Use of Multiple Surface Normal Approximations in the Shape-from-Shading Using Jacobi's Iterative Method

Osamu Ikeda
Faculty of Engineering, Takushoku University
815-1 Tate, Hachioji, Tokyo, 193-0985 Japan
oikeda@cc.takushoku-u.ac.jp

Abstract

We previously showed that the shape-from-shading algorithm using Jacobi's iterative method is promising in terms of accuracy and simplicity. In this paper we investigate the effects of approximations of the surface normal on the shape estimating process. Specifically the four different simplest approximations are shown to give four different reconstruction functions of shape, which are complementary to each other from the viewpoint of the contributing region. The previous algorithm is improved by combining the four approximations and by imposing appropriate boundary conditions, to estimate more accurate shape.

Keywords: shape-from-shading, boundary conditions, Jacobi iterative method

1 Introduction

Since the pioneering research on shape-from-shading by Horn [1], significant studies have been made [2], [3]. A large number of papers focus on estimating shape from a single shading image. They may be classified to local [4], minimization [5], [6], linear [7], [8] and propagation [9], [10] approaches. A smaller number of papers, on the other hand, focus on the estimation from multiple images [11].

Minimization approaches are based on minimizing a given energy criterion to estimate the shape. Zheng and Chellappa [5] introduced, for example, image gradient and integrability constraints to obtain fine details. Linear approaches linearize the reflectance map in tilts or depth. Tsai and Shah [7] linearized it in the depth and used the Jacobi iterative technique. We generalized their method to enhance its applicability and to improve accuracy [12]. Propagation approaches obtain a shape starting from some initial curve at the brightest or darkest points. Kimmel et al. show that good shape reconstruction is possible with boundary conditions [10]. As for the estimation using multiple images, Woodham shows that using three shading images can uniquely determine the surface normal map [13]. Several algorithms have been reported to convert the map to the corresponding depth map [11], but they appear successful to a certain degree.

In this paper we investigate the effects of the surface normal approximation on the estimation process and the resulting shapes, where we specifically compare four different simplest approximations. We show that they give rise to four different and asymmetrical reconstruction functions of shape and that they could be complementary to each other since they share none of the regions. Taking advantage of the properties, the previous algorithm is improved so as to reconstruct shapes with multiple approximations and average them during the iteration. We also show that the improvement makes it possible to effectively impose appropriate boundary conditions.

2 Basic Iterative Relations

The object is illuminated from a single direction to obtain its shading image. Given an appropriate reflectance map $R(p,q)$, it may be equal to the image $I(x,y)$:

$$R(p,q) = I(x,y) \tag{1}$$

where $x,y=1,\ldots,N$, and the image is normalized to unity. Let **P** and **S** be the surface normal of the depth $z(x,y)$ of the object and the illuminant vector, respectively:

$$\mathbf{P} = (p,q,1)^T \Big/ \sqrt{p^2+q^2+1} \tag{2}$$

$$\mathbf{S} = (s,t,1)^T \Big/ \sqrt{s^2+t^2+1} \tag{3}$$

where p and q are given by $-\partial z/\partial x$ and $-\partial z/\partial y$, respectively. Then, for the *Lambertian* surface, the reflectance function, normalized by the *albedo*, is given by their scalar product:

$$R(p,q) = \frac{1+ps+qt}{\sqrt{1+p^2+q^2}\sqrt{1+s^2+t^2}} \tag{4}$$

Here we may consider the four simplest approximations for p and q:

$$(p(x,y), q(x,y)) = \begin{cases} (z(x-1,y)-z(x,y),\ z(x,y-1)-z(x,y)) & \text{(5a)} \\ (z(x,y)-z(x+1,y),\ z(x,y)-z(x,y+1)) & \text{(5b)} \\ (z(x-1,y)-z(x,y),\ z(x,y)-z(x,y+1)) & \text{(5c)} \\ (z(x,y)-z(x+1,y),\ z(x,y-1)-z(x,y)) & \text{(5d)} \end{cases}$$

where we used the first one in the previous paper. In either case the R can be regarded to be a function of three z variables, for example, $z(x,y)$, $z(x\text{-}1,y)$ and $z(x,y\text{-}1)$ for the case of Eq. (5a). Then, first, defining $f(x,y)$ as

