Surface and Shading Models from Real Images for Computer Graphics

Jiping Lu Jim Little

Technical Report 97-10
August 1997

Laboratory for Computational Intelligence
Department of Computer Science
The University of British Columbia
Vancouver BC Canada V6T 1Z4
e-mail: {jplu, little}@cs.ubc.ca

Abstract

In this technical report we present an object modeling and rendering technique from real images for computer graphics. The technique builds the surface geometric model and extracts the surface shading model from a real image sequence of a rotating object illuminated under a collinear light source (where the illuminant direction of the light source is the same as the viewing direction of the camera). In building the surface geometric model, the surface reflectance function of the object is extracted from the real images and used to recover the surface depth and orientation of the object. In building the surface shading model, the ambient component, the diffuse component and the specular component, are calculated from the surface reflectance function extracted from the real images. Then the obtained surface shading model is used to render the recovered geometric model of the surface in arbitrary viewing and illuminant directions. Experiments have been conducted on both diffuse surface and specular surface. The synthetic images of the recovered object surface rendered with the extracted shading model are compared with the real images of the same objects. The results shows that the technique is feasible and promising.

Index terms: Computer vision, computer graphics, Surface recovery, Surface modeling, Surface reflectance, Shading function, Graphics rendering, Virtual reality.

1 Introduction

Computer vision and computer graphics are two complementary fields of computer science. Computer graphics deals with direct problems, that is, generating images from predefined shading models and geometric models, while computer vision studies inverse problems, that is, extracting reflectance and 3D geometric information from real images. Because of the nature of the two problems, graphics and computer vision have been on two separate tracks for many years. Since both of them are trying to achieve realistic descriptions of the physical world, they share some common aspects and their interaction is inevitable, especially, in reflectance modeling and surface modeling. The reflectance models [19, 13] used in computer vision are physical based and can be used in computer graphics for more realistic rendering. On the other hand, the Phong shading model in computer graphics, for its simplicity, has been used in by Horn [8] in computer vision. Both computer graphics and computer vision have to deal with surface modeling. The notion of a generalized cylinder was first introduced by Binford in computer vision long time ago and has been widely used in computer graphics. Terzopoulos's physics-based models, such as finite element model, [11], adaptive meshes [17] and deformable superquadrics [16], can be directly applied to animation in computer graphics. The oriented particle surface model developed by Szeliski *et al.* [15, 14] is useful in both computer vision and computer graphics.

We attempt to integrate computer vision with computer graphics in both reflectance modeling and surface modeling for more realistic visualization. The surface reflectance function is extracted from an image sequence of a rotating object illuminated under a collinear light source. The extracted reflectance function is used to recover surface shape of the object. A shading function is calculated from the extracted surface reflectance function by an optimization method. Then the calculated shading function is used for rendering the recovered object surface under the general illuminant and viewing conditions. Experiments with objects of diffuse surfaces and specular surfaces show that the technique is feasible and can be extended to more complicated objects.

2 Experimental Conditions

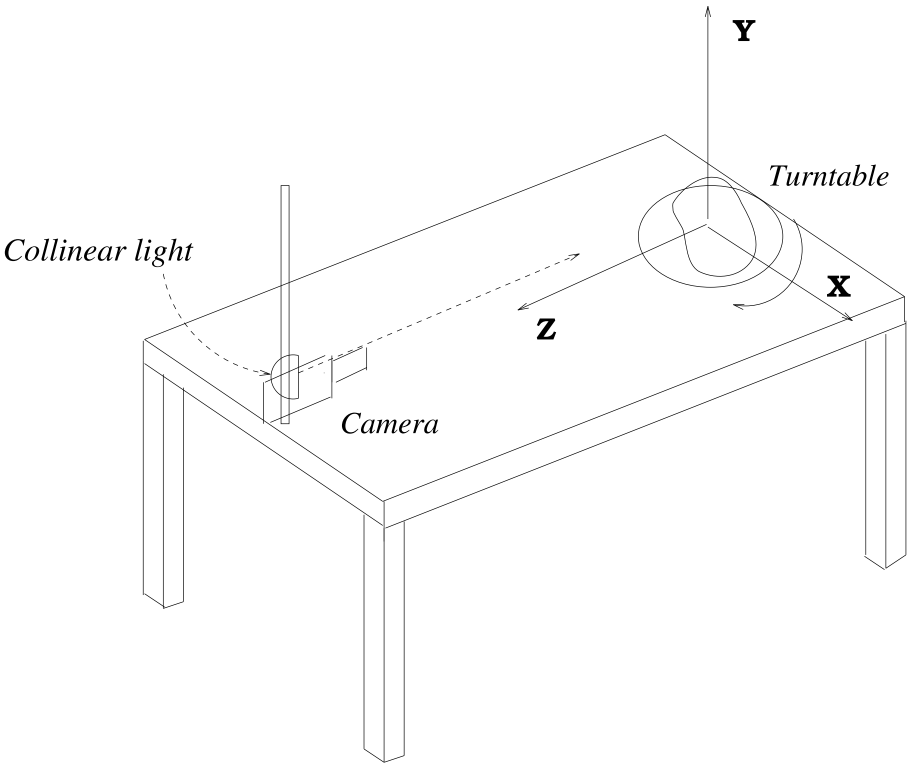


Figure 1: Experimental conditions

An image sequence of an object is taken when the object is rotating in front of a fixed camera. The imaging geometry is shown in Fig. 1. The object is on a turntable whose rotation angle can be controlled. The Y axis coincides with the rotation axis of the turntable. Both the camera viewing direction and the light illuminant direction are aligned with the Z axis. The light source is a distant light source with uniform radiance over time and illuminated area so it is considered a directional point light source. Since the camera is far away from the object, orthographic projection is used as an approximation. A simple calibration method is used to obtain the projection of the rotation axis in the image sequence [10]. Images are taken when the object rotates around the Y axis in the direction from the X axis to the Z axis.

