

FROM WIRE-FRAMES TO FURRY ANIMALS

Gavin S. P. Miller

Alias Research Inc.

Toronto, Canada, M5C 1P1.

Abstract

This paper describes an approximate method for rapidly generating reflectance functions for highly anisotropic materials. This is applied to the rendering of images of burnished metal. A method is then developed for rendering wire-frame drawings which have a wire-like appearance. Such wire-like elements are "grown" or created on the surface of parametric patches and the results resemble furry animals. This procedural modelling technique is extended to include the generation of scales.

Keywords: Wire-frames, anisotropic shading, procedural modelling, animals, fur, scales.

1. Introduction

The small scale features of a surface, known as its microstructure, determine the way it reflects light. For example, the surface of a gramophone record will catch the light in a way which depends on the groove direction as well as the orientation of the plane of the disc. Similarly hair and fur reflect light in a way which depends on the orientation of the hairs relative to the light sources and the viewer. In general, the light reflected from a surface will be a function of the surface normal and the tangent vectors. In Kajiyama 1985 and Cabral et al 1987, the reflectance function for a surface is expressed in terms of the surface tangents and normal, and it is derived from the surface topography. The elevation of the surface from a plane is used to characterise the geometry and the average reflected light for a given direction is then computed. While these methods are effective for rough anisotropic surfaces, they are also very computationally expensive.

Section 2 of this paper presents a special case of the above more general methods, which assumes that the surface is comprised of a large number of locally parallel cylinders. This proves to be much faster to compute than previous methods, and may be applied both to the rendering of burnished metal and hair. Section 3 uses the shading model to make realistic images of wire-frame structures which eliminate the ambiguities found in ordinary wire-frame drawings. Section 4 discusses the application of these techniques to the depiction of naturalistic objects which contain a large amount of hair-like detail. An algorithm is presented which allows the generation of geometric primitives on the surface of a patch which mimic the appearance of fur and scales.

2 Shading Models

2.1 Isotropic Reflectance Functions

An isotropic material has a reflectance function which is independent of the surface tangent vectors. For a specified view direction and lighting environment, the reflected intensity is only a function of the surface normal. Since reflectance calculations may be time-consuming, Kajiyama found speed improvements through the use of a look-up table instead. Such a table may be defined at leisure and then referenced rapidly during rendering.

Williams 1983 described how a sphere rendered in orthogonal projection samples the shading model over the entire range of surface normal orientations. In the resultant image, the x and y coordinates of the pixels correspond to the components of the surface normal. A "reflectance sphere" is just such a sphere rendered using the shading model of interest. When subsequently rendering a polygonal model, it is possible to use the interpolated surface normal to extract the reflected intensity from the reflectance sphere. Image 1 shows a multiple light-source Phong-shaded sphere. The highlights are clearly visible and the sphere has a "plastic" appearance. The jug in Image 2 was rendered using this shading model.

Rather than visualising the shape, a designer may be more interested in the overall aesthetics of a finished product. A polished silver jug would look very different from this plastic-like image. It is possible to depict polished metal objects in a number of ways including ray-tracing. (See Whitted 1980.) Unfortunately, such a process can be very expensive. A cheaper method is to create a reflectance sphere of the environment. Image 3 shows a silver ball reflecting a mountain scene. It was generated using the scan-line technique described in Miller 1987. The silver jug in Image 4 was rendered using the ball as the reflectance sphere. The result is aesthetically satisfying but it is not particularly informative about the shape of the jug. Also this approach cannot handle the reflections from a metal surface which has been burnished for instance using sand paper. To handle such effects it is necessary to develop anisotropic reflectance functions.

2.2 Anisotropic Reflectance Functions.

An anisotropic shading model is one which, in addition to surface normal data, makes use of the tangent vectors. A number of materials are anisotropic. Hair, for instance, has a directional character. Burnished metal has grooves along a

particular direction. If the surface is rotated about its normal then the grooves will reflect the light differently. Anisotropic reflectance functions are a recent innovation in computer graphics. Kajiya 1985 explains how a general surface topography may be combined with arbitrary optical properties to give a reflected light intensity. A large number of look-up tables are used to store the intensity for every conceivable tangent and normal vector combination.

