Visualizing the Fourth Dimension Using Geometry and Light

Andrew J. Hanson and Pheng A. Heng
Department of Computer Science
Indiana University
Bloomington, IN 47405

Abstract

We explore techniques for visualizing mathematical objects in four-dimensional space that exploit four-dimensional lighting effects. We analyze the geometry of image production, stereography, and shadows in 4D. We examine alternatives for smooth and specular shaded rendering of curves, surfaces, and solids in 4D, and propose a new approach that systematically converts curves or surfaces into uniquely renderable solids in four-dimensional space by attaching spheres or circles to each point. Analogs of 3D shading methods are used to produce volume renderings that distinguish objects whose 3D projections from 4D are identical. Analyzing the procedures needed to justify and evaluate a system such as ours for teaching humans to "see" in four dimensions leads us to propose a generally applicable four-step visualization paradigm.

1 Introduction

Four-dimensional mathematical objects [18, 21, 20] have long been of interest to computer graphics [6, 2, 3, 4, 8, 1, 19, 5, 11, 12] because computer graphics can enable us to take a simulated "look" into the fourth dimension.

Our goal here is to explore the interaction of geometry and light in four-dimensional worlds, and to show how the concepts of stereography, shadows, shading, and specularity can be usefully extended from the three-dimensional world of our everyday experience into four dimensions to help us visualize unfamiliar four-dimensional objects.[1]

Why Use Lighting in 4D? Many approaches can be used [5] to represent four-dimensional mathematical objects using computer graphics, including gray-scale range images, color-coded range images, and various types of projections and slices. Here we choose to concentrate on the question of how *shaded depictions* very close to images of familiar everyday objects can be used effectively to convey 4D structure. Among our motivations is the fact that even when both range (absolute depth) and shading (implying relative surface orientation) are available in 3D, shading information dominates the interpretation: when we see a photograph on a billboard, we describe the 3D object in the photograph, rarely perceiving the photograph as meaningless paint smears on a flat billboard.

By forming consistent analogs of the familiar ways that shadows, shading, and specularity let us perceive 3D shape in shaded 2D images (e.g., photographs), we explore methods that represent 4D shape in shaded 3D images (i.e., volume intensity arrays). Volume rendering techniques (see, e.g., Kaufman [17], Fuchs, et al. [10]), together with visualization tools such as stereography and motion parallax, can be used to see these 3D objects using 2D computer graphics images. We study two principal techniques: methods based on displaying multiple viewpoint information such as shadows, and methods motivated by 3D surface shading models. Our most interesting result is a new 4D shading technique that disambiguates the 4D rendering procedure for curves and surfaces by transforming them into 3-manifolds.

How Can We Know We See in Four Dimensions? At the conclusion of the paper, we ponder the question of verifying the utility of our proposed methods. A paradigm for validating visualization approaches such as ours is suggested and is summarized separately in the Appendix.

2 Multiple Viewpoint Methods

Stereographic images and monoscopic images with shadows provide rich cues to the viewer concerning the shape of objects in a 3D scene. In this section, we examine analogous phenomena that are potentially useful for the interpretation of 4D scenes.

[1] By "four-dimensional objects" we generally mean manifolds of dimension 1, 2, or 3 (curves, surfaces, or solids) that have coordinate representations in four-dimensional Euclidean space. (An n-manifold is a space that looks like ordinary n-dimensional Euclidean space in the vicinity of every point.)

CH3046-0/91/0000/0321/$01.00 © 1991 IEEE

Projecting to the Next Lower Dimension. Since in 3D, the film image is a 2D rectangle, as are the ideal "walls" upon which shadows are cast, we deduce that in 4D, the "film" must be thought of as a 3D rectangular solid, while shadows are cast on rectangular solid subsets of hyperplanes. A darkened, smoke-like cloud appears in the 3D virtual film volume whenever a ray cast from the 4D light source is obstructed by an object. Objects appear in the film volume as 3D projections from 4D; that is, all points on the 4D object that lie on a ray from the 4D focal point through a 3D point in the film volume are projected to that single 3D point in the film image. Depth-buffering methods must be used to choose the opaque scene point nearest the focal point, or to combine transparent objects. For additional realism, shadows cast by light sources should be computed before rendering.

To project from N dimensions to $(N-1)$-dimensional film or shadow hyperplanes, we let $\vec{X} = (X_1, X_2, \ldots, X_N)$ be the location of either a light source or a camera focal point, and let

$$\hat{n} \cdot \vec{x} = c \tag{1}$$

be the equation of the image (hyper)plane ($\hat{n} \cdot \hat{n} = 1$). Given any known point $\vec{H}$ in the (hyper)plane, we know $c = \hat{n} \cdot \vec{H}$. The image of a scene point $\vec{P} = (P_1, P_2, \ldots, P_N)$ is found by substituting the parametric equation of a line joining the point source to the scene point, $\vec{x}(t) = \vec{X} + t(\vec{P} - \vec{X})$, into Eq. (1), and solving for $t_0 = (c - \hat{n} \cdot \vec{X})/(\hat{n} \cdot (\vec{P} - \vec{X}))$. The image point $\vec{I}$ lying *within* the image (hyper)plane (1) is then:

$$\vec{I} = \vec{x}(t_0) = \frac{\vec{X}(\hat{n} \cdot \vec{P} - c) + \vec{P}(c - \hat{n} \cdot \vec{X})}{(\hat{n} \cdot \vec{P} - \hat{n} \cdot \vec{X})}. \tag{2}$$