The object surface is assumed to be piecewise continuous and differentiable. The surface orientation is defined as $(-p, -q, 1)$ with $p = \partial z(x,y)/\partial x$ and $q = \partial z(x,y)/\partial y$. When the object rotates, the coordinates and the orientation of a surface point on the object change. Let (x, y, z) be a surface point and $(-p, -q, 1)$ be the surface orientation of this point, After α degree rotation, the 3D location $(x_\alpha, y_\alpha, z_\alpha)$ of this point is $(x_\alpha,\ y_\alpha,\ z_\alpha) = (x\cos\alpha - z\sin\alpha,\ y,\ x\sin\alpha + z\cos\alpha)$ and the surface orientation $(-p_\alpha, -q_\alpha, 1)$ of this point is determined by

$$p_\alpha = \frac{p\cos\alpha + \sin\alpha}{\cos\alpha - p\sin\alpha}, \tag{1}$$

$$q_\alpha = \frac{q}{\cos\alpha - p\sin\alpha}. \tag{2}$$

We also assume the reflectance of the object surface is uniform. In computer graphics literature [5], the image intensity I at a surface point under a directional point light source is determined by the shading function:

$$I = I_a + I_d * (N \cdot L) + I_s * (N \cdot H)^n, \tag{3}$$

where N is the surface normal, L is the illuminant direction vector, and H is the half vector between the viewing direction and the illuminant direction. In the equation, I_a, I_b and I_s are coefficients for the ambient component, diffuse component, and specular component respectively. The exponent n is the roughness of the surface. The value for I_a can be determined from the intensity on the background of the images. Under a collinear light source, as shown in Fig. 2, the viewing direction is the same as the illuminant direction. Thus the half vector H is in the same direction as L and $N \cdot L = N \cdot H = \cos(i)$, where i is the angle from the illuminant direction L and the surface normal N and is called the incident angle in computer vision literature [8]. Therefore the total image intensity of a surface point under a collinear light source can be expressed as

$$I = R(i), \tag{4}$$

which is a function of only the incident angle i. We assume the function is strictly monotonic and its inverse exists. This is true for most real object surfaces.

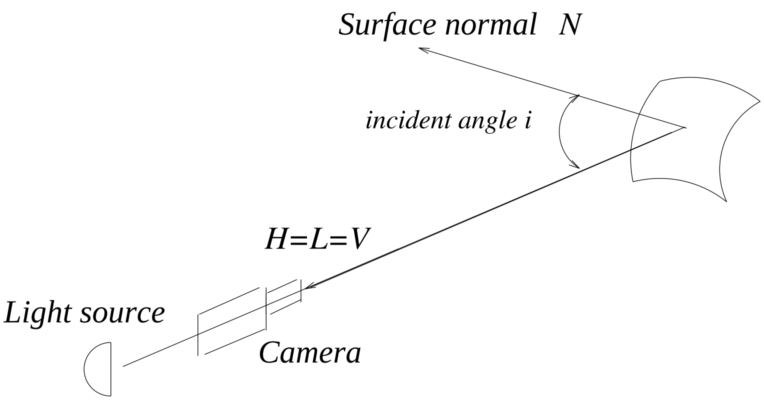
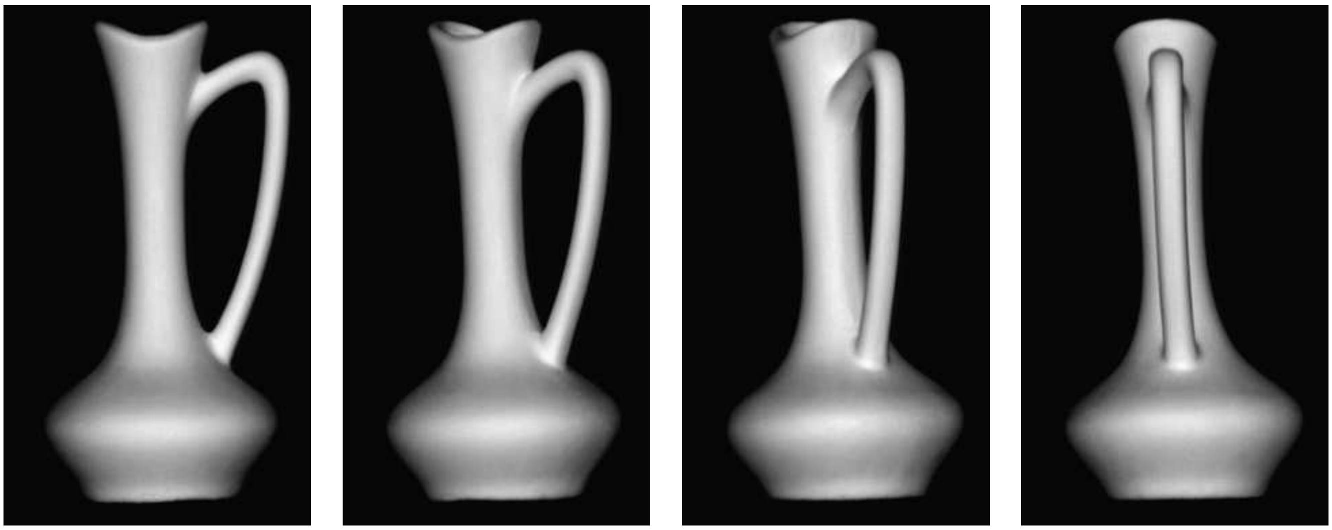


Figure 2: Under collinear light, the image intensity depends only on angle i

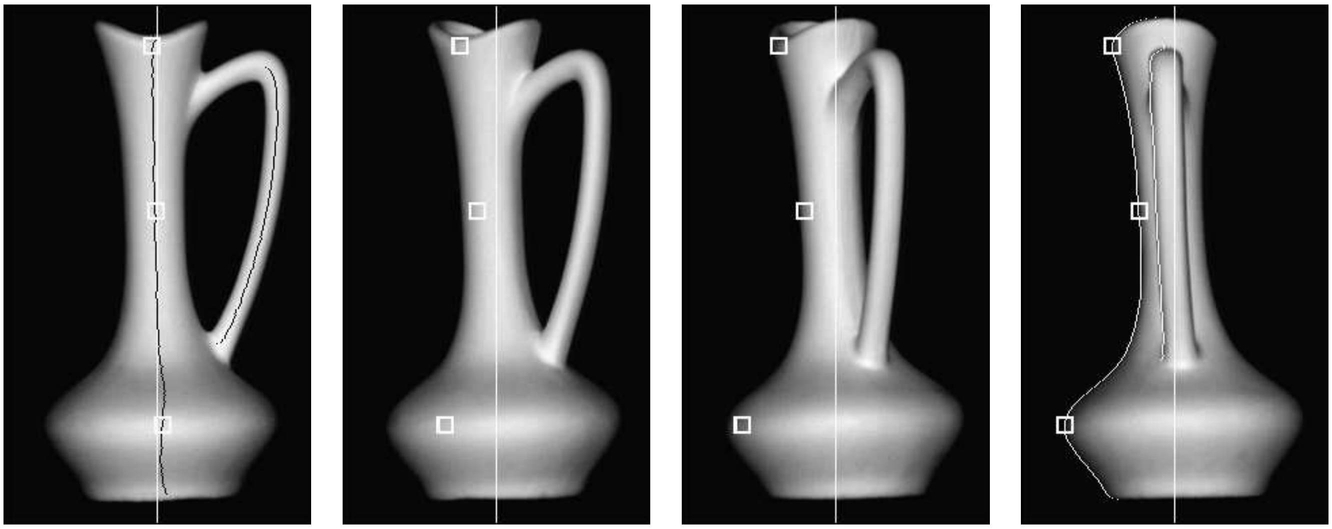
3 Extracting Surface Reflectance Function