The author suggests a simpler, faster approach. Rather than treating arbitrary surface types, a model is developed which is solely a function of one of the tangent vectors. The intensity data may be stored in a simple table called a "pseudo-reflectance map". (Kajiya's method allowed a complete continuum of models from isotropic to highly anisotropic. The method presented here is restricted to highly directional materials.) For an ordinary reflectance map the surface normal is used to reference an intensity for rendering. For a pseudo-reflectance map a single tangent vector is used to derive the intensity. Thus an anisotropic surface whose reflectance function depends solely on a tangent vector may be treated in the same way as an isotropic surface whose reflectance function is solely dependent on the surface normal.

When creating a pseudo-reflectance map, the surface is assumed to consist of fine cylindrical elements rather like hair. The cylinder axis direction corresponds to the tangent vector of interest. The intensity reflected by the surface is equivalent to the average intensity reflected across the width of each cylinder. Unlike a sphere, a cylinder does not contain every possible orientation of the surface normal. Instead, it has normals which are restricted to lie on half a great circle in the reflectance sphere. This is illustrated in Figure 1. Each point on the great circle corresponds to a surface normal orientation. Each surface normal is fed through the ordinary, isotropic shading model to compute an intensity. The samples are taken at equally spaced intervals along the diameter of the great circle, since this corresponds to equal areas in screen space. The mean intensity for these samples is then placed in the pseudo-reflectance map pixel which corresponds to the cylinder axis orientation. In this way, the whole pseudo-reflectance map is built up. When a surface is subsequently rendered, the eye space surface tangent vector is used to index into the pseudo-reflectance map directly.

When a surface is rendered in this way, the resultant image resembles photographs of burnished metal. Image 5 shows a sphere which has been burnished longitudinally. Highlights have been spread out in a direction which is perpendicular to the tangent direction. The image resembles a Christmas tree decoration made using a series of fine threads laid from pole to pole. Image 6 shows the sphere with grooves running latitudinally. It looks as though it was turned on a lathe. The grooves were made explicitly visible using a parametric texture map. The groove intensities were produced using an uncorrelated series of random numbers. Both the diffuse and specular components of the shading model were modulated by the groove texture so as to mimic the effect of self shadowing caused by uneven groove depth. Image 7 shows an inlet manifold shaded using this technique. It closely resembles brushed stainless steel. Image 8 shows the jug done using a burnished effect. The result is both aesthetically pleasing and informative about the shape. (It is worth noting that because the highlights are spread out by the anisotropic model, they also become relatively dim. A highlight which is visible in the original Phong model can almost disappear for the anisotropic case. The isotropic shading models used in this paper

were specified to have very bright highlights.)

A problem arises if the pseudo-reflectance map is stored as a function of the x,y components of the axis vector. Having a z component near to zero occurs often. Unfortunately, this region of the illumination model is not adequately sampled in the x, y formulation. This does not matter too much for ordinary reflectance functions since having the z component near to zero corresponds to outlines on the surface. For tangent vector shading however, having the z component near to zero corresponds to face-on surfaces. The solution chosen was to have two maps, one a function of x,z the other a function of y,z . Depending on which of the x or y components is closer to zero, the corresponding map is used. This solves the sampling problems.

It is worth considering the approximations in the method. In order to allow the storage of the anisotropic model in a pseudo-reflectance map, it was necessary to assume that the sight vector for the shading model was parallel to the eye-space z -axis. This is only approximately true for a perspective view and it renders the model unsuitable for ray-tracing. It also requires the table to be recomputed if the camera rotates or moves relative to the lights in the scene. A similar restriction applies to the methods for reflectance spheres in Williams 1983.

3. Visualising Wire-frame Models

One reason for choosing the cylindrical approximation is that the resultant shading model can also be applied to the rendering of wire-frames. Wire-frame models are shape descriptions which are made up purely of line segments. Line segments are simple to generate and transform, and they may be rendered rapidly using cheap equipment. Modern graphics systems may display and transform several thousand line segments in real time.

The generality of wire-frames is also their weakness. Because they do not necessarily imply anything about the underlying surface it is not always possible to do hidden line removal automatically. This, in turn, leads to new problems. It is often difficult to work out which part of a surface is nearer to the viewer. Indeed there is a depth ambiguity which is unresolvable purely from the visual information in a conventional wire-frame drawing. One way round this is to make lines dimmer as they recede from the observer. This does give a sense of relative depth. However if several objects are present in a scene it is not clear how to choose the appropriate range for dimming the lines. Also, depth-cued line segments are difficult to integrate convincingly with shaded surfaces.

