. 91-123, Elsevier Scientific Publishers B.V., North-Holland, 1988.

[21] "3-D Manipulator Model 67114," (user's manual), Optical Electronics, Inc., P.O. box 11140, Tucson, Arizona 85734, (602) 889-8811, ca. 1988.

[22] Pentland, Alexander, and Kuo, Jeff, "The Artist at the Interface," M.I.T. Media Laboratory - Vision Science Technical Report #114, Cambridge, Massachusetts, January, 1989.

[23] Freeman, William T., Adelson, Edward H., and Heeger, David J., "Motion Without Movement," to appear in *Proceedings of SIGGRAPH '91.*