(a. 0^o rotation) (b. 30^o rotation) (c. 60^o rotation) (d. 90^o rotation)

Figure 3: Images of a rotating vase

Fig. 3 shows four images in an image sequence of the rotating vase. Each image in the sequence is taken after 5 degree successive rotation of the vase. Under a collinear light source, the surface points whose normals are in the view direction have the maximum intensity value in an image. Some surface points which give the maximum intensity values



(a. 0^o rotation) (b. 30^o rotation) (c. 60^o rotation) (d. 90^o rotation)

Figure 4: Tracking the sample points over image sequence

in the first image are taken as the sample points for extracting the surface reflectance function. The x and y coordinates of the sample points can be determined by searching for image points of local maximum image intensity value in the first image. The center of a small square box in image (a) of Fig. 4 denotes x and y coordinates of a sample points before the object rotation. The z coordinates of the sample points can be determined from the silhouettes of the object in the image taken after 90 degree rotation. The silhouette (shown in Fig. 4 (d) as white curves) can be found by thresholding the background intensity value or Canny edges detector [2]. Since the y coordinate of a sample point is known, the z coordinate of the sample point can be determined from the horizontal distance of the corresponding point on the silhouette to the projection of the rotation axis.

Given the 3D coordinates of the sample points on the object surface before the rotation, we can calculated their 3D locations and image projections in the image sequence. Therefore their intensity values in the image sequence can be extracted. The center of the small square boxes in Fig. 4 gives the trace of these sample points in the image sequence. Before the rotation of the object, the surface normals of these sample points are in the viewing direction and the incident angles at the sample points are zero. During the object rotation, the surface normals rotate with the object and the incident angles at these sample points are the same as the rotation angle of the object.

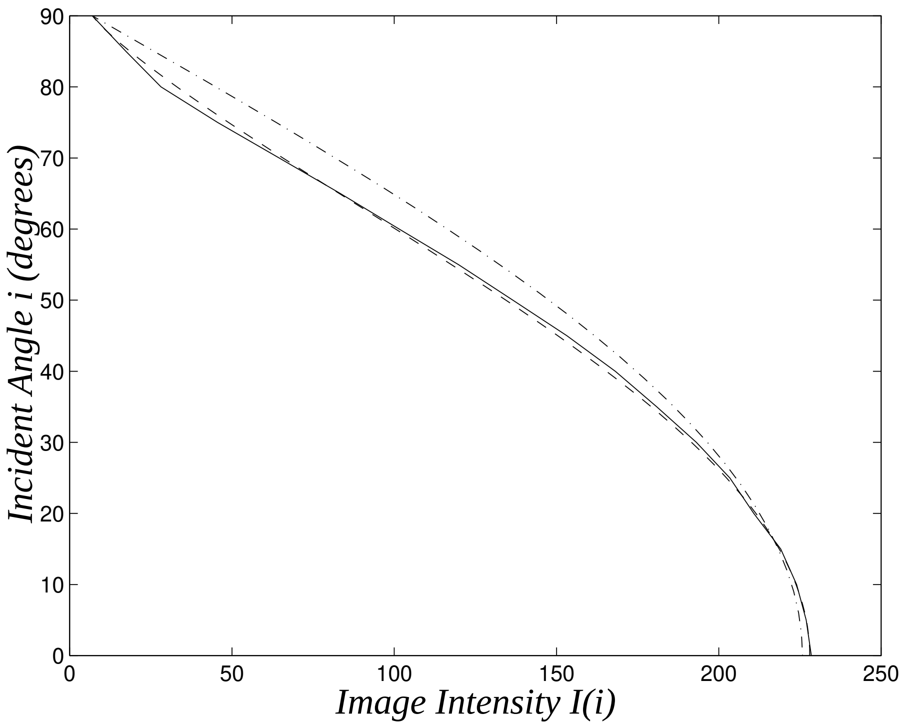


Figure 5: The extracted reflectance function (solid line), the Lambertian reflectance function (dash-dotted line) of the same maximum brightness, and the calculated shading function (dashed line) of Phong-Blinn model.

The extracted intensity values and the incident angles of the sample points in the image sequence are used to build the surface reflectance function. If more than one sample point is used, the average of their intensity values gives a more accurate result. The continuous line in Fig. 5 shows the surface reflectance function of the vase extracted from the three sample points and the dash-dot line is a Lambertian reflectance function of the same maximum brightness value. Although the surface of the vase is considered a diffuse surface but it is not a perfect Lambertian surface.

4 Surface Recovery

Surface recovery can be done by using any two images in the image sequence of the rotating object after the reflectance function has been obtained. Without losing generality, we use the first image and the image taken after the object has been rotated by 10 degree rotation. The depth and surface orientation are computed at every point in the first image. If the intensity values of a surface point in the two images are known, its surface orientation can be derived from the following calculation.