The A-buffer algorithm described in Carpenter 1984 may be extended to cope with line segments. A polygon is created which is the thickness of the line segment and has its minimum width oriented perpendicular to the view direction. Its maximum width joins the end points of the line segment.

Since subpixel areas are properly characterised, line segments can overwrite each other without creating aliasing artifacts. Also, z data may be used to give correct line overwrite priority. However, having an ordered rendering method may be of only limited value if the lines are of the same intensity for the whole wire-frame. Image 9a shows the combination of a wire-frame with a shaded surface. While the surface and line segments have the correct priority, the areas where there are only line segments and no surfaces, are somewhat confusing. To overcome this the author developed the illumination model outlined in Section 2. Image 9b shows the

same scene done using the new shading method. The wire-frame looks as though it is made from real wire. The front segments are distinguishable from the back ones. Also the brightness of the line segments gives some clue as to the axis-vector orientation.

Image 10 is a wire-frame chalice rendered using shaded line segments. A vivid 3-dimensional impression is created and there is no ambiguity as to which regions of the structure are closest to the viewer. Image 11 shows a mould design. One half of the mould is shown with a wire-frame being used to give an impression of the final product shape. The core of the mould is clearly visible through the wire-frame.

4. Surface Texture

4.1 Texture Mapping Methods

Smooth shaded images can be used to depict certain man-made components with a high degree of realism. However, there are plenty of manufactured and natural objects which have large amounts of surface detail. Image 12 shows the application of a simple chequer pattern to the colour of a torus.

While colour texture mapping may be used to depict certain surface markings, it is not appropriate for modelling rough surfaces. Blinn 1978 suggested using table entries to perturb the surface normal rather than the colour. Image 13 shows the effect of colour mapping a rocky texture onto a torus. The shading of the roughness does not match up with the lighting direction. Image 14, on the other hand, uses a perturbation of the surface normal. This is also known as bump mapping. The result is more convincing although the outline remains smooth.

4.2 Organic Surface Textures

A different sort of surface texture arises where geometric elements "grow" on a surface. Leaves and needles on trees, and fur and scales on animals are examples of this sort of texturing. For the purposes of this paper, the distinction between scales and fur is that fur is modelled using line segments, whereas scales are modelled using polygons. Fur is modelled using single straight lines while long hair would be modelled using many line segments per hair.

Line segments can be used to describe a number of naturalistic objects. The tree in Image 22 was defined using conical and cylindrical patches for each of the needles and rendered using a raycaster. The branches and needles were positioned using a growth model developed by the author which allowed the tree to be viewed at different ages. While acceptably realistic, the tree took 2.0 hours per frame on a Celerity 1230 for 1024 by 768 pixels. The A-buffer approach developed in Section 3 can be seen as less accurate but more efficient than ray-casting cylinders.

When modelling fur, hairs are generated based on an equally spaced grid of points in parameter space. For a given pair of (u,v) parameters, the root of the hair H_1 is produced using the surface equation for the patch. The other end of the hair, H_2 , is merely H_1 plus a linear combination of the tangent vectors and the surface normal. The combination factors give the orientation of the hair relative to the surface. This method of generating hair means that the hairs "stick" to the patch surface, so if the patches are animated the hairs will be animated too. If the hair roots are aligned exactly on a parametric grid of points the resultant hairs resemble a brush, which looks very artificial. To achieve a more naturalistic appearance the parametric coordinates for the roots are randomised by an

amount which varies from zero to one grid increment. This "dithering" is similar to placing the hairs randomly and then eliminating any hairs which are too close together. This idea was taken from methods in the literature for antialiasing ray tracing. (See Dippe and Wold 1985.)

Image 15 shows an A-buffered furry torus. The hair intensities were randomised slightly, as were the lengths. The shading model was the one described previously for the cylindrical elements. Image 16 is the same scene shown with seven light sources. About 7000 hairs were used which compares to 1600 polygons for the underlying surface. The hairs were made long so that the fur would look fairly "thick" or dense, i.e. the surface is almost completely obscured. The real test of a fur synthesis algorithm is what it looks like when applied to a biological form. Image 18 shows an animal rendered using the A-buffer technique. The top picture was rendered using just tangent vector shading. The result looks like a not-very-good toy made from synthetic material. However, when the hairs are added the animal, called "Oscar", looks quite convincing. Oscar was modelled by lofting between a series of cross-sections spaced out along a spline curve. This was done for the main body and for each of the limbs. Making a smooth blend between the limbs and the main body was unnecessary since the hair covers up the joins. However, this simple design did lead to collisions between the hairs and the limbs.

