USC-SIPI REPORT #172

Shape from Shading with A Linear Triangular Element Surface Model

by

Kyoung Mu Lee and C.-C. Jay Kuo

February 1991

Signal and Image Processing Institute
UNIVERSITY OF SOUTHERN CALIFORNIA
Viterbi School of Engineering
Department of Electrical Engineering-Systems
3740 McClintock Avenue, Suite 400
Los Angeles, CA 90089-2564 U.S.A.

Shape from Shading with A Linear Triangular Element Surface Model*

Kyoung Mu Lee† and C.-C. Jay Kuo †

February 22, 1991

Abstract

We propose to combine a triangular element surface model with the linearized reflectance map to formulate the shape from shading problem in this research. The key idea is to approximate a smooth surface by the union of triangular surface patches called triangular elements, and express the approximating surface as a linear combination of a set of nodal basis functions. Since the surface normal of a triangular element is uniquely determined by the heights of its three vertices (or nodes), the image brightness can be directly related to the nodal heights via a linearized reflectance map. The surface height can therefore be determined by minimizing a quadratic cost functional corresponding to the squares of the brightness error and solved effectively with the multigrid computational technique. The proposed method does not require any integrability constraint or artificial assumptions on boundary conditions. Simulation results for several synthetic and real images are demonstrated to show the performance and efficiency of our new method.

1 Introduction

The shape from shading (SFS) problem is one of the early computational vision problems. It extracts the 3-D shape information or, more precisely, the surface height $z(x, y)$ from a single 2-D shading image, and can be therefore viewed as an inversion problem of the image formation process. This problem was first formulated by Horn [9], [10] and has been studied intensively for the last two decades [3], [5], [6], [12], [13], [14], [15], [19], [20], [22], [23], [26], [27], [33], [35].

*This work was supported by the USC Faculty Research and Innovation Fund and a National Science Foundation Research Initiation Award.

†The authors are with the Signal and Image Processing Institute and the Department of Electrical Engineering-Systems, University of Southern California, Los Angeles, California 90089-0272. E-mail: kyoungmu@sipi.usc.edu and cckuo@sipi.usc.edu.

Most research work is based on the variational approach [5], [12], [13], [15], [20], [26], in which a surface orientation field, characterized by its slopes $p(x,y) = \partial z(x,y)/\partial x$ and $q(x,y) = \partial z(x,y)/\partial y$, is determined to minimize a certain cost functional of the brightness error. To achieve this objective, one usually uses the calculus of variations [4] to derive a set of coupled Euler equations involving $p(x,y)$ and $q(x,y)$ from the given functional. The Euler equations are then discretized into a system of difference equations and solved for $p(x,y)$ and $q(x,y)$ numerically by iterative methods. Finally, the surface height $z(x,y)$ is reconstructed by integrating $p(x,y)$ and $q(x,y)$. However, a straightforward implementation of this approach does not work properly due to the nonintegrability of computed $p(x,y)$ and $q(x,y)$ and the ill-posed nature of the SFS problem. Methods for enforcing integrability have been studied by Frankot and Chellappa [5] and Horn [12]. The ill-posedness of the SFS problem has often been handled by introducing a regularization technique, which assumes that the reconstructed surface is smooth [1], [15], [24]. As a result, it is common to impose the integrability and the smoothness constraints on the surface by adding several penalty terms in the cost functional weighted by factors called the Lagrangian multipliers. We refer to [12] for a thorough survey of the development of the variational approach until 1989.

Even with modifications, there still exist two major difficulties in the variational approach. First, we are usually led to extremely complicated first-oder nonlinear partial differential equations. It is difficult to find convergent iterative algorithms for their solution [12], [13], [20]. Even if a converging algorithm is available, it is still not easy to characterize and to accelerate its convergence rate. The number of iterations grows linearly with the image size N was reported in [12], and the total computational complexity is therefore proportional to $O(N^2)$. Second, since the problem of minimizing a functional is transformed to the solution of a set of partial differential equations, appropriate boundary conditions are needed. No boundary boundary conditions tend to lead to the ambiguities in the solution and the instabilities in the algorithm [12]. Although the natural boundary conditions are often used, they do not always give satisfactory results. How to impose appropriate boundary conditions is still an open question.

A very different approach for the SFS problem was recently proposed by Pentland [23]. It relates image brightness to surface height $z(x,y)$ in closed form with a linearized reflectance map in the Fourier transform domain. The resulting algorithm is a non-optimization and non-iterative one. Since the surface height $z(x,y)$, rather than its slopes $p(x,y)$ and $q(x,y)$,

is computed directly from the algorithm, there is no integrability problem in this approach. However, it has some shortcomings. The reconstructed surface is less accurate than that obtained by the variational approach, since it is one-step algorithm which uses only the global linearization of the reflectance map around the origin, and known physical information cannot be easily incorporated in its formulation. Besides, this approach is sensitive to the noise.

In this research, we propose a new approach to solve the SFS problem. On one hand, it is similar to Horn's approach in the sense that the formulation is based on the minimization of a cost functional of brightness error. On the other hand, it is similar to Pentland's approach in the sense that the surface height is computed directly so that there is no integrability problem. This new approach has several unique features. First, the minimization problem is discretized and solved directly. Since we do not derive and solve Euler equations with the variational principle, no boundary conditions are needed in our solution procedure. Second, by linearizing the reflectance map, we can greatly simplify the problem to be a quadratic functional minimization problem. It is much easier to understand the existence and uniqueness of the solution and to find effective convergent iterative algorithms for its computation from this viewpoint.

The basic idea of our approach is to approximate a smooth surface by the union of triangular surface patches called *triangular elements* and to express the approximating surface as a linear combination of a set of nodal basis functions of compact support. Since the surface normal of a triangular element is linearly determined by the heights of its three vertices (or nodes), only the nodal heights z_i are used as variables and the image brightness can be related directly to the nodal heights via the reflectance map. Furthermore, by using the linear approximation of the reflectance map, we can express the image brightness as a linear functional of nodal heights. By defining the cost functional to be the square of the brightness error, we are led to a quadratic functional minimization problem parameterized by nodal heights. Clearly, the surface height can be obtained without any additional integrability constraint or artificial assumptions on the boundary conditions. Besides, the optimization problem is equivalent to the solution of a large sparse linear system of equations to which very efficient multigrid computational algorithm can be applied. Empirically, we find that the number of iterations for our algorithm is independent of the image size N so that the computational complexity is only $O(N)$.

This paper is organized as follows. In Section 2, we describe the triangular element surface model and the image formation model, and formulate the SFS problem as a finite-dimensional optimization problem which minimizes a quadratic cost functional consisting of the squares of the brightness error. This formulation requires the knowledge of a stiffness matrix and a load vector, and their construction is discussed in Section 3. It is shown that the elements of the stiffness matrix and load vector only depend on the coefficients of the linearized reflectance map. We develop a successive linearization scheme, in which the linearization of the reflectance map is performed with respect to every local gradient point of the triangular patch obtained from the previous iteration. The stiffness matrix is shown to be singular in Section 4, which is consistent with the ill-posed nature of the SFS problem. In Section 5, we impose the smooth surface constraint by adding one extra cost term to regularize the SFS problem so that the uniqueness of the minimum can be easily guaranteed. The implementation details and experimental results are provided, respectively, in Sections 6 and 7.

2 Shape-From-Shading Problem Formulation

Consider the approximation of a smooth surface with a union of triangular surface patches [16], [25]. The approximating surface can be expressed as a linear combination of basis functions with local compact support known as the finite triangular elements. Based on such a triangular element surface model, the orientation (i.e. the surface normal vector) of a triangular patch can be determined by the heights of its three vertices (or nodes) in a linear functional form. Combining this surface model with the reflectance map image formation model which relates the image intensity to the surface normal vector, we can express the image intensity directly in terms of the nodal heights of triangular elements. Thus, unlike the conventional SFS problem formulation in which surface normal vectors and/or heights are treated as independent variables, we use only the nodal heights of triangular elements as independent variables, which is detailed below.

2.1 Triangular Element Surface Model

Consider a uniform triangulation of a square domain Ω with spacing h. As illustrated in Fig. 1, the domain Ω is divided into a set of nonoverlapping triangles T_i,

$$\Omega = \bigcup_{1 \le i \le M_t} T_i, \tag{2.1}$$

where M_t is the number of triangles. Let V_h denote the set of continuous piecewise linear surfaces defined on Ω and linear over all triangles T_i. The nodal basis function $\phi_i \in V_h$ are the function which takes unity at ith node and zero at other nodes (see Fig. 2). It is easy to see that any $z \in V_h$ can be represented as the linear combination of nodal basis functions,

$$z(x,y) = \sum_{i=1}^{M_n} z_i \phi_i(x,y), \tag{2.2}$$

where z_i is the value of $z(x,y)$ at the ith node and M_n is the number of nodal basis functions.

The gradient (p,q) of the surface $z(x,y)$ can be computed as

$$p(x,y) = \frac{\partial z(x,y)}{\partial x} = \sum_{i=1}^{M_n} z_i \frac{\partial \phi_i(x,y)}{\partial x}, \tag{2.3}$$

and

$$q(x,y) = \frac{\partial z(x,y)}{\partial y} = \sum_{i=1}^{M_n} z_i \frac{\partial \phi_i(x,y)}{\partial y}. \tag{2.4}$$

Since the partial derivatives of $\phi_i(x,y)$ with respect to x and y are simply some constants so that $p(x,y)$ and $q(x,y)$ are linear functions of nodal heights z_i. Note also that $p(x,y)$ and $q(x,y)$ are piecewise constant functions on Ω and they take constant values on each triangular domain T_i.

2.2 Reflectance Map Image Formation Model

Generally speaking, the image formation is a very complex process which depends on the projection angle, the surface reflection property and orientation, the light source type, and other factors. In formulating the SFS problem, it is often to adopt a simplified model which assumes the orthographic projection, the Lambertian surface, and a distant single point light source so that the irradiance or brightness E at a given point (x,y) is primarily due to the surface orientation at that point. This relationship is described by the image irradiance equation [11], [14],

$$E(x,y) = R(p,q), \tag{2.5}$$

where R is called the reflectance map function. The form of R is usually chosen as

$$R(p,q) = \begin{cases} \eta \frac{1+p_s p+q_s q}{\sqrt{(1+p^2+q^2)}\sqrt{(1+p_s^2+q_s^2)}}, & 1+p_s p+q_s q \geq 0, \\ 0, & 1+p_s p+q_s q < 0, \end{cases} \tag{2.6a}$$

or equivalently,

$$R(p,q) = \begin{cases} \eta \frac{K}{\sqrt{(1+p^2+q^2)}}, & K \geq 0, \\ 0, & K < 0, \end{cases} \qquad K = -p\cos\tau\sin\sigma - q\sin\tau\sin\sigma + \cos\sigma, \tag{2.6b}$$

where η is the albedo of the surface, (p,q) the gradient of the surface at point (x,y), $(-p_s,-q_s,1)$ the illumination direction pointing toward the light source, and τ and σ are the *tilt* and *slant* angles that the illumination direction makes with x and z axis, respectively. The Lambertian surface is a rough and matte surface such that it scatters the incident light uniformly in all directions. Consequently, the image intensity of a small surface is not dependent on the viewer's direction but only on the orientation of the surface.

As given in (2.6), the reflectance map R is a nonlinear function which can be depicted as nested contours in the gradient space (p,q). To remove the nonlinearity, techniques based on the linear approximation of the reflectance map have been recently proposed [12], [23]. That is, we take the Taylor series expansion of $R(p,q)$ about a certain reference point (p_0,q_0) through the first-order term,

$$R(p,q) \approx R(p_0,q_0) + (p-p_0)\frac{\partial R(p,q)}{\partial p}\Big|_{(p_0,q_0)} + (q-q_0)\frac{\partial R(p,q)}{\partial q}\Big|_{(p_0,q_0)} \tag{2.7}$$

The reference point (p_0,q_0) can be either fixed or varying for different values of (p,q).