Let $image_0$ be the image taken before the rotation and $image_\alpha$ be the image taken after α degree rotation of the object. Let (x_0, y_0) and (x_α, y_α) be the projections of a 3D surface point in $image_0$ and $image_\alpha$ respectively and their brightness values are $E(x_0, y_0)$ and $E(x_\alpha, y_\alpha)$. Using the inverse reflectance function $i = R^{-1}(E)$, we obtain the incident angle i_0 and i_α from $E(x_0, y_0)$ and $E(x_\alpha, y_\alpha)$. Let the surface orientation of the 3D point be $(-p_0, -q_0, 1)$ when $image_0$ is taken and the surface orientation of the same 3D point be $(-p_\alpha, -q_\alpha, 1)$ when $image_\alpha$ is taken. From the definition of the incident angle and the transformation between the object coordinates, we have

$$\cos i_0 = \frac{1}{\sqrt{p_0^2 + q_0^2 + 1}}, \tag{5}$$

$$\cos i_\alpha = \frac{1}{\sqrt{p_1^2 + q_1^2 + 1}}, \tag{6}$$

$$p_1 = \frac{p_0 \cos\alpha + \sin\alpha}{\cos\alpha - p_0 \sin\alpha}, \tag{7}$$

and

$$q_1 = \frac{q_0}{\cos\alpha - p_0 \sin\alpha}. \tag{8}$$

Substituting p_α and q_α in Equation 6, we get

$$\cos i_\alpha = \frac{1}{\sqrt{1 + (\frac{p_0 \cos\alpha + \sin\alpha}{\cos\alpha - p_0 \sin\alpha})^2 + (\frac{q_0}{\cos\alpha - p_0 \sin\alpha})^2}}. \tag{9}$$

The equation can be simplified to

$$\cos i_\alpha = \cos i_0 (\cos\alpha - p_0 \sin\alpha). \tag{10}$$

Solving Equation 10 for p_0 and and then solving Equation 5 for q_0, we have

$$p_0 = \frac{1}{\tan\alpha} - \frac{\cos i_a lpha}{\cos i_0 \sin\alpha}, \tag{11}$$

$$q_0 = \pm\sqrt{\frac{1}{\cos i_0^2} - p_0^2 - 1}. \tag{12}$$

The surface recovery procedure starts at the image points whose depth are known. These starting points could be the sample points we used to extract the reflectance function. For each starting point, a subprocedure called *find-correspondent* is used to compute its corresponding image point in $image_\alpha$. The second subprocedure called *invert-shading* is used to determined the orientation at a surface point from the brightness values of its two image points in $image_0$ and $image_\alpha$. The third subprocedure called *extend-depth* is used to calculate the depth of the image points in the neighborhood of the staring point. Applying the three subprocedures on the neighbor points, we expand surface orientation and depth over a larger image area in $image_0$. Iteratively applying the three subprocedures, the computation can be expanded over the whole object image to obtain the depth and surface orientation at the same time. The surface recovery procedure can be described as follows:

Surface recovery procedure

1. take a point in $image_0$ of known depth as an input point. For the input point, do step 2-4;

2. use *find-correspondent* to calculate the location of its corresponding image point in $image_\alpha$.

3. use *invert-shading* to determine its surface orientation from its brightness in $image_0$ and the brightness of the corresponding image point in $image_\alpha$.

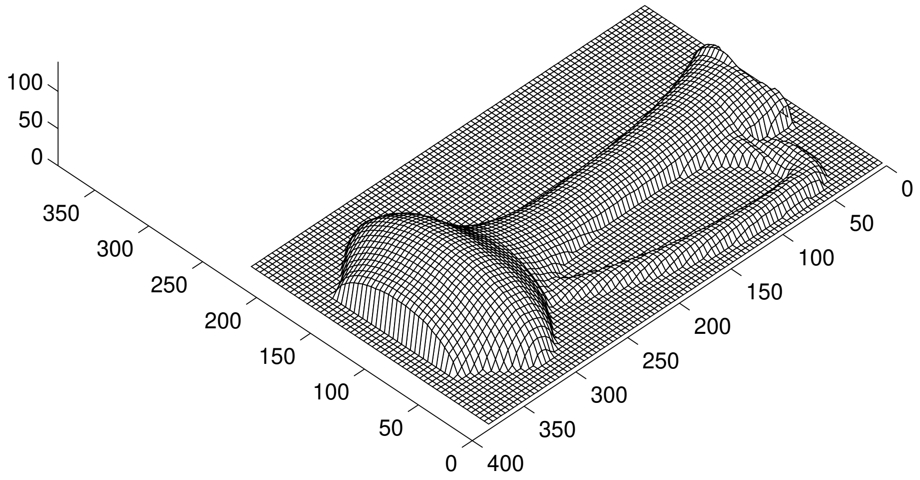
4. apply *extend-depth* to compute the depth for its neighboring points using first-order Taylor approximation, $z = p\delta x + q\delta y$.

5. take each of the neighboring points as the input point and repeat step 2-4 until the computation is extended to every pixel in $image_0$.

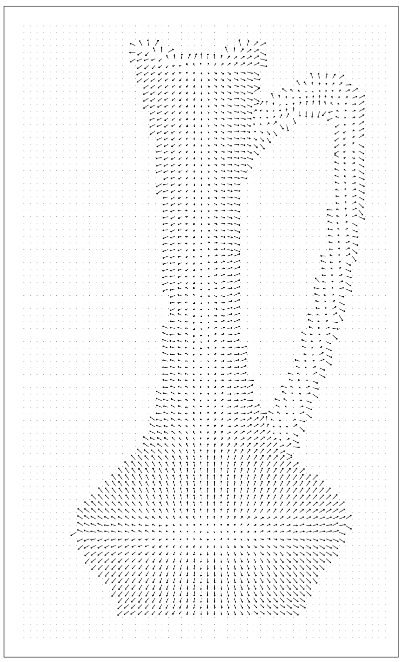
We first compute the depth and surface orientation along curves of $p = 0$ (see dark curves in image (a) of Fig. 4). The depth of the $p = 0$ curves are determined by the corresponding silhouette and occluding contour in the image taken after 90 degree rotation (see Fig. 4 (d)). The horizontal distance from the silhouette or the occluding contour gives the depth for the corresponding $p = 0$ curve. Then we expand the computation on the depth and surface orientation in the x direction by $z' = z + p\delta x$ until we reach the background. This process may not reach some areas, such as the top of the body part. Then we expand the computation in the y direction to reach the unrecovered areas. Fig. 6 shows the surface depth plot and surface orientation diagram of the recovered vase. The surface orientation diagram is obtained by projecting the unit surface normal on the original image. The surface orientation data is used in the surface rendering. A more detailed description of the surface recovery procedure can be found in [9].