In Image 19 the tangential shading model was applied to the grooves in the record on the turntable as well as to the turntable itself. A convincing radial highlight is the result. Image 20 shows a spider rendered using the same technique. 15,456 triangles and 36,737 hairs were used. The spider was made more natural-looking by altering the colour of the hairs near the joints in the legs. An important difference between the spider and Oscar is that the hairs on the spider were all of a fixed length. With Oscar the length of each hair was made proportional to the local u tangent vector magnitude.

While fur is applicable to certain biological forms it does require large numbers of line segments for the surface to be convincing. Scales, on the other hand, are larger and there may be less of them required to cover the underlying surface. The scales for this paper were generated parametrically without dithering. One scale was placed in the bottom left corner of each rectangle in parameter space. A second scale was placed in the centre of the rectangle. Two triangles were used for each scale and the normals at the triangle vertices were deflected outwards to give a rounded appearance. The scale intensities were also varied to prevent the scales from looking too uniform. See Image 17. The same technique was used to generate the scaly dragon in Image 21.

5. Areas for Future Work

The images of furry animals in this paper still look rather unconvincing and synthetic. This is for several reasons:

- The hairs are still too thick. This problem may be resolved by creating higher resolution images with a larger number of finer hairs.
- The hairs are all straight. Approximating hairs by using several polygons along their length may lead to a more realistic appearance, although it is still an open research question as to how each hair should be defined and controlled.
- The fur, as shown, is not very colourful. Using parametric colour maps on the skin patches to define the root colour should make the fur more attractive and realistic.

d. The roots of the hair are currently evenly distributed in parameter space. It would be better to have arbitrary density functions controlling the placement and number of the roots, and the lengths and orientations of the hairs. The orientation of the hairs could even be animated to simulate the hairs being buffeted by the wind, or to reflect the mood of the animal.

6. Conclusions

This paper presented a cheap method for rendering bur-nished metal. The technique was combined with an A-buffer algorithm to allow the convincing rendering of wire-frame objects. It was explained how to generate fur on a collection of surface patches. The resultant images looked like toy creatures covered with synthetic fur. While possibly useful for character animation, the techniques were found to be less than perfectly realistic, and a number of suggestions were given as to how the appearance could be improved.

7. Acknowledgements

This research took place in England as part of a Science and Engineering Research Council CASE Award agreement between Cambridge University Engineering Department and Deltacam Systems Ltd. The images were viewed using an ARGs 7000 Series frame buffer on loan from Sigmex Ltd. Dr. D. P. Sturge supervised the author's Ph.D project and Jon Hunwick designed the shape of Cassandra, Boris and Oscar in collaboration with the author. The mould was designed by Jon Hunwick and Dr. R. F. Field of Cambridge University Department of Anatomy. Image 22 was created using the Alias Research "Scene Description Language" and renderer. Thanks are due to the reviewers who gave helpful suggestions for making the paper more readable, with special thanks to one in particular who pointed out that repeated use of the word "whilst" has caused cancer in laboratory mice.

8 References

Blinn, James F. "Simulation of Wrinkled Surfaces." Computer Graphics, Vol. 12, No. 3, August 1978.

Cabral, B., N. Max, R. Springmeyer, "Bidirectional Reflection Functions from Surface Bump Maps", Computer Graphics, Vol. 21, No. 4, July 1987.

Carpenter, L. "The A-buffer, an Antialiased Hidden Surface Method", Computer Graphics, Vol. 18, No. 3, July 1984.

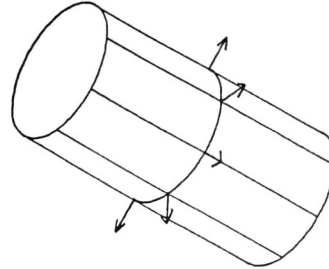
Dippe, Mark A. Z., Erling Henry Wold, "Antialiasing Through Stochastic Sampling." Computer Graphics, Vol. 19, No. 3, July 1985.