2.3 Image Formation on Modeled Surfaces

By substituting (2.3) and (2.4) into (2.7), we have

$$R(p,q) \approx \alpha p + \beta q + \gamma = \sum_{i=1}^{M_n} \Phi_i z_i + \gamma, \tag{2.8}$$

where

$$\Phi_i(x,y) = \alpha\frac{\partial\phi_i(x,y)}{\partial x} + \beta\frac{\partial\phi_i(x,y)}{\partial y}, \qquad \gamma = R(p_0,q_0) - \alpha p_0 - \beta q_0, \tag{2.8a}$$

and where

$$\alpha = \frac{\partial R(p,q)}{\partial p}\Big|_{(p_0,q_0)}, \qquad \beta = \frac{\partial R(p,q)}{\partial q}\Big|_{(p_0,q_0)}. \tag{2.8b}$$

Thus, combining (2.5) and (2.8). one can establish a linear relationship between the image brightness $E(x,y)$ and nodal values z_i, $1 \leq i \leq M_n$,

$$E = \alpha p + \beta q + \gamma = \sum_{i=1}^{M_n} \Phi_i z_i + \gamma. \tag{2.9}$$

To estimate the nodal heights z_i based on the shading information, we consider the cost functional

$$\mathcal{E}_b = \iint_\Omega (E_o - E_r)^2 \, dx \, dy, \tag{2.10}$$

where E_o is the observed image intensity, E_r is the image intensity formed by the reconstructed surface via (2.9), and the subscript b denote the cost due to the brightness error. By substituting (2.9) into (2.10), we obtain

$$\begin{aligned}
\mathcal{E}_b &= \iint_\Omega [E_o - (\sum_{i=1}^{M_n} \Phi_i z_i + \gamma)]^2 \, dx \, dy \\
&= \iint_\Omega [\sum_{i=1}^{M_n}\sum_{j=1}^{M_n} \Phi_i \Phi_j z_i z_j - 2(E_o - \gamma)\sum_{i=1}^{M_n} \Phi_i z_i + (E_o - \gamma)^2] \, dx \, dy \\
&= \frac{1}{2}\sum_{i=1}^{M_n}\sum_{j=1}^{M_n}(2\iint_\Omega \Phi_i \Phi_j \, dx \, dy) z_i z_j - \sum_{i=1}^{M_n}[2\iint_\Omega (E_o - \gamma)\Phi_i \, dx \, dy] z_i \\
&\quad + \iint_\Omega (E_o - \gamma)^2 \, dx \, dy \\
&= \frac{1}{2}\mathbf{z}^T \mathbf{A}\mathbf{z} - \mathbf{b}^T \mathbf{z} + \mathbf{c}.
\end{aligned} \tag{2.11}$$

Thus, the elements $a_{i,j}$ and b_i of matrix $\mathbf{A}$ and vector $\mathbf{b}$ can be determined, respectively, as

$$a_{i,j} = 2\iint_\Omega \Phi_i \Phi_j \, dx \, dy, \qquad b_i = 2\iint_\Omega (E_o - \gamma)\Phi_i \, dx \, dy \qquad 1 \leq i,j \leq M_n. \tag{2.12}$$

Following the finite-element terminology, $\mathbf{A}$ is called the *stiffness* matrix and $\mathbf{b}$ the *load* vector. Our objective is to determine the nodal height vector $\mathbf{z}$ that minimizes the quadratic functional (2.11). Note that the minimization problem can also be formulated as the solution of a system of linear equations

$$\mathbf{A}\mathbf{z} = \mathbf{b}. \tag{2.13}$$

It is obvious that the quadratic functional gives a unique minimum only when the *stiffness* matrix $\mathbf{A}$ is positive-definite. The property of $\mathbf{A}$ will be discussed in Section 4.

It is worthwhile to emphasize one major difference between our approach and the variational approach, namely, the role played by the discretization procedure. The discretization is introduced by using the triangular element surface model and is performed in the very beginning of the SFS problem formulation in our approach. As a result, we are led to a *finite*-dimensional cost functional suitable for further minimization. In contrast, the variational approach use the calculus of variations to derive the partial differential equations (PDEs) from an *infinite*-dimensional cost functional. The discretization is then applied to PDEs for their solution and primarily serves as a computational tool. The triangular element surface model provides some advantages. Since the surface heights are recovered directly, no additional integrability constraint is required. Besides, there is no need to impose boundary conditions for PDEs.

3 Construction of Stiffness Matrices and Load Vectors

In this section, we will use (2.12) to derive the expression of the elements of stiffness matrix **A** and load vector **b** in terms of the coefficients α, β and γ in (2.8). Note that $a_{ij} = 0$ if i and j are not the neighboring nodal points, since either $\Phi_i(x,y)$ or $\Phi_j(x,y)$ is zero for $(x,y) \in \Omega$. Thus, we only have to determine $a_{i,j}$ for j is equal to either the node i or one of its six neighboring nodes $j_1, \cdots, j_6$ as shown in Fig. 2. We use $T_k, \ldots, T_p$ to denote the 6 triangular domains surrounding node i, and $S_k, \ldots, S_p$ the triangular surface patches defined on $T_k, \ldots, T_p$. One can determine the gradients of the nodal basis function $\phi_i(x,y)$ in triangles $T_k, \ldots, T_p$ from Fig. 2. They are

$$(\frac{\partial \phi_i(x,y)}{\partial x}, \frac{\partial \phi_i(x,y)}{\partial y}) = \begin{cases} (-h^{-1}, 0), & \text{for } (x,y) \in T_k, \\ (0, -h^{-1}), & \text{for } (x,y) \in T_l, \\ (h^{-1}, -h^{-1}), & \text{for } (x,y) \in T_m, \\ (h^{-1}, 0), & \text{for } (x,y) \in T_n, \\ (0, h^{-1}), & \text{for } (x,y) \in T_o, \\ (-h^{-1}, h^{-1}), & \text{for } (x,y) \in T_p, \end{cases} \tag{3.1}$$

where h is the spacing.

3.1 Linearization of Reflectance Map

Consider the linearization of the reflectance map (2.8) by using different reference gradients for different triangles. Let (p_{0i}, q_{0i}) be the reference gradient and α_i, β_i and γ_i be the

coefficients of the linearized reflectance map on triangular domain T_i. By using (2.12), we can express the elements of $\mathbf{A}$ and $\mathbf{b}$ in terms of α_i, β_i and γ_i in a straightforward way.

The diagonal element $a_{i,i}$ of $\mathbf{A}$ can be determined as

$$\begin{aligned} a_{i,i} &= 2\iint_{\Omega} \Phi_i^2 \, dx\, dy \\ &= 2[\iint_{T_k} (-\alpha_k h^{-1})^2 \, dx\, dy + \iint_{T_l} (-\beta_l h^{-1})^2 \, dx\, dy + \iint_{T_m} (\alpha_m h^{-1} - \beta_m h^{-1})^2 \, dx\, dy \\ &\quad + \iint_{T_n} (\alpha_n h^{-1})^2 \, dx\, dy + \iint_{T_o} (\beta_o h^{-1})^2 \, dx\, dy + \iint_{T_p} (-\alpha_p h^{-1} + \beta_p h^{-1})^2 \, dx\, dy] \\ &= \frac{2}{h^2}[\alpha_k^2 \mathcal{S}_k + \beta_l^2 \mathcal{S}_l + (\alpha_m - \beta_m)^2 \mathcal{S}_m + \alpha_n^2 \mathcal{S}_n + \beta_o^2 \mathcal{S}_o + (\beta_p - \alpha_p)^2 \mathcal{S}_p] \\ &= \alpha_k^2 + \beta_l^2 + (\alpha_m - \beta_m)^2 + \alpha_n^2 + \beta_o^2 + (\alpha_p - \beta_p)^2, \end{aligned}$$

where $\alpha_k, \cdots, \alpha_p$ and $\beta_k, \cdots, \beta_p$ are coefficients of linearized reflectance maps on triangles $T_k, \cdots, T_p$, and $\mathcal{S}_k, \cdots, \mathcal{S}_p$ are the areas of the triangles which all equal to $h^2/2$.

The off-diagonal element a_{i,j_1} of $\mathbf{A}$ can be derived as

$$\begin{aligned} a_{i,j_1} &= 2\iint_{\Omega} \Phi_i \Phi_{j_1} \, dx\, dy \\ &= 2(\iint_{T_k} \Phi_i \Phi_{j_1} \, dx\, dy + \iint_{T_p} \Phi_i \Phi_{j_1} \, dx\, dy) \\ &= 2[\iint_{T_k} (-\alpha_k h^{-1})(\alpha_k h^{-1} - \beta_k h^{-1}) \, dx\, dy + \iint_{T_p} (-\alpha_p h^{-1} + \beta_p h^{-1})(\alpha_p h^{-1}) \, dx\, dy] \\ &= \frac{2}{h^2}[\alpha_k(\beta_k - \alpha_k)\mathcal{S}_k + \alpha_p(\beta_p - \alpha_p)\mathcal{S}_p] \\ &= \alpha_k(\beta_k - \alpha_k) + \alpha_p(\beta_p - \alpha_p). \end{aligned}$$

Other off-diagonal elements $a_{i,j_2} \cdots a_{i,j_6}$ can be similarly determined. The results are summarized in Table 1.

$a_{i,i}$	$\alpha_k^2 + \beta_l^2 + (\alpha_m - \beta_m)^2 + \alpha_n^2 + \beta_o^2 + (\alpha_p - \beta_p)^2$
a_{i,j_1}	$\alpha_k(\beta_k - \alpha_k) + \alpha_p(\beta_p - \alpha_p)$
a_{i,j_2}	$-(\alpha_k\beta_k + \alpha_l\beta_l)$
a_{i,j_3}	$\beta_l(\alpha_l - \beta_l) + \beta_m(\alpha_m - \beta_m)$
a_{i,j_4}	$\alpha_m(\beta_m - \alpha_m) + \alpha_n(\beta_n - \alpha_n)$
a_{i,j_5}	$-(\alpha_n\beta_n + \alpha_o\beta_o)$
a_{i,j_6}	$\beta_o(\alpha_o - \beta_o) + \beta_p(\alpha_p - \beta_p)$

Table 1: elements of stiffness matrix $\mathbf{A}$

Finally, the element b_i of the load vector $\mathbf{b}$ can be obtained by

$$b_i = 2\iint_{\Omega} (E_o - \gamma)\Phi_i \, dx\, dy$$

$$
\begin{aligned}
&= \frac{2}{h}[-\alpha_k \iint_{T_k}(E_o-\gamma_k)\,dx\,dy - \beta_l \iint_{T_l}(E_o-\gamma_l)\,dx\,dy \\
&\quad +(\alpha_m-\beta_m)\iint_{T_m}(E_o-\gamma_m)\,dx\,dy + \alpha_n \iint_{T_n}(E_o-\gamma_n)\,dx\,dy \\
&\quad +\beta_o \iint_{T_o}(E_o-\gamma_o)\,dx\,dy + (\beta_p-\alpha_p)\iint_{T_p}(E_o-\gamma_p)\,dx\,dy] \\
&= \frac{2}{h}[-\alpha_k(\overline{E}_k-\gamma_k)\mathcal{S}_k - \beta_l(\overline{E}_l-\gamma_l)\mathcal{S}_l + (\alpha_m-\beta_m)(\overline{E}_m-\gamma_m)\mathcal{S}_m \\
&\quad +\alpha_n(\overline{E}_n-\gamma_n)\mathcal{S}_n + \beta_o(\overline{E}_o-\gamma_o)\mathcal{S}_o + (\beta_p-\alpha_p)(\overline{E}_p-\gamma_p)\mathcal{S}_p] \\
&= h[-\alpha_k(\overline{E}_k-\gamma_k) - \beta_l(\overline{E}_l-\gamma_l) + (\alpha_m-\beta_m)(\overline{E}_m-\gamma_m) \\
&\quad +\alpha_n(\overline{E}_n-\gamma_n) + \beta_o(\overline{E}_o-\gamma_o) + (\beta_p-\alpha_p)(\overline{E}_p-\gamma_p)], \qquad (3.2)
\end{aligned}
$$

where $\overline{E}_k, \cdots, \overline{E}_p$ denote the average brightness on the triangles $T_k, \cdots, T_p$.