5 Calculating the shading function

In computer vision and computer graphics, the terminology used to describe the relation between image brightness and surface orientation is different. In computer vision, the relation is called the surface reflectance function. In computer graphics, the relation is called the shading function. The shading function is used to render object models and generate synthetic images. One of the objectives in computer graphics is to generate virtual images which look like real images. In computer graphics, a shading function usually has a set of parameters and these parameters are determined heuristically. Some work has been done in trying to get physics-based shading function by analytical modeling [3, 6] or by using measurement devices [18]. We attempt to get the shading function from real images.



a. The depth plot



b. The surface orientation plot

Figure 6: The surface depth and orientation of the recovered vase

The surface reflectance function extracted from real images gives a description of how bright a real surface point in the real physical world will be when a surface is illuminated and viewed under certain conditions. However, the surface reflectance function extracted from images can only give the relation between brightness value and surface orientation under some fixed illuminating and viewing conditions in which these images are taken. It can't be directly used in compute graphics to render the objects under arbitrary illuminating and viewing directions. The surface reflectance function can be considered as a shading function under a collinear light source and then the parameters for the shading function can be calculated from the surface reflectance function. Once the shading function is obtained, it can be used to render object surfaces under arbitrary illuminating and viewing directions.

The surface reflectance function we extracted is a function under a collinear light source and it has the form

$$E = Q(\cos i)$$

with i as the incident angle at a surface point. One of the most widely used shading models in computer graphics is the Phong-Blinn shading model [1]. It has the form

$$I = I_a + I_d(N \cdot L) + I_s(N \cdot H)^n, \tag{13}$$

where N is the surface normal, L is the illuminant direction vector, and H is the vector halfway between the viewing direction and the illuminant direction. In the function, I_a, I_d and I_s are coefficients for the ambient component, diffuse component, and specular component respectively. The exponent n in the function represents the roughness of the surface. In computer graphics rendering, these parameters are determined by *ad hoc* methods. We consider the surface reflectance function extracted from the real images as the shading function under a collinear light source. In this case,

$$N \cdot L = \cos i \quad \& \quad N \cdot H = \cos i$$

and

$$I = I_a + I_d \cos i + I_s \cos^n i.$$

Now our task is to find a set of values for I_a, I_d, I_s and n which minimizes the difference of the two functions under a collinear light source. The value for I_a can be determined from the background brightness in the real images. When $\cos i = 1$, the surface reflectance function takes the maximum brightness value and so should the shading function. Let the maximum brightness value be I_{max}; then

$$I_{max} = I_a + I_d + I_s \quad \& \quad I_s = I_{max} - I_a - I_d.$$

The shading function under a collinear light source can be rewritten as

$$I = I_a + I_d \cos i + (I_{max} - I_a - I_d) \cos^n i.$$

Since I_a and I_{max} are known, only I_d and n need to be determined.

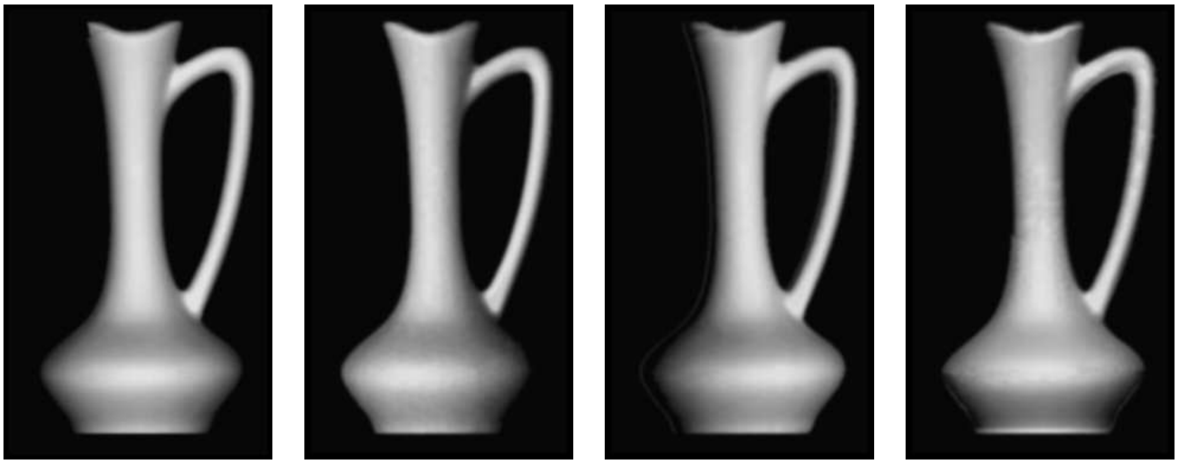
The values for I_d and I_{max} are determined by minimizing the least-squares error of the two functions which can be expressed as

$$f(\cos i, I_d, n) = \sum [Q(\cos i) - (I_a + I_d \cos i + (I_{max} - I_a - I_d) \cos^n i)]^2.$$

Starting from an initial guess for n, I_d can be calculated by solving $\partial f(\cos i, I_d, n)/\partial I_d$. By using one dimensional search with first derivatives, a new value of n which minimizes $f(\cos i, I_d, n)$ can be found. By using the new value of n, a new value for I_d can be calculated. Iterating the above process, the values of I_d and n which minimize the least-squares error can be determined. The parameters for the shading function calculated from the surface reflectance function of the vase are $I_a = 7$, $I_d = 80$, $I_s = 141$ and $n = 1.4$. Thus the obtained shading function is

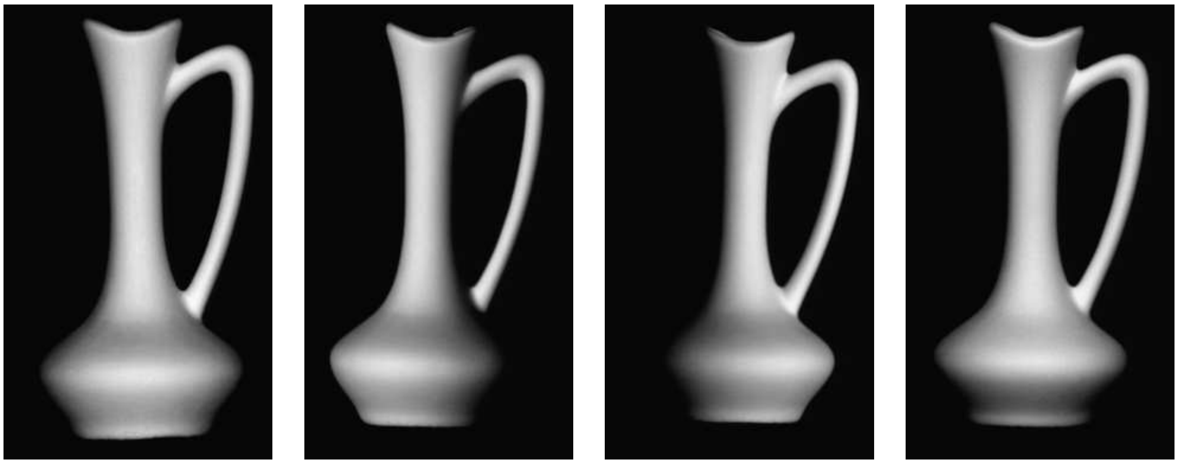
$$I = 7 + 80(N \cdot L) + 141(N \cdot H)^{1.4}$$

and shown as a dashed line in Fig. 5. Though the surface of the vase is close to a diffuse surface there is still a weak specular surface reflectance component in the shading function.