Kajiya, J., "Anisotropic Reflection Models", Computer Graphics, Vol. 19, No. 3, July 1985.

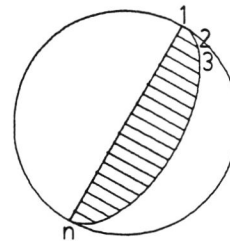
Miller, G. S. P., "Computer Display and Manufacture of 3-D Models", Ph.D Thesis, Cambridge University Engineering Department, Cambridge, England, June 1987.

Whitted T., "An Improved Illumination Model for Shaded Display," Comm. ACM, 23, 6, pp. 343-349, June 1980.

Williams, L., "Pyramidal Parametrics", Computer Graphics, Vol. 17, No. 3, July 1983.

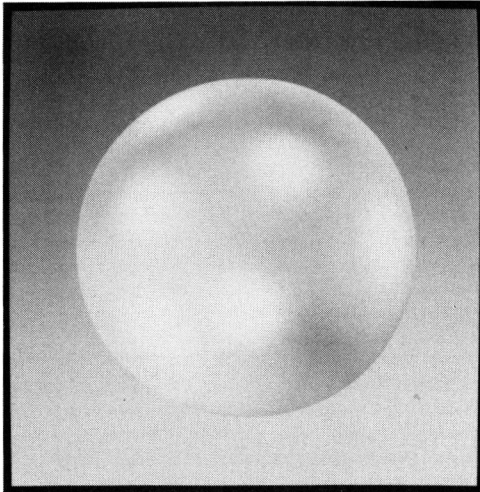


Cylindrical Element

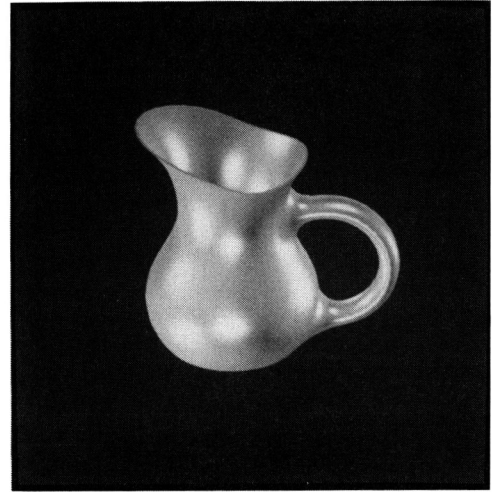


Pseudo-reflectance Sphere

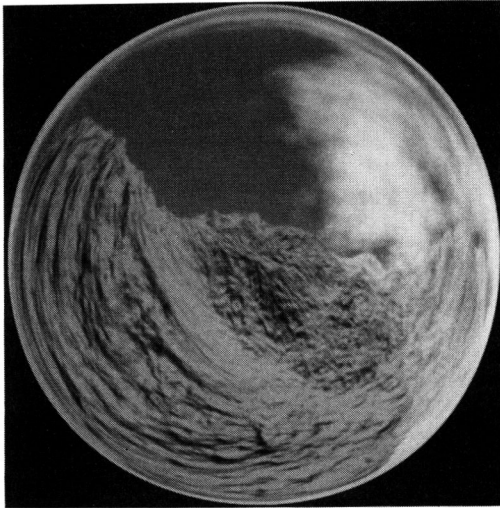
Figure 1 Cylindrical Lighting Model Geometry