If node i lies on the boundary of the image, the elements $a_{i,j}$ and b_i can also be determined from the above equations by setting α and β zero on the triangles outside the image.

Example: We use a simple but typical example to illustrate the construction of the stiffness matrix A. As depicted in Fig. 3, we consider a square partitioned into 8 triangular domains with 9 nodal points where the triangles and nodes are ordered in a rowwise fashion. Then, A is a 9 x 9 matrix of the form

$$
\mathbf{A} = \begin{bmatrix}
a_{11} & a_{12} & 0 & a_{14} & a_{15} & 0 & 0 & 0 & 0 \\
a_{12} & a_{22} & a_{23} & 0 & a_{25} & a_{26} & 0 & 0 & 0 \\
0 & a_{23} & a_{33} & 0 & 0 & a_{36} & 0 & 0 & 0 \\
a_{14} & 0 & 0 & a_{44} & a_{45} & 0 & a_{47} & a_{48} & 0 \\
a_{15} & a_{25} & 0 & a_{45} & a_{55} & a_{56} & 0 & a_{58} & a_{59} \\
0 & a_{26} & a_{36} & 0 & a_{56} & a_{66} & 0 & 0 & a_{69} \\
0 & 0 & 0 & a_{47} & 0 & 0 & a_{77} & a_{78} & 0 \\
0 & 0 & 0 & a_{48} & a_{58} & 0 & a_{78} & a_{88} & a_{89} \\
0 & 0 & 0 & 0 & a_{59} & a_{69} & 0 & a_{89} & a_{99}
\end{bmatrix},
$$

where the elements can be determined from Table 1. We list some values below as example,

$$
\begin{aligned}
a_{11} &= \alpha_2^2+\beta_1^2, & a_{22} &= \alpha_4^2+\beta_3^2+(\alpha_2-\beta_2)^2, \\
a_{12} &= \alpha_2(\beta_2-\alpha_2), & a_{55} &= \alpha_8^2+\beta_7^2+(\alpha_6-\beta_6)^2+\alpha_1^2+\beta_2^2+(\alpha_3-\beta_3)^2, \\
a_{15} &= -(\alpha_1\beta_1+\alpha_2\beta_2), & a_{59} &= -(\alpha_7\beta_7+\alpha_8\beta_8), \\
a_{45} &= \alpha_1(\beta_1-\alpha_1)+\alpha_6(\beta_6-\alpha_6), & a_{56} &= \alpha_3(\beta_3-\alpha_3)+\alpha_8(\beta_8-\alpha_8), \\
a_{25} &= \beta_2(\alpha_2-\beta_2)+\beta_3(\alpha_3-\beta_3), & a_{58} &= \beta_6(\alpha_6-\beta_6)+\beta_7(\alpha_7-\beta_7),
\end{aligned}
$$

where α_i and β_i are coefficients of linearized reflectance maps on triangles T_i, $1 \le i \le 8$. Similarly, one can use (3.2) to compute the elements of the load vector. □.

Note that node 5 in the above example is a typical case for nodes not lying on the boundaries. We can represent the corresponding nodal operator by a 7-point stencil as illustrated in Fig. 4.

3.2 Successive Linearization Scheme

One special case of the linearization scheme is to choose the reference (p_0, q_0) to be the same for all triangular surface patches. We call it the *global* linearization and denote the corresponding stiffness matrix by $\mathbf{A}_g$. The elements $a_{i,j}$ of $\mathbf{A}_g$ can be further simplified and summarized in Table 2. Since the values of α, β and γ are the same for all triangular domains, their subscripts are dropped.

$a_{i,i}$	$2(\alpha^2 - \alpha\beta + \beta^2)$
a_{i,j_1}, a_{i,j_4}	$2\alpha(\beta - \alpha)$
a_{i,j_2}, a_{i,j_5}	$-2\alpha\beta$
a_{i,j_3}, a_{i,j_6}	$2\beta(\alpha - \beta)$

Table 2: Coefficients of global stiffness matrix $\mathbf{A}_g$

Suppose that we do not have any a priori knowledge of the reconstructed surface, we may set all initial nodal values zero and proceed as follows.

Algorithm I: The basic SFS algorithm

Initialization $(k = 0)$

Set the reference gradient (p_{0i}^0, q_{0i}^0) at every triangle T_i to be $(0, 0)$, and construct the global stiffness matrix $\mathbf{A}_g^0$ and the load vector $\mathbf{b}^0$. Solve $\mathbf{A}_g^0 \mathbf{z}^0 = \mathbf{b}^0$ for nodal values $\mathbf{z}^0$.

Iterations $(k = 1, 2, \cdots)$

Set the reference gradient (p_{0i}^k, q_{0i}^k) at triangle T_i to be the local gradient determined by nodal values $\mathbf{z}^{k-1}$ and construct the corresponding stiffness matrix $\mathbf{A}_l^k$ and load vector $\mathbf{b}^k$, where the subscript l denotes local linearization. Solve $\mathbf{A}_l^k \mathbf{z}^k = \mathbf{b}^k$ for nodal values $\mathbf{z}^k$.
If $||\mathbf{z}^k - \mathbf{z}^{k-1}|| < \epsilon$, where ϵ is a predefined small quantity, then $\mathbf{z}^k$ is the desired solution. Otherwise, go to the next iteration.

In the above algorithm, we use a *successive linearization* scheme, where the linearization of the reflectance map is performed based on the local gradients obtained from the previous iteration. The motivation is simple. There are two kinds of error introduced in our SFS formulation: the surface approximation error introduced by the triangular element model,

and the reflectance map approximation error due to linearization. The first kind of error depends on the spacing h and can be reduced by using smaller spacings. The second kind of error can be reduced by approximating the reflectance map to the original nonlinear one as close as possible. Since the coefficients α_i and β_i are functions of local reference gradients (p_{0i}, q_{0i}), the choice of (p_{0i}, q_{0i}) is important. To determine an appropriate local reference gradient, we need good surface information. By doing the successive linearization, we can get more and more accurate surface gradients and surface values.

It is also important to point out two related issues. First, both $\mathbf{A}_g$ and $\mathbf{A}_l$ are singular so that we have to find ways to remove their singularities before the solution of the corresponding systems of equations. This will be detailed in Sections 4 and 5. Second, we do not have a convergence theory for the successive linearization procedure so far. However, it works for all our test problems in combination with the regularization technique described in Section 5.

4 Singularities of Stiffness Matrices

The stiffness matrix $\mathbf{A}$ is *sparse* since each nodal basis function $\phi_i(x, y)$ has a compact support and overlaps with only a finite number (≤ 6) of neighboring nodal basis functions. It is evident from (2.12) that $a_{i,j} = a_{j,i}$ so that $\mathbf{A}$ is *symmetric*. Besides, for any nonzero vector $\mathbf{z}$, we have

$$\mathbf{z}^T \mathbf{A} \mathbf{z} = \sum_{i=1}^{M_n} \sum_{j=1}^{M_n} z_i a_{i,j} z_j = 2 \int\int_{\Omega} [\sum_{i=1}^{M_n} \Phi_i z_i]^2 \, dx \, dy \geq 0, \tag{4.1}$$

so that $\mathbf{A}$ is *positive semidefinite*. However, $\mathbf{A}$ is singular and, consequently, the cost functional given in (2.11) does not have a unique minimum. In this section, we will investigate the reasons (or physical interpretations) for the singularity of $\mathbf{A}$ and discuss ways to remove the singularity. Let us start with the easier case, i.e. the stiffness matrix with global linearization. The general stiffness matrix with local linearization will then be examined in Section 4.2.

4.1 Stiffness Matrix with Global Linearization

One reason for $\mathbf{A}_g$ to be singular can be explained by the well known fact that we cannot determine the absolute heights of the object surface from its image since the brightness

of a surface patch is only determined by its gradient. According to Table 2, it is easy to verify that the row sum of $\mathbf{A}_g$ is zero for every row. Let $\mathbf{u}$ be the vector of 1's, i.e. $\mathbf{u} = [1, 1, \ldots, 1]^T$. We have $\mathbf{A}_g\mathbf{u} = 0$. Therefore, if $\mathbf{A}_g\mathbf{z}^* = \mathbf{b}$,

$$\mathbf{A}_g(\mathbf{z}^* + c\mathbf{u}) = \mathbf{b}, \tag{4.2}$$

where c is an arbitrary constant. This singularity is clearly the inherent limitation of the SFS problem. However, it can be removed by introducing an arbitrary reference nodal point.

Another reason for $\mathbf{A}_g$ to be singular can be understood by considering a single triangular patch. Suppose that one node is chosen to be the reference. There are still two nodal heights to be determined. Since the image irradiance equation $R(p, q) = E$, where E is the average brightness on the patch, only provides one constraint, the two equations consisting of these two relative nodal heights have to be linearly dependent. Hence, we conclude that the global stiffness matrix $\mathbf{A}_g$ has a null space of dimension at least two.

The singularity of $\mathbf{A}_g$ may also be resulted from special values of α and β. That is, if $\alpha = 0$, $\beta = 0$ or $\alpha = \beta$, we are not able to relate the gradient of a surface patch to its image intensity via (2.9). To see this, let us consider the triangular surfaces $S_k, \cdots, S_p$ over the triangular domains $T_k, \cdots, T_p$ as specified in Fig. 2. If $\alpha = 0$ (or $\beta = 0$), the image intensities of surfaces S_k and S_n (or S_l and S_o) are independent of p (or q) and thus the nodal height z_i. This phenomenon can also be explained by using Table 2. Note that

$$\begin{aligned} a_{i,j_1} = a_{i,j_4} = a_{i,j_2} = a_{i,j_5} = 0, \quad &\text{if } \alpha = 0, \\ a_{i,j_3} = a_{i,j_6} = a_{i,j_2} = a_{i,j_5} = 0, \quad &\text{if } \beta = 0. \end{aligned}$$

Similarly, we have

$$a_{i,j_1} = a_{i,j_4} = a_{i,j_3} = a_{i,j_6} = 0, \quad \text{if } \alpha = \beta.$$

It follows that if $\alpha = \beta$, the image intensities of surfaces S_m and S_p cannot be used to determine the nodal height z_i.

Motivated by the previous discussion, the sufficient conditions for the quadratic cost functional (2.11) to have a unique minimum can be states as follows.

Theorem 1 *The quadratic functional (2.11) with $\mathbf{A} = \mathbf{A}_g$ has a unique minimum, if the following two conditions are satisfied.*

1. *There are two neighboring nodal points whose heights are given.*

2. $\alpha \neq 0$, $\beta \neq 0$, *and* $\alpha \neq \beta$.