a. illuminated at $(0, 0, 1)$ | b. illuminated at $(-\tan\frac{\pi}{6}, 0, 1)$ | c. illuminated at $(\tan\frac{\pi}{6}, 0, 1)$ | d. illuminated at $(0, \tan\frac{\pi}{10}, 1)$

Synthetic images of the recovered surface



a. illuminated at $(0, 0, 1)$ | b. illuminated at $(-\tan\frac{\pi}{6}, 0, 1)$ | c. illuminated at $(\tan\frac{\pi}{6}, 0, 1)$ | d. illuminated at $(0, \tan\frac{\pi}{10}, 1)$

Real images of the original object

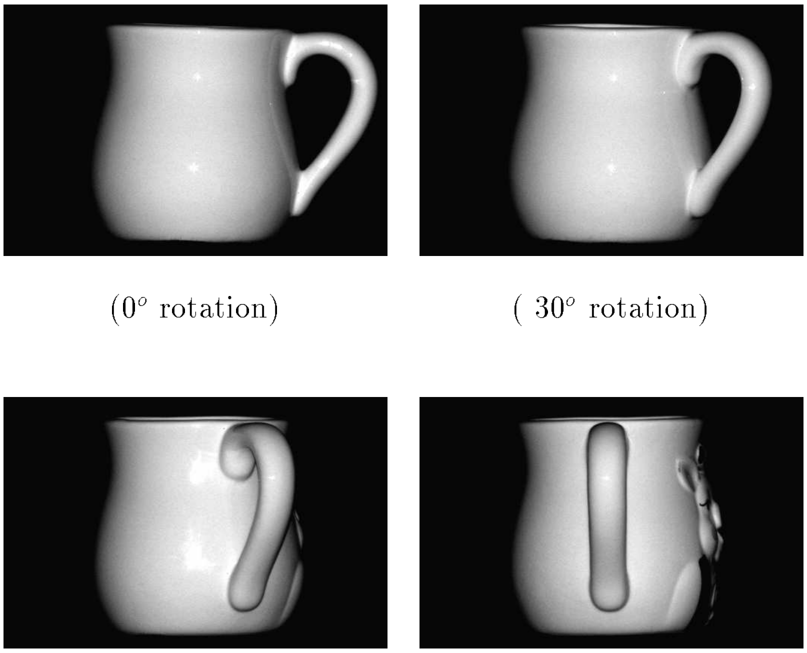
Figure 7: The synthetic images of (a)-(d) are rendered with the shading function estimated from the surface reflectance function extracted from the real images. The illuminant and viewing conditions for generating a synthetic image are similar to those used in taking the corresponding real image in the same column.

6 Rendering with the Calculated Shading Function

Using the calculated shading function and the surface orientation data recovered from the image sequence, we generate synthetic images of the recovered surface by rendering the surface under different illuminant directions but fixed viewing direction. Figure 7 shows two groups of images. The images in the upper row are synthetic images generated under different illuminant directions. The images in the lower row are the corresponding real images. The illuminant direction and viewing direction for the synthetic image and the real image in the same column are similar. The location and orientation of the object in the synthetic image and the corresponding real image are similar. The synthetic images are generated by a simple shading algorithm which only considers local shading, i.e., the shading directly caused by the light source illumination. To simulate the blurring effect of the camera, the synthetic images are filtered with a Gaussian filter of $\sigma = 1$. Comparing the synthetic images with the real images in the same column, we see that the differences between the synthetic images and real images are very small. The small differences show that the shading function is a good approximation of the real surface reflectance function and it gives realistic rendering.

7 Experiment with a Specular Surface

We performed another experiment with a porcelain cup and tried to calculate a shading function of the Phong-Blinn model from the surface reflectance function extracted from the images of the cup. The surface of the cup is quite specular. Some images from the image sequence of the rotating cup are shown in Fig. 8. The reflectance function extracted from the cup is shown in Fig. 9 as a solid line. The recovered surface of the cup in shown in Fig. 11. Since the surface of the cup is highly specular, the singular points in the images have saturated brightness values. The value for I_{max} should be larger than the saturated brightness value. An initial larger value is used as a guess for I_{max}. To calculate a shading function from the surface reflectance function of the cup, the least-squares error of the two



(0^o rotation) (30^o rotation)

(60^o rotation) (90^o rotation)

Figure 8: Images of a rotating cup

functions under a collinear light source is minimized. The initial value of I_{max} is set to 297 and the same minimization method (see Section 5) is used to get the shading function

$$I = 7 + 220(N \cdot L) + 70(N \cdot H)^{2000}.$$

The function is shown in Fig. 9 as a dashed line.

There are two groups of images in Fig. 12. Each image on the left side is a synthetic image and each image on the right side is the real image corresponding to the left side synthetic image. The synthetic images are generated by using the recovered surface orientation data of the cup and the calculated shading function. The location and orientation of the object in a synthetic image and the corresponding real image are similar. Each image in the synthetic images is rendered under the similar illuminating and viewing directions to those under which the corresponding real image is taken. Visual comparison can be made between a synthetic image and its corresponding real image. The specular spots in the synthetic images show that the specular spike is modeled reasonably well. The dif-