Four Dimensional Stereography. In any dimension N, a *pair* of images each consisting of intensity values in an $(N-1)$-dimensional hyperplane is sufficient to determine the N-dimensional position of a known scene point appearing in both images. Assuming that we know the focal centers and film (hyper)planes of both "cameras," the two lines joining the focal centers to the *images* of the same scene point must pass through the *actual* scene point (see, e.g., [15]).

In 4D, projected rays intersect the film volume at an interior point, rather than in a line as one might expect: in 4D, a line can "go directly" to an interior volume pixel in the same way as a 3D line can pass through a single interior pixel in the film plane. Thus the principles of stereography carry over to all dimensions, and two aligned 3D volume images are sufficient to determine the 4D coordinates of a visible point on a 4D object.

The Role of Shadows. The information provided by a single shadow seen alongside the object in a single view is like having some information from a *second camera*. Therefore, even a single shadow is sufficient to carry out some stereo identification, provided we know the location of the light source and of the (hyper)plane upon which the shadow is cast. Multiple shadows, cast by multiple light sources, can provide additional constraints resembling the various views in an engineering drawing (which would have four views in 4D). An example of a two-manifold in 4D with a 4D shadow, projected into 3D, illuminated with 3D lighting, and projected to 2D, is rendered in Figure 3.

3 Intensity Shading Methods

Images of illuminated 3D objects produce a wealth of orientation cues that the human visual system is able to interpret reliably, apparently by imposing constraints on the ambiguous data. In this section, we discuss 4D shading issues.

4D analogs of 3D Gouraud and Phong interpolation methods of course deal with rectangular or tetrahedral volumes, rather than square patches and triangles. Rectangular blocks are rendered scan-plane by scan-plane into a volume image by (1) dividing the block into 5 tetrahedra, and (2) dividing each tetrahedron into as many as 6 subtetrahedra with easy-to-handle horizontal terminating faces or edges. Up to 30 tetrahedra thus replace the 4 horizontal-based triangles used to shade a rectangular patch in 3D. Care must be used when defining the blocks to avoid "volume butterflying" of adjacent nonplanar rectangular faces. Finally, we must depth-buffer voxels lying on the same camera ray using either an opaque or transparent combination algorithm.

3-manifolds in 4D possess a unique normal vector at each point, and can be shaded using exact analogs of 3D methods. If we are *not* rendering a 3-manifold, however, we must deal with the fact that a single 4D light ray does not uniquely probe the orientation of a 1D curve or a 2D surface. The situation is identical to that of a point and a coil of wire in 3D: the point has three undetermined normal directions parameterizable as a 2-sphere, while the wire has one local tangent direction and two normal directions parameterized by a circle. Thus the intrinsic local properties of a point or a wire in 3D, and a wire or a surface in 4D, force us to deal with a normal *sphere* or *plane* instead of a normal *vector*.

3.1 Smooth Surface Shading

In 3D, a typical shading algorithm computes the normal vector $\hat{n}$ at each polygon vertex, and assigns a Lambertian diffuse intensity

$$I_D = I_0 \, \hat{n} \cdot \hat{L} \tag{3}$$

to the vertex, where $\hat{L}$ is the unit vector from the polygon vertex to the light source and the dot product is taken as zero when negative (the face is pointing away from the light). If both sides of a polygon might be visible, the absolute value is used instead (e.g., for one-sided surfaces). Then either the intensity or the normal vector is linearly interpolated to the interior of the polygon to determine the pixel values.

Note that $\hat{n}$ can actually be considered to be proportional to the *cross-product* of the tangent vectors $\vec{P}$, $\vec{Q}$ at a local point (u, v) in the surface, so

$$I_D = \frac{I_0}{H} \text{Det} \begin{vmatrix} L_1 & P_1 & Q_1 \\ L_2 & P_2 & Q_2 \\ L_3 & P_3 & Q_3 \end{vmatrix}, \tag{4}$$

where $\vec{L}$ is the vector to the light source and the normalization $H = \|\vec{L}\| \cdot \|\text{Cofactor } \vec{L}\|$ is chosen to make the maximum value of the determinant be unity.