Proof. Without loss of generality, it is assumed that we know the two nodal heights of the triangular surface S_1 over the domain T_1 as depicted in Fig. 3. If z_1 and z_4 are given, since $\alpha \neq 0$ we can use the linearized reflectance map and the image irradiance equation to determine the value of z_5. Similarly, we can argue that z_1 (or z_4) can be obtained from given z_4 and z_5 (or z_1 and z_5). In turn, similarly the nodal heights of adjacent triangular surface patches S_2, S_6 can be determined since they share two common nodes with S_1. Thus, by this manner, all nodal values z_i can be uniquely determined. □

In practice, to achieve the first condition we may simply select two neighboring nodal points in an estimated plane region as the zero reference points. Since α and β are functions of the reference gradient (p_0, q_0), the second condition can be satisfied by choosing a proper value of (p_0, q_0). To give an example, consider the reflectance map (2.6) with the illumination direction $(p_s, q_s, 1)$ and $(p_0, q_0) = (0, 0)$. It is straightforward to see that the second condition is satisfied except $p_s = 0$, $q_s = 0$, or $p_s = q_s$. If the illumination direction happens to be these cases, we may move the reference gradient slightly around the origin so that the second condition is still satisfied.

4.2 Stiffness Matrix with Local Linearization

For general stiffness matrix **A**, we can also state the sufficient conditions for the quadratic cost functional (2.11) to have a unique minimum as follows.

Theorem 2 *The quadratic functional (2.11) has a unique minimum, if the following two conditions are satisfied.*

1. *There are two neighboring nodal points whose heights are given.*

2. $\alpha_i \neq 0, \beta_i \neq 0$ *and* $\alpha_i \neq \beta_i$ *for the linearized reflectance map defined on every domain* T_i.

Its proof is omitted since it is very similar to that of Theorem 1, Note that these conditions are not necessary but sufficient. For example, (2.11) may have a unique minimum even if the second condition is not satisfied. As before, the first condition can be satisfied by

selecting two neighboring nodes in an estimated plane region as the zero reference points. To attain the second condition, one possibility is to consider the selection of a proper reference gradient for each triangle patch. This can be achieved by checking the values of α_i, β_i and perturbing the reference point slightly whenever necessary. Another possibility is to introduce a regularization method to ensure the well-poseness of the SFS problem.

5 Regularization with Smooth Surface Constraint

Regularization is often achieved by adding some terms to the cost functional so that the regularized problem is well-posed [1], [24], [32]. It is preferable that we associate the additional term with some physical interpretation. As discussed in Section 4, the nonuniqueness of the minimum of the quadratic functional (2.11) is primarily due to some free nodal variables which may have arbitrary values without affecting the cost functional. Therefore, to obtain a unique solution we have to restrict these free variables by some constraint, say, the smoothness surface constraint, so that each node is related to its neighboring nodes through other means. Another reason to impose the smoothness constraint is to make our algorithm less sensitive to noises such as the sensor noise, the quantization noise, and the imperfect reflectance map model for real images.

To impose the smooth surface constraint, we define a new cost functional

$$\mathcal{E} = \mathcal{E}_b + \lambda \mathcal{E}_s. \tag{5.1}$$

where $\mathcal{E}_b$ is the original cost functional given by (2.11), λ is the smoothing factor, and the smoothing cost functional $\mathcal{E}_s$ can be chosen as a discrete version of

$$\mathcal{E}_{s,c} = \frac{1}{2} \iint (z_x^2 + z_y^2)\, dx\, dy, \tag{5.2}$$

or

$$\mathcal{E}_{s,c} = \frac{1}{2} \iint (z_{xx}^2 + 2z_{xy}^2 + z_{yy}^2)\, dx\, dy \tag{5.3}$$

and where the subscript c is used since the cost functions are expressed in terms of continuous variables x and y. The cost functionals (5.2) and (5.3), known as the *membrane* and the *thin plate* model, are proportional to the small deflection bending energy of a membrane and a thin plate and characterize surfaces of C^0 and C^1 continuity, respectively [29],[31]. Both models have the property of rotational invariance. Since the membrane model leads

to excessive flattened surfaces, the thin plate model is more frequently used for 2-D surface interpolation. We will focus on the thin plate model.

By discretizing (5.3), we obtain

$$\begin{aligned}\mathcal{E}_s &= \frac{1}{2h^2}\sum_{n_x}\sum_{n_y}[(z_{n_x+1,n_y} - 2z_{n_x,n_y} + z_{n_x-1,n_y})^2 + 2(z_{n_x+1,n_y+1} - z_{n_x,n_y+1} \\ &\quad - z_{n_x+1,n_y} + z_{n_x,n_y})^2 + (z_{n_x,n_y+1} - 2z_{n_x,n_y} + z_{n_x,n_y-1})^2]\end{aligned}$$

where h is the spacing. We can also express (5.3) in matrix form as

$$\mathcal{E}_s = \frac{1}{2}\mathbf{z}^T\mathbf{B}\mathbf{z}, \tag{5.4}$$

where $\mathbf{z}$ is the vector of nodal variables and $\mathbf{B}$ is the *smoothness* matrix which is sparse and symmetric. It is convenient to view the matrix-vector product $\mathbf{Bz}$ as a local nodal operator operating on a 2-D array. The local nodal operator is of the following stencil form

$$\mathbf{B}: \quad \frac{1}{h^2}\begin{vmatrix} & & 1 & & \\ & 2 & -8 & 2 & \\ 1 & -8 & 20 & -8 & 1 \\ & 2 & -8 & 2 & \\ & & 1 & & \end{vmatrix} \tag{5.5}$$

where h denotes the spacing between nodal points. Some special operator stencils for nodal points near the boundary are given in Fig. 6 [29],[31].

Substituting (5.4) and (2.11) into (5.1), we obtain

$$\mathcal{E} = \frac{1}{2}\mathbf{z}^T\mathbf{C}\mathbf{z} - \mathbf{b}^T\mathbf{z} + c, \tag{5.6}$$

where

$$\mathbf{C} = \mathbf{A} + \lambda\mathbf{B}.$$

The following theorem states the conditions for the existence and uniqueness of the minimum of (5.6).

Theorem 3 *The quadratic functional (5.6) has a unique minimum, if the following two conditions are satisfied.*

1. *There is one nodal point whose height is given.*

2. *If there exists at least one pair of coefficients (α_i, β_i) and (α_j, β_j) of the linearized reflectance maps over triangular domains T_i and T_j satisfying*

$$\frac{\beta_i}{\alpha_i} \neq \frac{\beta_j}{\alpha_j},$$

Proof. Since matrices $\mathbf{A}$ and $\mathbf{B}$ are both positive semidefinite, $\mathbf{C} = \mathbf{A} + \lambda\mathbf{B}$ with $\lambda > 0$ is also positive semidefinite. Thus, the minimum of the quadratic functional (5.6) exists. In the following, we want to show that, under condition 2, the positive semidefinite matrix $\mathbf{C}$ has only one eigenvalue equal to 0, which is associated with the eigenvector $(1, 1, \cdots, 1)$. As a consequence, if both conditions 1 and 2 are met, the minimum is unique.

It is sufficient to prove that $\mathbf{z}^T\mathbf{C}\mathbf{z} = 0$ only if $\mathbf{z}$ is a constant vector. Let us rewrite $\mathbf{z}^T\mathbf{C}\mathbf{z}$ as

$$\begin{aligned} \mathbf{z}^T\mathbf{C}\mathbf{z} &= \mathbf{z}^T(\mathbf{A} + \lambda\mathbf{B})\mathbf{z} \\ &= \mathbf{z}^T\mathbf{A}\mathbf{z} + \lambda\mathbf{z}^T\mathbf{B}\mathbf{z} \end{aligned}$$

Note that for nonzero λ, above equation is zero only when both $\mathbf{z}^T\mathbf{A}\mathbf{z}$ and $\mathbf{z}^T\mathbf{B}\mathbf{z}$ are zero since each term represents nonnegative energy. Recall that $\mathbf{z}^T\mathbf{B}\mathbf{z}$ is the discretized version of

$$\iint (z_{xx}^2 + 2z_{xy}^2 + z_{yy}^2)\, dx\, dy$$

which becomes zero only when $z(x, y)$ is a linear function over Ω. Hence, $\mathbf{z}^T\mathbf{B}\mathbf{z}$ is zero only when (p, q) is constant over the whole domain Ω. On the other hand, from (2.8) and (4.1), we know that the term $\mathbf{z}^T\mathbf{A}\mathbf{z}$ is zero only when $\sum_{i=1}^{M_n} \Phi_i z_i = 0$ or, equivalently, $\alpha_k p_k + \beta_k q_k = 0$ for every triangular domain T_k.

Thus, $\mathbf{z}^T\mathbf{C}\mathbf{z}$ is zero only when $\alpha_k p + \beta_k q = 0$ for every triangular domain T_k with respect to some constant (p, q). This implies all (α_k, β_k), $1 \leq k \leq M_t$, are orthogonal to a certain (p, q). Now, consider the case where two coefficients (α_1, β_1) and (α_2, β_2) are not linearly dependent. For $\mathbf{z}^T\mathbf{C}\mathbf{z}$ to be zero, we have to require $(p, q) = (0, 0)$. In addition, if one nodal point is assigned with value c, if follows that $\mathbf{z} = (c, c, \cdots, c)^T$. □

As discussed earlier, since only relative nodal heights have to be determined, we can pick an arbitrary point as reference to satisfy the first condition. The second condition usually holds in practice when we perform the local linearization of the reflectance map based on reconstructed surface obtained from the previous iteration. Therefore, the unique minimum

of (5.6) can be easily guaranteed. Note that the conditions in Theorem 3 are sufficient but not necessary. For example, in the initialization stage of the successive linearization scheme described in Section 3.2, even though condition 2 in Theorem 3 does not satisfy, the unique minimum of (5.6) is still guaranteed by conditions of Theorem 1.

We incorporate the regularization into the basic SFS algorithm to obtain a modified SFS algorithm, which is the algorithm used in our experiments described in Section 7.

Algorithm II: The modified SFS algorithm with regularization.

Initialization $(k = 0)$

Set the reference gradient (p_{0i}^0, q_{0i}^0) at triangle T_i to be $(0, 0)$, and construct the coefficient matrix $\mathbf{C}^0 = \mathbf{A}_g^0 + \lambda \mathbf{B}$ and the load vector $\mathbf{b}^0$. Solve $\mathbf{C}^0\mathbf{z}^0 = \mathbf{b}^0$ for nodal values $\mathbf{z}^0$.

Iterations $(k = 1, 2, \cdots)$

Set the reference gradient (p_{0i}^k, q_{0i}^k) at triangle T_i to be the local gradient determined by nodal values $\mathbf{z}^{k-1}$, and construct the corresponding coefficient matrix $\mathbf{C}^k = \mathbf{A}_l^k + \lambda \mathbf{B}$ and load vector $\mathbf{b}^k$. Solve $\mathbf{C}^k\mathbf{z}^k = \mathbf{b}^k$ for nodal values $\mathbf{z}^k$.
If $||\mathbf{z}^k - \mathbf{z}^{k-1}|| < \epsilon$, where ϵ is a predefined small quantity, then $\mathbf{z}^k$ is the desired solution. Otherwise, go to the next iteration.

6 Implementation Details

In this section, we discuss some implementation details: the triangulation of an image, the multigrid computational method, and the surface interpolation technique.