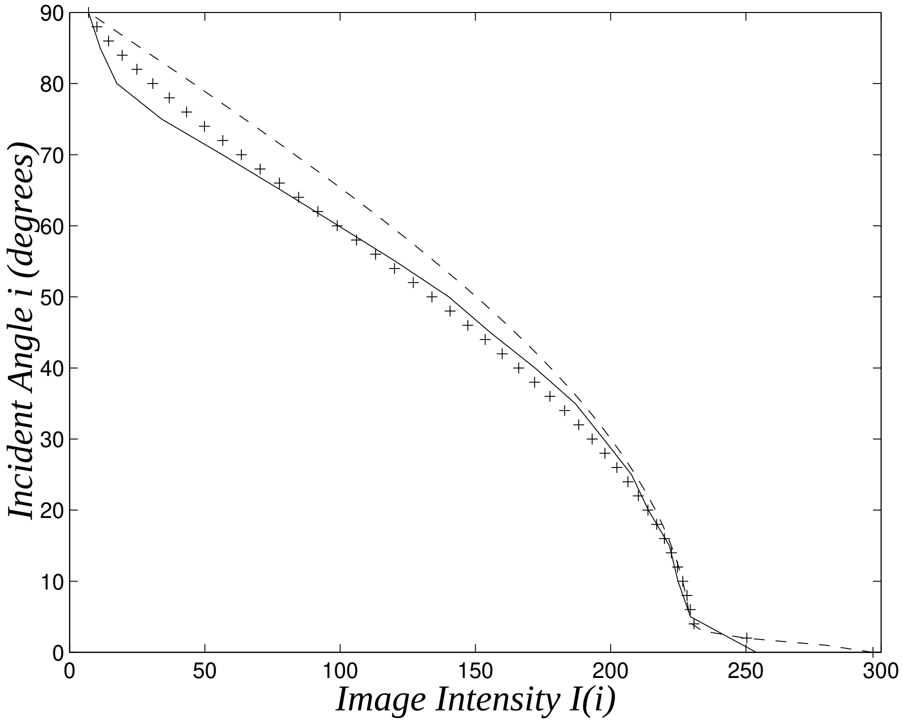


Figure 9: The solid line shows the surface reflectance function extracted from images. The dashed line shows the calculated shading function of the Phong-Blinn model. The star line shows a better shading function of different model which is closer to the surface reflectance function.

ferences between the synthetic images and real images are visible. The brightness in the synthetic images is not as smooth as that in the real images. The specular spots in the synthetic images are different from those in the real images. The differences are mainly caused by error in the recovered surface orientation. The orientation data, especially, the q component is not smooth.

8 Discussion and Future Work

The experimental results indicates that our technique works. However, the work described here is preliminary. The depth and surface orientation are recovered only from one view and