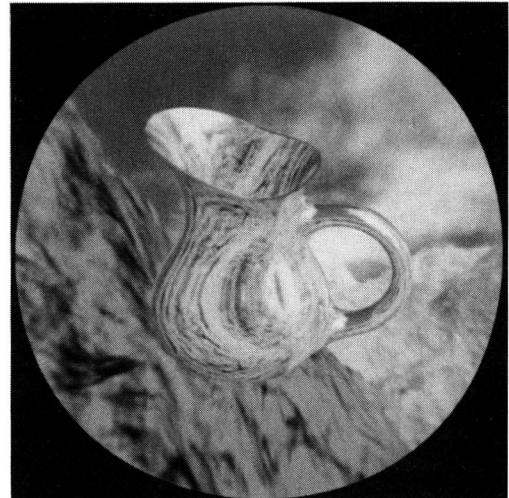
1 Phong Shaded Sphere



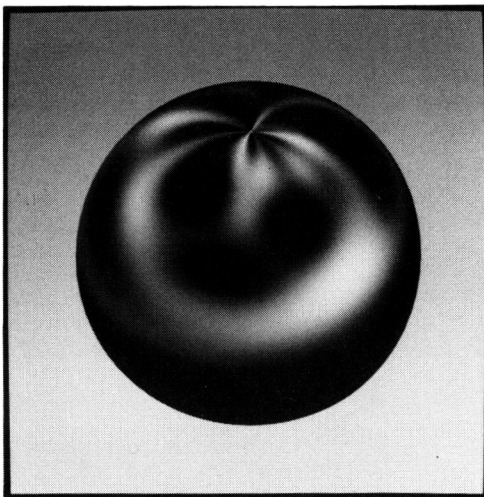
2 Phong Shaded Jug



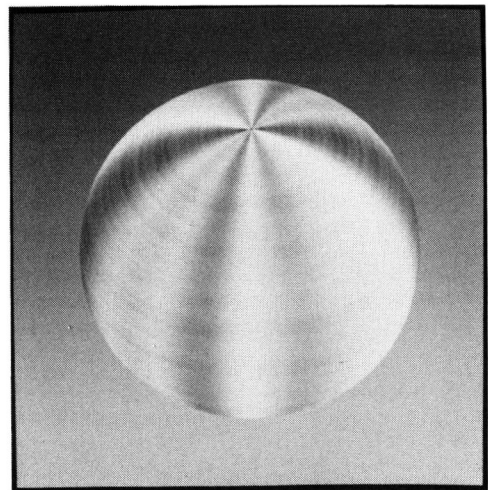
3 Mountain Scene Reflectance



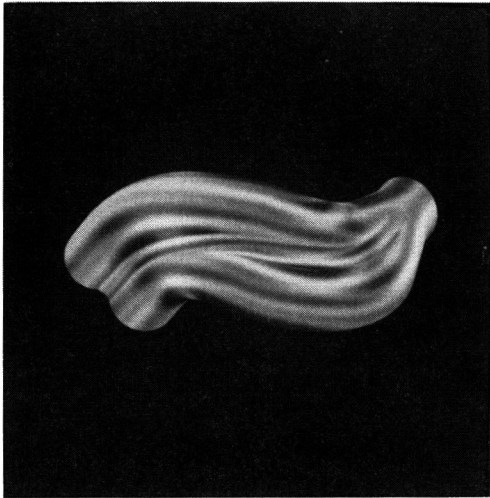
4 Magritte's Milk Jug



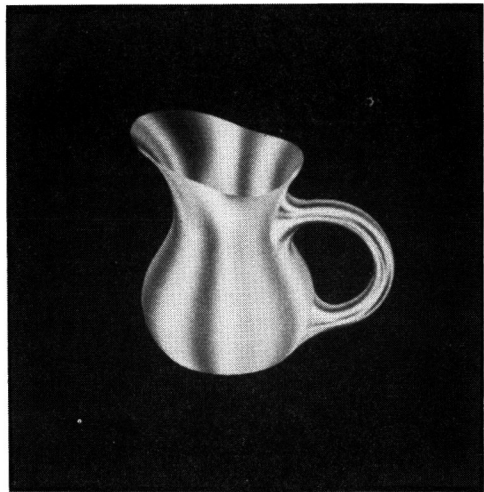
5 Longitudinal Burnish



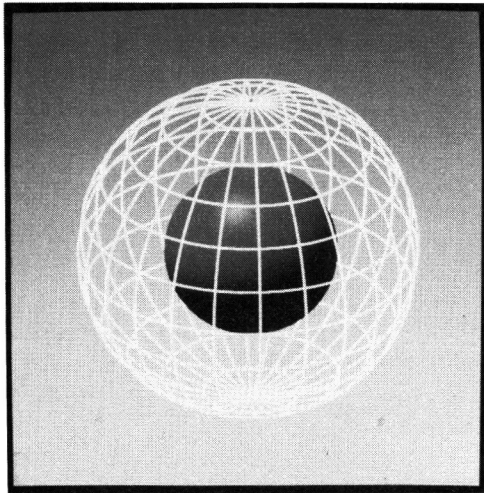
6 Latitudinal Burnish