$$f(x,y) \equiv I(x,y) - R(p,q) \tag{6}$$

and applying the Jacobi's iterative method to $f(x,y)$, we obtain the following relations corresponding to the approximations in Eq. (5):

$$-f_{x,y}^{(n-1)} = \left(\frac{\partial f_{x,y}}{\partial z_{x,y}}\right)^{(n-1)} \left(z_{x,y}^{(n)} - z_{x,y}^{(n-1)}\right) + \left(\frac{\partial f_{x,y}}{\partial z_{x-1,y}}\right)^{(n-1)} \left(z_{x-1,y}^{(n)} - z_{x-1,y}^{(n-1)}\right) + \left(\frac{\partial f_{x,y}}{\partial z_{x,y-1}}\right)^{(n-1)} \left(z_{x,y-1}^{(n)} - z_{x,y-1}^{(n-1)}\right) \tag{7a}$$

$$-f_{x,y}^{(n-1)} = \left(\frac{\partial f_{x,y}}{\partial z_{x,y}}\right)^{(n-1)} \left(z_{x,y}^{(n)} - z_{x,y}^{(n-1)}\right) + \left(\frac{\partial f_{x,y}}{\partial z_{x+1,y}}\right)^{(n-1)} \left(z_{x+1,y}^{(n)} - z_{x+1,y}^{(n-1)}\right) + \left(\frac{\partial f_{x,y}}{\partial z_{x,y+1}}\right)^{(n-1)} \left(z_{x,y+1}^{(n)} - z_{x,y+1}^{(n-1)}\right) \tag{7b}$$

$$-f_{x,y}^{(n-1)} = \left(\frac{\partial f_{x,y}}{\partial z_{x,y}}\right)^{(n-1)} \left(z_{x,y}^{(n)} - z_{x,y}^{(n-1)}\right) + \left(\frac{\partial f_{x,y}}{\partial z_{x-1,y}}\right)^{(n-1)} \left(z_{x-1,y}^{(n)} - z_{x-1,y}^{(n-1)}\right) + \left(\frac{\partial f_{x,y}}{\partial z_{x,y+1}}\right)^{(n-1)} \left(z_{x,y+1}^{(n)} - z_{x,y+1}^{(n-1)}\right) \tag{7c}$$

$$-f_{x,y}^{(n-1)} = \left(\frac{\partial f_{x,y}}{\partial z_{x,y}}\right)^{(n-1)} \left(z_{x,y}^{(n)} - z_{x,y}^{(n-1)}\right) + \left(\frac{\partial f_{x,y}}{\partial z_{x+1,y}}\right)^{(n-1)} \left(z_{x+1,y}^{(n)} - z_{x+1,y}^{(n-1)}\right) + \left(\frac{\partial f_{x,y}}{\partial z_{x,y-1}}\right)^{(n-1)} \left(z_{x,y-1}^{(n)} - z_{x,y-1}^{(n-1)}\right) \tag{7d}$$

where n is the number of iterations, $f_{x,y} \equiv f(x,y)$ and $z_{x,y} \equiv z(x,y)$. The four relations can be rewritten in matrix form as

$$-\mathbf{f}_a^{(n-1)} = \mathbf{g}_a^{(n-1)}(\mathbf{z}^{(n)} - \mathbf{z}^{(n-1)}), a=1,\ldots,4,\ n=1,2,\ldots \tag{8}$$

where $\mathbf{f}_a$ and $\mathbf{z}$ are N^2-elements column vectors of $f(x,y)$ and $z(x,y)$, respectively, and $\mathbf{g}_a$, a=1 to 4, which correspond to Eqs. (7a) to (7d), respectively, are N^2xN^2-elements matrices made of $\partial f(x,y)/\partial z(x,y)$, $\partial f(x,y)/\partial z(x\text{-}1,y)$, $\partial f(x,y)/\partial z(x\text{+}1,y)$, $\partial f(x,y)/\partial z(x,y\text{-}1)$ and/or $\partial f(x,y)/\partial z(x,y\text{+}1)$. Eq. (8) can be rewritten for $\mathbf{z}^{(n)}$ as

$$\mathbf{z}^{(n)} = \mathbf{z}^{(n-1)} - \left(\mathbf{g}_a^{(n-1)}\right)^{-1} \mathbf{f}_a^{(n-1)}, a=1,\ldots,4,\ n=1,2,\ldots \tag{9}$$

That is, the shape may be estimated iteratively using one of the relations in Eq. (9), typically beginning with null values $\mathbf{z}^{(0)}=\mathbf{0}$. In this case, since $\mathbf{g}_a$, a=1 to 4, are different, we may expect the resulting shapes to be different from each other, as will be described in the following section.