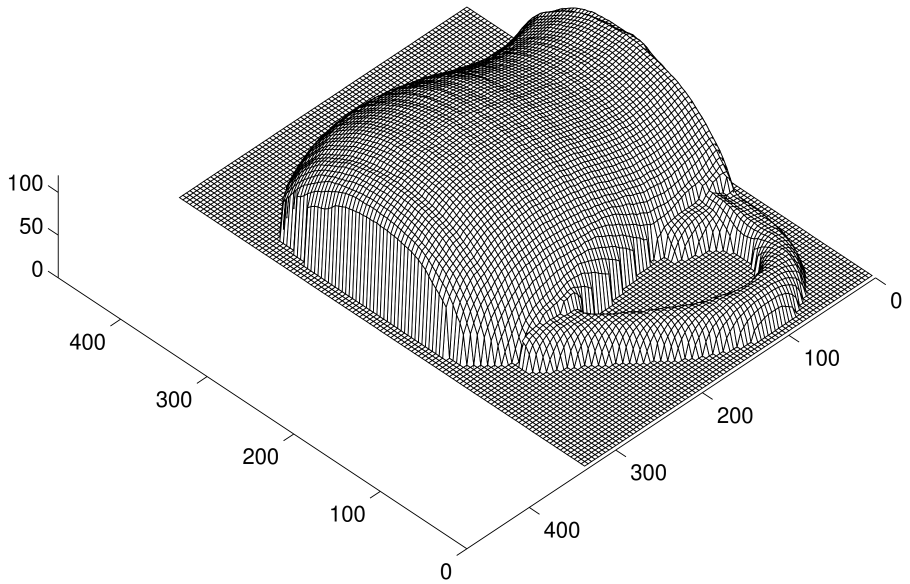


Figure 10: The depth plot of the final recovered cup

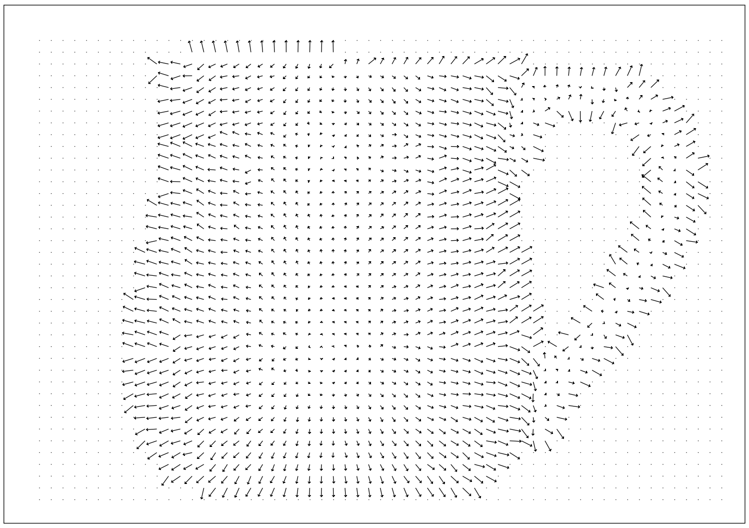


Figure 11: The surface orientation of the recovered cup

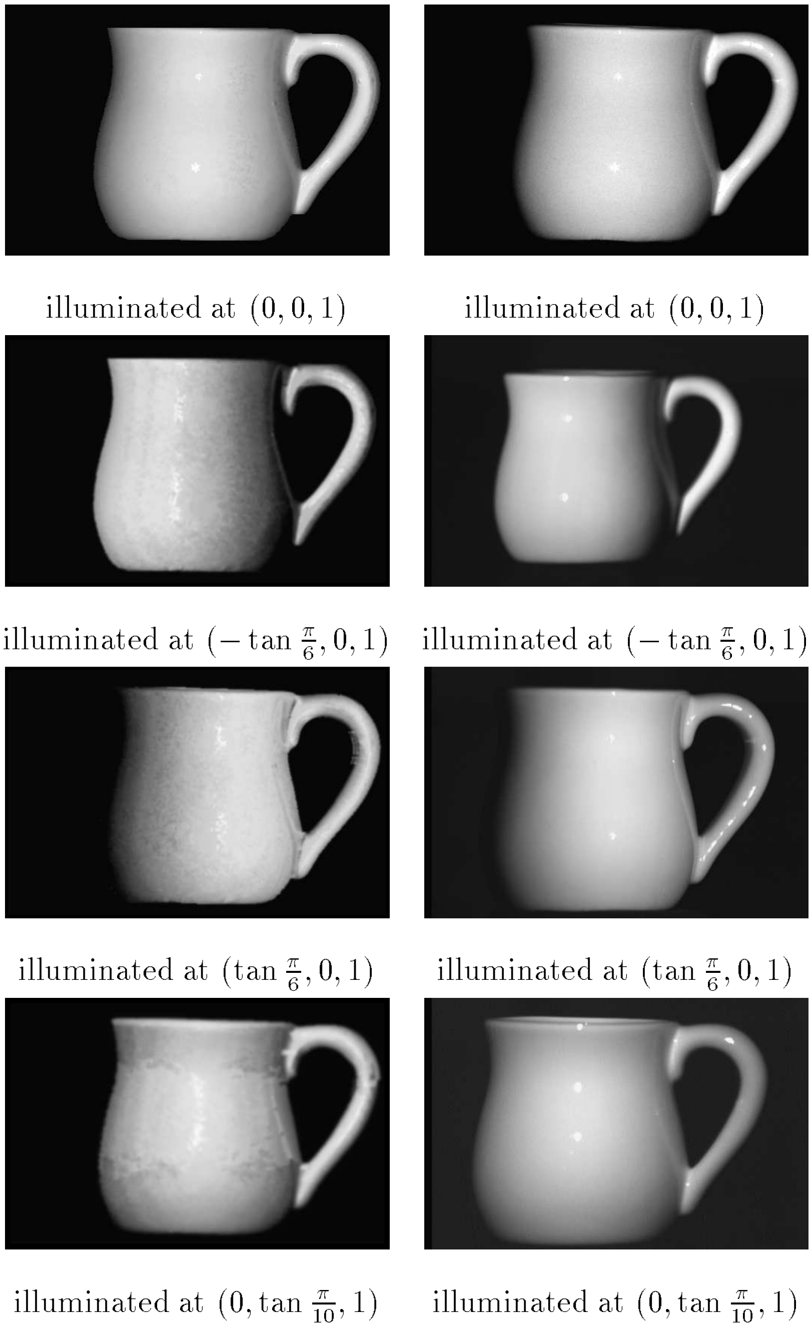


Figure 12: The left column is the synthetic images of the recovered surface and the right column is the real images of the original object. The synthetic images are rendered with the estimated shading function. The illuminant and viewing conditions for generating a synthetic image are similar to those used for taking the corresponding real image in the same row.

they are not continuous at some points along the y direction. The discontinuity in surface orientation is visible from the synthetic images of the recovered surface. We have tried to obtain smooth surfaces by using surface fitting techniques without achieving satisfactory results. The two surface fitting techniques [7, 4] we used require that the distance between adjacent data points be uniform. But the data obtained by our surface recovery technique does not meet the requirement. Fitting the depth and surface orientation data with smooth surfaces is not trivial so we take this as a future research direction.

The reflectance function of some surfaces cannot be model properly by the Phong-Blinn shading model. The four component model in computer vision [12] gives a better description of the surface reflectance of the porcelain cup. The model contains four components: ambient, diffuse, specular and specular spike. A shading function which fits the reflectance function of the cup better is

$$I = 7 + 10(N \cdot L) + 210(N \cdot H)^{1.3} + 70(N \cdot H)^{2000}.$$

It contains a weak specular component and a strong specular spike. The function is shown in Fig. 9 as a dashed line.

The integration of computer vision and computer graphics is inevitable. The positive initial results demonstrate the potential of computer vision techniques to enhance realistic modeling and rendering in computer graphics.

References

[1] J. F. Blinn. Models of light reflection for computer synthesized pictures. In *Proc. SIGGRAPH, 1977*, pages 192–198, 1977.

[2] J. F. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.

[3] R. L. Cook and K. E. Torrance. A reflectance model for computer graphics. *ACM Transactions on Graphics*, 1:7–24, 1982.

[4] H. Edelsbrunner and E. P. Mucke. Three dimensional alpha shape. *ACM Transactions on Graphics*, 13:43–72, 1994.

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

[6] X. D. He, K. E. Torrance, F. X. Sillion, and D. P. Greenberg. A comprehensive physical model for light reflection. In *Proc. SIGGRAPH, 1991*, pages 187–196, 1991.

[7] H. Hoppe and T. D. *et al* . Piecewise smooth surface reconstruction. In *Proceedings of SIGGRAPH '94, July 24–29*, Orlando, FL, 1994.

[8] B. K. P. Horn. Understanding image intensities. *Artificial Intelligence*, 8:201–231, 1977.

[9] J. Lu and J. Little. Reflectance function estimation and shape recovery from image sequence of a rotating object. In *Proc. 5th International Conference on Computer Vision*, pages 80–86, 1995.

[10] J. Lu and J. J. Little. Surface reflectance and shape from images using a collinear light source. TR-97–05, UBC Dept. of Computer Science, Vancouver, BC, 1997.

[11] T. McInerney and D. Terzopoulos. A finite element model for 3d shape reconstruction and nonrigid motion tracking. In *Proc. 4th International Conference on Computer Vision*, pages 518–523. IEEE, 1993.

[12] S. Nayar, K. Ikeuchi, and T. Kanade. Surface reflection: physical and geometrical perspectives. In *Proceedings: Image Understanding Workshop*, pages 185–212. DARPA, 1990.

[13] M. Oren and S. K. Nayar. Seeing beyond Lambert's law. In *Proc. 3rd European Conference on Computer Vision*, pages 269–280, May 1994.

[14] R. Szeliski and D. Tonnesen. Surface modeling with oriented particle systems. In *Computer Graphics, (SIGGRAPH'92)*, pages 185–193, 1992.

[15] R. Szeliski, D. Tonnesen, and D. Terzopoulos. Modeling surfaces of arbitrary topology with dynamic particles. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition, 1993*, pages 82–87, New York, NY, USA, 1993.

[16] D. Terzopoulos and D. Metaxas. Dynamic 3d models with local and global deformations: deformable superquadrics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(7):703–714, July 1991.

[17] D. Terzopoulos and M. Vasilescu. Sampling and reconstruction with adaptive meshes. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition, 1991*, pages 70–75. IEEE, 1991.

[18] G. J. Ward. Measuring and modeling anisotropic reflection. In *Proc. SIGGRAPH, 1992*, pages 265–272, 1992.

[19] L. B. Wolff. Diffuse-reflection model for smooth dielectric surfaces. *Journal of the Optical Society of America*, 11(11):2956–2968, Nov. 1994.