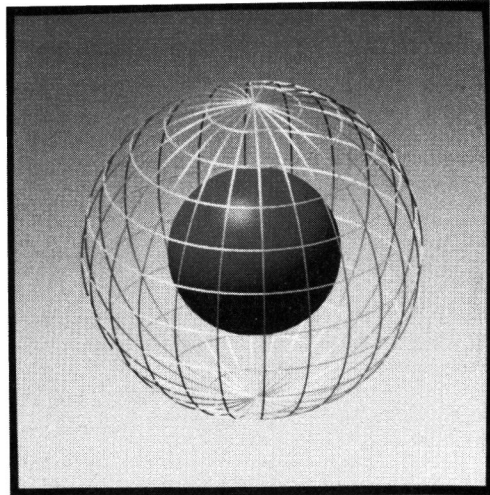
7 Burnished Exhaust Manifold



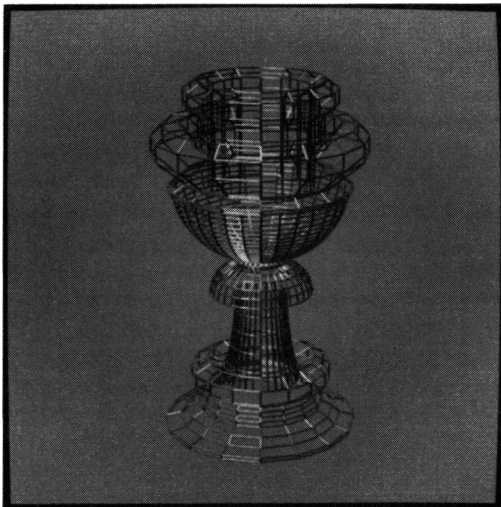
8 Burnished Jug



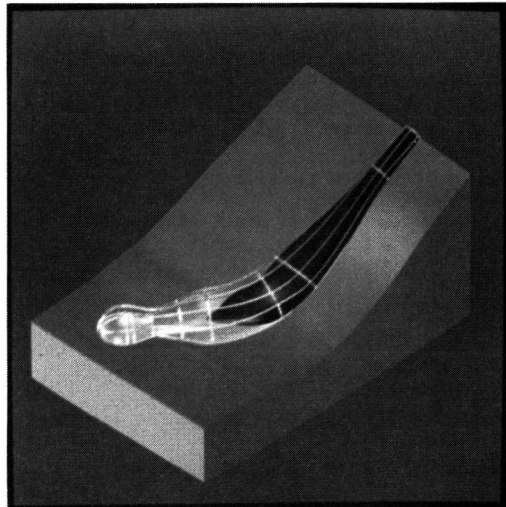
9a Wire-frame Cage



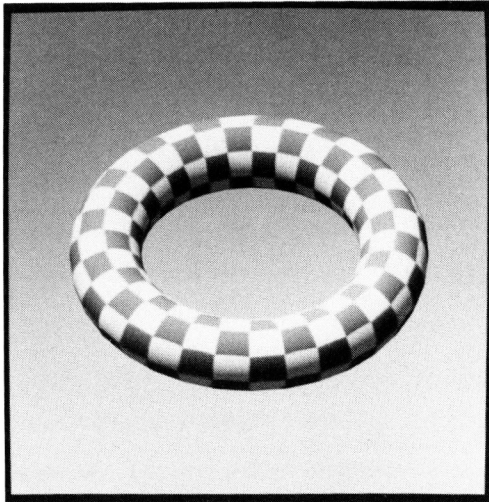
9b Shaded Cage



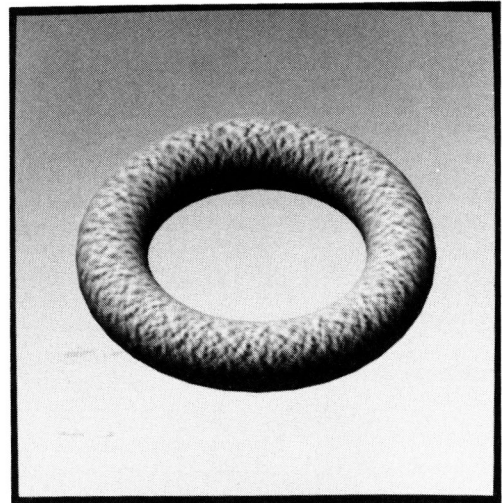
10 Shaded Wire-frame Chalice



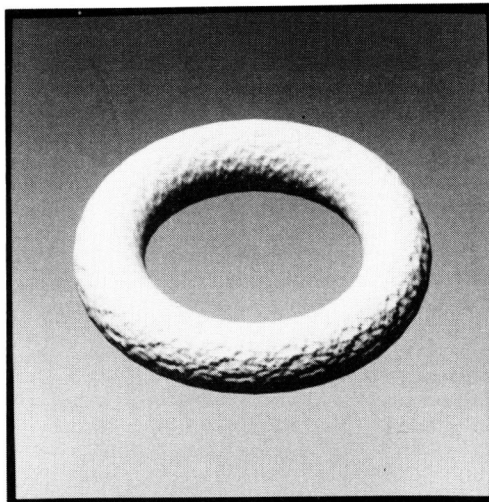
11 Mould Design