3 Four Reconstruction Functions

We set the elements of the two vectors of $\mathbf{z}=\{z_i\}$ and $\mathbf{f}=\{f_i\}$ as $z_{x+Ny}=z(x,y)$ and $f_{x+Ny}=f(x,y)$, respectively. The top left parts of the four matrices $\mathbf{g}_a$, $\{g_{i,j}\}$, i=1 to $2N$, j=1 to $2N$, are given in Eqs. (10a) to (10d), to show that the four different approximations give four matrices different from each other. They are also schematically illustrated on the left in Fig. 1, where non-zero elements exist along the three slant lines. Their inverse matrices are illustrated in the middle, where non-zero elements exist in the shaded regions. Then, the elements, which act as weights on the elements of $\mathbf{f}$ in the integration to get depth values, are shown to be in the shaded regions on the (x,y) plane on the right of Fig. 1. It is clearly seen that those four are completely different in region. That is, the four approximations result in four different reconstruction functions of shape.

$$\left[\begin{array}{cccc:cccc}
\frac{\partial f_{1,1}}{\partial z_{1,1}} & 0 & . & 0 & 0 & 0 & . & 0 \\
\frac{\partial f_{2,1}}{\partial z_{1,1}} & \frac{\partial f_{2,1}}{\partial z_{2,1}} & . & 0 & 0 & 0 & . & 0 \\
. & . & . & 0 & . & . & . & . \\
0 & 0 & . & \frac{\partial f_{N,1}}{\partial z_{N,1}} & 0 & 0 & . & 0 \\
\hdashline
\frac{\partial f_{1,2}}{\partial z_{1,1}} & 0 & . & 0 & \frac{\partial f_{1,2}}{\partial z_{1,2}} & 0 & . & 0 \\
0 & \frac{\partial f_{2,2}}{\partial z_{2,1}} & . & 0 & \frac{\partial f_{2,2}}{\partial z_{1,2}} & \frac{\partial f_{2,2}}{\partial z_{2,2}} & . & 0 \\
. & . & . & . & . & . & . & . \\
0 & 0 & . & \frac{\partial f_{N,2}}{\partial z_{N,1}} & 0 & 0 & . & \frac{\partial f_{N,2}}{\partial z_{N,2}}
\end{array}\right] \tag{10a}$$

$$\left[\begin{array}{cccc:cccc}
\frac{\partial f_{1,1}}{\partial z_{1,1}} & \frac{\partial f_{1,1}}{\partial z_{2,1}} & . & 0 & \frac{\partial f_{1,1}}{\partial z_{1,2}} & 0 & . & 0 \\
0 & \frac{\partial f_{2,1}}{\partial z_{2,1}} & . & 0 & 0 & \frac{\partial f_{2,1}}{\partial z_{2,2}} & . & 0 \\
. & . & . & 0 & . & . & . & . \\
0 & 0 & . & \frac{\partial f_{N,1}}{\partial z_{N,1}} & 0 & 0 & . & \frac{\partial f_{N,1}}{\partial z_{N,2}} \\
\hdashline
0 & 0 & . & 0 & \frac{\partial f_{1,2}}{\partial z_{1,2}} & \frac{\partial f_{1,2}}{\partial z_{2,2}} & . & 0 \\
0 & 0 & . & 0 & 0 & \frac{\partial f_{2,2}}{\partial z_{2,2}} & . & 0 \\
. & . & . & . & . & . & . & . \\
0 & 0 & . & 0 & 0 & 0 & . & \frac{\partial f_{N,2}}{\partial z_{N,2}}
\end{array}\right] \tag{10b}$$