6.1 Triangulation of an Image

Recall that the construction of the load vector b requires the average intensity E_k over triangular domain $T_k, k = 1, \ldots, M_t$. To determine the average intensity E_k, we have to first partition discrete image points so that they are contained by triangular domains. Consider the case where the spacing between every adjacent nodal point is $h = 4$. Two possible partitioning schemes are depicted in Fig. 6, where the 2-D subscript notation is used for convenience. As shown in Figs. 6(a) and (b), nodal points belong to a subset of image points in the first scheme whereas they are located between the image points in the second scheme. The average intensity over a triangular domain can be obtained by summing up all the intensity values inside the triangle and some fractions of intensity values for nodes lying on the boundary of the triangle, and then divide it by the area of the triangle. The

average intensities $\overline{E}_k$ for schemes given in Fig. 6(a) and (b) can be computed, respectively, as

$$\begin{aligned}\overline{E}_k &= [(E_{n_x+2,n_y+1} + E_{n_x+3,n_y+1} + E_{n_x+3,n_y+2}) \\ &\quad +\frac{1}{2}(E_{n_x+1,n_y+1} + E_{n_x+2,n_y+2} + E_{n_x+3,n_y+3} + E_{n_x+1,n_y} \\ &\quad +E_{n_x+2,n_y} + E_{n_x+3,n_y} + E_{n_x+4,n_y+1} + E_{n_x+4,n_y+2} + E_{n_x+4,n_y+3}) \\ &\quad +\frac{1}{8}(E_{n_x,n_y} + 2E_{n_x+4,n_y} + E_{n_x+4,n_y+4})]/(h^2/2),\end{aligned}$$

and

$$\begin{aligned}\overline{E}_k &= [(E_{n_x+1,n_y} + E_{n_x+2,n_y} + E_{n_x+3,n_y} + E_{n_x+2,n_y+1} + E_{n_x+3,n_y+1} + E_{n_x+3,n_y+2}) \\ &\quad +\frac{1}{2}(E_{n_x,n_y} + E_{n_x+1,n_y+1} + E_{n_x+2,n_y+2} + E_{n_x+3,n_y+3})]/(h^2/2),\end{aligned}$$

where $h^2/2 = 8$ is the area of every triangle. It turns out that these two schemes give similar results. In all experiments reported in Section 7, the first scheme with $h = 1, 2, 4$ is used to triangularize the input image.

6.2 Multigrid Computational Method

The minimization of (5.6) is equivalent to the solution of the linear system $\mathbf{Cz} = \mathbf{b}$. Let $N = M_n$ be the number of unknowns. To solve the system $\mathbf{Cz} = \mathbf{b}$ with direct methods such as Gaussian elimination requires $O(N^3)$ operations and $O(N^2)$ storages. In contrast, if the system is solved by iterative methods such as preconditioned conjugate gradient (PCG) and multigrid methods, we only have to store the nonzero element of $\mathbf{C}$, which is proportional to $O(N)$. Among various iterative methods, the multigrid method is considered to be optimal in the sense that the number of operations required is only proportional to $O(N)$ [2], [8]. It has also been reported recently that the PCG method with the hierarchical basis preconditioner [28], [34] provides another efficient computational alternative, and is applicable to several computational vision problems such as surface interpolation. We use the multigrid method for our experiments, which will be briefly described below.

Let us rewrite the system $\mathbf{Cz} = \mathbf{b}$ as

$$\mathbf{C}_h\mathbf{z}_h = \mathbf{b}_h, \tag{6.1}$$

where the subscript h is added to denote explicitly that the discretization is performed with respect to a grid Ω_h of spacing h. To solve (6.1), most single-grid relaxation methods share

one common limitation. That is, the low frequency components of the error decay slowly whereas the middle or high frequency components of the error decay relatively fast. To accelerate the convergence rate of the error in the low-frequency region, one may consider to use a grid of larger spacing, say, Ω_{2h}. This idea leads naturally to a two-grid method.

In the two-grid method, the solution is obtained by combining the original fine grid solution with the interpolated correction of the smooth error component computed on the coarse grid. The two-grid method consists of the following three steps [18]:

1. *Presmoothing*: Select a relaxation operator S_h for solving (6.1). Typically, S_h is the Gauss-seidel or Jacobi relaxation methods. Thus, with a given initial estimate $\mathbf{z}_h^{(0)}$, one can apply the relaxation operations v_1 times and obtain $\mathbf{z}_h^{(1)}$ as the result. The corresponding residual is

$$\mathbf{r}_h = \mathbf{b}_h - \mathbf{C}_h \mathbf{z}_h^{(1)}.$$

2. *Coarse-grid correction*: The residual $\mathbf{r}_h$ can be projected to a coarser grid Ω_{2h} by using a restriction operator I_h^{2h},

$$\mathbf{r}_{2h} = I_h^{2h} \mathbf{r}_h.$$

   Then, we assume an exact solver is available on the grid Ω_{2h} so that one can determine the solution $\mathbf{z}_{2h}$ of the coarse-grid equations easily,

$$\mathbf{C}_{2h} \mathbf{z}_{2h} = \mathbf{r}_{2h}.$$

   Finally, we interpolate the coarse grid correction $\mathbf{z}_{2h}$ into the fine grid and add it to the fine grid solution such that

$$\mathbf{z}_h^{(2)} = \mathbf{z}_h^{(1)} + I_{2h}^h \mathbf{z}_{2h},$$

   wher I_{2h}^h denotes an interpolation operator.

3. *Postsmoothing*: Using $\mathbf{z}_h^{(2)}$ as the initial value, we apply the S_h iteration v_2 times and the resulting approximate solution is $\mathbf{z}_h^{(3)}$.

The main weakness of the two-grid method is that an exact solver is assumed available on the coarse grid. Although this is generally an unreasonable assumption, it is easy to see

that the problem on the coarse grid can itself been solved by a two-grid method. Thus, by applying the two-grid method recursively, we obtain a multigrid method, where successive sequence of coarser grids are employed until very few discretization points are involved that a direct solver is applicable. There are three types of multigrid methods according to the recursion patterns: the V-cycle, W-cycle and full multigrid algorithm [2], [8], [18]. The V-cycle multigrid is simple, yet works well for most linear problems.

In our experiment, we use the V-cycle multigrid scheme with the lexicographical Gauss-seidel relaxation method as the smoothing operator. We choose the full-weighting restriction restriction operator to transfer the residual from finer grids to coarser grids and the linear interpolation operator to transfer the correction from coarser grids to finer grids [18]. The coefficient matrices $\mathbf{C}_{2h}, \mathbf{C}_{4h}, \cdots, \mathbf{C}_{2^l h} \cdots \mathbf{C}_{2^L h}$, $(L = \log M_n^{1/2} - 1)$ for coarser grids are defined as

$$\mathbf{C}_{2^l h} = \mathbf{A}_{2^l h} + \lambda \mathbf{B}_{2^l h}, \qquad 1 \leq l \leq L,$$

where $\mathbf{A}_{2^l h}$ and $\mathbf{B}_{2^l h}$ are the stiffness and smoothing matrices on grid $\Omega_{2^l h}$, and can be constructed using the same approaches described in Sections 3 and 5, respectively.

6.3 Surface Interpolation Technique

When the number of nodes in the triangular element model is less than the number of pixels in the observed image, we have to perform surface interpolation based computed nodal heights to increase the resolution of the final result. This problem has been well studied in the context of surface reconstruction from stereo images [7], [28], [29], [31].

One well known scheme is the variational spline fitting algorithm [7], [29], [31]. By using this algorithm, we minimize the cost functional

$$\tilde{\mathcal{E}}_i = \tilde{\mathcal{E}}_d + \tilde{\lambda} \tilde{\mathcal{E}}_s, \tag{6.2}$$

with $\tilde{\mathcal{E}}_d$ and $\tilde{\mathcal{E}}_s$ defined below. Let $\tilde{z}_i$ be the desired height at point i, and z_i be the height computed through successive linearization if i happens to be a nodal point or 0 if i is a point to be interpolated. Besides, suppose that we first order the nodal points with $i = 1, \cdots, M_n$ and then the interpolated points $i = M_n + 1, \cdots, M_i$, where M_i is the total number of nodal and interpolated points. Then,

$$\tilde{\mathcal{E}}_d = \frac{1}{2} \sum_{i=1}^{M_n} (\tilde{z}_i - z_i)^2,$$

and

$$\tilde{\mathcal{E}}_s = \frac{1}{2}\tilde{\mathbf{z}}^T\tilde{\mathbf{B}}\tilde{\mathbf{z}},$$

where $\tilde{\mathbf{B}}$ is the smoothing matrix characterized by the local stencil

$$\tilde{\mathbf{B}}: \quad \frac{1}{h_i^2}\begin{vmatrix} & & 1 & & \\ & 2 & -8 & 2 & \\ 1 & -8 & 20 & -8 & 1 \\ & 2 & -8 & 2 & \\ & & 1 & & \end{vmatrix},$$

and stencils given in Fig. 6, where h_i denotes the spacing between adjacent interpolated points, Note that the $\tilde{\mathbf{B}}$ is similar to $\mathbf{B}$ given by (5.5) except their dimensions. The $\mathbf{B}$ is an $M_n \times M_n$ matrix whereas $\tilde{\mathbf{B}}$ is an $M_i \times M_i$ matrix.

Let us write (6.2) in matrix form as

$$\tilde{\mathcal{E}}_i = \frac{1}{2}\tilde{\mathbf{z}}^T\tilde{\mathbf{C}}\tilde{\mathbf{z}} - \tilde{\mathbf{b}}^T\tilde{\mathbf{z}} + \tilde{c}, \tag{6.3}$$

where

$$\tilde{\mathbf{C}} = \tilde{\mathbf{D}} + \tilde{\lambda}\tilde{\mathbf{B}},$$

and where $\tilde{\mathbf{D}}$ is a diagonal matrix with element 1 for the row corresponding to a known data point and 0 otherwise and $\tilde{\mathbf{b}}$ the zero-padded vector whose element is z_i for node i. The problem (6.3) can also be solved efficiently by the multigrid method [29], [30] or the PCG method with the hierarchical basis preconditioner [28].

From our numerical experience, the SFS algorithm with surface interpolation oftens leads to a surface much smoother than the original one (see Test Problem 4 in Section 7). Thus, in order to preserve the quality of the reconstructed surface, it is advisable to use $h = 1$ in the finest level. In other words, all image points are chosen to be nodal points of triangles. However, the surface interpolation scheme may be applicable for an image having large smooth surfaces and triangularized with nonuniform patches.

7 Experimental Results

We have tested our algorithm on several synthetic and real images. Unless specified explicitly, all nodal points of finest triangles coincide with the image points so that the finest grid has the spacing $h = 1$ and no interpolation is performed. For all test images, we use

the global linearization scheme in the first iteration. Then, a few more successive iterations, where the local linearization of the reflectance map is performed, are used. Since the reconstructed surface of the previous iteration serves as the initial estimate of the current iteration, the number of multigrid V-cycles required to reach a given error bound decreases as the successive linearization scheme proceeds. All the following experimental results are obtained after 10 successive iterations with around 20 multigrid V-cycles in total. Since the number of iterations required for satisfactory results is independent of the problem size N, the computational complexity is proportional to $O(N)$. Thus, it is very computationally efficient. To give a rough idea, for our testes images the multigrid V-cycles required to solve the 10 successive approximating problems are typically 10, 2, $1, 1, \cdots 1$, and it takes less than 20 minutes to reconstruct the surface for a 256 x 256 image on a Sun-4 sparc workstation.

To illustrate the quality of the reconstructed surface, we present the 3-D surface plot of the reconstructed surface as well as three shaded views which are produced with three illuminating directions: the orthogonal direction, the same and the opposite directions used to generate the original image [12]. The parameters of the tilt, slant, albedo and bias have been chosen experimentally in our test and reported in the figure captions. Note that several approaches to estimate these parameters have been recently studied [19], [21], [35].