In four dimensions, the determinant in Eq. (4) has four columns, and so the shading equation for 3-manifolds is obtained either from Eq. (3) with a four-vector normal, or by placing the tangent directions $(\vec{P}, \vec{Q}, \vec{R})$ on the 3-manifold into the columns of the determinant, yielding the form

$$I_D = \frac{I_0}{H} \text{Det} \begin{vmatrix} L_1 & P_1 & Q_1 & R_1 \\ L_2 & P_2 & Q_2 & R_2 \\ L_3 & P_3 & Q_3 & R_3 \\ L_4 & P_4 & Q_4 & R_4 \end{vmatrix}. \tag{5}$$

For 1D curves and 2D surfaces, there are only one or two available tangent vectors instead of the three needed in Eq. (5), so we are missing some information needed to fill in the rest of the determinant. (This is equivalent to our earlier statement that, in 4D, curves have normal spheres and surfaces have normal planes.)

Multiple Light Sources. If we provide additional light sources, we can make a well-defined 4D shading algorithm if we simply replace the missing dimensions of the manifold tangent space by *additional lighting vectors*. For a curve or surface embedded in four dimensions, we simply replace the missing 4-dimensional tangent vectors $\vec{R}$ or $(\vec{Q}, \vec{R})$ by additional light sources $\vec{M}$ or $(\vec{M}, \vec{K})$ to fill up the missing columns of the determinant, e.g.,

$$I_D(\text{surface}) = \frac{I_0}{H} \text{Det} \begin{vmatrix} L_1 & M_1 & P_1 & Q_1 \\ L_2 & M_2 & P_2 & Q_2 \\ L_3 & M_3 & P_3 & Q_3 \\ L_4 & M_4 & P_4 & Q_4 \end{vmatrix}$$

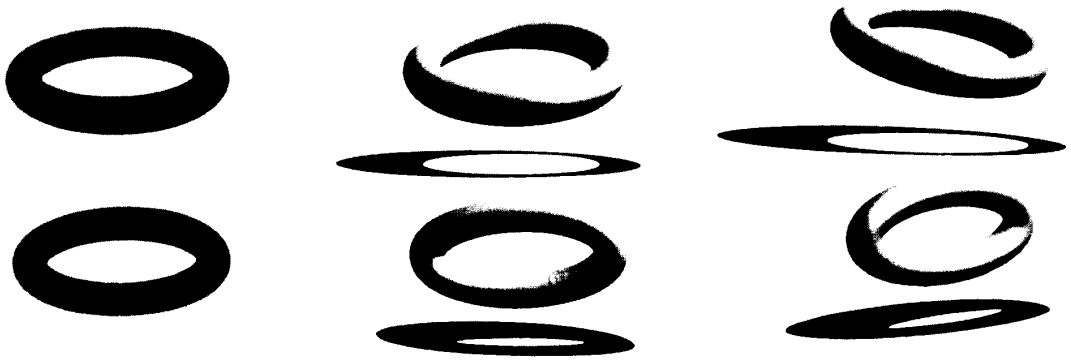
$$I_D(\text{curve}) = \frac{I_0}{H} \text{Det} \begin{vmatrix} L_1 & M_1 & K_1 & P_1 \\ L_2 & M_2 & K_2 & P_2 \\ L_3 & M_3 & K_3 & P_3 \\ L_4 & M_4 & K_4 & P_4 \end{vmatrix}. \tag{6}$$

The meaning of Eq. (6) is that to get maximum intensity, the light sources and the remaining tangent directions in the manifold must form an *orthogonal frame* in four dimensions. A moment's thought about the 3D Lambertian equation will confirm that an exactly analogous orthonormality property holds in the more familiar 3D case. Experiments with this method gave results that were distinctive but difficult to interpret, leading us to the development of the more intuitive swept surface representation to be described next.

Swept Surfaces and Volumes. Let us now return to the case of a wire embedded in 3D. When we think of the wire, we in fact do *not* necessarily imagine it as a pure one-dimensional object that would have to be shaded using the 3D version of the two-light model just introduced; it is much more natural to imagine the abstract one-dimensional wire as the *spine* of a *cylindrical* surface swept out by a circle lying in the perpendicular plane at each point on the wire. The normal to the cylindrical surface swept out in this way is *unique* in 3D. The shading that results, particularly when specularities are included, provides much more natural intuitive clues about the three-dimensional curve that the wire sweeps out in space than the two-light model does. From the theory of shape-from-shading [16], this depiction should carry ample information for reconstruction.

In Figure 1, we show how the visual ambiguities between a tilted 3D circular torus and a vertical 3D elliptical torus can be resolved by using images based on the swept surface with a wire ring spine.

This concept is easily extended to four dimensions as follows: If we consider a curve embedded in 4D and extended by attaching a *sphere* at each point, we find a swept 3-manifold in 4D with uniquely defined normals everywhere. To be precise, each surface patch on the sphere is swept by a line interval to form a 3D solid; the directions normal to the vertices are determined by drawing a line from the center of the sphere (on the curve segment) to a point on the surface of the sphere. Alternatively, the normal to the 3D solid can be computed from a signed "4D cross-product" of three vec-



(a) (b) (c)

Figure 1: (a) (above) A dimensionless wire circle tilted back so that, when drawn with a thick line, its 2D depiction is indistinguishable from (below), a true ellipse lying in the vertical image plane. (b) (above) A dimensionless wire circle converted into 2-manifold by sweeping a circle centered on the wire. When shading and specularity are added, this figure is distinguishable from the true swept ellipse shown in (below). (c) Rotating in the plane perpendicular to the film plane exposes the spatial differences between the two figures.

tors bordering the solid by extending the determinant formula for 3D cross-products of two vectors. The shading procedure simply replaces $\hat{n}$ in Eq. (3) by the appropriate 4D normal and interpolates vertex intensities or normals across each polyhedron.