$$\begin{bmatrix} \frac{\partial f_{1,1}}{\partial z_{1,1}} & 0 & . & 0 & \frac{\partial f_{1,1}}{\partial z_{1,2}} & 0 & . & 0 \\ \frac{\partial f_{2,1}}{\partial z_{1,1}} & \frac{\partial f_{2,1}}{\partial z_{2,1}} & . & 0 & 0 & \frac{\partial f_{2,1}}{\partial z_{2,2}} & . & 0 \\ . & . & . & 0 & . & . & . & . \\ 0 & 0 & . & \frac{\partial f_{N,1}}{\partial z_{N,1}} & 0 & 0 & . & \frac{\partial f_{N,1}}{\partial z_{N,2}} \\ 0 & 0 & . & 0 & \frac{\partial f_{1,2}}{\partial z_{1,2}} & 0 & . & 0 \\ 0 & 0 & . & 0 & \frac{\partial f_{2,2}}{\partial z_{1,2}} & \frac{\partial f_{2,2}}{\partial z_{2,2}} & . & 0 \\ . & . & . & . & . & . & . & . \\ 0 & 0 & . & 0 & 0 & 0 & . & \frac{\partial f_{N,2}}{\partial z_{N,2}} \end{bmatrix} \quad (10c)$$

$$\begin{bmatrix} \frac{\partial f_{1,1}}{\partial z_{1,1}} & \frac{\partial f_{1,1}}{\partial z_{2,1}} & . & 0 & 0 & 0 & . & 0 \\ 0 & \frac{\partial f_{2,1}}{\partial z_{2,1}} & . & 0 & 0 & 0 & . & 0 \\ . & . & . & 0 & . & . & . & . \\ 0 & 0 & . & \frac{\partial f_{N,1}}{\partial z_{N,1}} & 0 & 0 & . & 0 \\ \frac{\partial f_{1,2}}{\partial z_{1,1}} & 0 & . & 0 & \frac{\partial f_{1,2}}{\partial z_{1,2}} & \frac{\partial f_{1,2}}{\partial z_{2,2}} & . & 0 \\ 0 & \frac{\partial f_{2,2}}{\partial z_{2,1}} & . & 0 & 0 & \frac{\partial f_{2,2}}{\partial z_{2,2}} & . & 0 \\ . & . & . & . & . & . & . & . \\ 0 & 0 & . & \frac{\partial f_{N,2}}{\partial z_{N,1}} & 0 & 0 & . & \frac{\partial f_{N,2}}{\partial z_{N,2}} \end{bmatrix} \quad (10d)$$

Let us check the distribution of the $\mathbf{g}^{-1}$ around the reconstruction point of shape. Examples are shown in Fig. 2 for the approximation of Eq. (5a). These show that the patterns, or the reconstruction functions, have sharp peaks at the reconstruction point, and that their monotonically decreasing lobes are significant along the directions of tilt angles of the illuminant vector. The functions may in general be expected to be isotropic, but those in Fig. 2 are clearly an-isotropic. We can also observe that the functions are invariable with the slant angle, that the function in the range of $180<\tau<270$ is the same as that for τ-180 but it has an opposite sign, and that the functions are hard to get in the range of $90<\tau<180$ or $270<\tau<360$ due to that the inverse matrix of **g** is not available.

4 Algorithm

From the properties of the reconstruction functions, when the tilt angle of **S** lies in $0<\tau<90$ or $180<\tau<270$, two approximations, Eqs. (5a) and (5b), may be used, while when it lies in $90<\tau<180$ or $270<\tau<360$ the other two approximations, Eqs. (5c) and (5d), may be used. Examples of available two functions of $\mathbf{g}^{-1}$ are illustrated in Fig. 3 for two illuminant vectors, **S** = (5,5,7) and (-5,5,7).