Test Problem 1: Terrain

The tested image is a complicated synthetic image generated from the digital terrain model (DTM). Fig. 7(a) shows the 128 x 128 original DTM image of a wrinkled and steep surface (called the *alto-relievo* surface [12], [17]) of mountain area illuminated from northeast direction. The 3-D plot of the reconstructed surface and the shaded views of the reconstructed surface illuminated from the northeast, southwest and northwest directions are shown, respectively, in Fig. 7(b)-(e). Note that the original image in Fig. 7(a) and the reconstructed image in Fig. 7(c) are very similar, except that the reconstructed image is slightly smoother. The smoothing effect is due to the smoothing factor ($\lambda = 1000$). The reconstructed image is not too sensitive to the variation of λ in a certain range. However, the reconstructed surface tends to have visible local fluctuations not appearing in the original image for very small λ, and becomes too smooth for very large λ. The range of good λ values depends on the problem. This observation is in general true for all our test problems.

Test Problem 2: Lunar Surface

The tested image is a 128 x 128 real image of the lunar surface as shown in Fig. 8(a). The 3-D plot of the reconstructed surface and the shaded views of the reconstructed surface illuminated from the same, the opposite and the orthogonal directions with $\lambda = 1000$ are shown, respectively, in Fig. 8(b)-(e). It is interesting that even though the lunar surface may not be Lambertian, we obtained good estimate of the surface with the Lambertian model. In addition, we observe that, compared with the other three tested images, the reconstructed surface of this image is relatively insensitive to the variation of the parameters, i.e. tilt, albedo, slant, and bias, used in the reflectance map.

Test Problem 3: Pepper

The tested image is a 256 x 256 real image of a pepper which has a smooth surface as shown in Fig. 9(a). The 3-D plot of the reconstructed surface and the shaded views of the reconstructed surface illuminated from the same, the opposite and the orthogonal directions with $\lambda = 1000$ are shown, respectively, in Fig. 9(b)-(e). Note that this surface has only slowly changing slopes (called the *basso-relievo* surface [12], [17]). As a result, compared with the alto-relievo surface, it takes fewer iterations, say, 5 or 6, to achieve satisfactory results.

Test Problem 4: Lenna

The tested image is the 256 x 256 Lenna image as shown in Fig. 10(a). This image consists of both smooth and rapid varying areas with discontinuities in the surface orientation. The 3-D plot of the reconstructed surface and the shaded views of the reconstructed surface illuminated from the same, the opposite and the orthogonal directions with $\lambda = 2000$ are shown, respectively, in Fig. 10(b)-(e). This kind of test problem is considered difficult, since it contains several objects (possibly non-Lambertian) with different reflectivities and shadows. Besides, the lightening condition is not ideal. However, with the assumption of homogeneous reflectivity and Lambertian over the entire image, our algorithm still produces good results. Pay attention to the almost perfect reconstructed surface, especially in the regions of face, hat, hairs and the boundaries along the hat and hair lines. By comparing Fig. 10(c) with the original image in Fig. 10(a), we see that they are very similar.

We also test the idea of using a large triangular patches such as $h = 2$ and 4 in the finest grid and interpolating the result with the algorithm described in Section 6.3. The reconstructed surfaces and shaded views with the same illuminating directions are shown in Fig. 11. The result with $h = 1$, i.e. no interpolation, is shown in the same figure for

comparison. It is no surprise that larger triangular elements lead to smoother surfaces and degraded final results. Therefore, in order to get better results, we recommend to use finer triangular elements.

8 Conclusion

We presented a new efficient algorithm for the SFS problem in this research. Our algorithm recovers surface heights directly without any additional integrability constraint or artificial boundary assumption. It is based on the linear approximation of the reflectance map and a triangular element surface model, in which we express a surface as a linear combination of nodal basis functions. The nodal heights are determined by minimizing the cost functional, the total brightness error parameterized by nodal heights. This is equivalent to solving a large sparse linear system, to which the efficient multigrid method can be easily applied. We discussed the existence and uniqueness of the solution by investigating the properties of the stiffness matrix. A regularization technique using the smooth surface constraint was introduced to ensure the well-poseness of the SFS problem A successive linearization scheme was developed to obtain more and more correct surface and gradient information. Another advantage of our algorithm is that it maps naturally onto massively parallel architectures where each process covers one or several nodes.

Along our approach, there are several interesting problems worth further study. Real image in general may contain several objects having different irradiance models. Since no restrictive assumptions like the constant reflectivity and single illumination direction over the entire image are required for our algorithm, it seems that we may generalize our algorithm for complicated images by examining different reflectance map models in different segmented regions of the image. It is also natural to use our framework in the context of stereo images and develop a more efficient stereo SFS algorithm, which is under our current study.

Acknowledgments

The authors are grateful to Professor Rama Chellappa and Qinfen Zheng for helpful discussion.

References

[1] M. Bertero, T. A. Poggio, and V. Torre, "Ill-posed problems in early vision," *Proc. IEEE*, Vol. 76, No. 8, pp. 869–889, 1988.

[2] A. Brandt, "Multi-level adaptive solutions to boundary-value problems," *Mathmatics of Computation*, Vol. 31, pp. 333–390, April 1977.

[3] M. Brooks, "Surface normals from closed paths," in *Internal Joint Conference on Artificial Intelligence*, (Tokyo, Japan), pp. 98–101, August 1979.

[4] R. Courant and D. Hilbert, *Method of Mathematical Physics*, New York: John Willy & Sons, 1962.

[5] R. Frankot and R. Chellappa, "A method for enforcing integrability in shape from shading algorithm," *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 10, No. 4, pp. 439–451, 1989. Also in *Shape from Shading*, B.K. P. Horn and M. J. Brooks (eds.) 1989, MIT Press, Cambridge, MA.

[6] W. Grimson, *From Images to Surfaces*, Cambridge, MA: MIT Press, 1981.

[7] W. Grimson, "An implementationof a computational theory of visual surface interpolation," *Computer Vision, Graphics, and Image Processing*, Vol. 22, pp. 39–64, 1983.

[8] W. Hackbusch, *Multi-Grid Methods and Applications*, Berlin: Springer-Verlag, 1985.

[9] B. Horn, *Shape from shading: a method for obtaining the shape of a smooth opaque object from one view*, Ph.D. dissertation, Massachusetts Institute of Technology, Cambridge, MA, 1970.

[10] B. Horn, *Obtaining Shape from Shading Information*, Cambridge, Massachusetts: The MIT Press, 1975.

[11] B. Horn, "Understanding image intensities," *Artificial Intelligence*, Vol. 8, pp. 201–231, April 1977.

[12] B. Horn, "Height and gradient from shading," A.I. Memo 1105, Massachesetts Institute of Technology, Cambridge, MA, 1989.

[13] B. Horn and M. Brooks, "The variational approach to shape from shading," *Computer Vision, Graphics, and Image Processing*, Vol. 33, pp. 174–208, Nov. 1986. Also in *Shape from Shading*, B.K. P. Horn and M. J. Brooks (eds.) 1989, MIT Press, Cambridge, MA.

[14] B. Horn and R. Sjoberg, "Calculating the reflectance map," *Applied Optics*, Vol. 18, pp. 1770–1779, June 1979. Also in *Shape from Shading*, B.K. P. Horn and M. J. Brooks (eds.) 1989, MIT Press, Cambridge, MA.

[15] K. Ikeuchi and B. Horn, "Numerical shape from shading and occluding boundaries," *Artificial Inteligence*, Vol. 17, pp. 141–184, Aug. 1981. Also in *Shape from Shading*, B.K. P. Horn and M. J. Brooks (eds.) 1989, MIT Press, Cambridge, MA.

[16] C. Johnson, *Numerical Solutions of Partial Differential Equations by the Finite Element Method*, Cambridge: Cambridge University Press, 1987.

[17] J. Koenderink and van Doorn, "Photometric invarints related to solid shape," *Optica Acta*, Vol. 27, No. 7, pp. 981–996, 1980. Also in *Shape from Shading*, B.K. P. Horn and M. J. Brooks (eds.) 1989, MIT Press, Cambridge, MA.

[18] C.-C. J. Kuo and B. C. Levy, "Discretization and solution of elliptic PDEs - a digital signal processing approach," *Proc. IEEE*, Vol. 78, No. 12, pp. 1808–1841, 1990.

[19] C. Lee and A. Rosenfeld, "Improved methods of estimating shape from shading using light source coordinate system," *Artificial Intelligence*, Vol. 26, pp. 125–143, 1985. Also in *Shape from Shading*, B.K. P. Horn and M. J. Brooks (eds.) 1989, MIT Press, Cambridge, MA.

[20] D. Lee, "A provably convergent algorithm for shape from shading," in *Shape from Shading* (B. Horn and M. Brooks, eds.), pp. 349–373, Cambridge, Massachusetts: The MIT Press, 1989.

[21] A. Pentland, "Finding the illuminant direction," *Journal of Optical Society of America: A*, Vol. 72(4), pp. 448–455, April 1982.

[22] A. Pentland, "Local shading analisis," *IEEE Trans. Pattern Analysis and Machine Inteligence*, Vol. PAMI-16, pp. 170–187, March 1984. Also in *Shape from Shading*, B.K. P. Horn and M. J. Brooks (eds.) 1989, MIT Press, Cambridge, MA.

[23] A. Pentland, "Shape information from shading: a theory about human perception," in *Proc. of International Cof. on Computer Vision*, pp. 404–413, 1988.

[24] T. Poggio, V. Torre, and C. Koch, "Computational vision and regularization theory," *Nature*, Vol. 317, pp. 314–319, Sept. 1985.

[25] H. Schwartz, *Finite Element Methods*, Academic Press, 1988.

[26] G. Smith, "The relationship between image irradiance and surface orientation," in *IEEE Conference on Computer Vision and Pattern Recognition*, (Washington D.C.), pp. 404–413, June 1983.

[27] T. Strat, "A numerical method for shape from shading for a single image," Master's thesis, Massachusetts Institute of Technology, Cambridge, MA, 1979.

[28] R. Szeliski, "Fast surface intrpolation using hierachical basis functions," *IEEE Trans. Pattern Analysis and Machine Inteligence*, Vol. PAMI-12, pp. 513–528, June 1990.

[29] D. Terzopoulos, "Multilevel computational processes for visual surface reconstruction," *Computer Vision, Graphics, and Image Processing*, Vol. 24, pp. 52–96, 1983.

[30] D. Terzopoulos, "Image analysis using Multigrid relaxation methods," *IEEE Trans. Pattern Analysis and Machine Inteligence*, Vol. PAMI-8, pp. 129–139, March 1986.

[31] D. Terzopoulos, "The computation of visual surface representation," *IEEE Trans. Pattern Analysis and Machine Inteligence*, Vol. PAMI-10, pp. 417–438, July 1988.

[32] A. Tikhonov and V. Arsenin, *Method of Mathematical Physics*, Washington, DC: Winston and Wiley, 1977.

[33] R. Woodham, "A cooperative algorithm for determining surface orientation from a single view," in *Internal Joint Conference on Artificial Intelligence*, (Cambridge, MA), pp. 635–641, August 1977.

[34] H. Yserentant, "On the multi-level splitting of finite element spaces," *Numerical Mathematik*, Vol. 49, pp. 379–412, 1986.

[35] Q. Zheng and R. Chellappa, "Estimation of illumination direction, albedo, and shape from shading," *IEEE Trans. Pattern Analysis and Machine Inteligence*, 1991. To appear in July.

Figure Captions

Figure 1: A uniform triangulation of a square domain Ω.

Figure 2: A nodal basis function ϕ_i.

Figure 3: A simple example of a triangularized domain.

Figure 4: The 7-point stencil nodal operator.

Figure 5: The stencil forms of the nodal operators for B near the boundary.