In Figure 2, we show a volume rendering that distinguishes between a swept circular 3-manifold tilted away in 4D and an elliptical 3-manifold that has an indistinguishable shape in the volume rendering.

Surfaces: For surfaces in 4D, the radial direction of the ring attached at each point can be understood by taking the local tangent vectors $P^\mu(u,v)$ and $Q^\mu(u,v)$ at a point (u,v) of a parameterized surface and using a repeated Gram-Schmidt procedure to find candidates for the coordinate basis (N_1^μ, N_2^μ) of the plane perpendicular to the surface. Once $N_1^\mu(u,v)$ and $N_2^\mu(u,v)$ are found, they are normalized and the four-dimensional normal vector

$$\hat{n}^\mu(\theta,u,v) = \cos\theta \hat{N}_1^\mu(u,v) + \sin\theta \hat{N}_2^\mu(u,v) \qquad (7)$$

is substituted into the 4D version of Eq. (3). The three variables parameterize the corresponding 3-manifold for which $\hat{n}^\mu(\theta,u,v)$ is the current normal direction.

3.2 Specularity

Specular highlights give yet another independent source of information about the geometry in four dimensions. We next construct the analog of a specular Phong shading contribution (see, for example, [7]) to the intensity displayed in the 3D image of an object in 4D. We simply take as many lighting vectors $\hat{L}_i$ as we require, replace $\hat{L}_i$ by the normalized sum $\hat{B}_i$ of the camera direction $\hat{C}$ and the lighting vectors $\hat{L}_i$, then raise the appropriate dot product or determinant to a high power k.

For a 3-manifold swept out by a circle (or a sphere) attached to each sample point of the manifold, we simply add to the shading equation a specular term

$$I_S = I_1 \left|\hat{B}\cdot\hat{n}\right|^k , \qquad (8)$$

where $\hat{n}$ is the appropriate 4D normal. We then carry out volume rendering (for both diffuse and specular components) by interpolating the normals at the polyhedron vertices instead of directly interpolating the intensities.

4 Examples:

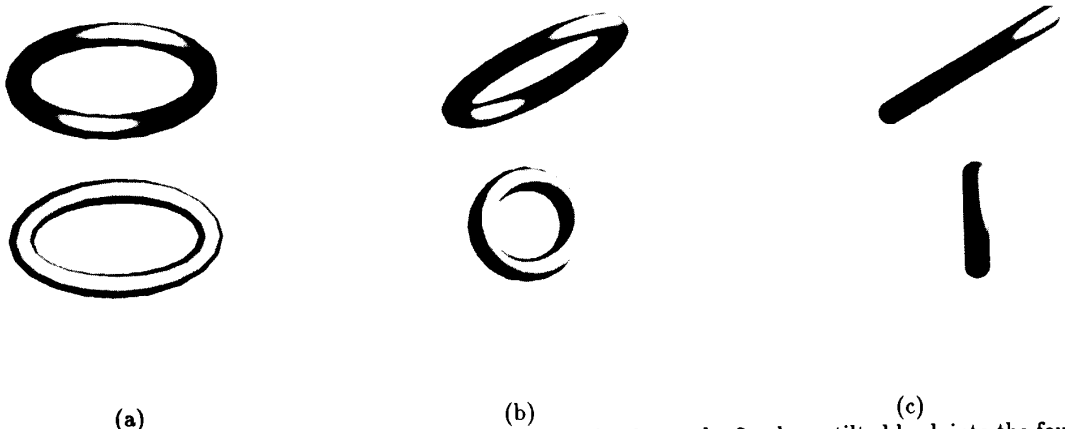
We next illustrate our basic concepts with some classic examples from four-dimensional topology.

Three-Sphere. The three-sphere is given by the implicit equation

$$x^2 + y^2 + z^2 + \left(\frac{w}{c}\right)^2 = R^2$$

or the parametric equations

$$\begin{aligned} x &= R\cos\alpha\sin\beta\sin\theta \\ y &= R\sin\alpha\sin\beta\sin\theta \\ z &= R\cos\beta\sin\theta \\ w &= Rc\cos\theta\ , \end{aligned}$$



(a) (b) (c)

Figure 2: (a) (above) A 3-manifold given by the cross product of a ring and a 2-sphere, tilted back into the fourth dimension w so that its 3D volume rendering is indistinguishable in shape from the elliptical 3-manifold (below) lying in the $w = 0$ hyperplane. Four-dimensional shading and specularity introduce features that distinguish the two 3-manifolds. (b,c) Rotating in the x-w plane reveals the difference between the two manifolds.

where c is a constant we may use to "squash" the sphere into a 4D pancake while leaving its 3D projection a completely symmetric 2-sphere. Figure 4 shows a rendering of the true 3-sphere compared to the corresponding squashed sphere. Substantial differences in the shading can be easily seen.