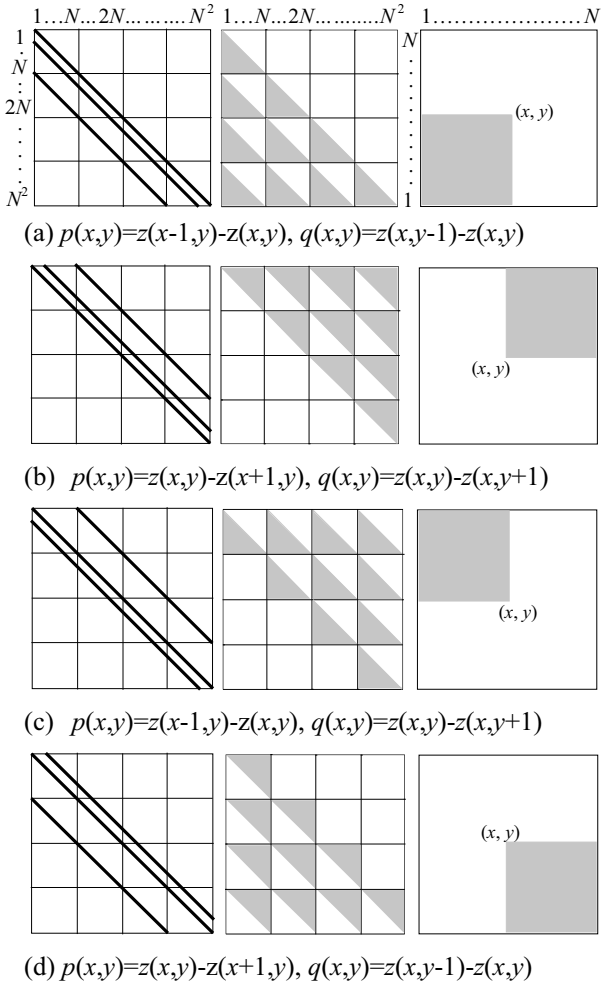


Fig. 1 Schematics of matrices of **g** (left column), their inverse matrices (middle) and integrating regions to get the depth value at (x,y) for the four approximations of p and q.

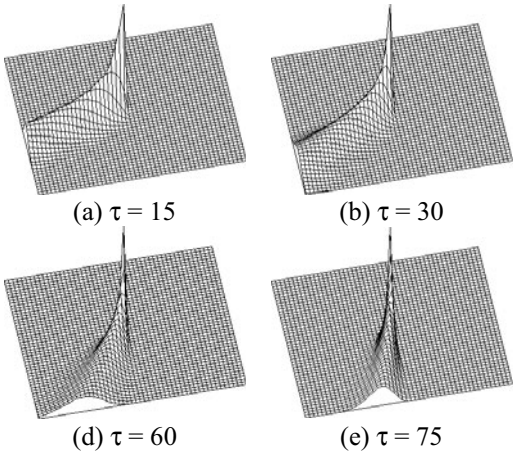
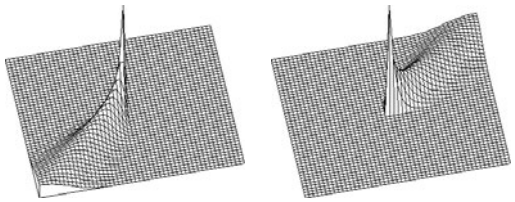
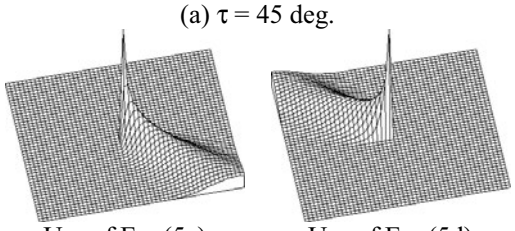


Fig. 2 Distributions of $\mathbf{g}^{-1}$ for four tilt angles of the illuminant vector, where Eq. (5a) is used as the approximation, and the parameters used are (x,y) = (25,25), n=1, N=50 and σ=45 deg.



Use of Eq. (5a) Use of Eq. (5b)

(a) τ = 45 deg.



Use of Eq. (5c) Use of Eq. (5d)

(b) τ = 135 deg.

Fig. 3 Available two distributions of $\mathbf{g}^{-1}$ for (a) **S**= (5,5,7) and (b) **S**=(-5,5,7), where n=1, (x,y)=(25,25), N=50 and σ=45 deg.

This means that it becomes possible to reconstruct shapes regardless of the tilt angle of the **S**. This also implies that using more than a single approximation may be useful to get better shapes. That is, distortions may accumulate through the integration process, but they may be reduced through averaging two shapes available. For this purpose Eq. (9) is modified as follows:

We estimate in each iteration two shapes using the two approximations and average them using appropriate weights as:

$$-\mathbf{f}_a^{(n-1)} = D\mathbf{g}_a^{(n-1)}(\mathbf{z}_a^{(n)} - \mathbf{z}_m^{(n-1)}) \text{ ,} a=1,2,\ n=1,2,\ldots \quad (11)$$

$$\mathbf{z}_m^{(n)} = w_1\mathbf{z}_1^{(n)} + w_2\mathbf{z}_2^{(n)} \text{ , } n=1,2,\ldots \quad (12)$$

where D≥1 is a de-accelerating factor to make the shape change smoothly. In the case of τ = 45 deg, taking into account the functions of $\mathbf{g}^{-1}$ in Fig. 3, the shape $\mathbf{z}_1$ may be more accurate for smaller value of x+y and so may be $\mathbf{z}_2$ for larger value of x+y. So, w_1 and w_2 may be given by

$$w_1(x,y) = 1 - \frac{x+y-2}{2(N-1)}, \quad w_2(x,y) = \frac{x+y-2}{2(N-1)} \quad (13)$$

In general, weights that vary along the direction of τ may be appropriate.

In the previous algorithm the shape ends up having significant accumulated distortions along some of the boundaries. This makes the imposition of some boundary conditions inappropriate and unworkable. On the other hand, in the improved algorithm the obtainable shape may have much reduced distortions along the boundaries. This may make the imposition workable. In this paper we impose z=0 for the boundary with a constant image intensity, p=0 for the vertical boundary with varying image, and q=0 for the horizontal boundary with varying image intensity.

5 Computer Experiments

Four shapes were used to investigate the method. They are a semi-sphere, a computer mouse, a ring with a center part, and the Mozart sculpture, among which the shape of the computer mouse was measured with a laser range scanner. 50x50 pixels shading images of them were computationally generated assuming that the surfaces are *Lambertian*.

First the method being presented is compared with a reconstruction method that uses three shading images of different **S**'s, to explain how our method works. In the latter method, local surface normals of the shape are determined from three images of the semi-sphere for **S** = (-5,-5,7), (5,-5,7) and (0,1,1), and they are integrated from one of the four corners, where no boundary conditions are applied. In our method, one of the two shading images for **S** = (-5,-5,7) and (5,-5,7) is used and the shape is estimated using the knowledge of the illuminant vector and the appropriate approximation, in which we give p=0 or q=0 along two of the four boundaries where they cannot be calculated from z(x,y). The results in Fig. 4 show that the shapes obtained by our method are relatively close to those by the method using the three **S**'s. So, averaging the two available shapes together with the imposition of the appropriate boundary condition appears to be crucial to get good shapes in our method. Fig. 4 also shows that the shape reconstructed using a single approximation is very different from the original one along some of the boundaries. In this case imposing the appropriate boundary condition on those boundaries is inappropriate.

The improved algorithm was tested using the images, which are summarized in Fig. 5. The boundary conditions used are z=0 along all the four edge lines for the images of semi-sphere, mouse and ring, and a combination of q=0 for the bottom edge line and z=0 for the other three lines for the image of Mozart. The iteration tends to converge before divergence which tends to occur when the minimal value of the eigenvalues, $\partial f_{x,y}/\partial z_{x,y}$, is less than 0.1. So the iteration was stopped when the change in shape with the number of iterations takes a local minimum or when the minimal value dips below 0.1.

The results in Fig. 5, where the original and estimated shapes are compared, show that the obtained shapes, on the whole, are relatively in good agreement with the original ones. We notice there that the estimated shapes still have moderate distortions and that they appear more serious for the Mozart image than for the others, where the Mozart image has one boundary where its image varies. We also notice that the method being presented lacks spatial resolution a little, that different shading images of the same object give rise to somewhat different shapes, that there

remain wave-like shape components, that is, the flat shape parts are not always reconstructed flat, and that there exist strong robustness to noise in the image due to the integration and averaging processes in the method.

6 Conclusions

We made clear on the shape from shading algorithm using Jacobi's iterative method that the four different simplest approximations for the surface normal of the shape give four different and asymmetrical reconstruction functions of shape, and we improved the previous algorithm so as to reconstruct shapes with multiple approximations and so as to impose appropriate boundary conditions, to get more accurate shapes and enhance the applicability. The improved method was verified by computer experiments.

References

[1] B.K.P. Horn, "Obtaining Shape from Shading Information", in *The Psychology of Computer Vision*, P.H.Winston (Ed.), McGraw Hill, New York, p. 115-155 (1975).

[2] B.K.P. Horn and M.J. Brooks, *Shape from Shading*, MIT Press, Cambridge, MA (1989).

[3] R. Zhang, P. Tsai, J.E. Cryer, and M. Shah, "Shape from Shading: A Survey," *IEEE Trans. PAMI*, vol. 21, no. 8, pp. 690-705 (1999).