Figure 6: Triangulation schemes of a discrete image

Figure 7: Results of our algorithm applied to the synthetic 128 x 128 DTM image: (a) original image generated with (tilt, slant, albedo, bias) = (30, 40, 230, 10), (b) reconstructed surface, (c)-(d) shaded views of (b) with the same, opposite and orthogonal illumination directions, with the parameter tilt equal to 30, 210 and 120, respectively.

Figure 8: Results of our algorithm applied to the real 128 x 128 lunar image: (a) original image, (b) reconsructed surface with (tilt, slant, albedo, bias) = (20, 60, 240, 10), (c)-(d) shaded views of (b) with the same, opposite and orthogonal illumination directions, with the parameter tilt equal to 20, 200 and 110, respectively.

Figure 9: Results of our algorithm applied to the real 256 x 256 Pepper image: (a) original image, (b) reconsructed surface with (tilt, slant, albedo, bias) = (40, 45 245 0), (c)-(d) shaded views of (b) with the same, opposite and orthogonal illumination directions, with the parameter tilt equal to 40, 220 and -50, respectively.

Figure 10: Results of our algorithm applied to the real 256 x 256 Lenna image: (a) original image, (b) reconsructed surface with (tilt, slant, albedo, bias) = (30, 60 245 3), (c)-(d) shaded views of (b) with the same, opposite and orthogonal illumination directions, with the parameter tilt equal to 30, 210 and -60, respectively.

Figure 11: Results of surface interpolation for the 256 x 256 Lenna image: (a) surface interpolated from 64 x 64 nodal points ($h = 4$), (b) surface interpolated from 128 x 128 nodal points ($h = 2$), (c) surface reconstructed without interpolation ($h = 1$), (d)-(f) are the shaded views of (a)-(c), respectively.

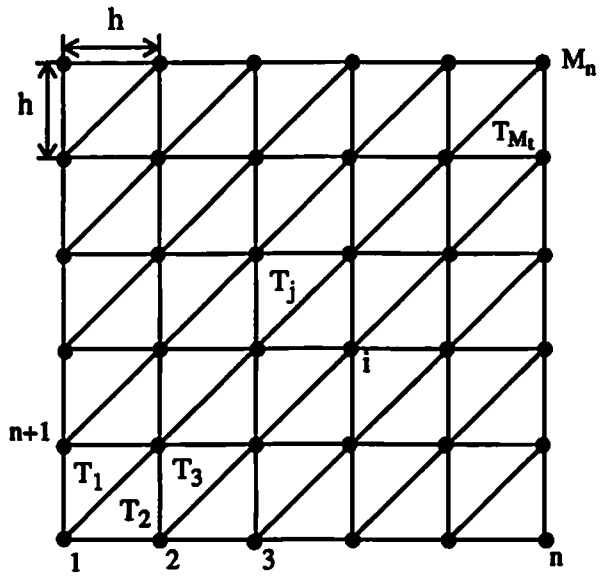


Fig. 1. A uniform triangulation of a square domain Ω.

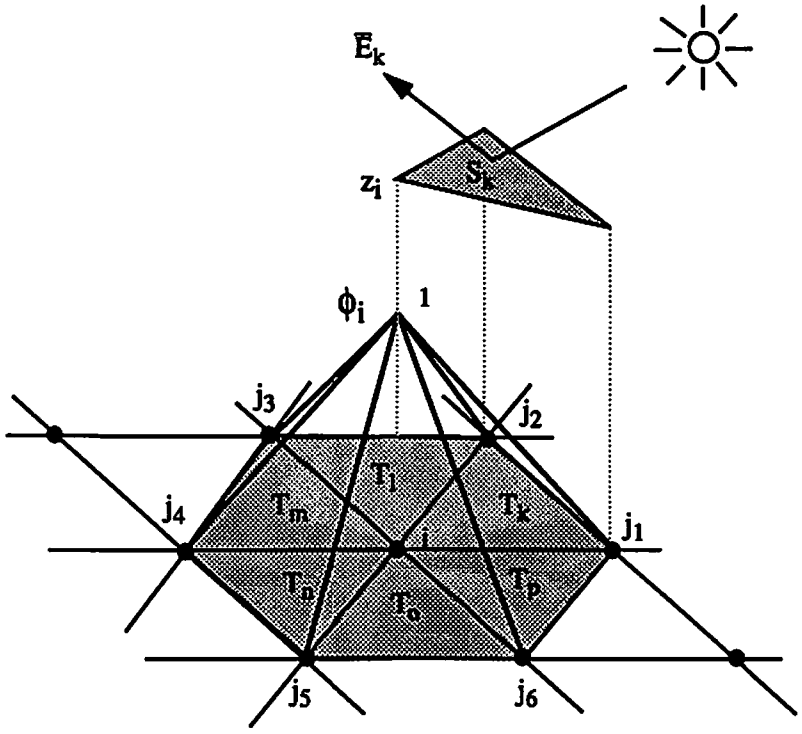


Fig. 2. A nodal basis function ϕ_i.

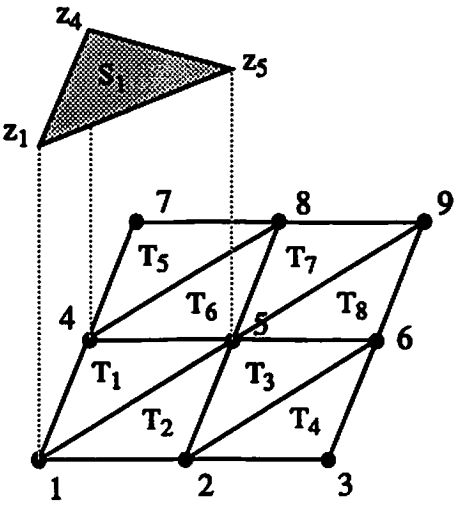


Fig. 3. A simple example of a triangularized domain.

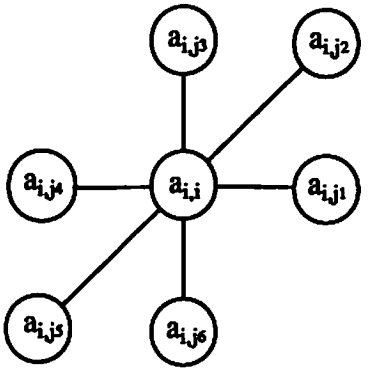


Fig. 4. The 7-stencil nodal operator.

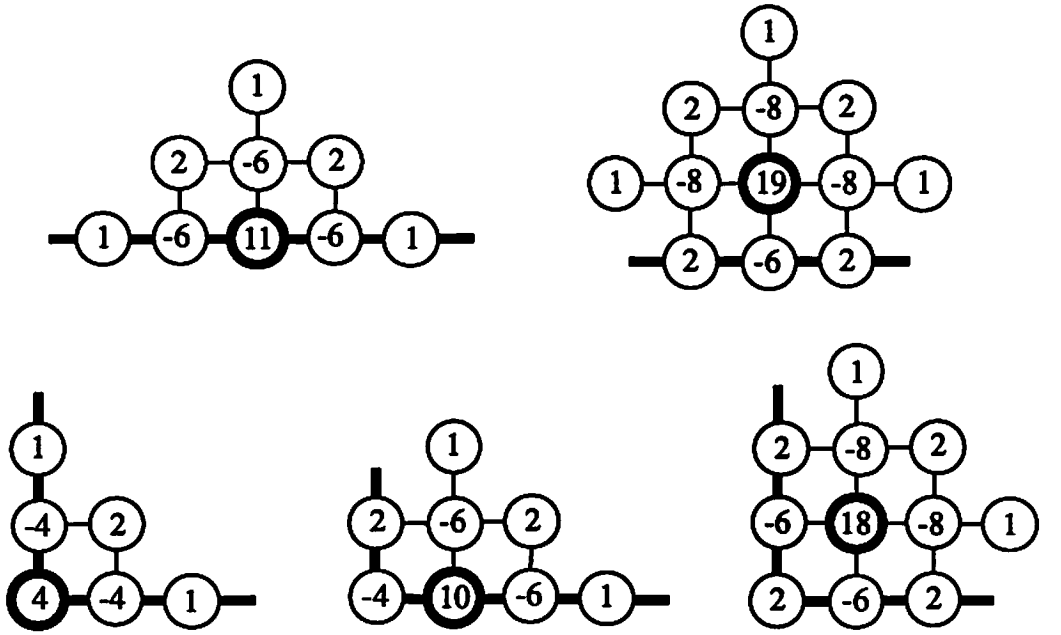
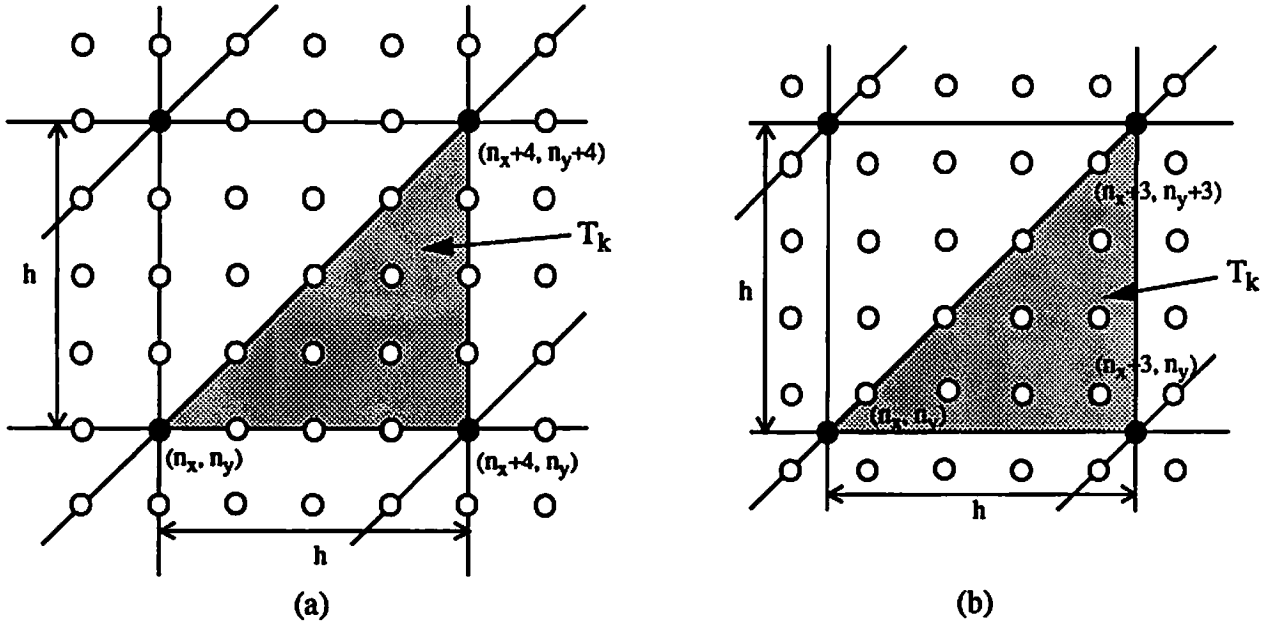


Fig. 5. The stencil forms of the nodal operators for B near the boundary

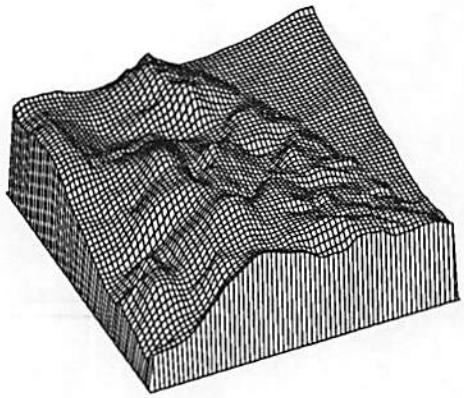


○ : image pixel

● : nodal point

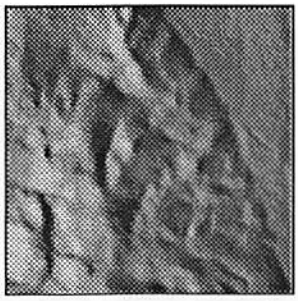
Fig. 6. Triangulation schemes of a discrete image