Steiner Surface/Whitney Crosscap. In an appendix to a chapter of their classic book *Geometry and the Imagination*, Hilbert and Cohn-Vossen [14, 8, 1] give a set of equations for a Whitney crosscap, a non-singular embedding of the projective plane in four dimensions. We represent this surface parametrically by the equations

$$x = \cos^2 u \cos^2 v - \sin^2 u \cos^2 v \quad (9)$$

$$y = \sin u \cos u \cos^2 v \quad (10)$$

$$z = \cos u \sin v \cos v \quad (11)$$

$$w = \sin u \sin v \cos v \ , \quad (12)$$

where $u \in [0, \pi]$, $v \in [0, \pi]$. The tangent vectors are computed locally by taking partial derivatives with respect to u and v and orthonormalizing as required.

The interesting fact about this equation is that when it is rotated in four dimensions, it changes smoothly from a Whitney crosscap to a Steiner Roman surface, depending upon the rotation axes chosen[6, 8]. In Figure 3, we showed how these surfaces can be viewed by casting 4D shadows and projecting both the object and the shadows to 3D before rendering using conventional 3D lighting methods. We see how the use of shadows can greatly clarify the nature of the projection that is taking place from 4D to 3D; we can simultaneously see the two states of the shape and watch them exchange places in the course of an animated 4D rotation.

Figure 5 shows a stereo pair of the shaded volume rendering of the Steiner surface when each point is replaced by a small specular circle in the normal plane and the resulting 3-manifold is illuminated by a single 4D point light source. To distinguish 4D from 3D effects, we render the full 4D surface alongside an artificially modified surface that has the same 3D appearance, but has had its fourth coordinate shrunk to a very small value before attachment of the circles in the normal plane. Finally, Figure 6 shows two stages of the 4D rotation that exposes the hidden differences between the two figures as the Roman surface is transformed into the Whitney crosscap. Once again, we can see that objects with identical shapes in the film volume have quite distinct shading signatures depending on their 4D characteristics; we also see that making sense of these distinctions is nontrivial and requires experience and training.

5 Conclusion

We have proposed a family of techniques for creating intuitively informative shaded images of four-dimensional mathematical objects. We deliberately chose to concentrate on depiction methods that are close to familiar 3D-world shaded images, as opposed to range images and slices that have an artificial appearance.

As we consider the requirements for a robust 4D visualization system, we come to the conclusion that the present work is incomplete: in particular, we have not

proven that the images we produced are *interpretable*, we have not shown that the necessary interpretation techniques can be taught to human users if they were not already known, and we have not verified that the desired intepretation of the images is *cognitively feasible* within the limitiations of the human intellect. A robust approach to visualization should address all these issues, leading us to propose the general visualization paradigm outlined in the Appendix.

Interpreting 4D Images. Among the issues that we are currently investigating is the question of extending machine vision techniques to four dimensions; the shape from shading methods such as that of Ikeuchi and Horn [16], and the photometric stereo technique of Woodham [22] seem particularly appropriate. We have already tested these methods successfully on simple synthetic data resembling the mathematical systems we have examined. These methods appear to be useful for *reinterpreting* 3D volume renderings as images of *mock* four-dimensional objects [13]. Our observation is that, while not all 2D images are derived from renderings of the 3D world, it is often helpful to organize 2D data as though it *were* a shaded image. Thus, given some basic assumptions, it is possible to reconstruct a completely fictitious 4D interpretation of any volume rendering using the same techniques we must use to check the validity of our 4D rendering methods. Depending on the nature of the data, such 4D interpretations might single out otherwise obscure characteristics of the data for further attention. Note that this is entirely different from approaches that treat voxel values as a fourth coordinate and simply rotate the data in four dimensions.

Verifying Learnability. Another open question that we hope to address is the issue of the cognitive nature of 4D perception. We must understand whether the proposed visualization procedures are useful to human observers, and to understand and model the most effective techniques for teaching an understanding of the resulting images. After all, from our experience with the 3D world and 3D lighting, we have learned to apply a few ad hoc constraints that let us deduce very reliably the 3D content of shaded 2D images of a scene, particularly if shadows, stereo, and/or motion parallax are available. Can humans acquire facility with four dimensions with enough practice? Is it possible that, just as we can "see" true 3D depth in a monocular sequence of images from motion parallax, we can learn to use our binocular vision on volume renderings combined with oscillating motion parallax in the fourth dimension to "see" true four-dimensional depth? With more practice, could a single shaded volume rendering in stereo be sufficient for 4D understanding, just as single photographs of a 3D scene are?