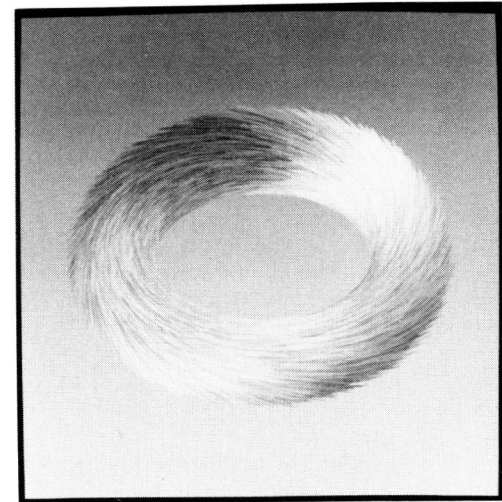
12 Chequered Torus



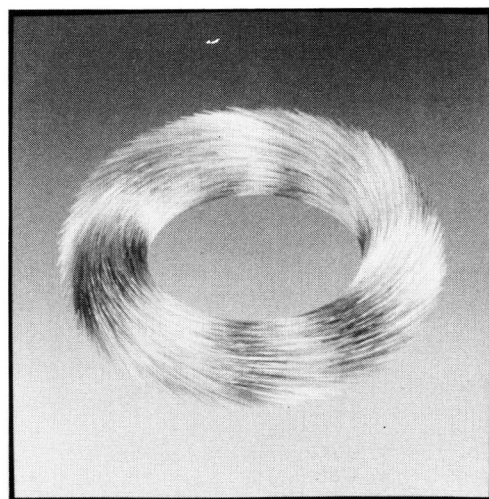
13 Patterned Torus



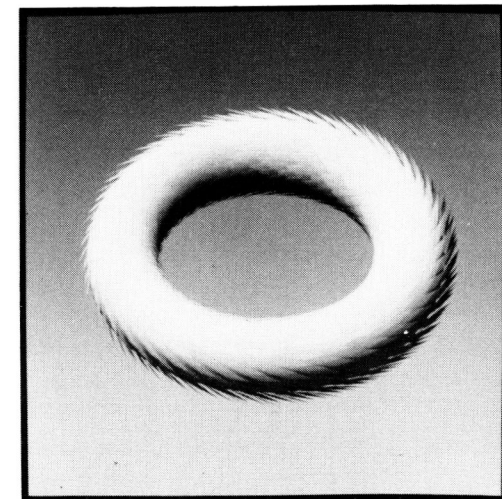
14 Bump Mapped Torus



15 Furry Torus



16 Multiple Light Sources



17 Scaly Torus



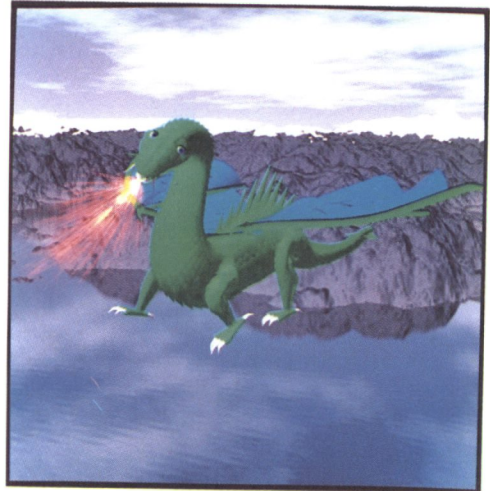
18 Oscar's Coat



19 Room with a View



20 Boris' Winter Coat



21 Cassandra the Dragon



22 Fir Tree with Sky