[4] P. Pentland, "Local shading analysis," *IEEE Trans. PAMI*, vol. 6, no. 2, pp. 170-187 (1984).

[5] Q. Zheng and R. Chellappa, "Estimation of Illuminant Direction, Albedo, and Shape from Shading," *IEEE Trans. PAMI*, vol. 13, no. 7, pp. 680-702 (1991).

[6] P. L. Worthington and E. R. Hancock, "New Constraints on Data-Closeness and Needle Map Consistency for Shape-from- Shading," *IEEE Trans. PAMI*, vol.21, no. 12, pp. 1250-1267 (1999).

[7] P. S. Tsai and M. Shah, "Shape from Shading Using Linear Approximation," *J. Imaging and Vision Computing*, vol. 12, no. 8, pp. 487-498 (1994).

[8] A. Pentland, "Shape Information from Shading: A Theory about Human Perception," *Proc. ICCV*, p. 404-413 (1988).

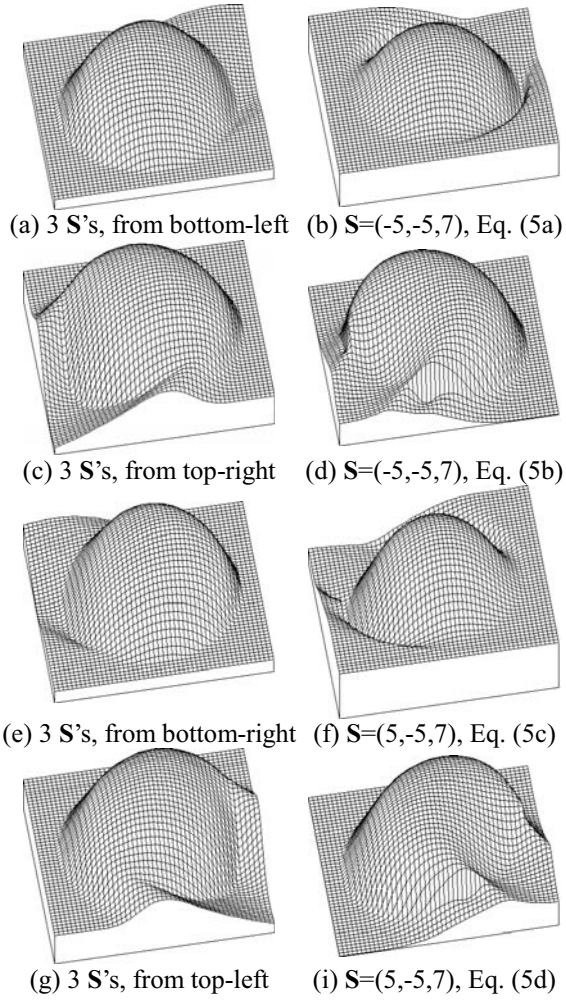
[9] M. Bichsel and A. Pentland, "A Simple Algorithm for Shape from Shading," *Proc. CVPR*, p. 459-465 (1992).

[10] R. Kimmel and A.M. Bruckstein, "Tracking Level Sets by Level Sets: A Method for Solving Shape from Shading Problem," *CVIU*, vol. 62, no. 1, pp. 47-58 (1995).

[11] R. Klette, A. Koschan and K. Schluns, *Three-Dimensional Data from Images*, Springer, Singapore, Chap. 8 (1998).

[12] O. Ikeda, "A Novel Shape-From-Shading Algorithm Using Jacobi Iterative Method and Bi-Directional Estimation," *Proc. IASTED CGIM*, p. 56-61 (2002).

[13] R.J. Woodham, "Photometric Method for Determining Surface Orientation from Multiple Images," Chap. 17 in *Shape from Shading*, MIT Press, Cambridge, MA (1989).



(a) 3 **S**'s, from bottom-left (b) **S**=(-5,-5,7), Eq. (5a)

(c) 3 **S**'s, from top-right (d) **S**=(-5,-5,7), Eq. (5b)

(e) 3 **S**'s, from bottom-right (f) **S**=(5,-5,7), Eq. (5c)

(g) 3 **S**'s, from top-left (i) **S**=(5,-5,7), Eq. (5d)

Fig. 4 Comparison of shapes obtained by our method using a single approximation (right) with those obtained by integrating local surface normal starting from one of the four corners for the case of using three shading images (left).