Acknowledgments

We gratefully acknowledge the support of the staff and the use of the facilities at CICA, the Indiana University Center for Innovative Computer Applications. We thank T. Banchoff for some useful remarks.

Appendix: A Visualization Paradigm

In this Appendix, we present our viewpoint on the nature of a complete scientific visualization paradigm for problems such as the representation of four-dimensional mathematical objects.

We begin with a statement of what we call the *Visualization Principle*:

> *A useful data depiction must allow the viewer to reconstruct a consistent and relevant model of the original data.*

That is, images produced by a visualization procedure are assumed to be based on real or simulated data consistent with an underlying mathematical model. The purpose of such an image is to allow the user to grasp some fact or property that is not intuitively obvious from looking at a formula, columns of numbers, or a text description. If the viewer cannot produce an accurate mental model for the relevant properties from the image alone, the image is not serving the purposes of the visualization process.

We suggest that the following steps are necessary and sufficient to guarantee that the Visualization Principle is satisfied:

- **Generate Images.** In this stage, the visualization team studies the available data and the classes of information that are relevant for display, and then proposes various data depiction techniques to elucidate the underlying models.
- **Interpret Images.** Next, a convincing argument must be made that the *images alone* contain sufficient information to allow the reconstruction of the desired data models. Specific, computer-implementable model reconstruction algorithms should be suggested and, if feasible, tested in order to discover unexpected ambiguities.
- **Teach Interpretation.** The existence of automatable algorithms to reconstruct the data model from the image does not necessarily mean that humans can perform this reconstruction

without training. It may be necessary, for example, to build interactive tools that help a human acquire an intuitive understanding of the reconstruction procedure. Typically, this might be accomplished by establishing a simulated environment (or perhaps a "virtual world") that incorporates tasks that cannot be solved successfully without mastery of the reconstruction algorithm.

- **Verify Learnability.** Finally, it is not necessarily true that any given reconstruction method *can* be learned by a human, even with extensive training. One must therefore evaluate both human ability to intuitively employ the proposed reconstruction algorithms and the effectiveness of any proposed teaching methods. This completes the cycle: if one can prove that humans can learn the reconstruction, and that the method can be taught effectively, the visualization procedure is guaranteed to meet its goals.

References

[1] F. Apéry, *Models of the Real Projective Plane*, Vieweg, Brahnschweig, 1987.

[2] T.F. Banchoff, "The Hypercube: Projections and Slicing," International Film Bureau, Chicago, Ill.

[3] T.F. Banchoff, "The Hypersphere: Foliation and Projections," T. Banchoff Productions, Providence, R.I.

[4] T.F. Banchoff, "Visualizing Two-Dimensional Phenomena in Four-Dimensional Space: A Computer Graphics Approach," in *Statistical Image Processing and Computer Graphics*, E. Wegman and D. Priest, eds., pp. 187–202, Marcel Dekker, Inc., New York, 1986.

[5] T.F. Banchoff, *Beyond the Third Dimension: Geometry, Computer Graphics, and Higher Dimensions*, Scientific American Library, New York, 1990.

[6] T.F. Banchoff and C. Strauss, "Complex Function Graphs, Dupin Cyclides, Gauss Map, and Veronese Surface," Computer Geometry Films, Brown University, Providence, 1977.

[7] J.D. Foley, A. van Dam, S.K. Feiner, and J.F. Hughes, *Computer Graphics: Principles and Practice*, Second Edition, Addison Wesley, 1990.

[8] G.K. Francis, *Topological Picturebook*, Springer-Verlag, New York, 1987.

[9] D. Cox, G.K. Francis, and R. Idaszak, "The Etruscan Venus," videotape, National Center for Supercomputing Applications, University of Illinois, 1987.

[10] H. Fuchs, M. Levoy, and S.M. Pizer, "Interactive Visualization of 3D Medical Data," *IEEE Computer*, Vol. 22, No. 8, pp. 46–51, Aug. 1989.

[11] A.J. Hanson (Author and Director), P.A. Heng and B.C. Kaplan (Animators), videotape entitled "Visualizing Fermat's Last Theorem," presented in the Animation Screening Room at *SIGGRAPH 90*, Dallas, Texas, August 8–10, 1990.

[12] A.J. Hanson, P.A. Heng, and B.C. Kaplan, "Techniques for Visualizing Fermat's Last Theorem: A Case Study," in Proceedings of *Visualization 90*, San Francisco, October 23–26, 1990, pp. 97–106, IEEE Computer Society Press, 1990.

[13] A.J. Hanson and P.A. Heng, "4D Shape from Volume Renderings," in preparation.

[14] D. Hilbert and S. Cohn-Vossen, *Geometry and the Imagination*, Chelsea, New York, 1952.

[15] B.K.P. Horn, *Robot Vision*, MIT Press, 1986.

[16] K. Ikeuchi and B.K.P. Horn, "Numerical Shape from Shading and Occluding Boundaries," *Artificial Intelligence*, Vol. 17, pp. 141-184, 1981.