(a)



(b)

(c)

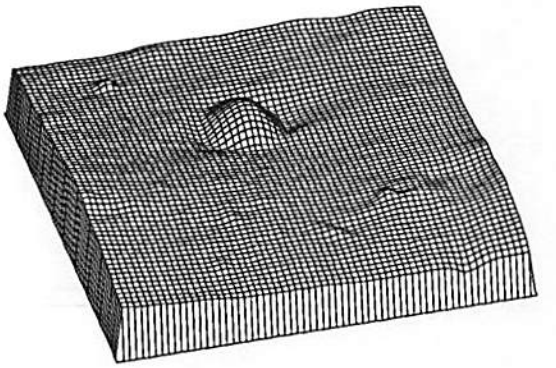


(d)

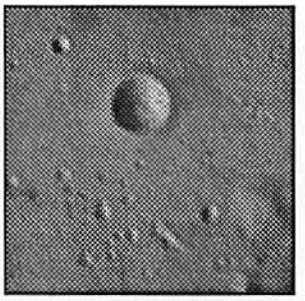
(e)

Figure 8: Results of our algorithm applied to the synthetic 128 × 128 DTM image: (a) original image generated with (tilt, slant, albedo, bias) = (30, 40, 230, 10), (b) reconstructed surface, (c)-(d) shaded views of (b) with the same, opposite and orthogonal illumination directions, with the parameter tilt equal to 30, 210 and 120, respectively.

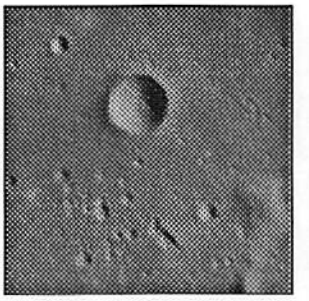
(a)



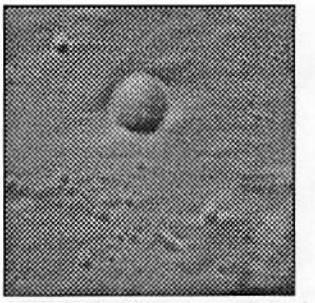
(b)



(c)

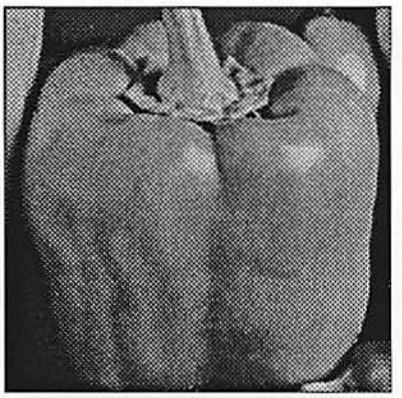


(d)

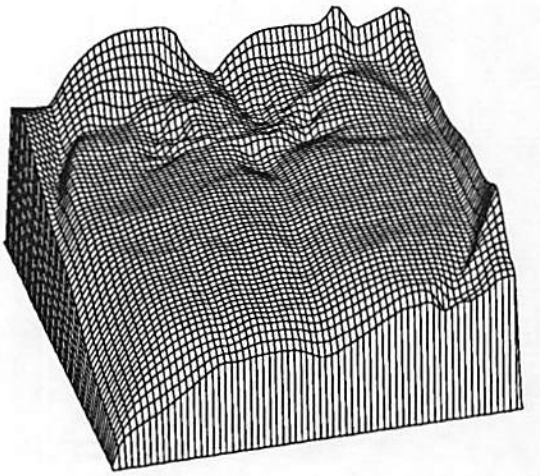


(e)

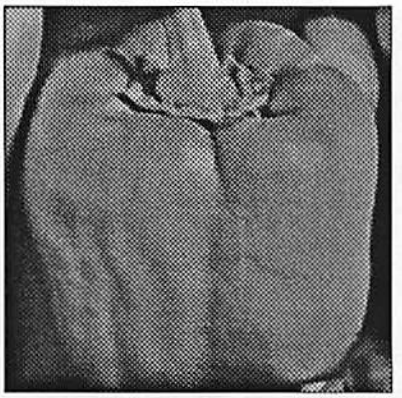
Figure 9: Results of our algorithm applied to the real 128 × 128 lunar image: (a) original image, (b) reconsructed surface with (tilt, slant, albedo, bias) = (20, 60, 240, 10), (c)-(d) shaded views of (b) with the same, opposite and orthogonal illumination directions, with the parameter tilt equal to 20, 200 and 110, respectively.



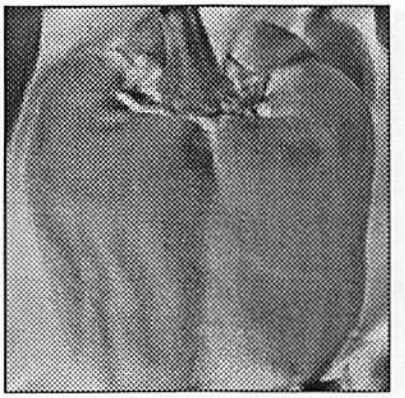
(a)



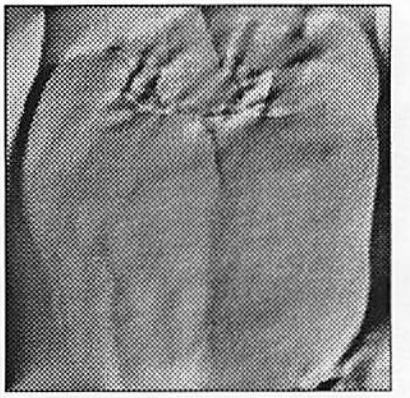
(b)



(c)



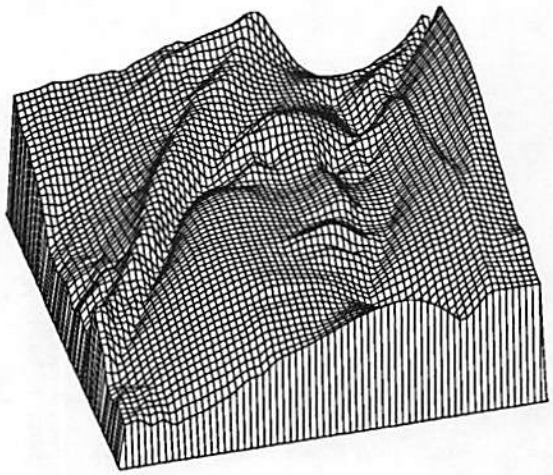
(d)



(e)

Figure 10: Results of our algorithm applied to the real 256 x 256 Pepper image: (a) original image, (b) reconsructed surface with (tilt, slant, albedo, bias) = (40, 45 245 0), (c)-(d) shaded views of (b) with the same, opposite and orthogonal illumination directions, with the parameter tilt equal to 40, 220 and -50, respectively.

(a)



(b)

(c)

(d)

(e)

Figure 11: Results of our algorithm applied to the real 256 x 256 Lenna image: (a) original image, (b) reconsructed surface with (tilt, slant, albedo, bias) = (30, 60 250 3), (c)-(d) shaded views of (b) with the same, opposite and orthogonal illumination directions, with the parameter tilt equal to 30, 210 and -60, respectively.

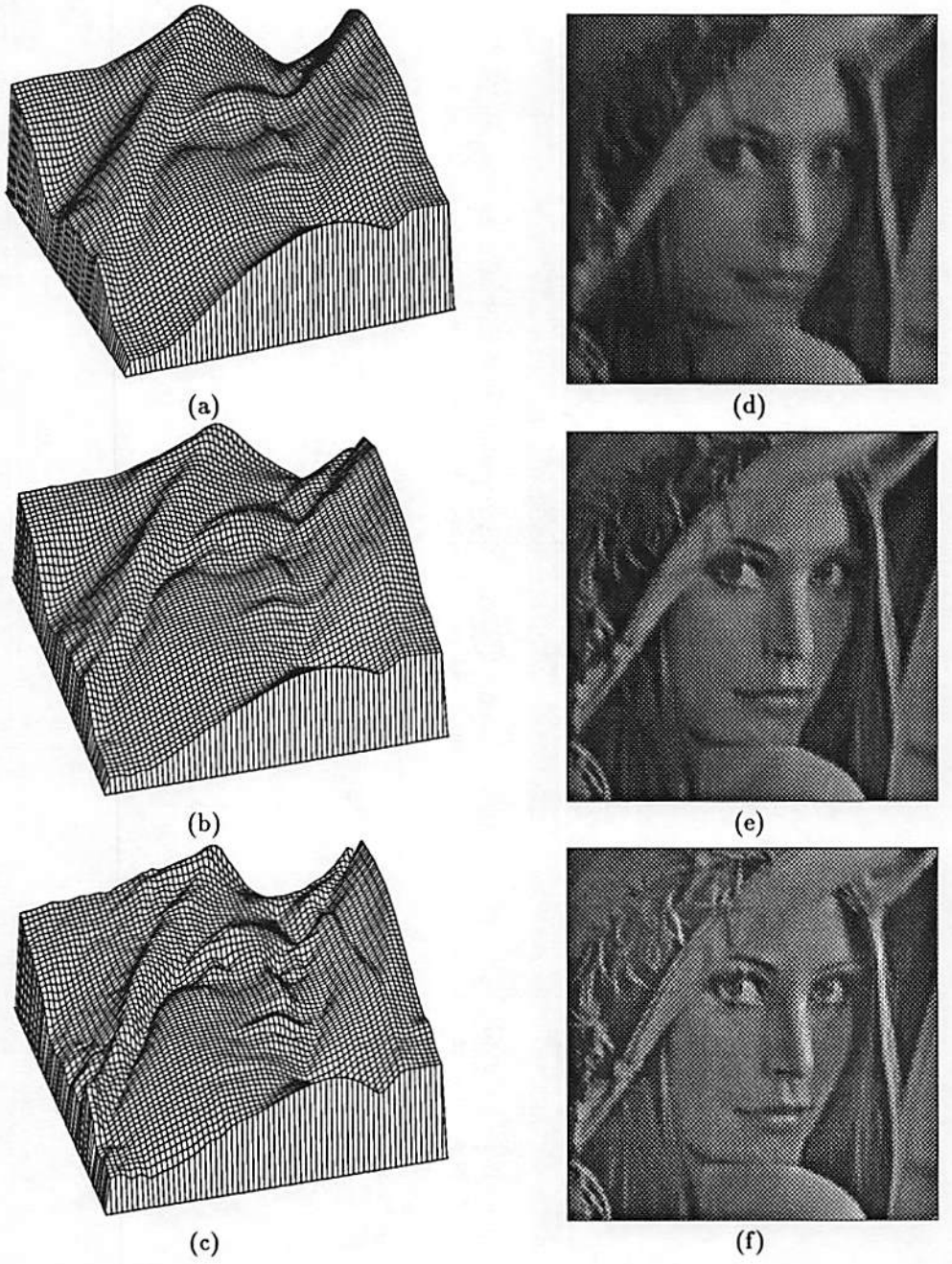


Figure 12: Results of surface interpolation for the 256 × 256 Lenna image:(a) surface interpolated from 64 × 64 nodal points ($h = 4$), (b) surface interpolated from 128 × 128 nodal points ($h = 2$), (c) surface reconstructed without interpolation ($h = 1$), (d)-(f) are the shaded views of (a)-(c), respectively.