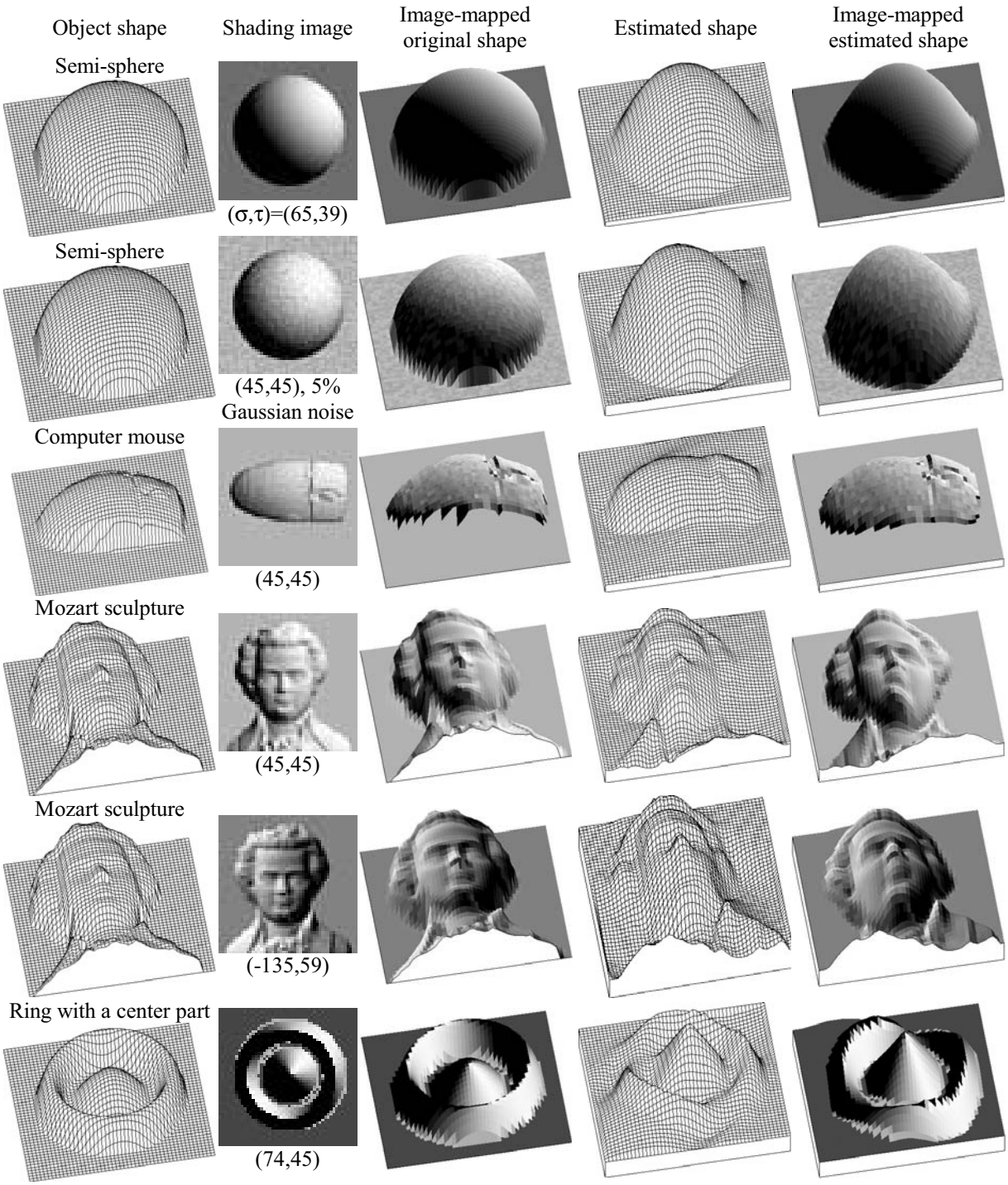


Fig. 5 Reconstructed shapes from synthesized shading images with the improved algorithm. The slant and tilt angles, σ are τ, respectively, are in degrees, the two texture-mapped shapes of the ring object are viewed from the right, and Gaussian noise is added to the shading image of semi-sphere on the second row.