[17] Arie Kaufman, "Efficient Algorithms for 3D Scan-Conversion of Parametric Curves, Surfaces, and Volumes," *Computer Graphics*, Vol. 21, no. 4, pp. 171–179, SIGGRAPH '87, July, 1987.

[18] Ivars Peterson, *The Mathematical Tourist*, W.H. Freeman, New York, 1988.

[19] I. Peterson, "A Different Dimension," *Science News*, Vol. 135, pp. 328–330, 1989.

[20] R.v.B. Rucker, *Geometry, Relativity, and the Fourth Dimension*, Dover, 1977.

[21] J.R. Weeks, *The Shape of Space*, Marcel Dekker, New York, 1985.

[22] R.J. Woodham, "Photometric Stereo: A Reflectance Map Technique for Determining Surface Orientation from Intensity," Proc. 22nd International Symp. Society of Photo-Optical Instrumentation Engineers, pp. 136–143, San Diego, CA, August 1978.

Figure 3: The projective plane and its shadow projected from 4D to 3D. Appearances of object and shadow interchange when rotated in 4D (on right).

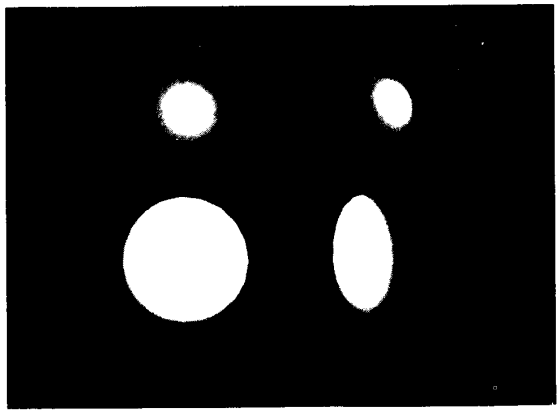
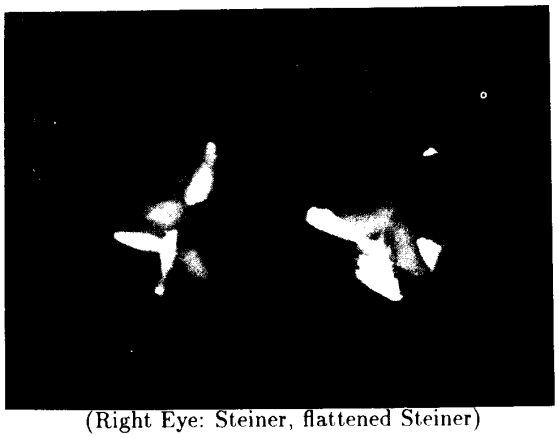
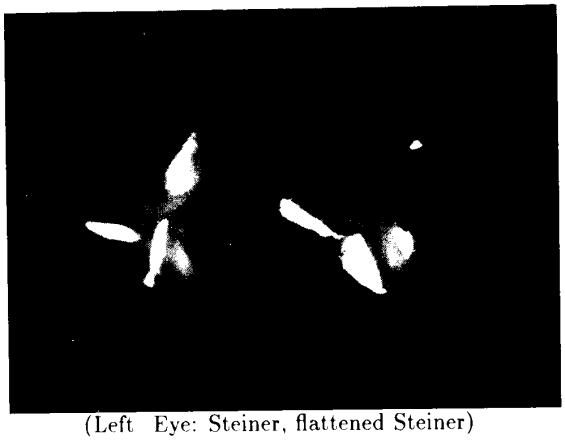


Figure 4: (left) 4D volume rendering of the 3-sphere (top) and a squashed 3-sphere (bottom). (right) Same scene rotated 60 degrees in 4D.

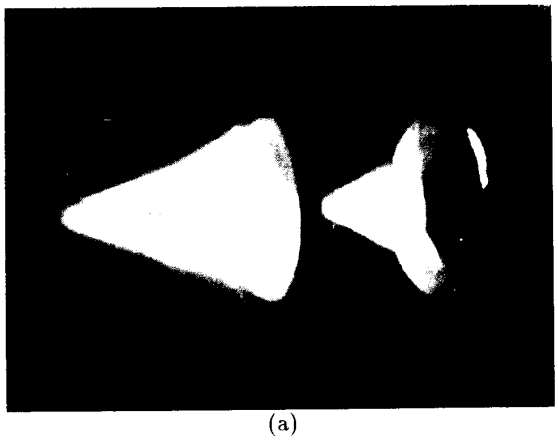


(Right Eye: Steiner, flattened Steiner)

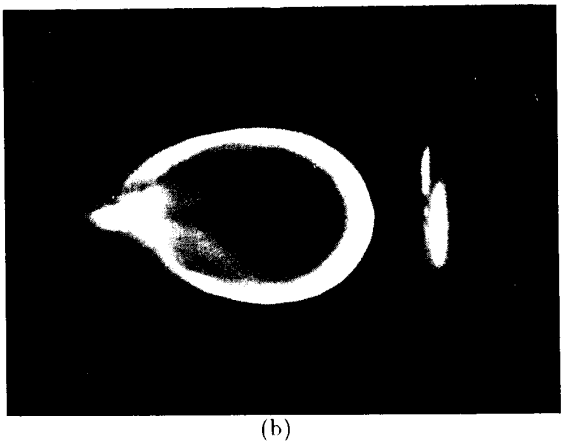


(Left Eye: Steiner, flattened Steiner)

Figure 5: Cross-eyed stereo pair of volume rendering of Eq. (12) using swept-circle method. Left object is a projection to the Steiner Roman surface, right object is *flat* in 4D, but has the same 3D projection in this view.



(a)



(b)

Figure 6: (a) Rotating Figure 5 to expose the hidden fourth dimension gives a shape between the Whitney crosscap and the Steiner Roman surface. (b) Rotating by 90 degrees gives the Whitney crosscap and a flat disk.

(See color plates, page 432.)

Visualizing the Fourth Dimension Using Geometry and Light, A.J. Hanson and P.A. Heng, pp. 321–328.

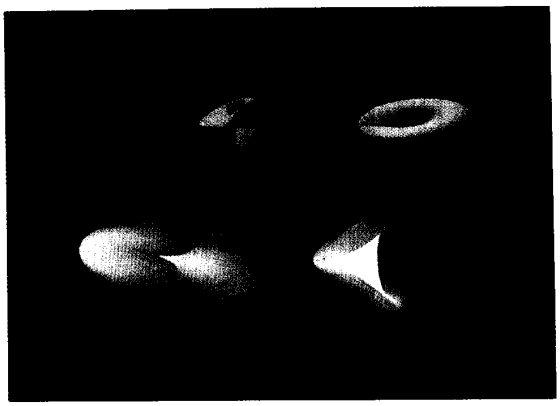


Figure 3: The projective plane and its shadow projected from 4D to 3D. Appearances of object and shadow interchange when rotated in 4D (or right).

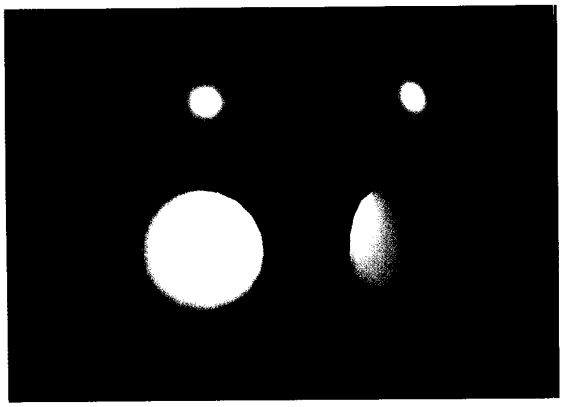
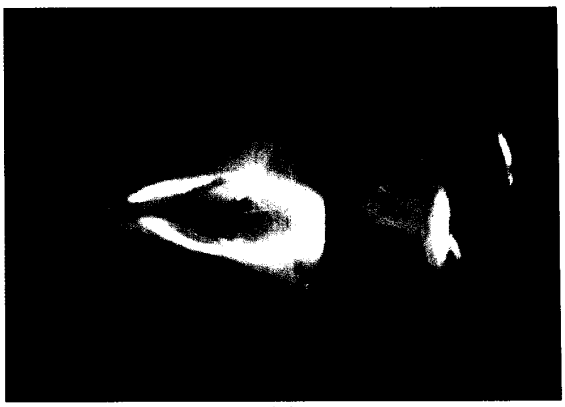


Figure 4: Left: 4D volume rendering of the 3-sphere (top) and a squashed 3-sphere (bottom). Right: Same scene rotated 60 degrees in 4D.

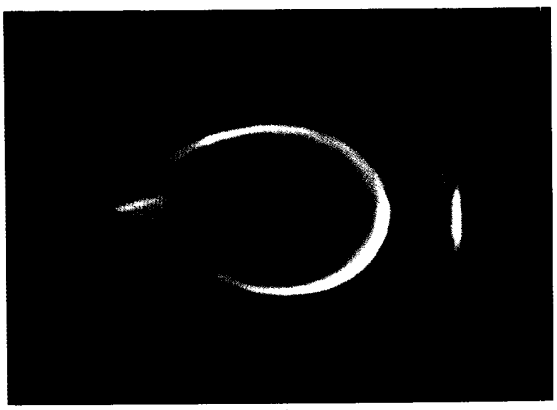
(Right eye: Steiner, flattened Steiner)

(Left eye: Steiner, flattened Steiner)

Figure 5: Cross-eyed stereo pair of volume rendering of Eq. (12) using swept-circle method. Left object is a projection to the Steiner Roman surface, right object is *flat* in 4D, but has the same 3D projection in this view.



(a)



(b)

Figure 6: (a) Rotating Figure 5 to expose the hidden fourth dimension gives a shape between the Whitney crosscap and the Steiner Roman surface. (b) Rotating by 90 degrees gives the Whitney crosscap and a flat disk.